International Society for Photogrammetry and Remote Sensing
Société Internationale de Photogrammétrie et de Télédétection
Internationale Gesellschaft für Photogrammetrie und Fernerkundung

THE INTERNATIONAL ARCHIVES OF THE PHOTOGRAMMETRY, REMOTE SENSING AND SPATIAL INFORMATION SCIENCES
ARCHIVES INTERNATIONALES DES SCIENCES DE LA PHOTOGRAMMÉTRIE, DE LA TÉLÉDÉTECTION ET DE L'INFORMATION SPATIALE
INTERNATIONALES ARCHIV FÜR PHOTOGRAMMETRIE, FERNERKUNDUNG UND RAUMBEZOGENE INFORMATIONSWISSENSCHAFTEN

VOLUME / VOLUME / BAND XXXVI PART / TOME / TEIL 3 / W49A

PIA07

Photogrammetric Image Analysis

Munich, Germany
September 19 – 21, 2007

Part A
Papers accepted on the basis of peer-reviewed full manuscripts

Editors
U. Stilla, H. Mayer, F. Rottensteiner, C. Heipke, S. Hinz

Organised by
Institute of Photogrammetry and Cartography,
Technische Universitaet Muenchen

in Cooperation with
ISPRS WG I/2 – SAR and LIDAR Systems
ISPRS WG III/2 – Surface Reconstruction
ISPRS WG III/4 – Automatic Image Interpretation for City-Modelling
ISPRS WG III/5 – Road Extraction and Traffic Monitoring
ISPRS WG IV/3 – Automated Geo-Spatial Data Acquisition and Mapping

ISSN 1682-1750

Cooperating ISPRS Working Groups
- WG I/2 – SAR and LIDAR Systems
- WG III/2 – Surface Reconstruction
- WG III/4 – Automatic Image Interpretation for City-Modelling
- WG III/5 – Road Extraction and Traffic Monitoring
- WG IV/3 – Automated Geo-Spatial Data Acquisition and Mapping

ISPRS Headquarters 2004-2008
c/o ORHAN ALTAN, ISPRS Secretary General
Division of Photogrammetry, Faculty of Civil Engineering
Istanbul Technical University
Ayazaga, 34469 Istanbul, Turkey
Phone: +90 212 285 38 10
FAX: +90 212 285 65 87
Email: oaltan@itu.edu.tr
ISPRS WEB Homepage: http://www.isprs.org

Published by
Institute of Photogrammetry and Cartography
Technische Universitaet Muenchen

Available from
GITC bv
P.O.Box 112
8530 AC Lemmer
The Netherlands
Tel: +31 (0) 514 56 18 54
Fax: +31 (0) 514 56 38 98
E-mail: mailbox@gitc.nl
Website: www.gitc.nl

Workshop Committees

Conference Chairs:

Uwe Stilla (Technische Universitaet Muenchen)
Helmut Mayer (Bundeswehr University Munich)
Franz Rottensteiner (University of Melbourne)
Christian Heipke (Leibniz Universität Hannover)

Cooperating Working Groups:

WG I/2 – SAR and LIDAR Systems
WG III/2 – Surface Reconstruction
WG III/4 – Automatic Image Interpretation for City-Modelling
WG III/5 – Road Extraction and Traffic Monitoring
WG IV/3 – Automated Geo-Spatial Data Acquisition and Mapping

Local Organizing Committee:

Stefan Auer (Technische Universitaet Muenchen)
Konrad Eder (Technische Universitaet Muenchen)
Karl Heiko Ellenbeck (University of Bonn)
Christine Elmauer (Technische Universitaet Muenchen)
Stefan Hinz (Technische Universitaet Muenchen)
Ludwig Hoegner (Technische Universitaet Muenchen)
Jens Leitloff (Technische Universitaet Muenchen)
Dominik Lenhart (Technische Universitaet Muenchen)
Manfred Stephani (Technische Universitaet Muenchen)

Program Committee:

Richard Bamler (DLR, Oberpfaffenhofen)
Claus Brenner (Leibniz Universität Hannover)
Joachim Ender (FGAN-FHR, Wachtberg)
Wolfgang Förstner (University of Bonn)
Paolo Gamba (University of Padua)
Marco Gianinetto (Politecnico di Milano)
Norbert Haala (Universität Stuttgart)
Lars Harrie (Lund University)
Christian Heipke (Leibniz Universität Hannover)
Olaf Hellwich (Technical University Berlin)
Stefan Hinz (Technische Universitaet Muenchen)
Jie Jiang (National Geomatics Center of China, Beijing)
Boris Jutzi (FGAN-FOM, Ettlingen)
Peter Krzystek (Munich University of Applied Sciences)
Helmut Mayer (Bundeswehr University Munich)
Chris McGlone (SAIC, Chantilly, VA)
Liqiu Meng (Technische Universitaet Muenchen)
Bryan Mercer (Intermap Technologies Corp., Calgary)
Franz Meyer (Alaska Satellite Facility, Fairbanks)
Moritz Neun (University of Zurich)
Kian Pakzad (Leibniz Universität Hannover)
Nicolas Paparoditis (IGN / MATIS, St. Mande)
Ingo Petzold (University of Zurich)
Ross Purves (University of Zurich)
Franz Rottensteiner (University of Melbourne)
Daniel Scharstein (Middlebury College)
Jochen Schiewe (University of Osnabrueck)
Monika Sester (Leibniz Universität Hannover)
Uwe Soergel (Leibniz Universität Hannover)
Uwe Stilla (Technische Universitaet Muenchen)
Charles Toth (Ohio State University, Columbus)
Markus Ulrich (MVTec Software GmbH, Munich)
Stephan Winter (University of Melbourne)

Preface

Automated extraction of topographic objects from remotely sensed data is an important topic of research in Photogrammetry, Remote Sensing, GIS, and Computer Vision. This joint conference of ISPRS working groups I/2, III/2, III/4, III/5, and IV/3, held at Technische Universitaet Muenchen (TUM), discussed recent developments, the potential of various data sources, and future trends both with respect to sensors and processing techniques in automatic object extraction. The focus of the conference lay on methodological research.

The conference addressed researchers and practitioners from universities, research institutes, industry, government organizations, and private companies. The range of topics covered by the conference is reflected by the cooperating ISPRS working groups:

- SAR and LIDAR Systems (WG I/2)
- Surface Reconstruction (WG III/2)
- Automatic Image Interpretation for City-Modelling (WG III/4)
- Road Extraction and Traffic Monitoring (WG III/5)
- Automated Geo-Spatial Data Acquisition and Mapping (WG IV/3)

Prospective authors were invited to submit a full paper of maximum 6 pages and we received 49 papers for review. The presented papers have undergone a rigorous “double blind” review process of full papers, with a rejection rate of 30%. Each paper was reviewed at least by three members of the program committee. Accepted papers (34) and one invited paper are published as printed proceedings in the IAPRS series as well as on CD labelled as "Part A". Only a subset of these papers could be presented orally due to the single track design of PIA07 and the generous time slots for intensive discussion.

Authors who intended to present application oriented work that was in particular suitable for interactive presentation were invited to submit an extended abstract. A group of the program committee selected 32 out of 55 contributions for presentation. Accepted contributions based on abstract review were invited to submit full papers which are published on CD labelled as "Part B".

In total, we received contributions from authors coming from 26 countries. The proceedings include 66 papers from authors coming from 19 countries. There were 7 oral sessions with altogether 20 papers and two interactive sessions where 46 papers were presented.

Finally, the editors wish to thank all contributing authors and the members of the Program Committee. In addition, we like to express our thanks to the Local Organising Committee, without whom this event could not have taken place. Konrad Eder and Christine Elmauer did a great job in arranging the event. Ludwig Hoegner was very helpful especially with the management of the ConfTool. The final word processing of all incoming manuscripts and the preparation of the CD by Dominik Lenhart is gratefully acknowledged. Jens Leitloff organised the internet connection during the PIA07 event.

Munich, August 2007

Uwe Stilla, Helmut Mayer, Franz Rottensteiner, Christian Heipke, Stefan Hinz

Contents

DATA DRIVEN RULE PROPOSAL FOR GRAMMAR BASED FACADE RECONSTRUCTION

Nora Ripperda and Claus Brenner

Institute of Cartography and Geoinformatics
Leibniz University of Hannover
{nora.ripperda, claus.brenner}@ikg.uni-hannover.de

KEY WORDS: facade modelling, building extraction, Markov Chain

ABSTRACT:

Today the demands on 3d models are steadily growing. At the same time, the extraction of man-made objects from measurement data is quite traditional. Often, the processes are still point based, with the exception of a few systems, which allow to automatically fit simple primitives to measurement data. The need to be able to automatically transform object representations, for example, in order to generalize their geometry, enforces a structurally rich object description. Likewise, the trend towards more and more detailed representations requires to exploit structurally repetitive and symmetric patterns present in man-made objects, in order to make extraction cost-effective. In this paper, we address the extraction of building facades in terms of a structural description. We extend our former work on facade reconstruction, which is based on a formal grammar to derive a structural facade description in the form of a derivation tree and uses a stochastic process based on reversible jump Markov Chain Monte Carlo (rjMCMC) to guide the application of derivation steps during the construction of the tree. We use measurements to improve the control of the rjMCMC process. This data driven approach reduces the number of false proposals and therefore the execution time.

1 INTRODUCTION

1.1 Motivation

The extraction of man-made objects from sensor data has a long history in research (Baltsavias, 2004). Especially for the modelling of 3D buildings, numerous approaches have been reported, based on monoscopic, stereoscopic, multi-image, and laser scan techniques. While most of the effort has gone into sensor-specific extraction procedures, very little work has been done on the structural description of objects.

Modelling structure though is very important for downstream usability of the data, especially for the automatic derivation of coarser levels of detail from detailed models.

Representing structure is not only important for the later usability of the derived data, but also as a means to support the extraction process itself. A fixed set of structural patterns allows to span a certain subspace of all possible object patterns, thus forms the model required to interpret the scene. Patterns can also guide the measurement process. Especially for man-made structures such as building facades, a large number of regularity conditions hold, which can be introduced into the measurement process as constraints.

Our aim is to extract facade elements from image and range data automatically. This paper extends our former work on the grammar based extraction of facade descriptions (Ripperda and Brenner, 2006) in which the grammar guides the generation of possible facade layouts using a reversible jump Markov Chain Monte Carlo (rjMCMC) process to explore solution space. The rjMCMC algorithm is used for other applications e.g. image segmentation as well. Tu et al. (2005) integrated generative and discriminative methods for image parsing. We present a way to derive distributions of facade attributes like the position of windows. These distributions are used for the rule proposal to evade the large number of wrong proposals which where so far only based on general prior knowledge on facades.

1.2 Related Work

Grammars have been extensively used to model structures. For modelling plants, Lindenmayer systems were developed by Prusinkiewicz and Lindenmayer (1990). They have also been used for modelling streets and buildings (Parish and Müller, 2001; Marvie et al., 2005). But Lindenmayer systems are not necessarily appropriate for modelling buildings. Buildings differ in structure from plants and streets, in that they don't grow in free space and modelling is more a partition of space than a growth-like process.

For this reason, other types of grammars have been proposed for architectural objects. Stiny and Gips (1972) introduced shape grammars which operate on shapes directly. The rules replace patterns at a point marked by a special symbol. Mitchell (1990) describes how grammars are used in architecture. The derivation is usually done manually, which is why the grammars are not readily applicable for automatic modelling tools.

Alegre and Dallaert (2004) use a stochastic context free attribute grammar to reconstruct facades from image data by applying horizontal and vertical cuts.

Wonka et al. (2003) developed a method for automatic modelling which allows to reconstruct different kinds of buildings using one rule set. The approach is composed of a split grammar, a large set of rules, which divide the building into parts, and a control grammar, which guides the propagation and distribution of attributes. During construction, a stochastic process selects among all applicable rules.

Dick et al. (2004) introduce a method which generates building models from measured data, i.e. several images. This approach is also based on the rjMCMC method. In a stochastic process, 3D models with semantic information are built. Mayer and Reznik (2006) also use a MCMC method for the facade reconstruction from images.

2 FACADE RECONSTRUCTION USING A GRAMMAR AND MCMC

2.1 The facade grammar

A grammar is used to model facade structure. The facade is presented by the derivation tree of the word of the language of the grammar, which corresponds to the facade. The grammar is built in a way that the derivation describes a recursive partition of space. We obtain a partition from the application of a derivation rule of the split grammar. A derivation tree represents the overall facade partitioning. Each node of this tree corresponds to one of the symbols of the grammar. There are two kinds of symbols, nonterminals and terminals. The terminal symbols represent facade geometry and cannot be subdivided further. Geometrically, nonterminals do not represent facade geometry directly but serve as containers, which hold other objects, represented in the derivation tree by nonterminal or terminal children.

Some of the containers imply that their children have identical properties while others don't (see figure 1). SYMMETRICFACADE indicates symmetries in the facade and can be replaced by SYMMETRICFACADESIDE which represents the left side and the mirrored right side of the facade and an optional SYMMETRICFACADEMIDDLE. Implicitly, left and right side have the same content. In contrast, FACADE implies nothing about its children. In figure 1 on the right hand side a FACADE is subdivided in two PARTFACADEs, the upper and lower part, that have no similarities.

Figure 1: Symbols with (left: SYMMETRICFACADE) and without (right: FACADE) implications to their children.

The start symbol is the symbol FACADE. Starting from it, the subdivision can be made by rules similar to the ones introduced by Wonka et al. (2003). The model is expressed as a derivation tree with FACADE being the root. Derivation rules have a left side, which consists of one symbol, and a right side, which may comprise several symbols in a certain spatial layout. As an example, a grammar rule splits FACADE into GROUNDFLOOR and PARTFACADE. Figure 2 shows two examples of the subdivision of facades. In both cases the facade is subdivided into GROUNDFLOOR and the upper floors represented by PARTFACADE. The GROUNDFLOOR is partitioned in different FACADEELEMENTS that contain a DOOR or a WINDOW each. The upper floors are modelled in different ways. In the first case it is a SYMMETRICPARTFACADE with an IDENTICALFACADEARRAY of WINDOWS inside. In the second case two different IDENTICALFACADEARRAYS with different types of WINDOWS are derived.

The model is described by a parameter vector θ which contains the derivation tree and the attributes of the symbols. E.g. the parameter vector of the configuration in figure 1 right is represented by the hierarchic structure

$$\theta = Facade(0, 0, w, h, (PartFacade(0, 0, w, h_s),$$
$$PartFacade(0, h_c, w, h - h_s))),$$

where w and h are the width and hight of the facade and h_s is the height of the split.

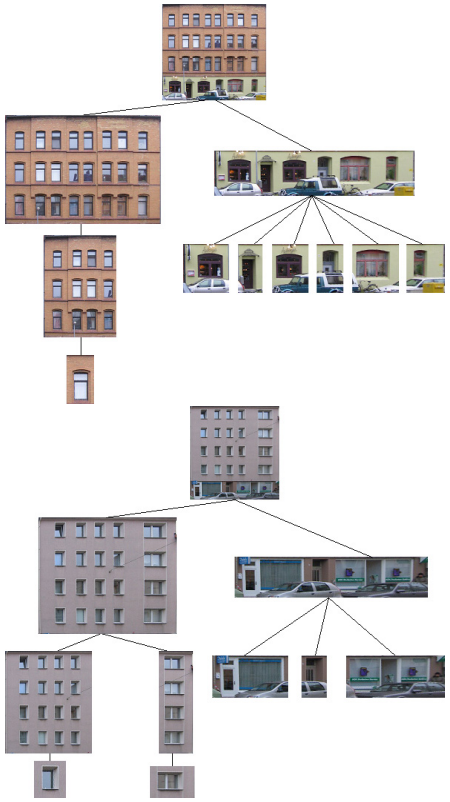


Figure 2: Example subdivision of facades.

2.2 Exploration of the Derivation Tree Using RjMCMC

We obtain the model of the facade using a stochastic process. We are searching for the model given by parameter vector θ with the highest probability $p(\theta|D_S D_I)$ under given scan (D_s) and image data (D_I) where the parameter vector θ encodes the current state of the derivation tree, including attributes.

We use a Markov Chain simulation to obtain the value of θ. This simulates a random walk in the space of θ. The process is led by a transition kernel $J(\theta_t|\theta_{t-1})$ and converges to a stationary distribution $p(\theta|D_S D_I)$.

The transition kernel $J(\theta_t|\theta_{t-1})$ assigns a probability to each rule and is made up from the commonness of the result in a dataset of facade images and some functions of the processed facade which will be described later. With the transition kernel in each iteration a rule is proposed. This is accepted with the acceptance probability

$$\alpha = \min(1, \frac{p(\theta_t|D_S D_I) \cdot J(\theta_{t-1}|\theta_t)}{p(\theta_{t-1}|D_S D_I) \cdot J(\theta_t|\theta_{t-1})}). \quad (1)$$

This depends on the unknown distribution $p(\theta_t|D_S D_I)$. Using Bayes' law, this is proportional to $p(D_S D_I|\theta_t) \cdot p(\theta_t)$, a product of likelihood and prior of the facade.The acceptance probability decides whether the rule is applied or not.

During the simulation, facade elements are added, deleted or changed. The first two operations change the number of elements on the facade and thus the dimension of the parameter vector θ.

The basic Markov Chain Monte Carlo method does not support dimension changes of θ and therefore we use rjMCMC instead. This method allows a change in the dimension of the parameter vector θ and thereby the number of facade elements can vary during the simulation. The rjMCMC method requires reversibility. For each change from state θ_1 to state θ_2 there must exist a reverse change from θ_2 to θ_1.

2.3 Jumping Distribution

A change is proposed depending on the jumping distribution $J_t(\theta_t|\theta_{t-1})$ that expresses the likelihood for each change. Each state change is in one of the following categories:

- Application of a split rule from the grammar. Facade elements are divided horizontally, vertically or in both directions and each part becomes a new symbol.The split indicates a change in the facade. If the ground floor differs from the rest of the facade, a split is applied.

  In fact, one grammar rule comprises a set of changes to the parameter vector θ, since the associated attributes have to be chosen, such as the number and size of children. For example a rule divides FACADE into several PARTFACADES, the general rule stands for all rules of this kind with any number and position of columns. The number of columns and their width is determined randomly.

- Changes in structure. Even after derivation of new containers according to the previous step, a second set of state changes allows to modify parameters, e.g. the number of rows or the position of the parting lines between rows.The same can be done starting from a child symbol. The position and extent of a symbol may change. In this case, the neighbour symbols, which are involved in the change, have to be changed as well.

- Replacement of symbols. This allows to interchange one symbol in the derivation tree by another symbol. In this case, the geometry stays the same, but the denotation changes. This is for example used if a FACADE is declared symmetric. FACADE → SYMMETRICFACADE

To ensure reversibility, each change can be applied from left to right and vice versa. This is a difference to the way split grammars are used, but is a requirement for the rjMCMC approach.

We have to define two kinds of distributions. The first one is the probability to choose a rule and the second one defines the parameter like the position of a split line or the number of windows. At the moment, the probability for rules is assigned manually depending on an assumed likelihood of the result. For example, a change FACADE → IDENTICALFACADEARRAY is more likely than FACADE → FACADEARRAY because facades build regular structures of similar elements. Some hints for the assumptions are taken from a database of facade images from Hannover.

To determine the parameter for the rules we need information about the distribution of colour or depth on the facade to control the split operation and to determine the distribution of the windows. Both depend on regularities and differences. For window grids we use autocorrelation and for splits a function based on a norm.

For splitting the facade into parts a change in colour or depth on a large part of the facade is needed. Other indications are breaks in regularity. The changes of colour and depth occur in different scales. We search for changes, which influence a great part of the facade, or separately changes caused by windows. Smaller artefacts in the facade may disturb the result. So we have different ways to score splits but in each we have to mask the small changes, which falsify the result. One way to suppress such unwanted changes is to use a scale space image (see figure 3). Another possibility is to cluster the facade depending on the colour value and in another step depending on the depth value. The results are shown in figure 4. From these images we can derive a probability for the splits. Therefore we compute the norm of two regions next to the split line (see figure 5), the upper region R_u and the lower region R_l. To evaluate the split line we compute the norm of the difference of both regions

$$||R_u - R_l||_2 = \sqrt{\sum_{x,y}(R_u(x,y) - R_l(x,y))^2},$$

where $R_u(x, y)$ is the rgb value at position (x, y).

The results are shown in figure 6. For a better visual understanding the original facade image is overlaied to the resulting graph. With the cluster image (blue line) we achieve better results than with the scaled image (red line) because on the scale image lines at top edges of windows are scored better than colour changes throughout the entire facade.

Figure 3: Image with lower scale maintains only large changes in facade structure.

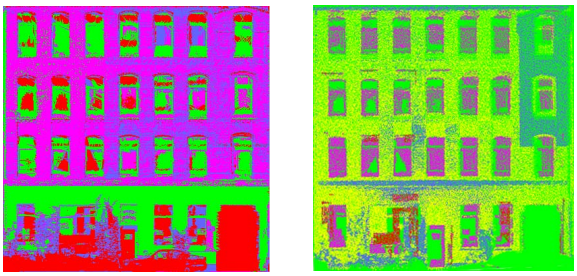


Figure 4: Clustered facade calculated by colour value and depth.

Using autocorrelation, we can predict the distribution of windows. We correlate the overlapping parts of the facade image and a copy of it which we shift horizontally resp. vertically. Figure 7 shows the result. In the case of a regular window grid the correlation values show peaks in a regular distance. The number of peaks is the number of window rows resp. columns plus one for the identical image plus one for the case when the overlap tends towards zero. In the example the horizontal correlation shows seven peaks because of the seven window columns plus two for identical and border cases. This pattern is not so clear for the vertical correlation because of the different ground floor.

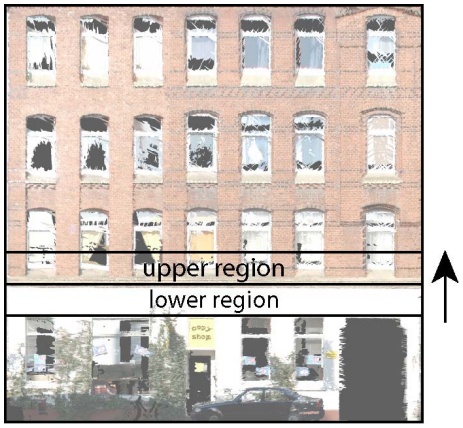


Figure 5: Two regions above and below the tested split line were moved over the facade.

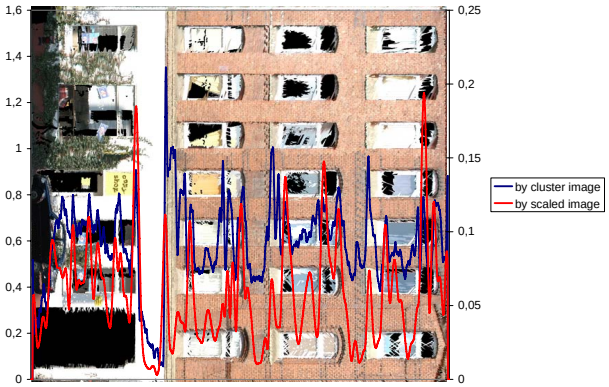


Figure 6: Facade image overlayed with the probability of splits evaluated by a scaled image and cluster image.

Another operation we can use to determine the window distribution is the planar segmentation of the scan. We use the segmentation descibed in (Dold and Brenner, 2004). Figure 8 shows the detected facade planes, but also sone smaller planes detected in the windows.

More information about the windows is given by point clouds from different standpoints. The laser beam penetrates the glass partly and is reflected from inside the building. If we compare two point clouds from different standpoints the differences mean windows or points, which can be seen only from one standpoint (see figure 9). The latter should not occur if we limit the point cloud to the facade.

To determine the differences we need the registration of the point clouds. This is the transformation matrix from the coordinate system of one standpoint to the one of another. We transform one point cloud in the coordinate system of the other and transform the cartesic coordinates into polar coordinates. The point cloud of one standpoint is stored as a raster addressed by polar and azimuth angle. Therefore with the received polar and azimuth angle the corresponding scan point can be read. A difference in the range value means a different point and therefore a window hypothesis. In figure 10 white pixel mean window hypothesis, black pixels have no corresponding pixel in the second scan and grey pixels are others.

2.4 Scoring Functions

The scoring functions affect the acceptance probability (eq. 1) in the term $p(D_S D_I|\theta_t) \cdot p(\theta_t)$ respectively $p(D_S D_I|\theta_{t-1}) \cdot p(\theta_{t-1})$.

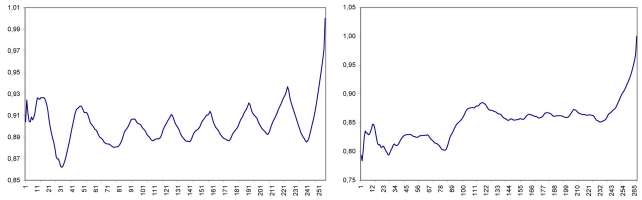


Figure 7: Autocorrelation coefficent in horizontal and vertical direction for the facade in figure 5.

Figure 8: Segmentation of the scan leads to different planes for facade and windows.

For the evaluation we use different methods, which can be divided into two groups. The first group contains methods, which test the general plausibility of the model of the facade corresponding to the factor $p(\theta_t)$. They depend on the alignment, the extent and the position of the facade elements. Here we use the same scoring functions as given in (Dick et al., 2004), which where described in (Ripperda and Brenner, 2006) as well.

The second group evaluates how good the model fits the data by comparing it to range and image data corresponding to the first term $p(D_S D_I|\theta_t)$. In any case, the evaluation functions return a score, which builds an acceptance probability for the change. To determine $p(D_S D_I|\theta_t)$ we have different possibilities which use scan and image data. We develope measures for depth and colour and use correlation, entropy and variance as well.

Depth In the first case, the fact that window points typically lie behind the facade is exploited. The average $\bar{d}$ of the facade depth is calculated. The variation of the points inside the proposed window constitutes the measure

$$\alpha_d = \frac{\sum |d - \bar{d}|}{A},$$

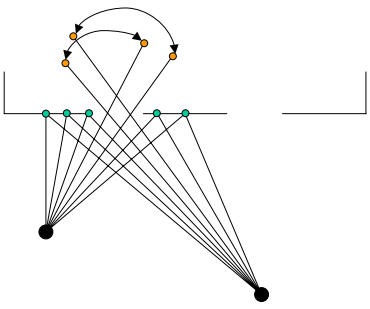


Figure 9: Principal sketch for window hypothesis.

Figure 10: Window hypothesis from different standpoints.

where A is the total number of points. α_d is typically close to zero for facade points and large for window points.

Colour In the second case, colour has been used since windows typically appear darker than the surrounding facade (or in some cases brighter because of reflections). Here we use the clustered images as well. We consider one region for the window and a boundary region (see figure 12 left). Let N_{max} be the number of pixels of the largest cluster inside the proposed window region, N_0 the number of unclassified pixels, A_{win} the area of the window, A_{bound} the area of the boundary and N_{bound} the number of pixels of the boundary which belong to the largest cluster inside the window. α_C gives a measure for the window.

$$\alpha_C = \frac{1 + \frac{N_{max}+N_0}{A_{win}} - \frac{N_{bound}}{A_{bound}}}{2}$$

In colour and depth cases, the information is used for the sub-

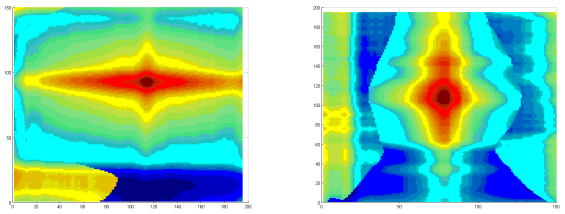


Figure 11: Score function with the depth cluster method (left) and color cluster method (right).

division of the facade. A proposed split of a container demands that the children have different properties.

Correlation In the case of similarity we use the correlation function (see sec. 2.3). For example, upon division into rows, the resulting row strips are correlated to determine whether the split is accepted or not.

Entropy To score arrays of windows we used entropy and variance for homogeneity measure. Entropy is

$$I = \sum_{i=1}^{n} \frac{|C_i|}{A} \log_2 \frac{A}{|C_i|},$$

where n is the number of clusters, A the total area and $|C_i|$ the number of points in the i-th cluster. We divide the facade with a mask like in figure 12, right, according to the proposed array of windows. Entropy respectivly variance are calculated for white and gray areas separately.

We test the entropy for different grid positions. The grid has six degrees of freedom but for a better visualisation we fix the number of grid points and the distance between them. The results are shown in figure 13 and 14 on the left hand side. In both figures the yellow surface is the score of the window part of the facade, the blue one of the boundary part. In the diagramm for entropy the boundary part produces the better result because the window part isn't as homogeneous as the facade without windows. To obtain a probability we use the boundary part. The maximum possible result is $\log_2 n$ so we normalize the function with this factor. The probability (see figure 13, right) is $\alpha_I = 1 - \frac{I}{\log_2 n}$.

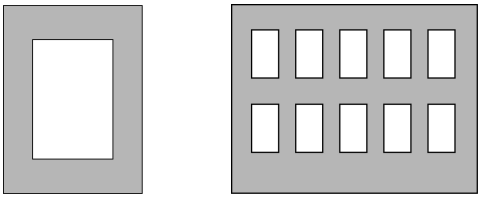


Figure 12: Mask for a single window (left) and an array of windows (right). The window area is white and the boundary area gray.

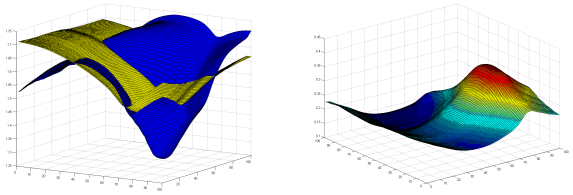


Figure 13: Entropy of window (blue) and boundary (yellow) and the probability derived from the window entropy.

Variance For another homogeneity measure, the variance, we use the original facade image because cluster labels are artificial numbers which would weight the differences arbitrarily. With this measure the boundary part of the facade leads to good results while the variance of the window part is higher than the one of the boundary part or mixed parts. Using $\alpha_V = 1 - \frac{\sqrt{V}}{255}$ we get a probability (see figure 14, right).

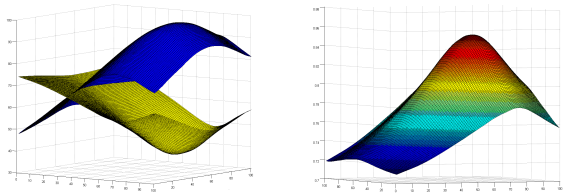


Figure 14: Variance of window (blue) and boundary (yellow) and the probability derived from the boundary variance.

3 RESULTS

We've tested the method on facades of dwelling houses. The input data are the point cloud and an orthophoto, which is generated with the RiScanPro software. The other required data are computed in a first step.

For a better understanding we first test parts of the modelling process separately. Therefore we cut out a single window. For this small data set we compute the score for each value and compare the result of the MCMC process (see figure 15) with the distribution given by the score function (see figure 11).

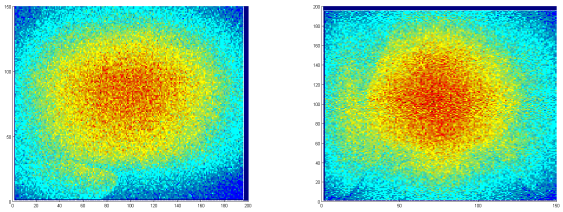


Figure 15: Sampled points with the depth cluster method (left) and color cluster method (right).

In the complete process, windows may not be modelled at the correct position in early derivations. Figure 16 shows two interim results of facade models. Not all modelled windows fit to the real ones. The reason is the assumption that the windows are arranged in a regular grid, which is not true. After further derivation steps the facade part is subdivided and single parts contain a regular grid. It is also possible that the grid pattern changes. Figure 17 shows a model of the right facade from figure 16. This final model uses a grid of window pairs and reproduces the facade in a better way.

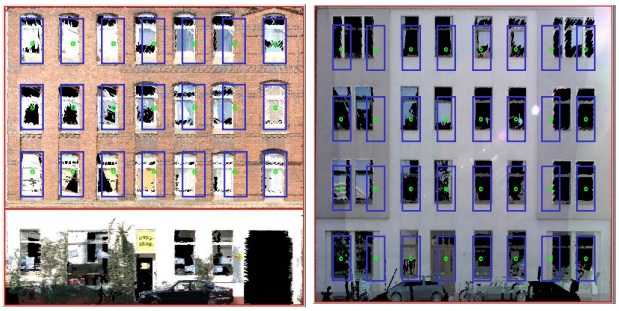


Figure 16: Displaced windows because of the wrong assumption of a regular grid.

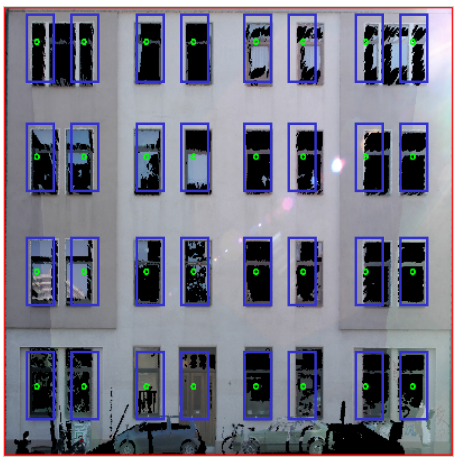


Figure 17: Facade model with a grid of window pairs.

4 CONCLUSION AND OUTLOOK

In this paper, we have presented an advancement of our previous work on grammar based facade reconstruction. It also combines the generation of artificial facade structures using grammars, and the reconstruction of facades using rjMCMC. Compared to existing grammar-based approaches, we gain the ability to reconstruct facades based on measurement data. Compared to existing rjMCMC approaches, by using a grammar, we obtain a hierarchical facade description and the ability to evaluate superstructures such as regularity and symmetry at an early stage, i.e., before terminal symbols such as WINDOW are instantiated.

We presented several measures to improve the rule proposals. These are no longer based only on general prior knowledge of facades. The measured facade influences the process not only in the scoring part but also in the proposal part.

Acknowledgements This work was done within in the scope of the junior research group “Automatic methods for the fusion, reduction and consistent combination of complex, heterogeneous geoinformation”, funded by the VolkswagenStiftung, Germany.

References

Alegre, F. and Dallaert, F., 2004. A probabilistic approach to the semantic interpretation of building facades. In: International Workshop on Vision Techniques Applied to the Rehabilitation of City Centers.

Baltsavias, E. P., 2004. Object extraction and revision by image analysis using existing geodata and knowledge: current status and steps towards operational systems. ISPRS Journal of Photogrammetry and Remote Sensing 58, pp. 129–151.

Dick, A., Torr, P., Cipolla, R. and Ribarsky, W., 2004. Modelling and interpretation of architecture from several images. International Journal of Computer Vision 60(2), pp. 111–134.

Dold, C. and Brenner, C., 2004. Automatic matching of terrestrial scan data as a basis for the generation of detailed 3D city models. In: International Archives of Photogrammetry and Remote Sensing, Vol. XXXV, Part B3, Proceedings of the ISPRS working group III/6, Istanbul, pp. 1091–1096.

Marvie, J.-E., Perret, J. and Bouatouch, K., 2005. The fl-system: a functional l-system for procedural geometric modeling. The Visual Computer 21(5), pp. 329 – 339.

Mayer, H. and Reznik, S., 2006. Building facade interpretation from uncalibrated wide-baseline image sequences. ISPRS Journal of Photogrammetry & Remote Sensing 61, pp. 371–380.

Mitchell, W. J., 1990. The Logic of Architecture : Design, Computation, and Cognition. Cambridge, Mass.: The MIT Press.

Parish, Y. and Müller, P., 2001. Procedural modeling of cities. In: E. Fiume (ed.), ACM SIGGRAPH, ACM Press.

Prusinkiewicz, P. and Lindenmayer, A., 1990. The algorithmic beauty of plants. New York, NY: Springer.

Ripperda, N. and Brenner, C., 2006. Reconstruction of faade structures using a formal grammar and rjmcmc. In: K. Franke, K.-R. Mller, B. Nickolay and R. Schfer (eds), Pattern Recognition, Proceedings of the 28th DAGM Symposium, pp. 750–759.

Stiny, G. and Gips, J., 1972. Shape Grammars and the Generative Specification of Painting and Sculpture. Auerbach, Philadelphia, pp. 125–135.

Tu, Z., Chen, X., Yuille, A. and Zhu, S., 2005. Image parsing: Unifying segmentation, detection, and recognition. International Journal of Computer Vision 63(2), pp. 113–140.

Wonka, P., Wimmer, M., Sillion, F. and Ribarsky, W., 2003. Instant architecture. ACM Transaction on Graphics 22(3), pp. 669–677.

REFINEMENT OF BUILDING FASSADES BY INTEGRATED PROCESSING OF LIDAR AND IMAGE DATA

Susanne Becker, Norbert Haala

Institute for Photogrammetry (ifp), University of Stuttgart, Germany
Geschwister-Scholl-Straße 24D, D-70174 Stuttgart
Forename.Lastname@ifp.uni-stuttgart.de

KEY WORDS: Three-dimensional, Point Cloud, Urban, LIDAR, Modelling, Façade Interpretation

ABSTRACT:

Urban models extracted from airborne data have to be refined for tasks like the generation of realistic visualisations from pedestrian viewpoints. Within the paper, terrestrial LIDAR data as well as façade imagery is used to increase the quality and amount of detail for the respective 3D building models. These models as they are available from airborne data collection provide a priori information, which can be integrated efficiently both for the georeferencing of the terrestrial data and the subsequent geometric refinement. After alignment of the terrestrial data to the given 3D model, window structures are first extracted approximately from the LIDAR point clouds. These structures are then further refined by 3D edges which are extracted from the overlapping façade images. Our modelling process applies a 3D object representation by cell decomposition, which can be used efficiently for building reconstruction at different scales.

1. INTRODUCTION

The area covering collection of urban models is usually based on the evaluation of aerial data like stereo images or LIDAR. The available algorithms provide 3D building representations which are sufficient for applications like simulations and visualisations at small or medium scale. However, for large scale applications like the generation of very realistic visualisations from pedestrian viewpoints, the quality and amount of detail for urban models from aerial data has to be improved. As an example, due to the viewpoint restrictions of airborne platforms, detailed information for the facades of the buildings frequently is not available. Thus, to improve the visual appearance of the buildings, terrestrial images are often mapped against the facades. However, this substitution of geometric modelling by real world imagery is only feasible to a certain degree. For instance, protrusions at balconies and ledges, or indentations at windows will disturb the visual impression of oblique views. Thus, geometric refinement is still necessary for a number of applications.

In order to enable the geometric modelling of building facades, either terrestrial laser scanning or image measurement can be used. Within this paper, the potential of these data sets for facade interpretation is demonstrated exemplarily for the extraction of window objects. In our opinion, an image based approach like it is for example presented by (Mayer & Reznik, 2006) considerably profits from the additional availability of densely sampled point clouds from terrestrial laser scanning. An integrated collection of such data sets is feasible by mobile systems, where a laser scanner and a camera are mounted on a car. Such a system was for example applied by (Früh & Zakhor, 2003) to generate textured meshes for visual representation of building facades. In our investigations, standard equipment consisting of a digital camera and a terrestrial laser scanner is used. To avoid data collection from scratch and to facilitate both the georeferencing and the modelling process, existing building models as they are provided from airborne data collection are closely integrated to all steps. Thus, we aim at a data driven geometric enrichment of building facades, whereas approaches using grammar based façade descriptions are more likely to focus on semantic modelling and interpretation (Brenner & Ripperda, 2006), (Alegre & Dallaert, 2004).

In contrast to other approaches based on building representations by constructive solid geometry (CSG) or boundary representation (B-Rep), we apply a representation of the buildings by cell decomposition. By these means, the problems to correctly generate topologically correct boundary representations can be avoided. The same holds true if geometric constraints such as meeting surfaces, parallelism and rectangularity have to be met. The formulation of such regularization conditions is also simplified if an object representation based on CSG is used. However, while CSG is widely used in computer aided design since it allows for powerful and intuitive modelling (Mäntylä, 1988), most visualization and simulation applications require the additional derivation of a boundary representation. While this is conceptually easy, its correct and efficient implementation can be difficult. Problems can arise from error-prone measurements, limited numerical precision and unstable calculation of intersections.

These problems are facilitated by the concept of cell decomposition. Similar to CSG, complex solids are described by a combination of relatively simple, basic objects in a bottom up fashion. In contrast to CSG, which combines simple primitives by means of regularized Boolean set operators, decomposition models are limited to adjoining primitives. Since the basic primitives must not intersect, they are thus 'glued' together to get the final model. In this sense, cell decomposition is similar to a spatial occupancy enumeration, where the object space is subdivided by non overlapping cubes of uniform size and orientation. Nevertheless, cell decompositions are based on a variety of basic cells, which may be any objects that are

topologically equivalent to a sphere i.e. do not contain holes. This allows for a simplified combination of the respective elements, while the disadvantages of exhaustive enumeration like large memory consumption and the restricted accuracy of the object representation can be avoided.

Since it is a prerequisite for further processing, the georeferencing process of the collected images and LIDAR data is described in Section 2. For this purpose, the collected data sets are aligned to the existing buildings from airborne data collection. The geometric refinement of the facades presented in Section 3 is implemented as a two-step approach. In order to integrate window objects to the existing coarse building model, cell decomposition is used. First, the windows and doors are modelled from the terrestrial LIDAR data, while the window frames are further refined by photogrammetric analysis of the images in a subsequent step.

2. DATA PREPARATION AND ALIGNMENT

The combined evaluation of the terrestrial LIDAR and image data requires the co-registration of the different data sets as a first processing step. The alignment of single images to a triangulated 3D point cloud can for example be realised based on corresponding linear primitives provided by a suitable edge detection process (Haala & Alshawabkeh, 2006). In our application, approximate geometry of the respective buildings is already available and can therefore be used to facilitate the georeferencing process. The quality and amount of detail of this data set is typical for such 3D models, like they are available area covering for a number of cities. Our exemplary 3D city model, which is maintained by the City Surveying Office of Stuttgart, features roof faces collected semi-automatically by photogrammetric stereo measurement. In contrast, the outlines of the buildings were captured by terrestrial surveying. Thus, the horizontal position accuracy of façade segments, which were generated by extrusion of this ground plan, is relatively high, despite the fact that they are limited to planar polygons.

2.1 Georeferencing of LIDAR data

During the collection of the 3D point clouds, a low-cost GPS and a digital compass were mounted on top of the used HDS 3000 laser scanner to allow for a direct georeferencing of the terrestrial scans. This approximate solution is then refined by an automatic registration of the laser scans against the 3D building model using a standard iterative closest point (ICP) algorithm (Böhm & Haala, 2005).

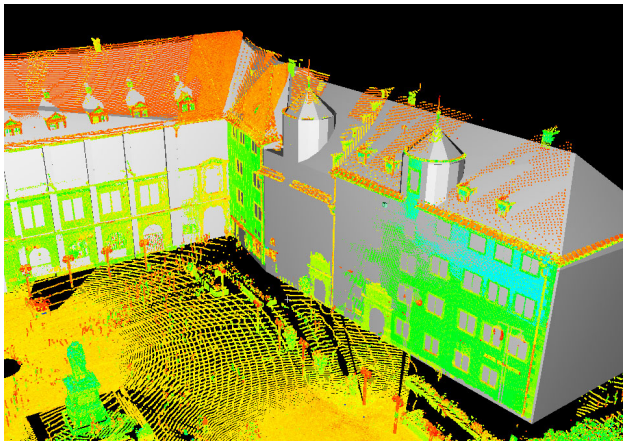


Figure 1: 3D point cloud from laser scanning aligned with a virtual city model.

As it is demonstrated in Figure 1, after this step the 3D point cloud and the 3D city model are available in a common reference system. Thus, relevant 3D point measurements can be selected for each building façade by a simple buffer operation. These 3D points are then transformed to a local coordinate system as defined by the façade plane. Figure 2 shows the resulting point cloud, which has an approximate spacing of 4cm.

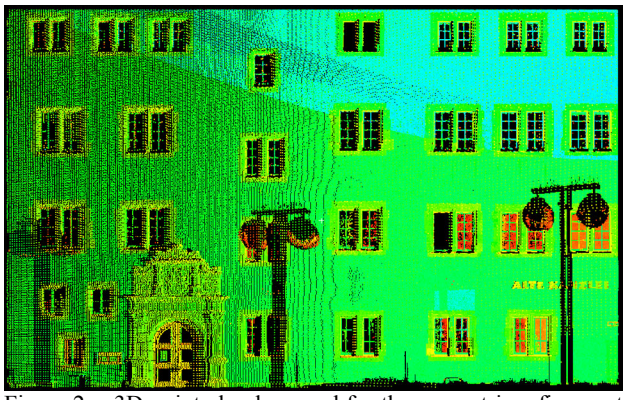


Figure 2. 3D point cloud as used for the geometric refinement of the corresponding building façade.

Since the LIDAR measurements are more accurate than the available 3D building model, the final reference plane is determined from the 3D points by a robust estimation process. After mapping of the 3D points to this reference plane, further processing can be simplified to a 2.5D problem. As an example, while assuming that the refined geometry of the façade can be described sufficiently by a relief, the differences between the measured 3D laser points and the given façade polygon can be interpolated to a regular grid.

2.2 Alignment of image data

Image orientation is the first step within the photogrammetric 3D modelling. Usually bundle adjustment is the method of choice if accurate orientation parameters are to be estimated. The determination of initial orientation parameters by spatial resection requires control points that can be obtained from the images and the LIDAR point cloud. Additionally, tie points are necessary for connecting the images. In the recent past, much effort has been made to develop approaches that automatically extract such tie points from images of different types (short, long, and wide baseline images) (Remondino & Ressl, 2006). While matching procedures based on cross-correlation are well suited for short baseline configurations, images with a more significant baseline are typically matched by means of interest points. However, these techniques would fail in case of wide baseline images acquired from considerably different viewpoints. This is due to big perspective effects that are caused by the large camera displacement. Points and corners cannot be reliably matched. Thus, interest point operators have to be replaced by region detectors and descriptors. As an example, the Lowe operator (Lowe, 2004) has been proved to be a robust algorithm for wide baseline matching (Mikolajczyk & Schmid, 2003).

Figure 3 shows the image data from a calibrated camera (NIKON D2x Lens NIKKOR 20mm). For the automatic provision of tie points the SIFT (scale invariant feature transform) operator has been applied to extract and match keypoints. Wrong matches are removed by a RANSAC based estimation (Fischler & Bolles, 1981) of the epipolar geometry using Nister's five point algorithm (Nister, 2004).

Figure 3. Image data for photogrammetric modelling.

The control points for the final bundle adjustment, which is performed with the Australis software package, are measured manually in the images and the 3D laser points.

3. FAÇADE RECONSTRUCTION

The reconstruction algorithm presented in this paper is a two-step approach based on terrestrial LIDAR and image data. It aims at the geometric facade refinement of an existing coarse building model by the integration of window objects. At first, cell decomposition is used to model windows and doors from the LIDAR data. In a second step, the window frames are further refined by photogrammetric analysis of the images.

3.1 Façade Refinement By Terrestrial LIDAR

The idea of the first part of our reconstruction algorithm is to segment a 3D object with a flat front face into 3D cells. Each 3D cell represents either a homogeneous part of the façade or a window area. Therefore, they have to be differentiated based on the availability of measured LIDAR points. After this classification step, window cells are eliminated while the remaining façade cells are glued together to generate the refined 3D building model. The difficulty is finding planar delimiters from the LIDAR points that generate a good working set of cells. Since our focus is on the reconstruction of the windows, the planar delimiters have to be derived from the 3D points that were measured at the window borders. These points are identified by a segmentation process.

3.1.1 Cell Generation

Point cloud segmentation. As it is visible for the façade in Figure 2, usually fewer 3D points are measured on the façade at window areas. This is due to specular reflections of the LIDAR pulses on the glass or points that refer to the inner part of the building and were therefore cut off in the pre-processing stage. If only the points are considered that lie on or in front of the façade, the windows will describe areas with no point measurements. Thus, our point cloud segmentation algorithm detects window edges by these no data areas. In principle, such holes can also result from occlusions. However, this is avoided by using point clouds from different viewpoints. In that case, occluding objects only reduce the number of LIDAR points since a number of measurements are still available from the other viewpoints.

During the segmentation process, four different types of window borders are distinguished: horizontal structures at the top and the bottom of the window, and two vertical structures that define the left and the right side. For instance, the edge points of a left window border are detected if no neighbour measurements to their right side can be found in a pre-defined search radius at the façade plane. The search radius should be set to a value a little higher than the scan point distance on the façade. The extracted edge points are shown in Figure 4. While most of the edge points can be correctly identified this way, the algorithm often fails to find points at window corners. However, this is not a real problem, as long as there are enough points to determine the window borders. For this purpose, horizontal and vertical lines are estimated from non-isolated edge points. The resulting set of window lines is depicted in Figure 5.

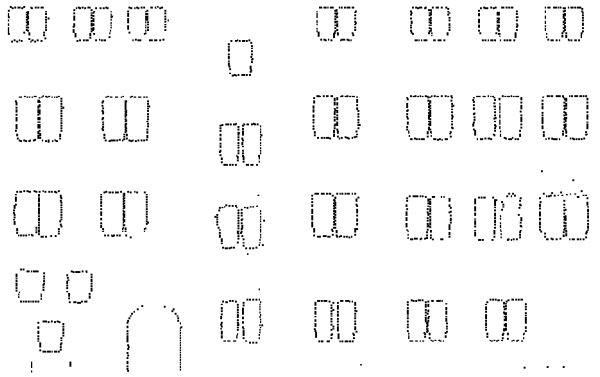


Figure 4. Detected edge points at horizontal and vertical window structures.

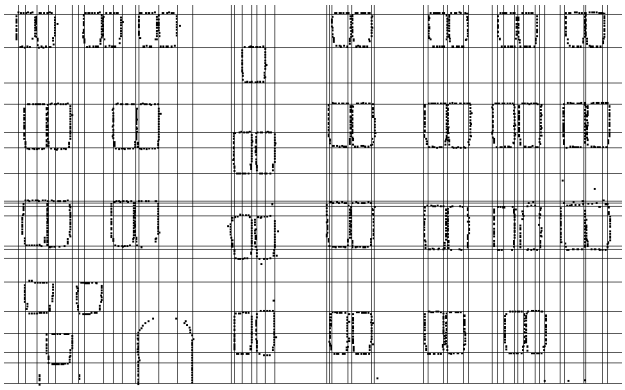


Figure 5. Detected horizontal and vertical window lines.

Spatial-Partitioning. Each boundary line defines a partition plane, which is perpendicular to the building façade. For the determination of the window depth, an additional partition plane is estimated from the LIDAR points measured at the window crossbars. These points are detected by searching a plane parallel to the façade, which is shifted in its normal direction. The set of all partition planes provides the structural information for the cell decomposition process. Therefore, it is used to intersect the existing building model producing a set of small 3D cells.

3.1.2 Classification of 3D cells

In a next step, the generated 3D cells have to be classified into building and non-building fragments. For this purpose, a 'point-availability-map' is generated. It is a binary image with low resolution where each pixel defines a grid element on the façade. The optimal size of the grid elements is a value a little higher than the point sampling distance on the facade.

As it can be seen in Figure 6, black pixels are raster elements where LIDAR points are available, while white pixels represent grid elements with no 3D point measurements. Of course, the already extracted edge points in Figure 4 and the resulting structures in Figure 5 are more accurate than the rasterized point-availability-map. However, this limited accuracy is acceptable since the binary image is only used to classify the 3D cells, which are already created from the detected horizontal and vertical window lines. This is implemented by computing

the ratio of façade to non-façade pixels for each generated 3D cell.

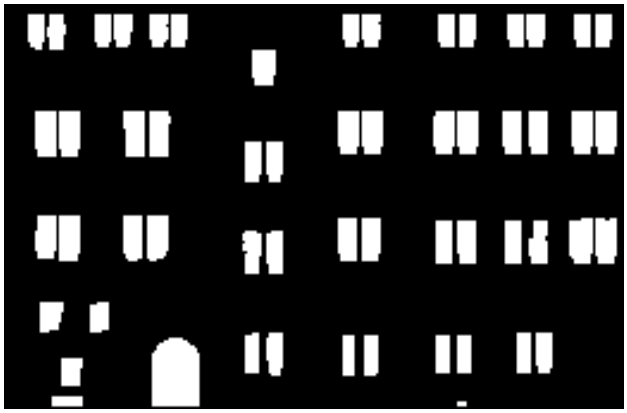


Figure 6. Point-availability-map.

As a consequence of the relative coarse rasterization and the limited accuracy of the edge detection, the 3D cells usually do not contain facade pixels or window pixels, exclusively. Within the classification, 3D cells including more than 70% façade pixels are defined as façade solids, whereas 3D cells with less than 10% façade pixels are assumed to be window cells. These segments are depicted in Figure 7 as grey (façade) and white (window) cells.

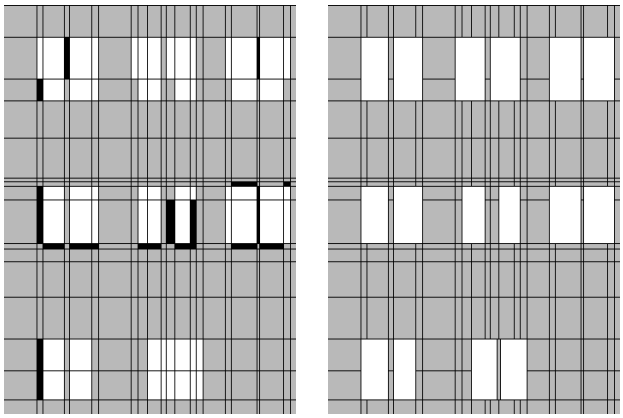
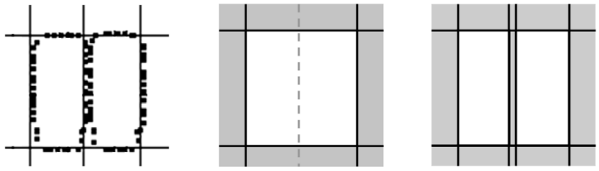


Figure 7. Classification of 3D cells before (left) and after enhancement (right).

Classification Enhancements. While most of the 3D cells can be classified reliably, the result is uncertain especially at window borders or in areas with little point coverage. Such cells with a relative coverage between 10% and 70% are represented by the black segments in the left of Figure 7. For the final classification of these cells, neighbourhood relationships as well as constraints concerning the simplicity of the resulting window objects are used. As an example, elements between two window cells are assumed to belong to the façade, so two small windows are reconstructed instead of one large window. This is justified by the fact that façade points have actually been measured in this area. Additionally, the alignment as well as the size of proximate windows is ensured. For this purpose, uncertain cells are classified depending on their neighbours in horizontal and vertical direction. Within this process, it is also guaranteed that the merge of window cells will result in convex window objects. Figure 7 (right) illustrates the enhanced classification result.



a) b) c)

Figure 8. Integration of additional façade cell.

As it is depicted in Figure 8, additional façade cells can be integrated easily if necessary. Figure 8a shows the LIDAR measurements for two closely neighboured windows. Since in this situation only one vertical line was detected, a single window is reconstructed (Figure 8b). To overcome this problem, the window object is separated into two smaller cells by an additional façade cell. This configuration is kept if façade points are available at this position (Figure 8c).

3.1.3 Façade Modelling

Within the following modelling process, the window cells are cut out from the existing coarse building model. The result of the building façade reconstruction is given in Figure 9. The front of the pyramidal wall dormer is not considered as being a part of the façade. Therefore, the reconstruction approach is applied on the roof extension, separately.

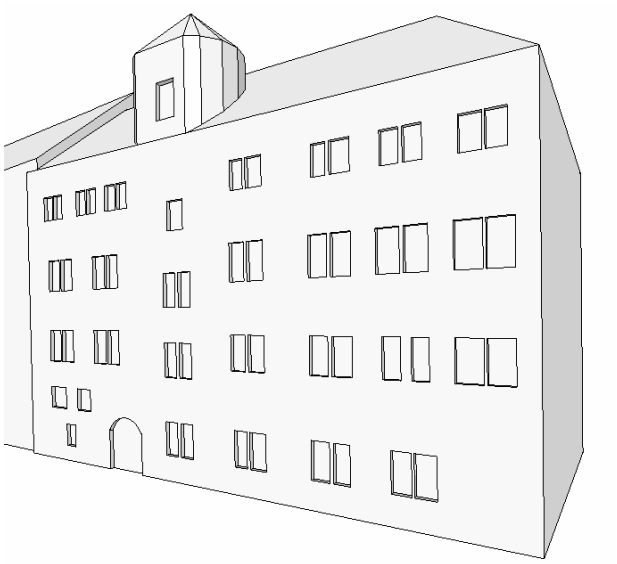


Figure 9. Refined facade of the reconstructed building.

While the windows are represented by polyhedral cells, also curved primitives can be integrated in the reconstruction process. This is demonstrated exemplarily by the round-headed door of the building. Furthermore, our approach is not limited to the modelling of indentations like windows or doors. Details can also be added as protrusions to the façade. LIDAR points that are measured at protrusions can be detected easily since they are not part of the façade plane but lying in front of it. If these points are classified as non-façade points, protrusion areas can be identified in the same way as window regions, just by searching no data areas within the set of points that belong to the facade. The availability of LIDAR points in front of the façade helps to classify the derived 3D cells as protrusion cells. Their extent in the façade's normal direction can be reconstructed by fitting planes to the measured protrusion points.

3.2 Facade Refinement By Photos

The level of detail for 3D objects that are derived from terrestrial laser scanning is limited depending on the point sampling distance. Small structures are either difficult to detect or even not represented in the data. By integrating image data in the reconstruction process the amount of detail can be increased. This is exemplarily shown for the reconstruction of window crossbars.

3.2.1 Derivation of 3D edges

Having oriented the image data, 3D information can be derived from corresponding image features in order to reconstruct details of the façade such as crossbars. For this purpose, edge points are extracted from the images by a Sobel filter. These edge point candidates are thinned and split into straight segments. Afterwards, the resulting 2D edges of both images have to be matched. However, frequently occurring façade structures, such as windows and crossbars, hinder the search for corresponding edges. Therefore, the boundaries of the windows that have already been reconstructed from the LIDAR points are projected into both images. Only the 2D edges lying inside these image regions are considered for the following matching process. Thus, possible mismatches are reduced, even though, they cannot be avoided entirely. Figure 10 depicts the selected 2D edges for an exemplary window in both images.

Figure 10. Selected 2D edges for a window in both images.

Remaining false correspondences lead to 3D edges lying outside the reconstructed window. Therefore, these wrong edges can be easily identified and removed. In addition, only horizontal and vertical 3D edges are considered for the further reconstruction process. The reconstructed wrong (green) and correct (red) 3D edges are shown in local façade coordinates in Figure 11. The position of the window that has been derived from the LIDAR data is illustrated in black.

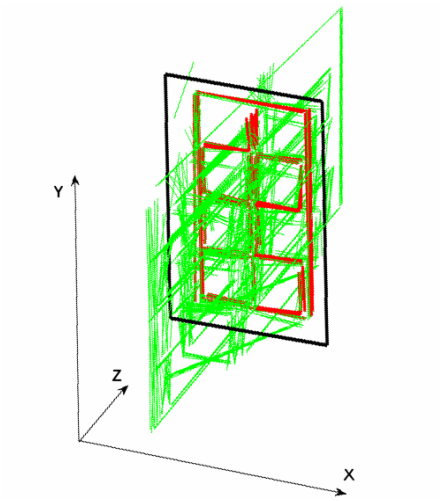


Figure 11. Wrong (green) and correct (red) 3D window edges.

3.2.2 Reconstruction of additional façade structures

Photogrammetric modelling allows the extraction of well-defined image features like edges and points with high accuracy. By contrast, points from terrestrial laser scanning are measured in a pre-defined sampling pattern, unaware of the scene to capture. That means that the laser scanner does not explicitly capture edge lines, but rather measures points at constant intervals. Furthermore, laser measurements at edges and corners may provide erroneous and unpredictable results because of the laser beam split that is caused at the object border. For these reasons, the positional accuracy of window borders that are reconstructed from LIDAR points is limited compared to the photogrammetrically derived 3D edges at crossbars. As a consequence, the 3D reconstructions from laser points and images may be slightly shifted. Therefore, the reconstruction of the crossbars is done as follows:

For each window, hypotheses about the configuration of the crossbars are generated and tested against the 3D edges derived from the images. Possible shapes are dynamically generated as templates by recursively dividing the window area in two or three parts. Recursion stops when the produced glass panes are too small for a realistic generation of windows. The minimum width and height of the glass panes are restricted by the same threshold value. After each recursion step, the fitting of the template with the 3D edges is evaluated. The partition is accepted if 3D edges are available within a buffer area around the dividing line. In a final step, the crossbars and the window frame are modelled. For this purpose, new 3D cells with a pre-defined thickness are generated at the accepted horizontal and vertical division lines as well as at the window borders. The result is exemplarily shown for two windows in Figure 12.

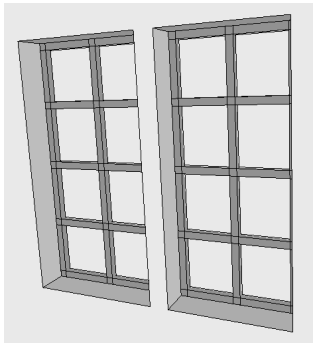


Figure 12. Reconstructed crossbars for two windows.

While most of the crossbars can be reconstructed reliably, problems may arise for windows that are captured under oblique views. Perspective distortions or occlusions make it difficult to detect 2D edges at crossbars (Figure 13). Consequently, fewer 3D edges can be generated thereof in those areas.

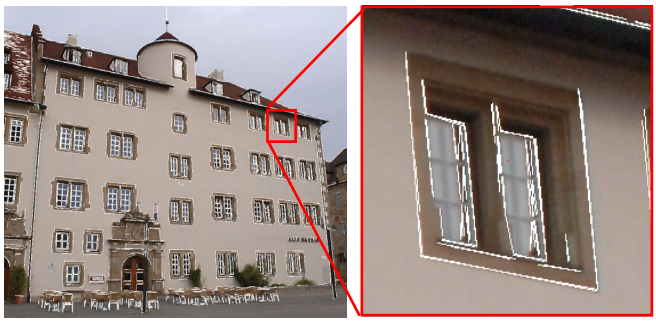


Figure 13. Detected 2D edges for a window captured under an oblique view.

To overcome this problem, neighbourhood relationships are taken into account within the final modelling step. The crossbar configuration is assumed to be equal for all windows of similar size which are located in the same row or column. Based on this assumption, similar windows can be simultaneously processed. Thus, the crossbar reconstruction leads to robust results even for windows that are partially distorted or feature strong perspective distortions in the respective image areas.

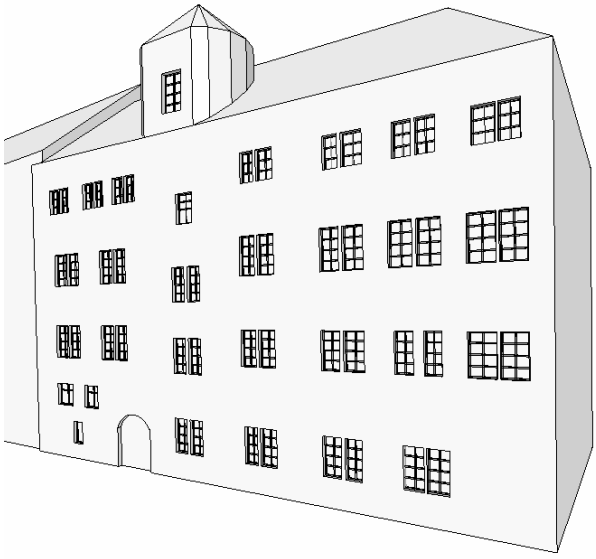


Figure 14. Refined facade with detailed window structures.

Figure 14 shows the final result of the building façade reconstruction from terrestrial LIDAR and photogrammetric modelling. This example demonstrates the successful detection of crossbars for windows of medium size. However, the dynamic generation of templates even allows for the modelling of large window areas as they often occur at facades of big office buildings.

4. CONCLUSION

Within the paper, an approach for the refinement of 3D building models based on cell decomposition was presented. As it was already proved for the automatic generation of topologically correct building models at different levels of detail (Haala et al, 2006), this approach allows the simple integration and removal of geometric detail for given building models. Even more important, symmetry relations like coplanarity or alignment can be guaranteed even for larger distances between the respective building parts. Thus, despite of the limited extent of the window primitives, which were extracted by the analysis of terrestrial LIDAR and images, structural information can be generated for the complete building. In principle, this information can then be used to support the façade interpretation at areas where measurements are only available with reduced quality and reliability. For these reasons, in our opinion this approach has a great potential for processes aiming at the reconstruction and refinement of building models from multiple data sets.

5. REFERENCES

Alegre, F. & Dallaert, F., 2004. A Probabilistic Approach to the Semantic Interpretation of Building Facades. International Workshop on Vision Techniques Applied to the Rehabilitation of City Centres, pp. 1-12.

Böhm, J. & Haala, N., 2005. Efficient Integration of Aerial and Terrestrial Laser Data for Virtual City Modeling Using LASERMAPS. IAPRS Vol. 36 Part 3/W19 ISPRS Workshop Laser scanning, pp.192-197.

Brenner, C. & Ripperda, N., 2006. Extraction of Facades Using RjMCMC and Constraint Equations. *Remote Sensing and Spatial Information Sciences* (36) 3.

Fischler, M. A. & Bolles, R. C., 1981. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *Communications of the ACM*, Vol. 24, pp. 381-395.

Früh, C. & Zakhor, A., 2003. Constructing 3D City Models by Merging Ground-Based and Airborne Views. *IEEE Computer Graphics and Applications,* Special Issue Nov/Dec.

Haala, N. & Alshawabkeh, Y., 2006. Combining Laser Scanning and Photogrammetry - A Hybrid Approach for Heritage Documentation. The 7th International Symposium on Virtual Reality, Archeology and Cultural Heritage VAST, pp.163-170.

Haala, N., Becker, S. & Kada, M., 2006. Cell Decomposition for the Generation of Building Models at Multiple Scales. IAPRS Vol. XXXVI Part III, Symposium Photogrammetric Computer Vision, pp. 19-24.

Lowe, D., 2004. Distinctive image features from scale-invariant keypoints. IJCV, Vol. 60(2), pp. 91-110.

Mäntylä, M., 1988. *An Introduction to Solid Modeling*. Computer Science Press, Maryland, U.S.A.

Mayer, H. & Reznik, S., 2006. MCMC Linked With Implicit Shape Models and Plane Sweeping for 3D Building Facade Interpretation in Image Sequences. IAPRS Vol. XXXVI, Part. 3.

Mikolajczyk, K. & Schmid, C., 2003. A performance evaluation of local descriptors. Proc. Conf. Computer Vision and Pattern Recognition, pp. 257-264.

Nistér, D., 2004. An efficient solution to the five-point relative pose problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 26(6), pp. 756-770.

Remondino, F. & Ressl, C., 2006. Overview and experiences in automated markerless image orientation. IAPRSSIS, Vol. 36, Part 3, pp.248-254.

AUTOMATIC REGISTRATION OF LASER POINT CLOUDS OF URBAN AREAS

M. Hebel [a], U. Stilla [b]

[a] FGAN-FOM, Research Institute for Optronics and Pattern Recognition, 76275 Ettlingen, Germany - hebel@fom.fgan.de
[b] Photogrammetry and Remote Sensing, Technische Universitaet Muenchen, 80290 Muenchen, Germany - stilla@bv.tum.de

KEY WORDS: Laser scanning, LIDAR, ICP, RANSAC, multiple scans, urban data, registration

ABSTRACT:

Many tasks in airborne laserscanning require the registration of different scans of the same object. Especially data acquired in urban environments with buildings viewed obliquely from different directions need to be aligned. In this paper we propose a method to filter these point clouds based on different techniques to speed up the computations and achieve better results with the ICP algorithm. A statistical analysis is used and planes are fitted to the data wherever possible. In addition to this, we derive extra features that are used for better evaluation of point-to-point correspondences. These measures are directly used within our extension of the ICP method instead of pure Euclidean distances. Both the intensity of reflected laser pulses and normal vectors of fitted planes are considered. The extended algorithm shows faster convergence and higher stability. We demonstrate and evaluate our approach by registering four data sets that contain different oblique views of the same urban region.

1. INTRODUCTION

1.1 General purpose and overview

Due to their ability to deliver direct 3D measurements, laser scanners (LIDAR) are highly operational remote sensing devices that can be used for many photogrammetric applications. Data can be collected even at night, since LIDAR is an active illumination technique. Future need for monitoring and observation devices has led to the development of advanced technology in this field. Accurate ground-based as well as agile airborne sensors have been studied by researchers and companies in recent years. Urban regions are scanned, e.g. to provide telecommunication companies with up-to-date 3D city models. The ever-increasing level of detail and additionally measured features are raising interest in the scientific community. Currently available laser scanners are capable of acquiring the full waveform of reflected pulses, thus enabling new methods of data analysis (Jutzi & Stilla, 2006). In addition to multiple return analysis, features like intensity of reflected pulses and pulse-width can be considered. The collected data are registered by using navigational sensors, which typically consist of an inertial measurement unit (IMU) and a GPS receiver. Airborne laser scanners are able to look obliquely at urban environments to obtain information concerning the facades of buildings. These data sets resulting from different viewing directions need to be co-registered, since navigational sensors usually show small errors or suboptimalities (Maas, 2000). Although the point density is different, that task is related to stitching of terrestrial laser scanning data, where there exist proven methods in literature (Makadia et al., 2006). While these methods provide a rough alignment of the data sets, the points have to be fine-aligned to achieve the desired results.

The iterative-closest-point (ICP) algorithm, originally proposed by Besl and McKay (1992), is the standard approach to correct these discrepancies. In urban areas different scanning directions typically lead to shadow effects, occurring at one or more sides of buildings. These occlusions are a severe problem if the ICP algorithm is applied directly to these point sets, as the classical ICP algorithm is susceptible to non overlapping regions (Rabbani et al., 2007). It doesn't consider the underlying geometry and may lead to incorrect results by ending up in local minima (Rusinkiewicz & Levoy, 2001). To avoid this, our method aims at filtering the point clouds before looking for correspondences. The filter operation is intended to keep only points that are most promising to result in correct pose estimation. Consequently, points that form the facades are discarded during the registration process. Since we have an airborne sensor in oblique configuration, most occlusions occur at the rear of buildings and at ground level. In the course of the data analysis we automatically detect the ground level and remove all points belonging to it. Referring to (Maas, 2000), data points on objects with an irregular shape like trees may also falsify the matching results. Therefore these points are removed in our approach by a robust estimation technique (RANSAC). If data originating from typical urban environments are processed this way, the remaining points mostly belong to the rooftops of buildings. This method was used to cut out vegetated areas rather than analysis of the full waveform and multiple laser returns for the following reason: during the filter operation we derive an additional feature for each remaining point, namely its local normal direction. We use a combination of this and the intensity value of the reflected laser pulses to improve the distance measure of the ICP algorithm, which classically takes only Euclidean distances into account. The presented approach is applied to test data sets and results are shown in the paper.

1.2 Related work

Numerous articles on point cloud registration have been published in recent years. In some parts our work follows or is based on the ideas presented in other articles that are especially mentioned in this section. Since Besl and McKay proposed their ICP algorithm in 1992, this approach has become the standard solution to the registration problem. Nevertheless, some work has been done on alternative concepts for geometric alignment

of point clouds, e.g. by relying on the geometry of the squared distance function of a surface (Pottman et al., 2004). Other least squares matching techniques can be used to establish correspondences between data from neighboring strips (Maas, 2000). Laserscanner data are usually irregularly distributed, which is handled by many authors by introducing a triangulated irregular network (TIN) structure (Kapoutsis et al., 1998). However, we do not follow this approach: during the filter operation, we distribute the 3D data to a two-dimensional array of an appropriate size, wherein each cell is filled with the 3D coordinates of the highest occurring data point at that location. This is done to discard all points belonging to the facades of buildings. After the filtering of inapplicable points, we are able to use the full 3D information and this mapping structure for efficient search operations within the ICP implementation.

Many different approaches have been made to improve the classical iterative-closest-point (ICP) algorithm. A summary and numerical comparison of several ICP variants has been given by Rusinkiewicz and Levoy (2001). They introduced a taxonomy for categorizing ICP variants. Therein the particular category depends on which state of the original algorithm is affected: (1) selection of subsets, (2) matching of points, (3) weighting of correspondences, (4) rejecting of pairs, (5) assignment of an error metric, (6) minimizing of the error metric. In their scheme our approach would be classified as type one, two and four, since we pre-select some of the points in each data set, we affect the matching of these points and perform a threshold-based rejection of outliers. Except for this, we leave the original ICP algorithm almost untouched.

All known modifications of ICP try to improve robustness, speed and/or precision. One critical point of ICP is its lacking robustness, because it needs outlier-free data to establish valid correspondences (Chetverikov et al., 2005). We overcome this problem by using only those points that should correspond well to points in the other data sets. Another possibility is filtering of correspondences, e.g. by comparing with a dynamic distance threshold to detect wrong assignments (Zhang, 1994). Besl and McKay (1992) originally suggested establishing point-to-point correspondences by evaluation of Euclidean distances. Various improved versions and variants have been proposed that use alternative distance measures, e.g. by analyzing local surface approximations of the data sets (Chen & Medioni, 1992). We consider enhancements of the Euclidean distance by taking additional features into account while establishing point-to-point relationships.

Segmentation of the point clouds into planar surfaces is part of our approach. Other researchers have presented many different techniques concerning this topic. A summary is given by Hoover et al. (1996). Some authors are interested in detecting planes, spheres, cylinders, cones, and even more primitives. Rabbani et al. (2007) describe two methods for registration of point clouds, in which they fit models to the data by analyzing least squares quality measures. Vosselman et al. (2004) use a 3D Hough transform to recognize structures in point clouds. Among all available methods, the RANSAC algorithm (Fischler & Bolles, 1981) has several advantages to utilize in the 3D shape extraction problem (Schnabel et al., 2006). We use a RANSAC-based robust plane detection method and additionally take its score and outlier information to distinguish between buildings and irregularly shaped objects like trees. Normal vectors are assigned to each remaining data point to use this as an additional feature supporting the registration.

2. EXPERIMENTAL SETUP

We used several commercial-off-the-shelf components for the data acquisition: an FPA infrared camera (data not considered in this paper), a laser scanning device and an inertial measurement unit. The sensors that are briefly described here have been attached to a Bell UH1-D helicopter and flights were carried out over Munich, Germany.

2.1 Navigational sensors

The APPLANIX POS AV comprises a GPS receiver and a gyro-based inertial measurement unit (IMU), which is the core element of the navigational system. The GPS data are used for drift compensation and geo-referencing, whereas the IMU determines accelerations with high precision. These data are transferred to the position and orientation computing system (PCS), where they are fused by a Kalman filter, resulting in position and orientation estimates for the sensor platform.

2.2 Laser Scanner

The RIEGL LMS-Q560 is a laser scanner that gives access to the full waveform by digitizing the echo signal. The sensor makes use of the time-of-flight distance measurement principle with nanosecond infrared pulses. Opto-mechanical beam scanning provides parallel scan lines, where each measured distance is approximately geo-referenced according to the estimated position and orientation of the sensor. Waveform analysis yields intensity and pulse-width as additional features of each 3D point in the resulting point cloud. Figure 1 shows a rendered visualization of a geo-referenced point cloud, in which each point is depicted with its associated intensity.

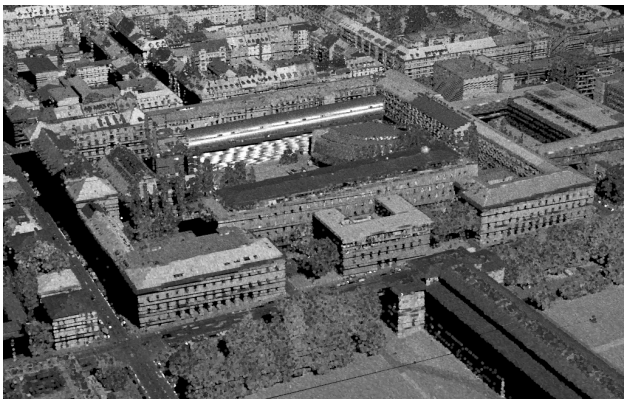


Figure 1. Laser data of an urban area scanned in 45° oblique view. Flight direction east to west (data set 4)

3. OVERVIEW OF USED METHODS

3.1 Random sample consensus (RANSAC)

The random-sample-consensus paradigm (RANSAC) as described by Fischler and Bolles (1981) is a standard technique to estimate parameters of a mathematical model underlying a set of observed data. It is particularly used in case that the observed data contain data points which can be explained by a set of model parameters (inliers) and such data points that do not fit the model (outliers). To apply the RANSAC scheme, a procedural method has to be available that determines the parameters to fit the model to a minimal subset of the data. In this paper we use RANSAC to fit planes to subsets of the point

clouds. If we have a set of n points $\{\boldsymbol{p}_1, \ldots, \boldsymbol{p}_n\}$ and we assume that this set mostly contains points that approximately lie on one plane (inliers) and some others that do not (outliers), simple least squares model fitting would lead to poor results because the outliers would affect the estimated parameters. RANSAC estimates a plane only by taking the inliers into account, provided that the probability of choosing only inliers among the data points is sufficiently high. To compute a plane, we select a random sample of three non collinear points (the minimal subset) $\boldsymbol{p}_i$, $\boldsymbol{p}_j$, and $\boldsymbol{p}_k$. The resultant plane's normal vector $\boldsymbol{n}_0$ is computed as $\boldsymbol{m} = (\boldsymbol{p}_i - \boldsymbol{p}_j)\times(\boldsymbol{p}_i - \boldsymbol{p}_k)$, $\boldsymbol{n}_0 = \boldsymbol{m}/|\boldsymbol{m}|$ and after that, with $(\boldsymbol{x}-\boldsymbol{p}_i)\cdot\boldsymbol{n}_0 = 0$ the plane's Hessian normal form is given. Using this representation, we can check all the other points $\boldsymbol{p}$ in $\{\boldsymbol{p}_1, \ldots, \boldsymbol{p}_n\}$ if they are inliers or outliers simply by computing their distance $d = |(\boldsymbol{p}-\boldsymbol{p}_i)\cdot\boldsymbol{n}_0|$ to the previously obtained plane. If the distance d is below a pre-defined threshold, we assess that point as inlier. The number of inliers and the average distance of all inliers to the plane are used to evaluate the quality of the fitted plane. This procedure is repeated several times in order to converge to the best possible plane.

3.2 Iterative closest point (ICP)

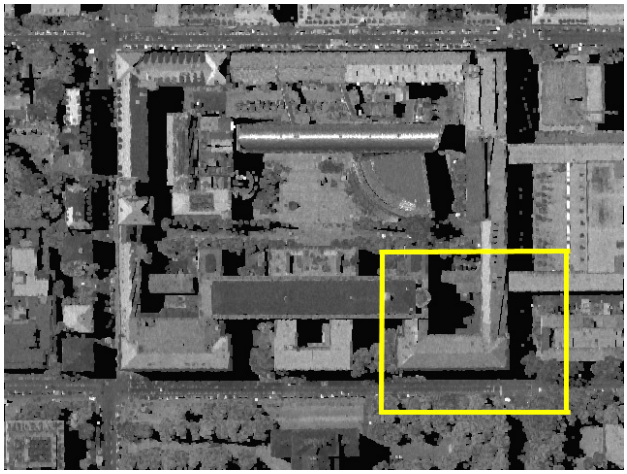
The iterative-closest-point algorithm (Besl & McKay, 1992) is intended to accurately and efficiently register 3D shapes like point sets, line segments, free-form curves, faceted surfaces, or free-form surfaces. In this paper we only consider the registration of two point sets. In literature usually one point set is seen as "data" and the other as "model", so we follow that terminology. It is assumed that both data sets are approximately in the same position, which is the case for our data.

During the ICP operation the whole data shape D is iteratively moved to be in best alignment with the model M. The first step of each iteration is to find the closest point $\boldsymbol{m}$ in M for each data point $\boldsymbol{d}$ in D. The identification of closest points between D and M should be accomplished by an efficient method, because it is the algorithm's most time consuming part. The result of this step is a sequence $(\boldsymbol{m}_1, \boldsymbol{m}_2, \ldots, \boldsymbol{m}_n)$ of closest model points to all n data points in $(\boldsymbol{d}_1, \boldsymbol{d}_2, \ldots, \boldsymbol{d}_n)$. The next step of each iteration is to find a translation and a rotation that moves the data points closest to their corresponding model points, such that the average squared (Euclidean) distance is minimized. This problem can be solved explicitly by the use of quaternions or singular value decomposition. Inconsistencies in the data sets due to missing points or occluded objects contribute directly to errors in the accuracy of the alignment. After the transformation of the data shape, the procedure is repeated and it converges monotonically to a local minimum. All the technical details and the proof of the last statement are thoroughly described in (Besl & McKay, 1992).

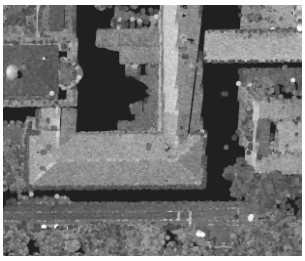
4. OUR EXTENSION OF THE ICP ALGORITHM

4.1 Preparing the data

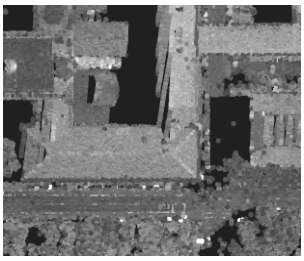
The basic idea of our approach is to filter the different point clouds to get only those points that are most promising to result in correct correspondences. By this we can improve the convergence behavior of the ICP algorithm, which is susceptible to occlusions, shadows and non overlapping regions in the data sets (Rabbani et al., 2007). In the first step we want to remove all points belonging to facades of buildings, since their presence in the data sets depends highly on the viewpoint. To achieve this, the 3D data are distributed to a horizontal two-dimensional array of appropriate size, in which each cell is filled with the 3D coordinates of the highest occurring data point at that position. The cell size corresponds to the average distance of adjacent points, which is dictated by the hardware specifications. We do not interpolate or down-sample the data during this process, since we keep the original 3D information in each cell. Anyway, by suppression of all non-maximal points we remove data belonging to the facades. Full waveform processing also yields the intensity of reflected laser pulses, which will later on be used as an additional feature for the registration, but can also provide a gray value for an image representation of the obtained 2D array. Figure 2 exemplarily shows details of the same urban region, measured in 45° oblique view from different directions.



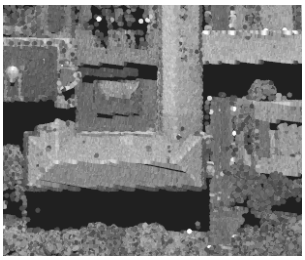
data set 1: south to north



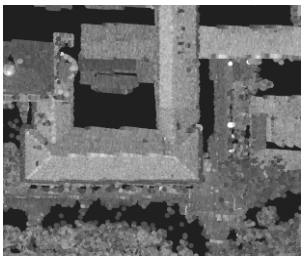
data set 1: south to north



data set 2: north to south



data set 3: west to east



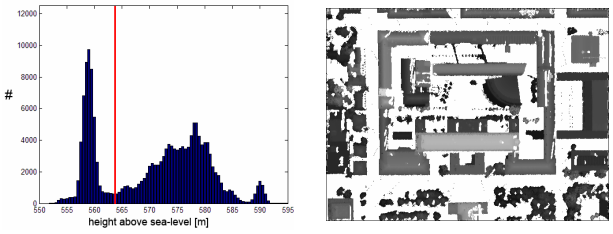
data set 4: east to west

Figure 2. Distribution of the points to a two-dimensional array and comparison of the four data sets

4.2 Detection of the ground level

The different scanning directions in Figure 2 are indicated by the position of the shadows. Due to active illumination of the scene by the airborne sensor, most shadows and occlusions occur at the rear of buildings and at ground level. As occlusions are not handled by the ICP algorithm, we need to detect the ground level and remove all points belonging to it. The detection of the ground level is fairly easy for urban

environments. We simply analyze the histogram of height values derived from the previously generated data matrix. The histogram shows a multimodal distribution, in which the laser points at ground level appear as the lowest distinct peak (Figure 3a). It is easy to find the optimal threshold that discards most data associated with the ground level by an analysis of local maxima and local minima or an expectation-maximization (EM) algorithm. Once that level is found, we take out all laser points below (Figure 3b).



a b

Figure 3. a) Determination of a threshold to remove points at ground level, b) remaining points depicted as gray-value coded height data

4.3 Robust elimination of clutter objects

The residual points in Figure 3b clearly show the contours of buildings, but also clutter objects like vegetation remain persistent. The following procedure is used to remove these irregularly shaped objects:

(1) Choose an unmarked grid-position *(i, j)* at random among the available data in the matrix.
(2) Check a sufficiently large neighborhood of this position for available data, resulting in a set *S* of 3D points.
(3) Set the counter *k* to zero.
(4) If *S* contains more than a specific number of points (e.g. at least six), continue. Otherwise mark the current position *(i, j)* as discarded and go to step 14.
(5) Increase the counter *k* by one.
(6) Perform a RANSAC-based plane fitting with the data points in the specified set *S*.
(7) If RANSAC is not able to find an appropriate plane or the number of inliers is low, mark the current position as discarded and go to step 14.
(8) Obtain the plane's Hessian normal form $(\boldsymbol{x}-\boldsymbol{p})\cdot\boldsymbol{n_0} = 0$ and push the current position *(i, j)* on an empty stack.
(9) Pop the first element *(u, v)* off the stack.
(10) If the counter *k* reached a predefined maximum and the number of points in *S* is high enough, store the normal vector information $\boldsymbol{n_0}$ at position *(u, v)* and mark that position as processed.
(11) Check each position in a neighborhood of *(u, v)* that has not already been looked at if it contains data and in that case, check if the 3D point lies sufficiently near to the plane. If so, push its position on the stack and include the point in a new set *S'*.
(12) While the stack is not empty, go to step 9. Otherwise continue with step 13.
(13) If the counter *k* reached its maximum (e.g. three cycles), set it to zero and continue with step 14. Otherwise go to step 4 with the new set of points *S:=S'*.
(14) Go to step 1 until a certain number of runs has been performed or no more unmarked data is available.

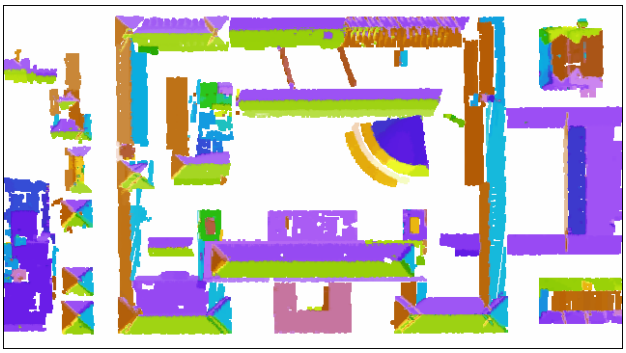


Figure 4. Remaining points after robust elimination of clutter objects, color-coded according to the normal direction

The suggested method is intended to distinguish between man-made objects like buildings and clutter objects like bushes or trees. In each run, we randomly select a position in the previously generated matrix of laser points and try to fit a plane to the neighboring data of that position. The RANSAC technique provides a robust estimation of the plane's parameters, with automatic evaluation of the quality, e.g. by the number of outliers. If the fitted plane is of poor quality, we assess the data associated with the current location as clutter. Otherwise, we try to optimize the plane fitting by looking for all data points that support the obtained plane. The underlying operation is accomplished in steps 9, 10, 11 and 12, which actually represent a seed-fill algorithm. The local plane fitting process is repeated with the supporting points to get a more accurate result. The final plane's normal vector is stored at all positions of points assigned to that plane and the corresponding data is classified as building. Figure 4 shows detected rooftops for one of the datasets, depicted with an appropriate color-coding according to their normal direction.

4.4 Executing the ICP algorithm

Now that we have removed all data that might lead to an incorrect registration, we are ready to start the ICP procedure. As mentioned in Section 3.2, the most computationally inefficient part of ICP is the search operation. We need to implement an operator that finds the closest point $\boldsymbol{m}$ in point cloud *M* for each data point $\boldsymbol{d}$ in the other data set *D*. Many improvements of this step have been proposed in literature, most of them using efficient search strategies based on k-d-trees, Voronoi diagrams or a Delaunay tessellation of the data (Kapoutsis et al., 1998). In general, this is the best known way to handle this issue, especially when dealing with irregularly distributed points. However, we do not follow this way because we already have all we need: during the filtering of points we introduced a two-dimensional data matrix among which the 3D data were distributed such that each cell contains the highest occurring data point at that position. This non-maximum suppression was done to discard the facades of buildings. Now we can use this regular grid to perform the search operation within the ICP algorithm. Let **M** be the matrix that holds *M* and let **D** be the matrix containing *D*. First we specify all indices describing cells of **D** where a 3D point of *D* is included, resulting in a list L. After that we build a look-up table T comprising all relevant indices of **M** in a sufficiently large neighborhood N_{ij} of each listed entry *(i, j)* in L. This has to be done only once. The search for closest points is done in every iteration of ICP by going through the look-up table T.

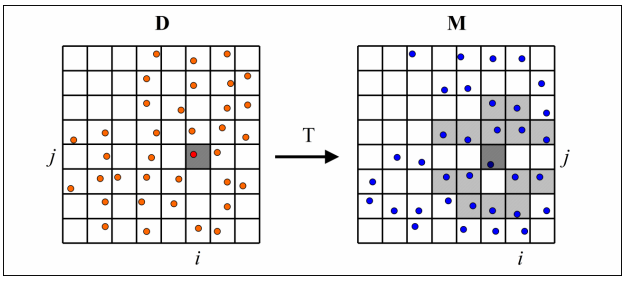


Figure 5. Assignment of a subset of M to every point in D. Each dot represents 3D coordinates together with normal direction and intensity

The functionality of the look-up table T is illustrated in Figure 5. The considered neighborhood N_{ij} at each position *(i, j)* depends on the pre-registration of the data sets. N_{ij} is essentially larger than it is depicted here, but it can be scanned reasonably fast. At each list entry in L, the assigned subset of M is processed to look for the best candidate for a point-to-point correspondence. This weighting is classically done by evaluation of Euclidean distances between $\boldsymbol{d_{ij}}$ and each point $\boldsymbol{m}$ in N_{ij} where the minimal distance is chosen. The Euclidian distance is defined by:

$$D_E = \sqrt{(\boldsymbol{d}-\boldsymbol{m})\cdot(\boldsymbol{d}-\boldsymbol{m})} \tag{4.1}$$

Due to the fact that we have an estimated normal direction and the measured intensity of the reflected laser pulse as additional information, we are able to evaluate the point-to-point relationship better than just by Euclidian distances. Given the normal direction $\boldsymbol{n_d}$ corresponding to data point $\boldsymbol{d}$ and accordingly $\boldsymbol{n_m}$ corresponding to model point $\boldsymbol{m}$, their distance can be expressed by the angle between these two vectors:

$$D_N = \arccos(\boldsymbol{n_d}\cdot\boldsymbol{n_m}),\quad D_N \in [0,\pi] \tag{4.2}$$

For better performance, the arc cosine in (4.2) can be replaced by a predefined look-up table. Finally, if I_d and I_m denote the intensity of the respective reflected laser pulses, a suitable distance measure would be:

$$D_I = \left|I_d - I_m\right|,\quad D_I \in [0,1] \tag{4.3}$$

To combine the distances D_E, D_N and D_I for each pair of points, the most straightforward approach is a linear combination D_c of the distances. The problem of finding a suitable metric D_c consists of finding the optimal set of weighting factors α, leading to the best performance of the ICP algorithm. Finding the optimal distance measure in this sense is difficult, since optimality depends on sensor and scene characteristics. One possibility to estimate the weighting factors is to analyze domain, mean and variance of each component, comparable to the Mahalanobis distance metric. After the search for corresponding points we have to find a translation and a rotation that moves the data points closest to their corresponding model points, such that the average squared Euclidean distance is minimized. To solve this problem explicitly, we use singular value decomposition (SVD) as described in (Arun et al., 1987). We briefly summarize the essential steps of their method. First the centroids of all points among the set of n correspondences are computed:

$$\boldsymbol{c}_m = \frac{1}{n}\sum_{i=1}^{n}\boldsymbol{m}_i,\qquad \boldsymbol{c}_d = \frac{1}{n}\sum_{i=1}^{n}\boldsymbol{d}_i \tag{4.4}$$

M and D are translated to the origin by $\boldsymbol{c}_m$ and $\boldsymbol{c}_d$ respectively, resulting in the new point clouds:

$$\begin{aligned}\bar{M} &= \left\{\bar{\boldsymbol{m}}_i \mid \bar{\boldsymbol{m}}_i = \boldsymbol{m}_i - \boldsymbol{c}_m,\ i=1,\ldots,n\right\},\\ \bar{D} &= \left\{\bar{\boldsymbol{d}}_i \mid \bar{\boldsymbol{d}}_i = \boldsymbol{d}_i - \boldsymbol{c}_d,\ i=1,\ldots,n\right\}\end{aligned} \tag{4.5}$$

Then a matrix H is defined as

$$H = \sum_{i=1}^{n}\bar{\boldsymbol{d}}_i\bar{\boldsymbol{m}}_i^{\,T} \tag{4.6}$$

The singular value decomposition $H = UAV^T$ of this 3x3 matrix is fast to compute and leads to the optimal rotation R and translation $\boldsymbol{t}$:

$$R = VU^T,\quad \boldsymbol{t} = \boldsymbol{c}_m - R\boldsymbol{c}_d \tag{4.7}$$

The proof of this is given in (Arun et al., 1987). After computation of optimal rotation and translation, we transform the data set D with respect to R and $\boldsymbol{t}$ and continue with the next ICP iteration until a stop criterion is met.

5. FIRST RESULTS AND EVALUATION

The proposed method was tested by using four data sets containing different oblique views of the same urban region (Figure 2). Overall, the different parameterized runs of ICP resulted in acceptable alignment of the point clouds when visually assessed. It is rather difficult to give a quantitative evaluation of the final registration, as the ICP method always converges monotonically to a local minimum of the proposed distance measure D_c. Instead of using the final sum of all values D_c to assess the alignment, one may consider other quality measures. Using the average distance between points in one data set and corresponding surface-patches in the other set is more significant than just counting point-to-nearest-point distances for assessment. This point-to-tangent-plane distance D_t can easily be quantified since we have the normal direction available for every point in M. Given a data point $\boldsymbol{d}$ and the direct neighborhood of its corresponding model point $\boldsymbol{m}$ with normal directions $\boldsymbol{n_m}$, we define D_t as $D_t = min\ |(\boldsymbol{d}-\boldsymbol{m})\cdot\boldsymbol{n_m}|$. The sum of all values D_t is used to evaluate the registration's accuracy.

We tried several combinations and weightings of D_E, D_N and D_I, and also used D_E and D_N to perform a threshold-based rejection of clearly outlying correspondences. Figure 6 illustrates the decreasing distance measure D_t against the ICP iteration number. The measured pose error decreases dramatically with the first iteration and then it converges smoothly in few steps. In some degree, the convergence behavior depends on the tested parameters. The first group of parameters does not include outlier rejection, so the average distance is comparatively high even after twenty ICP iterations. Threshold based filtering of obviously wrong assigned points outperforms influencing the error metric, so this method should be used in any case. It should be avoided to lower the threshold too much, since this may emphasize a local minimum. In our experiments we always kept at least 80 percent of the correspondences. Regarding the registration of point clouds measured from different directions, we found that the difference of intensity can be quite erratic due to active illumination of the scene, so we do not suggest using D_I in general. In addition to rejection of bad correspondences, best results were achieved by

a combination of the Euclidean distance D_E with the radian measure D_N of the normal vectors ($D_E + 10D_N$).

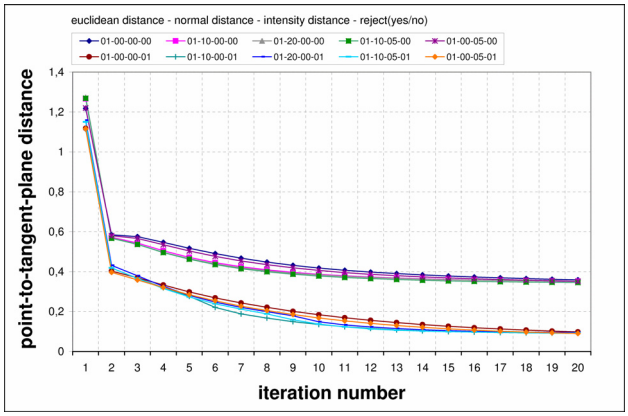


Figure 6. Course of the modified ICP algorithm with different parameters

At the moment, filtering and registering two point clouds takes about 10 minutes on a standard PC, each set containing 150.000 points. All results presented in this paper were obtained with programs that were developed under MATLAB®. Our implementation is far from being optimal, and due to a lot of unnecessary code for visualization and evaluation, there exists some potential to get much shorter computation time. The final average displacement of corresponding points is 10 cm (Figure 6) and that is comparable to the laser's range resolution. One result of the final data alignment is depicted in Figure 7. It is a rendered visualization of the registered point clouds, each depicted in slightly different color. The respective brightness is defined by the intensity of each laser echo.

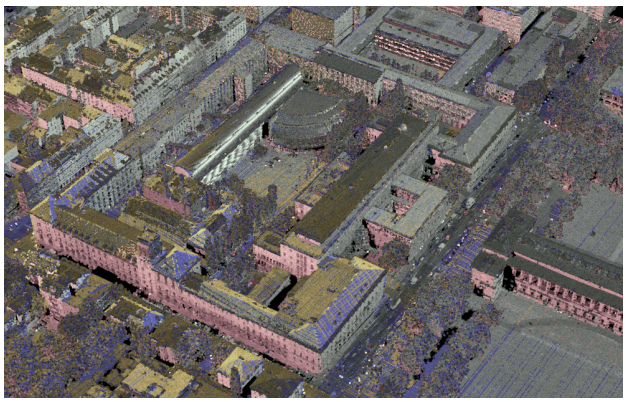


Figure 7. Illustrative result of the final data alignment

6. CONCLUSION AND FUTURE WORK

We proposed a method to filter laser point clouds of urban areas based on different techniques to achieve better results when the ICP algorithm is applied. Both the intensity of reflected laser pulses and normal vectors of fitted planes were considered to influence the ICP performance. The extended registration algorithm shows faster convergence and higher stability. We demonstrated and evaluated our approach by registering four data sets containing different oblique views of the same urban region. Future work will focus on updating the navigational data rather than aligning the point clouds. Up till now we did not consider multiple returns for vegetation mapping and discarded intensity as distance measure because of the relative nature of that signal, so the full waveform information was not used at all. In future work we will put more emphasis on full waveform analysis and we will also consider the simultaneously recorded IR image data for fusion aspects.

7. REFERENCES

Arun, K.S., Huang, T.S., Blostein, S.D., 1987. Least square fitting of two 3-d point sets. IEEE Transactions on Pattern Analysis and Machine Intelligence 9 (5), pp. 698-700.

Besl, P.J., McKay, N.D., 1992. A method for registration of 3-D shapes. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 14, No. 2, pp. 239-256.

Chen, Y., Medioni, G., 1992. Object Modelling by Registration of Multiple Range Images. Image and Vision Computing, Vol. 10, No. 3, pp. 145-155.

Chetverikov, D., Stepanov, D., Krsek, P., 2005. Robust Euclidean alignment of 3D point sets: the trimmed iterative closest point algorithm. Image and Vision Computing 23 (3), pp. 299-309.

Fischler, M.A., Bolles, R.C., 1981. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Communications of the ACM 24 (6), pp. 381-395.

Hoover, A., Jean-Baptiste, G., Jiang, X., Flynn, P.J., Bunke, H., Goldof, D.B., Bowyer, K., Eggert, D.W., Fitzgibbon, A., Fisher, R.B., 1996. An Experimental Comparison of Range Image Segmentation Algorithms. IEEE Transactions on Pattern Analysis and Machine Intelligence 18 (7), pp. 673-689.

Jutzi, B., Stilla, U., 2006. Precise range estimation on known surfaces by analysis of full-waveform laser. Photogrammetric Computer Vision PCV 2006. International Archives of Photogrammetry and Remote Sensing Vol. 36 (Part 3).

Kapoutsis, C.A., Vavoulidis, C.P., Pitas, I., 1998. Morphological techniques in the iterative closest point algorithm. Proceedings of the International Conference on Image Processing ICIP 1998, Vol. 1 (4-7), pp. 808-812.

Maas, H.-G., 2000. Least-Squares Matching with Airborne Laserscanning Data in a TIN Structure. International Archives of Photogrammetry and Remote Sensing 33 (3a), pp. 548-555.

Makadia, A., Patterson A., Daniilidis K., 2006. Fully Automatic Registration of 3D Point Clouds. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition CVPR 2006, Vol. 1, pp. 1297-1304.

Pottmann, H., Leopoldseder, S., Hofer, M., 2004. Registration without ICP. Computer Vision and Image Understanding 95 (1), pp. 54-71.

Rabbani, T., Dijkmann, S., van den Heuvel, F., Vosselman, G., 2007. An integrated approach for modelling and global registration of point clouds. ISPRS Journal of Photogrammetry and Remote Sensing 61 (6), pp. 355-370.

Rusinkiewicz, S., Levoy, M., 2001. Efficient Variants of the ICP Algorithm. Proceedings of 3D Digital Imaging and Modeling 2001, IEEE Computer Society Press, 2001, pp. 145-152.

Schnabel, R., Wahl, R., Klein, R., 2006. Shape Detection in Point Clouds. Technical report No. CG-2006-2, Universitaet Bonn, ISSN 1610-8892.

Vosselman, G., Gorte, B.G.H., Sithole, G., Rabbani, T., 2004. Recognising structure in laser scanner point clouds. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences 46 (8), pp. 33–38.

Zhang, Z., 1994. Iterative Point Matching for Registration of Free-Form Curves and Surfaces. International Journal of Computer Vision 13 (2), pp. 119-152.

DIGITAL TERRAIN MODEL ON VEGETATED AREAS: JOINT USE OF AIRBORNE LIDAR DATA AND OPTICAL IMAGES

Frédéric Bretar [a], Nesrine Chehata [b]
[a] Institut Géographique National
2-4 Av. Pasteur 94165 St. Mandé cedex, France
Email: Frederic.Bretar@ign.fr
[b] Institut EGID - Université Bordeaux 3
1 Allée Daguin 33607 Pessac
Email: Nesrine.Chehata@egid.u-bordeaux.fr

KEY WORDS: Airborne Lidar, DTM, Vegetation Index, Classification

ABSTRACT:

Airborne Lidar system provides the Earth's topography as 3D point clouds. Many algorithms have been implemented to sort out the automatic classification problem as well as the Digital Terrain Model generation (DTM). This is mainly due to the various aspects of landscapes within a global survey which can include urban, forested or mountainous areas. This paper is focused on the generation of DTM over rural areas that are composed of open fields and forests. The methodology we propose is based on the joint use of optical images and Lidar data. It aims at adapting the window size of a morphological-based filtering algorithm to the presence of vegetated areas. In this context, Lidar intensity and optical images are combined to generate a Hybrid Normalized Difference Vegetation Index (HNDVI). A vegetation mask is then calculated with HNDVI and Lidar variance information. The window size continuously varies from a predefined minimum distance to an automatically processed upper boundary. We show with conclusive results the potentiality of a full combination of Lidar data and RGB optical images for improving the generation of fine DTMs on rural environments.

1 INTRODUCTION

Airborne Lidar systems are nowadays a popular technique to acquire representations of landscapes as 3D point clouds. One of the first process to be applied to raw Lidar data is a classification step, providing ground and off-ground points, and a Digital Terrain Model generation step. These two steps have been a research topic for some years. The generation of DTMs requires efficient algorithms to process large data volumes on various and complex landscapes such as urban areas (Dellcqua et al., 2001), forest areas (Kraus and Pfeifer, 1998) (Haugerud and Harding, 2001) or mountainous areas (Wack and Stelzl, 2005). Many algorithms have been implemented and tested so far, but no generic solution appeared (Sithole and Vosselman, 2003).

Methodologies based on a progressive TIN (Axelsson, 2000) are popular but parameters highly depend on the terrain slope as well as on the relevancy of laser points to belong to the terrain: last pulse is not always a true ground point, especially in presence of dense vegetation coverage. In a DTM production context, the terrain surface as well as the classification result have to be locally and manually corrected.

Methodologies based on a local estimation of the terrain (morphological approaches) suffer from the same drawbacks (Eckstein and Munkelt, 1995) (Kilian et al., 1996). More specifically, the potential of morphological filters to provide a good estimate of the ground depends on the filtering window size and on the distribution of the buildings and trees in the data. If a small window size is used, the local topography will be well represented, provided that there are enough true ground points within the neighborhood. Nevertheless, points belonging to large roof structures will not be filtered as off-ground points. On the contrary, a large window size will tend to over-filter Lidar points and to smooth the final DTM. A solution to overcome these effects is to affine locally the window size of the filter (Kilian et al., 1996) (Zhang et al., 2003).

This study is focused on the generation of DTM over vegetated areas. We propose in this paper a methodology which aims to adapt the window size of a morphological-based filtering algorithm to the presence of vegetated areas (Bretar et al., 2004). In this context, Lidar intensity and optical images are combined to generate a Hybrid Normalized Difference Vegetation Index (HNDVI). A vegetation mask is then calculated with HNDVI and Lidar variance information. The window size continuously varies from a predefined minimum distance to an automatically processed upper boundary.

After presenting the filtering algorithm, we will describe the generation of the vegetation mask as well as the adaptive window size strategy. Results are finally presented and analyzed.

2 BACKGROUND

This part briefly reminds the classification algorithm presented in (Bretar et al., 2004). From an initial location (minimal altitude of the point cloud), the filtering algorithm propagates within the point cloud following the processed region frontier $\mathcal{F}_{\leq}$, namely following the steepest local slope over a 4-connexity neighboring system $\mathbb{V}_s^{4_c}$. Eligible locations evolve within a sorted (ascending order) container structure ($\mathcal{F}_{\leq}$). At each site* $s \in \mathcal{F}_{\leq}$, a square neighborhood $\mathcal{V}_s$ of dimension $\mathbf{d}_s$ is extracted. $\mathbf{d}_s$ is set so that the overlapping ratio between two adjacent locations should be at least of 50%. In the previous work, $\mathbf{d}_s$ is kept constant for each site s. At site s, an initial estimate of the terrain elevation is performed by calculating an average value of laser point height belonging to a rank filtered subset ($\overline{\mathscr{R}_{0.2}(s)}$). In our case, a ratio of 0.2 has been defined, but it depends on the data quality.

The filtering algorithm is based on a bipartite voting process. Laser points will be classificated as **ground** or **off-ground** points depending on their height difference to the local terrain estimate $|\hat{h}_{\textbf{ground}}(s) - l_z| < T, T \in \mathbb{R}$. $\hat{h}_{\textbf{ground}}(s)$ is calculated by averaging the altitudes of laser points belonging to $\mathcal{V}_s$ and classificated as **ground**. Considering the overlapping ratio of the neighborhoods, laser points are classificated several times either as ground ($n_s^{\textbf{ground}}$) or off-ground ($n_s^{\textbf{offground}}$) points.

In order to check the coherence of $\hat{h}_{\textbf{ground}}(s)$ with regard to the DTM values over a 3×3 window centered in s ($DTM^{3\times3}(s)$), which is a memory of all previously calculated terrain altitudes, we integrate a linear correction to the final value of the DTM at location s. This correction depends on a coefficient α, on

* In image processing, a site corresponds to a pixel (i,j).

$\hat{h}_{\textbf{ground}}$ and on a mean DTM over a 3×3 window centered in s ($\overline{DTM^{3\times3}(s)}$).

$$DTM(s) = \alpha \hat{h}_{\textbf{ground}}(s) + (1-\alpha)\overline{DTM^{3\times3}(s)} \quad (1)$$

Finally, for each neighborhood extraction $\mathcal{V}_s$, laser points will be labeled following local criteria. At the end of the propagation, a laser point will have been labeled n times as ground and m times as non-ground. We then affect the final label corresponding to $\max(n, m)$, which is the most representative vote. Algorithm 1 summarizes the algorithm.

Algorithme 1: Algorithm for classifying laser points

begin
  Input : $\alpha \in [0,1]$, T =0.5m
  while $\mathcal{F}_{\leq} \neq \emptyset$ **do**
    Extraction of $\mathcal{V}_s$ of dimension $\mathbf{d}_s$
    $\hat{h}_{\textbf{ground}}(s) = \overline{\mathscr{R}_{0.2}(s)}$
    foreach *laser point* $l \in \mathcal{V}_s$ **do**
      if $|\hat{h}_{\textbf{ground}}(s) - l_z| < T$ **then**
        $l \in$ **ground**; $n_s^{\textbf{ground}} ++$
      else $l \in$ **off-ground**; $n_s^{\textbf{offground}} ++$
      $\hat{h}_{\textbf{ground}}(s) = mean(l_z / l \in \textbf{ground})$
    $DTM(s) = \alpha \hat{h}_{\textbf{ground}}(s) + (1-\alpha)\overline{DTM^{3\times3}(s)}$
    $\mathcal{F}_{\leq} = \mathcal{F}_{\leq} \backslash \{s\}$
    $\mathcal{F}_{\leq} = F_{\leq} \bigcup \mathbb{V}_s^{4c}$
end

3 METHODOLOGY

3.1 Predicting vegetated areas

This part is dedicated to the generation of a high vegetation mask including hedges, isolated trees and forest areas, but agricultural fields. It is a prediction of vegetated areas based on both the analysis of images and Lidar data.

Typical vegetation has higher reflectance in the near-infrared wavelengths (700-1350 nm) than in the visible domain because red light is mostly absorbed by the plant's chlorophyll (90%). The contrast in reflectance between the red and the near-infrared makes possible to create an image that separates vegetated land cover from non-vegetated land cover by calculating the Normalized Difference Vegetation Index. As usual in a Lidar survey, Lidar data are very often combined with a RGB image acquisition, but infrared channel is not always available, such is the case in this study. We therefore decided to investigate the potential of Lidar intensity information as infrared channel.

Recorded intensity is a function of many variables such as laser power, target reflectivity, range, incidence angle, media absorption (Coren et al., 2005). It also depends on the detection mode applied in the first/last pulse systems (Wagner et al., 2004). The intensity values need to be better calibrated by system developers (Ahokas et al., 2006) or at least to be corrected by scanning homogeneous targets to compute and validate a backscattering model (Coren and Sterzai, 2006). However, if these assumptions are particularly relevant, we decided to investigate the potential of using raw uncalibrated Lidar intensity in case of a joint index computation, which is generally derived from image-based infrared data.

Lidar intensities are therefore resampled at a resolution which depends on the point density. The resampled intensity is calculated on a regular grid by extracting a circular neighborhood of 2.5 m diameter. This choice ensures that enough Lidar points belong to the neighborhood. The final intensity value is the mean of the intensities of 3D Lidar points included in this neighborhood (with a point density of 0.7 pt/m^2 there are $\backsim$ 14 Lidar points). The dynamic of raw intensity values is low with very few saturated values (out of the main distribution). Seeing that the orthophoto is an 8-byte image, the main distribution of intensity values is stretched between 0 and 255. An Hybrid-NDVI is then calculated by:

$$\text{HNDVI} = \frac{\mathcal{I}_{Lidar} - R}{\mathcal{I}_{Lidar} + R} \quad (2)$$

where R is the red channel of the optical image and $\mathcal{I}_{Lidar}$ is the Lidar intensity image.

Vegetation is detected by thresholding the HNDVI image. According to (Lillesand and Kiefer, 1994), the values of NDVI for vegetation range is from a low 0.05 to a high 0.66. Applied to our Hybrid-NDVI, these thresholds provided fairly good results.

In order to segregate high vegetation from fields, we crossed this threshold with a binary standard deviation mask: only sites s such as $\sigma(s)$ greater than 1 m are considered. Finally, the vegetation mask is defined as a set $\mathcal{M}$ defined as:

$$\mathcal{M} = \left\{ s / \text{HNDVI}(s) \in [0.05, 0.66] \bigcap \sigma(s) \geq 1 \text{ m} \right\} \quad (3)$$

and is represented as an image of the same resolution as both the orthophoto and the DTM's one. Figure 1(b) illustrates a vegetation mask calculated for this study.

3.2 Adapting the local neighboring system

As mentioned in the introducing part, the window size of the neighboring system $\mathbf{d}_s$ (defined in section 2) in case of a morphological-based classification process should be small enough to keep all ground details but large enough to ensure the removal of up-ground objects such as trees or/and buildings. The section describes an algorithm for adapting the window size $\mathbf{d}_s$ of the structural element ($\mathcal{V}_s$ is a square window) at site s to vegetated areas. The adaptative window size $\mathbf{d}_s$ is processed over laser points belonging to mask $\mathcal{M}$. By definition, if a laser point is included into $\mathcal{M}$, it is likely to belong to a vegetated area. $\mathbf{d}_s \in [\mathbf{d}_s^{\text{min}}, \mathbf{d}_s^{\text{max}}]$ should therefore be enlarged to ensure that enough laser points within $\mathcal{V}_s$ belong to the true terrain.

$\mathbf{d}_s^{\text{min}}$ is a critical parameter and has to be defined so that a minimum number of laser points should be processed within $\mathcal{V}_s$. Besides, $\mathbf{d}_s^{\text{min}}$ ensures the overlapping structure of neighborhoods. We therefore constrain $\mathbf{d}_s$ as:

$$\mathbf{d}_{\text{min}}^{\text{abs}} < \mathbf{d}_s^{\text{min}} \leq \mathbf{d}_s \leq \mathbf{d}_s^{\text{max}} \quad (4)$$

where $\mathbf{d}_{\text{min}}^{\text{abs}}$ is a global minimal window size over the entire survey and is independent on site s. If p (0.2 in this paper as mentioned previously) is the percentage of lowest laser points within $\mathcal{V}_s$, r the DTM ground resolution and $\overline{\delta}$ the global average point density, $\mathbf{d}_{\text{min}}^{\text{abs}}$ is defined as:

$$\mathbf{d}_{\text{min}}^{\text{abs}} = \max\left(\frac{1}{p * \overline{\delta}}, r\right) \quad (5)$$

In our algorithm, $\mathbf{d}_s^{\text{min}}$ depends on two parameters: i) the local standard deviation σ_s^{local} calculated on the $p\%$ lowest laser points of $\mathcal{V}_s$ and ii) the neighboring laser points that belong to $\mathcal{M}$. The higher the local standard deviation σ_s^{local}, the larger the minimum window size $\mathbf{d}_s^{\text{min}}$. Statistically, low standard deviations of altitudes are over represented in rural areas. Therefore, $\mathbf{d}_s^{\text{min}}$ has to be highly increasing with low values of σ_s^{local}. We then define the variations of $\mathbf{d}_s^{\text{min}}$ as:

$$\mathbf{d}_s^{\text{min}} = \mathbf{d}_{\text{min}}^{\text{abs}} + K \log(1 + \sigma_s^{local}) \; K \in \mathbb{R} \quad (6)$$

$K = 6$ was found to be a good compromise for processing our data. To ensure the regularity of adjacent $\mathbf{d}_s^{\text{min}}$ values, a Gaussian

(a) 2.5 m-Orthoimage.

(b) 2.5 m-Vegetation mask (white pixels) superimposed on the orthoimage.

Figure 1: Generation of a vegetation mask using Lidar data and optical images.

filter is applied over the $\mathbf{d}_s^{\min}$ image providing that equation 4 is still satisfied.

$\mathbf{d}_s^{\min}$ also depends on the neighboring laser points that belong to $\mathcal{M}$. This criteria discriminates small vegetated regions from forests. From an initial value calculated in equation 6, $\mathbf{d}_s^{\min}$ is increased by one DTM's resolution unit until there is at least one cell of mask $\mathcal{M}$ that is not considered as a vegetation point.

$\mathbf{d}_s^{\max}$ is set proportional to $\mathbf{d}_s^{\min}$. A low value of $\mathbf{d}_s^{\min}$ should correspond to a small vegetated area and $\mathbf{d}$ has therefore to vary in a small interval. On the contrary, a high $\mathbf{d}_s^{\min}$ is likely to correspond to a forest area. To ensure terrain points be statistically represented in $\mathcal{V}_s$, $\mathbf{d}$ has to vary within a large interval. In this study, we set $\mathbf{d}_s^{\max} = 3\mathbf{d}_s^{\min}$.

For each site s and a window size of $\mathbf{d}_s^{\min}$, let us consider the percentage of predicted vegetation area in $\mathcal{V}_s$:

$$x_s = \frac{\text{Vegetated surface of } \mathcal{V}_s}{\mathbf{d}_s^{\min} * \mathbf{d}_s^{\min}} \in [0,1] \qquad (7)$$

The behavior of $\mathbf{d}_s$ between $\mathbf{d}_s^{\min}$ and $\mathbf{d}_s^{\max}$ is not a linear function because the window size has to be strongly enlarged in case of a high vegetated ratio where lowest points are not guaranteed to belong to the true terrain. Meanwhile, in case of low ratios, one can expect that lowest laser points belong to the true terrain and describe it in details. $\mathbf{d}_s$ will consequently increase exponentially with x_s following equation 8.

$$\mathbf{d}_s(x_s) = Ae^{\beta\, x_s^2} + B \qquad (8)$$

With

$$\begin{cases} \mathbf{d}_s(0) & = \mathbf{d}_s^{\min} \\ \mathbf{d}_s(1) & = \mathbf{d}_s^{\max} \end{cases}$$

we have

$$A = \frac{\mathbf{d}_s^{\max} - \mathbf{d}_s^{\min}}{e^{\beta} - 1} \text{ and } B = \mathbf{d}_s^{\min} - A$$

Parameters in equation 8 were chosen so that $\mathbf{d}_s$ should be highly enlarged when more than half the structural element size contains dense vegetation, i.e. when $x_s > 0.5$. We therefore choose a x_s^2 dependency of the exponential function and $\beta = 3$ for two main grounds (figures 2 and 3):

i. the slope is smaller than a simple exponential when $x_s < 0.5$. This ensures a regularized $\mathbf{d}_s$ map that is not sensitive to low vegetated areas.

ii. the slope is higher than a simple exponential when $x_s > 0.5$. This ensures a quick increase of the window size in case of dense vegetated areas.

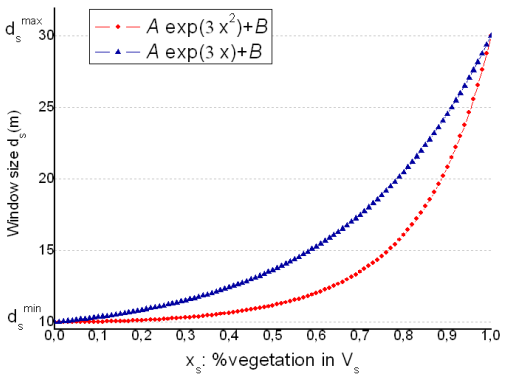


Figure 2: Comparison of two parametric forms of d_s.

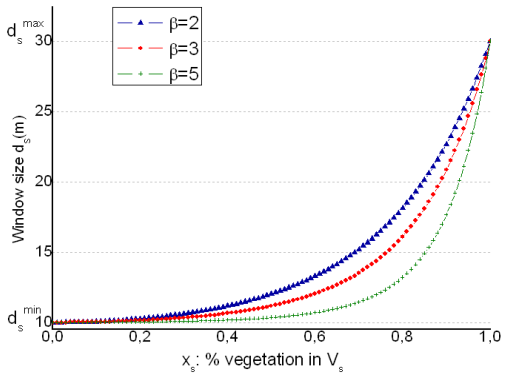


Figure 3: Variations of d_s with $\beta \in \{2, 3, 5\}$.

4 THE DATA SET

Lidar data have been collected in 2004 by the Institut Français de Recherche pour l'Exploitation de la Mer (IFREMER) over the Morbihan's Gulf, France. It has been funded by the fundation TOTAL. The entire survey is composed of 230.10^6 points with

intensities and has been acquired with an ALTM (Optech) system 1210. The point density is 0.7 pt/m^2. The Lidar wavelength is 1064 nm.

Optical images are extracted from the BDOrtho®(French orthophoto data basis) of the Institut Géographique National (IGN) with a nominal resolution of 0.5 m, but resampled at 2.5 m for the generation of the Hybrid-NDVI image.

5 Results and Discussion

This part describes the results of the algorithm as well as the impact of the joint use of Lidar data and RGB images on the generation of fine DTMs on vegetated areas. The algorithm has been tested on a large data set described above. We present the results obtained from two 2km×2km square subset of the Morbihan's Gulf called *GM-7-5* and *GM-6-5* in figure 7.

DTMs presented in Figures 7(a) and 7(e) have been calculated with a constant $\mathbf{d}_s$ =10 m with solely 3D Lidar data. We clearly observe that such value of $\mathbf{d}_s$ is well adapted to the retrieval of the terrain over open field areas with a high level of details (field delineations, roads). However, when comparing figure 7(a) with the corresponding aerial image in figure 1(a), one can notice that forested areas are mis-classificated providing an erroneous estimate of the DTM over these areas as expected. It is also visible on both profiles presented in figures 4 top and 5 top where grey curves represent the DTM calculated with a constant window size. When increasing the structural element size $\mathbf{d}_s$ up to 30 m (figures 7(b) and 7(f)), most of vegetated areas have been filtered off. But, many details were lost during this process, providing a smooth DTM.

Figures 7(c) and 7(g) show two DTM calculated with the adaptive window size strategy using 3D Lidar data, Lidar intensity and RGB optical image. Both of them have been post-processed by a Markovian regularization (Bretar, 2007). This post-process consists of minimizing an energy in a Bayesian context. This energy is composed of a data term and a regularization term. The first one describes the Euclidian distance between the surface (the DTM) and the Lidar points classificated as terrain points. The second one aims to compensate the effect of the data term so that the final surface should not be too noisy. This term depends on the intrinsic geometry of the surface. We define the regularization term $\mathcal{E}_r$ as a function of the trace and the determinant of the Hessian matrix $\mathbf{H}$.

$$\mathcal{E}_r = \alpha_1 tr(\mathbf{H})^2 - \alpha_2 \det(\mathbf{H}) \quad (9)$$
$$\text{with } \alpha_2 \geq 0 \text{ and } \alpha_1 \geq \frac{\alpha_2}{2}$$

The trace describes the local convexity of the surface while the determinant is linked to the shape of the surface with regard to its tangent plane (parabolic, elliptic, hyperbolic). A steepest gradient algorithm has been used to solve the optimization problem.

One can observe that microrelieves calculated with a constant $\mathbf{d}_s$=10 m are preserved while terrain points are better estimated under vegetated areas. The calculated DTM shows interesting meso-relieves such as shallow valleys covered by dense vegetation. It is a cross validation of our algorithm since no exhaustive field campain have been performed so far.

Figure 7(d) shows the related distance image based on the Hybrid-NDVI mask (see section 3.2). Inverse grey levels are related to distance values included in [5m,130m]. Darker pixels correspond to large structural elements and are linked to vegetated areas, while brighter pixels correspond to open fields. When looking closely to the image of distances, one can observe the strip-like parallel pattern of the Lidar acquisition survey. The higher point density of overlapped areas are visible as slightly darker vertical strips on figure 7(d). This effect can be explained by the definition of $\mathbf{d}_{\text{min}}^{\text{abs}}$ in equation 5 where the density is taken into account.

We show on figures 4, 5 and 6 three relevant profiles where Lidar points are plotted along a transect of $\sim$ 2km. The final terrain elevation values are plotted as a back lines. The secundary plots represent the corresponding adaptative window sizes along the profile. These curves show that the structural element of the filter evolves depending on the complexity of the off-gound topography and is well-adaped to the estimations of the terrain elevation. Sometimes, when the window size is the largest, the terrain can be over-estimated since it results from the averaging of minimal elevations.

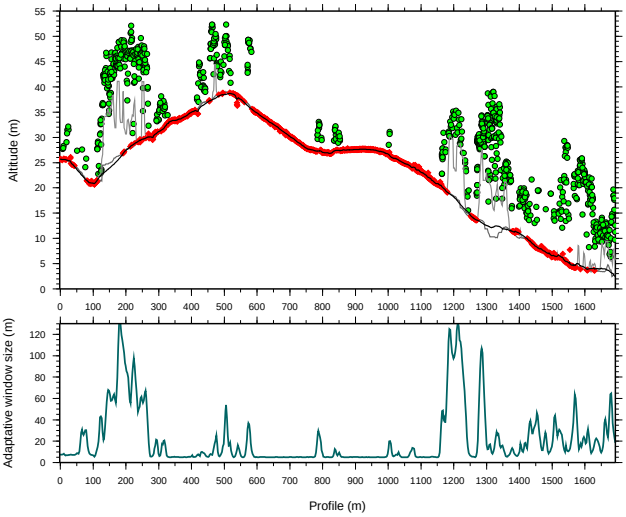


Figure 4: **Top**: Profiles (*GM-7-5*) of classificated Lidar points (green→ off-ground, red→ ground). Grey lines are computed with a constant $\mathbf{d}_s$ =10 m and $\mathbf{d}_s$ =30 m. The final terrain is represented as a black line. **Down**: Adpatative window sizes corresponding to the profile.

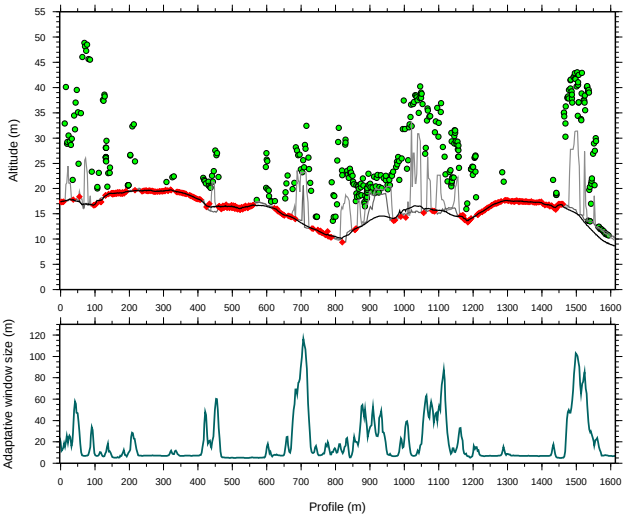


Figure 5: **Top**: Profiles (*GM-7-5*) of classificated Lidar points (green→ off-ground, red→ ground). Grey lines are computed with a constant $\mathbf{d}_s$ =10 m and $\mathbf{d}_s$ =30 m. The final terrain is represented as a black line. **Down**: Adpatative window sizes corresponding to the profile.

In this study, we have not given any physical interpretation of Lidar intensity. The distribution of the Lidar intensity image has been artificially stretched for coherence purpose with regard to the orthophoto red channel. As future work, we plan to compare optical infrared channel with Lidar intensity in order to give more physical content of the Lidar intensity image as well as to calibrate both infrared sources.

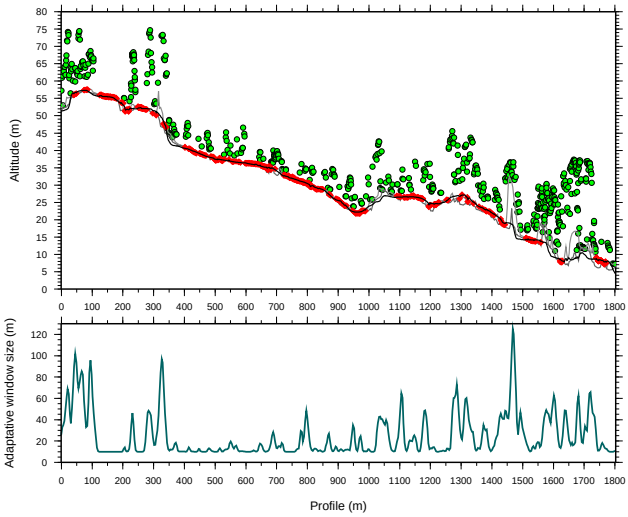


Figure 6: **Top**: Profiles (*GM-6-5*) of classificated Lidar points (green→ off-ground, red→ ground). Grey lines are computed with a constant $\mathbf{d}_s$ =10 m and $\mathbf{d}_s$ =30 m. The final terrain is represented as a black line. **Down**: Adpatative window sizes corresponding to the profile.

The vegetation mask results from the coarse thresholding of the Hybrid-NDVI image (equation 3). The vegetation areas may be punctually under-detected leading to non-dense vegetated regions (figure 1(b)). Indeed, the Lidar survey and the aerial optical images have not been acquired at the same time. Moreover, Lidar intensity is not as reliable as an optical infrared channel. Nevertheless, the overlapping constraint as well as the regularity of the window size of adjacent $\mathcal{V}_s$ ensure that under-detected vegetation areas are treated as they were. Therefore, there is no need to process a finer vegetation mask.

6 Conclusion

The paper presents a full methodology for using jointly 3D Lidar data, Lidar intensity and RGB images within the context of DTM generation on vegetated areas. We showed that mixing Lidar intensity values together with RGB optical images in an Hybrid Normalized Vegetation Index is a promising approach for processing rural landscapes with open fields and high vegetation even if both data sets have not been acquired simultaneously. Besides, we showed that, in a typical acquisition framework of RGB images with Lidar data (point cloud and intensity), it is possible to highly improve the classification process for generating a fine DTM.

REFERENCES

Ahokas, E., Kaasalainen, S., Hyyppa, J. and Suomalainen, J., 2006. Calibration of the Optech ALTM 3100 laser scanner intensity data using brightness targets. In: Proc. of the ISPRS Commission I Symposium, IAPRS, Marne-la-Vallee, France.

Axelsson, P., 2000. Dem generation from laser scanner data using adaptative tin models. IAPRS, Vol. XXXIII part B4/1, pp. 110–117.

Bretar, F., 2007. Processing fine digital terrain models by markovian regularization from 3D airborne lidar data. In: ICIP 2007, IEEE, San-antonio, Texas.

Bretar, F., Chesnier, M., Pierrot-Deseilligny, M. and Roux, M., 2004. Terrain modeling and airborne laser data classification using multiple pass filtering. In: Proc. of the XXth ISPRS Congress, IAPRS, Vol. XXXV part B, ISPRS, Istanbul, Turkey, pp. 314–319.

Coren, F. and Sterzai, P., 2006. Radiometric correction in laser scanning. International Journal of Remote Sensing 27(15), pp. 3097–3104.

Coren, F., Visintini, D., G., P. and Sterzai, P., 2005. Integrating lidar intensity measures and hyperspectral data for extracting of cultural heritage. In: Proc. of Workshop Italy-Canada for 3D Digital Imaging and Modeling: applications of heritage, industry, medicine and land.

Dellcqua, F., Gamba, P. and Mainardi, A., 2001. Digital terrain models in dense urban areas. In: Proc. of the ISPRS Workshop on land surface mapping and characterization using laser altimetry, IAPRS, Vol. XXXIV–3/W4, Annapolis, U.S., pp. 195–202.

Eckstein, W. and Munkelt, O., 1995. Extracting objects from digital terrain models. In: Proc. Int. Society for Optical Engineering: Remote Sensing and Reconstruction for Three-Dimensional Objects and Scenes, Vol. 2572, pp. 43–51.

Haugerud, R. and Harding, D., 2001. Some algorithms for virtual deforestation of lidar topographic survey data. In: Proc. of the ISPRS Workshop on land surface mapping and characterization using laser altimetry, IAPRS, Vol. XXXIV, Annapolis, U.S., pp. 211–218.

Kilian, J., Haala, N. and Englich, M., 1996. Capture and evaluation of airborne laser scanner data. IAPRS, Vol. XXXI, pp. 383–388.

Kraus, K. and Pfeifer, N., 1998. Determination of terrain models in wooded areas with airborne laser scanner data. ISPRS Journal of Photogrammetry and Remote Sensing 53, pp. 193–203.

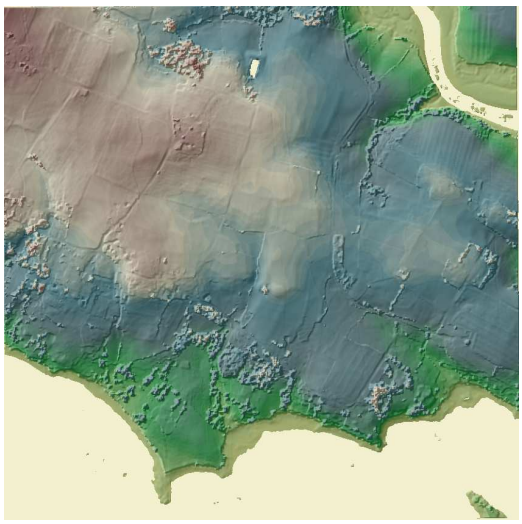
Lillesand, T. and Kiefer, R., 1994. Remote Sensing and Image interpretation. John Wiley & Sons.

Sithole, G. and Vosselman, G., 2003. Comparison of filtering algorithms. In: Proc. of the ISPRS Workshop III/3 '3D Reconstruction from Airborne Laserscanner and InSAR', IAPRS, Vol. XXXIV, Dresden, Germany, pp. 71–78.

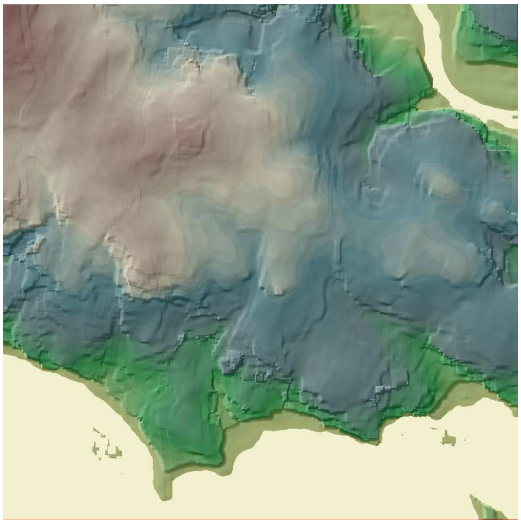
Wack, R. and Stelzl, H., 2005. Laser DTM generation for South-Tyrol and 3D-visualization. In: Proc. of the ISPRS Laserscanning 2005, IAPRS, Vol. XXXVI–3/W19, Enschede, the Netherlands, pp. 48–53.

Wagner, W., Ullrich, A., Melzer, T., Briese, C. and Kraus, K., 2004. From single-pulse to full-waveform airborne laser scanners: Potential and practical challenges. IAPRS, Vol. 35, Part B3, pp. 201—206.

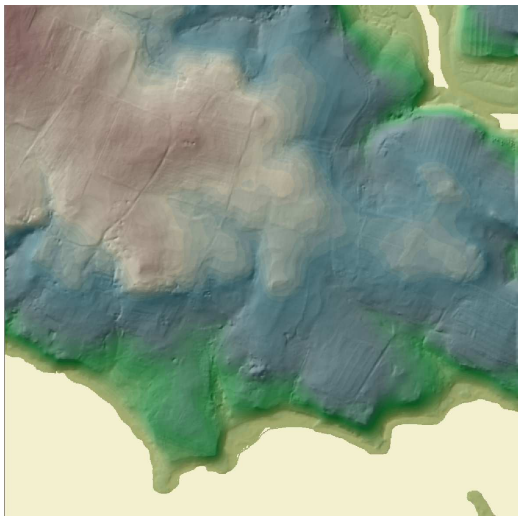
Zhang, K., Chen, S.-C., Whitman, D., Shyu, M., Yan, J. and Zhang, C., 2003. A progressive morphological filter for removing nonground measurements from airborne lidar data. IEEE Transactions on Geoscience and Remote Sensing 41(4), pp. 872–882.



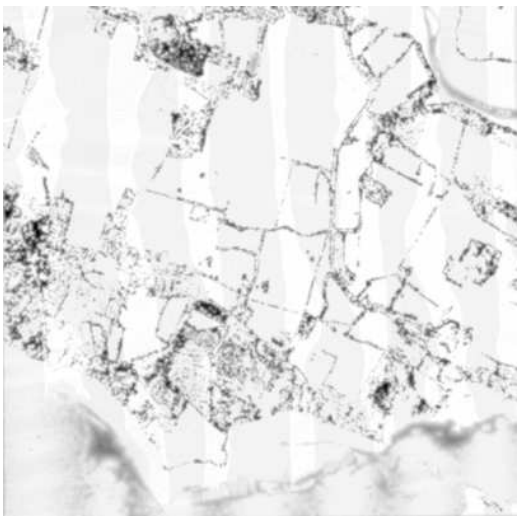
(a) DTM from Lidar data with $\mathbf{d}_s$ =10 m.



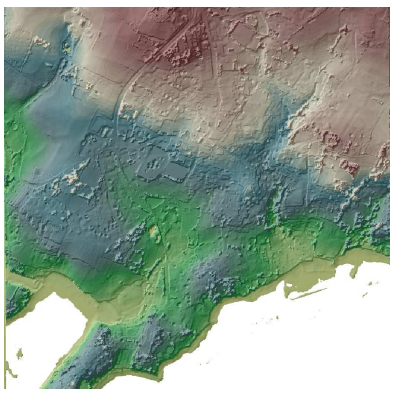
(b) DTM from Lidar data with $\mathbf{d}_s$ =30 m.



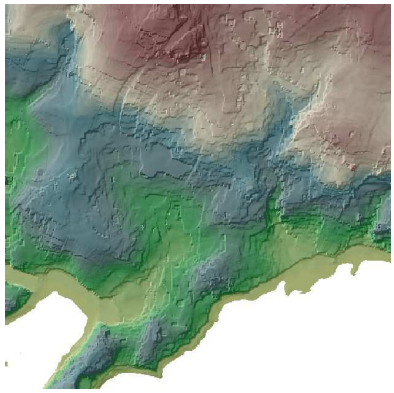
(c) DTM from Lidar data and RGB images with adaptive $\mathbf{d}_s$



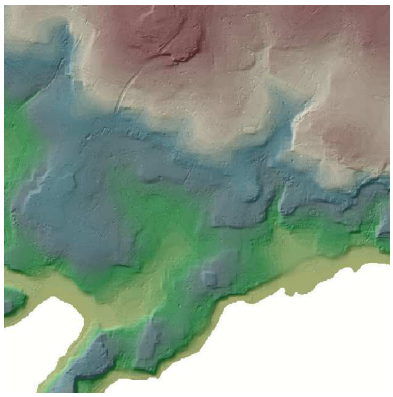
(d) Image $\mathbf{d}_s$ coded in inverse grey level scale.



(e) DTM from Lidar data with $\mathbf{d}_s$ =10 m.



(f) DTM from Lidar data with $\mathbf{d}_s$ =30 m.



(g) DTM from Lidar data and RGB images with adaptive $\mathbf{d}_s$

Figure 7: Results of DTM processing over the area *GM-7-5* (figures a, b, c, d) and *GM-6-5* (figures e,f,g).

DETECTION OF WEAK LASER PULSES BY FULL WAVEFORM STACKING

U.Stilla [a], W. Yao [a, *], B. Jutzi [b]

[a] Photogrammetry and Remote Sensing, Technische Universitaet Muenchen, Arcisstr. 21, 80290 Munich, Germany
(uwe.stilla, wei.yao)@bv.tum.de
[b] Research Institute for Optronics and Pattern Recognition, FGAN-FOM, 76275 Ettlingen, Germany
jutzi@fom.fgan.de

KEY WORDS: Weak pulse signal, Pulse detection, Waveform, Stacking, Laser scanning, Correlation.

ABSTRACT:

Pulse detection is a fundamental task of processing data of pulsed laser systems for extracting features of the illuminated object. Weak pulses below the threshold are discarded by classic methods and the involved information will get lost. In this paper we present an approach for detecting weak pulses from waveform laser data based on analyzing neighbourhood relations given by coplanarity constraint. Waveform stacking technique is used to improve the signal to noise ratio (SNR) of object with poor surface response by means of mutual information accumulation, thus hypotheses for planes of different slopes are generated and verified. Each signal is assessed by a likelihood value with respect to accepted hypotheses. At last signals will be classified according to likelihood values using two thresholds and visualized by the traffic-light paradigm. The presented method was applied as a low-level operation to a whole waveform cuboid of an urban area and shows promising results. Results contain detected pulses reflected from objects, which can not be predicted by the previously detected point cloud.

1. INTRODUCTION

Nowadays commercial full-waveform laser scanning systems are available to capture the waveform of the backscattered laser pulse. Beside the large-footprint spaceborne system, namely Geoscience Laser Altimeter System (GLAS) (Brenner *et al.*, 2003), additional small-footprint airborne systems are available: the Scanning Hydrographic Operational Airborne Lidar Survey system (SHOALS) is used for monitoring nearshore bathymetric environments (Irish & Lillycrop, 1999), where the OPTECH ALTM 3100, TOPEYE MK II, and TOPOSYS HARRIER 56 (it contains the RIEGL LMS-Q560) are mainly used to survey forestry environment (Hug *et al.*, 2004; Reitberger *et al.*, 2006; Söderman *et al.*, 2005).

Typically for these systems the waveform is captured for a predefined range interval, where the backscattered laser pulses are expected or a trigger signal is detected. Then the measurement of a scene containing objects expanded over a large range area may lead to an incomplete recording, because the range area is above the predefined range interval. For our investigations we use data of an experimental laser scanning system that allows capturing the complete full-waveform data within a range area of 200m.

The recording of the received waveform offers the possibility for the end user to select different methods to extract features and range information. The most popular methods are peak detection, leading edge detection, average time value detection, and constant fraction detection. This topic was investigated by different authors, e.g. Der *et al.*, 1997; Jutzi & Stilla, 2003; Wagner *et al.*, 2004. To derive a parametric description of the pulse properties range, width and amplitude a decomposition method on the waveform is proposed by Hofton *et al.* (2000). Further improvements on reliability and accuracy can be derived by signal processing methods based on the transmitted and the received waveform, e.g. cross-correlation (Hofton & Blair, 2002) and inverse filtering (Jutzi & Stilla, 2006).

When a threshold-based approach is used, attenuation of the signal by transmission through aerosol, fog, rain, snow, etc., reflection on a weakly backscattering cross section, or strong material absorption can produce subliminal signal values, where the detection of the object is not possible. Full-waveform laser data has provided us a possibility to utilize the neighborhood relation between laser signals, which can be utilized to support pulse detection. This idea is based on the common characteristic of man-made objects in the scene – regular distribution in local neighbourhood.

In this work we investigate full-waveform data with weak laser pulses by exploiting neighbourhood relation to support pulse detection. The experimental setup for a fast recording of a scene with different urban objects is described in section 2. Section 3 gives us a detailed description and discussion of purposed algorithm. Results of performed test are presented in section 4.

2. EXPERIMENTAL SETUP

2.1 Laser system

The laser scanning system has three main components: an emitter unit, a receiver unit, and a scanning unit.

For the emitter unit, we use a short duration laser pulse system with a high repetition rate (42 kHz). The pulsed Erbium fibre laser operates at a wavelength of 1.55 µm. The average power of the laser is up to 10 kW and the pulse duration is 5 ns. The beam divergence of the laser beam is approximately 1 mrad.
The receiver unit to capture the waveform is based on an optical-to-electrical converter. This converter contains an InGaAs photodiode sensitive to wavelengths of 900 to 1700 nm.

* Corresponding author.

Furthermore, we use a preamplifier with a bandwidth of 250 MHz and an A/D converter with 20 GSamples/s.

The scanning unit for the equidistant 2-d scanning consists of a moving mirror for elevation scan (320 raster steps of 0.1°) and a moving platform for azimuth scan (600 raster steps of 0.1°). The field of view is 32° in vertical and 60° in horizontal direction.

2.2 Test scene

For the investigations, a measuring platform is placed at a height of 15 m, pointing at an outdoor scene. The different urban objects in the scene are buildings, streets, vehicles, parking spaces, trees, bushes, and grass. Some objects are partly occluded and the materials show various backscattering characteristics.

2.3 Scanning and data

For each orientation of the beam within the scanning pattern, the emitted signal and the received signal are recorded over the time t for the time interval $t=t_{min}$ to $t=t_{max}$. The time interval selected for the recording of the signal depends on the desired recording depth of the area (in our case up to 200m). For each discrete range value the intensity value of the pulse is stored. The entire recording of a scene can be interpreted and visualised as a discrete data cuboid I[x y t], where the measured intensity at each time t and each beam direction [x y] is stored. It has to be taken into account the recording geometry for correct interpretation of the data.

3. STRATEGY

An automatic approach for detecting laser weak pulses based on neighbourhood relation preserved in the waveform cuboid is presented in this section.

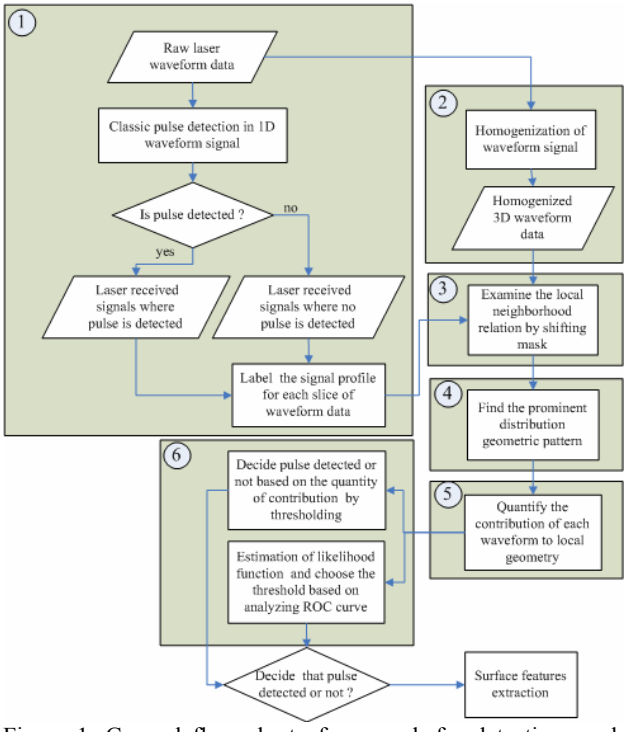


Figure 1. General flow chart of approach for detecting weak signal

The general strategy is sketched in Fig. 1. At first, a classic pulse detection method is adopted to extract the significant pulse signal backscattered from objects. Every laser ray in the waveform cuboid will be labelled as detected or undetected. On the other hand, the waveform cuboid is undergone a homogenization process. Afterwards, we generate a 1-D mask and shift it through every vertical slice of waveform cuboid to analyze the local neighbourhood relation by waveform stacking, and the prominent geometric pattern can be found. By comparing original waveforms to averaged stacked waveform the contribution of each waveform to local geometry can be assessed by a likelihood value. At last, according to this value, we can attribute a class (***'pulse', 'no pulse' or 'uncertain'***) to every original waveform by selecting thresholds in advance or by statistical inference.

3.1 Classic pulse detection method

Because of algorithmic efficiency and simplicity, it is always wise to apply the classical 1-D pulse detection methods (e.g. peak detection or correlation method) to raw waveform data firstly; high-energy reflected pulse can be detected, as displayed in Fig. 2. The rest laser rays where no pulse has been detected contain either weak reflection pulse from specific objects (e.g. window glass or roof behind the trees) or no backscattered signal at all due to inexistence of objects (sky)

Figure 2. Pulses detected by peak method, white pixels indicate detected pulses.

For pulse detection a noise dependent threshold was estimated to separate a signal pulse from background noise. Therefore the background noise was estimated and if the intensity of the waveform is above $3\,\sigma_n$ of the noise standard deviation for the duration of at least 5ns, then the waveform will be accepted.

For peak detection method, the range value of the detected pulses is determined by the maximum pulse amplitude, where the largest reflectance is expected. Another two typical surface features are roughness and reflectance which correspond to width and amplitude of waveform. Then, a Gaussian curve can be fitted to recorded waveform to get a parametric description using iterative Gauss-Newton method ((Jutzi & Stilla, 2006)). The estimated parameters for waveform features are the average time τ , standard deviation σ and maximum amplitude a .

$$w(t) = \frac{a}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(t-\tau)^2}{2\sigma^2}\right) \qquad (1)$$

According to results of the classic pulse method, every waveform of laser beams will be labelled as detected (1) or undetected (0) and arranged in arrays corresponding to various vertical slices for further study.

3.2 Homogenization of waveform laser data

As stated above the raw waveform signal recorded by an experimental laser system is noisy and has strong fluctuation caused by the emitted pulse. In Fig. 3a, a section of waveform is plotted overlaid with an adjacent waveform reflected from surface of same material. The none-pulse part (indicated by two arrows in Fig.3a) of blue waveform shows a lower amplitude value than the corresponding part of red one, actually, they should be recorded in same level.

The waveform data cuboid should provide us a unified and consistent data base; otherwise it is difficult to exploit neighbourhood relations hiding in it. Therefore a pre-processing step needs to be performed to normalize the whole data set. The y-t slice of the cuboid is expected to exhibit more consistent after homogenization, when observed from side-look.

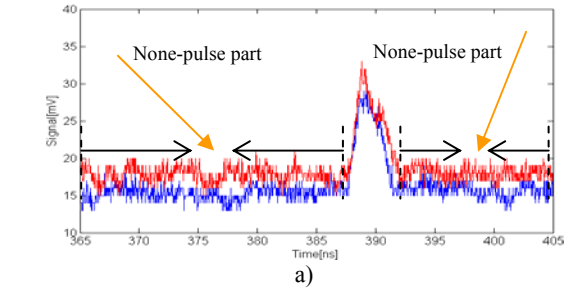


Figure 3. Necessity of homogenization operation a) neighbouring raw waveforms, b) e-t (elevation-temporal) slice image section before homogenization, c) after homogenization.

The homogenized waveform data is generated via the zero-mean operation, namely every waveform along the travel path of laser will be built into a zero-mean signal by subtracting its own mean value, and thus the waveform amplitude inside a slice or among the different slices will become consistent. The strip noise that was obvious previously (Fig. 3b) has decreased. (Fig. 3c)

3.3 Examine neighbourhood relation

Test scene was scanned by an experimental laser sensor in the order of e-r slice (elevation angle – range); the recorded waveform cuboid was also organized in the structure of e-t slice image, therefore we decided to utilize the local neighbourhood relation in this 2D image, namely to examine whether there are pulses hiding in waveforms of the local neighbourhood tending to demonstrate an identical geometry and enhance each other by mutual information, e.g. lying in an regular geometric pattern, in our algorithm this geometric pattern is limited to plane, namely straight line in 2D waveform slice.

We create a $1 \times 14 \times t$ (14 pixels in E axis is a little larger than minimum size of plane we assume; t is equal to number of range values of waveform) mask and shift it inside a slice along elevation angle direction (Fig.4 b), the step size of mask shift is half of mask size in y axis to guarantee 50% overlapping. By shifting this mask, we can process all the recorded waveforms successively without any prior information given beforehand.

While shifting the mask along the y axis, it is obvious that we have to distinguish between two different situations for considering the local neighbourhood relation – waveforms of detected pulse and those of undetected pulse.

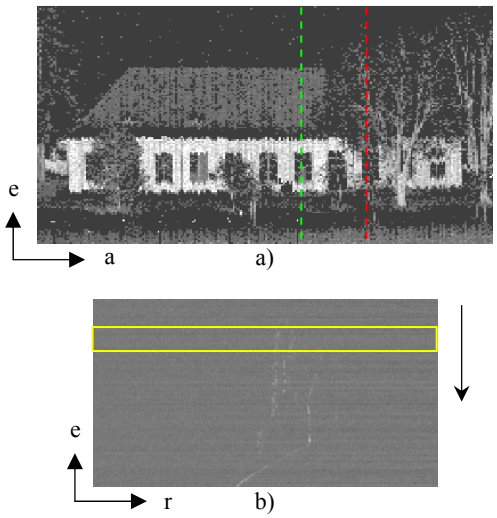


Figure 4. Mask shifting operation a) amplitude image generated by peak detection (e-a view, black pixels indicate waveforms of undetected pulse), b) e-r slice image marked by red dotted line, yellow rectangle denotes the mask, the black arrow denotes shift direction

3.3.1 Waveform of detected pulse

For the waveforms where pulse is already detected, its features, such as range, width and amplitude, are available by fitting Gaussian curve into pulse signals simultaneously. We can use the coordinate information of scattered points, which is represented in the vector format, to yield the local neighbourhood relation directly.

Principal component analysis (PCA) has been chosen here to accomplish this task. We regard three coordinates [x y t] as observed variable $\mathbf{x}$, and fit an n-dimensional hyperplane in 3-dimensional space ($n < 3$). The choice of r is equivalent to choosing the number of components to retain in PCA. When we want to evaluate whether the detected pulses are located in a straight line to build a local geometric pattern and how good they do, the variance of each component can act as the measure for it, because each component explains as much of the variance in the data as is possible with the relevant dimension.

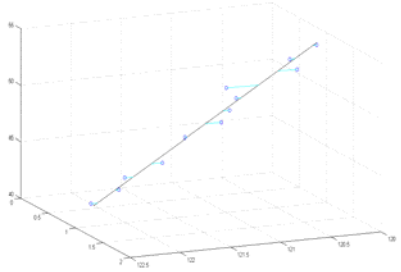


Figure 5. Local neighbourhood relation (line) found for detected pulses (blue circle) via PCA

The latent roots (or eigenvalues of covariance matrix $\mathbf{C}$) from the PCA define the amount of explained variance for each component, and the proportion of each variance can be derived:

$$\mathbf{C} = \mathbf{E}[\mathbf{B} \otimes \mathbf{B}] = \frac{1}{N}\mathbf{B} \cdot \mathbf{B}^*$$

$$\mathbf{V}^{-1}\mathbf{C}\mathbf{V} = \mathbf{D} \qquad (2)$$

where $\mathbf{B}$ is mean-subtracted matrix of $\mathbf{x}$, $\mathbf{D}$ is the diagonal matrix of eigenvalues of $\mathbf{C}$, $\mathbf{V}$ is matrix of eigenvectors diagonalizing the covariance matrix $\mathbf{C}$.

$$\begin{aligned} \text{roots} &= \mathbf{diagonal}(\mathbf{D}) \\ \text{propoVAR} &= \text{roots/sum(roots)} \\ &= [0.998 \quad 0.001 \quad 0] \end{aligned} \qquad (3)$$

As showed above the variance of first component has held a dominant proportion against other components according to propoVAR, the straight line is the best 1-D linear approximation to the data (Fig 5).

3.3.2 Waveform of undetected pulse

For another group of waveforms where pulse is not detected yet, we must deal with whole waveform signal instead of point cloud. Waveform stacking method is adopted to locate potential geometric pattern within the mask.

Image stacking technique is used to increase the signal to noise ratio (SNR) of weak objects in the final output image. This basic idea can be easily transformed to the concept of waveform stacking. We take the 1D signal along laser ray as the unit for waveform stacking, and try to identify prominent peak information in the stacked waveforms generated along specific slopes, which means existence of significant regular geometry such as plane. By stacking multiple waveforms on top of each other along different slopes, the weak pulse is expected to be enhanced against the random noise. The random noise will be counteracted with each other in spite of low SNR, whereas the true reflection information representing object features will be emphasised to some level, so they can be identified again.

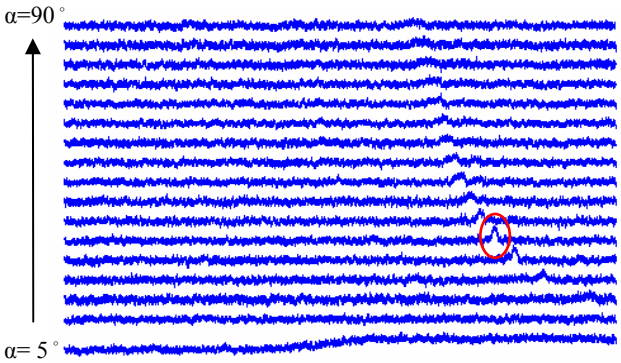


Figure 6. Waveform stacking along slope angles from $\alpha = 5^\circ$ to 90°, in steps of $\Delta\alpha = 5^\circ$ for roof area (from the slice marked by green dotted line in Fig 4a). Red ellipse marks the maximal peak information of the best stacked waveform.

Through waveform stacking, we can obtain a series of stacked waveforms corresponding to various stacking slope angles (Fig.6). When there is only one significant distribution of regular geometry like straight line, only one maximal peak signal interval of certain stacking slope angle corresponding to local geometry can be identified, and the result of waveform stacking along this slope is called best stacked waveform. If there are barely or multiple distinct peak signals, we fail to find out the regular geometric pattern in local neighbourhood due to no or ambiguity of extremum distribution, such as sky and holes and volume scatters (tree leaves).

In Fig 7 the cyan curve denotes the maximal value of stacked waveform along each slope toward stacking angle. The red curve is the smoothed version of the cyan curve. The approximation of red curve by cubic is plotted in magenta. Thus, the slope angle of best stacked waveform can be improved by locating maximum of fitting curve with "sub-pixel" accuracy. The green dotted line indicates the max value of cubic.

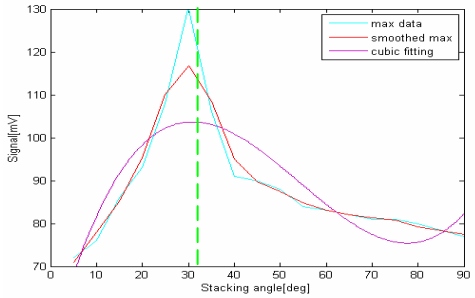


Figure 7. Maximum values of stacked waveforms vs. stacking slope angle

3.4 Find the prominent geometric pattern

After examining the local neighborhood relation for waveforms of detected and undetected pulses respectively, we have to consider them as an entirety. The local neighbourhood where both kinds of pulses coexist is examined by shifting a mask (Fig.8a). The prominent geometric pattern for a mixed set of waveforms (detected and undetected pulses) within the mask has to be identified and is to be recorded in a data list $\{\mathbf{P}_i\}$.

First we have to distinguish between three different situations encountered while shifting the mask (Fig.8b):

a) Waveform of undetected pulse almost occupies the mask, Number (Num) of $\{\mathbf{waveform\ of\ no\ pulse}\} > 9$

b) Waveform of detected pulse almost occupies the mask
Num of $\{\mathbf{waveform\ of\ pulse}\} > 9$

c) Both kinds of waveforms appear balanced
Num $\{\mathbf{waveform\ of\ pulse}\} < 9$ &Num $\{\mathbf{waveform\ of\ no\ pulse}\} < 9$

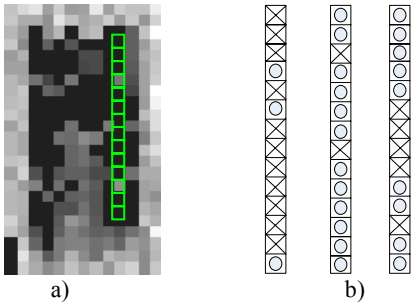


Figure 8. a) Example of mask shifted over waveform cuboid (x-y view), b) three typical situations encountered, circle indicates waveform of pulse detected, and cross indicates waveform of no pulse

Algorithm Find out prominent geometric pattern from the waveforms delimited by mask

Input: Waveform slice, label array by point cloud
Initialize Prominent geometric pattern list $\{\mathbf{P}_i\}$
while $\{\mathbf{S}\}$ **is not reached**
  if Num $\{$**waveform of no pulse**$\} > 9$ **then**
    Examine geometric pattern via waveform stacking
  else if Num $\{$**waveform of pulse**$\} > 9$ **then**
    Examine geometric pattern via PCA
  else
    Examine geometric pattern for two kinds of waveform respectively, and define a multiple neighborhood relations for waveforms
  end if
  end if
  if the geometric pattern is found **then**
    Add to prominent geometric pattern list $\{\mathbf{P}_i\}$ for this local mask
  end if
end while
Return $\{\mathbf{P}_i\}$

The algorithm described above has given us prominent geometric pattern list $\{\mathbf{P}_i\}$ as result. If a multiple neighborhood relation is defined for 3rd situation, the waveforms will be treated in terms of different geometric patterns. For some local waveform groups covered by the shifting mask, there may be no prominent geometric pattern found, e.g. sky or holes, the next processing step (Fig.1-5) will be skipped and the likelihood value for assessing weak pulses is set to zero directly.

3.5 Contribution determination of single waveform

In the last step of the whole algorithm, a measure should be defined to evaluate relation between the single original waveforms and local geometric pattern found by analyzing the local neighborhood relation. This measure quantifies how the single waveform contributes to building the prominent geometric pattern and can be further perceived as the likelihood value for assigning the corresponding waveform as pulse or not.

On the basis of considerations above, the correlation coefficient ρ between both waveform sections is calculated and used to serve as the measure, which describes similarity between the shape of waveforms to be compared.

$$\rho(i,j) = \frac{\mathrm{cov}(i,j)}{\sqrt{\mathrm{var}(i,i) \times \mathrm{var}(j,j)}} \quad \textbf{and} \quad -1 \le \rho(i,j) \le 1 \qquad (3)$$

For those local neighbourhoods whose prominent geometric pattern is built by waveforms of undetected pulse, the peak signal fraction will be cropped from the averaged stacked waveform (section between two green lines in Fig.9). We compare this peak signal fraction to the corresponding section of original waveform through correlation in order to acquire the contribution of every original waveform to the best stacked waveform. The same procedure is performed towards another kind of local neighbourhood, only the averaged stacked waveform will be replaced by averaged waveform of detected pulse. As a result of existence of overlapping areas through shifting the mask and multiple neighborhood relations, the identical waveform may be repeated to correlate with multiple prominent geometric patterns, from which the maximal correlation coefficient will be chosen as the likelihood value for this waveform.

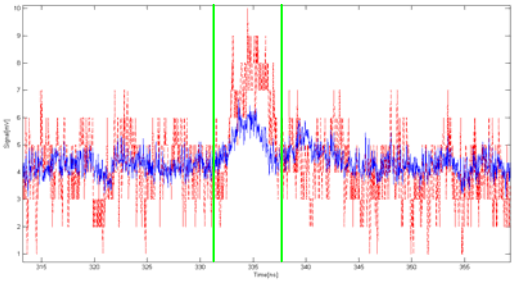


Figure 9. Original waveform (red) plotted overlaid with averaged stacked waveform corresponding to the prominent geometric pattern.

The hypothesis of weak pulses to be detected or not can be verified based on the likelihood value. This is the problem of two-class classification; one has to make a decision between two hypotheses either by thresholding empirically or based on estimation of likelihood function.

4. RESULTS

The presented algorithm was applied to the waveform data cuboid acquired by an experimental laser system. According to the correlation coefficient, detection of weak pulses can be achieved by two-class classification. For those experiments, where we do not have any priori-knowledge or assumption concerning pulses expected to be detected, we decide pulse detected or not by selecting threshold empirically. In order to make the detection results more flexible and avoid crisp decision, up (t_u) and bottom (t_b) thresholds are to be set, if $\rho(waveform) \le t_b$ then accepted as detected pulse, if $\rho(waveform) \ge t_u$ then fail to detect pulse, if $t_b < \rho(waveform) < t_u$ remain to be checked further, and so we used a traffic-light paradigm to visualize the detection result which is shown in Fig 10. The most signals lying among roof region (without occlusion) are classified as detected pulse requiring no further check, whereas the most ones in the sky belong to the red category rejected directly(Fig.10). This result corresponds to our expectation and demonstrates reasonability of the algorithm.

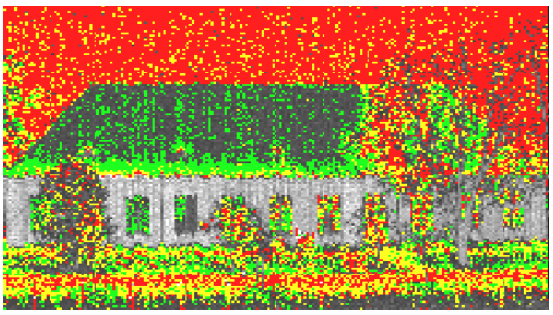


Figure 10. Result of weak pulse detection (x-y view), Red: no pulse; Yellow: uncertain; Green: pulse

For two critical parts of the test scene – window and roof behind trees, a number of signals have been detected, thus the object feature can also be recovered (Fig.11a, b), e.g. the detected weak signals almost lie in one plane, moreover, the window plane formed by detected pulses appears to lie in some offsets backwards with respect to wall; The outline of the roof partially occluded by tree is to certain degree recovered. Many signals reflected from the ground have failed to be detected

perhaps due to poor local neighbourhood relation and very weak reflection.

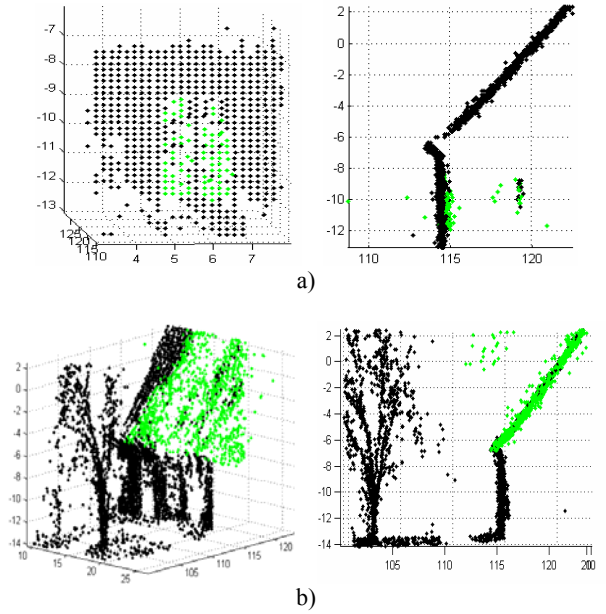


Figure 11. Results of local sections of test scene a) point cloud of window section with detected weak pulses showed in green from front and side view, b) section of roof behind the tree

After making a decision on three categories, the yellow category will be delivered to a further check based on SNR, so that we can finally obtain a result of only two classes - detected or not detected. In Fig.12 the relative histograms of correlation coefficient of two classes are depicted overlaid with each other. The Gaussian curve used to fit to the histograms can be perceived as an approximation of likelihood function for each hypothesis, few overlapping area proved separability of two classes and appropriateness of correlation coefficient as feature value.

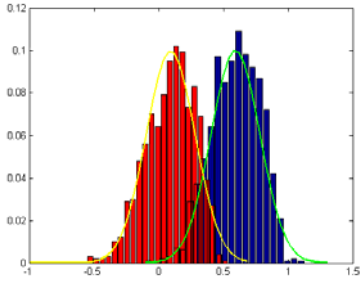


Figure 12 Overlapping relative histograms of correlation coefficients of two classes – pulse detected (blue) and not detected (red), with corresponding Gaussian fitting.

5. CONCLUSION

In this work an automatic approach for extracting weak laser pulses from full-waveform laser data has been presented. The algorithm uses waveform stacking to analyze the local neighbourhood of laser signals. Hypotheses for planes of different slopes are generated and verified. Each signal is assessed by a likelihood values with respect to accepted hypotheses. This contribution measure to local geometry is used for the subsequent operation of detection. The results on waveform data acquired from urban area show ability to detect partially occluded objects or objects with poor surface response, which can not be geometrically predicted by previous detected point cloud. In this paper only the neighbourhood relation within a vertical slice was used, Future work will focus on hypothesis generation combining neighbouring vertical slices.

References

Brenner, A.C., Zwally, H.J., Bentley, C.R., Csatho, B.M., Harding, D.J., Hofton, M.A.,Minster, J.B., Roberts, L.A., Saba, J.L., Thomas, R.H., Yi, Y., 2003. Geoscience Laser Altimeter System (GLAS) —derivation of range and range distributions from laser pulse waveform analysis for surface elevations, roughness, slope, and vegetation heights. AlgorithmTheoretical Basis Document—Version 4.1. http://www.csr.utexas.edu/glas/-pdf/Atbd_20031224.pdf (Accessed March 1, 2007).

Der, S., Redman, B., Chellappa, R., 1997. Simulation of error in optical radar measurements. Applied Optics 36 (27), pp. 6869-6874.

Hofton, M.A., Blair, J.B., 2002. Laser altimeter return pulse correlation: A method for detecting surface topographic change. Journal of Geodynamics special issue on laser altimetry 34, pp. 491-502.

Hofton, M.A., Minster, J.B., Blair, J.B., 2000. Decomposition of laser altimeter waveforms. IEEE Transactions on Geoscience and Remote Sensing 38 (4), pp. 1989–1996.

Hug, C., Ullrich, A., Grimm, A., 2004. LITEMAPPER-5600 - a waveform digitising lidar terrain and vegetation mapping system. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences 36 (Part 8/W2), pp. 24–29.

Irish, J.L., Lillycrop, W.J., 1999. Scanning laser mapping of the coastal zone: the SHOALS system. ISPRS Journal of Photogrammetry & Remote Sensing 54 (2-3), pp. 123-129.

Jutzi, B., Stilla, U., 2003. Laser pulse analysis for reconstruction and classification of urban objects. In: Ebner, H., Heipke, C., Mayer, H., Pakzad, K. (Eds) Photogrammetric Image Analysis PIA'03. International Archives of Photogrammetry and Remote Sensing 34 (Part 3/W8), pp. 151-156.

Jutzi, B., Stilla, U., 2006. Range determination with waveform recording laser systems using a Wiener Filter. ISPRS Journal of Photogrammetry and Remote Sensing 61 (2), pp. 95-107.

Reitberger, J., Krzystek, P., Heurich, M., 2006. Full-Waveform analysis of small footprint airborne laser scanning data in the Bavarian forest national park for tree species classification. In: Koukal, T., Schneider, W. (Eds) 3D Remote Sensing in Forestry, pp. 218-227.

Söderman, U., Persson, Å., Töpel, J., Ahlberg, S., 2005. On analysis and visualization of full-waveform airborne laser scanner data. Laser Radar Technology and Applications X. In: Kamerman, W. (Ed) SPIE Proceedings Vol. 5791, pp. 184-192.

Wagner, W., Ullrich, A., Melzer, T., Briese, C., Kraus, K., 2004. From single-pulse to full-waveform airborne laser scanners: Potential and practical challenges. In: Altan, M.O. (Ed) International Archives of Photogrammetry and Remote Sensing 35 (Part B3), pp. 201-206.

EXPLOITING SPATIAL PATTERNS FOR INFORMAL SETTLEMENT DETECTION IN ARID ENVIRONMENTS USING OPTICAL SPACEBORNE DATA

Mattia STASOLLA and Paolo GAMBA
Department of Electronics, University of Pavia
Via A. Ferrata, 1, I-27100 Pavia, Italy
{name.surname}@unipv.it

KEY WORDS: Informal settlements, urban remote sensing, regional mapping

ABSTRACT

In this paper human settlement detection using SPOT data in arid environments is addressed, with stress on informal settlement analysis. We show that a proper use of spatial patterns may improve the delineation of the extent of these areas and the discrimination between formal and informal settlements. A comparison with existing global data sets shows the potentials of this approach for human settlement mapping in arid environments.

1 INTRODUCTION

Unstructured human settlement mapping and monitoring is an important topic for many national and international initiatives including the European Global Monitoring for Environment and Security (GMES) initiative and the humanitarian and development aid policies of the United Nations. Also, monitoring settlements is useful to acquire information on phenomena, like illegal immigration, that are very high-ranked on any current political agenda. The application of these techniques has a global scope and would be particularly relevant for the developing world.

Unstructured ("informal") human settlements are usually defined (Mason and Fraser, 1998) as dense settlements with groups living in self-constructed shelters without any formal structure which is usual in urban areas. Often, no real subdivision of the land is done, and these areas are characterized by rapid, unstructured and unplanned development.

Spatial technology may help a lot in analyzing the patterns of these settlements, forecasting their possible changes and provide information on how to make the living conditions in these areas much better than they are today. However, for spatial technology to be effective in informal settlement environments, it has to be cheap, both in data acquisition and processing, as automated as possible to achieve faster and more reliable results, simple to use and largely based on tested routines and algorithms. Nevertheless, traditionally, field survey and visual interpretation of satellite data are used to produce reliable information. These are, to a large extent, manual operations and require a wide expertise; moreover, besides the operator skill bias, they are time-consuming and hard to catch up with the rapid developing pace.

The aim of this research is to develop a semi-automatic algorithm for the extraction of such human settlements.

2 STRUCTURED AND UNSTRUCTURED SETTLEMENT DISCRIMINATION

The overall problem discussed in this paper may be subdivided into two sub-problems. The first one is the discrimination of human settlements against their surrounding natural environment. The second one is the detection of informal settlements as a subset of the previously identified areas. According to this scheme, the challenge of informal settlement detection using satellite remote sensing data is also twofold. On the one hand, a robust algorithm for detecting human settlement areas in these images is required. On the other hand, the methodology must be flexible enough to incorporate very different settlement environments, and be able to discriminate among them.

A first comment, based on current technical literature and the long term aim to exploit all available sources of data, is that spatial patterns more than spectral features have to be investigated. With the improvement in spatial resolution, between-class spectral confusion and within-class spectral variation were found to increase for land cover/ land use studies (Barnsley and Barr, 1996). Thus, spatial information gets essential to reduce mapping confusion, especially in areas where the settlements are realized using locally available and very diverse materials.

A second comment is that simple classification procedures are highly desirable, but very complex to design, due to the wide variety of settlement characteristics, both spatially and spectrally. Even working at a regional level, where significant correlation between settlement areas exists, the goal of a uniquely tailored automatic algorithm may be very difficult to achieve.

A general overview of literature survey about this topic shows a sort of rift within the choice of sensors to be used for urban detection, as described hereafter. Before the launch of VHR satellites one of the most employed sources of data were SPOT satellites. For such imagery it is impossible to identify every single house because of the small size of the objects, and the most common approach is based on statistical textures, which take into account the spatial distribution and variation of neighborhood pixel values within an appropriate moving window. There are a few methods to quantify them, from Grey-Level Co-occurrence Matrix (GLCM) (Haralick *et al.*, 1973) to Markov Random Fields (MRF) (Descombes *et al.*, 1999). Even though the MRF approach provides good performances, it is not considered here, since it requires a significant computational part that does not fulfil the aforementioned requirement of easiness. Proposed algorithms based on the GLCM com-

putation cause instead other remarks. First of all, they basically focus on single locations (Zhang *et al.* (2001)-Shaban and Diksit (1999)), and thus they do not allow any generalization of the results. Moreover, they are based on supervised techniques,make use of very diverse texture sets, and lead to overall accuracies widely variable between 40% and 80%.

Since 2000, despite the launch of the fifth satellite of the family, SPOT spatial capabilities have been overshadowed by the availability of the aforesaid VHR sensors: in the urban analysis field, researchers are more interested in exploiting the sources that can theoretically lead to best performances. Some internationally funded projects are at the moment providing satellite data to extract suitable information about settlements from remotely sensed imagery. For example, informal settlement detection, such as the mapping of refugee camps, has been addressed within the *GMOSS* (Global Monitoring for Stability and Security) network. In the framework of this project the assessment of suitable indicators for refugee camp detection were computed by means of night-time sensors' data or VHR satellite imagery. Moreover, the European Union Joint Research Center carried out a comparison among four different methods (supervised and unsupervised classification, multi-resolution segmentation and morphological analysis) aiming at single tent detection in order to deduce the population of the Lukole camp, Tanzania (Giada *et al.*, 2003). For this task, methods based on interpretation by a human operator are practically not feasible because of the large number of tents, and computer-based techniques are the only alternative.

Nevertheless, it is worthwhile to consider that, in many cases, the full spatial resolution available from VHR sensors is not really required, and a trade-off among accuracies, tasks, easiness and generalization should be addressed.

3 ALGORITHM OVERVIEW

An interesting technique developed for urban settlement detection and characterization in the past and applied mainly to satellite SAR data is based on a supervised neural network classification chain, with spectral and spatial analysis steps (Gamba and Dell'Acqua, 2003). Additionally, spatial analysis, according to previous comments, may be performed using textural features, possibly extracted using a locally adaptive window width (Gamba *et al.*, 2006).

In this work a simpler, but equally effective, methodology has been developed and tested for arid environments, with the aim of a better automatization of the process. The procedure is intended to a general purpose detection of settlements, therefore, besides the simplicity, it is fast and easily repeatable. In particular, the co-occurrence matrix textural features suggested by Haralick (Haralick *et al.*, 1973) form the basis for the analysis. On the one hand, they allow adopting an unsupervised approach (and thus to cope with the significant problems due to the lack of knowledge about most of the developing areas), but, on the other hand, they reduce the application fields of the system, since they

Figure 1: The urban test areas in Sudan considered in this work.

only permit a very poor (in terms of class number) land-use classifications of the images. Actually they allow to detect with high precision the boundaries and the extent of the settlements, both formal and informal; to some extent it is possible to separate this two classes, but just in case they present different building densities. In general this condition is verified, even if there are some cases where this density threshold is not clear-cut, for example in the city boundaries. Anyway, such a confusion in the results does not necessarily imply they are useless, since they give a first screening of the data and have an important role in assisting human experts.

The algorithm is modular, but sequential. The basic step - the detection of the settlements - starts from the computation of the Grey Level Co-occurrence Matrix (GLCM) of the original SPOT-5 image - resized by a factor of 2 to reduce computational cost - with moving window 21×21 and displacement vector (1,1). The choice of the moving window size (no considerations have been made upon the standard displacement) comes from the evaluations of the average dimensions of usual building blocks in many towns all over the world.

Among the eight most common co-occurrence features, *Homogeneity* is required. It belongs to the "contrast" group and it assigns high values to homogenous areas. In arid backgrounds, which are very uniform, it points the settlements out very efficiently.

This feature is then fed to an unsupervised classifier (*K-Means*) with a fixed number of output classes. This number is expected to range between 3 and 10, a good choice being 5. The crucial point is the merging of the output classes into the two macro-classes *City* and *Soil*, which represent the settlements and the background. The required knowledge about the test area for this merging operation is very scanty, especially in respect with the number of training points needed by supervised classifiers. In any case a good choice is to merge into the *City* category the first two classes

found by the K-Means algorithm, if the order is defined by increasing values of class centroids.

The second step of the algorithm tries and discriminates the city core from the refugee camp. Since the scale and the resolution do not allow to detect the single building (or tent), the only way for this purpose is to find differences in the texture properties of building agglomerations. From a theoretical point of view, buildings - in the common sense of the word - and tents or shacks have different geometrical properties and texture features. Actually, at this resolution, they are not properly defined, thus there is an objective difficulty in finding selection criteria. To this aim, the algorithm exploits one more feature, computed using the same parameters as before: *Variance*. Variance measures the dispersion of the values around the mean,and is useful since, in general, the impact of the camp on the environment is lower than the town. This implies that variations around the mean of the camp are not significant, and it appears darker than the city, which is more heterogenous. By classifying this feature using once again the *K-Means* algorithm it is possible to extract the informal settlement position.

4 EXPERIMENTAL RESULTS

The proposed procedure was evaluated using a test set composed by panchromatic SPOT-5 images of informal human settlements areas. The test sites are the towns of Al-Fashir and Nyala, respectively the capitals of Northern and Southern Darfur, Sudan, Africa (see Fig. 1). As a result of the war situation in that region since 2003, vast tent camps has been organized to accommodate thousands of refugees. The Abu Shok camp is located North-West from Al-Fashir, and its development is still occurring. Similarly, the refugee camp of Intifida is located East of Nyala, far away from the town center.

A visual analysis of the data sample in Fig. 2(a) shows that there is confusion among the materials of the human settlements (both formal and informal) and the surrounding natural environment. Therefore, it confirms that pixel values might not be enough to detect urban areas, let alone to efficiently discriminate more structured urban areas from informal settlements. As proposed in the previous section, more refined results may come from an analysis including spatial relationships between intensity values.

In the Al Fashir area the data set consists of two panchromatic SPOT-5 images with 2.5 m spatial resolution that were acquired on November 14, 2004 and April 7, 2005.

Results presented in Table 1 show that both images can be used to detect human settlements with very high precision, over 90%, fully exploiting arid background, with completely different pattern than the populated area.

Even with worse results, the obtained map would be at present the best chart available for such territories, as shown in Fig. 2(c) and (d), which depicts the existing maps of the

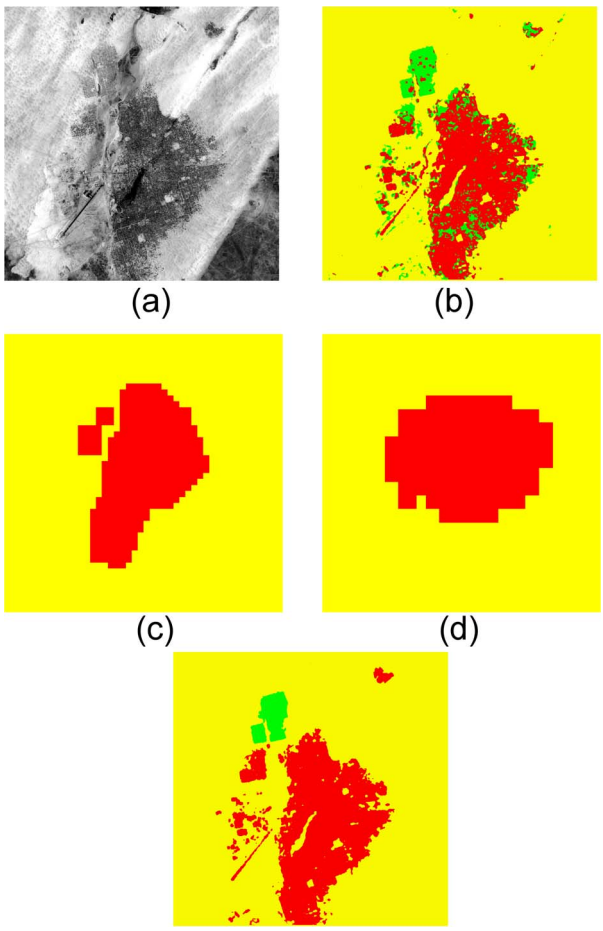


Figure 2: Al Fashir: (a) SPOT image, (b) classification results (green = camp, red = city, yellow = desert), to be compared with Africover (c) and GRUMP (d) maps, as well as the manually extracted ground truth in (e).

K=0.90	City	Soil
91.65%	1195580	108951
98.08%	95035	4850434
96.73%	92.64%	97.80%

K=0.97	City	Soil
95.43%	1290284	61793
99.99%	331	4897592
99.00%	99.97%	98.75%

Table 1: Confusion matrices for the two SPOT images of Al Fashir (2 classes), using the manually extracted GT.

K=0.87	City	Soil	Camp
92.82%	1064235	72814	9452
98.08%	74114	4850416	20939
47.37%	47046	36133	74851
95.83%	89.78%	97.80%	71.12%

K=0.87	City	Soil	Camp
97.64%	991209	13483	10504
99.99%	309	4897592	22
28.12%	193877	48288	94716
95.73%	83.62%	98.75 %	90.00%

Table 2: Confusion matrices for the two SPOT images of Al Fashir (3 classes), using the manually extracted GT.

city obtained from global and regional databases, Africover and GRUMP (Global Rural-Urban Mapping Project).

After detecting the settlement position, the algorithm tries and identifies different building densities (Fig. 2b). The quantitative evaluation in Table 2 shows that, as before, the classifier suffers the intrinsic confusion between shacks and outskirts, and there is a high commission error for the *Camp* class.

As for the Nyala area a second data set was employed: a panchromatic SPOT-5 image and a Quickbird image. The main difference with the first data set consists in the presence of more arid land vegetation. Unfortunately, shrubs have the similar texture properties than settlements at SPOT spatial resolution, leading to a significant mix-up even in case of two classes. The numerical evaluations of the confusion matrix in the upper part of Table 3 confirms the high commission error for the *City* class. As the first algorithm step is not effective, also the Formal/Informal discrimination cannot give useful outcomes. Lower part of Table 3 indeed reports an overall accuracy of 72% and a very low K index.

The first comment relates to the fact that the only available input was a panchromatic SPOT image: spatial resolution is high but, conversely, spectral information is poor. Besides ancillary data like GIS data and cadastral maps, a basic way to improve detection results would be more spectral information. Unfortunately, the only multi-spectral available data was a Quickbird pan-sharpened image that was acquired one year before and does not cover the exact zone (see Fig. 3a). Anyway, since refinement procedure described afterwards is very general, it is possible to process the QB image and then extend the results to the SPOT

K=0.40	City	Soil	
36.84%	759676	1302505	
97.55%	102699	4092982	
77.54%	88.09%	75.86%	

K=0.32	City	Soil	Camp
36.32%	382299	661907	8372
97.55%	55821	4092982	46878
8.48%	283354	640598	85651
72.88%	52.99%	75.86%	60.79%

Table 3: Confusion matrices for SPOT data of Nyala, with two or three classes, using the manually extracted GT.

K=0.55	City	Soil
52.97%	353481	313895
96.57%	45710	1286914
82.01%	88.55%	80.39%

K=0.66	City	Soil
64.32%	347753	192939
96.48%	51438	1407870
87.78%	87.11%	87.95%

Table 4: Confusion matrices for Quickbird data of Nyala without and with NDVI masking.

data. Basically, the trick is to mask the vegetation pixels within the original scene, first evaluating the NDVI index, then removing noise blobs (in our case those with area less then 100 pixels) and finally computing the *Homogeneity*. The further testing data set results are depicted in Fig. 3(b) and (c): the refugee camp is not included and the whole city has been captured. Confusion matrix In Table 4 shows the results after the original processing scheme, and (lower part) after the masking procedure. With respect to the previous case, overall accuracy is better, since the considered scene contains less textural zones that can lead to confusion in the final classification. In any case, the purpose of this section is to evaluate the quality of the refinements: the percentage comparison points out an effective improvement of the accuracy and the K coefficient, that approaches the typical quality gauge of 0.7.

5 CONCLUSIONS

In this paper we proposed a semi-automatic procedure to detect settlements in arid environments exploiting spatial patterns.

The results of this work show that panchromatic information by the sensors mounted on SPOT satellites are not immediately suited for our aim. The introduction of textures, however, considerably improves results, especially in differentiating settlements from arid areas. The main shown drawback is the confusion in classification between the patterns of groups of buildings and shrubs and/or rock aggregate ones. A first attempt to avoid this problem has been addressed, lacking SPOT multi-spectral information, by means of a Quickbird pansharpened image. More tests on SPOT imagery must be carried out to confirm the achieved improvements.

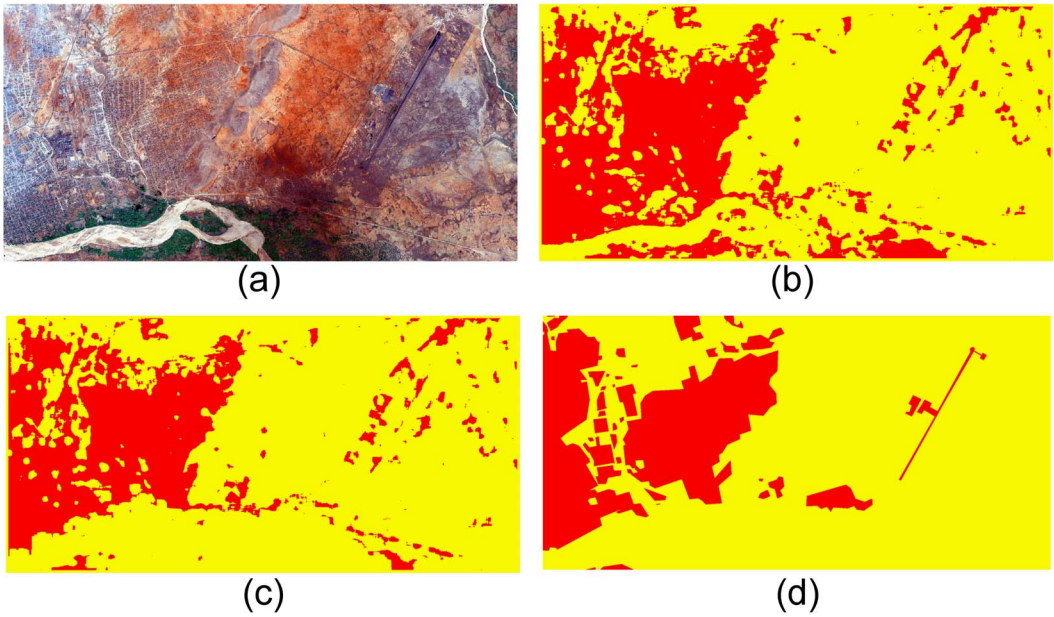


Figure 3: Nyala: (a) Quickbird image, and classification results (red = city, yellow = desert) without (b) and with (c) vegetation masking using NDVI, to be compared with the manually extracted ground truth in (d).

ACKNOWLEDGMENT

The authors would like to thank JRC, and especially M. Pesaresi and C. Louvrier, for providing the Quickbird image of Nyala, and the *O.A.S.I.S.* (Optimising Access to Spot Infrastructure for Science) Programme for all SPOT data sets.

We also thank F. Dell'Acqua for the useful discussions about this work.

References

S.O. Mason and C.S. Fraser, "Image sources for informal settlement management", *Photogrammetric Record*, Vol. 16, n. 92, pp. 313-330, 1998.

M. Barnsley, and S. Barr, "Inferring urban land use from satellite sensor images using kernel-based spatial reclassification", *Photogrammetric Engineering and Remote Sensing*, Vol. 62, n. 8, 949-958, 1996.

R. Haralick, K. Shanmugam, I. Dinstein, "Textural features for image classification". *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 3, n. 6, 610-621, 1973.

X. Descombes, M. Sigelle, F. Preteux, "Estimating Gaussian Markov random field parameters in a nonstationary framework: application to remote sensing imaging", *IEEE Transactions on Image Processing*, Vol. 8, n. 4, 490 - 503, April 1999.

Q. Zhang; J. Wang; P. Gong; P. Shi, "Texture analysis for urban spatial pattern study using SPOT imagery", *Geoscience and Remote Sensing Symposium, IGARSS '01*, Vol. 5, 2149 - 2151, 2001.

M. Shaban, O. Dikshit, "Evaluation of merging SPOT multispectral and panchromatic data for classification of urban environment", *1999 IEEE Geoscience and Remote Sensing Symposium, IGARSS '99*, Hamburg (Germany), Vol. 2, pp. 1214 - 1216, 1999.

S. Giada, T. De Groeve, D. Ehrlich, and P. Soille, "Information extraction from very high resolution satellite images over Lukole refugee camp, Tanzania", *International Journal of Remote Sensing*, Vol. 24, n. 22, 4251-4266, November 2003.

P. Gamba, F. Dell'Acqua, "Improved multiband urban classification using a neuro-fuzzy classifier," *International Journal of Remote Sensing*, Vol. 24, n. 4, pp. 827-834, Feb. 2003.

P. Gamba, F. Dell'Acqua, G. Trianni, "Semi-automatic choice of scale-dependent features for satellite SAR image classification", Pattern Recognition Letters, Vol. 27, n. 4, pp. 244-251, Mar. 2006.

AFRICOVER, http://www.africover.org

GRUMP, http://beta.sedac.ciesin.columbia.edu/gpw/

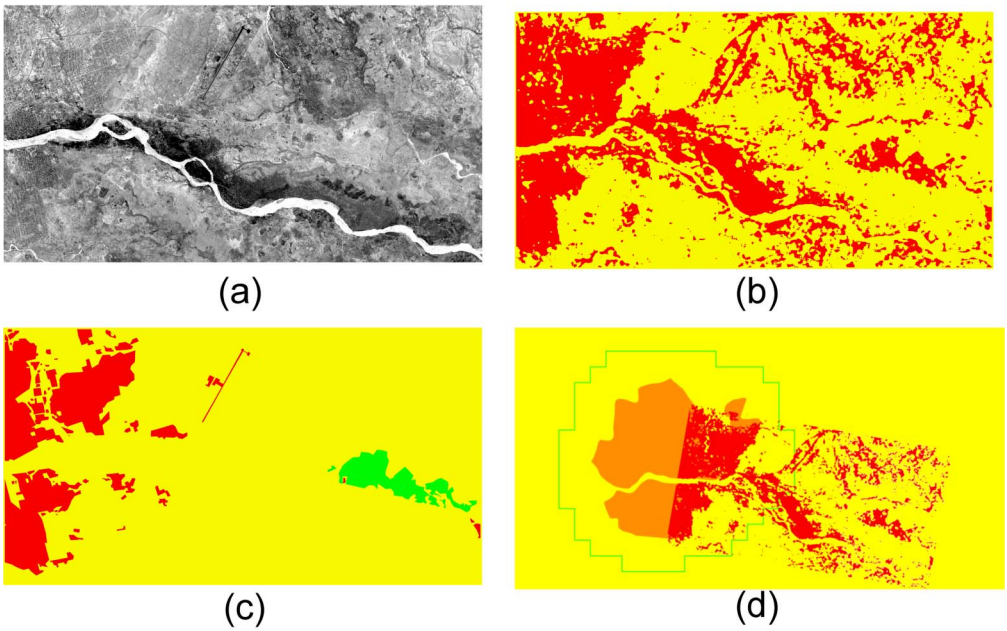


Figure 4: Nyala: (a) Spot image, and (b) classification results (red = city, yellow = desert), to be compared with the manually extracted ground truth in (c), and GRUMP/Africover (blu/red) urban extents in (d).

CONTINUOUS SELF-CALIBRATION AND EGO-MOTION DETERMINATION OF A MOVING CAMERA BY OBSERVING A PLANE

Jochen Meidow, Michael Kirchhof

FGAN-FOM Research Institute for Optronics and Pattern Recognition,
Gutleuthausstr. 1, 76275 Ettlingen, Germany
{meidow|kirchhof}@fom.fgan.de

KEY WORDS: camera calibration, homography, parameter estimation, Kalman filter

ABSTRACT:

In many vision applications the depth relief of an observed scene is small compared with the extent of the image. Beside man-made environments these scenes may be approximated by a plane. Due to environmental influences camera parameters can gradually change which motivates the need for a continuous self-calibration. Based on the theory of recursive parameter estimation we present an update scheme for the parameters to handle endless video streams in real time. The geometric parametrization of the frame to frame homographies allows to incorporate information of other sensors. The application is the visual navigation of robots and unmanned aerial vehicles moving above nearly planar environments. The approach will be empirically evaluated with a synthetic data set and demonstrated with a real data set. A typical example is given by a real data set of an indoor robot observing the ground floor with one camera.

1 INTRODUCTION

Motivation. Many vision applications deal with a small depth relief of an observed scene compared with the extent of the image. Often these scenes can be approximated by a plane e.g. the ground plane. Especially man-made environments essentially consist of planes. Therefore a homography based approach is proper to model such configurations. The main application at our focus is an indoor robot observing the ground floor in order to determine its trajectory visually. The odometry of an indoor robot is often unreliable due to the slip of the wheels. Therefore image based determination of ego-motion should be used in addition to standard odometry.

Due to environmental influences camera parameters can gradually change. Therefore the investigation of odometry determined from a monocular sensor observing a ground plane requires continuous self-calibration. The application also requires to handle an endless video stream in real time. Therefore a method apart from the computational effort of bundle adjustment is needed.

Related Work. Fundamental work on geometric decomposition of homography was done in (Faugeras and Lustman, 1988). But this method requires a proper calibration of the used cameras. Camera self-calibration from views of a 3D scene has widely investigated, see for example (Pollefeys et al., 1999, Maybank and Faugeras, 1992). But it is known that these techniques in general fail for planar or almost planar scenes since they run into singularities. Nevertheless, self-calibration from planar scenes is possible (Triggs, 1998, Zhang, 1999, Malis and Cipolla, 2002).

In (Zhang, 1999) every available metric information of the scene is used and the calibration is determined by plane to frame homographies. A closed form solution is given and improved by non-linear optimization based on maximum likelihood criterion. The work was inspired by (Triggs, 1998). He computes constraints from the dual image of the absolute conic again from plane to frame homographies. Frame to frame homographies are used in (Malis and Cipolla, 2002) to build up a so-called super-collineation-matrix and to enforce multi-view constraints e.g. rank constraint. The cost-function for the self-calibration is then given by the difference of the eigenvalues of matrices with similar properties to the essential matrix. Fundamental work on relative orientation was done by Nistér. His solution to the five-point relative pose problem (Nistér, 2004) using the essential matrix can deal 3D as well as planar scenes. But the determination of the essential matrix requires calibrated cameras and the precision of the relative orientation is decreased when the scene becomes planar. Therefore the relative orientation computed from the essential matrix is preferable if the cameras move with known calibration in 3D environments.

Contribution. We present odometry visually determined from a monocular camera including update of the intrinsic camera parameters. We have to deal with endless video streams and have to ensure real time capability. The update of the intrinsic camera parameters is relevant because for example the focal length varies depending on the temperature. Nevertheless we can assume an initial guess because the used cameras are known.

It is a well known result in photogrammetry that the best results can be obtained by bundle adjustment (McGlone et al., 2004). On the opposite the relative orientation between subsequent frames can be computed very efficiently. One possible compromise is an incremental bundle adjustment. A typical method for an incremental bundle adjustment of homographies can be found in (Zelnik-Manor and Irani, 2002, Han and Kanade, 1998).

We introduce a different technique inspired by Kalman filtering (Welch and Bishop, 2003, Kalman, 1960), where the information of the past is subsumed in one parameter vector and its covariance matrix. The method reflects the possible smooth change of the calibration and also allows to incorporate other sensor information (e. g. GPS, INS, odometry) by using a geometric parametrization. The computational effort is comparable to adjustment over two homographies and is independent on the number of frames taken into account for estimating the current state.

Notation. For formulation and representation we use the framework of algebraic projective geometry. Homogeneous vectors and matrices will be denoted with upright boldface letters, e.g. $\mathbf{x}$ or $\mathbf{H}$, Euclidean vectors or matrices with slanted boldface letters, e.g. $\boldsymbol{x}$ or $\boldsymbol{H}$. In homogeneous coordinates '=' means an assignment or an equivalence up to an scaling factor $\lambda \neq 0$. Many

parameters have to be represented in various coordinate systems. Observations in the coordinate system S_k attached to the k-th frame are denoted by an upper index e.g. $\overset{k}{\boldsymbol{x}}$. Relative orientations or mappings between two cameras are written as $(\boldsymbol{R}_{kl}, \boldsymbol{t}_{kl})$ representing the motion from S_k to S_l.

2 MODELING

2.1 Basic Relations

For a 3D point $\boldsymbol{X}_i$ in the plane π the incidence relation

$$\boldsymbol{n}_k^\mathsf{T} \overset{k}{\boldsymbol{X}}_i - d_k = 0 \tag{1}$$

holds, where the plane is represented by its normal vector $\boldsymbol{n}_k$ and its distance d_k to the origin of the camera coordinate system S_k. The normal vector $\boldsymbol{n} = [n_X, n_Y, n_Z]^\mathsf{T}$ is oriented so that it points towards the camera, i.e. $n_Z \leq 0$. The representation of the point $\boldsymbol{X}_i$ in an other camera coordinate system S_l is equivalent to the coordinate transformation

$$\overset{k}{\boldsymbol{X}}_i = \boldsymbol{R}_{kl} \overset{l}{\boldsymbol{X}}_i - \boldsymbol{t}_{kl} \tag{2}$$

being a rigid 3D motion. Equations (1) and (2) reveal that the object points and their corresponding image points are related by the 2D homography

$$\overset{k}{\boldsymbol{X}}_i = \left(\boldsymbol{R}_{kl} - \boldsymbol{t}_{kl}\boldsymbol{n}_l^\mathsf{T}/d_l\right) \overset{l}{\boldsymbol{X}}_i = \boldsymbol{H}_{lk} \overset{l}{\boldsymbol{X}}_i \tag{3}$$

induced by the plane π. The rotation $\boldsymbol{R}_{kl}$ and the translation $\boldsymbol{t}_{kl}$ constitute the relative orientation between the two cameras with five parameters since only the ratio $\boldsymbol{t}_{kl}/d_l$ is determinable from (3), cf. figure 1. The decomposition of $\boldsymbol{H}_{lk}$ according to (3) has up to eight solutions which can be reduced to two reasonable solutions (Faugeras and Lustman, 1988).

Assuming straight-line preserving cameras the projection of the object points $\boldsymbol{X}_i$ is $\mathbf{x}_{ik} = \mathbf{K}_k \overset{k}{\mathbf{X}}_i$ with the homogeneous calibration matrix $\mathbf{K}_k$ introducing five additional intrinsic parameters per camera. Thus the corresponding 2D homography reads

$$\overset{k}{\mathbf{x}}_i = \mathbf{K}_k \boldsymbol{H}_{lk} \mathbf{K}_l^{-1} \overset{l}{\mathbf{x}}_i = \mathbf{H}'_{lk} \overset{l}{\mathbf{x}}_i \tag{4}$$

allowing to determine eight parameters from the correspondencies of one image pair. Note that the matrix $\boldsymbol{H}_{lk}$ — in contrast to $\mathbf{H}'_{lk}$ — can be decomposed into Euclidean entities.

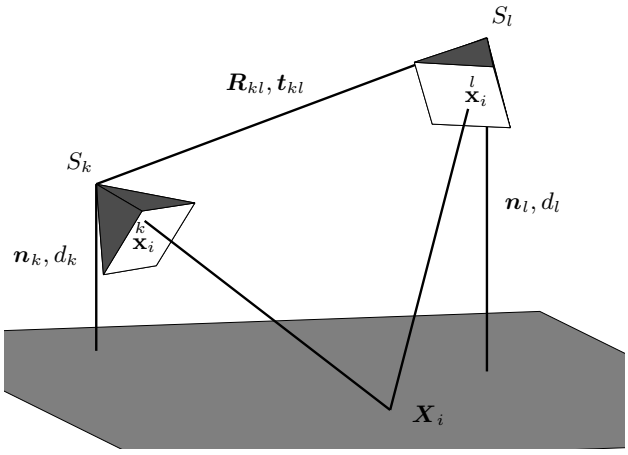


Figure 1: shows two cameras in general position and orientation observing a point $\boldsymbol{X}_i$ on the plane.

2.2 Adjustment Model

Functional Model. Without loss of generality we consider a parameter estimation for two groups of observations $\boldsymbol{l}_1$ and $\boldsymbol{l}_2$ being explicit functions of the unknown parameters $\boldsymbol{x}$, i.e. $\boldsymbol{l}_1 = \boldsymbol{f}_1(\boldsymbol{x})$ and $\boldsymbol{l}_2 = \boldsymbol{f}_2(\boldsymbol{x})$. This model is the basis for recursive and sequential parameter estimation schemes, where observations are added at later stage resulting in an update for the parameters.

With the additional restrictions $\boldsymbol{h}(\boldsymbol{x}) = \boldsymbol{0}$ for the parameters the corresponding linear model for the estimated entities reads (McGlone et al., 2004)

$$\boldsymbol{\Delta l}_1 + \widehat{\boldsymbol{v}}_1 = \boldsymbol{A}_1 \widehat{\boldsymbol{\Delta x}} \tag{5}$$
$$\boldsymbol{\Delta l}_2 + \widehat{\boldsymbol{v}}_2 = \boldsymbol{A}_2 \widehat{\boldsymbol{\Delta x}} \tag{6}$$
$$\boldsymbol{H} \widehat{\boldsymbol{\Delta x}} = \boldsymbol{c} \tag{7}$$

with the Jacobians $\boldsymbol{A}$ and $\boldsymbol{H}$, the estimated corrections $\widehat{\boldsymbol{v}}$, the differences $\boldsymbol{\Delta l} = \boldsymbol{l} - \boldsymbol{f}(\boldsymbol{x}_0)$ and $\widehat{\boldsymbol{\Delta x}} = \widehat{\boldsymbol{x}} - \boldsymbol{x}_0$, and the contradictions $\boldsymbol{c} = -\boldsymbol{h}(\boldsymbol{x}_0)$, evaluated at the approximate values $\boldsymbol{x}_0$.

Stochastic Model. We assume statistically independent observation groups, i. e. $\boldsymbol{\Sigma}_{l_1 l_2} = \boldsymbol{O}$, with the known covariance matrices $\boldsymbol{\Sigma}_{l_1 l_1}$ and $\boldsymbol{\Sigma}_{l_2 l_2}$. These covariance matrices are related to the true covariance matrices by an unknown variance factor σ_0^2 which can be estimated from the estimated corrections $\widehat{\boldsymbol{v}}$, see below. If the initial covariance reflects correctly the uncertainty of the observations, this variance factor is $\sigma_0^2 = 1$ (McGlone et al., 2004).

Normal Equations. Minimizing the squared and weighted sum of residuals

$$\Omega = \widehat{\boldsymbol{v}}_1^\mathsf{T} \boldsymbol{\Sigma}_{l_1 l_1}^{-1} \widehat{\boldsymbol{v}}_1 + \widehat{\boldsymbol{v}}_2^\mathsf{T} \boldsymbol{\Sigma}_{l_2 l_2}^{-1} \widehat{\boldsymbol{v}}_2 \tag{8}$$

under the linear constraints (7) the corresponding normal equation system becomes

$$\begin{bmatrix} \boldsymbol{N} & \boldsymbol{H}^\mathsf{T} \\ \boldsymbol{H} & \boldsymbol{O} \end{bmatrix} \begin{bmatrix} \widehat{\boldsymbol{\Delta x}} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \boldsymbol{h} \\ \boldsymbol{c} \end{bmatrix} \tag{9}$$

with the Lagrangian multipliers $\boldsymbol{\lambda}$ and

$$\boldsymbol{N} = \boldsymbol{A}_1^\mathsf{T} \boldsymbol{\Sigma}_{l_1 l_1}^{-1} \boldsymbol{A}_1 + \boldsymbol{A}_2^\mathsf{T} \boldsymbol{\Sigma}_{l_2 l_2}^{-1} \boldsymbol{A}_2 \tag{10}$$
$$\boldsymbol{h} = \boldsymbol{A}_1^\mathsf{T} \boldsymbol{\Sigma}_{l_1 l_1}^{-1} \boldsymbol{\Delta l}_1 + \boldsymbol{A}_2^\mathsf{T} \boldsymbol{\Sigma}_{l_2 l_2}^{-1} \boldsymbol{\Delta l}_2. \tag{11}$$

The solution of the system (9) is identical to the solution of the alternative normal equation system (McGlone et al., 2004)

$$\begin{bmatrix} \boldsymbol{A}_1^\mathsf{T} \boldsymbol{\Sigma}_{l_1 l_1}^{-1} \boldsymbol{A}_1 & \boldsymbol{A}_2^\mathsf{T} & \boldsymbol{H}^\mathsf{T} \\ \boldsymbol{A}_2 & -\boldsymbol{\Sigma}_{l_2 l_2} & \boldsymbol{O}^\mathsf{T} \\ \boldsymbol{H} & \boldsymbol{O} & \boldsymbol{O} \end{bmatrix} \begin{bmatrix} \widehat{\boldsymbol{\Delta x}} \\ \boldsymbol{\mu} \\ \boldsymbol{\nu} \end{bmatrix} = \begin{bmatrix} \boldsymbol{A}_1^\mathsf{T} \boldsymbol{\Sigma}_{ll} \boldsymbol{\Delta l}_1 \\ \boldsymbol{\Delta l}_2 \\ \boldsymbol{c} \end{bmatrix} \tag{12}$$

with the additional Lagrangian multipliers $\boldsymbol{\mu}$ and $\boldsymbol{\nu}$. This solution can address the situation $\boldsymbol{\Sigma}_{l_2 l_2} = \boldsymbol{O}$, that is one can fix parameters by setting their covariances to zero. Furthermore one can undo information by substituting $-\boldsymbol{\Sigma}_{l_2 l_2}$ with $\boldsymbol{\Sigma}_{l_2 l_2}$.

By applying the definition for a pseudo inverse, i.e. $\boldsymbol{\Sigma}^+ = \boldsymbol{\Sigma}^+ \boldsymbol{\Sigma} \boldsymbol{\Sigma}^+$ and $\boldsymbol{\Sigma} = \boldsymbol{\Sigma} \boldsymbol{\Sigma}^+ \boldsymbol{\Sigma}$, it can be shown that the equivalence of (9) and (12) holds for singular covariance matrices $\boldsymbol{\Sigma}_{l_2 l_2}$, too. Thus the weighted sum

$$\Omega = \widehat{\boldsymbol{v}}_1^\mathsf{T} \boldsymbol{\Sigma}_{l_1 l_1}^{-1} \widehat{\boldsymbol{v}}_1 + \widehat{\boldsymbol{v}}_2^\mathsf{T} \boldsymbol{\Sigma}_{l_2 l_2}^{+} \widehat{\boldsymbol{v}}_2 \tag{13}$$

becomes minimal subject to the constraints for the parameters.

Precision. The covariance matrix of the estimated parameters $\widehat{\boldsymbol{x}}$ results from the inverse normal equation matrices by

$$\begin{bmatrix} \boldsymbol{\Sigma}_{\widehat{x}\widehat{x}} & . \\ . & . \end{bmatrix} = \begin{bmatrix} \boldsymbol{N} & \boldsymbol{H}^\mathsf{T} \\ \boldsymbol{H} & \boldsymbol{O} \end{bmatrix}^{-1} \tag{14}$$

or

$$\begin{bmatrix} \boldsymbol{\Sigma}_{\widehat{x}\widehat{x}} & . & . \\ . & . & . \\ . & . & . \end{bmatrix} = \begin{bmatrix} \boldsymbol{A}_1^\mathsf{T}\boldsymbol{\Sigma}_{l_1l_1}^{-1}\boldsymbol{A}_1 & \boldsymbol{A}_2^\mathsf{T} & \boldsymbol{H}^\mathsf{T} \\ \boldsymbol{A}_2 & -\boldsymbol{\Sigma}_{l_2l_2} & \boldsymbol{O}^\mathsf{T} \\ \boldsymbol{H} & \boldsymbol{O} & \boldsymbol{O} \end{bmatrix}^{-1} \tag{15}$$

and reveals the theoretical precision of the parameters.

The unknown variance factor can be estimated by

$$\widehat{\sigma}_0^2 = \frac{\Omega}{R} \tag{16}$$

with the redundancy R of the system. Thus the empirical precision becomes

$$\widehat{\boldsymbol{\Sigma}}_{\widehat{x}\widehat{x}} = \widehat{\sigma}_0^2\boldsymbol{\Sigma}_{\widehat{x}\widehat{x}}. \tag{17}$$

2.3 Prior Information

The need for introducing prior information into the estimation process is manifold: Geometric weak configurations of the sensors or the objects may lead to degeneracies resulting in multiple solutions, a parametric family of solution, or no solution at all. To cope with these degeneracies prior information can be introduced in a Bayesian manner by parameter values and their uncertainties in form of covariance matrices. These can be either real observations of the sought parameters or fictitious observations. Beside the eventually necessity to effect determinability the use of prior information leads to stabilization and smoothness.

Applying (9) in a rigorous successive way would mean keep the entire information of the past for the estimations of future states. But a sneaking change of parameters due to changing environmental conditions requires a changing point of linearization, too. To cope with this situation we introduce a factor $\alpha \in (0, 1)$ which reflects the amount of information which should be used to estimate the current state of the parameters (memory length). Furthermore this factor determines the relative weight for the contributions of current data and the prior, i.e. the past information.

The amount of data taken from the past to determine the current state can be estimated in advance. If the number m of observations per image pair is constant the limit of the geometric series $s = \sum_{i=1}^{n} m\alpha^{i-1}$ approaches $s = 1/(1-\alpha)$ for $n \to \infty$ and $\alpha < 1$. Thus for $\alpha = 0.95$ the amount of approximate 20 image pairs will be used to estimated the current state for instance.

2.4 Relation to Kalman filtering

Neglecting the constraints (7) in (9) yields the formulas for the recursive parameter estimation or the Kalman filtering (Kalman, 1960). With the estimation $\widehat{\boldsymbol{\Delta x}}_1 = \boldsymbol{\Sigma}_{\widehat{x}_1\widehat{x}_1}\boldsymbol{A}_1\boldsymbol{\Sigma}_{l_1l_1}^{-1}\boldsymbol{\Delta l}_1$ because of the first observation group $\boldsymbol{l}_1$ and the corresponding covariance matrix $\boldsymbol{\Sigma}_{\widehat{x}_1\widehat{x}_1} = (\boldsymbol{A}_1\boldsymbol{\Sigma}_{l_1l_1}^{-1}\boldsymbol{A}_1)^{-1}$ the Kalman-Filter gain matrix is (Welch and Bishop, 2003)

$$\boldsymbol{F} = \boldsymbol{\Sigma}_{\widehat{x}_1\widehat{x}_1}\boldsymbol{A}_2^\mathsf{T}\left(\boldsymbol{\Sigma}_{l_2l_2} + \boldsymbol{A}_2\boldsymbol{\Sigma}_{\widehat{x}_1\widehat{x}_1}\boldsymbol{A}_2^\mathsf{T}\right)^{-1}. \tag{18}$$

The update of the parameters due to the new observations $\boldsymbol{l}_2$ is:

$$\widehat{\boldsymbol{\Delta x}} = \widehat{\boldsymbol{\Delta x}}_1 + \boldsymbol{F}(\boldsymbol{A}_2\widehat{\boldsymbol{\Delta x}}_1 - \boldsymbol{\Delta l}_2) \tag{19}$$

$$\boldsymbol{\Sigma}_{\widehat{x}\widehat{x}} = (\boldsymbol{I} - \boldsymbol{F}\boldsymbol{A}_2)\boldsymbol{\Sigma}_{\widehat{x}_1\widehat{x}_1} \tag{20}$$

These equations constitute the so-called measurement update (correction) of the extended Kalman filter.

In an intermediate step the time update equations (prediction) are responsible for forward projection of the parameters, e.g. because of a change in the coordinate system accompanied by error propagation.

3 REALIZATION

3.1 Parametrization

For the recursive parameter estimation we set the distances $d_k \doteq 1$ for all pairs of images. For the plane normal vectors $\boldsymbol{n}_k$ we assume all three components to be unknown and introduce the (hard) constraint

$$\boldsymbol{n}_k^\mathsf{T}\boldsymbol{n}_k = 1. \tag{21}$$

The over-parametrization is justified by the fact that eq. (21) is bilinear in the parameters and therefore convenient for linearization. For the rotation $\boldsymbol{R}_{kl}$ three parameters $\boldsymbol{r}_{kl}$ are introduced with $l = k + 1$.

For the single camera we choose the affine model

$$\mathbf{K} = \begin{bmatrix} c & cs & x_0 \\ 0 & cm & y_0 \\ 0 & 0 & 1 \end{bmatrix} \tag{22}$$

with the camera constant c, the scale m, the skew s, and the principal point $[x_0, y_0]$. The intrinsic camera parameters are therefore $\boldsymbol{k} = [c, m, s, x_0, y_0]^\mathsf{T}$. Parameters with known or assumed values can easily be fixed by setting the corresponding variances in $\boldsymbol{\Sigma}_{l_2l_2}$ to zero.

Dropping the indices the parameter vector is $\boldsymbol{x} = [\boldsymbol{k}^\mathsf{T}, \boldsymbol{n}^\mathsf{T}, \boldsymbol{r}^\mathsf{T}, \boldsymbol{t}^\mathsf{T}]^\mathsf{T}$, where $\boldsymbol{k}$ are global parameters, $\boldsymbol{n}$ are global parameters which have to be propagated, and $\boldsymbol{r}$ and $\boldsymbol{t}$ are local parameters for each consecutive image pair.

3.2 Observations, Weights and Model Validation

The correspondencies of image points can be established essentially in two ways: The first method first extracts interest points in both images and then finds the correspondencies, e.g. by correlation followed by RANSAC (Fischler and Bolles, 1981). The second methods extracts interest points in one image and searches them in the second image, e.g. by a tracker (Lucas and Kanade, 1981).

Empirical covariance matrices of the sub-pixel positions can be derived anyway – either by an residual-based or an derivative-based approach (Kanazawa and Kanatani, 2001). The inverse covariance matrices are then the weight matrices for the adjustment procedure. The first observation group $\boldsymbol{l}_1$ consists of the coordinates of the set of observed points. By considering errors in one image for each correspondence i and mapping $\boldsymbol{H}$

$$\overset{k}{x}_i + \overset{k}{v}_{x_i} = \frac{H_{11}\overset{l}{x}_i + H_{12}\overset{l}{y}_i + H_{13}}{H_{31}\overset{l}{x}_i + H_{32}\overset{l}{y}_i + H_{33}} \tag{23}$$

$$\overset{k}{y}_i + \overset{k}{v}_{y_i} = \frac{H_{21}\overset{l}{x}_i + H_{22}\overset{l}{y}_i + H_{23}}{H_{31}\overset{l}{x}_i + H_{32}\overset{l}{y}_i + H_{33}} \tag{24}$$

holds with the reprojection errors v_i.

The application of the RANSAC procedure enforces the validity of the mapping model (4). Thus model violations by non-planar

scenes can be treated as long as a dominant plane is visible. Furthermore, with the realistic weights for the observations the estimation of the variance factor (16) can be tested statistically since its expectation value is one.

3.3 Approximation Values

In general the parameter estimation model requires approximation values $\boldsymbol{x}_0$ for the sought parameters. For applications processing a dense video stream it is usually sufficient to use the estimation results from the previous image pair as approximation values for the next image pair. Furthermore for the relative rotation $\boldsymbol{R}_0 = \boldsymbol{I}_3$ holds. For applications with a long base line approximation values can be obtained from the decomposition (3) with an approximately given calibration matrix.

3.4 Application I: Calibration of an Airborne Camera

For a high-flying airborne camera the observed scenes usually appear flat. The heights above ground and the relative orientation between two consecutive frames is individual. Thus we have 3 rotation parameters and 3 translation components for each frame pair and quasi global parameters $\boldsymbol{k}$ and $\boldsymbol{n}$. Prior information is introduced by the estimated parameters and their uncertainties from the respective previous image pair. The normal vector has to be transformed into the next camera coordinate system. Therefore the prior information reads $\boldsymbol{l}_2 = (\boldsymbol{k}_l^\mathsf{T}, \boldsymbol{n}_l^\mathsf{T})^\mathsf{T}$ with

$$\begin{aligned} \boldsymbol{k}_k &= \boldsymbol{k}_l & (25)\\ \boldsymbol{n}_l &= \boldsymbol{R}_{lk}\boldsymbol{n}_k, & (26) \end{aligned}$$

$\boldsymbol{R}_{lk} = \boldsymbol{R}(\boldsymbol{r}_{lk})$, and the covariance matrix $\boldsymbol{\Sigma}_{l_2l_2}$ estimated with (14) or (15) from the previous stage. The Jacobian of (25) and (26) is

$$\boldsymbol{A}_2 = \begin{bmatrix} \boldsymbol{I}_5 & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} \\ \boldsymbol{O} & \boldsymbol{R}_{lk}^\mathsf{T} & (\partial \boldsymbol{R}_{lk}^\mathsf{T}/\partial \boldsymbol{r}_{lk})\boldsymbol{n}_k & \boldsymbol{O} \end{bmatrix}. \quad (27)$$

Note that $\boldsymbol{\Sigma}_{l_2l_2}$ has a rank deficiency of 1 due to the constraint $\boldsymbol{n}_k^\mathsf{T}\boldsymbol{n}_k = 1$ on the parameters. Its null space $\mathcal{N}(\boldsymbol{\Sigma}_{l_2l_2}) = (\boldsymbol{0}^\mathsf{T}, \boldsymbol{n}_k^\mathsf{T})^\mathsf{T}$ can be use to compute the pseudo inverse $\boldsymbol{\Sigma}_{l_2l_2}^+$ if needed.

For a metric reconstruction of the sensor trajectory the scale has to be introduced and updated by $d_k = d_j + \boldsymbol{n}_j^\mathsf{T}\boldsymbol{t}_{jk}$ (Han and Kanade, 1998).

3.5 Application II: Camera Calibration for Indoor Robots

For an indoor robot moving on a ground plane the camera height above ground $d_l = d_k = d$ is constant. Furthermore the rotation is restricted to rotations by an angle ϕ around the plane normal $\boldsymbol{n}$ being the only rotation parameter $\boldsymbol{r} = (\phi)$. Thus an appropriate parametrization of the rotation matrix is the angle-axis-representation

$$\boldsymbol{R}(\boldsymbol{n}, \phi) = \cos(\phi)\boldsymbol{I}_3 + \sin(\phi)\boldsymbol{S}_n + (1 - \cos(\phi))\boldsymbol{D}_n \quad (28)$$

with $\boldsymbol{S}_n$ inducing cross product $\boldsymbol{S}_n\boldsymbol{m} = \boldsymbol{n} \times \boldsymbol{m}$ and $\boldsymbol{D}_n$ denoting the dyadic product $\boldsymbol{D}_n = \boldsymbol{n}\boldsymbol{n}^\mathsf{T}$. The plane normal is perpendicular to the translation vector and therefore

$$\boldsymbol{n}^\mathsf{T}\boldsymbol{t} = 0 \quad (29)$$

holds and can be used as an additional constraint.

Since the normal vector remains unaffected by the rotation, the prior information is simply $\boldsymbol{k}_l = \boldsymbol{k}_k$ and $\boldsymbol{n}_l = \boldsymbol{n}_k$ with the Jacobian

$$\boldsymbol{A}_2 = \begin{bmatrix} \boldsymbol{I}_5 & \boldsymbol{O} & \boldsymbol{0} & \boldsymbol{O} \\ \boldsymbol{O} & \boldsymbol{I}_3 & \boldsymbol{0} & \boldsymbol{O} \end{bmatrix} \quad (30)$$

w.r.t. to all parameters $\boldsymbol{x} = [\boldsymbol{k}^\mathsf{T}, \boldsymbol{n}^\mathsf{T}, \phi, \boldsymbol{t}^\mathsf{T}]^\mathsf{T}$.

3.6 Algorithm

The outline of the proposed algorithm is as follows: For each consecutive frame pair do

1. *Feature extraction*. Determine interest points in the first image, e.g. by the Förstner operator (Förstner and Gülch, 1987).
2. *Tracking*. Find the corresponding points in the subsequent image, e.g. with the KLT-tracker (Lucas and Kanade, 1981) together with estimated covariance matrices for the estimated shifts.
3. *Outlier elimination.* Apply the RANSAC procedure to determine an inlier set (Fischler and Bolles, 1981) in conjunction with minimizing algebraic distances.
4. *Recursive parameter estimation.* Calculate the parameter updates and the corresponding covariance matrices according to (9) and (14).

For applications with a wide stereo base line the interest points have to be determined independently with sub-pixel accuracy. Furthermore, in the presence of a moderate number of outliers the RANSAC procedure can be replaced by an adjustment procedure with a robust cost function.

4 EXPERIMENTAL TESTS

For the evaluation of the approach we used synthetic and real data sets. With the help of the synthetic data we show that the algorithm converges to the correct solution and produces feasible results. The applicability of the approach is shown with a real data set from a camera mounted on a dolly driving through a corridor.

For the initialization of the procedure we introduced a rough guess of the parameters and their uncertainty whereas the a rank deficiency of the covariance matrix of the normal vector has been enforced by spherical normalization accompanied by error propagation (Heuel, 2004).

4.1 Synthetic Data

Test Setup. For the evaluation of the approach we simulated the data of an indoor robot with a camera moving above a virtual chess board (see figure 2). The camera moved on a circle of constant height above the plane, the angle increment for the 200 positions were 1.8 deg.

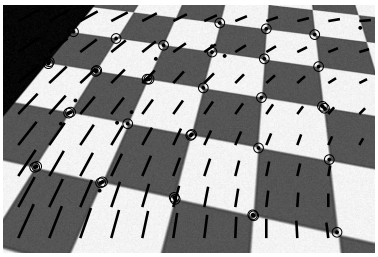


Figure 2: shows an image of the synthetic image sequence with the extracted interest points.

The image of the chess board was mapped into images of size 512 pxl × 768 pxl with the help of the calibration matrix

$$\mathbf{K} = \begin{bmatrix} 512 & 0 & 384 \\ 0 & 512 & 256 \\ 0 & 0 & 1 \end{bmatrix}. \quad (31)$$

White noise $\sigma = 2$ gr has been added to the gray values. The orientation angles of the camera w.r.t. the driving direction were

$$\begin{array}{lrl} \text{roll:} & 0.0 & \text{deg} \\ \text{nick:} & 55.0 & \text{deg} \\ \text{gear:} & 1.8 & \text{deg,} \end{array} \tag{32}$$

where the gear angle denotes the relative change in the azimuth between two consecutive camera orientations. With this test setup we have constant unknown parameters.

The images have been processed in the way described in section 3.2. Thus the results include possibly systematic errors of the tracking algorithm or the correspondences search. For the prior for the first image pair we used true parameter values (31), (32) and $\sigma_c = 20$ pxl, $\sigma_{x_0} = \sigma_{y_0} = 5$ pxl, $\sigma_{n_x} = \sigma_{n_y} = \sigma_{n_z} = 0.1$ as a rough guess of the uncertainties of the parameters $\boldsymbol{k} = (c, x_0, y_0)^\mathsf{T}$ and $\boldsymbol{n}$.

Figure 3 shows the distribution of the estimated variance factor (16). Its mean is approximately 1. Thus we conclude that the weights for the observations used within the adjustment procedure are plausible and reasonable since the model of a planar scene really holds and outliers are rejected by RANSAC.

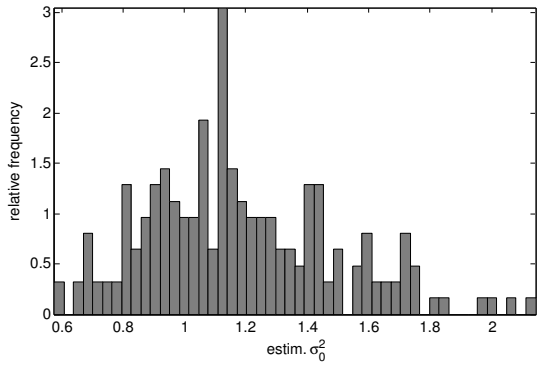


Figure 3: shows the distribution of the estimated variance factor (16) with expectation value 1.

For the evaluation the χ^2-distributed Mahalanobis distances

$$d_M = (\widehat{\boldsymbol{x}} - \widetilde{\boldsymbol{x}})^\mathsf{T} \widehat{\boldsymbol{\Sigma}}^+_{\widehat{x}\widehat{x}} (\widehat{\boldsymbol{x}} - \widetilde{\boldsymbol{x}}) \tag{33}$$

of the estimated parameters $\widehat{\boldsymbol{x}}$ and their true values $\widetilde{\boldsymbol{x}}$ can be computed for each image pair using the pseudo inverse. Figure 4 shows the empirical distribution of these distances for all parameters $\boldsymbol{x} = [\boldsymbol{k}^\mathsf{T}, \boldsymbol{n}^\mathsf{T}, \phi, \boldsymbol{t}^\mathsf{T}]^\mathsf{T}$ and confirms the expected shape of a χ^2-distribution. For a rigorous common adjustment of all image pairs the expect average degree of freedom is 3 because the intrinsic parameters and the normal vector would be global parameters. The result illustrated in figure 4 is slightly worse because of the underlying mixed distribution.

4.2 Real Data

To demonstrate the feasibility of the approach we acquired a real data set with a video camera mounted on a dolly. The projection center was approximate 1.55 meters above the ground floor. The nick angle was approximate 33 deg, roll angle approximate 0 deg and the azimuth w.r.t. the driving direction approximately 45 deg. The image resolution is 576×720 pixel. The trajectory is curved with radii up to 2 meters. Figure 5 shows exemplary an image of the observed floor. Points on the visible part of the wall and on the fire extinguisher have been tracked, too. But since they do not lie in the dominant plane they have been discarded by the RANSAC.

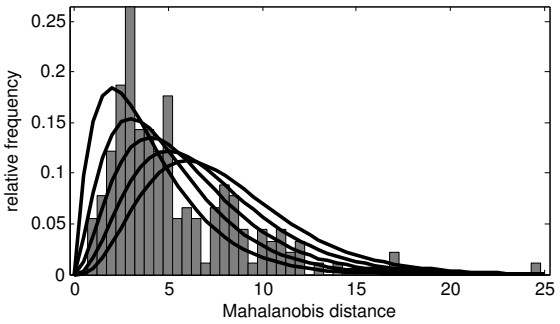


Figure 4: shows the empirical distribution of the Mahalanobis distances (33) for all parameters with the expected shape of the χ^2_4-, χ^2_5-, χ^2_6-, χ^2_7-, and χ^2_8-distribution.

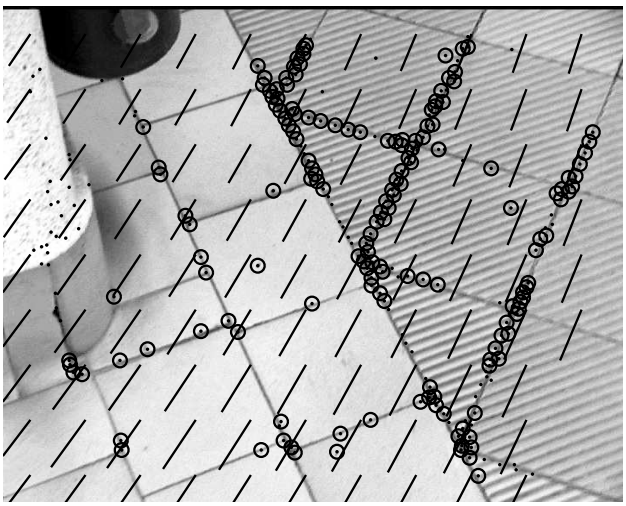


Figure 5: shows the tiled floor, the tracked interest points ($\cdot$), the points in the dominant plane determined by RANSAC ($\circ$) and the resulting motion field.

We chose a memory factor of $\lambda = 0.99$ to bridge sequence parts with critical configurations e.g. straight forward motions. Figure 6 shows the evolution of the estimated camera constant. The parameter has been initialized 50 pixels larger than the value determined separately by Zhang's calibration method (Zhang, 1999). The runs for parameters of the relative orientation are plotted in figure 7. The concatenation of these relative motions delivers the sensor trajectory.

The computations require approximately 3 seconds per image pair with a non optimized MATLAB Code including graphical output. Thus the real time capability is within easy reach.

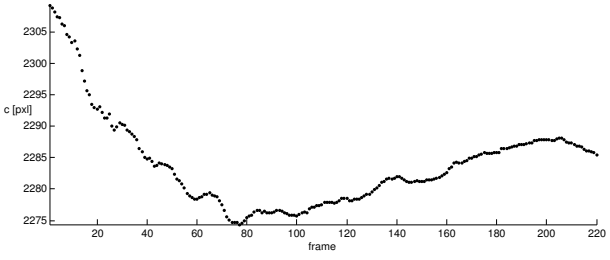


Figure 6: shows the estimated the camera constant c for the first 220 frame pairs.

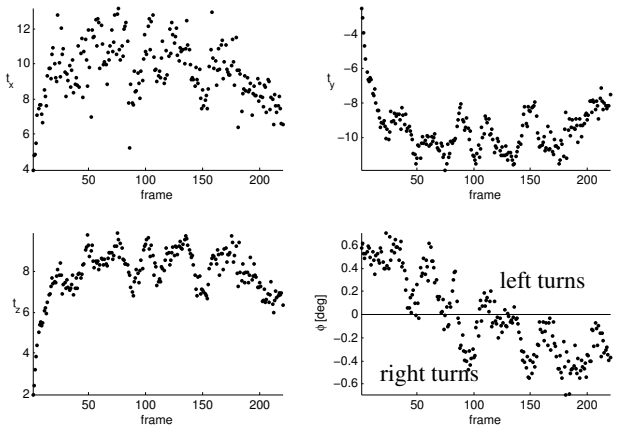


Figure 7: shows the estimated parameters t_x, t_y, t_z and ϕ of the relative orientations for the first 200 frame pairs.

5 CONCLUSIONS AND OUTLOOK

We presented and studied a continuous self-calibration method for mobile cameras observing a plane. The procedure is suitable for real time applications such as autonomously navigating indoor robots capturing endless video streams. By the application of the RANSAC the approach is able to cope with model violations as long as the scene plane dominants. Because of the geometric parametrization of the frame to frame mappings information from other sensors can easily be incorporated. Experiments with synthetic and real data sets confirm the feasibility of the approach.

Conclusions. For the determinability of the parameters a noticeable relative rotation and translation between consecutive frames is required. Image frames with almost identical orientations should be discarded e.g. by enforcing a disparity limit of say $>$10 pixel for at least one image point (Nistér, 2001). Critical motion sequences such as straight forward motions can be bridged by introducing an appropriate memory length.

Outlook. Reference data is needed to perform a more meaningful evaluation of the results for real data sets. For the intrinsic parameters this ground truth information could stem from a lab camera calibration with superior accuracy. The influence of the prior information and past information respectively can be controlled by an adaptive weighting of the information sources. Furthermore the estimation process can be stabilized by introducing further (soft) constraints which enforce a smooth trajectory of the sensor.

REFERENCES

Faugeras, O. and Lustman, F., 1988. Motion and Structure from Motion in a piecewise planar Environment. International Journal of Pattern Recognition in Artificial Intelligence 2, pp. 485–508.

Fischler, M. A. and Bolles, R. C., 1981. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. Communications of the Association for Computing Machinery 24(6), pp. 381–395.

Förstner, W. and Gülch, E., 1987. A Fast Operator for Detection and Precise Location of Distinct Points, Corners and Centres of Circular Features. In: ISPRS Intercommission Workshop, Interlaken.

Han, M. and Kanade, T., 1998. Homography-Based 3D Scene Analysis of Video Sequences. In: Proceedings of the DARPA Image Understanding Workshop.

Heuel, S., 2004. Uncertain Projective Geometry. Statistical Reasoning in Polyhedral Object Reconstruction. Lecture Notes in Computer Science, Vol. 3008, Springer.

Kalman, Rudolph, E., 1960. A New Approach to Linear Filtering and Prediction Problems. Transactions of the ASME–Journal of Basic Engineering 82(Series D), pp. 35–45.

Kanazawa, Y. and Kanatani, K., 2001. Do we really have to consider covariance matrices for image features? In: International Conference on Computer Vision, pp. 301–306.

Lucas, B. T. and Kanade, T., 1981. An Iterative Image Registration Technique with an Application to Stereo Vision. In: Proc. of Image Understanding Workshop, pp. 212–130.

Malis, E. and Cipolla, R., 2002. Camera self-calibration from unknown planar structures enforcing the multi-view constraints between collineations. IEEE Transactions on Pattern Analysis and Machine Intelligence 4(9), pp. 1268–1272.

Maybank, S. and Faugeras, O., 1992. A theory of self-calibration of a moving camera. International Journal of Computer Vision 8(2), pp. 123–151.

McGlone, J. C., Mikhail, E. M. and Bethel, J. (eds), 2004. Manual of Photogrammetry. 5th edn, American Society of Photogrammetry and Remote Sensing.

Nistér, D., 2001. Frame decimation for structure and motion. In: Workshop on 3D Structure from Multiple Images of Large-Scale Environments, Lecture Notes on Computer Vision, Vol. 2018, pp. 17–34.

Nistér, D., 2004. An Efficient Solution to the Five-Point relative Pose Problem. IEEE Transactions on Pattern Recognition and Machine Intelligence 26(6), pp. 756–769.

Pollefeys, M., Koch, R. and Gool, L. V., 1999. Self-calibration and metric reconstruction inspite of varying and unknown intrinsic camera parameters. International Journal of Computer Vision 32(1), pp. 7–25.

Triggs, B., 1998. Autocalibration from planar scenes. In: Proceedings of the 5th European Conference on Computer Vision, Freiburg, Germany.

Welch, G. and Bishop, G., 2003. An Introduction to the Kalman Filter. Technical Report TR 95-041, Department of Computer Science, Univ. of North Carolina at Chapel Hill.

Zelnik-Manor, L. and Irani, M., 2002. Multiview Constraints on Homographies. IEEE Transactions on Pattern Recognition and Machine Intelligence 24(2), pp. 214–223.

Zhang, Z., 1999. Flexible Camera Calibration by Viewing a Plane from Unknown Orientations. In: International Conference on Computer Vision (ICCV'99), Corfu, Greece, pp. 666–673.

SCALE BEHAVIOUR PREDICTION OF IMAGE ANALYSIS MODELS FOR 2D LANDSCAPE OBJECTS

Janet Heuwold, Kian Pakzad, Christian Heipke

Institute of Photogrammetry and GeoInformation, Leibniz Universität Hannover,
Nienburger Str. 1, D-30167 Hannover, Germany – (heuwold, pakzad, heipke)@ipi.uni-hannover.de

KEY WORDS: Multiresolution, Scale Space, Modelling, Image Analysis, Interpretation, Urban, Method

ABSTRACT:

This paper presents a new methodology for the automatic adaptation of image analysis object models for the extraction of 2D landscape objects to a lower image resolution. The knowledge of the object models is represented in form of semantic nets. The developed adaptation method includes a prediction of the object's behaviour in linear scale-space using analysis-by-synthesis. The scale behaviour prediction also takes into account scale events possibly occurring during scale change. The presented algorithm extends previous work concerning the adaptation of parallel linear object parts to object models consisting of 2D area object parts with arbitrary orientation. An example for the adaptation of an object model describing a 4-arm road junction with symbol markings demonstrates the application of the methodology. Finally, conclusions point out enhancements of the method.

1. INTRODUCTION

The appearance of landscape objects varies in aerial or satellite images of different resolution. Hence, for knowledge-based object extraction in images of different resolution, different models describing the objects are usually required. The objective of this paper is to introduce a new methodology for the automatic adaptation of image analysis models for object extraction to a lower image resolution. The objects that are modelled consist of 2D arbitrarily oriented object parts. The models for object extraction are represented as semantic nets. Since the image structure often severely changes between different image resolutions, the appearance of objects in lower resolution is to be altered by the adaptation method. The central problem is the prediction of the scale behaviour of object parts. The second core issue is the automatic handling of the given and adapted object model.

The method presented in this paper is an extension of work on the adaptation of object models consisting of linear parallel object parts [Pakzad & Heller, 2004; Heller & Pakzad, 2005]. The prediction of scale behaviour simplifies for linear-type parallel objects to a 1D problem, since an analysis of the cross-section is in principal sufficient. In the 2D case, which is tackled here, the scale behaviour of the objects is more complex. Particularly challenging is the prediction of 2D scale events that may occur during scale change.

We use linear scale-space theory for the prediction of the object's scale behaviour. The linear or Gaussian scale-space as defined first by [Witkin, 1983] and [Koenderink, 1984] is created by convolution of an image $L(x,y)$ with the Gaussian $g(x;t)$ of varying width. Thereby, a family of signals $L : R^2 \times R_+ \rightarrow R$ is derived depending only on a single scale parameter t corresponding to the square of the Gaussian standard deviation with $t=\sigma^2$. For more details concerning the characteristics of linear scale-space, see e.g. [Florack et al., 1994]. The analysis of image structure in different scales is also referred to as *deep structure* [Koenderink, 1984]. Two main approaches exist for the analysis of image structure in scale-space including the linking of 2D events between scales: the *scale-space primal sketch* proposed by Lindeberg [Lindeberg, 1993; Lindeberg, 1994] and the *scale-space hierarchy* suggested by Kuijper [Kuijper, 2002; Kuijper et al., 2003]. While the scale-space primal sketch is based on linking blob primitives between adjacent scales, the scale-space hierarchy is constructed by linking critical points between scales. Since blobs are not only more stable and easier to track between scales, and linking of critical points between scales can also be ambiguous (and hence requires the determination of non-generic catastrophes) [Lindeberg, 1994], we developed a methodology for the automatic scale behaviour prediction of 2D object models based on blob linking.

In literature some other approaches dealing with scale behaviour analysis of landscape objects from remote sensing data can be found, e.g.: scale events for buildings were analysed in morphological scale-space by [Forberg & Mayer, 2002]; [Mayer & Steger, 1998] give an analytical analysis of the behaviour of a cross-section of a road with a vehicle in linear scale-space; the scale-space primal sketch was used by [Hay et al., 2002] for the scale behaviour description of whole landscapes as complex systems. However, the prediction of the scale behaviour of complete 2D image analysis object models and their adaptation to a lower image resolution is new.

The next section first summarizes the previously developed strategy for the adaptation of 1D object models and then gives an overview of the scale-space primal sketch. The third section deals with the requirements regarding the composition of automatically adaptable semantic nets. The adaptation method is outlined in section 4, while an example for the adaptation of a model for a junction area with road symbol markings is given in section 5. At last, conclusions finish this paper.

2. STATE OF THE ART

2.1 Scale-space primal sketch

The *scale-space primal sketch* was introduced by Lindeberg as an explicit representation of features in scale-space and their relations at different levels of scale [Lindeberg, 1993; Lindeberg, 1994]. The sketch provides a qualitative description of image structure and was designed as a basis for the extraction of significant image features at stable scales for later processing towards object extraction.

Blobs serve as primitives of the scale-space primal sketch. *Grey-level blobs* are smooth image regions that are brighter or

darker than the background and thereby stand out from its surrounding. Figure 1 illustrates the concept of grey-level blobs. By definition a grey-level blob *B(E)* is a region of a scale-space image *L(x,y;t)* associated with a pair of critical points (or regions in discrete scale-space) consisting of one local extremum *E* and one delimiting saddle *S*. The grey-level blob is a 3D object with extent both – in space and grey-level (indicated by *z*). The spatial extent of the blob is given by its *support region Supp(B)*. The grey-value of the delimiting saddle *S* equals the base level $z_{base}(B)$ of the blob. The support region of the blob is delimited by those points having grey-values exceeding or equal to the base level. On the other side, the volume of the grey-level blob *Vol(B)* is defined by the integral of the image function over its support region. Finally, the blob contrast *C(B)* is given by the grey-value difference of the base level $z_{base}(B)$ and the local extremum *E*. A sequential blob detection algorithm for the delineation of grey-level blobs in 2D discrete images using grey-value sorting initiated from local maxima serving as blob seeds is outlined in [Lindeberg, 1994].

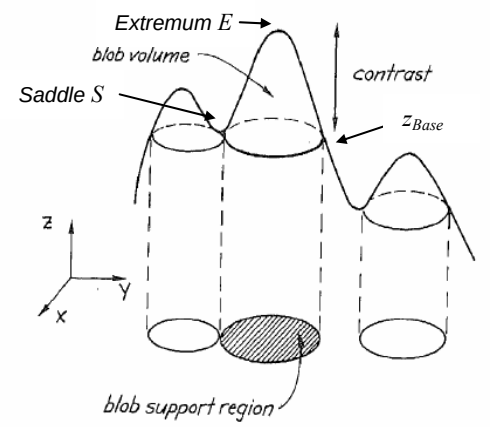


Figure 1. Definition of grey-level blobs (adapted from: [Lindeberg, 1993])

Generally, for a grey-level blob existing at a certain level of scale a corresponding blob at a slightly finer or coarser scale can be found. By linking these grey-level blobs between different scales so-called *scale-space blobs* are established. These four-dimensional objects have an extent in space, grey-level and scale. However, a plain linking of blobs between adjacent scale levels is not always possible, because scale events of blobs (blob events) induce topological changes in image structure between discrete scale levels. In terms of catastrophe theory, where scale acts as perturbation parameter, one of the two critical points of a blob – the local extremum *E* and the delimiting saddle *S* – is involved in a bifurcation [Lindeberg, 1992].
There are four types of blob events with increasing scale for 2D image structures to be distinguished:

- *Annihilation*: a blob disappears
- *Merging*: two blobs or more merge into a single one
- *Split*: one blob splits into two or more
- *Creation*: a new blob appears

Due to the singularities introduced by blob events in scale direction, the extent of a scale-space blob is delimited by the respective scale levels where the blob events occur. A scale-space blob is therefore associated with a scale-space lifetime, given by the difference of its appearance scale t_A and the disappearance scale t_D. In order to avoid ambiguous matching in the linking process of grey-level blobs between successive scales Lindeberg [1994] proposes an algorithm of adaptive scale sampling, which refines the scale sampling until all blob relations between adjacent scale levels can be unambiguously traced back to the four blob events. In this way, a complete scale-space representation of the image structure over all scales is derived. This information is desirable for the detection of significant image features in the most suitable scales, which complies with the aim the primal sketch was originally developed for. However, for the goal of our work – the automatic adaptation of object models to a given coarser scale – a complete description of behaviour over all scales is not required. Thereby, we introduce in section 4.2 a modified algorithm for the prediction of object appearance in a coarser target scale. This new method is inspired by the primal sketch and accomplishes blob linking between different scales.

2.2 Strategy and methodology of 1D adaptation

The scale-dependent adaptation of image analysis models describing objects composed of linear parallel object parts follows a process in three steps – decomposition, scale change analysis, and fusion. For details of the strategy see [Pakzad & Heller, 2004]. The same strategy (depicted in Figure 2) is applied in the adaptation procedure for 2D object models presented in this paper.

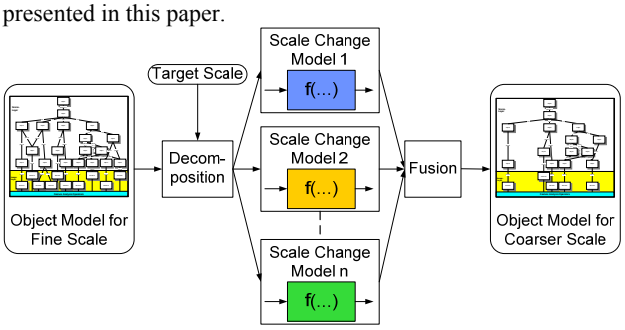


Figure 2. Strategy for adaptation process

The first stage of the automatic adaptation process decomposes the given object model for the fine scale into separate object parts that can be analysed separately regarding their scale behaviour. The decomposition takes into account the possibility of mutual influence of the appearance of nearby objects when scale changes – denoted as *interaction.* In this case, the respective object parts are analysed simultaneously in groups in the following scale change analysis phase.
Scale change models predict the appearance in the target scale for each interacting group or for single non-interacting object parts resulting from the decomposition. The prediction is carried out using analysis-by-synthesis, simulating the appearance of the object in the target scale by generated synthetic images.
At last, in the fusion stage all predicted object parts are recomposed back into a complete object model that is suitable for the extraction of the object in the target resolution. All hierarchical and spatial relations are maintained under consideration of the predicted scale events, which affect the resulting number of remaining object parts.

In contrast to the one-dimensional case, where only two scale events are to be considered, four different scale-space events may occur for 2D image structure, including the creation of new blobs with increasing scale (see also section 2.1). The prediction of scale events of participating object parts is thus not as straight forward and requires a more sophisticated approach to scale behaviour prediction than the previously presented method for linear parallel object parts.

3. COMPOSITION OF ADAPTABLE MODELS

The knowledge representation of the models for object extraction to be adapted automatically is realized in form of *semantic nets*. The concept of semantic nets allows the description of the individual parts of landscape objects in *nodes* and their mutual relations by *edges*. The object parts can be described in different layers: the real world, material and the image. The semantic nets we use for object description in this study describe the landscape object both in the real world and its appearance in the image. In the following, the reader is assumed to be familiar with semantic nets. For a review on semantic nets see e.g. [Niemann et al., 1990; Tönjes et al., 1999].

The concept of semantic nets enables many possibilities for the composition of an object model for the extraction of a particular object, i.e. the knowledge representation for the description of an object can be realized in very different semantic nets. Obviously, not all variations of semantic nets describing the same object are suitable to be treated in an automatic way. An automatic scale-dependent adaptation requires a few constraints concerning the composition of the given object model for high resolution. Details on necessary constraints concerning the generation of suitable semantic nets for the adaptation of linear parallel object parts are derived in [Pakzad & Heller, 2004]. Most of the previously formulated constraints for the adaptation of the 1D case in principal also apply to 2D objects. In particular, the requirements for the automatic decomposition of the given high-resolution semantic net and the assignment of a suitable feature extraction operator to each node must also hold for area-type arbitrarily oriented object parts.

Nevertheless, the additional dimension has to be considered in the given object models. The attributes in the nodes and edges have to deliver the spatial information concerning the appearance of the object in the image that is needed for the scale behaviour prediction in the analysis-by-synthesis process.
We define therefore the following attributes for nodes representing area-type object parts:

- Object Type: Objects with the same object type can be extracted by the same type of feature extraction operator. There are two categories for area primitives:
  - geometric object types: *rectangle, triangle, ellipse*
  - *arbitrary patterns* for more complex area-type object parts: its shape is defined by templates.
- Extent: The extent specifies the size of the object part and is stated in pixel numbers for the respective image resolution. For patterns the extent of the bounding box is relevant: its *length* and *width.* Geometric types are specified by their individual distinctive parameters.
- Orientation: This attribute is given in relation to the main axis of the object.
- Grey Value: This value describes the radiometric properties of the primitive.

The attributes of the spatial relations in the models are to be extended in comparison to line-type objects for specifications concerning:

- Distance: The value of the distance refers to a fixed point of the object acting as the origin of a local coordinate system. 2D objects require distance values concerning two perpendicular directions.

4. ADAPTATION METHOD

In this section the developed methodology for the three stages of the automatic adaptation process is explained in detail.

4.1 Decomposition

In order to facilitate the scale behaviour prediction of the object in the scale change models, a separate analysis of the individual parts of the modelled object is desirable. However, while scale changes adjacent objects may influence each other's appearance when they lie close to each other in the target scale. In this case, the participating objects need to be analysed together and form an *interaction group* in the successive scale change models. All other object parts that are not subject to interaction can be treated as single object parts in the scale behaviour analysis.

Whether interaction exists between object parts in the target scale depends on the distance of the respective objects and on the size of the Gaussian kernel associated with the target scale σ_t. As basis for decomposing the object parts in interacting groups or non-interacting single object parts, an *interaction zone* is established in form of a buffer surrounding each object part. If the individual interaction zones of (at least) two object parts intersect the respective object parts are assigned into an interaction group. Otherwise, the object part can be analysed separately as a single non-interacting object. The size of the buffer is determined by the size of the discrete Gaussian given by the target scale σ_t. Figure 3 sketches the concept of interaction zones.

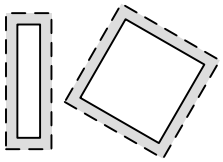


Figure 3. Interaction zones (grey, dashed border) around two nearby object parts (white, continuous border)

4.2 Scale change analysis

Scale change models predict the scale behaviour for each separate non-interacting object part and interaction group. The prediction is carried out in an analysis-by-synthesis procedure, analysing the objects in synthetic images in original scale, target scale, and target resolution. The analysis takes into account possible scale events of the object parts that may occur during scale change. The result is a description of the appearance of the object parts in the given target resolution in terms of attributes for the nodes and edges. The workflow of the analysis-by-synthesis process applied in the scale change analysis stage is depicted in Figure 4.

First, from the specifications of the object parts' appearance in the nodes of the given semantic net for the high resolution a synthetic image is created for each object part or interaction group to be analysed. This initial image L_0 simulates the appearance of the object parts in the original scale σ_0. In a second step, the initial image is transferred to the target scale σ_t by convolution with the respective discrete Gaussian into the target scale image L_t. The analysis regarding possibly occurred scale events is accomplished in this image. Since the prediction of the attributes describing the appearance of the object parts in the low resolution requires for the simulation a synthetic image in the target resolution, the target scale image L_t is subsequently down-sampled to the corresponding lower resolution R_t.

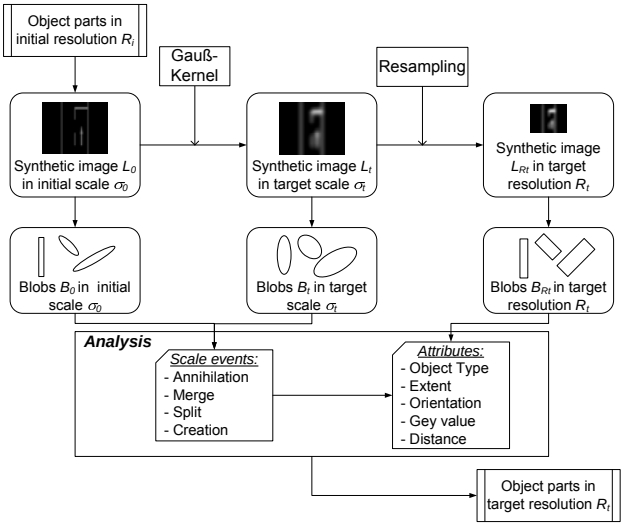


Figure 4. Analysis-by-synthesis process

4.2.1 Scale event prediction

The prediction of scale events considers the four possible blob events Annihilation, Merging, Split and Creation, as listed in section 2.2. However, the construction of a complete scale-space primal sketch over all scales is not necessary. Instead, for our goal – the adaptation of object models to a given lower resolution – the prediction of the number of remaining blobs in the target scale is sufficient. In accordance with the sampling theorem of signal theory, the total number of blobs in the target scale image corresponds to the remaining number of object parts in the given target resolution. Thus, only blob linking between the original and the target scale is necessary.

At first, blob detection is carried out in both the initial image L_0 and the target scale image L_t. The algorithm proposed for sequential blob detection in [Lindeberg, 1994] is used for this purpose. The results are the position of the support regions *Supp* and the extrema *E* and saddles *S* of the blobs in both scales.

Blob linking between the initial image and the target scale image is then carried out by matching blobs with intersecting support regions in original and target scale. We use the support region of the initial scale as search space in the target scale and assume the target scale extremum not to drift outside its corresponding blob support region in initial scale. [Lindeberg, 1992] also states the observation that the blob support region acts as a coarse bound on the drift of local extrema. Although the drift velocity of local extrema may tend to infinity near a bifurcation (i.e. scale event), "the grey-level blob support region defines a natural spatial region to search for blobs in the next level of scale".

We assume that most blobs are not subject to a scale event during the scale change given by the specified target scale. In the blob linking process we thus first try to establish a so-called *plain link* between the initial and target scale for all blobs. Based on the number of blobs in initial scale $\#B_0=m$ of the set of support regions in initial scale $Supp_0$ and the number of blobs in target scale $\#B_t=n$ of the set of support regions in target scale $Supp_t$ a plain link must fulfil the following condition we define in this study:

Plain Link: One particular blob in initial scale has one and only one direct correspondence in target scale. The support region of a blob B_i in initial scale $Supp_0(B_i)$ intersects the support region of a blob B_q in target scale $Supp_t(B_q)$. All other blob support regions in target scale must not intersect $Supp_0(B_i)$.

$$\exists^{=1}\, q\ (Supp_0(B_i) \cap Supp_t(B_q) \neq 0) \quad , i\in\{1\ldots m\},\ q\in\{1\ldots n\} \tag{1}$$

If a plain link cannot be established for all blobs, blob events must have occurred. The types of the respective scale events are then resolved for the remaining blobs. We therefore set up in this paper the following postulates for the occurrence of blob events during scale change:

Annihilation: One particular blob in initial scale has no correspondence in target scale. The set of support regions in target scale $Supp_t$ is empty at the position of a blob support region in initial scale $Supp_0(B_i)$.

$$\exists\ i\ (Supp_0(B_i) \cap Supp_t = 0) \quad , i\in\{1\ldots m\} \tag{2}$$

Merging: Two (or more) initial blobs have one and the same blob as correspondence in target scale. The support regions of at least two initial blobs $Supp_0(B_i)$ and $Supp_0(B_j)$ intersect the support region of one and the same blob in target scale $Supp_t(B_q)$.

$$\exists\ q\ ((Supp_0(B_i) \cap Supp_t(B_q) \neq 0) \wedge (Supp_0(B_j) \cap Supp_t(B_q) \neq 0)) \quad , i,j\in\{1\ldots m\},\ q\in\{1\ldots n\},\ i\neq j \tag{3}$$

Split: One initial blob has two (or more) blobs as correspondence in target scale. The support region of an initial blob $Supp_0(B_i)$ intersects the support regions $Supp_t(B_q)$ and $Supp_t(B_s)$ of at least two blobs in target scale.

$$\exists\ q\ \exists\ s\ ((Supp_0(B_i) \cap Supp_t(B_q) \neq 0) \wedge (Supp_0(B_i) \cap Supp_t(B_s) \neq 0)) \quad , i\in\{1\ldots m\},\ q,s\in\{1\ldots n\},\ q\neq s \tag{4}$$

Creation: One particular blob in target scale has no correspondence in initial scale. The set of support regions in initial scale $Supp_0$ is empty at the position of a blob support region in target scale $Supp_t(B_q)$.

$$\exists\ q\ (Supp_0 \cap Supp_t(B_q) = 0) \quad , q\in\{1\ldots n\} \tag{5}$$

4.2.2 Attribute prediction

The attributes in the nodes specify the appearance of an object part in the image. The values of the attributes in the nodes of the adapted semantic net for the lower resolution are therefore analysed in the synthetic target resolution image L_{Rt} simulating the object. Blobs correspond to the individual object parts. Hence, the number of blobs in the target resolution equals the number of nodes in the semantic net for the lower resolution. For each blob in the target resolution the following attributes are derived:

- Object Type: Due to scale events and deformations during scale change, the shape of primitives of exact geometric type may change significantly in the target resolution for a larger scale change. Instead, the shape of the object parts has to be described by an arbitrary pattern. Interacting patterns always lead to new patterns, which deliver new templates for the image analysis operators. These templates can be directly obtained from the analysis-by-synthesis process.

- Extent: The support region of a blob serves as estimation of the respective object part's extent and position. The blob support region in target resolution $Supp_{Rt}(B)$ is delimited by its saddle point region S_{Rt}.
- Grey Value: The maximum grey value is determined for each blob in its previously determined extent (which is given by its support region).
- Orientation: As orientation is invariant with respect to scale change, the orientation of a blob only needs to be specified, if object parts are merged in target resolution. For this task, the main orientation of the bounding box spanning the blob is determined.

4.3 Fusion

The fusion is the last stage of the automatic adaptation process. All remaining nodes including their attributes representing the object parts in the given target resolution are compiled to a complete semantic net.

The hierarchical relations between the nodes remain unchanged as long as no scale event occurred. In case of Annihilation, the respective node is simply omitted. For merged blobs only a single *part-of* relation remains. For the Split and Creation events new *part-of* relation are introduced into the respective hierarchy level.

The type of the spatial relations stays unaffected. However, the distances between the object parts are to be adapted. The adapted distance values are derived from the position of the borders of the blobs support regions $Supp_{Rt}$ in target resolution.

5. ADAPTATION EXAMPLE

5.1 Model in high resolution

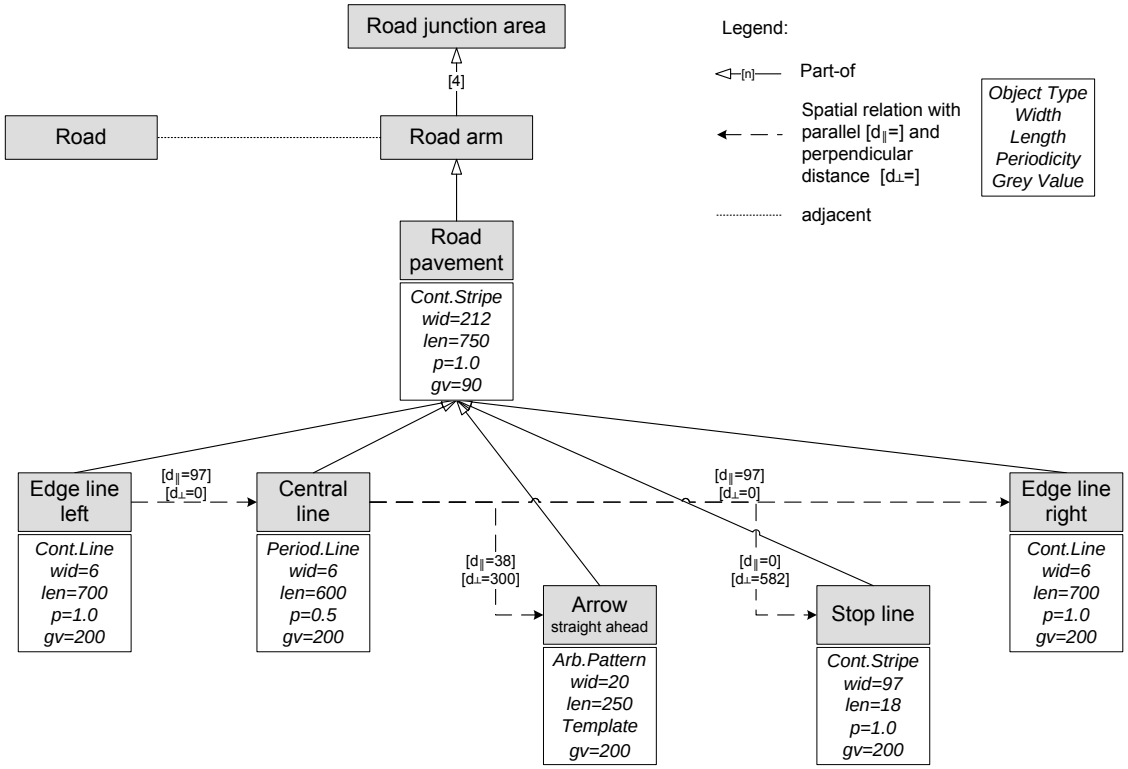


Figure 5. Object model for junction area in a resolution of 0.03-0.05m/pixel

In order to demonstrate the applicability of the adaptation methodology, this section gives an example. A semantic net for the extraction of a 4-arm road junction with 2D symbols (arrows and stop lines) serves as given model for high resolution (3-5cm per pixel). We chose σ_t=15 as target scale; corresponding to an approximate target resolution of $R_t \approx$ 1.5m. This junction model can be seen as a further development of work on road models consisting of linear parallel object parts towards modelling a road network. The node for an adjacent *Road*, for instance, could be represented by the model given in [Heller & Pakzad, 2005].

The junction consists of four identical road arms with a dashed central line. These arms converge and meet at a right angle in the junction area. The junction area contains modified lane markings and additional traffic markings (in our example direction arrows and stop lines). The width of the road pavement is modelled here to be constant. The model assumes a constant roadway width until the junction centre. The 2D position of object parts, given by distances, start from the beginning of the junction arms in direction towards the junction centre. The model also contains information concerning the extent of its object parts, i.e. also the length and width of the linear object parts and of the bounding boxes for arbitrary pattern. Since the assignment of suitable image analysis operators is a requirement for the composition of adaptable models, cf. section 3, all modelled object parts in the image have connections to a particular image analysis operator. The operators for the object parts of type pattern use cross-correlation matching with provided templates, whereas the line-type objects use the road marking operators developed in [Heuwold, 2006]. The junction model for the high resolution is depicted in Figure 5.

5.2 Adaptation process

The scale behaviour prediction of the junction example is carried out in the scale change analysis stage by an analysis-by-synthesis procedure. In the scale change models the four junction arms are treated separately as long as the interaction of adjacent junction arms affects only a small zone.
The number of blobs in initial scale (5) differs from the number of blobs in target scale σ_t (6), suggesting the occurrence of a scale event: a Split event is detected for the right edge line. Here, the corresponding blob splits up due to the strong influence of the nearby arrow blob and the adjacent stop line. The results of the blob detection in the synthetic images of the initial and target scale, illustrated in Figure 6b) and c), reflect the postulated condition for Split events from section 4.2.1 – the two support regions $Supp_t(B_1)$ and $Supp_t(B_2)$ in target scale σ_t intersect the support region $Supp_0(B_S)$ of the splitted blob from initial scale and the initial extremum $E_{S,0}$ intersects both support regions in σ_t.
For all resulting six blobs the node attributes are derived from the target resolution image L_{Rt}, cf. Figure 6d): as object types there is one *Continuous Line* (left edge line) and five *Arbitrary Pattern*; their extents are given by the enclosing bounding boxes of their blob support regions $Supp_{Rt}$; the grey values are determined from the blob contrast in its extent. The distances as attributes of the spatial relations (edges) are derived from the position of the blob support regions for area primitives.

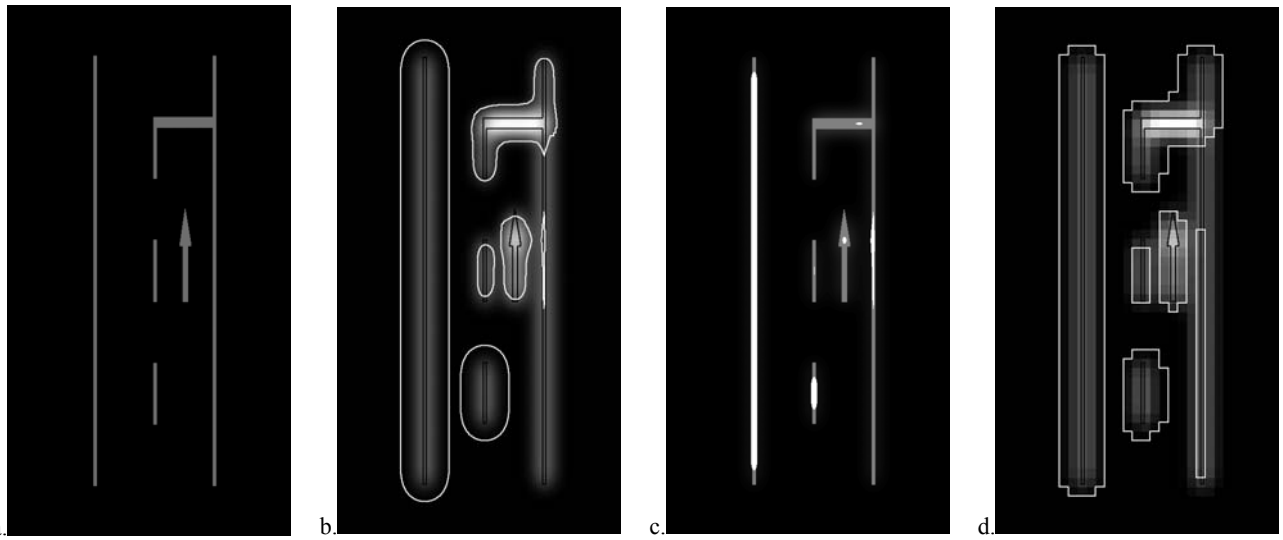


Figure 6. Blob detection results: initial blob features and target blob features superimposed on synthetic images;
a. initial image L_0, b. support regions $Supp_0$ (black), $Supp_t$ (white) on target scale image L_t (grey-value stretched), c. extrema E_0 (grey), E_t (white) on target scale image L_t, d. support regions $Supp_0$ (black), $Supp_{Rt}$ (white) on target resolution image L_{Rt} (grey-value stretched and enlarged)

The adapted object model in the target resolution $R_t \approx 1.5$m is illustrated in Figure 7.

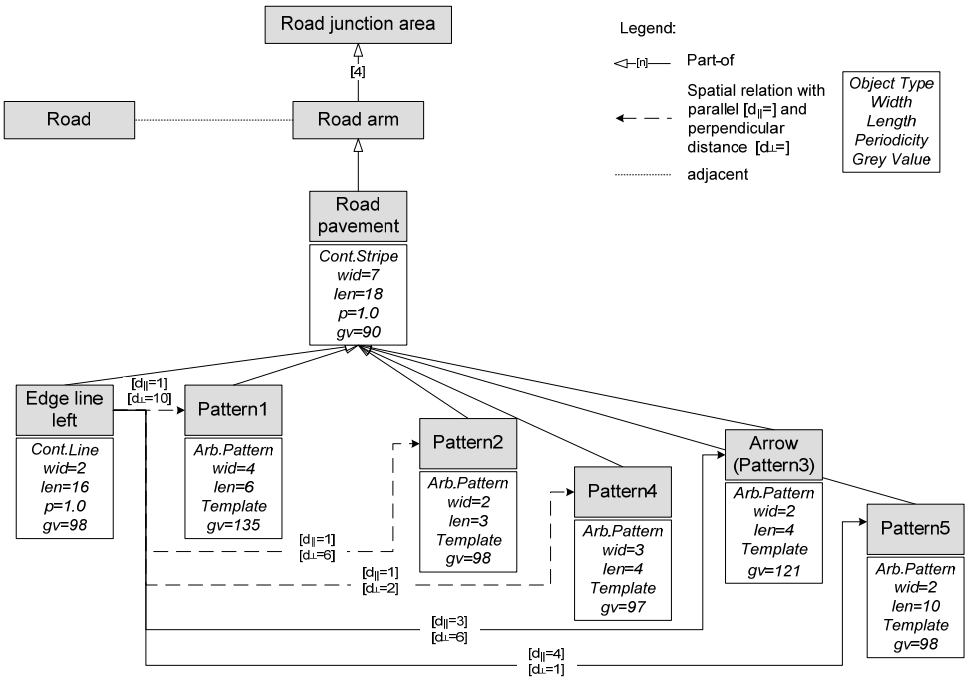


Figure 7. Adapted example object model in target resolution $R_t \approx 1.5$

6. CONCLUSIONS

In this paper a new methodology was presented for the automatic adaptation of image analysis object models, created for a specific high-resolution, to a lower image resolution. The modelled landscape objects can consist of arbitrarily oriented line-type or area-type object parts. In order to adapt the representation of the objects, the scale behaviour of the objects is analysed taking into account scale events and changes in the object's appearance.
The algorithm described here for 2D objects still needs to be verified for real image data by application of an example object model for high-resolution and corresponding adapted models for low resolution in a knowledge-based image interpretation system. Moreover, extensions of the methodology are planned regarding the flexibility of the adaptable models. The approach is to be extended for more realistic scene modelling, e.g. the adaptation of more complex image analysis operators for area-type primitives and the incorporation of relevant local context objects into the adaptation process.

ACKNOWLEDGEMENTS

This study has been funded by Deutsche Forschungs-gemeinschaft under grant HE 1822/13. The project is part of the bundle "Abstraction of Geoinformation in Multi-Scale Data Acquisition, Administration, Analysis, and Visualisation".

REFERENCES

Florack, L., Haar Romeny, B., Koenderink, J. and Viergever, M., 1994. Linear Scale-Space. *Journal of Mathematical Imaging and Vision*, 4(4), pp. 325-351.

Forberg, A. and Mayer, H., 2002. Generalization of 3D Building Data Based on Scale-Spaces. *IntArchPhRS,* Vol. XXXIV, Part 4, pp. 225-230.

Hay, G., Dubé, P., Bouchard, A., Marceau, D., 2002. A scale-space primer for exploring and quantifying complex landscapes. *Ecological Modelling*, 153(1-2), pp. 27-49.

Heller, J. and Pakzad, K., 2005. Scale-Dependent Adaptation of Object Models for Road Extraction. *IntArchPhRS,* Vol. XXXVI, Part 3/W24, pp. 23-28.

Heuwold, J., 2006. Verification of a methodology for the automatic scale-dependent adaptation of object models. *IntArchPhRS,* Vol. XXXVI, Part 3, pp. 173-178.

Koenderink, J., 1984. The Structure of Images. *Biological Cybernetics*, 50(5), pp. 363-370.

Kuijper, A. 2002. *The Deep Structure of Gaussian Scale Space Images*. Dissertation, Utrecht University, the Netherlands, 208p.

Kuijper, A., Florack, L. and Viergever, M. 2003. Scale Space Hierarchy. *Journal of Mathematical Imaging and Vision*, 18(2), pp. 169-189.

Lindeberg, T. 1992. Scale-Space Behaviour of Local Extrema and Blobs. *Journal of Mathematical Imaging and Vision*, 1, pp. 65-99.

Lindeberg, T. 1993. Detecting Salient Blob-Like Image Structures and Their Scales with a Scale-Space Primal Sketch: A Method for Focus-of-Attention. *International Journal of Computer Vision*, 11(3), pp. 283-319.

Lindeberg, T., 1994. *Scale-Space Theory in Computer Vision.* Dordrecht, The Netherlands, Kluwer Academic Publishers, 423p.

Mayer, H. and Steger, C., 1998. Scale-space events and their link to abstraction for road extraction. *ISPRS Journal of Photogrammetry & Remote Sensing*, 53, pp. 62-75.

Niemann, H., Sagerer, G., Schröder, S., and Kummert, F., 1990. ERNEST: A Semantic Network System for Pattern Understanding. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(9), pp. 883-905.

Pakzad, K. and Heller, J., 2004. Automatic Scale Adaptation of Semantic Nets. *IntArchPhRS*, Vol. XXXV, Part B3, pp. 325-330.

Tönjes, R., Growe, S., Bückner, J. and Liedtke, C.-E., 1999. Knowledge Based Interpretation of Remote Sensing Images Using Semantic Nets. *Photogrammetric Engineering & Remote Sensing*, 65(7), pp. 811-821.

Witkin, A., 1983. Scale Space Filtering. In: *Proceedings of the 8th Int. Joint Conference on Artificial Intelligence*, Karlsruhe, pp. 1019-1022.

ROAD EXTRACTION IN SUBURBAN AREAS BASED ON NORMALIZED CUTS

A. Grote *, M. Butenuth, C. Heipke

Leibniz Universität Hannover, Institute of Photogrammetry and GeoInformation, 30167 Hannover, Germany - (grote, butenuth, heipke)@ipi.uni-hannover.de

KEY WORDS: Segmentation, Normalized Cuts, Road extraction, Urban areas, Grouping

ABSTRACT:

This paper deals with road extraction of high resolution aerial images of suburban scenes based on segmentation using the Normalized Cuts algorithm. The aim of our project is the extraction of roads for the assessment of a road database, however, this paper is restricted to road extraction. The segmentation as our basic step is designed to yield a good division between road areas and the surroundings. We use the Normalized Cuts algorithm, which is a graph-based approach that divides the image on the basis of pixel similarities. The definition of these similarities can incorporate several features, which is necessary for the segmentation in complex surroundings such as built-up areas. The features used for segmentation comprise colour, hue, edges and road colour derived with prior information about the position of the centerline from the database. The initial segments have to be grouped due to an enforced oversegmentation. The grouping is based on the criteria of mean colour difference, edge strength of the shared borders and colour standard deviation of merged initial segments. The grouped segments are then evaluated using shape criteria in order to extract road parts. Results on some test images show that the approach provides reliable road parts. Concluding remarks are given at the end to point out further investigations concerning the evaluation of the road segments and their use in database assessment.

1. INTRODUCTION

Roads are among the most important objects that are extracted from aerial images; they are necessary for many applications, for example navigation systems or spatial planning. Extracted roads are recorded in geospatial databases. As roads are subject to frequent changes, it is necessary to check road databases frequently to eliminate errors and to add new road objects. Manual database assessment is very time-consuming, which is why automatic database assessment is a major research topic (Zhang, 2004; Gerke, 2006). In many approaches, up-to-date aerial or satellite images are used to automatically extract objects and to compare them to objects contained in the database (Baltsavias, 2004).

Many approaches for road extraction have been developed; some of them are summarized in (Mayer et al., 2006). However, only few approaches work in urban or suburban areas due to the highly complex structure found in urban scenes which complicates the task of automatic road extraction.. In (Price, 1999; Youn and Bethel, 2004) the road network is expected to be a more or less regular grid but this constraint is not suitable for many European urban areas. Another approach uses a very sophisticated road and context model and is based on grouping small extracted entities to lanes, carriageways and road networks (Hinz, 2004). It employs a large set of parameters that must be carefully adapted for different scenes. In recent work, colour properties are exploited, for example in (Zhang and Couloigner, 2006): the authors perform a pixel-based multispectral classification and use shape descriptors to reduce the number of misclassifications. But they still only have a completeness and correctness rate of approximately 50 per cent. In our opinion, a reason for this is that the multispectral classification does not take into account the spatial relations of the pixels and that colour and shape properties are treated separately.

From the above mentioned works we can deduce that a proper segmentation algorithm is essential for the extraction of roads in suburban areas and that it is important to combine several features for the segmentation. A simple line based road model as used in many road extraction approaches for rural areas is not applicable. Consequently, this paper deals with road extraction in suburban scenes with a focus on segmentation. For segmentation we use Normalized Cuts, a graph-based method which is capable of combining several features to describe pixel similarity. After the segmentation, the initial segments are grouped to reverse oversegmentation. The result are segments that are further evaluated in order to find road segments.

In section 2 the Normalized Cuts algorithm is explained. Our approach for segmentation and grouping is described in section 3, results are presented in section 4. Some conclusions and an outlook on further work are given in section 5.

2. NORMALIZED CUTS

In this section, the Normalized Cuts method, which is used in our approach as starting point, is described in brief. Normalized Cuts is a graph-based method which is used to divide an undirected graph with weighted edges into segments with similar features. The method and its use in image segmentation are described in detail in (Shi and Malik, 2000). Pixels are defined as nodes and connected by weighted edges. The weights describe the similarity between the connected pixels. Theoretically, every pixel can be connected to all other pixels, but in practice only pixels in the vicinity of one pixel are connected with weights different from zero. The similarity

* Corresponding author.

measure is chosen according to the application, it is also possible to combine several similarity criteria. After defining the graph in this way, it is divided into segments aiming at a large dissimilarity between different segments and a large similarity inside each segment. This goal is achieved by cutting the graph such that it meets the following minimization condition:

$$Ncut(A_1,...,A_n) = \sum_{i=1}^{n} \frac{link(A_i, V \setminus A_i)}{link(A_i, V)} = \min \qquad (1)$$

The graph is divided into n sets of nodes A_i. V is the set that contains all nodes in the whole graph. *Link* is the sum of all weights of the connecting edges between two sets:

$$link(P,Q) = \sum_{p \in P, q \in Q} w(p,q) \qquad (2)$$

where $w(p,q)$ is the weight between two nodes p and q belonging to the two sets P and Q. The weight assigned to each pixel pair is inserted into a similarity matrix. The matrix is symmetric, its row and column dimensions are equal to the number of pixels. The minimization is obtained by computing the eigenvectors of a matrix derived from the similarity matrix. Multiple eigenvectors are calculated for multiple segments. The details of the calculation can be found in (Shi and Malik, 2000). The result is a set of discretised eigenvectors with the same number of elements as the number of pixels. The number of segments wanted has to be specified before the calculation because each eigenvector defines one segment.

One advantage of this method is the possibility to combine several different features in one step, a property that is important in complex surroundings. The difficulties that arise from the task of combining the results of several segmentation steps can be avoided in this way. Another advantage is that the algorithm takes both local and global characteristics into account. Local characteristics are incorporated in the similarity matrix which contains the weights of neighbouring pixels. In this way the similarity of pixels in a close neighbourhood is regarded. Global characteristics are considered when the optimal cut is calculated: a global minimum criterion must be met. This is a considerable advantage of the method, because in this way, small disturbances like short or weak edges are ignored by the algorithm.

3. APPROACH

As mentioned before, road extraction in suburban areas is more complicated than in rural areas due to the inhomogeneous background. We use an area based road model and apply our strategy to high resolution CIR images.

3.1 Road Model

The road model shown in Fig. 1 is adapted from (Hinz, 2004). A road segment consists of one or more lanes that are bordered by the roadsides or road markings, these borders are parallel. Other road markings can be found on the lanes, for example arrows or zebra crossings. The road surface appears in the image as a more or less homogeneous area while the roadsides appear as edges and the line markings as bright lines. These properties are used for the definition of the pixel similarities which are used in turn for the Normalized Cuts algorithm.

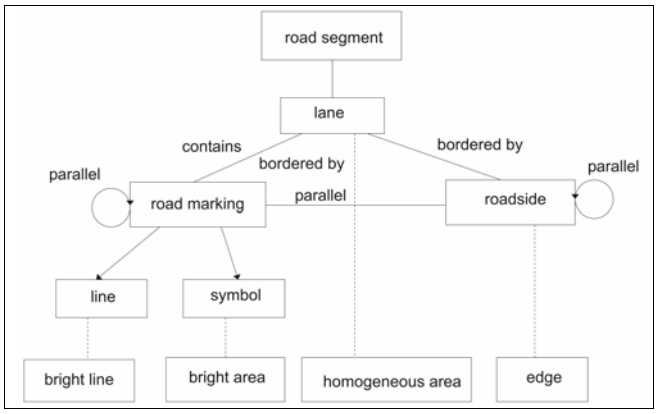


Figure 1. Road model, adapted from (Hinz, 2004).

3.2 General Strategy

The overall aim of our research is the assessment of a road database. Therefore, it is not necessary to analyse the whole image. The search space is restricted to a region of interest whose length matches the length of the road part to be assessed and whose width exceeds the expected road width in order to avoid forcing extracted segments into a road shape, thus distorting the evaluation.

The image is segmented using the Normalized Cuts method as described in detail in subsection 3.3. The result is an oversegmentation which is necessary to make sure that as many parts as possible of the road border belong to a segment border, even if the image information is weak. Therefore, the initial segments have to be grouped in a second step to coherent bigger segments before they can be evaluated and divided into road segments and non-road segments. It is not possible to evaluate the initial segments correctly based on their shape characteristics because they are too small for deriving reliable shape attributes. The grouping is currently done with a simple iterative algorithm merging initial segments with similar mean colour, weak edges at the shared border and low overall colour variance. After grouping, the segments are evaluated in order to extract road parts. The evaluation is mainly based on shape criteria, additionally the NDVI is used. The goal of the evaluation is the extraction of reliable road objects, keeping the number of false road objects to a minimum.

3.3 Segmentation

For the segmentation the Normalized Cuts method described in Section 2 is used. The region of interest is divided into small subsets of equal size and for each of the subsets the similarity matrix is set up individually because the similarity matrix is large even for smaller images and thus the computational resources needed become prohibitive for larger images. The aim of the segmentation is to separate the road parts of the image from the non-road parts. Each segment should contain only road pixels or non-road pixels, not a mixture of both. Therefore, the similarity criteria are derived from the road model. The similarity criteria are:

- Presence and strength of edges between two pixels

- Colour difference
- Hue difference
- Road colour derived from database information.

According to the model, roads are divided from their surroundings by edges, therefore edges are used as a criterion: if there is an edge between two pixels, the pixels are considered dissimilar. The edge criterion is incorporated into the similarity matrix in the following way: first, a Laplacian of Gaussian operator is applied to the image which yields an image of the second derivative in which edges are indicated by sign changes. For each pixel pair it is checked whether a sign change occurs along their connecting line or not. If a sign change occurs, the edge amplitude at this point is taken from an edge image calculated with the Canny operator. The edge amplitude is used to calculate the first part of the similarity measure, following the way described in (Shi and Malik, 2000):

$$w_{edge} = e^{\frac{-f^2}{2\sigma^2}} \quad (3)$$

f is the edge intensity and σ is ten per cent of the range of the edge intensity.

The second criterion is colour because roads usually have homogeneous surfaces and the pixel colour stays approximately the same. A measure for the colour similarity of two pixels is the distance between these two pixels in colour space. The colour similarity is calculated in the same way as the edge similarity in Equation 3. Here, f is the distance, and σ is defined as ten percent of the possible range of distance vector lengths.

These two similarity measures are multiplied to form a combined similarity measure:

$$w = w_{edge} \cdot w_{colour} \quad (4)$$

As a third criterion hue is used because a significant hue difference almost certainly indicates a different object. In parts darkened by shadows the hue of an object remains the same if certain conditions are met (Perez and Koch, 1994). Therefore, the weight is reduced by multiplying with a scale factor smaller than 1 (hue scale factor) if the two pixels have a hue difference that is greater than a defined threshold (hue threshold). A hard threshold is used because the weight should not be diminished at all if the hue difference is small but it should decrease significantly as soon as the difference exceeds the threshold, indicating two different objects. There is certainly some correlation between the colour and the hue criterion; however, our experiments have shown that the use of both yields the best results.

The database information is used to obtain colour information about the roads: assuming that the position of the road is approximately correct, we compute the average colour values of the road in the image for each channel from the position of the database road centerline. For every pair of pixels it is checked if both pixels lie inside an interval of the average values defined by the standard deviation. If both pixels lie inside or outside this interval in all channels, the weight is multiplied by a scale factor larger than 1 (road colour scale factor). The weight is set to 1 if it exceeds 1. By this means, it becomes more likely that segments are divided along a road border.

After the similarity matrix is set up using the defined similarity criteria, the Normalized Cuts algorithm is applied. The number of segments that has to be specified before the algorithm is started must be large enough to prevent merging of road and non-road segments.

3.4 Local Grouping

The image segmentation algorithm results in an oversegmentation. This is necessary in order to avoid the loss of any important road borders. The initial small segments are then grouped to bigger, more meaningful segments before being further evaluated. In the literature, there are only few examples dealing with grouping of image regions. One of them is (Luo and Guo, 2003): they aim at a general grouping algorithm as a bridge between image segmentation and high-level extraction algorithms. The region properties they use include, among others, the colour mean difference between two regions and the edge strength along a shared border. These two criteria are particularly suitable for our approach, because using them as merging criteria can reverse the enforced oversegmentation, which often produces segment borders at places where the image information does not justify a separation (see the results in section 4.1).

At present, we use a simple iterative approach for grouping the segments: in each iteration step two segments are merged, with regard to several criteria that are calculated for each pair of initial segments, partly based on the similarity criteria used by (Luo and Guo, 2003):

- Difference of mean colour (separate for the three bands)
- Edge strength of the intensity channel in the region around the shared border (border region)
- Joint standard deviation of colour of the regions if merged (separate for the three channels)

The border region is a seven pixel wide band along the shared border. For all criteria, the calculated values have to be below defined thresholds for the segments to be considered for merging. The thresholds are determined empirically. In each iteration step, only the two segments with the best values for all criteria (the least colour differences, the least edge strength and the least colour standard deviations) are merged. The iteration continues until the values for every segment pair exceed the thresholds.

3.5 Evaluation

The next step after grouping is the evaluation of the segments in order to extract road parts. The evaluation is based on shape and spectral characteristics of roads. The following characteristics are currently used for evaluation:

- Elongation
- Width
- Rectangularity
- NDVI

The elongation indicates the difference of the object from a circle. It is given by the ratio of the squared perimeter and the area of the segment. Road parts should have a high elongation and thus a high ratio. The width of a road part should not be much larger than the average width of a road. The

rectangularity is a measure for the similarity of an object with a rectangle. It is calculated using the discrepancy method described in (Rosin, 1999). In this method the region of the object is compared to the region of the bounding rectangle of an ellipse with the same first- and second-order moments as the object region. The fourth criterion, the NDVI (Normalized Difference Vegetation Index), is employed because road parts should not contain vegetation. The average NDVI is calculated for each segment and for road parts the NDVI should be low.

In our tests, thresholds are defined empirically for each of these criteria and segments are extracted as road parts if they fulfil all of them. The thresholds are set rather strict in order to extract reliable road parts at the expense of missing some of them.

4. RESULTS

The approach was tested on CIR aerial orthoimages with a resolution of 0.1m. The images are from a suburban scene in Grangemouth, Scotland. Road database data were simulated by manually digitizing the visible roads in the images.

4.1 Segmentation

For segmentation with Normalized Cuts, the images are divided into subsets of approximately 200 x 200 pixels. Each subset is segmented by the Normalized Cuts algorithm yielding 20 segments, an empirical value that is suitable for this image size and scene complexity. The width of the region of interest is set to approximately three times the expected road width. The hue scale factor is set to 0.01, the hue threshold is set to 30. This value was derived from some manually taken sample objects. The road colour scale factor is set to 2.

Fig. 2 and Fig. 3 show examples of segmentation results obtained with the similarity criteria that are described in section 3.3. Segment borders are indicated by yellow lines; the green line shows the database road centerline. The results demonstrate that the segmentation in general has succeeded: road and non-road areas are in most instances clearly divided by initial segment borders. Exceptions can be found in shadow areas or where the contrast between the road and the surroundings is low, as in Fig. 3 in the right part of the image.

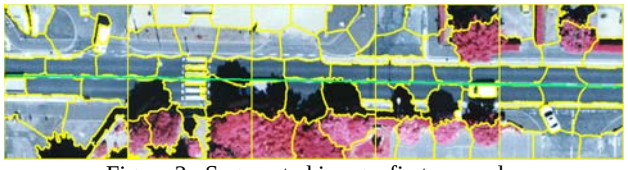


Figure 2. Segmented image, first example.

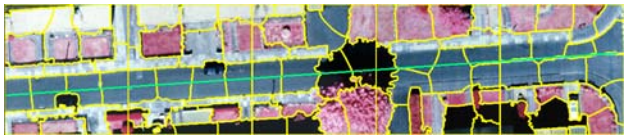


Figure 3. Segmented image, second example.

Fig. 4 shows an example where the database information is not used to obtain the road colour information as described in section 3.3. Consequently, the road colour is not considered in the segmentation. This example points out the benefit of using the prior information of the database: in Fig. 4 more road segments contain non-road areas than in Fig. 3.

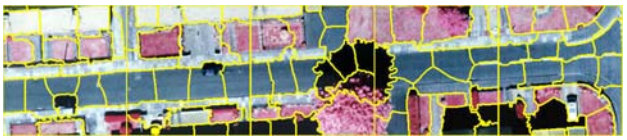


Figure 4. Segmentation without using road colour information derived from the database road.

As the results are to be used for the assessment of the road database, one has to consider the case that the database information is not correct. Fig. 5 shows a segmentation result with a false database road. This example illustrates that incorporating the database information into the segmentation does not lead to wrong segmentation results: the initial segments are clearly defined by image content. Accordingly, the database information does not corrupt the results if it is wrong but can improve the segmentation results.

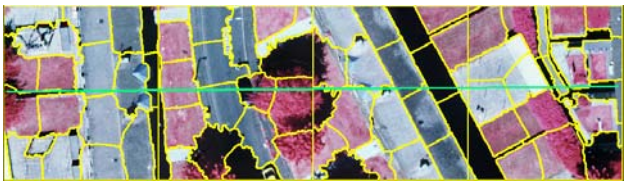


Figure 5. Segmented image with false database road.

4.2 Local Grouping

The initial segmentation results are now grouped. Fig. 6 shows the grouping result from the segments of Fig. 2, Fig. 7 shows the grouping result from the segments of Fig. 3, and Fig. 8 shows the results from Fig. 5.

The parameters for grouping are the same for all three images. The mean colour of two segments to be merged should be the same, the maximum for the mean edge strength is set to 50, and the maximum standard deviation for two merged images is set to 40.

In Fig. 6 most of the undisturbed road parts are merged into two bigger parts. These parts show some characteristics for road segments: their shape is elongated with parallel lines; their width corresponds to the road width. Parts that contain shadows, vehicles or salient road markings (zebra crossing) are not merged with the other road parts. Here, context knowledge is essential for further evaluation.

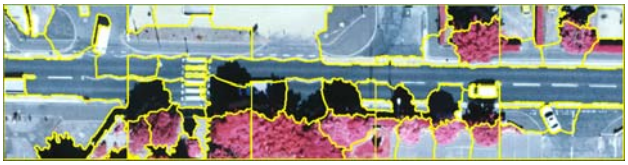


Figure 6. Grouping result, first example.

The second example, shown in Fig. 7, contains a road which is mostly undisturbed by context objects. The grouping result shows one distinctive road segment in the left part of the image. To the right, there is one big road segment that is more problematic: it contains one part of a parking lot and some parts of the pavement. The pavement parts are parts of an initial road segment and are not critical because they do not affect the overall shape of the road to a great extent. The parking lot poses more difficulties for an evaluation but such a case cannot be avoided because the parking lot has the same colour

characteristics as the road and is not separated by a distinct border.

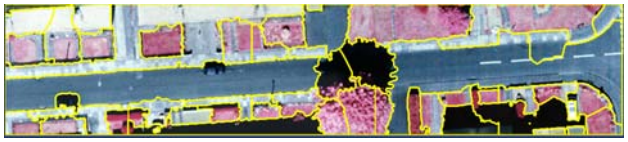


Figure 7. Grouping result, second example

Fig. 8 shows the grouping result for the example with the wrong database information. The segments here are again grouped to meaningful bigger segments. As there are no road-shaped segments along the direction of the database road, this road probably would be rejected in an evaluation step.

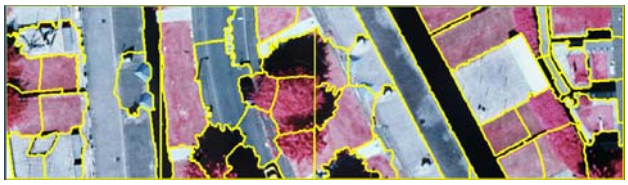


Figure 8. Grouping result, false database example

4.3 Evaluation

In the next step, the grouped segments are evaluated using the criteria described in section 3.5. The figures 9 and 10 show the experimental evaluation results for the first and second example. The thresholds used for the evaluation are: elongation more than 40, rectangularity more than 0.6, width no more than two meters above assumed road width, NDVI less than 0.

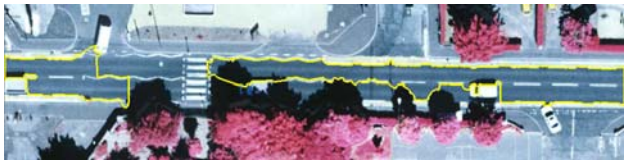


Figure 9. Evaluation result, first example.

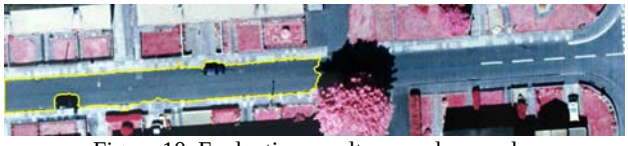


Figure 10. Evaluation result, second example.

In Fig 9, two segments were correctly extracted as road parts, in Fig. 10 one segment was found. No false road parts are extracted, which is important because the goal is a reliable rather than a complete extraction.

5. CONCLUSIONS AND OUTLOOK

The exemplary results in this paper show the general usability of the approach for the detection of roads in aerial images of suburban areas. The Normalized Cuts algorithm is suitable for the segmentation step. By considering global aspects of the image as well as local ones, the algorithm is able to ignore noise, small surface changes and weak edges, and borders are rather placed at continuing edges, as can be seen in the results. The division of the image depends on the overall image content which allows the segments to be more coherent and perceptually meaningful than segments obtained by a local segmentation only. We believe that this combination of local and global aspects is a very important characteristic of the Normalized Cuts algorithm.

One drawback of the Normalized Cuts algorithm is that the calculation is computationally expensive and the image has to be divided into smaller subsets to make the calculation possible with our current hardware. As a consequence, some segment borders are defined by the subset borders and not by image content. In many cases, this does not pose a problem because the segments are still merged in the grouping step, but not always. Another drawback is the fact that the number of segments has to be determined before starting the calculation. It is desirable to find a way to estimate the appropriate number of segments from the given data. One possibility is an iterative approach, repeating the Normalized Cuts algorithm with a varying number of segments and selecting the optimal segmentation. Possible criteria for a good segmentation are average segment size (not too small) and a satisfying homogeneity.

The grouping results show that it is possible to use the oversegmented results from the Normalized Cuts algorithm and group them to bigger segments whose shape can be assessed regarding their correspondence with the road model. The grouping works well for road parts without many disturbances by context objects. One problem are areas that are directly connected to the road and have the same colour as the road, like the parking lot in Fig. 8. Here, an additional grouping criterion, for example border continuation, could be helpful. The grouping algorithm itself, especially the combination of the different criteria, could also still be improved.

The first experimental evaluation results show that reliable road parts could be extracted, but there is still much room for improvement. For example, road parts should be close to rectangular in our current implementation of the evaluation. This can pose problems with long road parts that belong to curved roads. Therefore, we plan to change this criterion into one that requires an elongated object to have a constant width.

Our next steps will be the improvement of the grouping and evaluation steps, as indicated above. We will also investigate if the number of thresholds currently employed can be reduced and if the remaining thresholds can be estimated from the image data themselves. As the goal of our project is the assessment of a road database, we will use the extracted road parts for the assessment, adapting the strategy developed by (Gerke, 2006). In connection with the assessment, context objects like trees, vehicles, buildings and shadows will also be considered, in order to explain gaps between extracted road parts.

ACKNOWLEDGEMENTS

This project is funded by the DFG (German Research Foundation). The calculations were made with a C++ program partly adapted from a MATLAB program written by Timothée Cour, Stella Yu and Jianbo Shi. Their program can be found on http://www.seas.upenn.edu/~timothee/software_ncut/software.html (last checked July 2007).

REFERENCES

Baltsavias, E.P., 2004. Object extraction and revision by image analysis using existing geodata and knowledge: current status and steps towards operational systems. *ISPRS Journal of*

Photogrammetry and Remote Sensing, Vol. 58, No. 3-4, pp. 129-151.

Gerke, M., 2006. Automatic quality assessment of road databases using remotely sensed imagery. *Dissertation, Universität Hannover, DGK Reihe C, Nr. 599: Verlag der Bayerischen Akademie der Wissenschaften*, 105 p.

Hinz, S., 2004. Automatic road extraction in urban scenes – and beyond. In: *International Archives of Photogrammetry and Remote Sensing*, Vol. 35, part B3, pp. 349-355.

Mayer, H., Hinz, S., Bacher, U. and Baltsavias, E., 2006. A test of automatic road extraction approaches. In: *International Archives of Photogrammetry,Remote Sensing and Spatial Information Sciences*, Vol. 36, Part 3, pp. 209-214.

Luo, J. and Guo, C., 2003. Perceptual grouping of segmented regions in color images. *Pattern Recognition*, Vol. 36, No. 12, pp. 2781-2792.

Perez, F. and Koch, C., 1994. Toward color image segmentation in analog VLSI: algorithm and hardware. *International Journal of Computer Vision*, Vol. 12, No. 1, pp. 17-42.

Price, K., 1999. Road grid extraction and verification. In: *International Archives of Photogrammetry and Remote Sensing*, Vol. 32, part 3-2W5, pp. 101-106.

Rosin, P.L., 1999. Measuring rectangularity. *Machine Vision and Applications*, Vol. 11, No. 4, pp. 191-196.

Shi, J. and Malik, J., 2000. Normalized cuts and image segmentation. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 22(8), pp. 888-905.

Youn, J. and Bethel, J.S., 2004. Adaptive snakes for urban road extraction. In: *International Archives of Photogrammetry and Remote Sensing*, Vol. 35, part B3, pp. 465-470.

Zhang, C., 2004. Towards an operational system for automated updating of road databases by integration of imagery and geodata. *ISPRS Journal of Photogrammetry and Remote Sensing*, Vol. 58, No. 3-4, pp. 166-186.

Zhang, Q. and Couloigner, I, 2006. Automated road network extraction from high resolution multi-spectral imagery. In: *Proceedings of ASPRS Annual Conference, Reno, Nevada, May 2006*.

SEGMENTATION OF TREE REGIONS USING DATA OF A FULL-WAVEFORM LASER

H. Gross, B. Jutzi, U. Thoennessen

FGAN-FOM Research Institute for Optronics and Pattern Recognition, 76275 Ettlingen, Germany
{gross,jutzi,thoe}@fom.fgan.de

Commission I, WG I/2

KEY WORDS: Laser data, full-waveform, point clouds, intensity, environment, covariance of points, eigenvalues, segmentation.

ABSTRACT:
The new sensor technique for receiving full-waveform laser data delivers not only height information but also additional features like intensity. The echoes generated by reflecting objects are approximated by amplitude and width of a Gaussian kernel. These features are used for tree region segmentation. The inclusion of the available information in the environment around each 3d point of the cloud can be used to stabilize the segmentation process. Two kinds of environments spherical and cylindrical are discussed. The calculation of the covariance inside these environments is discussed without and with weighting all points by their measured intensity. An application of the described method is demonstrated for a typical scene. A method for the correction of the number of points inside an environment produced by the movements of the sensor platform is presented. Two proposals are described for the segmentation of trees. One of them uses the eigenvalues of the covariance matrix in an environment. The other method considers only the features of the echoes for each laser pulse without considering an environment.

1. INTRODUCTION

The automatic generation of 3d models from laser scanning data to gain a description of man-made objects is of great interest to the photogrammetric research (Brenner *et al.*, 2001; Geibel & Stilla, 2000; Gross *et al.*, 2005). Spaceborne, airborne as well as terrestrial laser scanning systems allow a direct and illumination-independent measurement of laser scanning data from 3d objects in a fast, contact free and accurate way.

Recent developments of commercial airborne laser scanners led to small footprint laser systems that allow capturing the waveform of the backscattered laser pulse, namely the OPTECH ALTM 3100, TOPEYE MK II, and TOPOSYS HARRIER 56. The latter one is based on the RIEGL LMS-Q560. These systems mentioned above are specified to operate with a transmitted pulse width of 4-10 ns and allow digitization and acquisition of the waveform with approximately 0.5-1 GSample/s.

To interpret the received waveform of the backscattered laser pulse, a fundamental understanding of the physical background of pulse propagation and surface interaction is important. The waveform offers the possibility to study different features like the *range, elevation variations*, and *reflectivity* of the surface based on the inclination between the divergent laser beam and object plane. These specific features have an influence on the received waveform. The waveform of each pulse is described by a series of range values combined with amplitude values and can be approximated by one or more parameterized Gaussian curves (Hofton *et al.*, 2000). Due to this approximation, specific features like *temporal position*, *width* and *amplitude* (cf. Figure 1) caused by the object surfaces are estimated (Jutzi & Stilla, 2006).

Nowadays, the analysis of full-waveform data is more and more of interest especially for forestry applications, because it provides the opportunity for a detailed investigation of vertical distributed surface structures (Hug *et al.*, 2004). With full-waveform analysis additional points can be extracted from the measured data compared to the conventional first pulse and last pulse technique. These additional points and their corresponding surface can lead to a better description of vertical structures like vegetation (Persson *et al.*, 2005). Furthermore these features can be used for segmentation and classification beside the geometric information (Wagner *et al.*, 2006). For instance, Reitberger *et al.* (2006) demonstrated that the features and the additional points, derived by full-waveform decomposition, are useful to classify tree species.

In this paper we focus on the features derived by full-waveform decomposition to segment vegetation, namely trees. Beside the raw features, like signal amplitude, signal width, and total number of echoes for each emitted laser beam, a spherical and cylindrical environment in the close neighborhood is used to improve the segmentation. Both methods are compared to each other by a ROC (Receiver Operating Characteristic) curve to evaluate the detection and the false alarm rate.

In Section 2 a brief description of the used full-waveform data is given. The used raw data and volumetric approaches are presented in Section 3. The methodology to calculate additional point features based on the covariance matrix is explained in Section 4. We visualize in Section 5 the scene and the corresponding features derived from the measured laser scanning data. A required correction of features is proposed in Section 6. The segmentation by using the volumetric and the raw data approach is described in Section 7. The results are presented in Section 8 including a comparison between ground truth data and ROC curves. Finally the used methods and derived results are discussed in Section 9.

2. FULL-WAVEFORM DATA

We operate on data measured by a RIEGL LMS-Q560 sensor with a field of view of 60° and a point density of about 3.7 points per m². The flying height was about 400m above ground. For each beam the total number of detected backscattered pulses is known and assigned to the corresponding echoes. Each echo is described by a point with its 3d coordinate, signal amplitude, and signal width derived from the Gaussian approximation.

Figure 1 shows the received signal produced by three different objects along the beam corridor. The shape of the received waveform depends on the illuminated surface area, especially on the material, reflectivity of the surface and the inclination angle between the surface normal and the laser beam direction.

The intensity (energy) is estimated by the width multiplied with the amplitude of the Gaussian approximation and corrected by the range between sensor and object. It describes the reflectivity influenced by geometry and material of the object at this point. For each particular echo caused by partially illuminated object surfaces an own intensity value is received.

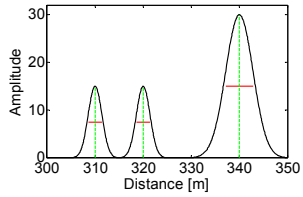


Figure 1. Example for three received echoes derived by a single emitted laser pulse with several signal widths (solid lines) and amplitudes (dashed lines).

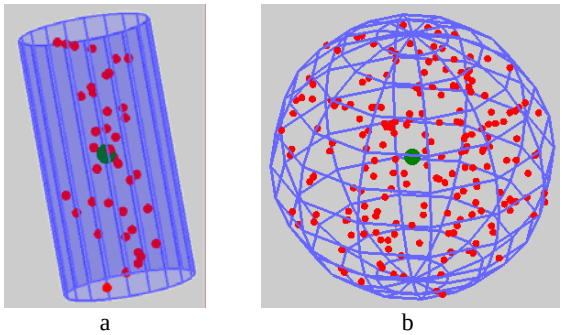
3. RAW DATA AND VOLUMETRIC APPROACH

Due to the wide field of view the sensor delivers measurements between the nadir and an oblique angle. This varying angle causes problems discriminating objects. The angle influences the echoes even for the same kind of objects at different positions with respect to the flight path. Particularly for oblique emitted laser beams the received echoes are reflected by objects with essential different horizontal positions. Several approaches to overcome these problems are discussed in this paper.

The Gaussian decomposition of the full-waveform data allows the interpretation of the received echoes with signal widths and amplitudes. We call this the raw data approach.

Another possibility is a volumetric approach. For each point we mark all points inside a sphere (Figure 2b) or a vertical cylinder (Figure 2a) centred at the trigger point. The sphere includes all points inside a restricted 3d environment. The vertical cylinder includes all points inside a 3d environment without restriction on the vertical position.

All measured points calculated by the approximation of the waveforms of the same or different laser beams, which are located within the sphere or the cylinder, are considered as neighbors in the further processing.



a b

Figure 2. Environment with point cloud and centre point:
a) cylindrical environment,
b) spherical environment.

4. POINT FEATURES BASED ON THE COVARIANCE

For the defined volumes the 3d moments, as described by Maas & Vosselman (1999), are discussed and improved by considering the vertical dimension in the same manner as the horizontal ones (Gross & Thoennessen, 2006). After calculation of the covariance an interpretation of the eigenvalues is possible to define discriminating features for planes, edges and corners.

4.1 Moments of the covariance matrix

In a continuous domain, the moments are defined by

$$m_{ijk} = \int_V x^i y^j z^k f(x,y,z)\,dv\,, \tag{1}$$

where $i,j,k \in \mathbb{N}$, and $i+j+k$ is the order of the moment integrated over a predefined volume weighted by $f(x,y,z)$. We use as weighting function the intensity of the reflected laser pulse. Considering only the moments with $i+j+k \le 2$ we get the weight, the center of gravity and the matrix of covariance. Invariance against translation is achieved by subtraction of the center of gravity

$$\bar{x} = \frac{m_{100}}{m_{000}},\quad \bar{y} = \frac{m_{010}}{m_{000}},\quad \bar{z} = \frac{m_{001}}{m_{000}}, \tag{2}$$

and invariance against the units of the coordinates is achieved by normalization with the radius R of the volume cell. The approximation of the integral by the sum over all points inside the cell yields the components of the covariance matrix

$$\tilde{m}_{ijk} = \frac{\sum_{l=1}^{N}\left(x_l-\bar{x}\right)^i\left(y_l-\bar{y}\right)^j\left(z_l-\bar{z}\right)^k f(x_l,y_l,z_l)}{R^{i+j+k}\sum_{l=1}^{N} f(x_l,y_l,z_l)}. \tag{3}$$

Finally we calculate for each point of the whole data set the covariance matrix

$$M = \begin{pmatrix} \tilde{m}_{200} & \tilde{m}_{110} & \tilde{m}_{101} \\ \tilde{m}_{110} & \tilde{m}_{020} & \tilde{m}_{011} \\ \tilde{m}_{101} & \tilde{m}_{011} & \tilde{m}_{002} \end{pmatrix}. \tag{4}$$

The eigenvalues λ_i and eigenvectors $\vec{e}_i$ with $i=1,2,3$ of the symmetrical matrix deliver additional features for each point. The eigenvalues are invariant concerning rotation of the coordinate system. If the weighting function $f(x,y,z)$ depends on the points we calculate the normalized weight by

$$\tilde{m}_{000} = \frac{\sum_{l=1}^{N} f(x_l,y_l,z_l)}{N}. \tag{5}$$

For object classification, West *et al.* (2004) uses the following features which depend on the eigenvalues:

$$\text{structure tensor omnivariance} = \sqrt[3]{\prod_{i=1}^{3}\lambda_i}\,, \tag{6}$$

$$\text{structure tensor planarity} = \frac{\lambda_2-\lambda_3}{\lambda_1}. \tag{7}$$

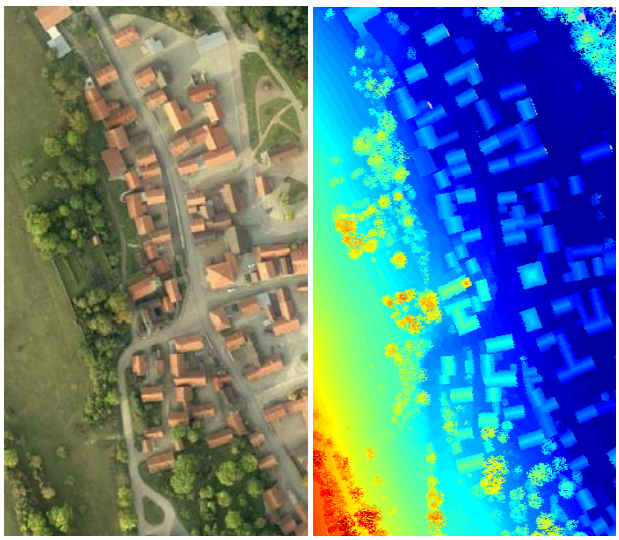
4.2 Weighting of points

For comparison we selected two weighting functions $f(x,y,z)$ given by the equations (1) and (3). Therefore each point is weighted by a constant or weighted by its own intensity value.

5. SCENE DESCRIPTION

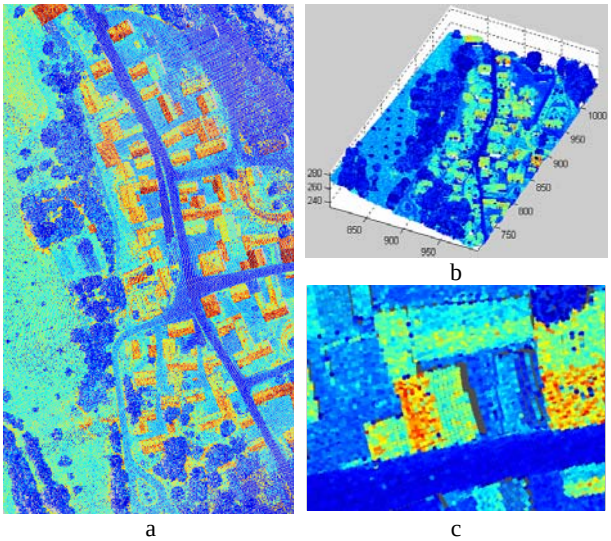
Our investigations are focused on a scene of a rural village as shown in Figure 3a. The scene includes streets, buildings, lawn and trees. The corresponding height image colored by the elevation is depicted in Figure 3b. The left part and the right upper corner show mainly grassland and trees. In the middle part several kinds of buildings with different shape and height can be recognized.

A more precise impression of the data as point cloud shows Figure 4. It demonstrates the influence of the intensity due to the material and the angle between sensor and object plane. We achieve a low intensity for trees and many streets. Higher values arise for grassland. Roof planes yield the highest intensities but vary depending on the angle between sensor and surface orientation (Figure 4c).



a b

Figure 3. a) RGB orthophoto,
b) laser elevation data colored by height.

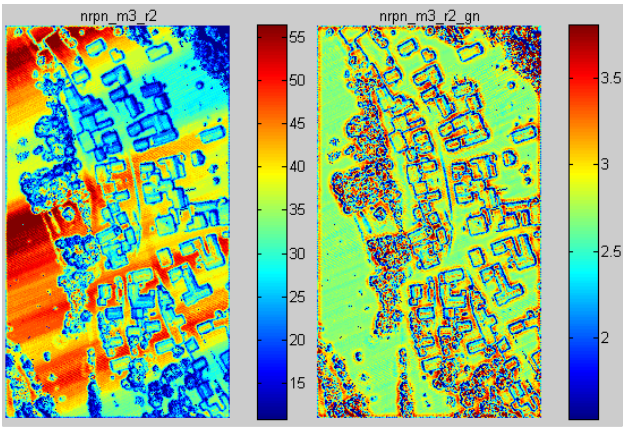


a b c

Figure 4. Point cloud of the laser elevation data colored by intensity:
a) vertical view,
b) oblique view,
c) dependency of the intensity signal by the surface orientation.

6. DATA PREPROCESSING

The received number of echoes depends on the measurement situation. As an example we expect more echoes for tree regions than for grassland or roofs. Figure 5a shows the number of points inside the spherical environment at a radius of $R = 2m$ and demonstrates the variation of this number. This depends not only on the measured objects but also on the movements of the sensor platform. These movements produce not equal spaced scan lines. Therefore it is essential to define a normalization measure for the number of echoes in the sphere.
An image based method is proposed to compensate the point number variation, where the image is a rasterized representation of the point cloud. We filtered the image by a 2d Gaussian kernel with a support region of 17x17 pixels and a standard deviation of 8 pixels. Finally, we subtracted the result from the original image. This calculation steps yield to Figure 5b. The number of echoes has only a small variation at grassland and at roofs, but a high variation at walls and trees. The high values for the walls are caused by the wide field of view of up to 60 degrees. For cylindrical environment we get the same behavior. For high walls more measured points fall into the cylindrical environment especially if the viewing angle differs from nadir angle.



a b

Figure 5. Spherical environment:
a) number of points inside the sphere,
b) normalized number of echoes inside the sphere.

7. SEGMENTATION OF TREE REGIONS

In the following sections the different methods presented above, will be applied to the segmentation of tree regions for example.

7.1 Influence of different environments and weighting

For the selected rural area the omnivariance (equation (6)) is calculated both with a cylindrical and with a spherical environment. Results are depicted in Figure 6a&b. The walls of buildings and tree regions are represented by a high omnivariance, whereas objects which appear like a plane inside the environment are showing small values.

Figure 6a includes only those points lying inside the spherical environment. We get a little more details due to the limited extension of the sphere in the vertical dimension.

Figure 6c&d demonstrate the influence of the weighting on the omnivariance feature. This is shown by the difference of the omnivariance with and without using the intensity of the laser beam as weighting factor during the calculation of the covariance. The dynamic of the difference lies within 10% of the dynamic of the omnivariance and is to the further consideration of minor importance.

By considering the different volumes it is observable, that for a cylindrical environment the trees and walls have greater differences than planes (Figure 6c). Whereas using the spherical environment the difference is not as significant to the walls of the buildings as in the former case (Figure 6d).

7.2 Segmentation of tree regions

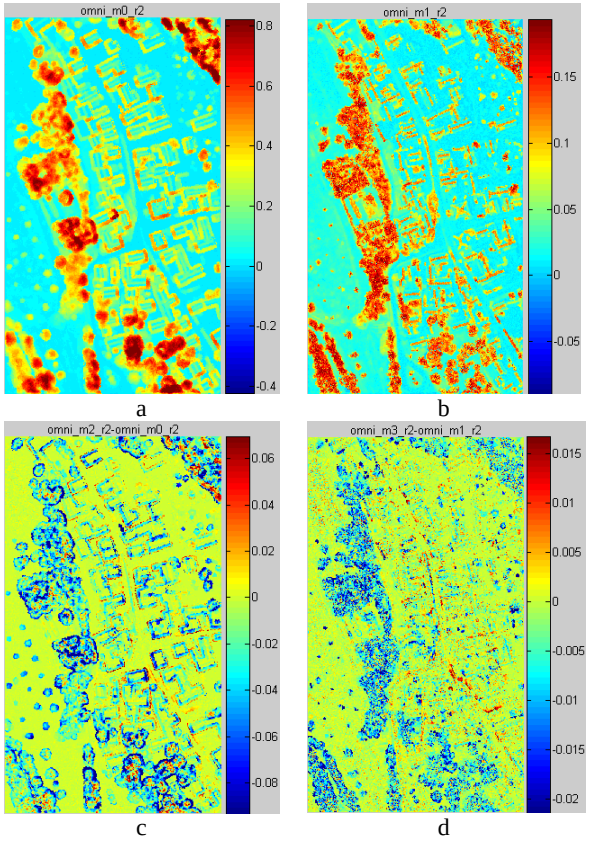


Figure 6. Omnivariance for different environments and weighting of the points: cylindrical environment (a&c), spherical environment (b&d), without weighting by the intensity (a&b), difference between the weighted and not weighted omnivariance (c&d).

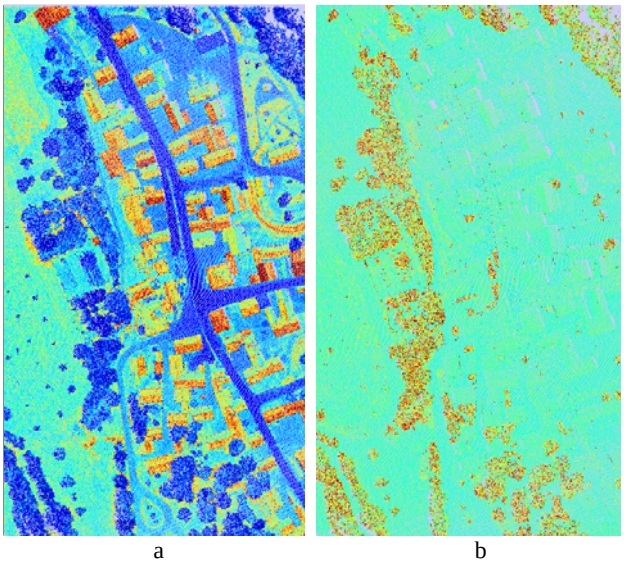


Figure 7. Features useable for tree segmentation:
a) signal amplitude,
b) signal width.

The results of the previous section have shown that the cylindrical volumetric approach is more appropriate for the segmentation of tree regions than the spherical solution. Therefore in the following steps we will only discuss the cylindrical approach vs. the raw data feature exploitation.

The signal amplitude shown in Figure 7a and the signal width presented in Figure 7b are significant features for the detection of tree regions, because the geometry of trees deliver in most cases signal returns with very small amplitudes but high values for the width.

7.2.1 Volumetric approach

Investigating the different features the following behaviour can be observed: Trees have a low reflectivity resulting in a low value for averaged intensity W (equation (5)). But the low reflectivity of streets will have the same effect. Therefore we need an additional discriminating feature. The feature planarity P (equation (7)) shows high values especially for plane objects. In contrast to this the omnivariance O (equation (6)) delivers high values for volumetric objects.

Due to the different ranges of values these features are normalized by the sigmoid function defined by equation:

$$\sigma(x, x_0, s) = 1/\left(1 + e^{s(x-x_0)}\right). \tag{8}$$

For trees we adapt the centre point and scaling factor.
The three features are transformed into

$$\begin{aligned} \sigma_W &= \sigma(W, 1000, 0.3) && \text{for the averaged intensity,} \\ \sigma_P &= \sigma(P, 0.1, 5) && \text{for the planarity,} \\ \sigma_O &= \sigma(O, 0.4, -1) && \text{for the omnivariance.} \end{aligned} \tag{9}$$

Positive scaling values indicate a decreasing of the sigmoid function. Figure 8a shows the values for the planarity after normalization. Plane objects are marked by blue and volumetric objects by red regions. Finally, the three indicators are combined into a single measure

$$\sigma_T := \sigma_W \sigma_P \sigma_O > 0.25 \tag{10}$$

and restricted by a threshold. The result is given in Figure 8b where tree regions are marked. An optimization process with respect to the parameters of the sigmoid function has the aim to gain the best detection and false alarm rate. The results have to be compared to results derived from data of other regions.

7.2.2 Raw data approach

In this section we consider for each point of the cloud its own feature value delivered by the sensor. For segmentation of tree regions the features signal width and amplitude are used, derived by Gaussian decomposition of the full-waveform. Further the feature total number of echoes is considered. They are normalized by

$$\begin{aligned} \sigma_w &= \sigma(w, 45, 0.7) && \text{for the width,} \\ \sigma_a &= \sigma(a, 50, -0.4) && \text{for the amplitude,} \\ \sigma_t &= \sigma(t, 2, -5) && \text{for the total number of echoes.} \end{aligned} \tag{11}$$

The result is shown in Figure 9a for the width of each echo. We define the tree regions for all points with

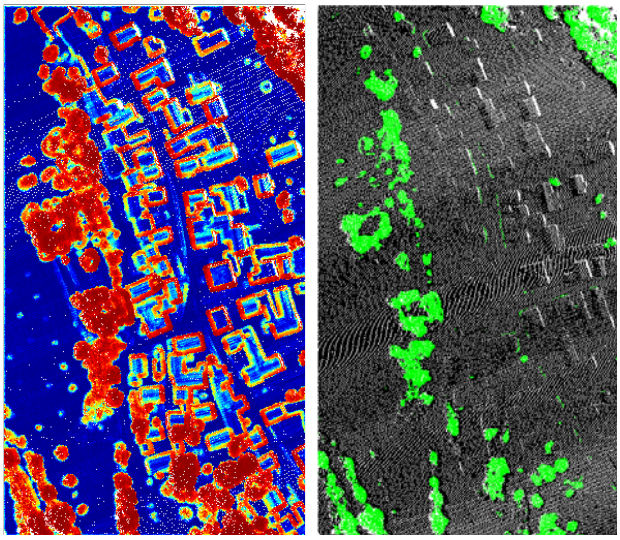
$$\sigma_{Tp} := \sigma_w \sigma_a \sigma_t > 0.1, \tag{12}$$

which delivers Figure 9b.

8. RESULTS

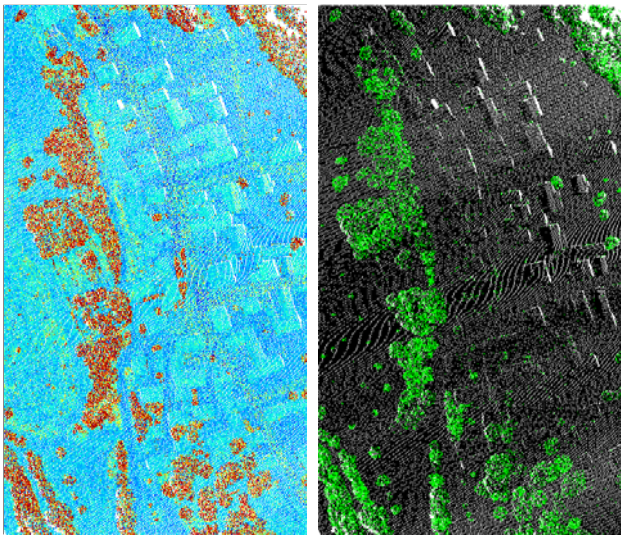
The different approaches discussed in the previous sections are applied to the data. A comparison with Figure 8b the result including the cylindrical environment indicates that tree regions may be segmented more precise by calculation of the

covariance matrix, their eigenvalues and derived features, than by considering only the raw data as shown in Figure 9b. For a final decision it would be necessary to optimize both results separately with respect to a target function by variation of all parameters of the used sigmoid function. The target function has to compare the calculated results to a ground truth definition of the tree regions.



a b

Figure 8. Indicator for tree regions including the environment:
a) based on the planarity,
b) based on width, planarity and omnivariance with threshold.



a b

Figure 9. Indicator for tree regions without consideration of the environment: a) based on the width of the echo, b) based on width, amplitude and total number of echoes with threshold.

After definition of real tree regions (Figure 10) by an interactive determination, we are able to calculate false alarm and detection rate by modifying the thresholds in equation (10) and (12). The tree region information is back propagated into the point cloud to assign the class information to each point.

The ROC curves for both methods are shown in Figure 11. Using eigenvalues by consideration of a cylindrical environment delivers a smaller false alarm rate than the segmentation by the features of the raw data only. On the other hand the detection rate for the raw data approach method is higher than for the volumetric approach.

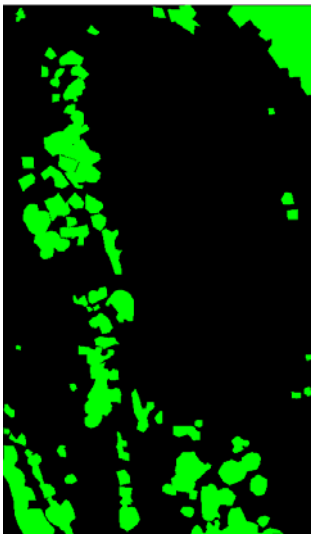


Figure 10. Interactive defined tree regions.

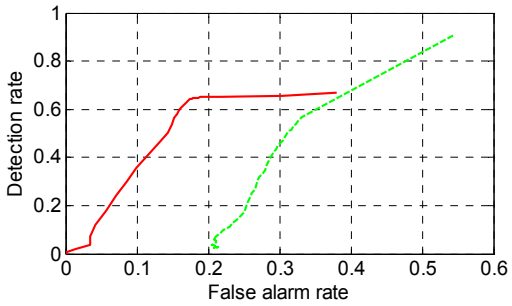


Figure 11. Detection rate vs. false alarm rate for different thresholds of equation (10) including environment (solid line) and equation (12) for the features of the raw data (dash line).

Finally, a 3d visualization of the tree regions based on width, planarity and omnivariance is shown in Figure 12. The tree crowns are represented by ellipsoids. As Straub (2003) mentioned, this assumption is true, if the fine structure of the crown is ignored and the coarse structure is revealed in an appropriate level of the multi-scale representation of the surface model. For visualization purposes the scales of ellipsoids correspond to the eigenvalues and eigenvectors given by the relevant points inside a spherical environment with a $radius = 6m$. On the other side we know by analytical determination the eigenvalues of an ellipsoid as

$$\lambda_i = \frac{a_i^2}{5} \quad \text{with} \quad i = 1,2,3, \tag{13}$$

where a_i are the half axes of an ellipsoid. After calculation of the eigenvalues of the covariance matrix we get the half axes by

$$a_i = \sqrt{5\lambda_i}\,. \tag{14}$$

For each point falling inside the tree regions the ellipsoids are determined. If its centre point lies near the considered point the ellipsoid is accepted. Already processed points within a tree crown are ignored for further processing. The tree trunks with a fix diameter are additionally plotted as junction between bottom and ellipsoid centre to give an impression of the tree position in the 3d visualization.

Figure 12. Virtual trees visualized by ellipsoids.

9. DISCUSSION AND CONCLUSION

New sensor technology of laser scanners delivers more detailed information. The new features include additional information about the measured objects.

The used data does not contain a homogeneous point distribution due to the movements of the sensor. This discrepancy is compensated by an adapted filter.

We consider features derived by calculation of the eigenvalues of the covariance matrix for cylindrical and spherical environments with respect to the intensity of the echoes. For our investigations only data with low point density was available. Therefore the spherical environment should have a large radius to get a representative number of points for a valuable distribution. This distribution of the points can be inhomogeneous (Gross & Thoennessen, 2006). For a cylindrical environment the trees and walls have greater significance than by calculation with a spherical environment. Therefore the segmentation process is executed only for cylindrical environments. The usage of normalized weight, omnivariance and planarity for tree segmentation results in acceptable detection and false alarm rate.

Without including the neighbored point information, only considering the features of the raw data like signal amplitude, signal width and total number of targets, we get a good detection rate but an unacceptable false alarm rate.

Modifications of the parameters of the sigmoid function may influence the values of detection and false alarm rate but not their principle behavior.

The usage of point clouds instead of images requires modified methods for data representation, processing and segmentation.

The analysis of full-waveform data delivers additional features beside the range value. These features are included in the data processing steps for visualizing relevant objects. The full-waveform analysis yields more details and allows a more precise segmentation of the different kind of objects. This will be supported by considering a spherical or cylindrical environment. Former investigations show that a spherical environment is suitable for edge, corner, and plane detection (Gross & Thoennessen, 2006). For the segmentation process of horizontal regions without considering the vertical dimension a cylindrical environment delivers more suitable results.

The decision of cylindrical or spherical environment depends on the kind of object class and should be investigated in more detail in the future.

REFERENCES

Brenner, C., Haala, N. Fritsch, D., 2001. Towards fully automated 3D city model generation. In: Baltsavias, E., Grün, A., van Gool, L. (Eds) Proceedings of the 3rd International Workshop on Automatic Extraction of Man-Made Objects from Aerial and Space Images, pp. 47-56.

Geibel R., Stilla, U., 2000. Segmentation of Laser-altimeter data for building reconstruction: Comparison of different procedures. International Archives of Photogrammetry and Remote Sensing 33 (Part B3), pp. 326-334.

Gross, H., Thoennessen, U., v. Hansen, W., 2005. 3D Modeling of Urban Structures. In: Stilla, U., Rottensteiner, F., Hinz, S. (Eds) Joint Workshop of ISPRS/DAGM Object Extraction for 3D City Models, Road Databases, and Traffic Monitoring CMRT05, International Archives of Photogrammetry and Remote Sensing 36 (Part 3/W24), pp. 137-142.

Gross, H., Thoennessen, U., 2006. Extraction of Lines from Laser Point Clouds. In: Förstner, W., Steffen, R. (Eds) Symposium of ISPRS Commission III: Photogrammetric Computer Vision PCV06. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences 36 (Part 3), pp. 86-91.

Hofton, M.A., Minster, J.B., Blair, J.B., 2000. Decomposition of laser altimeter waveforms. IEEE Transactions on Geoscience and Remote Sensing 38 (4), pp. 1989–1996.

Hug, C., Ullrich, A., Grimm, A., 2004. LITEMAPPER-5600 - a waveform digitising lidar terrain and vegetation mapping system. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences 36 (Part 8/W2), pp. 24–29.

Jutzi, B., Stilla, U., 2006. Range determination with waveform recording laser systems using a Wiener Filter. ISPRS Journal of Photogrammetry and Remote Sensing 61 (2), pp. 95-107.

Maas, H., Vosselman, G., 1999. Two algorithms for extracting building models from raw Laser altimetry data. ISPRS Journal of Photogrammetry and Remote Sensing 54 (2-3), pp. 153-163.

Persson, Å., Söderman, U., Töpel, J., Ahlberg, S., 2005. Visualization and analysis of full-waveform airborne laser scanner data. In: Vosselman, G., Brenner, C. (Eds) Laser scanning 2005. International Archives of Photogrammetry and Remote Sensing 36 (3/W19), pp. 109-114.

Reitberger, J., Krzystek, P., Stilla, U., 2006. Analysis of Full Waveform LIDAR Data for Tree Species Classification. In: Förstner, W., Steffen, R. (Eds) Symposium of ISPRS Commission III: Photogrammetric Computer Vision PCV06. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences 36 (Part 3), pp. 228-233.

Straub, B., 2003. Automatic Extraction of Trees from Aerial Images and Surface Models. In: Ebner, H., Heipke, C., Mayer, H., Pakzad, K. (Eds) Photogrammetric Image Analysis PIA'03. International Archives of Photogrammetry and Remote Sensing 34 (Part 3/W8), pp. 157-164.

Wagner, W., Ullrich, A., Ducic, V., Melzer, T., Studnicka, N., 2006. Gaussian Decomposition and Calibration of a Novel Small-Footprint Full-Waveform Digitising Airborne Laser Scanner. ISPRS Journal of Photogrammetry and Remote Sensing, 60 (2), pp. 100-112.

West, K.F., Webb, B.N., Lersch, J.R., Pothier, S., Triscari, J.M., Iverson, A.E., 2004. Context-driven automated target detection in 3d data. In: Sadjadi, F.A. (Ed) Automatic Target Recognition XIV. Proceedings of SPIE Vol. 5426, pp. 133-143.

INTERACTIVE IMAGE-BASED URBAN MODELLING

Vladimir Vezhnevets, Anton Konushin and Alexey Ignatenko

Graphics and Media Laboratory Lomonosov Moscow State University, Moscow, Russia.
dmoroz@graphics.cs.msu.ru, ktosh@graphics.cs.msu.ru, ignatenko@graphics.cs.msu.ru

KEY WORDS: urban modelling, 3d reconstruction, building reconstruction, semi-automatic, 3D city models, vanishing points

ABSTRACT:

In this paper we describe a novel interactive modelling technique, which allows the non-expert user to create plausible models of urban scenes with 2D image operations only. This technique may be treated as an enhancement of the famous image based modelling and photo editing (IBPE) (Oh et al., 2001) approach towards more automation and ease to use. The are several contributions in this paper. First, we propose a method for automatic correction of camera pitch and roll for urban scene photographs, which simplifies the building segmentation, camera pose estimation for multi-view modelling and allows to reconstruct building walls from limited information. Second, we have developed a set of automated tools for building boundary specification that relies on simple user operations. Third, have extended the vertical polygons modelling methods from IBPE for the case when the contact line of the building and ground is heavily occluded or unreliable and proposed several algorithms for automating the process. Finally, we extend this modelling approach to modelling from several viewpoints.

1 INTRODUCTION

3D urban models are widely used in various applications - for example in geographic information systems (GIS) to support urban planning and analysis applications, car navigation systems to provide the 3-dimensional, photorealistic display of the surrounding area to help making intuitive orientation easier for the driver. Another popular application is 3D content authoring for the entertainment purposes or multimedia content creation.

In this paper a image-based reconstruction system is proposed that aims for achieving a comfortable balance between the model realism and user's convenience. We represent city scene as a set of buildings, walls of which are modelled as a set of vertical polygons, and a set of non-building objects (stationary cars, traffic lights, etc.) modelled as billboards. The user is provided with a set of automated tools, which allow reconstructing a scene from a single or several images (if available) taken from different viewpoints within few minutes.

1.1 Related work

A classic approach is implemented in ImageModeler (Rea, 2004) software package. First the user solves the problem of camera calibration by setting point matches and sets of parallel lines to estimate the camera matrices. The final model, represented as a textured 3D mesh, is obtained by triangulation using some specified corresponding points in different images.

This approach requires several images are too tedious to gain popularity in 3D modelling. Photo3D software (Sof, 2005) gives an opportunity to reconstruct 3D model from one image only, but the user interface is far from intuitive. Popular SketchUp (Goo, 2006) software package makes 3D modelling process easier, compared to traditional 3D content editors like 3DS Max, but needs at least basic 3D modelling skills (the user should learn how to "project back" the shape of the object in the photo onto 3D model).

In Image-based Photo Editing (IBPE) (Oh et al., 2001) a different approach is used, based on image segmentation and mostly 2D editing operation on original image and depth map. Auto Photo popup use the same approach, but employ fully automatic image segmentation method based on machine-learning. As a result, the model is obtained without any user interaction, but is too crude, compared with other approaches.

Our system is capable of reconstructing both from one and several images. It follows the image segmentation approach of IBPE and Auto Photo Popup and provides user-guided automated image segmentation tools. The resulting model is more detailed and accurate than that of Auto Photo Popup, and requires significantly less user interaction then using IBPE or ImageModeler.

2 PROPOSED ALGORITHM - OVERVIEW

Highly automated image-based reconstruction of a fully accurate model of a city scene is still in the future. To make the task tractable we use several assumptions about the urban scene. The main objects of interest for the city scenes are the buildings. Other objects are traffic lights, posts, maybe some stationary cars, kiosks, etc. (Rottensteiner and Schulze, 2003) distinguish three levels of detail for 3D city models, namely LoD1 consisting of vertical prims with flat roofs, LoD2 containing geometrically simplified building models with approximated roof shapes, and LoD3 containing buildings as complex geometric bodies, including facade details. Our main goal is to create models that will be observed from approximately the same height as the people observe them while walking in the streets (application example - creation of virtual walkthroughs and car navigation systems). In such case we do not need to model the roof shapes. So we choose LoD1, as gives enough realism when walking on the ground level, but keeps the building models relatively simple, delivering small details in texture.

Other objects of interest can be modelled as billboards (Horry et al., 1997) - flat surfaces with mapped object images. If a correct billboard orientation is specified, even relatively stretched objects, like traffic lights pendent of street, can be realistically models with single billboard. Also, in most cases, planar ground model is sufficient.

The whole reconstruction process consists of several steps. First we correct pitch and roll angles of the input images. This is done by mapping of original photo to virtual camera with zero pitch and roll. We propose a new algorithm to estimate pitch

and roll angles for virtual view generation. Then we apply semantic image segmentation from Auto Photo Popup (Hoiem et al., 2005) system to obtain initial segmentation of image into ground/objects/sky regions. Then each building modelled separately. We provide a number of automated tools for specification of side and up boundaries of building, which use only approximate mouse click or 'brush' stroke as input. We also have developed a set of algorithm for automatic or semi-automatic estimation of building walls orientation through specification of the so-called 'middle line'. From building boundary and middle line the geometry of building is reconstructed, and then model is textured using the original photos. Non-building objects can then be separately modelled with help of interactive image segmentation technique. As a last step, we apply image completion methods to regions of the model, occluded from view by other objects in original image. This significantly increase the rendering realism.

3 IMAGE PRE-PROCESSING

To capture the building fully in one image from top to bottom the camera is usually pitched up that causes each image to become 'keystoned' or 'tilted'. Keystoning causes rectangles (e.g. window frames and door frames) to become trapezoids that are wider at the top (camera pitching down) or at the bottom (camera pitching up). The vertical (with respect to the ground plane) lines like boundaries of the buildings are projected to inclined lines in images. If camera roll angle is also non-zero the horizon line of keystoned images becomes also inclined.

We propose a new tilt-correction algorithm to calculate 'virtual views' with zero pitch and roll angles from keystoned images. Such virtual views have several advantages over original images that simplifies the subsequent image segmentation and reconstruction process: side borders of the (most) buildings are projected to vertical lines in virtual views, which are easier to detect; virtual views extrinsic calibration is defined by 4 parameters only (camera viewing direction is parallel to the ground plane), which helps in both 3D modelling and pose estimation.

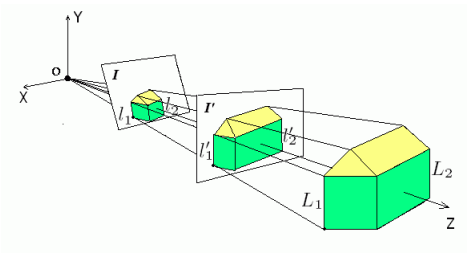
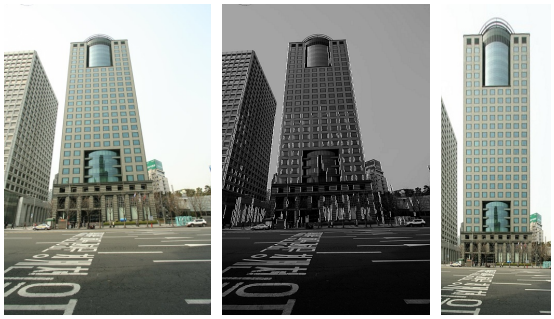


Figure 1: Camera pitch and roll correction illustration

The proposed tilt-correction algorithm is based on constraining the vertical lines (in real world) are that abundantly present in man-made environments (see figure 2(b)) to become vertical in the virtual images as shown on figure 1. Camera orientation for virtual image I' and original image I differs by pitch and roll angles. Thus image transformation between I and I' is described by rotational homography, which can be parameterized by these 2 angles only. Using the fact that 3D vertical lines L_i of the building project to vertical 2D lines l'_i on virtual view I' and to inclined 2D lines l_i on source image I. We estimate pan and tilt by formulating the objective function that penalizes the non-verticality of virtual view lines l'_i and minimizing it by the gradient descent algorithm. The algorithm outline is as follows. First, extract line segments that correspond to vertical vanishing point by the method in section 5.1.2, applied to straight line segments pointing approximately 'up' ($\pm\pi/6$ to vertical direction). Second - estimate pan and tilt angle of virtual view I' and calculate intrinsic calibration parameters for virtual view I' so that all pixels from I projects inside I'. Finally apply warping transformation to create the I' image.



(a) Source image (b) Detected lines (c) Corrected image

Figure 2: Camera pitch and roll correction

4 GEOMETRY MODELLING

Fully automatic reconstruction algorithms like based on dense stereo create whole unparsed model, which is difficult to use and refine. In such system as ImageModeler, 3D scene is incrementally generated from user-specified building blocks. Specification of each block requires tedious and precise point matching from the user. In IBPE other approach for creation of parsed 3D scene is proposed, which is based on manual image segmentation. The image segmentation is simpler for the user then point specification but also very time consuming. The total modelling time for each small scene is very large in this case. The Auto Photo Popup system is fully automatic, but 3D scene is only approximately parsed.

We use the image segmentation approach for creation a parsed 3D scene like IBPE but focus on automation of user interaction which drastically reduce modelling time.

4.1 Building segmentation

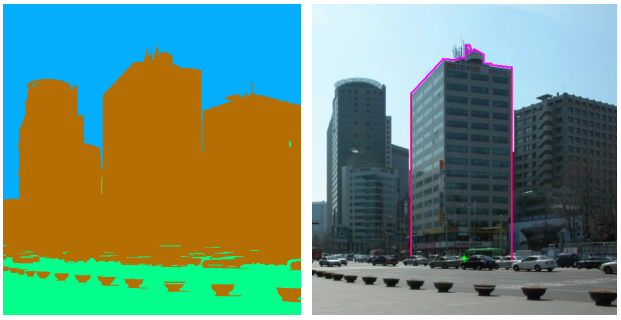
In the proposed system building boundary is specified by the side borders, the ground contact point and the upper border part. We doesn't specify the whole bottom boundary, we specify only one point on the bottom boundary of the building. Each boundary is extracted separately. To obtain initial image segmentation we apply algorithm from Auto Photo Popup paper (Hoiem et al., 2005) (we used the SW available on the Internet). The algorithm segments images into three regions - ground, buildings and sky (so-called 'GBS-map').

User starts from the estimation of the side boundaries of the modelled building. Because we use 'virtual' images with zero pitch and roll, vertical boundaries of the buildings are vertical lines in images. User clicks twice near the left and right building boundaries. Note that those clicks need not to be precise. The search range for automatic refinement is bounded by brush size. A image column that maximizes contrast between left and right neighboring regions is selected as a result. Only those columns are analyzed, where at least one pixel belongs to 'Buildings' region in the GBS map. Our experiments have shown, that this simple methods works file in 95% on our test dataset. If false boundary is found then user can correct it with single precise click.

In urban environments, especially for the tall buildings, upper boundary of the building is also the boundary of sky region. After side boundaries are specified, the boundary between 'sky' and

'building' regions is selected as initial approximation. If this is correct, we can proceed to bottom point specification. On our test database this happens in 70% of cases. If small building is modelled then its upper boundary can be inside 'buildings' region. In this case we apply 'GrowCut' (Vezhnevets and Konushin, 2005) interactive image segmentation algorithm. Usually, only a loose brush stroke along the building boundary is enough for input.

Specification of bottom point is the most difficult task. Usually, bottom part of the building is occluded by cars, trees, advertisements, etc. In our system user makes an approximate click inside ground region, where boundary between buildings and ground is closest to the truth. The highest point of ground region from GBS-map near the user click is selected as bottom point. This is accurate in 50% of test cases. In other cases we rely on precise manual specification of bottom point. However, because it is only one point per building, this operation is not very time-consuming.



(a) Results of automatic segmentation (the GBS map) (b) Specified building border and ground contact point

Figure 3: Building segmentation

5 MIDDLE LINE ESTIMATION METHODS

After the building boundary and bottom point are extracted from image we need to estimate the number and orientation of its visible walls. In (Oh et al., 2001) this was achieved by requiring user to draw the building 'bottom line',which is the contact line of the object and the ground plane. This method encounters several problems, when applied to urban images, taken with a hand-held camera. First, the building bottom line is often heavily occluded by the objects in the view (people, cars, plants, etc.) Second - in case when camera is positioned on the height of 1-2 meters from the ground (which is a normal condition for a hand-held camera or tripod) the estimation of the wall orientation and position based on the bottom line becomes very sensitive to even smallest errors in the line specification.

Notably, in urban environment an abundant number of straight lines can be found, either parallel to the ground or orthogonal to it. Instead of bottom line, we can select a 'middle line', which is a polyline, parallel (in 3D space) to the ground plane positioned at any height of the building. The middle line can be specified at arbitrary height, so estimation of building wall orientation and position from middle line is less sensitive to small errors than using bottom line. Additionally, modern buildings are usually higher than trees and cars, so that middle line can be specified on occlusion-free upper parts of building walls.

We propose several algorithms for middle line estimation. Two of them based on vanishing point (further - VP) detection. Using vanishing points in single view or multi-view 3D modelling is not new (Kosecka and Zhang, 2005), but vanishing points are mostly used for the purpose of camera calibration in both research papers, and commercial products (Goo, 2006), (Rea, 2006).

We organize the middle line estimation process as a cascade - starting from the fully automated method and reverting to semi-automatic one in case the results of the automatic detection are unsatisfactory.

5.1 Middle line estimation from vanishing points

Most existing papers describing methods for estimating vanishing points focus on finding only 3 ones that correspond to 3D orthogonal directions and ignore others, because the goal is camera calibration. In our case each group of parallel lines from each building wall should be identified. The middle line estimation is 3-step process:

1. Find all the vanishing point that correspond to all visible building walls
2. Identify building corners to specify building walls merge points
3. Construct a middle line from known building wall position in images and known VP for each wall

Note that we work with virtual views with zero pitch and roll, so only horizontal position of building corners should be identified.

(a) (b)

Figure 4: Detection of vanishing points (VP), (a) - Straight edges, (b) - Edges grouped by dominant VPs with estimated building corner and middle line

5.1.1 Automatic estimation Algorithm from (Kosecka and Zhang, 2005) was used to estimate the vanishing points automatically. The result of the algorithm is the found vanishing points and grouped straight line segments (see figure 4). This grouping allows us to estimate the building corner (shown by the yellow line in figure 4(b)) by finding the margin between two sets of the straight lines.

In practice, for our test dataset this automatic approach worked successfully for 35% of all examples. In case when the results are bad, the user can use one of our semi-automatic tools, explained further. After the VPs are known, the middle line is easily constructed.

5.1.2 Estimation from the building corners The user may correct (or specify) the columns where the building corners should be found (the yellow line in figure 4(b)). This gives a separation of the image into vertical strips - each of each is a single wall. Inside one single wall there should exist one dominant vanishing point, which relates to the wall orientation. We used a method based on RANSAC (Fischler and Bolles, 1981) ideology to estimate it.

We keep the straight line segments estimates by the automatic algorithm in its first stage and use them for semi-automatic VP estimation. Each vertical strip is analyzed separately, additionally, the lower 20% of the image is ignored, because of a lot of spurious edges exist there from other objects - cars, people, trees, etc. Then this algorithm is launched:

```
1.θ_t = π/32;
2.Repeat for i = 1,..,N iterations:
    Select arbitrary line sections l_j1, l_j2;
    Estimate point v_i as their intersection;
    Reset VP counter c_i = 0;
    For all existing line segments l_j, j = 1,..,M:
      Compute line l', passing middle point of l_j and v_i;
      If angle between l' and l_j is < θ_t then:
         increase VP counter c_i = c_i + 1;
3.Select the VP with maximum counter c_i';
  If c_i' > 40% · M then:
    v_i' is the dominant VP, go to step 4;
  Else
    Coarsen the threshold θ_t = θ_t * 2;
    If θ_t > π/4 then:
      failed to find dominant VP, exit;
    Else
      goto step 2;
4.Refine the result by re-estimating v_i' from
  all inlier lines;
```

Where M is the total number of straight line segments inside the vertical strip. The number of iterations $N = 500$ was estimated to be enough for reliable detection in our test dataset.

5.1.3 Middle line estimation from roof boundary Some buildings do not have enough vivid horizontal lines on their walls (see figure 5). For such buildings the already described methods will fail to estimate the VPs and so middle line. Nevertheless, usually the walls orientation of such buildings can be inferred by the shape of their roof boundary or some vivid single horizontal line.

To cope with such cases we integrated another tool to our system,that takes an approximate horizontal polyline as input and refines it guided by strong luminance edges in the image. The method works as follows: **Step 1:** Detect edges by Canny (Canny, 1986) algorithm; **Step 2:** For each line section of the middle line re-compute line section direction by fitting straight line to the edges within the search range of the current section position by robust least squares; **Step 3:** After line directions are adjusted, the intersection points of the line sections are recomputed.

(a) Initial position image (b) Edges used for refinement (c) The result

Figure 5: Detecting middle line by the roof boundary

During Step 2 iteratively re-weighted total least squares are used to accurately fit the straight line sections in presence of the outliers.

$$(a, b, c) = \arg \min_{a,b:a^2+b^2=1} \sum_i w_i \cdot (ax_i + by_i + c)^2 \quad (1)$$

Where (a, b, c) are the line parameters, and (x_i, y_i) are the edge points detected by Canny algorithm.The well-known Tukey bi-square function is used for calculating the points weights w_i. Only the points of edges within user-selected range are considered, check figure 5(b). As the last resort, middle line can be manually specified, but in our experiments in happens in less then 10% of cases.

5.2 3D model creation

Given the "middle line", building boundaries and a single ground contact point we can correctly place the building walls in the scene. The idea of the algorithm is illustrated in figure 6.

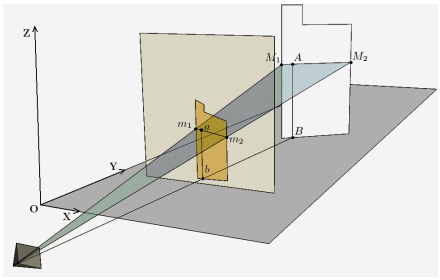


Figure 6: Wall 3D model construction

5.2.1 Geometry Each building is modelled as a set of (connected) vertical rectangular polygons. One vertical polygon is reconstructed for each straight segment of middle line. Building image is used as texture for the reconstructed model. To keep the geometry simple, the polygon dimensions are defined by the oriented bounding box of the building region. The accurate building boundaries are modelled using building's texture opacity channel (see section 5.2.2). The algorithm for building walls reconstruction is described below. We will use the coordinate system, defined as figure 6 shows. The OX and OY axes are spanning the ground plane, while OZ axis is pointing up.

We consider virtual views with zero pitch and roll angles (this means that camera view vector is parallel to the ground plane). The virtual view camera has same position as original camera. The Z coordinate of the camera center affects only the scale of the resulting model. If it is specified exactly, we get a metric reconstruction as a result, if - not s reconstruction accurate up to scale. So camera projection matrix can be defined as:

$$P = K \cdot C = K \cdot [R' | -R'T] = K \cdot \left[\begin{array}{ccc|c} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -h \end{array}\right] \quad (2)$$

Where K - is the camera intrinsic calibration matrix, h - the camera Z coordinate. In practice the K matrix can be defined as:

$$K = \begin{bmatrix} f & 0 & ImageWidth/2 \\ 0 & f & ImageHeight/2 \\ 0 & 0 & 1 \end{bmatrix} \quad (3)$$

Where f - is the focal length, measured in pixel. It can be calculated from EXIF data of JPEG file. Consider reconstructing a single planar wall (in this case middle line is a single straight

segment), as shown on figure 6. Consider reconstructing a single planar wall (in this case middle line is a single straight segment): m_1 and m_2 - end points of middle line segment, specified in image. M_1 and M_2 - end point of 3D line segment (line on the building surface). Because middle line (in 3D) is parallel to the ground plane - M_1 and M_2 share same Z coordinate, $M_1 = [X_1, Y_1, Z_M, 1]$, $M_2 = [X_2, Y_2, Z_M, 1]$. The building bottom point in image is denoted b and B is 3D point on ground plane, placed at the intersection of ray cast from the camera center through the b point. Let point A be the intersection of vertical line segment starting from B and middle line $\overline{M_1 M_2}$. Obviously, line $\overline{AB}$ lies in building wall plane. Line $\overline{AB}$ projects to line $\overline{ab}$ in image, where a - intersection of vertical line segment, starting from b, with middle line $\overline{m_1 m_2}$. This gives us a simple algorithm to calculate the 3D position of M_1 and M_2.

First, we calculate the coordinates of $B = [X_B, Y_B, 0, 1]$. The $b = [x_b, y_b, 1]$ coordinates are known. Projection $b = PB$ gives us two linear equation on X_B, Y_B, which can be easily solved. Then we calculate the height of middle line from projection of $\overline{AB}$. A can be written as $A = [X_B, Y_B, Z_M, 1]$, because it is intersection of vertical line $\overline{AB}$ and middle line $\overline{M_1 M_2}$. a - is intersection of $\overline{m_1 m_2}$ and vertical line from b, so $a = [x_b, y_a, 1]$. y_a can from standard line equation, parameterized by two points.

After Z_M is obtained, from projection equations $m_1 = P \cdot M_1$ and $m_1 = P \cdot M_2$ we can estimate $[X_1, Y_1]$ and $[X_2, Y_2]$. M_1 and M_2 define the position, orientation and width of building wall. Low boundary is naturally defined by ground plane. The height of the building wall is estimated by analyzing each pixel of building upper boundary in image. For each pixel a Z coordinate of corresponding 3D point on building wall is calculated, and maximum is selected. Of course, the method is not limited to planar walls only - curved walls like in 11 are modelled as a set of planar polygons. The ground plane is modelled with single large polygon. The size of the polygon is defined by the distance from camera position to the farthest building.

5.2.2 Texturing The texturing of building walls is performed by straightforward projective mapping of tilt-corrected images onto reconstructed geometry. The image regions outside building borders, but inside its bounding box (used to define the wall polygon) are set to be transparent. So after model texturing the correct building boundary is achieved, regardless of its complexity. The ground texture is either synthesized or extracted from source images.

In urban environments part of the building is usually occluded by other buildings, road signs, advertisement hoardings, etc To keep the scene realism it is necessary to reconstruct the texture behind the occluding object. We have used two automatic methods for texture reconstruction - (Criminisi et al., 2003) and fast approximation for reconstructing smooth surfaces (Drori et al., 2003). The texture reconstruction is done separately for each building wall and ground plane. For each of these regions image is rectified to account for perspective effects, which helps to get better results. For complex cases we used cloning brush with perspective correction.

In future we plan to exploit the regular and repetitive structure of urban textures to increase the quality of texture reconstruction.

5.3 Modelling non-buildings

We represent non-building objects as flat *billboards*, similarly to the Tour into the picture paper (Horry et al., 1997). The segmentation of the object boundaries is performed by the GrowCut interactive image segmentation method. It allows both fast and accurate construction of the object boundary, however any other suitable image segmentation method may be applied. After the object region is segmented, the reconstruction is straightforward - same method as described in the previous section is used, assuming that billboard is oriented perpendicularly to camera view vector (this equals one-segment horizontal middle line in the image).

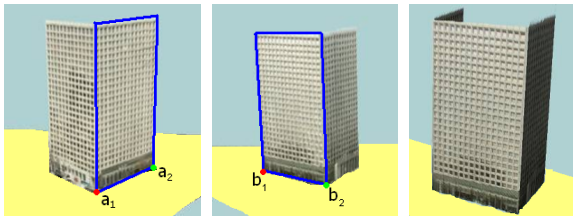
6 MULTI-VIEW MODELLING

In multi-view case two task arise - image registration and merging of single-view partial reconstruction into consistent model.

Most widely used image registration approach is based on matching point features in both images. According to our experiments, in urban environment wide-baselines matching techniques (even most powerful as renown SIFT features (Lowe, 2003)) performs poorly due to large number of very similar elements present in the scene (e.g. windows, doors, etc.). Manual point matches specification is both not robust and tedious to the user, especially compared to our easy single-view modelling algorithm.

Consider the case of two images. In each image two walls of the building are specified with middle line, one wall is common for both images. For each input image a partial 3D model of the building is reconstructed. User specifies the common wall by two clicks inside common wall region in first and second images. Then two 3D models of the same wall are available, and their position and orientation relative to image cameras are known. These 2 models can be superposed, so that relative camera position and orientation is identified. The registration pipeline is demonstrated on figures below.

As have stated in previous sections, because we use virtual views, only camera position and pan angle should be estimated. This can be done by matching two walls in terms of position on ground and scale (scaling may be necessary if the camera Z positions are unknown or inaccurate for the reconstructed views). Wall height can be unreliable due to some constructions on the roof, but wall width should be reliable enough.



(a) Model from the first image (b) Model from the second image (c) Merged result

Figure 7: Merging model from two views

This transformation can be easily represented by the following matrix:

$$M = \begin{bmatrix} s \cdot cos\alpha & sin\alpha & 0 & T_x \\ -sin\alpha & s \cdot cos\alpha & 0 & T_y \\ 0 & 0 & s & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4)$$

The matching is calculated by solving this system of equations:

$$M \cdot a_1 = b_1; \quad (5)$$

$$M \cdot a_2 = b_2; \quad (6)$$

The points a_1, a_2, b_1, b_2 are the lower points of the matched walls (color-coded in the figure 7. The system is fully constrained and

easily solved. The result is - merging two models into one (see figure 7(c)).

7 RESULTS AND DISCUSSION

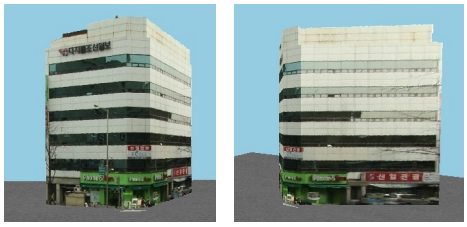
The proposed system has been tested on more than 50 images of Moscow and Seoul. Several results are shown on figures 8- 11. In 50% of test cases only a few loose mouse clicks is sufficient for model reconstruction. In other examples the complexity of modelling process is significantly lower than that of existing systems. The experiments show that proposed approach is promising.

(a) Novel view 1 (b) Novel view 2

Figure 8: Model built from 3 photos

(a) Novel view 1 (b) Novel view 2

Figure 9: A street part built from 3 photos



(a) Novel view 1 (b) Novel view 2

Figure 10: A building reconstructed from 1 image

8 CONCLUSIONS AND FUTURE WORK

In this paper we have described a novel interactive modelling technique, which allows the non-expert user to create plausible models of urban scenes with 2D image operations only. This technique may be treated as an enhancement of the famous Image-based Photo-Editing approach towards more automation and ease to use.

We planning to improve the system in several directions. First, semantic image segmentation module will identify new type of region - 'unknown'. This will increase the robustness and precision of building boundary estimation. Second, missing walls of the building will be automatically synthesized, based on visible walls. Such reconstruction may not be fully geometrically accurate, but the visual impression will be better. Third, road topological information can be extracted from images and used to increase the accuracy of building position estimation.

(a) Source image (b) Novel view 2

(c) Novel view 3 (d) Novel view 4

Figure 11: A scene reconstructed from 1 image

REFERENCES

Canny, J., 1986. A computational approach to edge detection. IEEE Trans. Pattern Anal. Mach. Intell. 8(6), pp. 679–698.

Criminisi, A., Perez, P. and Toyama, K., 2003. Object removal by exemplar-based inpainting. cvpr 02, pp. 721.

Drori, I., Cohen-Or, D. and Yeshurun, H., 2003. Fragment-based image completion. In: SIGGRAPH '03: ACM SIGGRAPH 2003 Papers, pp. 303–312.

Fischler, M. A. and Bolles, R. C., 1981. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Commun. ACM 24(6), pp. 381–395.

Goo, 2006. Google SketchUp 6. http://www.sketchup.com/.

Hoiem, D., Efros, A. A. and Hebert, M., 2005. Automatic photo pop-up. In: SIGGRAPH '05: ACM SIGGRAPH 2005 Papers, pp. 577–584.

Horry, Y., Anjyo, K.-I. and Arai, K., 1997. Tour into the picture: using a spidery mesh interface to make animation from a single image. In: SIGGRAPH '97: Proceedings of the 24th annual conference on Computer graphics and interactive techniques, pp. 225–232.

Kosecka, J. and Zhang, W., 2005. Extraction, matching, and pose recovery based on dominant rectangular structures. Comput. Vis. Image Underst. 100(3), pp. 274–293.

Lowe, D., 2003. Distinctive image features from scale-invariant keypoints. In: International Journal of Computer Vision, Vol. 20, pp. 91–110.

Oh, B. M., Chen, M., Dorsey, J. and Durand, F., 2001. Image-based modeling and photo editing. In: SIGGRAPH '01: Proceedings of the 28th annual conference on Computer graphics and interactive techniques, pp. 433–442.

Rea, 2004. ImageModeler. http://www.realviz.com.

Rea, 2006. VTour. http://www.realviz.com.

Rottensteiner, F. and Schulze, M., 2003. Performance evaluation of a system for semi-automatic building extraction using adaptable primitives. In: Proceedings of the ISPRS Workshop 'Photogrammetric Image analysis'.

Sof, 2005. Photo3D. http://www.photo3d.com/eindex.html.

Vezhnevets, V. and Konushin, V., 2005. "growcut"-interactive multi-label n-d image segmentation by cellular automata. In: Proceedings of the Graphicon 2005 conference, pp. 225–232.

3D LEAST-SQUARES-BASED SURFACE RECONSTRUCTION

Duc Ton, Helmut Mayer

Institute of Photogrammetry and Cartography, Bundeswehr University Munich, Germany
Duc.Ton|Helmut.Mayer@unibw.de

KEY WORDS: Surface Reconstruction, 3D, Least-Squares Matching, Robust Estimation

ABSTRACT:

This paper addresses extensions of the classical least-squares matching approaches of (Wrobel, 1987, Ebner and Heipke, 1988) particularly in the direction of full three-dimensional (3D) reconstruction. We use as unknowns the movement in the direction of the normals for a triangulation of the surface. To regularize the ill-posed inverse reconstruction problem, we smooth the surface by enforcing a low curvature in terms of that the vertices of the triangulation are close to the average plane of their direct neighbors. We employ a hierarchy of resolutions for the triangulation linked to adequate levels of image pyramids, to expand the range of convergence, and robust estimation, to deal with occlusions and non-Lambertian reflection. First results using highly precise and reliable, but sparse points from the automatic orientation of images sequences as input for the triangulation show the potential of the approach.

1 INTRODUCTION

The goal of this paper is to generate a dense three-dimensional (3D) model from given orientations of cameras and reliable but sparse points obtained by an automatic orientation procedure.

In a recent survey for two images (Scharstein and Szeliski, 2002) the four steps (1) Matching cost computation (2) Cost (support) aggregation (3) Disparity computation / optimization and (4) Disparity refinement are named for a typical stereo algorithm. The test described in (Scharstein and Szeliski, 2002) has sparked a large interest into stereo matching. Here we report shortly only about approaches that deal with more than two images.

Our work is inspired by (Fua and Leclerc, 1996) which also employ 3D triangular facets for the surface. Opposed to them, we only focus on stereo, we optimize the vertices of the 3D facets along their normals, and we employ robust least-squares optimization to deal with occlusions.

In recent work on 3D reconstruction such as (Lhuillier and Quan, 2005) or (Strecha et al., 2004) points from the image orientation are used as starting point for dense surface reconstruction. In the former case a bounded regularization approach is employed for surface evolution by level-set methods. The approach is different from ours as it is not focusing on wide-baseline scenarios and it therefore can use a very dense set of points stemming from the orientation. Wide baselines are the scope of the latter approach. As we they use the 3D points as starting points, but they formulate the 3D reconstruction problem in terms of an Bayesian approach and use the EM-algorithm to solve it.

A computationally very efficient approach is presented in (Hirschmüller, 2006). It employs a semi-global matching in the form of dynamic programming in 16 directions. This together with a mutual information based computation of the matching costs results into well regularized results and still a high performance allowing to work with very large images.

Opposed to the above approaches we decided to extend the classical least-squares matching approaches of (Wrobel, 1987, Ebner and Heipke, 1988) in the direction of full 3D reconstruction from wide-baseline image sequences in a similar way as (Schlüter, 1998). We move the vertices of a triangulation resulting from a densification of a triangulation obtained from our initial reliable but sparse 3D points in the direction of their normals. To deal with occlusions, we employ robust estimation. Regularization is based on additional observations modeling the local curvature of the surface.

According to the above four steps of (Scharstein and Szeliski, 2002) we do matching cost computation by squaring brightness differences between transformed values for individual images and an average image. The latter can be considered as an orthophoto of the surface. The costs are aggregated over the whole surface consisting of planar triangles and the computation of disparities or in our case of 3D coordinates is done together with the refinement in the least-squares estimation.

The potential of the least-squares approach lies in its high possible accuracy. Yet, least-squares matching is known to converge to local minima and thus good approximations are necessary. They are obtained here by using as basis sparse but highly precise and reliable points. They stem from a multi-image matching and robust bundle adjustment approach suitable for large baselines (Mayer, 2005) extended by the five point algorithm of (Nistér, 2004). The radius of convergence is additionally expanded by a coarse-to-fine optimization for different levels of resolution for the triangles.

The paper is organized as follows. After sketching basic ideas and giving an overview of the algorithm we detail the ideas in the following sections. Finally we give results and end up with conclusions.

2 BASIC IDEAS AND OVERVIEW OF ALGORITHM

The problem of surface reconstruction is formulated in terms of least-squares adjustment. To be able to work in full 3D, we triangulate the surface and move the vertices of the triangulation along a path independent from the definition of the coordinate system, namely the direction of the normal at the vertex of the triangulated surface. The direction of the normal in the vertex N_u for the unknown number u is estimated as the average of the normal vectors of the planes attached to the vertex. E.g., for Figure 1 this means $N_u = \frac{N_1+N_2+N_3+N_4+N_5}{5}$. It is normalized by $N_u^{norm} = \frac{N_u}{\|N_u\|}$.

The basic ideas of our approach can be summarized as follows:

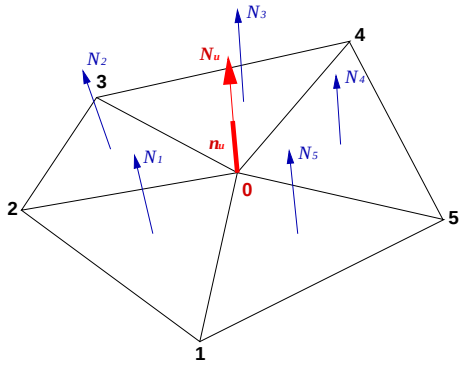


Figure 1: Relation of normal N_u at (center) vertex to normals of neighboring planes and unknown size of movement n_u

- It is based on a triangulated 3D surface.
- The vertices of the triangles move along their respective normals. The sizes of movement are the unknowns n_u (cf. Figure 1).
- Points inside the 3D triangles are projected into the images resulting into the observations.
- The goal of moving the vertices along the normal vectors is to obtain as small a squared gray value difference as possible between the back-projected points in the images and their average value supposed to be the reflectance value of the surface in a least-squares sense.
- Additionally to the image observations representing the data term the surface is regularized by observations enforcing its local smoothness in terms of curvature.
- To deal with outliers, e.g., in the form of local occlusions, robust estimation is used.

The algorithm consists of:

- Creation of triangulated surface from the given sparse 3D points
- Densification of the triangulation at different resolution levels by splitting the triangles of the surface. This results in the unknowns for whom the initial values are interpolated from the neighboring given 3D points.
- By splitting of the triangles of the unknowns and projection of the resulting points into the images the image observations are obtained. The analysis of a local neighborhood of the unknowns leads to additional smoothness observations.
- Robust least-squares adjustment to estimate improved values for the unknowns at the different levels of resolution

Before describing the steps of the algorithm, we detail the contents of the design matrix of the least-squares estimation problem which will be constructed in the course of the algorithm.

3 PARTIAL DERIVATIVES FOR THE DESIGN MATRIX

The image observations are devised to describe how well the intensities in all images showing a 3D point fit to an average intensity computed from all these images by taking the difference between the individual values and the average. Unfortunately, the lighting might be different for the images, the camera might have used a different gain, or the surfaces have a non-Lambertian bidirectional reflection distribution function (BRDF). Therefore, to reduce the bias of the estimation, the overall brightness of the images is estimated at the beginning from a small neighborhood of all given sparse, but reliable 3D points seen in an image.

For the given non-linear problem the design matrix consists of the partial derivatives of the intensity value I_i of observation i in an image according to the change of the size of movement n_u of unknown u. They are given by

$$\begin{aligned}\frac{\partial I_i}{\partial n_u} &= \frac{\partial I}{\partial x}\frac{\partial x}{\partial n_u} + \frac{\partial I}{\partial y}\frac{\partial y}{\partial n_u} \\ &= \frac{\partial I}{\partial x}\left(\frac{\partial x}{\partial X}\frac{\partial X}{\partial n_u} + \frac{\partial x}{\partial Y}\frac{\partial Y}{\partial n_u} + \frac{\partial x}{\partial Z}\frac{\partial Z}{\partial n_u}\right) \\ &\quad + \frac{\partial I}{\partial y}\left(\frac{\partial y}{\partial X}\frac{\partial X}{\partial n_u} + \frac{\partial y}{\partial Y}\frac{\partial Y}{\partial n_u} + \frac{\partial y}{\partial Z}\frac{\partial Z}{\partial n_u}\right)\end{aligned}$$

with

- $\frac{\partial I}{\partial x}, \frac{\partial I}{\partial y}$ the image gradients in x and y direction which can, e.g., be estimated by the Sobel operator,
- $\frac{\partial x|\partial y}{\partial X|\partial Y|\partial Z}$ describing how the position in x- or y-direction in the image is affected by changing the 3D point coordinates X, Y, or Z corresponding to observation i, and
- $\frac{\partial X|\partial Y|\partial Z}{\partial n_u}$ the derivative of the 3D point coordinates X, Y, or Z according to the size of the unknown movement. The points move in the direction of N_u. The size of their movement depends on the distance of the 3D point from the line connecting the other two vertices of the triangle the point is lying in.

We model the projection of (homogeneous) 3D points $\mathbf{X}$ to image points $\mathbf{x}$ by (Hartley and Zisserman, 2003)

$$\mathbf{x}' = \mathbf{P}\mathbf{X} \tag{1}$$

with the projection matrix $\mathbf{P}$

$$\mathbf{P} = \mathbf{K}\boldsymbol{R}(\boldsymbol{I}| - \boldsymbol{X}_0)$$

describing the collinearity equation consisting of the calibration matrix $\mathbf{K}$ comprising principal point, principal distance, scale difference and sheer as well as translation described by the Euclidean vector $\boldsymbol{X}_0$ and rotation by the matrix $\boldsymbol{R}$.

We additionally employ quadratic and quartic terms to model the radial distortion to obtain an accuracy in the range of one fifth to one tenth of a pixel, but we will not include this issue in the further discussion, to make the paper more readable.

For improved flexibility we work in a relative coordinate system which can be obtained from images alone. The first camera position is used as the origin of the coordinate system. The rotation of the first camera is fixed and is supposed to point to the negative z-direction. The distance of the first and the second camera is set to one.

4 TRIANGULATION OF GIVEN SPARSE 3D POINTS

We assume that the given sparse 3D points stemming from a highly accurate bundle adjustment using possibly many images are precise and reliable. We thus fix their 3D position.

One basic problem for a full 3D approach is the linking of triangles. It is at least difficult, often even impossible to link points in 3D just based on proximity. E.g., consider a thin surface, where points on both sides of the surface should not be linked, but might be much closer than points on the same side of the surface.

To avoid the above problem, we split the images into overlapping triplets. For them we assume that the topology of the 3D points can in essence be modeled in two dimensions (2D) in the images. We therefore can triangulate the points for the triplets in one of the images. To obtain compact triangles, we employ Delaunay triangulation. This reduces problems with elongated thin triangles.

First, we project via equation (1) and the known camera parameters the given 3D points into the central image of the triplet. There they are triangulated. After this, triangulations for different triplets can be stitched together which leads to full 3D triangulations. Yet, for us this is subject of further work. All following steps can now work on this basic global triangulation in as many images as available. The given 3D points are shown as (black) numbers in Figure 2.

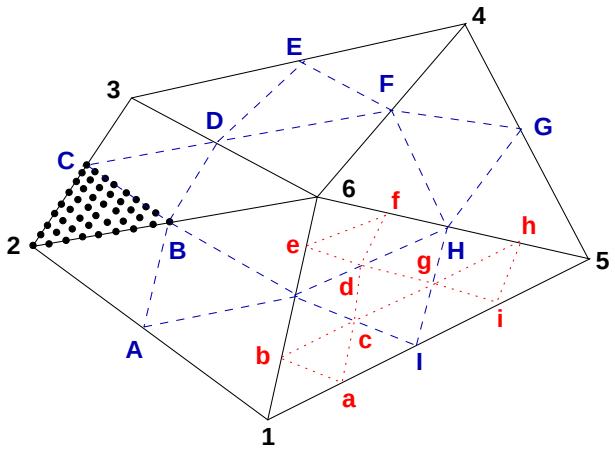


Figure 2: Creation of the vertices of the triangulation corresponding to unknowns and observations – The given 3D points are considered as control points and are marked as (black) numbers. The first level of unknowns are denoted by (blue) capital letters and the second level, which is detailed only for one original triangle, by (red) small letters. Observations are sketched for triangle 2CB (left) as black dots. They are denser because the threshold employed is $\frac{1}{10}$ of the side length for the unknowns.

5 CREATION OF UNKNOWNS AND COARSE-TO-FINE STRATEGY

The vertices of the triangulation corresponding to the unknowns are generated by splitting the sides of the triangles obtained from the given 3D points (cf. preceding Section) at their center if the length of the side is beyond a given threshold. The new unknowns receive their 3D position by linear interpolation. This leads to the first level of unknowns marked by blue capital letters in Figure 2. If the length of the sides should still be above the given threshold, i.e., the triangle is rather big as no 3D point could be found inside it, we split the triangle again along the sides and obtain a second level of unknowns marked by red small letters in Figure 2. This is done recursively until all side lengths are below the threshold.

As a well-known feature of least-squares matching is its rather restricted radius of convergence, we employ a coarse-to-fine strategy. It comprises

- different levels of densification of the triangles by setting the thresholds for the lengths of the sides of the triangles differently and
- use of image resolutions adapted to the sizes of the triangles by selecting a corresponding level of an image pyramid.

6 DETERMINATION OF IMAGE OBSERVATIONS

The coordinates of the 3D points corresponding to the observations are generated similarly as above for the unknowns except that a smaller threshold, namely 10% of the threshold of the unknowns is used. The resulting 3D points are sketched as black dots on the left hand side of Figure 2. The 3D points are projected into all images they can be seen from. The intensity value I_i of observation i at the projected homogeneous image point $\mathbf{x} = \alpha(xy1)^\mathsf{T}$ in an image is given by $I_i = g(x, y)$, with g the bilinear interpolation function.

For the design matrix A an unknown is affected only by the observations belonging to neighboring triangles. This leads to a sparse design matrix. We employ this by only computing those parts belonging to the actual observations, i.e., which are non-zero.

Yet, it also means that only unknowns in the normal equations are correlated which have common triangles. To obtain a banded normal equation matrix, for which efficient solutions are available, with a band-width as small as possible, we traverse the triangles along the shorter side of the given area for 3D reconstruction. For the example in Figure 3 this leads to a banded normal equation matrix sketched in Figure 4. One can regard the first unknowns to belong to the triangles marked in red in the lower left corner of the triangulation in Figure 3, the next unknowns to the triangles marked in green right of it, the next the blue, etc. All vertices of the triangles, i.e., the unknowns, are linked only to two layers of triangles which leads to a normal equation matrix with just one band parallel to the main diagonal. The width of the band depends on the length of the layer. Thus, it is useful to traverse the triangulation along the shorter side.

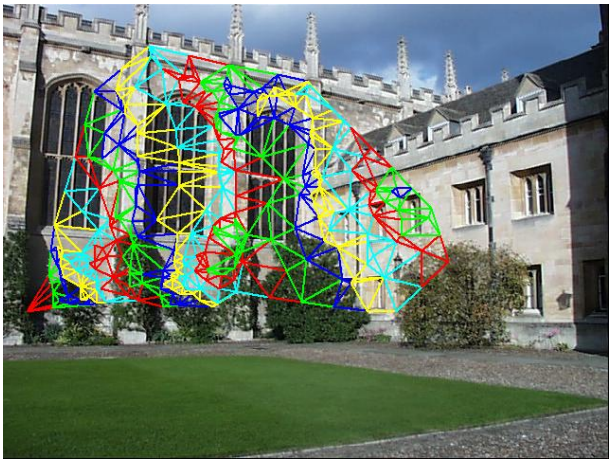


Figure 3: Traversal of triangles along the shorter side. The different colors correspond to different layers of the traversal. The traversal starts in the lower left corner (red triangles) – image Trinity from web-page Criminisi and Torr

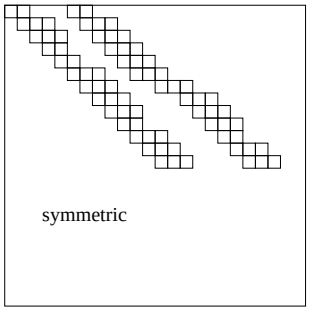


Figure 4: Banded structure of normal equation matrix resulting from traversal of triangulation along the shorter side in Figure 3.

7 REGULARIZATION BY SMOOTHNESS OBSERVATIONS

Due to noise and occlusions 3D surface reconstruction is an ill-posed problems which has to be regularized. One way to accomplish this is via additional observations enforcing the smoothness of the surface. Their influence is controlled via the ratio between the weights for the image and the smoothness observations.

We describe smoothness in terms of the deviation $h_{change}$of a vertex from an average plane derived from the neighboring vertices in the direction of its normal N.

The average plane is computed as weighted average of the heights of the vertices h_i above the plane through the given vertex and perpendicular to the normal N at the given vertex. For this the vertices are projected along the normal N, resulting in the primed (blue) numbers in Figure 5. The weighting is done according to the inverse distance d_i of the points. The average height of the vertices is at the same time the height of the given vertex at height 0 above or below this plane:

$$h_{smooth} = \sum_{i=1}^{n} \frac{h_i}{d_i} / \sum_{i=1}^{n} \frac{1}{d_i}$$

h_{smooth} is combined with the average inverse distance to an observation describing the curvature at the vertex:

$$l_{smooth} = h_{smooth} * \frac{\sum_{i=1}^{n} \frac{1}{d_i}}{n}$$

8 ROBUST LEAST-SQUARES ADJUSTMENT

To solve the least-squares adjustment for the unknowns x, we must factorize the normal equation matrix $A^{\mathsf{T}}PA$, with the design matrix A and the weight matrix P. As there might be thousands or even tens of thousands of unknowns, the factorization of the matrix requires special attention. We basically employ that as detailed above in Section 6 we obtain a (symmetric positive definite) band matrix. We then use a Cholesky factorization for banded symmetric matrices and solve for x.

To stabilize the solution, we employ the Levenberg-Marquardt algorithm. I.e., we multiply the elements on the main diagonal with a factor ranging from 1.0001 to 1.1 and take the result with the smallest average standard deviation σ_0. The non-linear optimization is done a couple of times until the ratio of the σ_0 between two iterations falls below 1.01.

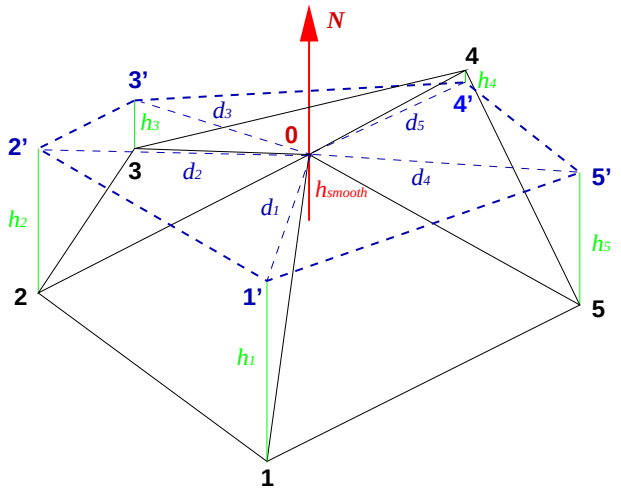


Figure 5: Smoothing – The (black) numbers denote the original vertices. The (blue) primed numbers show their projection on the plane perpendicular to the normal N through the given point 0 with height h_i. h_{smooth} corresponds to the height of the given vertex above or below the (weighted) average plane.

As weight matrix P we use a diagonal matrix. It is normalized to unity before being multiplied with the design matrix or the vector of the observations l. Initially all weights are set to one besides a scaling factor weighing image and smoothness observations against each other as explained in Section 7.

Because there might be bad or wrong matches due to occlusions or non-Lambertian behavior of the surface, robust estimation is used. We particularly base robust estimation on standardized residuals $\bar{v}_i = v_i/\sigma_{v_i}$ involving the standard deviations σ_{v_i} of the residuals, i.e., the differences between observed and predicted values. As the computation for the individual observation is computationally costly, we substitute it by an estimate of the average standard deviation of the gray value, particularly 3 gray values. We then do reweighting of the elements of P with $w_i = 1/\sqrt{2 + \bar{v}_i{}^2}$ (McGlone et al., 2004).

9 RESULTS

In this section we report about initial results. In all cases we compare the initial triangulation on the first level consisting of the reliable but sparse points from the orientation with the final densified result to show the improvement obtained by our approach. The output is done in VRML – virtual reality modeling language format.

Figures 6 and 8 give results for two scenes derived from image triplets from the web-page of Antonio Criminisi and Phil Torr while in Figure 7 we present a 3D reconstruction for an image triplet showing a part of the Zwinger in Dresden.

The final results in Figures 6 and 7 demonstrate that the densification of the triangulation leads to a better reconstruction of the details of the scene. This can be even better appreciated in Figures 8 and particularly 9, where one can see that, e.g., for the front part of the baguette or the left rim of the red basket on the right hand side, the densified triangulation is accompanied with improved normals.

10 CONCLUSIONS

We have shown an extension of the classical least-squares matching approaches of (Wrobel, 1987, Ebner and Heipke, 1988) which

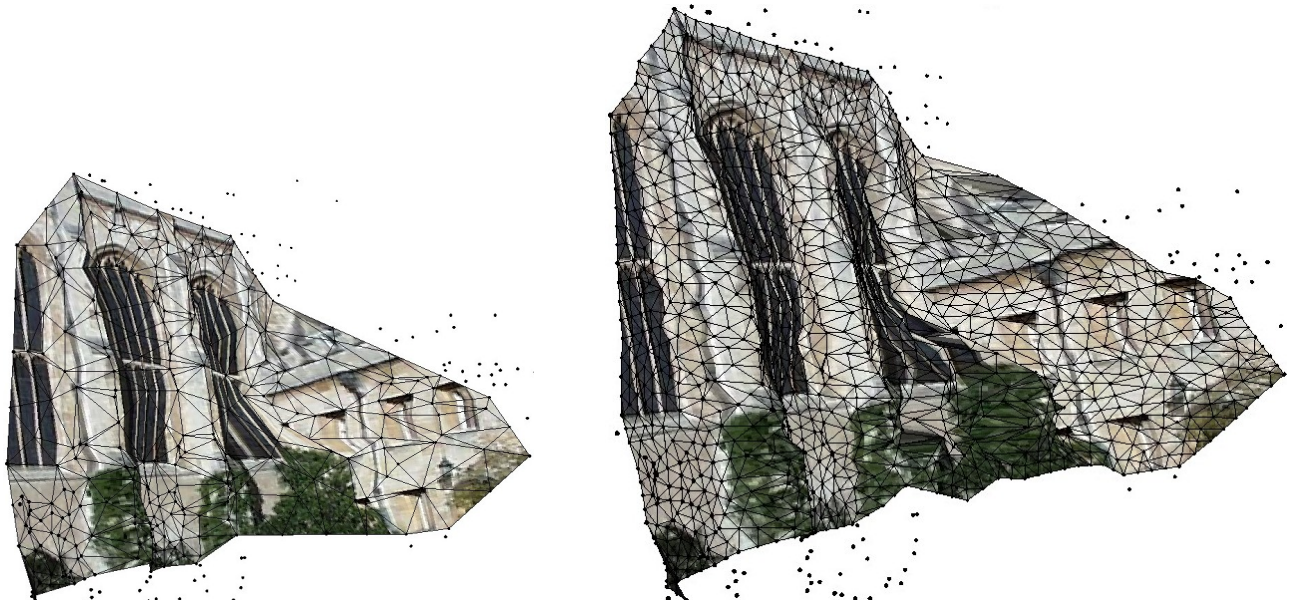


Figure 6: Result for triplet Trinity (from web-page Criminisi and Torr) at first level of resolution (left) and on third level after optimization (right). Please particularly look at the drainpipe on the right facade.

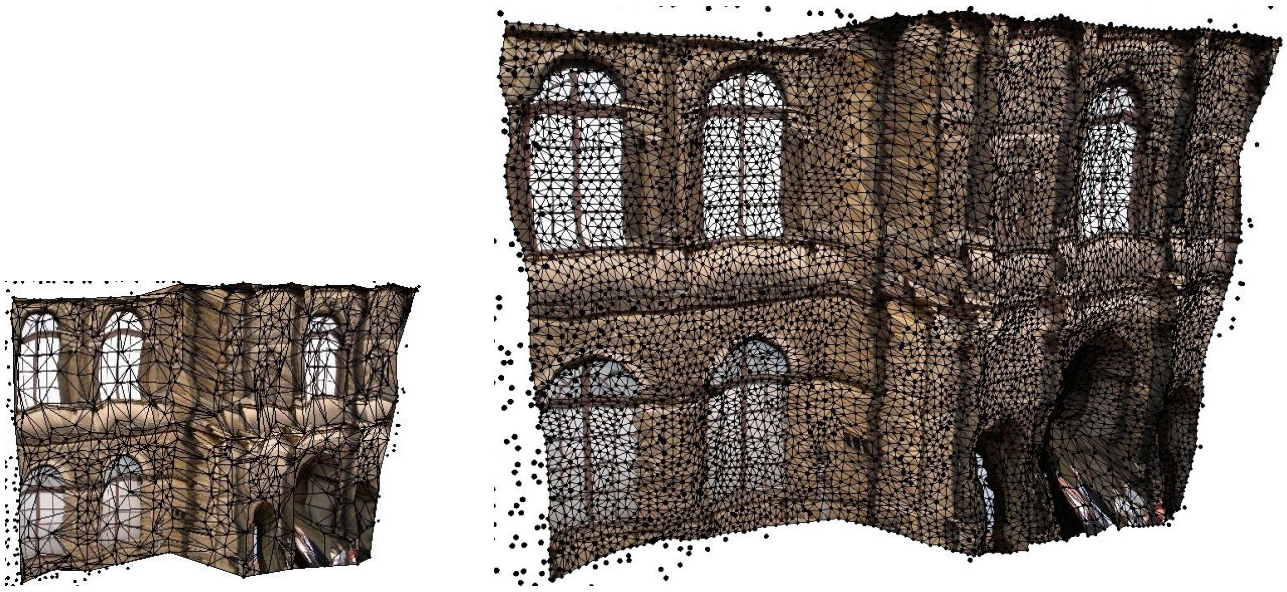


Figure 7: Result for triplet Zwinger at first level of resolution (left) and on third level after optimization (right)

were confined to 2.5D surfaces to 3D by employing the normals of a triangulation similarly as (Schlüter, 1998). Opposed to the latter, our approach is focusing on wide-baseline settings and we employ robust estimation to deal with occlusions.

First results show the potential but also the shortcomings of the approach. We still need to extend it by linking the triangulation of image triplets into triangulations for larger number of images and many parts of the algorithm need to be refined. We also consider to move the vertices towards edges in the image, as the latter tend to give hints on break-lines of the surface, though we note that this problem is mitigated as the initial points are at corners by definition. Finally, we want to check the fit in terms of least-squares error of each triangle before subdividing them to avoid small triangles in homogeneous areas.

Very recently, (Pons et al., 2007) have presented an approach with similarities to ours, though they link surface reconstruction with scene flow estimation over time. They employ graphics hardware to speed up processing. This idea could also help to speed up our algorithm as the determination of the observations entails large numbers of projections from 3D space into the images which could very well be solved by graphics hardware.

ACKNOWLEDGMENTS

We want to thank the republic of Vietnam for supporting Duc Ton by a PhD scholarship

REFERENCES

Ebner, H. and Heipke, C., 1988. Integration of Digital Image Matching and Object Surface Reconstruction. In: International Archives of Photogrammetry and Remote Sensing, Vol. (27) B11/III, pp. 534–545.

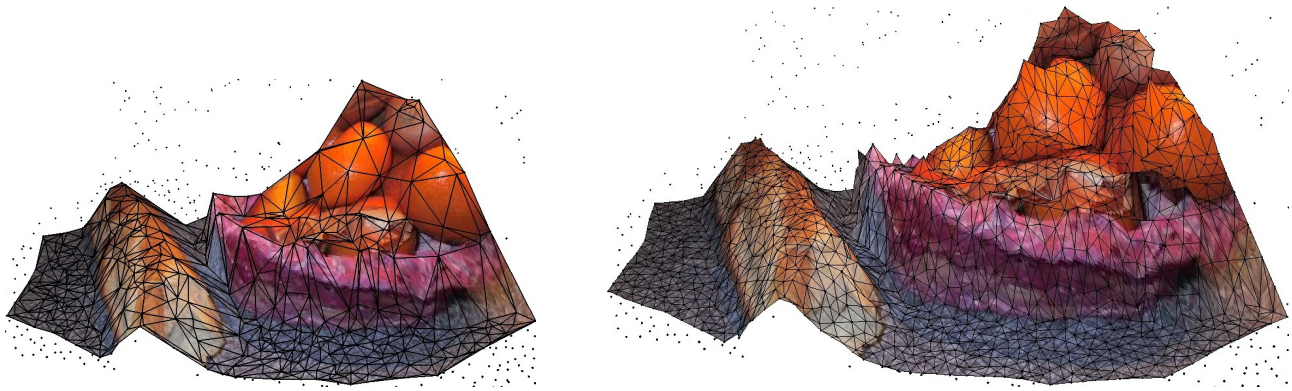


Figure 8: Result for triplet kitchen (from web-page Criminisi and Torr) at first level of resolution (left) and on third level after optimization (right)

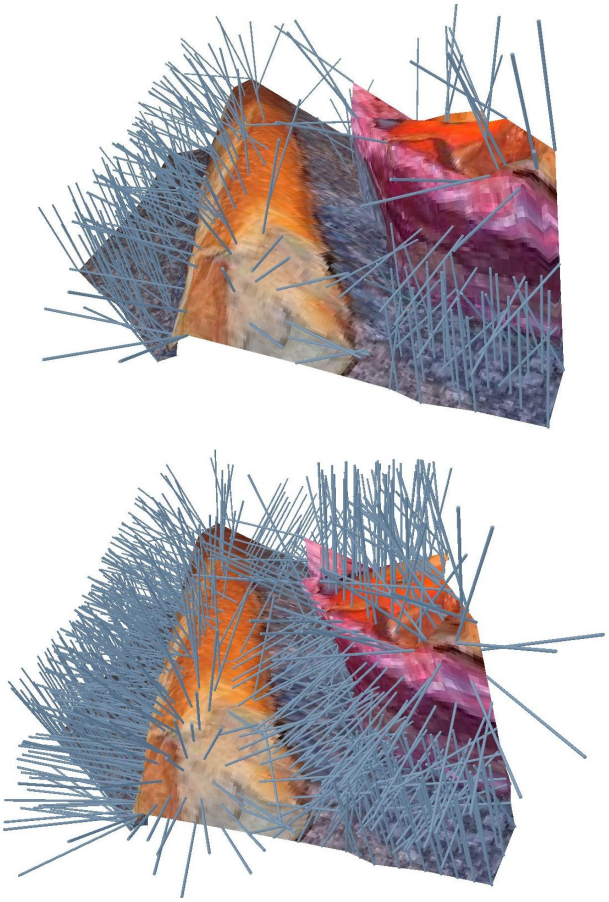


Figure 9: Detail of result for triplet kitchen (cf. Fig. 8) with normal vectors

Fua, P. and Leclerc, Y., 1996. Taking Advantage of Image-Based and Geometry-Based Constraints to Recover 3–D Surfaces. Computer Vision and Image Understanding 64(1), pp. 111–127.

Hartley, R. and Zisserman, A., 2003. Multiple View Geometry in Computer Vision – Second Edition. Cambridge University Press, Cambridge, UK.

Hirschmüller, H., 2006. Stereo Vision in Structued Environments by Consistent Semi-Global Matching. In: Computer Vision and Pattern Recognition, Vol. 2, pp. 2386–2393.

Lhuillier, M. and Quan, L., 2005. A Qasi-Dense Approach to Surface Reconstruction from Uncalibrated Images. IEEE Transactions on Pattern Analysis and Machine Intelligence 27(3), pp. 418–433.

Mayer, H., 2005. Robust Least-Squares Adjustment Based Orientation and Auto-Calibration of Wide-Baseline Image Sequences. In: ISPRS Workshop in conjunction with ICCV 2005 "Towards Benchmarking Automated Calibration, Orientation and Surface Reconstruction from Images" (BenCos), Beijing, China, pp. 1–6.

McGlone, J., Bethel, J. and Mikhail, E. (eds), 2004. Manual of Photogrammetry. American Society of Photogrammetry and Remote Sensing, Bethesda, USA.

Nistér, D., 2004. An Efficient Solution to the Five-Point Relative Pose Problem. IEEE Transactions on Pattern Analysis and Machine Intelligence 26(6), pp. 756–770.

Pons, J.-P., Keriven, R. and Faugeras, O., 2007. Multi-View Reconstruction and Scene Flow Estimation with a Global Image-Based Matching Score. International Journal of Computer Vision 72(2), pp. 179–193.

Scharstein, D. and Szeliski, R., 2002. A Taxonomy and Evaluation of Dense Two-Frame Stereo Correspondence Algorithms. International Journal of Computer Vision 47(1), pp. 7–42.

Schlüter, M., 1998. Multi-Image Matching in Object Space on the Basis of a General 3-D Surface Model Instead of Common 2.5-D Surface Models and its Application for Urban Scenes. In: International Archives of Photogrammetry and Remote Sensing, Vol. (32) 4/1, pp. 545–552.

Strecha, C., Fransen, R. and Van Gool, L., 2004. Wide-Baseline Stereo from Multiple Views: A Probabilistic Account. In: Computer Vision and Pattern Recognition, pp. 552–559.

Wrobel, B., 1987. Digital Image Matching by Facets Using Object Space Models. In: Symposium on Optical and Optoelectronic Applied Science and Engineering – Advances in Image Processing, Vol. 804, SPIE, pp. 325–333.

INFORMATION MINING FOR DISASTER MANAGEMENT

Ch. Lucas*, St. Werder, H.-P. Bähr

Institute of Photogrammetrie and Remote Sensing (IPF), Universität Karlsruhe (TH), Englerstr. 7, 76128 Karlsruhe, Germany - (christian.lucas, stefan.werder, hans-peter.baehr)@ipf.uni-karlsruhe.de

Commission IV, WG IV/3

KEY WORDS: Disaster Management, Data Modelling, Automation, Knowledge Base, Contextual

ABSTRACT:

Within the domain of disaster management, visualisation of geo-related information in situation maps is elementary. The data acquisition for these maps is based on hundreds of written messages containing up-to-date information from damage sites. The approach and experience of a human operator used for processing these messages has to be put into practice for automation. In this paper an existing data model of the military domain is analysed. With regard to the requirements of disaster management the data model has been adapted to create an ontology supported knowledge base. For a comprehensive interpretation of the message contents, context helps to grasp the whole situation.

1. INTRODUCTION

During the management of disasters and the prevention of further hazards by *emergency operation centres (EOC)*, up-to-date information is essential for decision making. This information is based on hundreds of written situation reports, given by several on-site units and passer-by. The structure of an EOC depends on the type and extent of the diverse disaster situations. However, they all apply the same principles for information sharing. This knowledge sharing is made possible by the situation map, a representation of the current and global state of damage events based on all messages. The variety of incoming reports has to be analysed by a single operator of the management staff. This data mining process includes an assessment of the content, which is done with the help of semantic considerations, heuristic assumptions and by using context knowledge.

Aim of the presented work is to automate this process in order to assist the human operator. This paper focuses on modelling of information and processing of context knowledge based on textual reports in German language. Discussions of possible visualisation models as well as an overview about vagueness within the reports are given by Werder et. al (2006).

1.1 Conception of the SOKRATES Prototype

SOKRATES has been developed for integration in military command and control (C2) systems by the Forschungsgesellschaft für Angewandte Naturwissenschaften (FGAN). The prototype displays troop movements in a tactical map based on interpretation of military reports. These reports largely consist of free form text which has to be processed in order to produce an up-to-date map of the on-site situation. There are some striking similarities between the objectives of SOKRATES and the disaster management application (cf. Werder et. al, 2006).

The SOKRATES workflow is essentially based on several processing steps. The first step is pre-processing, which is initialized by sentence recognition. Subsequently the relevant information from the messages is transformed into a formal structure by the information extraction component, discussed in more detail in chapter 1.2. In the semantic augmentation component the structured information is enriched with the ontology supported knowledge base. In particular supplementary information like spatial references or potential dangers is added. Within post-processing the information is finally represented in the situation map. The architecture of SOKRATES is described in detail by Schade (2004).

For the integration of SOKRATES components for application in disaster management, many adaptations have to be carried out. This arises from divergent tactical symbols, domain specific jargon, differing databases and stricter regulations of military reports.

1.2 Information Extraction Components

In general, *information extraction (IE)* can be seen as a kind of data retrieval from a domain specific source. Especially textual representations, like messages in the disaster management domain, serve as input source for IE. Systems that perform IE have to be able to “find and link relevant information while ignoring extraneous and irrelevant information” (Cowie and Lehnert, 1996). What relevant information means has to be defined before processing, by extracting rules and creating a domain specific lexicon. Thereby it is necessary to define the rules as detailed as possible, in order to provide an accurate and result-oriented extraction by a minimal syntax analysis (cf. Cowie and Lehnert, 1996).

The information extraction component of SOKRATES is based on the *Saarbrücker Message Extraction System (SMES)*. The SMES has been developed by the German Research Centre for Artificial Intelligence (DFKI) and is especially designed for the requirements of German language (Neumann et. al, 1997). This

* Corresponding author

tool will also be adapted for application in the disaster management domain.

For extracting relevant information, the SMES has to be enlarged and modified regarding the lexicon and the transducers. The responsibility of this lexicon is to denominate possible unit types, events and locations. In order to be able to classify locations, all street names, towns and points of interest have to be integrated into the lexicon. Furthermore the important terms for describing damage states have to be added, like the states of a fire. The finite-state transducers, which are used by the SMES, represent a framework for syntactic analysis of language. Transducers assign a domain specific function to each term. They extract information into *typed feature structures*, which is a standard formalism in computational linguistics (cf. Pollard and Sag, 1994). The important role of the typed feature structures is to provide information for padding the ontology. The architecture of the IE within SOKRATES as well as application scenarios are described in detail by Hecking (2004).

2. DISASTER MANAGEMENT ONTOLOGY

The extracted information from each message adds additional knowledge to the situation picture. In order to represent this knowledge formally an ontology is used. According to Gruber (1993) the term ontology is defined as "an explicit specification of a conceptualization". Because it is impossible to specify knowledge completely, an ontology is always restricted to a set of objects that it is able to represent, the so-called *universe of discourse*. These objects are defined in an ontology by classes and the relationships between them. A human-readable textual description of both, along with rules that constrain interpretation and usage of objects, finally add meaning to the ontology.

In the following the ontology of the military domain along with the adaptation of this ontology for the disaster management domain are presented in more detail.

2.1 The Command and Control Information Exchange Data Model (C2IEDM)

Sharing information is an important aspect of multinational, combined and joint military operations. In modern armies Command and Control Information Systems (C2IS) are used to manage own forces and to obtain situational awareness. The Multilateral Interoperability Programme (MIP), an association of 24 nations and several organisations, developed the *Command and Control Information Exchange Data Model (C2IEDM)* to define the information exchange between different C2IS. The minimum requirement demanded by the MIP is that the "meaning and relationships of the information to be exchanged" (C2IEDM, 2005) need to be preserved.

The C2IEDM ontology defines a total of 194 entities in its logical data model. From these only 15 are independent, which means that their identification does not depend on any other entity. The independent entities provide an overview of the data model (see Figure 1).

The intensely connected entities *Object-Type* and *Object-Item* are both used to model a particular object in the C2IEDM. *Object-Type* defines the attributes which values are common among all objects of a particular type (e.g. fuel capacity in litre of a vehicle type). In contrast, the entity *Object-Item* defines the attributes which values can differ between all objects of a particular type (e.g. hull number of a vehicle). At the top level of their hierarchies the object entities define five different subtypes that can be modelled in the C2IEDM – *Facility* (e.g. airfield, road), *Feature* (meteorological, geographic and control features), *Material* (consumable material and equipment), *Organisation* (administrative and functional) and *Person*.

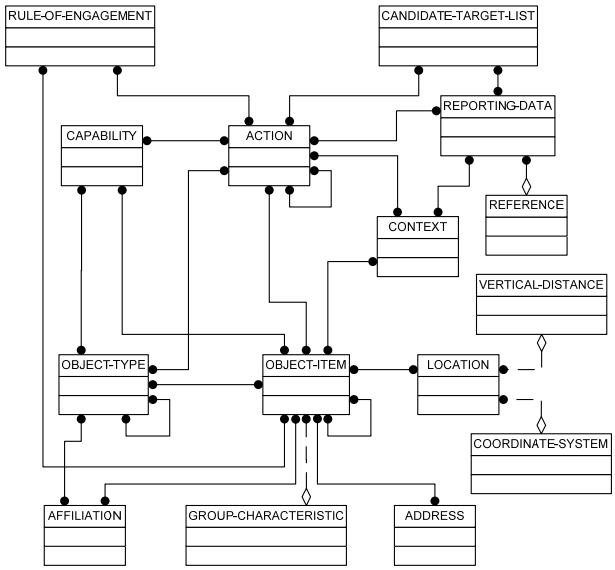


Figure 1. Independent entities of the C2IEDM (C2IEDM, 2005)

Spatial reference is summarized in the entity *Location* and is based on the definition of both absolute and relative points. The geometry of an object can be defined by a point, line, surface, geometric volume or by their respective subtypes (e.g. surface can be a corridor, polygon, polyarc, fan, track, orbit, or an ellipse). Relative points hereby permit positioning of objects relative to other objects as well as a simplified specification of geometries by the use of Cartesian offsets. The geographical reference is set by the definition of absolute points. Their coordinates are defined by latitude and longitude in the World Geodetic System 1984 (WGS 84) along with their vertical distance. The entity *Address* is not related to *Location* in the data model because it is only used for communication purposes. It is either an electronic address, accessible via a network service, or a physical one, reachable e.g. via postal services.

In the C2IEDM activity is represented by the entity *Action*. Planned and carried out activities as part of military operations are covered by *Action-Task*. In contrast, activities whose plan is unknown are covered by the entity *Action-Event*.

Information about the changing situation is stored directly in the corresponding entities of the data model. Subsequently the entity *Reporting-Data* is linked to these entities and provides amplifying data such as source, quality and timing. If the information is gathered from external sources, e.g. from a telephone conversation, the entity *Reference* can be used to provide metadata for the reported information.

The entity *Context* is in the terms of the C2IEDM a "collection of information that provides in its entirety the circumstances, conditions, environment, or perspective for a situation" (C2IEDM, 2005). *Context* therefore bundles only information that is already available and is in many cases limited to

collecting instances of *Reporting-Data* under a single label. It can also provide amplifying information for an *Object-Item*, e.g. hostile units that are approaching to the object's position. Additionally *Context* can be used to specify the prerequisites and estimated results of an *Action*.

The other independent entities of the C2IEDM shown in Figure 1 are less important for the adaptation and will not be discussed here.

2.2 The Disaster Management Data Model (DM²)

The *Disaster Management Data Model (DM²)* has been developed for an application in disaster management based on the C2IEDM. The data model does not cover all facets, e.g. resource management is not considered in detail yet. Nevertheless, many considerations concerning the DM² apply to ontologies in the domain of disaster management in general.

The most important entities of the DM² are shown in Figure 2 and the differences and new concepts in comparison to the C2IEDM are discussed in the following.

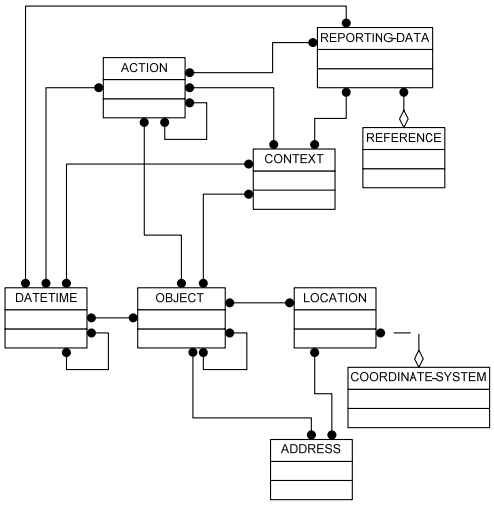


Figure 2. Important entities of the DM²

In contrast to the C2IEDM, which has been designed for information exchange, the DM² has been designed for serving as a knowledge base, so information storage, retrieval, and processing play an important role. For an easier usage in object-oriented programming languages, the disaster management ontology is modelled in terms of inheritance. The most obvious change is the unification of the two entities *Object-Type* and *Object-Item* into a single entity, which reduces some overhead.

The universe of discourse of the disaster management differs from the military one; therefore several changes had to be applied to the subtypes of the objects. To *Facility* the concept of buildings including their role (e.g. school), crossings and squares were added. The entity *Feature* was enhanced to handle administrative structures (e.g. district, town), damage sites, operation sections and standby areas. *Material* was renamed to the more common term *Resources* and holds the resource hierarchy. This depends on the actual conditions, because e.g. the actual fleet of vehicles differs from country to country. *Organisation* and *Person* were also adapted to the circumstances of disaster management, e.g. emergency operation centres were added. Objects can be set as part of an object hierarchy, e.g. all districts are subordinated to the corresponding town.

The DM² also differences between planned and unplanned activities. The subclass *Action-Task* has been extended by the tasks of disaster management, like fire fighting. *Action-Event* covers several disasters types in the C2IEDM, but for disaster response several additional attributes describing the nature of disasters are needed. These were introduced to the DM², e.g. for an earthquake the location of its hypocenter and magnitude are of significant importance for the decision making process.

Written report forms used in disaster management often provide a field to indicate the priority of the message. The terms and graduations that are used differ. The Common Alerting Protocol (CAP, 2005) provides a field called "urgency" with five levels that denote the available response time. In the DM² the four priority levels from the German standard DV 810 (1977) were introduced to the entity *Reporting-Data*.

Defining geometries is elaborated in detail by the *Location* entity of the C2IEDM. Nevertheless, the DM² uses only a subset of the possible geometry types. For the situation map only two dimensional geometries are needed and special surface subtypes like corridor are normally not used for describing disaster events. Additionally the concept for setting the geographical reference of objects and geometries had to be adapted. The restriction of the C2IEDM to coordinates defined in WGS 84 is too strict for application in disaster management, because in many cases the geographic information used as a basis for disaster response is available only in other coordinate systems. The DM² therefore permits the usage of another reference frame than the WGS 84. Nevertheless the DM² is restricted to a single reference frame, in order to avoid negative side effects due to inhomogeneous coordinate definitions.

The impacts of disasters depend on several aspects. One aspect is their location, e.g. in densely inhabited areas the damages and risks are often more severe. In urban areas buildings play an important role – they can be affected by disasters but they can also be used during disaster response, e.g. as gathering places for homeless people. Because in free form text the location of a building is normally given by its address, in the DM² the entities *Address* and *Location* are associated. The translation between the two entities can be performed by a simple geocoder as part of the augmentation component.

Concerning the practical implementation of the system and the transfer of the logical data model to a physical data model several aspects have to be considered. Because the DM² serves as a knowledge base, all relevant information can be aggregated in a database system. This information includes the geospatial data, e.g. the topographic information. For the processing step of semantic augmentation geo-related computations are often necessary, e.g. obtaining the distance between a damage site and nearby high risk buildings. For these computations either an embeddable GIS component or geospatial database functions can be used.

In ontologies associations between entities are of crucial importance. Regarding the DM² this is especially the case for the association entities that capture the highly dynamic situation in terms of time (*DateTime*), geometry and geographical reference (*Location*) and changing attribute values (*Status*). The principles behind this concept can be best shown by the association entities connecting the *Object* entity in the DM² (see

Figure 3). An individual object can move, so tuples with (*Object*, *DateTime*, *Location*) track the position in *Object-Location*. Also some parameters of an object can change over time, e.g. the availability of units. Therefore these values are tracked in tuples (*Object*, *DateTime*, *Attributes describing the status*) in *Object-Status*. But tracking is not only important for units but also for the position and extent of damages, e.g. fires or flooded areas. Because the affected area of an event is not restricted to facilities but can also be of feature type, flexible tracking of events is possible.

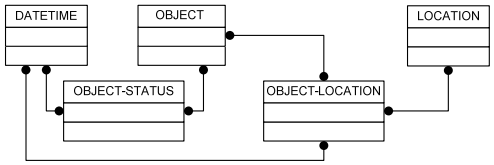


Figure 3. Associations of *Object*, *DateTime* and *Location*

Context in the DM² is no longer an exclusive collection of already available data as it is in the C2IEDM – it can be used for evaluation of information as well as for the inference of new information, as shown in the following chapter.

3. CONTEXT – A FUSION OF CONTENT

The importance of context is well-defined within the use of language. An isolated word refers to a denotation but is still meaningless. Thus, it is the context of the sentence which adds the meaning. Context is not necessarily restricted to language since this principle is similar in diverse disciplines. Processing isolated pixels within image analysis is as almost impossible as the interpretation of an individual report in the domain of disaster management. These reports are based on impressions and individual observations of one-site units. Also, the knowledge level of the messages depends on the source and the type of observation. As a consequence of the free text form, fuzziness of the content is inherent in the messages. Furthermore, vagueness is included whenever the author expresses assumptions. According to these different aspects, reports represent a generalised description of the situation in diverse abstraction levels. So-called *semantic gaps* are a conceptual summary of these facets.

During processing and analysis of incoming messages, human operators instinctively create references between different reports and organize these depending on the content. This type of semantic enrichment is based on the usage of context knowledge, which is evolved by the multitude of reports. For an automatic application in disaster management, the usage of context knowledge in the message analysis has to be adapted from human approaches. Content dependent fusion of diverse messages offers an elegant solution. For detecting these similar and significant content structures, methods of information technology are necessary.

Such methods are a kind of *data mining* because of the process-oriented detection of significant structures in a dataset (Bensberg 2001). Contrary to a number of publications, Bensberg emphasizes the importance of structures. Hence the limitation to large amounts of data is secondary for their processing.

3.1 Context in the Disaster Management Data Model

Within the DM² context is a powerful instrument for semantic enrichment. The implementation is related to the C2IEDM, by the direct cross linking of the main entities *Action*, *Object*, *DateTime* and *Reporting-Data* (cf. Figure 2). By this approach the essential information is concentrated in association to a single entity. Additional information, like the geographical position, is already given inherently by the enlisted objects.

Nevertheless, the philosophy behind the context differs as the context in the DM² is organized task-related. Thus the main context of a disaster is subdivided into three separate types of contexts providing three complementary perspectives on the aggregated information. The first context *Action-Context* is focused on the relations between the diverse actions and their dependences. Thus it becomes possible to model e.g. injured people in consequence of a fire. The *Feature-Context* is focused on the coherences of objects, actions and time. In this manner individual objects like an organizational unit or individual actions like a burning facility are represented as a whole situation including the temporal references. The third type of context, the *Reference-Context*, relates the reported facts of a message to each other. That way the facts of a report are bundled and relate to the same attributes. Following this approach a multitude of either-way independent, crossed or coincided context-entries arise.

According to this method, it becomes possible to identify and detect all coherences within the ontology for semantic enrichment.

3.2 Processing Context Knowledge

The discussed ontology offers the possibility to link different facts case related in different types of context. However, the types of context should be identified as automatically as possible for a set of data. Thus the processing of context starts with the “least common denominator” of the reports.

For finding this “least common denominator” the course of messages representing the chronology of a disaster has to be considered. In the first phase of a disaster the incoming messages show common characteristics, which are quite simple. Normally, the messages include indications for a disaster along with the source of information and a location. In this manner the disaster location offers the first reference for further processing and the basis for creating each type of context. These considerations are quite similar to the approach of a human operator.

The definition of “the same location” for creating context is possible by the introduction of a so-called *event horizon*. The event horizon is the sphere of influence of a reported fact and its shape and size depend on the meaning of the content. The sphere of influence in this application is a buffer, individually defined for each type of reported fact. That way the sphere of influence of a derailment is much smaller than the sphere of influence of smoke, which can be seen and smelled over a long distance. According to that approach, reported facts are related whenever their event horizons overlap. Thereby it is important not to confound the “virtual” sphere of influence of the reported fact with the fact itself.

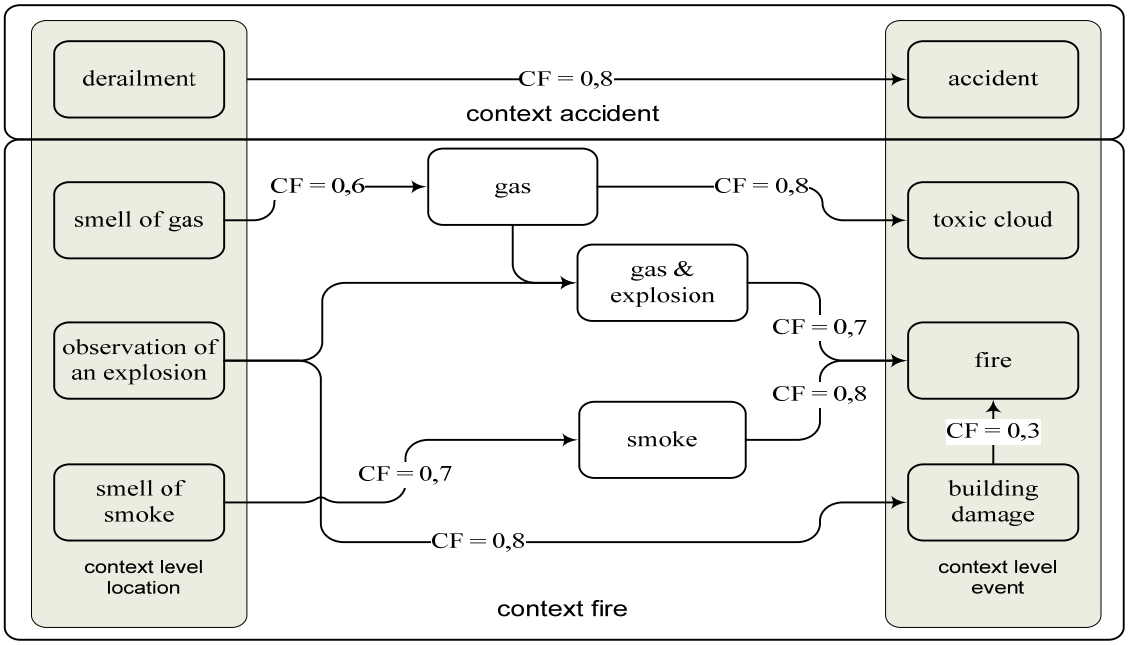


Figure 4. Inference net including the certainty factors and the levels of context

The location context offers the possibility for a new kind of interpretation of message content. Similar locations, in the meaning of overlapping event horizons, which are mentioned in various reports, establish the first evidence for a particular disaster, a so-called *hypothesis*. Conditions respectively restrictions for matching diverse types of facts to the same context have to be defined. The inference nets from the knowledge based diagnose system *MYCIN* offer a solution for this problem, presented in the following (Beierle and Kern-Isberner, 2006).

MYCIN (cf. Buchanan and Shortliffe, 1984) is one of the first knowledge based expert systems designed for processing and representation of uncertain knowledge. MYCIN has been developed since 1972 by the University of Stanford for diagnosing different diseases in dependence to the different symptoms. Modelling these coherences is made possible by defining conditions and relations of different evidences. These basic relations are "if A then B", "A and B" and "A or B". This assortment of relations is quite simple, however similar to the approach of a human operator. Additionally the relation "if A then B" is complemented by a *certainty factor (CF)* for implementing a weighting as shown in Equation 1.

$$A \xrightarrow{CF(A \to B)} B \qquad (1)$$

This equation expresses the degree of belief of the conclusion B if the premise of A is true. Thereby it is important not to confound the degree of belief and the probability for B on the condition of A. The confidence region for the CF is given by the interval [-1, 1] $\in \mathbb{R}$. The meaning of this range is [-1 ≡ confutes], [0 ≡ neutrally] and [1 ≡ confirmed]. In this manner it is possible to model evidences for and against a *hypothesis*.

CF [B] represents the *cumulative certainty factor* with dependence to the "rule base" (A→B) and the *evidence* (in this case represented by *CF [A]*). So the cumulative certainty factor exists for the *evidence* as well as the *hypothesis.* This serial combination is shown in Equation 2.

$$CF[B] := CF[B,\{A\}] = CF(A \to B) * \max\{0, CF[A]\} \qquad (2)^*$$

For processing context, the propagation rules for the conjunction and disjunction are represented by Equation 3. These equations make the empiric definition of the relations "A and B" and "A or B" possible. In this manner the interaction between different *evidences* to a *hypothesis* can be balanced.

$$\begin{aligned} CF[B,\{A \wedge B\}] &= \min\{CF[A], CF[B]\} \\ CF[B,\{A \vee B\}] &= \max\{CF[A], CF[B]\} \end{aligned} \qquad (3)^*$$

The final case for modelling is the parallel combination of *evidences.* This is necessary whenever different *evidences* link to the same *hypothesis*. Equation 4 represents the mathematical interpretation of this case.

$$CF[B,\{A_1,...,A_n\}] = f(CF[B,\{A_1,...,A_{n-1}\}], CF[B,\{A_n\}]) \qquad (4)^*$$

With these equations the different reports can be processed and matched to the different types of context. This procedure is illustrated by an example, in which the initial situation is given by four reports with the content of derailment, smell of gas, observation of an explosion and smell of smoke. The messages comply with the basic condition of overlapping event horizons, because of "the same location" of the reported facts. That is why they are contained in a common location context (cf. Figure 4).

The spatial information needed for this analysis is given by sender's explanations, like: "I have seen an explosion at the ARAL petrol station". The location given in this report is the distinctive ARAL petrol station which has to be part of the DM² database. The reported fact is the observation of an explosion.

Processing these messages creates possible *Action-Events* and defines also the specific event contexts. The report about observation of an explosion is an evidence for the *hypothesis* of

* (Beierle and Kern-Isberner, 2006)

building damage. The degree of belief for this *hypothesis* is [0,8] and results from Equation 2. On the other hand the same content from the observation of the explosion creates in combination with the *evidence* of gas a new *evidence* for the *hypothesis* fire. This conjunction can be modelled by Equation 3. Also, the *hypothesis* fire is a parallel combination of the *evidence* gas & explosion, smoke and building damage. This dependence is described by Equation 4.

For the creation of context, the dependencies between *evidences* are important. When *evidences* depend on each other, they create a common context. Following this rule, the context of fire is created by joining the content of the three dependent messages. By contrast the contexts of independent *evidences* are generally incompatible. In the example (Figure 4) this can be seen concerning the report content of the derailment, which does not fit the context of fire. The evidence of the derailment uniquely links to the *hypothesis* of accident. According to that link, the context of accident arises and includes the derailment report by a reverse processing. So the context of fire and the context of accident can be seen as independent.

hypothesis	accident	toxic cloud	fire	building damage
belief	[0,90]	[0,48]	[0,91]	[0,80]

Table 5. Degree of belief for the *hypotheses* of *Action-Events*

The cumulative certainty factor shows the degree of belief for a *hypothesis*. This is important whenever the *evidences* provide different *hypotheses*. Within the example (Figure 4) the *evidences* smell of gas, observation of an explosion and smell of smoke lead to the *hypotheses* of toxic cloud, fire and building damage. On the basis of the belief shown in Table 5, the decision can be made in favour of the *hypothesis* fire.

CONCLUSION

A specific application of "automated geo-spatial data acquisition and mapping" offers the transformation of verbally given geo-information into a situation map. For a successful transformation it is essential to develop a domain specific knowledge base. The presented Disaster Management Data Model is based on the Command and Control Information Exchange Data Model. Although the C2IEDM is a sophisticated standard, the DM² points out important considerations that have to be taken into account for disaster management ontologies in general.

The semantic of the reports, which is given by the use of context, plays an important role for the transformation. The first common context level is defined by the spatial reference of the reports. Starting from the context level of location, it is possible to develop the domain context.

The future research focus is on uncertainty and reliability, which has not been covered by the SOKRATES system yet. These are different facets of messages in the disaster management domain. Uncertainties exist concerning the location, the dimension and the quantity. Considerations of reliability, resulting from message source as well as adverbial phrases like "probably", "presumably" or "perhaps", have to be integrated. A possible solution for this problem is also seen in the usage of context knowledge.

REFERENCES

Beierle, C., Kern-Isberner, G., 2006. *Methoden wissensbasierter Systeme*. Friedr. Vieweg & Sohn Verlagsgesellschaft / GWV Fachverlag GmbH, Wiesbaden.

Bensberg, F., 2001. *Web log mining als Instrument der Marketingforschung: ein systemgestaltender Ansatz für internetbasierte Märkte*. Dt. Univ.-Verl., Wiesbaden: Gabler.

Buchanan, B.G., Shortliffe, E.H., 1984. *Rule-Based Expert Systems: The MYCIN Experiments of the Stanford Heuristic Programming Project*. Addison-Wesley Publishing Company.

C2IEDM, 2005. The C2 Information Exchange Data Model v. 6.15e. http://www.mipsite.org/publicsite/03-Baseline_2.0/C2IEDM-C2_Information_Exchange_Data_Model/ (accessed 17 Feb. 2007)

Cowie, J., Lehnert, W., 1996. Information Extraction. *Communications of the ACM*, 39(1), pp. 80-91.

DV 810, 1977. KatS-Dv 810 Sprechfunkdienst. http://gsb.download.bva.bund.de/BBK/KatS_Dv_810.pdf (accessed 18 Mar. 2007)

Gruber, T-R., 1993. A Translation Approach to Portable Ontology Specifications. *Knowledge Acquisition*, 5(2), pp. 199-220.

Hecking, M., 2004. Improve Interoperability by Formalizing the Natural Language Parts of Military Messages. In: *Proc. of the Information Systems Technology Panel Symposium (IST-042/RSY-014) 'Coalition C4ISR Architectures and Information Exchange Capabilities'*, Hague, Netherlands.

Neumann, G., Backofen, R., Baur, J., Becker, M., Braun, C. 1997. An Information Extraction Core System for Real World German Text Processing. In: *Proc. of the 5th International Conference of Applied Natural Language*, Washington, USA.

OASIS-CAP, 2005. Common Alerting Protocol. v. 1.1. http://www.oasis-open.org/committees/download.php/15135/emergency-CAPv1.1-Corrected_DOM.pdf (accessed 20 Mar. 2007)

Schade, U., 2004. Automatic Report Processing. In: *Proc. of the 9th International C2 Research and Technology Symposium (ICCRTS)*, Copenhagen.

Werder, S., Mueller, M., Mueller, M., Kaempf, C., 2006. Integrating Message Information into Disaster Management Maps: Transferability of a System of the Military Domain. In: *Proc. of the ISPRS Commission IV Symposium on Geospatial Databases for Sustainable Development*, Goa, India.

ACKNOWLEDGEMENTS

The presented work has been funded by the Deutsche Forschungsgemeinschaft (DFG), project no. BA 686/16 "Abstraction of Graphically and Verbally Represented Geoinformation" (Christian Lucas) and as part of the Collaborative Research Center (CRC) 461 "Strong Earthquakes: a Challenge for Geosciences and Civil Engineering" (Stefan Werder).

AUTOMATIC DISCRIMINATION OF FARMLAND TYPES USING IKONOS IMAGERY

P. Helmholz[*, a], M. Gerke[b], C. Heipke[a]

[a] Institute of Photogrammetry and GeoInformation, Leibniz Universität Hannover, Nienburgerstr. 1, 30167 Hannover, Germany, helmholz@ipi.uni-hannover.de
[b] International Institute for Geo-Information Science and Earth Observation - ITC, Department of Earth Observation Science, Hengelosestraat 99, P.O. Box 6, 7500AA Enschede, Netherlands, gerke@itc.nl

KEY WORDS: Automation, Change Detection, GIS, Quality, Updating

ABSTRACT:
The aim of this paper is to introduce an approach for the discrimination between the farmland types cropland and grassland from single satellite images by estimating the main direction of cultivation in cropland. The approach uses structural features caused by the cultivation, in particular straight lines caused by agriculture machines. The core of the approach is the transformation of an edge image into Hough space, and a following interpretation of the results to determine the main direction. The new approach is presented in detail and is illustrated with the help of examples. The examples and the evaluation demonstrate the potential and the limits of this approach.

1. INTRODUCTION

An immense amount of decisions in private and public life relies on geospatial information. The automatic management of spatial data is performed in geoinformation systems. The usefulness and acceptance of such geoinformation systems mainly depend on the quality of the underlying geodata. The availability of high resolution optical satellite imagery appears to be interesting for geospatial database applications, namely for the capture and maintenance of geodata. Among others, Büyüksalih and Jacobsen (2005) show that the geometry of IKONOS and Quickbird imagery is accurate enough for topographic mapping.

The main topic of this paper is the automated verification of existing topographic data using high resolution satellite imagery. For this task we have set up an interdisciplinary project called WiPKA-QS[**]. One of the main tasks in WiPKA-QS is to extend the approach regarding the discrimination between deciduous and coniferous forests and between cropland and grassland. For the verification of these object classes we use explicit radiometric features as well as structural features (Busch et al., 2006).

The object classes cropland/grassland cover a large area in many countries and are therefore of prime relevance. Hence, we focus on the discrimination of these classes at first. A main differentiation between grassland and cropland is the exploitation of structures caused by the cultivation, which is conducted more frequently in crop fields compared to grassland. The agricultural machines normally cause parallel straight lines which are observable in the image. Our approach for the detection of these parallel straight lines is divided into three steps; we detect edges which then are transformed into Hough space, and finally the orientation is estimated.

In this paper, we present at first related work and discuss briefly the applicability of this work to our problem of discrimination between cropland and grassland. After an introduction to the project WiPKA-QS, we describe our Hough-based approach, and we give four examples. The last section concludes the paper.

2. RELATED WORK

In this selection we briefly review approaches for extracting different vegetation types based on structural and radiometric features. A more complete review of extracting vegetation objects, e.g. based on hyperspectral information or multi-temporal imagery, is beyond the scope of this paper.

The idea to use structural features is also pursued in (Trias-Sanz, 2006) who uses structural properties to discriminate objects with similar radiometric and textural properties (e.g. forest and plantation) in high resolution satellite images. These object classes can be distinguished only by precise orientation characteristics e.g. forest and untilled fields have none, tilled fields have one, and orchards and vineyard have two main structure directions. All computations are carried out within a pre-selected window called texton, whose shape and size can be arbitrary. The starting point is the calculation of a variogram which is similar to autocorrelation. After the transformation of the variogram into a special accumulation space, a histogram of this space is derived. The maximum of the function in this histogram corresponds to the primary direction in image space. A disadvantage of this approach is that the appearance of the structural features like cultivation structures and field crop has

* Corresponding author

** **Wi**ssensbasierter **P**hotogrammetrisch-**K**artographischer **A**rbeitsplatz - **Q**ualitätss**i**cherung (Knowledge-based photogrammetric-cartographic workstation - quality management)

to be homogeneous. This approach can be used to discriminate a large number of object classes by properly choosing the texton, but can yield wrong results if the texton parameters are selected inappropriately. In contrast, we focus on the discrimination of only two object classes (grassland and cropland).

Additional methods for the determination of structural features are the Fourier and Radon transformation. Chanussot et al. (2005) estimate the orientation of vineyard rows automatically using the Fourier spectrum of a pre-processed image and its Radon transformation. However, the authors apply their method to high resolution aerial data only, and furthermore, a very important assumption is a regular spacing between the rows. This assumption is usually satisfied for vineyards, but not necessarily for cropland. In cropland the distance between rows can vary from one field to the next, depending on the culture of vegetation, and the kind of machine which was used, and furthermore, the visibility of the structures in the image of one field.

A huge number of publications deals with various approaches of orientation estimation.. Le Pouliquen et al. (2002) use convolution masks for a scale-adaptive orientation estimation which is divided into a gradient based and valleyness operator. Compared to this approach, De Costa et al. (2002) use orientation difference histograms. The focus of the approach of De Costa et al. is not the quantitative estimation of the direction, instead only the existence of a direction is of interest. This idea is also sufficient to characterize cropland. However, both approaches were tested for synthetic images only.

Warner and Steinmaus (2005) identify orchards and vineyards in IKONOS panchromatic imagery. In this approach the classes are detected using autocorrelation. After the definition of a square kernel and the normalization of each pixel of this kernel the autocorrelation for the cardinal directions and both diagonals is determined. For each autocorrelation calculation (called autocorrelogram) each pixel is analyzed separately to identify orchards. An orchard pixel is detected if an orchard pattern is identified in more than one autocorrelogram centered on the same pixel. However, the trees have to be approximately equally spaced. Similar to the aforementioned approach of Chanussot et al. (2005) this assumption is usually not met in cropland.

Radiometric features were used by Itzerott and Kaden (2006 and 2007) to discriminate between various farmland types. They analysed typical economic plants and grassland in the German federal state of Brandenburg. It was shown that grassland possesses a non-zero NDVI (**N**ormalised **D**ifference **V**egetation **I**ndex) in all seasons, whereas during the winter season several agricultural plants have a very low NDVI which is significantly different from the NDVI of grassland.

The literature review shows that some work on the classification of farmland types using structural and radiometric features has been done. However, either the approaches rely on training samples or on precise knowledge on the structure or radiometric properties of the field class, i.e. they are model-driven. For our approach we want to be independent of such conditions and therefore persue a more data-driven approach.

3. AUTOMATIC DISCRIMINATION OF FARMLAND TYPES

3.1 Workflow of WiPKA-QS

The aim of the project WiPKA-QS is the automated verification of the German topographic reference dataset ATKIS*** or in general GIS. The main components of ATKIS are the object based digital landscape models (DLM) encompassing several resolutions with a geometric accuracy of up to +/-3m.

The core of the automated procedure is the knowledge-based image interpretation system GeoAIDA (Bückner et al., 2002). GeoAIDA uses a semantic network that represents the scene to be analyzed. First, in a top-down or so called model-driven step, the system searches for evidence for the object to be verified in the orthoimage. Evidence can be the existence of a main direction of cultivation. Thereby, the system focuses only on objects of interest. Afterwards, in the bottom-up or so called data-driven step, the system derives an acceptance or rejection decision assessing the evidence. Hence, discrepancies between objects and the image features can be detected. Is the verification of an object successful the system labels this object as accepted (green); otherwise the object is labelled as rejected (red). For the rejected objects, a final decision is made by a human operator. Further details of the system are available in (Busch et al, 2004) and furthermore in (Müller and Zaum, 2005), its workflow is sketched in Figure 1.

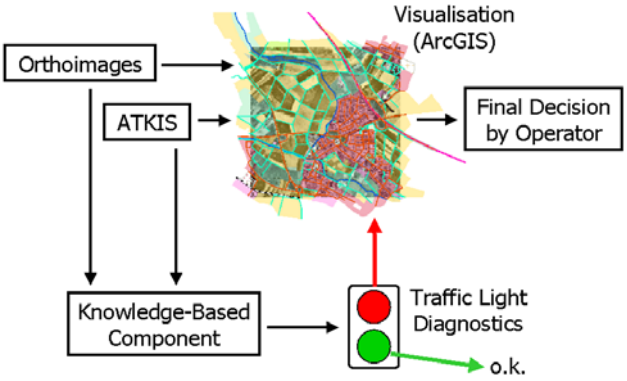


Figure 1: Workflow WiPKA-QS

3.2 Strategy

The underlying semantic model of the approach is shown in Figure 2. The first level of the semantic net describes the *Real World*: farmland can contain cropland and grassland, and furthermore cropland consists of untilled or tilled cropland. The second level *Geometry/Material* explains the geometrical and material characteristic of the objects. Finally, the *Imagery* level shows the characteristics which are visible in the image.

*** **A**mtlich **t**opographisch-**k**artographisches **I**nformationssystem (Authoritative Topographic Cartographic Information System)

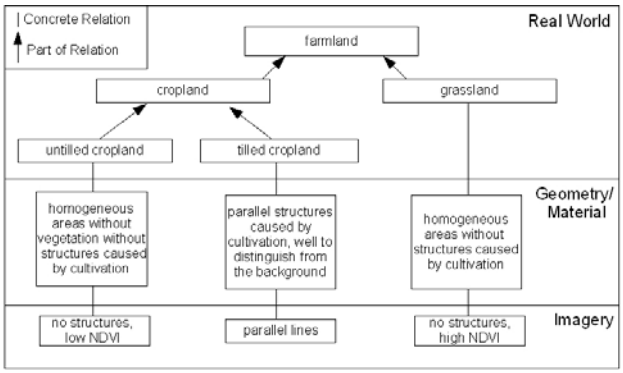


Figure 2: Knowledge Presentation using Semantic Net

Untilled cropland shows no structural features, at least in the image resolution we are dealing with. The discrimination between untilled crop fields and grassland in which structural line features are not visible is achieved using radiometric features like in this case the NDVI. Untilled cropland has a low NDVI, whereas the NDVI of grassland is usually rather high. If we detect no vegetation, grassland can be ruled out (Itzerott and Kaden, 2006 and 2007).

Furthermore, three issues must be considered. First, due to disturbances in the object border area, e.g. structures caused by turning agricultural machines, the approach is restricted to the interior object area. The reduction is done using simple erosion.

Second, in ATKIS or other GIS, inside one object the existence of more than one land cover class are tolerated if a size threshold is not exceeded. Furthermore, several objects of the same land cover type are permitted. For example, in an ATKIS object "cropland" the existence of a small area of grassland is allowed and it is possible that several crop fields with different cultivation directions are present. Therefore, a verification of one object is performed subdividing the objects into segments of radiometrically homogeneous regions, or alternatively, tiles as shown in Figure 3 before further processing. In the top-down step our approach searches for evidence for the existence of a main direction in each segment or tile. In the bottom-up step adjacent segments or tiles with the same main direction are merged, and subsequently, the object is assessed.

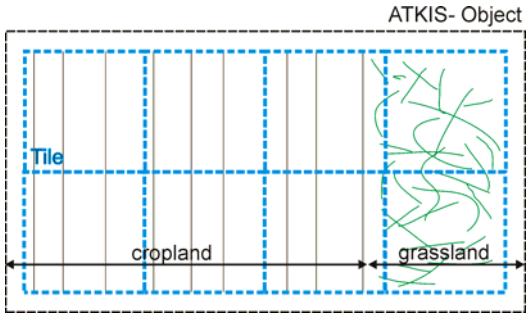


Figure 3. Schematic sketch for tiling a GISobject

In this paper we focus on the image analysis step – the top-down step where we discriminate cropland and grassland using structural as well as radiometric features in one segment or tile.

3.3 Approach to the estimation of structural features

3.3.1 Edge detection

We work with pansharpened four-channel (RGB and Near Infrared) images with a spatial resolution of 1m. An example of a cropland object is shown in Figure 4. Currently, we compute an average grey value for each pixel from the four channels (termed 'intensity channel' in the following).

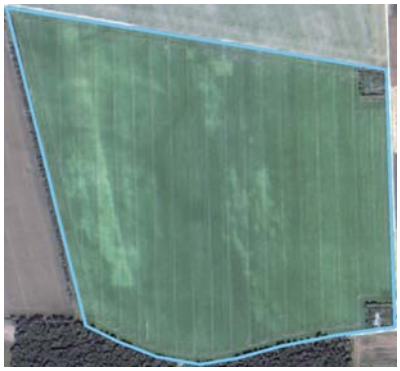


Figure 4: Image of a cropland object represented in RGB

In a pre-processing step the images are enhanced such that the contrast is optimized and further an edge-preserving smoothing limits the impact of noise to edge extraction. Then, a pre-processing an edge image is computed using the Canny operator (Canny, 1986). Compared to other edge detection operators, the Canny operator permits a better detection of edges, especially under noise conditions (Sharifi et al., 2002). Due to the described pre- processing step only little attention needs to be paid to the trimming of parameters for the edge extraction. The edge image of the cropland object depicted in Figure 4 is shown on the left side of Figure 5.

An alternative to edge extraction would be to extract lines (bar egdes), e.g. using the sophisticated line extraction operator proposed by Steger (1998). However, not all structures in the field appear as lines with a distinct and constant width. Therefore, to extract edges is the more general approach here.

3.3.2 Hough space

An analysis of the structure inside the field is carried out by transformation of the edge image (image space) to a proper accumulation space (Hough space). The line parameters in image space are the angle between the normal vector of the line and the x-axis (ϕ), the distance of the line from the origin (d). Figure 5, right side, shows an illustration of a cropland object in image space and in Hough space.

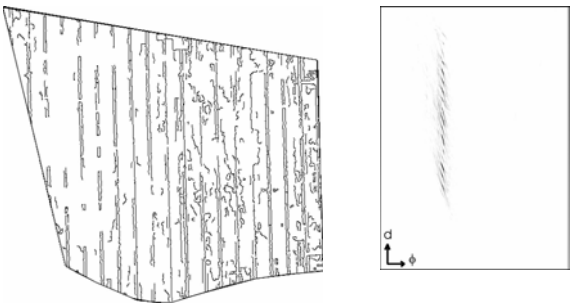


Figure 5: Edge image (left) and its Hough space representation (right) of the cropland object shown in Figure 4

Thus, parallel lines are mapped into points situated vertically above each other, assuming the ϕ-parameter is mapped to the horizontal axis in Hough space. Furthermore, if the space between lines in image space is constant, a periodicity of the point positions in Hough space is observable.

By extracting these points in the Hough image we focus on salient lines in image space. Points are extracted using the Förstner operator (Förstner and Gülch, 1987) and are called points of interest (POI).

3.3.3 Orientation Estimation

In the next step, a histogram of the extracted points along the ϕ–axis in Hough space is derived. The histogram of the previous cropland object is shown in Figure 6. The unit for ϕ on the x-axis is grad, the y-axis shows the number of occurrences of the angles.

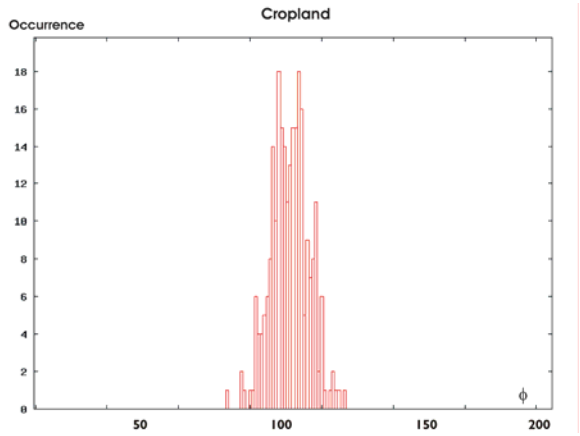


Figure 6: Histogram of angles in Hough Space of Figure 5 (right)

3.3.4 Assessment

As a final step we investigate the in the histogram probable direction of cultivation by computing the largest peak in the histogram, and in addition the standard deviation σ for this peak. For cropland σ must lie below a pre-defined threshold t, whereas for grassland σ is assumed to le larger than t. In addition, a number of at least s lines with the same direction must have been detected. The parameters s and t are defined heuristically. In Figure 6 σ amounts to approximately 5grad and at least 5 lines with the same direction are detectable. Finally, the object depicted in Figure 4 is accepted as a cropland object.

3.4 Examples and Evaluation

The described approach was implemented and tested on a number of IKONOS images. Here, we show results obtained from a scene acquired on June-24m, 2003 in the area of Weiterstadt, Hessen. In addition, we present results from a scene close to Rostock, Mecklenburg- Vorpommern acquired on July-27, 2006.

Examples of images, edge images, and derived histograms of ATKIS grassland and croplands objects are shown in Figure 9-Figure 11. In the histogram of grassland of Figure 9 a significant edge direction is not detectable, and the object is accepted by the verification system.

The results presented in Figure 10 indicate that object can be accepted as cropland. Although, the edge image is not as clear as the one in Figure 6, a main direction can still be detected unambiguously.

In contrast to the preceding example, in Figure 11 a successful verification of the cropland object is not possible. The cultivation lines are not separable from the background as can also be seen in the edge image. In the histogram a significant peak can not be detected. The verification system rejects the object – a false negative decision.

For the second scene depicting an area close to Rostock the approach was tested on the whole image compared to the few shown examples of the scene Weiterstadt. An example of a cropland object in this scene is shown in Figure 12. The peak of the main direction is at 12grad. The standard deviation is 3.6grad. In the histogram three peaks are visible. The first peak has the highest occurrence and is the main direction of cultivation. The second peak has an occurrence of less than five and lies below the threshold. This peak is noise in the image caused by disturbances, i.e. in this case the trees inside the field. The third peak is a part of the first one, appearing at approximately 200grad due to the periodicity of the edge direction.

The results of the scene Rostock are investigated using a confusion matrix (Figure 7). The percentage of corresponding acceptance indicates the efficiency of the approach. There will also be undetected errors if objects which have been accepted by the automatic procedure, are rejected by the human operator. The percentage of undetected errors has to be as small as possible.

Automatic / Human Operator	Green	Red
Green	Efficiency	Interactive Final Check
Red	Undetected Errors	Interactive Final Check

Figure 7: Confusion Matrix

In Figure 8 the confusion matrix for the scene Rostock is shown. At this scene 77 cropland objects are checked by a human operator and by our approach. Objects rejected by automatic system need to be checked by the human operator interactively. The threshold t for σ is 5grad, for s a value of 5 is chosen. As shown in Figure 8 the efficiency of our approach is around 55%, the false alarms are 31%. 11 objects (14%) which were wrong in the GIS are detected by the system automatically. Undetected errors are not present.

Automatic / Human Operator	Green	Red
Green	55%	31%
Red	0%	14%

Figure 8: Confusion Matrix of the scene Rostock

3.5 Discussion

The examples show the potential of the approach. In general the presented approach is rather robust, because given all the edge pixels we are only interested in two single value, namely the

number of occurrences in the highest peak of the angle histogram and its standard deviation. Therefore, gaps in the edge image have little influence on the determination of the main direction.

It should be noted that rather than transforming the edge image into Hough space followed by projection to the ϕ-axis, we could have derived the edge direction histogram directly from the edge image. However, the resulting histograms are much noisier, if no operation analogous to the POI selection in Hough space is carried out.

The whole strategy of this approach fails if

- line structures caused by cultivation are not observable (e.g. maize close to harvest, untilled crop fields)
- lines in crop fields are not straight respectively parallel to each other (e.g. on hillsides),
- grassland possesses parallel lines

Regarding the first point, we already described the verification of untilled cropland at the beginning in section 3.2 using NDVI. The last two aforementioned cases are not very common in Germany. However, the influence of these problems is to be investigated, and if necessary the strategy is to be modified.

4. CONCLUSIONS

We describe a strategy for the automatic discrimination of the farmland types grassland and cropland using IKONOS imagery by detecting parallel lines caused by cultivation.

Concentrating on the interior object area, first, we check the NDVI to rule out untilled fields. The core of the structure based approach is the detection of edges using the Canny operator. The edge image is then transformed into Hough space. After the determination of points of interest (POI) in Hough space, a histogram is calculated. This histogram represents the number of occurrences of POI depending on the angles. The assessment which land cover object is present is conducted by using a statistical interpretation of the histogram.

An advantage of our approach is the fact that periodic rows or a minimum distance between rows are not required. Furthermore, since we are using edge detection in contrast to texture a training of parameters is not necessary. Therefore, an intervention of a human operator is only necessary to check rejected objects. First results for cropland objects have shown the efficiency of our approach to be around 55%.

We conclude the paper with a brief outlook to the next steps which are the segmentation of fields and the definition of tiles in every GIS object. Additionally, a final assessment step of the respective GIS object in the bottom-up step by merging the results of every segment/ tile, and the results of other operators of WiPKA will be developed. In addition the approach will be tested on more scenes for cropland and grassland objects.

5. ACKNOWLEDGMENT

This work is funded by the German Federal Agency for Cartography and Geodesy (BKG). We would like to thank also the anonymous reviewer for their valuable comments.

6. REFERENCES

Busch, A., Gerke, M., Grünreich, D., Heipke, C., Helmholz, P., Liedtke, C. E.; Müller, S., 2006: Automated Verification of a Topographic Reference Dataset using IKONOS Imagery. In *IntArchPhRS, Band XXXVI/4*, Goa, 2006, pp. 134-139

Busch, A., Gerke, M., Grünreich, D., Heipke, C., Liedtke, C.-E., Müller, S., 2004. Automated verification of a topographic Reference dataset: system design and practical results. In: *International Archives of Photogrammetry & Remote Sensing*, Vol.XXXV, Part B2, pp.735–740.

Büyüksalih, G. and Jacobsen, K., 2005: Optimized geometric handling of high resolution space images. In *ASPRS annual convention, Baltimore*, 9p (on CD-ROM).

Bückner, J., Pahl, M., Stahlhut, O., Liedtke, C.-E., 2002: A knowledge-based system for context dependent evaluation of remote sensing data. In: L. J. v. Gool (ed.), *DAGM-Symposium*, Lecture Notes in Computer Science, Vol. 2449, Springer, Zurich, Switzerland, pp. 58–65.

Canny, J.F., 1986: A computational approach to edge detection. In *IEEE Trans. of Pattern Analysis and Machine Intelligence*, 8(6): pp 679-698

Chanussot, J., Bas, P., Bombrun, L., 2005: Airborne Remote Sensing of Vineyards for the Detection of Dead Vine Trees. In *Proc. IGARSS*, Seoul, Korea, Aug. 2005, pp. 3090- 3093

De Costa, J.P., Germain, C., Baylou, P., 2002: Orientation difference statistics for texture description. In *Proc. 16th Intl. Conf. on Pattern Recognition (ICPR 2002)*, Quèbec City, Quèbec, Canada, Aug. 2002, pp. 652- 655

Förstner, W. and Gülch, E., 1987: A Fast Operator for Detection and Precise Location of Distict Point, Corners and Centres of Circular Features. In *Proc. of the ISPRS Conference on Fast Processing of Photogrammetric Data*, Interlaken, June 1987, pp.281-305

Itzerott, S. and Kaden, K., 2006: Ein neuer Algorithmus zur Klassifizierung landwirtschaftlicher Fruchtarten auf Basis spektraler Normkurven. In *Photogrammetrie, Fernerkundung, Geoinformation 6/2006*, pp. 509- 518

Itzerott, S. and Kaden, K., 2007: Klassifizierung landwirtschaftlicher Fruchtarten. In *Photogrammetrie, Fernerkundung, Geoinformation 2/2007*, pp. 109- 120

Le Pouliquen, F., Germain, C., Baylou, P., 2002: Scale-adaptive Orientation Estimation. In *Proc. 16th Intl. Conf. on Pattern Recognition (ICPR 2002)*, Quèbec City, Quèbec, Canada, Aug. 2002, pp.688- 691

Müller, S., Zaum, D. W., 2005: Robust building detection in aerial images. In: *International Archives of Photogrammetry and Remote Sensing*, Vol. XXXVI, Part B2/W24, pp. 143-148.

Sharifi, M., Fathy, M., Mahmoudi, M.T., 2002: A Classified and Comparative Study of Edge Detection Algorithms. In *Proc. of the Intern. Conf. on Information Technology: Coding and Computing (ITCC 2002)*, Las Vegas, Nevada, USA April 2002, pp. 117-120

Steger, C., 1998: An Unbiased Detector of Curvilinear Structures. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(2), pp. 311–326.

Trias-Sanz, R., 2006: Texture Orientation and Period Estimation for Discriminating Between Forests, Orchards, Vineyards, and Tilled Filled. In: *IEEE Transactions on Geoscience and Remote Sensing*, vol. 44, no. 10, Oct. 2006, pp. 2755

Warner, T.A. and Steinmaus, K., 2005: Spatial Classification of Orchards and Vineyards with High Spatail Resolution Panchromatic Imagery. In *Photogramm. Eng. & Remote Sensing*, vol. 71, no.2, Feb. 2005, pp. 179- 187

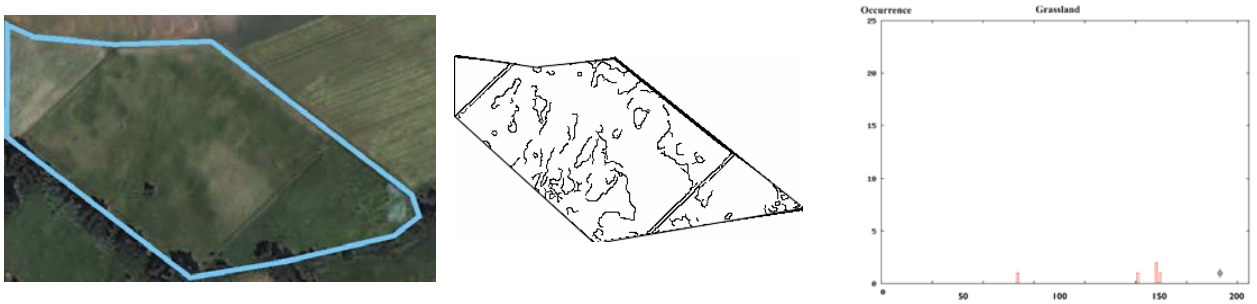


Figure 9: Image space, edge image and histogram of a grassland example (from left to right)

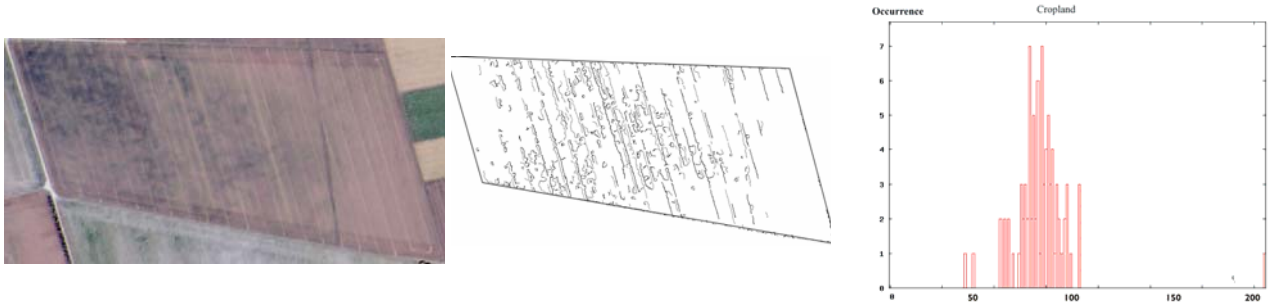


Figure 10: Image space, edge image and histogram of a cropland example (from left to right)

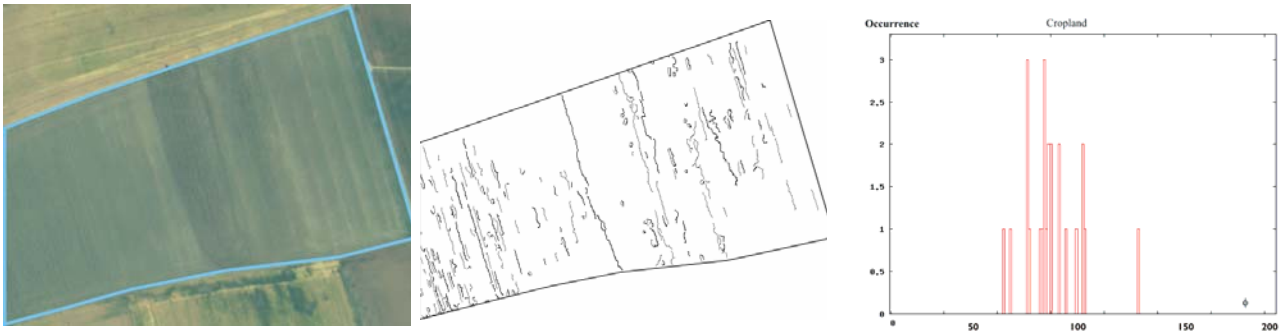


Figure 11: Image space, edge image and histogram of a cropland example (from left to right)

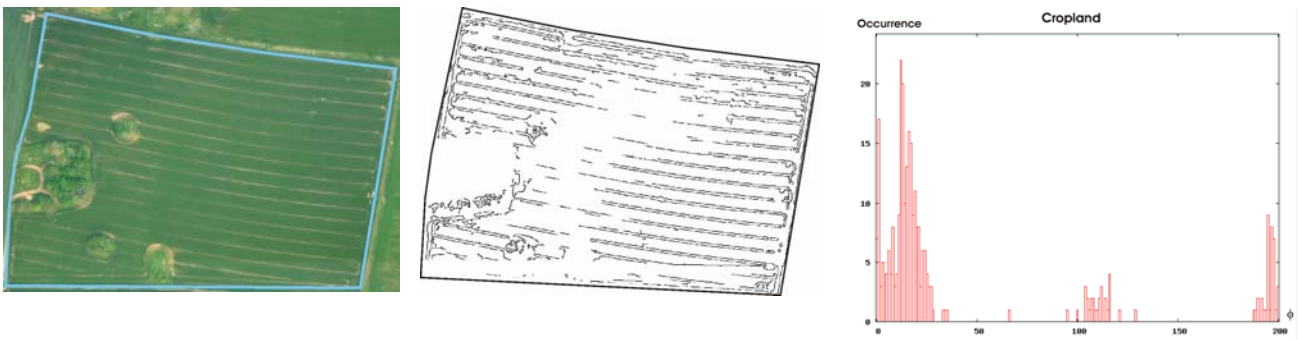


Figure 12: Image space, edge image and histogram of a cropland example (from left to right)

MODEL-DRIVEN AND DATA-DRIVEN APPROACHES USING LIDAR DATA: ANALYSIS AND COMPARISON

F. Tarsha-Kurdi*, T. Landes, P. Grussenmeyer, M. Koehl

Photogrammetry and Geomatics Group, MAP-PAGE UMR 694, INSA de Strasbourg,
67000 Strasbourg, France. Phone/Fax: +33 3 88 14 47 33
fayez.tarshakurdi|tania.landes|pierre.grussenmeyer|mathieu.koehl@insa-strasbourg.fr

Commission III, WG III/4

KEY WORDS: LIDAR, Detection, Three-dimensional, Building, Modelling, Automation, Reconstruction, Accuracy

ABSTRACT:

Following the development of 3D data acquisition techniques like digital photogrammetry, remote sensing and airborne laser scanning, the automatic building modelling is still a challenging task. Indeed, it takes a growing place in many research fields like 3D city modelling, cartographic analysis, urban planning, visualization and Geographic Information Systems (GIS) database construction. Generally, two types of approaches characterize the automatic building reconstruction. The first one is a model-driven approach. This parametric modelling approach consists of searching the most appropriate model among basic building models contained in a models library. The second one is a data-driven approach, also called non-parametric modelling approach. This technique attempts to model a primitive or a complex building by using series of more or less complex operations. It allows the generation of a model without belonging to a specific library. After an extensive state of the art, this paper confronts methods belonging to each type of automatic building construction approach. Based on a concrete experiment, the essential points characterizing each approach, including their concept and the obtained model characteristics are analysed. It is of great interest to study these issues, since not many of such investigations have been made before and have delivered the useful formulas. Consequently, improvements are introduced and have led to develop a new method for each approach. After comparison of the accuracy of these two methods and the characteristics of the resulting models, some solutions and recommendations are proposed.

1. INTRODUCTION

Generally, in different surveying domains like city modelling, database construction of GIS, cartographic analysis or urban planning, 3D building modelling becomes an important common technique. The majority of available building modelling methods requires manual intervention. In the last ten years the research of automatic methods has grown by reason of the informatics progress. Currently, many automatic building modelling approaches have been carried out, but some limitations are inherent to each one of them. So the automation of a building modelling process still remains a challenging task.

The presented study focuses on automatic building modelling using airborne laser scanner data. After the acquisition of 3D city laser data, the 3D point cloud has to be segmented into three main classes: terrain, vegetation and buildings. Several approaches have been developed to carry out the Lidar data segmentation automatically like the approaches suggested by (Tarsha-kurdi *et al*., 2007a; Tóvári and Vögtle, 2004). Once the building point clouds are extracted, the automatic building modelling procedures can begin.

The definition of building modelling in the laser scanning domain is the construction of a 3D model of buildings composed of planes and edges extracted from the building point cloud. According to the literature (Maas and Vosselman, 1999), there are two principal approaches of building modelling starting from airborne laser scanner data: the model-driven or parametric approach and the data-driven or non-parametric approach. This study aims to compare both techniques in order to be able to conceive an optimized technique.

2. RELATED WORK

As mentioned above, two kinds of approaches characterize automatic building reconstruction: the model-driven approach and the data-driven modelling approach.

The model-driven approach makes beforehand a selection between the primitive and the complex buildings. So for a point cloud acquired on a primitive building the approach consists of searching the most appropriate model among basic building shapes contained in a models library. Then the most probable parameter values are calculated and assigned to the parameters of the selected model. Several solutions based on this approach have been developed. For instance, (Maas and Vosselman, 1999; Maas, 1999) propose a method based on the analysis of invariant or static moments of building point clouds. Another method has been presented by (Weidner and Förstner, 1995; Weidner, 1996) and concerns the automatic extraction of model-driven and prismatic building models from dense digital elevation models generated by photogrammetric techniques or airborne laser scanning. In addition to the last two methods, (Schwalbe *et al*., 2005) developed a method based on the use of the building vertical profiles. At last, several authors, e.g. (Haala *et al*., 1998; Brenner and Haala, 1998) suggest the introduction of the DSM (Digital Surface Model) surface normals.

In the case of complex buildings, several authors suggest to segment the complex building point cloud into primitives using ground plans or another type of additional data (Brenner and Haala, 1998; Haala *et al*., 1998; Schwalbe *et al*., 2005; Park *et al*., 2006). Then, every building can be handled independently.

The data-driven approach models a building regardless of its form. So it attempts to model a primitive or a complex building by using the point cloud as initial data. Thus, after series of more or less complex operations, this technique allows generating models without belonging to a specific library. Many works can be cited and classified in four categories:

Methods using the 3D Hough-transform: (Vosselman and Dijkman, 2001; Oda *et al*., 2004) use it for detecting the roof planes; (Hofmann, 2004) introduces it for the analysis of tin-structure parameter spaces.

Methods using the RANdom SAmpling Consensus algorithm (RANSAC): For instance (Ameri and Fritsch, 2000; Brenner, 2000) use it for detecting the roof planes. So planes are accepted or rejected based on a list of rules which present the possible relationships between planes and ground plan edges.

Methods using a region growing algorithm: (Alharthy and Bethel, 2004; Elaksher and Bethel, 2002) developed an algorithm that gathers together all pixels fitting a plane in raster data; (Rottensteiner, 2003) extracts roof planes using seed regions and applies a region growing algorithm in a regularized DSM. Then, the homogeneity relationships between the neighbour points are evaluated by calculating the point normals.

Methods using Douglas-Peucker technique: (Wang *et al*., 2006; Tarsha-kurdi *et al*., 2007a) propose to construct the facade models before studying the roof construction; so the resulting 3D building model is firstly constructed with plane roofs. They use Douglas-Peucker technique to segment the building contour polygon according to its facades.

Complementary data like ground plans are sometimes used in addition to the building point cloud (Ruijin, 2004; Vosselman and Dijkman, 2001; Haala *et al*., 1998).

Before comparing concretely each approach, it is necessary to detail them in the next paragraph.

3. MODEL-DRIVEN APPROACH FOR BUILDING RECONSTRUCTION

Since the model-driven approach searches the most appropriate model among basic building models contained in a library, it requires the extraction of the primitive buildings composing the area under study. The primitive building is a simple building which can be described by a set of parameters. The values of the parameters are calculated before constructing the 3D model. In the next step, the building roof details can be determined and then constructed.

As already mentioned, in the case of a complex building, many authors suggest to segment it into primitive buildings using the building ground plan. This decomposition can also be done through the building roof breaklines derived from the DSM. Figure 1 shows a successful and a failed result of this decomposition when applied on two different point cloud densities. The reliability of this latter method depends not only on the point cloud density, but also on the roof plane surfaces, the quantity and the dimensions of details occurring on the building roof.

In general, the parameters describing a building in a model-driven approach are of two types: parameters defining the building ground polygon (footprint parameters) and parameters describing the building space (space parameters). The first set of parameters defines the building footprint position, orientation and dimensions in addition to the building facades equations. The second set of parameters defines the building roof plane equations.

Based on these two parameter sets, the 3D building model can be constructed.

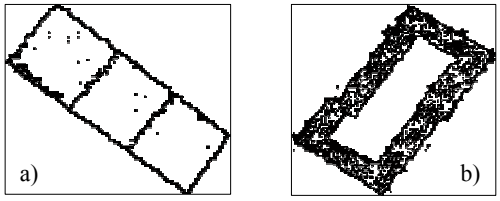


Figure 1. Test point cloud: a) Successful building breaklines extraction from a DSM (point cloud density: 7 points/m²); b) Failed building breaklines extraction from a DSM (point cloud density: 1.3 points/m²).

4. DATA-DRIVEN APPROACH FOR BUILDING RECONSTRUCTION

The data-driven approach usually assumes that a building is a polyhedral model. It attempts to model an unspecified building without segmenting it into primitives. It analyzes the building point cloud as a unity, without relating it to a set of parameters. This modelling category proposes series of operations allowing initially to generate an unspecified 3D building model starting mainly from the laser data. In spite of the probable risks of obtaining deformed models, it remains the only approach which treats the general case of unspecified building, i.e. as well the case of a complex building as the case of buildings blocks. Regardless of the used methods, the modelling is composed of two stages: building roofs modelling and facades modelling.

Concerning the building roof modelling, different methods have been evoked previously to detect the roof planes, like the RANSAC technique, the 3D Hough-transform and the region growing algorithms. These methods use sometimes complementary data in addition to the building point cloud, either to improve the plane roof detection or segmentation, or to improve the 3D building model quality. The next step is the determination of the neighbourhood relationship between the building roof planes. That is why (Ameri and Fritsch, 2000) propose to use the Voronoi diagram. Then, according to (Rottensteiner and Briese, 2003), the mutual relations between every two neighbour roof planes have to be determined (intersection, step edge or intersection and step edge together).

For the purpose of building facades modelling, two possibilities exist. The building contour polygon has to be detected either before segmenting the roof in planes, or after the building roof segmentation. In the first case, it is necessary to use line generalization algorithms which allow simplifying or segmenting the building contour polygon according to its facades like for instance the Douglas-Peucker technique. In the second case, the building contour polygon is segmented automatically following to the roof segmentation. The difference between these two cases is that in the first one, one facade is presented by several vertical planes according to the number of their adjacent roof planes. Whereas in the second case, one facade is presented by only one plane, under the assumption that the facade was previously well filtered (noise attenuation).

5. MODEL-DRIVEN AGAINST DATA-DRIVEN APPROACH: EXAMPLE OF BUILDINGS IN STRASBOURG CITY

The aim of the following example is to compare the accuracy and the quality of the 3D building models generated by the two approaches. For testing a model-driven approach, the method based on the analysis of invariant or static moments as initiated by (Maas and Vosselman, 1999; Maas, 1999) has been adapted and applied. For applying a data-driven approach, a method based on the Douglas-Peucker algorithm and the RANSAC technique has been extended and developed as initiated by (Ameri and Fritsch, 2000; Tarsha-kurdi *et al*., 2007a).

The data used is a point cloud located along the Rhine river, in the "Bord de Rhin" quarter of Strasbourg city (Fig.2). As it will be seen later, the data-driven approach is only applicable in the case of primitive buildings. That is why, the test area has been chosen in order to contain primitive buildings with gable roofs and some trees. The point cloud density is 1.3 point/m². Only the first pulse is available.

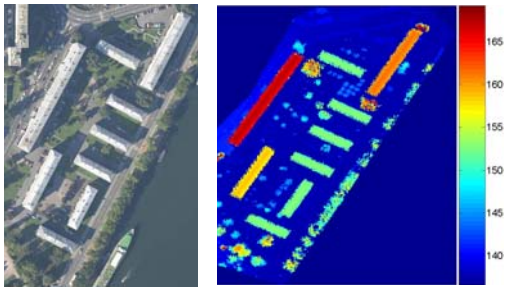


Figure 2. Aerial image and DSM of the sample used

5.1 Model-driven approach

The use of the analysis of static moments of a building point cloud implies two main processing steps. Firstly, the building point cloud has to be projected onto a horizontal plane in order to calculate the building orientation and the footprint parameters in 2D (Fig.3). Then, the process is continued in order to provide the 3D parameters of the building. After projection, the whole set of "new" points presents almost the building footprint form. The new points can be considered as infinitely small elements. Therefore, the application of the static moment equations on the new point cloud allows calculating the geometric elements of the building footprint, like the gravity centre of the building footprint (Equation 1) and the principal axes orientation (Equation 2). Figure 3b shows the coordinates system occurring in the calculation. Equation (3) presents the transformation from the original coordinates system OXY to the building footprint principal axes system O' X'Y'.

$$\overline{X} = \frac{\sum_{i=1}^{n} X_i}{n} \qquad \overline{Y} = \frac{\sum_{i=1}^{n} Y_i}{n} \tag{1}$$

Where n: point number; Xi and Yi: point cloud abscissas and ordinates in OXY.

$$\theta = \frac{1}{2}\arctan \frac{2\sum_{i=1}^{n}(X_i - \overline{X})(Y_i - \overline{Y})}{\sum_{i=1}^{n}(X_i - \overline{X})^2 - \sum_{i=1}^{n}(Y_i - \overline{Y})^2} \tag{2}$$

Where θ: rotation angle between original coordinates system OXY and building footprint principal axes O'X'Y'.

$$X_i' = (X_i - \overline{X})\cos(\frac{\pi}{2} - \theta) - (Y_i - \overline{Y})\sin(\frac{\pi}{2} - \theta)$$
$$Y_i' = (X_i - \overline{X})\sin(\frac{\pi}{2} - \theta) + (Y_i - \overline{Y})\cos(\frac{\pi}{2} - \theta) \tag{3}$$

If the building footprint form is known, its dimensions can also be calculated. Hence when the building footprint is rectangular, the equations (4) are used to calculate the length L_X and width L_Y of the building footprint.

$$L_x = \sqrt{\frac{12\sum_{i=1}^{n} X_i'^2}{n}} \qquad L_y = \sqrt{\frac{12\sum_{i=1}^{n} Y_i'^2}{n}} \tag{4}$$

The whole parameters can be calculated under the condition that the building point cloud has a homogeneous density.

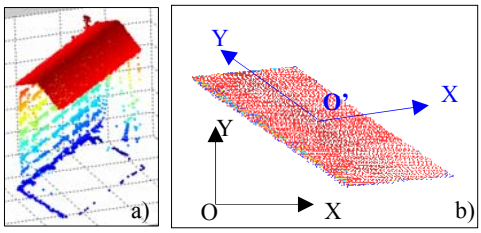


Figure 3. a) Visualization of the 3D building point cloud; b) Projection of the building point cloud on the horizontal plane of the original coordinates system OXY (O'X'Y': Principal axes of the coordinates system related to the building footprint; O': Building footprint gravity centre)

In this context, this method has been improved in order to be appropriate for irregular point distributions and for different densities of point clouds. Indeed, by creating a new point cloud generated from the building DSM, a homogeneous point density can be reached even if the data are slightly smoothed by the interpolation. So, this procedure is used to decrease the errors made in the calculation of the coordinates of the building gravity centre and of the building footprint dimensions. Moreover, the second new idea is to use the histogram of the original point cloud to extract especially the roof points and to continue the parameter calculation.

These additional operations improve on the one hand the building type determination, because it allows eliminating the "noisy" points (roof details, ground points). On the other hand, it enables to increase the precision of the determination of the building parameter values. Indeed, as illustrated in Fig.4a, the histogram analysis of the Z values occurring in a primitive building point cloud shows the possibility to divide the building point cloud into four parts: surrounding building points (I), facade points (II), roof points (III) and roof detail points (IV).

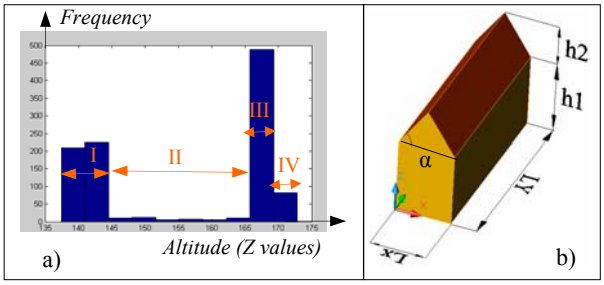


Figure 4. a) Histogram of the Z values of a building point cloud b) Primitive building parameters (where I: Surrounding building points; II: Building facade points; III: Building roof points; IV: Building roof detail points)

Once the building footprint principal axes are calculated and the building roof points are detected, the roof type can be determined and the building space parameters can also be calculated (h1, h2 and α). Fig.4b shows those parameters referred to a primitive building.
At this stage, the total 3D building model (Fig.5) can be constructed.

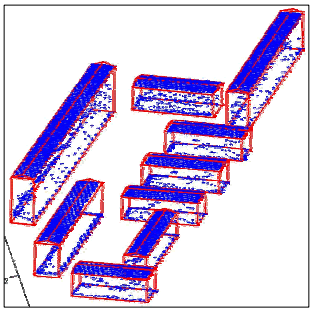


Figure 5. Superimposition of the building point clouds over the 3D parametric obtained models

Regarding the computing time, it can be noted that the model-driven approach is very fast, because it calculates only the values needed for defining the building parameters. On the other hand, it is limited by the fact that it is only reliable for primitive buildings having known footprints.

5.2 Data-driven approach

The data-driven method applied to the sample consists of two steps: building facades modelling and building roof modelling. After detection of the building contour polygon, this polygon is segmented according the building facades using the Douglas-Peucker algorithm (Douglas and Peucker, 1973). Then the equations of the facade planes are fitted based on the least square principle and the intersection between every two adjacent planes is calculated. These processing steps lead to the construction of 3D building models with plane roofs (Tarsha-kurdi *et al*., 2007a) as shown in Fig.6a.

The second step focuses on the modelling of the roofs. So, an extended RANSAC technique is applied to detect the roof planes (Tarsha-kurdi *et al*., 2007b). Then, the neighbourhood relationship between roof planes is formalized using the neighbourhood matrix which represents the Voronoi diagram of the label image. Finally, the mutual relations between every two adjacent roof planes have to be determined. At the same time, the intersections between the roof planes (adjacent to building contour) and the building facades are calculated. The two last steps allow constructing the total 3D building models (Fig. 6b).

The new idea introduced in this data-driven method consists in combining the Douglas-Peucker and the RANSAC algorithms. Indeed, it allows modelling the facades and the roof separately and helps to decrease the deformations quantity of the final building model. Moreover, the extended RANSAC technique allows harmonizing the mathematical aspect of the classical RANSAC algorithm with the geometry of a roof (Tarsha-kurdi *et al*., 2007b).

In the last paragraphs, one method of each approach types of automatic building modelling have been presented and applied concretely on a data set. In order to compare them, several items will be analysed like the 3D accuracy, the characteristics of obtained models, the advantages and disadvantages of each one.

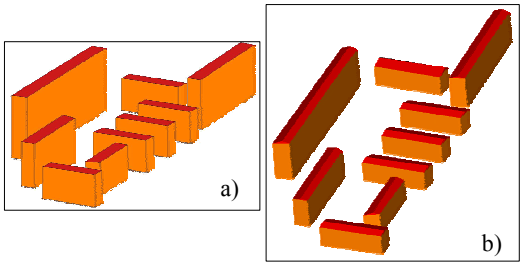


Figure 6. Application result of data-driven approach.
a) 3D building models with plane roofs
b) total 3D building models

5.3 Comparison of the accuracy of both approaches

In order to compare the two approaches regarding the accuracy of their resulting models, a 3D reference model of a set of nine building is created by semi-automatic photogrammetric digitizing. Its accuracy is about ±20 cm in X, Y, Z. All the buildings used in the sample present gabled roofs (Fig.2). The parameters extracted from these reference buildings are the footprint parameters (Lx and Ly) and the space parameters (h1, h2 and α).

Thus, three sets of parameters are calculated for each building: one set for the reference model, one set for the model-driven approach and at last one set for the data-driven approach. Then, the differences observed in the building parameters are compared with the reference building parameters. Also the mean of the parameter differences are calculated in addition to their standard deviations (Table 1).

Approach \ Parameters		Footprint parameters		Space parameters		
		ΔLx (m)	ΔLy (m)	Δh1 (m)	Δh2 (m)	Δα°
data-driven	Mean	0,12	0,04	0,42	-0,29	-2,75
	σ	0,38	0,97	0,97	0,37	3,42
model-driven	Mean	1,24	1,47	0,23	-0,06	-2,6
	σ	0,65	0,52	0,97	0,34	3,08

Table 1. Mean parameter differences and standard deviation values obtained with each approach.

Table 1 shows that the footprints parameters in the data-driven approach are more reliable than those in the model-driven approach. Indeed the mean values confirm this tendency since the footprint parameters are closer to the real values ($0.12 < 1.24$ and $0.04 < 1.47$). The slight difference observed between the total standard deviations ($1.04 > 0.83$) is not significant enough to contradict previous remark. Moreover, it can be observed that the space parameters in the model-driven approach are slightly more accurate than those in the data-driven approach. The mean values are systematically lower. Also, the standard deviations confirm this conclusion.

Another interesting way to compare both approaches consists in listing the techniques used in 3D building models calculation. Table 2 classifies the techniques used in 3D building models calculation into two families. The first family contains techniques based on a pure mathematical principle (PMP), e.g. calculation and intersection of plane equations. The second family is based on image processing principles (IPP), e.g. histogram analysis. Additionally, Table 2 contains signs (+,–)

for summarizing the accuracy results of Table 1. For example, the footprint parameters are more accurate (sign +) in the data-driven approach than in the model-driven approach (sign –).

	Footprint parameters	Space parameters
data-driven	(+) Douglas-Peucker (**IPP**)	(–) Calculation and intersection of plane equations (**PMP**)
model-driven	(–) Analysis of invariant moments (**PMP**)	(+) Histogram analysis (**IPP**)

Table 2. Techniques used in 3D building models calculation (**PMP:** Pure Mathematical Principle; **IPP**: Image Processing Principle)

It can be observed that the accuracy of the final calculated model is less related with the type of the modelling approach (model-driven or data-driven) than with the nature of the operations composing each approach. So, regardless of the modelling approach type, when pure mathematical principles are used on the whole cloud (such as the analysis of static moments, or calculation and intersection of plane equations) results will be less accurate than when image processing principles are used (like for example Douglas-Peucker technique and histograms analysis).

5.4 Comparison of the characteristics of the resulting models

The advantages of a model-driven approach are that it provides geometrical models without visual (geometric) deformations, because it is based on the calculation of values of parameters. Thus, it is important to underline the high computing speed gained by these approaches in comparison with data-driven ones. The only errors in this type of model can come from the calculation of the building parameters values. Moreover, the failure probability in choosing a building type among the model library is limited, because the number of thresholds needed for calculation is very small.

The major disadvantage of a model-driven approach is to be dependent on the buildings types which are available in the building library.

The essential advantage of data-driven approach is that the case of an unspecified building is studied, i.e. as well the case of a complex building as the case of building blocks. On the other hand, among several disadvantages, the considerable visual deformations produced by the data-driven approach of a complex building can be cited. Indeed, during the calculation of the building roof edges, the use of plane intersections presents real risks of misconstruction. The causes of possible misconstructions are listed below:

- In the general case, building modelling is based on the assumption that a building is composed of planes and lines (edges of the building). It is well known that building surfaces do not present planes in a mathematical sense. Thus, an equation of plane deduced from the points distributed on roof planes does not perfectly characterize a roof, but only approximates it.
- The point coordinates can contain errors (position inaccuracy, artefacts, and multi ways).
- The irregularity of the point distribution on the building roof can increase the errors. Indeed, inside the same surface of a building roof, it is possible to find variable points distributions.
- The point cloud density influences drastically the level of deformations. If the density increases, the quantity of deformations decreases and vice versa.
- The point cloud interpolation can generate a positive effect and a negative one simultaneously. On the one hand, it allows to eliminate the building facade points, to obtain a regular point grid and to smooth a large quantity of errors (bilinear or bicubic interpolation). In addition, it generates undesirable effects if the sampling interval chosen for the grid cells does not correspond to the density of points or if some parts of the point cloud are empty.
- The noise and the building roof details can be considered as obstacles because of the intolerable deformations which they could generate in the final model.

Several solutions were proposed in the literature in order to try solving this problem of deformations occurring in the data-driven approach:

- Application of geometrical constraints in the parallelism and the orthogonal level for calculating the roof plane edges (Haala and Brenner, 1997).
- Determination of the principal axis of each roof plane to observe the symmetry conditions (Elaksher and Bethel, 2002).
- Reiteration of calculation to eliminate the points having large residues (Rottensteiner and Briese, 2003).
- Preprocessing of the point cloud or of the DSM in order to eliminate the noise and to obtain homogeneous data (Alharthy and Bethel, 2004).
- Application of mathematical morphology operations which allow improving the forms of the obtained plan segments (Rottensteiner, 2003).
- Application of filters to eliminate the undesirable points before beginning the calculation (Haala and Brenner, 1997).

In some cases, each one of the propositions listed above improves the obtained models. But at the same time, in other cases, it can produce errors in the form obtained for the reconstructed building. In the algorithm we developed for the experiment, no specific improvement operations of the last list are applied. Two reasons explain this decision: The first one is the wish to avoid the negative consequences of some proposed improvements; the second one is that the buildings used in this experiment are relatively simple, so the foreseeable deformations are negligible. Additional tests need to be carried out in order to generalise our remarks to a greater set of building types.

6. CONCLUSION

This paper achieves a comparison between the data-driven and the model-driven approaches. Several 3D buildings resulting from the application of each approach are compared and confronted with a reference 3D model. Thus, the accuracy and the reliability of each modelling approach have been evaluated. Furthermore, one new method based on improved modelling principles has been proposed for each approach. These

improvements increase the potential of the modelling method and the final model accuracy.

In summary, the model-driven approach considers the entire building point cloud. Its main advantages are that it provides in a very fast way geometrical models without visual deformations. On the other hand, the data-driven approach tends to simulate each part of the building point cloud for obtaining the nearest or the more reliable polyhedral model. Its main disadvantages are that it provides models with visual deformations and it needs more processing time. Nevertheless, if the point cloud is characterized by a homogenous distribution and if its density is in relation to the elements dimension which are relevant to be extracted (building roof, roof details), then the obtained data-driven model will be very faithful to the original building.

Concerning the model precision, it can not be said in the general case that the data-driven or the model-driven approach is more accurate than the other. Indeed, the model accuracy is more related to the techniques used in building modelling approaches than to the approach type. Nevertheless, in spite of the probable risks of obtaining deformed models, the data-driven approach remains the only approach which treats the general case of building of an unspecified form. Finally, further experiments should confirm these conclusions and improve the data-driven approach for generating a realistic and accurate 3D model.

REFERENCES

References from Journals:

Ameri, B., Fritsch, D., 2000. Automatic 3D building reconstruction using plane-roof structures, ASPRS, Washington DC.

Brenner, C., Haala, N., 1998. Fast reality production of Virtual Reality City Models. IAP, Vol. 32, Part 4.

Douglas., D.H. Peucker., T.K. 1973. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. The Canadian Cartographer 10 (2), 112 –122. 1973.

Maas, H.-G., Vosselman, G., 1999. Two algorithms for extracting building models from raw laser altimetry data. ISPRS Journal of Photogrammetry & Remote Sensing Vol. 54, No. 2/3, pp. 153-163

Rottensteiner, F., 2003. Automatic generation of high-quality building models from Lidar data. IEEE CG&A 23(6), pp. 42-51.

Wang, Y. Weinacker, H. Koch, B., 2006. Automatic non-ground objects extraction based on multi-returned Lidar data. Photogrammetrie Fernerkundung Geoinformation (PFG). Jahrgang 2006, heft 2. ISSN: 1432-8364.

Weidner, U., Forstner, W., 1995. Towards automatic building extraction from high resolution digital elevation models. ISPRS Journal, 50(4):38--49.

References from Other Literature:

Alharthy, A., Bethel, J., 2004. Detailed building reconstruction from airborne laser data using a moving surface method. Arch. Photogrammetry and Remote Sensing, Vol. XXXV, part B3.

Brenner, C., 2000. Towards fully automatic generation of city models. Int. Arch. Photogramm. Remote Sensing, vol. 32, Part 3. Amsterdam, pp. 85–92.

Elaksher, A. F., Bethel, J. S., 2002. Reconstructing 3D buildings from Lidar data. Int. Arch. Photogrammetry and Remote Sensing, Vol. XXXIV, part 3A/B, pp102-107.

Haala, N., Brenner, C., Anders, K.-H., 1998. 3D urban GIS from laser altimeter and 2D map data. Int. Arch. Photogrammetry and Remote Sensing, Vol. 32, Part 3, pp. 339-346.

Haala, N., Brenner, K., 1997. Generation of 3D city models from airborne laser scanning data. Proceedings EARSEL Workshop on Lidar remote sensing on land and sea, Tallinn/Estonia.

Hofmann, A. D., 2004. Analysis of tin-structure parameter spaces in airborne laser scanner data for 3D building model generation. Int. Arch. Photogrammetry and Remote Sensing, Vol. XXXV, part B3.

Maas, H.-G., 1999. The potential of height texture measures for the segmentation of airborne laserscanner data. Proceedings of the 4th International Airborne Remote Sensing Conference, Ottawa, 21.-24.6.99, Vol. I, pp. 154-161.

Oda, K., Takano, T., Doihara, T., Shibasaki, R., 2004. Automatic building extraction and 3-d city modeling from Lidar data based on Hough transformation. Int. Arch. Photogrammetry and Remote Sensing, Vol. XXXV, part B3.

Park, J., Lee, I., Choi, Y., Lee, Y-J, 2006. Automatic extraction of large complex buildings using Lidar data and digital maps. Workshop ISPRS. Com III, Photogrammetric Computer Vision PCV Bonn, Germany 20 – 22 September 2006.

Rottensteiner, F., Briese, Ch., 2003. Automatic generation of building models from Lidar data and the integration of aerial image. Int. Arch. Photogrammetry and Remote Sensing, Vol. XXXIV.

Ruijin, M., 2004. Building model reconstruction from Lidar data and aerial photographs. Doctoral dissertation, Ohio State University. USA.

Schwalbe, E., Maas, H-G., Seidel, F., 2005. 3D building generation from airborne laser scanner data using 2D GIS data and orthogonal point cloud projections. Workshop ISPRS. Laser scanning. Enschede, the Netherlands, September 12-14, 2005.

Tarsha-Kurdi, F., Landes, T., Grussenmeyer, P., 2007a. Joint combination of point cloud and DSM for 3D building reconstruction using airborne laser scanner data. Urban remote sensing joint event URBAN/URS 2007. 11-13 April Paris.

Tarsha-Kurdi, F., Landes, T., Grussenmeyer, P., 2007b. Hough-transform and extended RANSAC algorithms for automatic detection of 3D building roof planes from Lidar data. ISPRS Workshop on Laser Scanning and SilviLaser 2007. Espoo, September 12-14, 2007, Finland.

Tóvári, D., Vögtle, T., 2004. Classification methods for 3D objects in laserscanning data. Int. Arch. Photogrammetry and Remote Sensing, Vol. XXXV, part B3.

Vosselman, G., Dijkman, S., 2001. 3D building model reconstruction from point clouds and ground plans. Int. Arch. Photogrammetry and Remote Sensing, XXXIV-3/W4:37-43.

Weidner, U., 1996. An Approach to building extraction from digital surface models. 18th Workshop of the ISPRS. Comm. III, WG 2, building detection from a single Image. Vienna, Austria, 1996, pp. 924-929. 43.

AUTOMATIC DETECTION OF ZENITH DIRECTION IN 3D POINT CLOUDS OF BUILT-UP AREAS

Wolfgang von Hansen

FGAN-FOM, Gutleuthausstr. 1, 76275 Ettlingen, Germany
wvhansen@fom.fgan.de

KEY WORDS: Automation, Orientation, Algorithms, Urban, LIDAR, Point Cloud

ABSTRACT:

3D modeling of built-up areas has become an important issue during the past years. One technique for model creation is LIDAR. Terrestrial laser systems have become popular, but due to limited range and occlusions, many scanner positions are needed for data acquisition, requiring co-registration and fusion of the resulting models.
Two datasets can be co-registered by translation and rotation which are independent of each other. In this paper, we focus on the rotation and propose a method for the automatic detection of the zenith direction and thereby providing the ground plane. This yields an implicit surface for each dataset so that only one additional surface correspondence is needed to solve for the unknown rotation. The method exploits the geometric characteristics specific to built-up areas: Many vertical walls exist as well as a roughly horizontal ground surface. Results are presented for tests on a total of 26 datasets.

1 INTRODUCTION

1.1 Overview

3D modeling of built-up areas has become an important issue during the past years. One technique for model creation is laser scanning (LIDAR), which can be subdivided into airborne and terrestrial platforms. While airborne laser scanners cover large areas, their geometric resolution is low and only roof surfaces are well captured.

Terrestrial laser systems overcome these disadvantages, but due to limited range and occlusions, many scanner positions are needed for data acquisition, requiring co-registration and fusion of the resulting models. Many approaches exist for an automatic coarse registration of the scan data. (Akca, 2003, Bae, 2006, Dold, 2005, Dold and Brenner, 2006, von Hansen, 2006, He et al., 2005, Liu and Hirzinger, 2005, Rabbani and van den Heuvel, 2005, Ripperda and Brenner, 2005, Wendt, 2004) The fine registration is usually solved through the ICP *(iterative closest point)* algorithm (Besl and McKay, 1992, Rusinkiewicz and Levoy, 2001).

Segmentation techniques include clustering based on local surface normal analysis (Bretar and Roux, 2005, Liu and Hirzinger, 2005), region growing using scan geometry and point neighborhoods (Dold and Brenner, 2004), and a split-and-merge scheme applying an octree structure (Wang and Tseng, 2004). The objects are often represented by planar elements recovered through RANSAC schemes (Bretar and Roux, 2005) or least squares adjustment (Wang and Tseng, 2004).

Two datasets can be co-registered by one translation and one rotation which both are independent of each other. Three corresponding surfaces are sufficient for translation and two for rotation. However, as there is only a small overlap between two models and many surfaces are only partially seen due to occlusions, it is not easy to determine correct correspondences.

In this paper, we focus on the rotation and propose a method for the automatic detection of the zenith direction and thereby the ground plane. As this yields an implicit surface for each dataset, only one pair of corresponding surfaces is needed to solve for the unknown rotation between two datasets. As this can be done independently for each dataset, this is an important aid for the coarse registration process. Our method exploits geometric characteristics specific to built-up areas: Many vertical walls exist as well as a large and roughly horizontal ground surface.

1.2 Methodology

According to (Dold and Brenner, 2004), both components of registration – rotation and translation – can be carried out independently. When planar surfaces have been extracted from the point clouds, two corresponding planes define the rotation about all three axes and three corresponding planes define the translation in space. In fact, three correspondences would solve the complete problem, but if there is only a little overlap between two datasets, many of the possible combinations are false.

Therefore, a bottom up approach seems more feasible. If a part of the problem can be solved more easily, it leads to additional information that can be used in following steps. This paper proposes an automatic orientation of a single point cloud from outdoor scenes in built-up areas, so that the zenith direction is aligned with the z-axis, i. e. is pointing upwards. This is done separately for each single point cloud – therefore no correspondence search is necessary at this stage. After orientation, only a single plane correspondence is needed to solve for rotation.

In a preprocessing step, the raw point cloud is converted into a set of plane elements. Then, a RANSAC based scheme is applied to their normal vectors, partitioning them into ground, walls and other elements. This allows a robust computation of the zenith direction. As by-product, the segmentation into these three classes is returned.

We have not found any reference to such single dataset orientations during our literature research.

1.3 Geometrical axioms

Our approach is based on some simple axioms on planar structures that are valid for built-up areas. These axioms give constraints that allow to segment planes into ground, building walls and other objects:

1. The ground surface is large compared to other surfaces.

Ambiguity interval	53.5 m
Resolution range	16 Bit 1 mm/lsb
Range noise 10 m	1.3–3.0 mm rms
Range noise 25 m	3.0–9.0 mm rms
Laser output power	23 mW (red)
Beam divergence	0.22 mrad
Field of view vertical	310°
Field of view horizontal	360°
Resolution vertical	0.018°
Resolution horizontal	0.01°
Accuracy	0.02°rms
Number of pixels in dataset	200 million

Table 1: Specifications of the Z+F Imager 5003 laser scanner.

2. The normal vector of the ground surface is approximately – but not exactly – orthogonal to the horizontal plane and pointing roughly to the zenith direction.

3. Building walls have a normal vector that lies exactly in the horizontal plane.

4. Roof surfaces are inclined and point upwards.

5. Other surfaces, like e. g. trees or small structures, have normal vectors pointing to random directions.

The distinction between roofs and other surfaces is not a simple decision, as it requires statistical knowledge about the surroundings of a position. Here, we will not discriminate between these two types and regard roofs as other surfaces.

The objective to find the zenith direction is replaced by the detection of the horizontal plane. From the axioms we see two complementary hints at the correct horizontal plane: First, there are many normal vectors lying in this plane. Their orientation/ direction is not important. Every building wall – which we assume vertical – will contribute to this criterion. Second, the normal vector of the ground already is an estimate for the zenith direction. But we have to acknowledge, that these estimates may contain systematic errors because the terrain may be sloped even in built-up areas. Even the steepest roads typically have an inclination of less than 20% which is roughly 11°. This can be used as cut-off for ground surfaces. A terrestrial system is usually placed below the roof, so that flat roofs normally will not appear in the datasets and therefore can not be confused with the ground.

1.4 Laser scanner and test data

The laser scanner used for data acquisition is a Zoller+Fröhlich Imager 5003 – some of its technical data is given in Tab. 1. It has an omnidirectional field of view, so that there is no tilt mechanism necessary to point the laser scanner to the object of interest. Instead, the scanner is leveled prior to use by means of a circular level. This way, the true zenith direction is contained in the data and can be utilized for the evaluation of our results in the sense that gross errors can be detected. It should be emphasized, that this information is *not* exploited by the algorithm. The datasets that we dispose of are 26 overlapping outdoor scans from a village scene.

2 METHODOLOGY

2.1 Generation of planar surface elements

As a preprocessing step we estimate locally delimited plane elements from the point cloud as proposed by (von Hansen, 2006).

Pos	planes	Pos	planes
1	6841	14	3510
2	5163	15	4955
3	3741	16	4072
4	5065	17	4982
5	3685	18	4360
6	2616	19	4995
7	2955	20	5503
8	4629	21	7078
9	5498	22	5184
10	4726	23	3290
11	7160	24	3567
12	4769	25	2859
13	5117	26	9971

Table 2: Number of planes extracted for each position.

The measured point cloud is split into a regular 3D raster and the plane that is supported by most of the points in each raster cell is computed in a robust way. Large object surfaces are thereby split into several coplanar elements. An alternate approach would be to try to reconstruct the complete surfaces, but the advantage of many small planes is the introduction of an implicit weighting factor: As large object planes will be divided into many plane elements, they will contribute a lot to the result.

We have implemented the generation of surface elements via the well known RANSAC strategy (Fischler and Bolles, 1981): From three randomly chosen, non collinear points the uniquely defined plane is computed. Then for all points, their distance to the plane is computed, counting those with a distance below a certain threshold as inliers. This procedure is repeated, finally returning the plane parameters for the plane with the largest inlier count. A plane can be represented by the Hesse normal form

$$\mathbf{n}^\top \mathbf{x} + d = 0 \tag{1}$$

where $\mathbf{n}$ is the plane's normal vector, d its distance to the origin and $\mathbf{x}$ the set of all points on the plane. Since we are only interested in the rotation, it is sufficient to keep just the normal vectors. The set of all normal vectors $\mathbf{n}_i$ of a dataset will be denoted as $\mathcal{N}$.

2.2 Zenith direction detection

We will present a two step approach to the determination of the zenith direction. The first is to determine an approximate zenith direction from axioms 1 and 2, that is to find a maximum number of parallel normal vectors. The second step is the determination of the exact zenith direction from axioms 2 and 3: Using the approximate zenith direction, the normal vectors of the walls are identified and from these the final solution is estimated.

2.2.1 Robust detection of approximate ground plane From axiom 1 we assert that the majority of the data is representing the ground surface and from axiom 2 we see that its normal vector already is a first estimate for the solution. The task therefore is to find a maximum number of parallel normal vectors. For fast computation, a RANSAC scheme is used.

A single normal vector $\mathbf{n}_i$ is randomly drawn from $\mathcal{N}$ and tested with all other vectors $\mathbf{n}_j \in \mathcal{N}$ for parallelity to generate the i-th inlier set

$$\mathbf{N}_i := \{\mathbf{n}_j \in \mathcal{N} : \mathbf{n}_i \parallel \mathbf{n}_j, j = 1 \dots |\mathcal{N}|\} \tag{2}$$

where $\parallel$ denotes parallelity of two vectors which is implemented as

$$\mathbf{a} \parallel \mathbf{b} :\Leftrightarrow \mathbf{a}^\top \mathbf{b} = \cos\alpha \overset{!}{\geq} \theta_\parallel \tag{3}$$

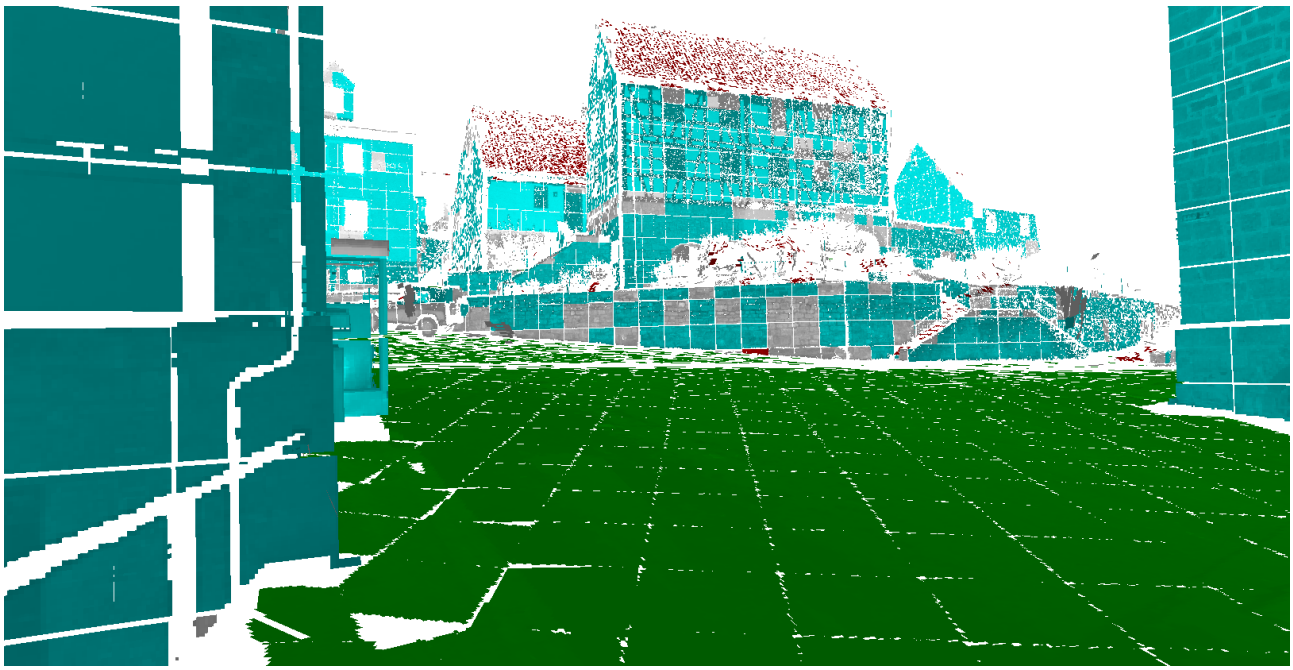


Figure 1: 3D model of position 22, showing the segmentation into different surface types.

where $\mathbf{a}$ and $\mathbf{b}$ are normalized vectors and $\theta_{\parallel}$ is a predefined threshold parameter for parallelity given by a maximum angle α between $\mathbf{a}$ and $\mathbf{b}$. The vector $\hat{\mathbf{n}}$ yielding the largest inlier set

$$\hat{\mathbf{N}} := \arg\max(|\mathbf{N}_i|) \tag{4}$$

is kept as the approximate zenith direction $\mathbf{z}_0$.

2.2.2 Estimation of exact zenith direction An approximate zenith direction $\mathbf{z}_0$ has already been determined as shown in the previous section. The ground surface usually is not horizontal but sloped, so that $\mathbf{z}_0$ is not guaranteed to be correct. A precise solution can be retrieved by taking into account that most walls are built vertically and therefore their normal vector can be utilized to compute a result (axiom 3).

The set of normal vectors of the walls $\mathcal{W}$ can be determined through axiom 1 as the set of all vectors roughly orthogonal to $\mathbf{z}_0$

$$\mathcal{W} := \{\mathbf{n}_i \in \mathcal{N} : \mathbf{n}_i \perp \mathbf{z}_0, i = 1 \ldots |\mathcal{N}|\} \tag{5}$$

where $\perp$ denotes orthogonality of two vectors which is implemented as

$$\mathbf{a} \perp \mathbf{b} :\Leftrightarrow |\mathbf{a}^\top \mathbf{b}| = |\cos\alpha| = |\sin\beta| \overset{!}{\leq} \theta_\perp \tag{6}$$

where $\mathbf{a}$ and $\mathbf{b}$ are normalized vectors and $\theta_\perp$ is a predefined threshold parameter for orthogonality given by a small angle β for the maximum difference to a right angle. $\theta_\perp$ should not be chosen too small because we have to account for sloped terrain.

All vectors $\mathbf{w}_i \in \mathcal{W}$ lie in the horizontal plane, typically pointing to different directions. The final zenith direction $\hat{\mathbf{z}}$ is orthogonal to all $\mathbf{w}_i$ and can be computed as the null space of the matrix $\mathbf{W}$ composed from all vectors of $\mathcal{W}$ via singular value decomposition.

There is a sign ambiguity in the returned vector as both zenith and nadir direction are valid solutions. This can easily be corrected by changing the sign if necessary to ensure that $\hat{\mathbf{z}}$ points into the same direction as $\mathbf{z}_0$.

2.2.3 Segmentation With the known zenith direction $\hat{\mathbf{z}}$, an additional result can be obtained without much further computation. After rotation of the dataset such that $\hat{\mathbf{z}}$ points upwards, all normal vectors are put into some classes based on the axioms and their inclination with respect to the horizontal plane. This can help in the construction of a 3D model from the data.

3 EXPERIMENTS AND RESULTS

The laser scanner is already aligned to the horizontal plane prior to operation so that the "true" zenith direction is already known, however this knowledge is *not* exploited by the algorithm. The advantage here is, that the correctness of a result can immediately be checked. As a by-product, it is possible to get an idea on the quality of the instrument's leveling which had been carried out only roughly using the circular level of the laser scanner.

All datasets have been converted to plane elements using a distance threshold of 3 cm. Tab. 2 shows the number of planes generated for each position. The different scene complexities are mainly due to the presence of natural objects as many planes are generated to describe e. g. the volume of a tree. Especially position 26 is located at the end of the village where several trees were in the range of the scanner.

An example for a resulting dataset is presented in Fig. 1, showing a view across a larger town square. The buildings at the far end are located on top of a small hill. The different classes have been color coded in the images. Green represents the ground, cyan marks building walls and the roofs are colored in red. Other surfaces are shown in light gray. Some of the wall elements are shown in gray which is due to a tight threshold and some surface roughness. The texture as acquired by the laser scanner is mapped onto the surfaces. The square pattern that can be seen on all object surfaces is a result of boundary effects from the spatial data partitioning into cubes.

The algorithm for the estimation of the zenith direction had been applied successfully to all datasets. This experimentally proves that the assumptions defined through the axioms were valid:

- The ground surface is represented by the largest group of parallel normal vectors. Other object planes like building facades may have a combined area that is larger, but their normals are pointing to different directions.
- In real data, there is enough variation in the wall's normal vectors to define the zenith direction. In a perfectly U-shaped street canyon, the wall's vectors would be collinear with an ambiguous solution $\mathbf{z}_0$ for the zenith direction. In this rare case, the approximate solution from the first step would still be available. In addition, it could also be used to check for such situations.

Some numerical results are given in Tab. 3. The column "Inliers" refers to the number of normal vectors of the ground plane with respect to the total number of normal vectors. This directly gives the percentage of the ground surface area in the scene. On the average, about 20% of structures in the acquired scenes are ground surface. In a certain way, this contradicts axiom 1, because on the average, only one fifth, and at maximum only one third, of the normal vectors are from the ground. But obviously, these small but systematic areas are sufficient to outweigh wall or roof areas. The worst case is position 11, where only one ninth of the scene is detected as ground.

The next column lists the number of iterations required for the RANSAC scheme. These numbers are computed from the inlier rate such that there is a 99.9% probability that one correct sample had been drawn. This is the reason for the strong correlation between inlier rate and number of iterations. The computation itself is sufficiently fast – less than 0.5 s on an ordinary computer – as the test for one iteration step amounts to n scalar products to check for parallelity where n is the number of planes as shown in Tab. 2.

The last column shows the angular deviation given in mrad between the scanner's zenith direction from the leveling and the estimated zenith direction. The values are distributed uniformly leading to the interpretation that both the leveling and the algorithm delivered good results. The maximum deviation is 41.2 mrad which amounts to 2.36°. Results of the segmentation are shown as different colors in Fig. 1 and Fig. 2. The cyan color denotes those wall elements that are used for the computation of the zenith direction. Other surfaces that have not been put into a specific class are colored in light gray and could be considered as *rejects*. The majority of the walls has been segmented correctly, while there are still some surface elements that have been rejected. This is acceptable as the threshold had been chosen rather tight here in order to include only good wall elements.

Red is assigned to roof surfaces, but there are also other surfaces assigned to this class. The main reason is that the allowed inclination for the roof surface is a large interval. As the absolute height above the ground or neighborhoods have not been considered, other upwards pointing surfaces have been marked as roofs as well. However, all real roof elements have been assigned to the correct class.

The ground is colored in green and has been segmented very well.

Pos	Inlier/%	Iterations	Angle/mrad
1	28.8	22	2.2
2	27.5	23	9.0
3	23.1	28	10.0
4	25.8	24	3.9
5	23.7	27	5.5
6	14.5	45	1.5
7	15.5	42	6.8
8	12.0	50	23.9
9	22.9	28	17.0
10	13.7	48	39.6
11	10.7	61	34.3
12	20.1	32	24.7
13	21.7	29	14.2
14	17.5	37	13.6
15	22.9	28	17.4
16	22.4	28	7.7
17	22.6	27	2.7
18	18.9	34	13.9
19	21.1	30	5.9
20	22.5	28	12.4
21	19.4	33	41.2
22	29.1	21	5.7
23	27.4	23	12.4
24	32.6	19	13.6
25	26.9	23	18.7
26	21.2	30	2.6

Table 3: Results zenith directions.

4 SUMMARY AND CONCLUSIONS

We have shown a new method that determines the zenith direction in a 3D point cloud of a built-up area. In a two step procedure, the raw point cloud is first transformed into a set of locally delimited object plane elements. An initial zenith direction is then recovered from the set of plane normal vectors through a RANSAC scheme that clusters parallel vectors in order to detect the ground plane. From this, the walls can be identified and used to estimate the final zenith direction as the vector orthogonal to their normal vectors.

Results have been shown for a total of 26 datasets. Being successful for all positions shows that the automatically recovered ground plane is correct. The small angular deviations are a combined quality measure for both algorithm and the scanner's circular level. No gross errors exist so that the automatically found walls can be deemed suitable for zenith direction estimation.

The main objective was to detect a global plane normal from a single 3D point cloud as one corresponding plane for the relative rotation of two datasets. For all of the available datasets, the proposed method was successful, making it useful for pre-rotation of point clouds in order to help coarse registration of multiple positions. Results also show that the leveling of the laser scanner had been carried out very precisely, so that the upwards direction of the dataset could already be regarded as zenith direction.

REFERENCES

Akca, D., 2003. Full automatic registration of laser scanner point clouds. In: Optical 3-D Measurements VI, pp. 330–337.

Bae, K.-H., 2006. Automated Registration of Unorganised Point Clouds from Terrestrial Laser Scanners. PhD thesis, Curtin University of Technology. URL: adt.curtin.edu.au/theses/available/adt-WCU20060921.094236/.

Besl, P. J. and McKay, N., 1992. A Method for Registration of 3-D Shapes. PAMI 14(2), pp. 239–256.

Bretar, F. and Roux, M., 2005. Hybrid Image Segmentation Using LIDAR 3D Planar Primitives. In: G. Vosselman and C. Brenner (eds), Laser scanning 2005, IAPRS, Vol. XXXVI-3/W19. URL: www.commission3.isprs.org/laserscanning2005/papers/072.pdf.

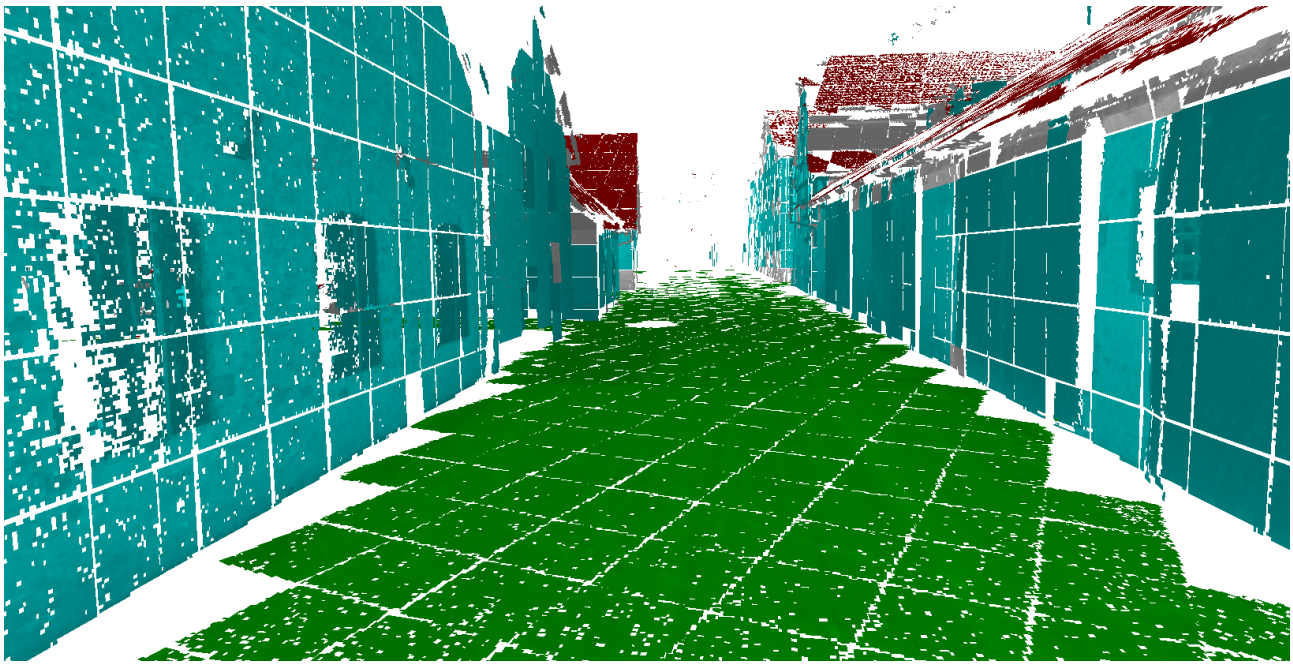


Figure 2: 3D model of position 25.

Dold, C., 2005. Extended Gaussian Images for the Registration of Terrestrial Data. In: G. Vosselman and C. Brenner (eds), Laser scanning 2005, IAPRS, Vol. XXXVI-3/W19. URL: www.commission3.isprs.org/laserscanning2005/papers/180.pdf.

Dold, C. and Brenner, C., 2004. Automatic Matching of Terrestrial Scan Data as a Basis for the Generation of Detailed 3D City Models. In: O. Altan (ed.), Proc. of the XXth ISPRS Congress, IAPRS, Vol. XXXV-B3. URL: www.isprs.org/istanbul2004/comm3/papers/429.pdf.

Dold, C. and Brenner, C., 2006. Registration of terrestrial laser scanning data using planar patches and image data. In: H.-G. Maas and D. Schneider (eds), Image Engineering and Vision Metrology, IAPRS, Vol. XXXVI Part 5. URL: www.isprs.org/commission5/proceedings06/paper/DOLD_637.pdf.

Fischler, M. A. and Bolles, R. C., 1981. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. Comm. of the ACM 24(6), pp. 381–395.

He, W., Ma, W. and Zha, H., 2005. Automatic registration of range images based on correspondence of complete plane patches. In: Proceedings of the Fifth International Conference on 3-D Digital Imaging and Modeling, pp. 470–475.

Liu, R. and Hirzinger, G., 2005. Marker-free Automatic Matching of Range Data. In: R. Reulke and U. Knauer (eds), Panoramic Photogrammetry Workshop, IAPRS, Vol. XXXVI-5/W8. URL: www.informatik.hu-berlin.de/sv/pr/PanoramicPhotogrammetryWorkshop2005/Paper/PanoWS_Berlin2005_Rui.pdf.

Rabbani, T. and van den Heuvel, F., 2005. Automatic point cloud registration using constrained search for corresponding objects. In: 7th Conference on Optical 3-D Measurements.

Ripperda, N. and Brenner, C., 2005. Marker-free Registration of Terrestrial Laser Scans Using the Normal Distribution Transform. In: S. El-Hakim, F. Remondino and L. Gonzo (eds), 3D-ARCH 2005, IAPRS, Vol. XXXVI-5/W17. URL: www.commission5.isprs.org/3darch05/pdf/33.pdf.

Rusinkiewicz, S. and Levoy, M., 2001. Efficient variants of the icp algorithm. In: Proceedings of the Third Intl. Conf. on 3D Digital Imaging and Modeling, pp. 142–152.

von Hansen, W., 2006. Robust automatic marker-free registration of terrestrial scan data. In: W. Förstner and R. Steffen (eds), Photogrammetric Computer Vision, IAPRS, Vol. XXXVI Part 3. URL: www.isprs.org/commission3/proceedings06/singlepapers/O_08.pdf.

Wang, M. and Tseng, Y.-H., 2004. LIDAR Data Segmentation and Classification Based on Octree Structure. In: O. Altan (ed.), Proc. of the XXth ISPRS Congress, IAPRS, Vol. XXXV-B3. URL: www.isprs.org/istanbul2004/comm3/papers/286.pdf.

Wendt, A., 2004. On the automation of the registration of point clouds using the metropolis algorithm. In: O. Altan (ed.), Proc. of the XXth ISPRS Congress, IAPRS, Vol. XXXV-B3. URL: www.isprs.org/istanbul2004/comm3/papers/250.pdf.

ADAPTING, SPLITTING AND MERGING CADASTRAL BOUNDARIES ACCORDING TO HOMOGENOUS LULC TYPES DERIVED FROM SPOT 5 DATA

D. Tiede [a] *, M. S. Moeller [b], S. Lang [a], D. Hoelbling [a]

[a] Z_GIS, Center for Geoinformatics, University Salzburg, Schillerstrasse 30, A 5020 Salzburg - (dirk.tiede, stefan.lang, daniel.hoelbling)@sbg.ac.at

[b] Austrian Academy of Sciences, GIScience,. Schillerstrasse 30, A 5020 Salzburg - matthias.moeller@oeaw.ac.at

KEY WORDS: LULC, adaptive per-parcel approach, object-based image analysis (OBIA), Cognition Network Language (CNL), object modelling

ABSTRACT:

The process of satellite-based land use and land cover (LULC) mapping often needs to integrate a priori geo-spatial data. A way to consider pre-existing boundaries is per-parcel classification. In the case of crop monitoring on agricultural fields, a per-parcel approach facilitates the classification process by providing pre-defined, ready-to-use boundaries. In other cases, e.g. when using cadastral boundaries, the outlines may not necessarily coincide with LULC information. In this paper we discuss an adaptive per-parcel approach of object generation based on multi-spectral satellite data incorporating the given outlines. The approach differentiates between three cases: (1) a parcel coincides with a homogenous image object; (2) a set of parcels needs to be merged because of homogeneity of the underlying spectral information (3) a single parcel is spectrally heterogeneous, needs to be split and new boundaries are to be generated by object-specific segmentation. The study was carried out in a 3654 km² sized study area covering the Stuttgart Region in Germany. We used orthorectified, mosaicked SPOT 5 multispectral data (5 m ground sample distance; GSD), co-registered and orthorectified. The digital cadastre data were from 2005. We applied object-based image analysis (OBIA) and used cognition network language (CNL) for modelling objects individually in a semi-automated way. About one fifth of the initial cadastre units have been further subdivided due to internal heterogeneity. But the majority of the units have been merged due to redundancy of the boundaries within. By this, the initial number of units has been reduced to less than one fourth. Expert assessment revealed that more than 96 % of the boundaries dissolved were removed correctly. The accuracy of newly introduced boundaries and also the accuracy of retained boundaries was about 86 %. The result met the demand of the given task, although combining data sets of different scales implied some methodological weaknesses. Overall, high potential of the approach can be attributed to the high degree of automation ensuring cost-efficiency, transferability and compatibility of the results. With regard to related applications, further perspectives of using this approach are given in the conclusion.

1. INTRODUCTION

For the mapping of land use and land cover (LULC) remote sensing imagery is a useful source of information due to its synoptic capabilities, i.e. the acquisition of information for large areas at one time. In many cases digital mapping data for the area of interest is already available and might be used as an additional source of information. Especially change detection analysis at a given time (*t*) often relies on mapping results acquired at *t*-1. In those cases it is not necessary to perform a complete new LULC classification; the required update is rather limited to objects where significant changes have occurred. Objects with no change could be sorted out and then, in a second step, those objects with changes could be analyzed in detail.

A priori knowledge to be utilized in this process is available in form of pre-existing geo-spatial data representing areal objects with either spectral homogeneous appearance, usage, administration or belonging. In this research we used digital cadastral data, where outlines usually represent property. Although it is often observed, that property is somewhat linked to homogeneous land cover or land use, the property is not necessarily defined by homogeneity in terms of LULC. To tackle these observations we differentiated three cases (1) the cadastral object matches a homogenous image object; (2) several cadastral objects need to be merged because of homogeneity in the underlying image information; (3) a single cadastral object is spectrally heterogeneous, and therefore needs to be split and new boundaries have to be generated by object-specific segmentation. One premise was that whenever possible, object boundaries should coincide with the original cadastral borders, so only if cadastral limits do indicate a change in category, the respective boundary is retained. Otherwise, if changing categories are not reflected by the cadastral data (e.g. different forest types) a new object border will be generated.

The final result of these steps is a LULC map sharing the same object boundaries as provided by the digital cadastral map, wherever these boundaries indicate changing LULC. Such a LULC map is an accurate and compatible dataset with administrative importance and higher geometric accuracy compared to data sets already available, such as CORINE (Coordinated Information on the (European) Environment).

2. PROJECT SETTING, DATA AND METHODS

2.1 Area of interest and aim of the study

This study has been carried out in the 3654 km^2 sized Stuttgart Region located in the south western part of Germany, in the federal state of Baden-Wurttemberg (Fig. 1). In this area, as in many others of the state, property had been constantly split and

* Corresponding author.

divided while inherited over centuries, which has led to comparatively small parcels of real estate at present.

The research reported on is embedded in a broader context and reflects the first part of a project called BIMS (Biotope Information- and Management System), which was carried out under the lead of the Stuttgart planning office GÖG (Gruppe für ökologische Gutachten). The project aims at establishing a monitoring system of biotope complexes for observing and assessing the ongoing land-use transformation of biotope complexes (Schumacher and Trautner, 2006). For the automated delineation of these complexes, the study discussed in this paper was a crucial first step.

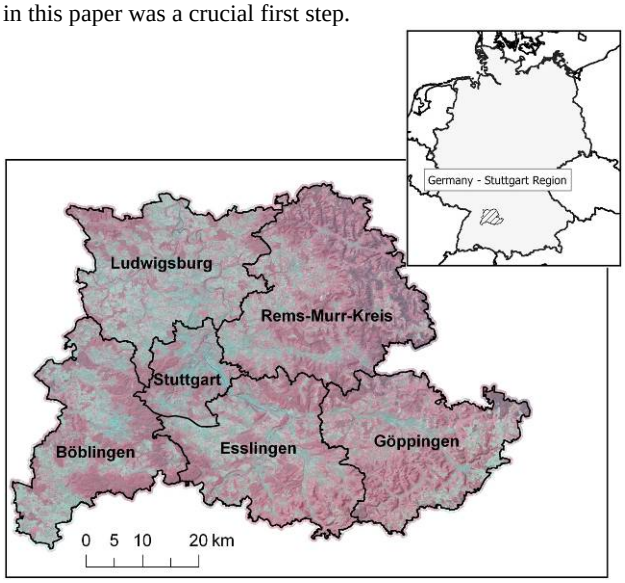


Figure 1. Stuttgart Region and administrative districts in south western Germany

2.2 Data Material and Preprocessing

The boundaries of the Automated Cadastral Map (ALK) represent the spatially most accurate geodata available for the entire area of federal Germany. The ALK contains the outlines of objects and the attached Automated Cadastral Book (ALB) lists the ownership information. Each object of both databases is connected via unique IDs. As a logical consequence each parcel (object) listed in the ALK belongs to one owner. In this research we focus only on parcel boundaries, however, there are a number of additional objects listed in the ALK database, e.g. buildings, power-lines, etc. The digital cadastre information used in this study dated from 2005. Some minor parts of the cadastre were not available in the revised form of 2005 and therefore had to be complemented by cadastre information from 2004 (see Fig. 2, left). Whereas using cadastre information from 2004 throughout would have been more agreeable with the date of the satellite data, working on the most recent cadastral data was required from client side. Additionally, in some parts the boundaries between the cadastral data sets of neighbouring administrative districts were inconsistent (Fig. 2, right). The workflow is illustrated in Fig. 3.

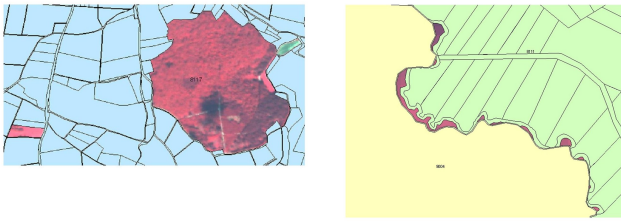


Figure 2. Problems encountered using the digital cadastre data

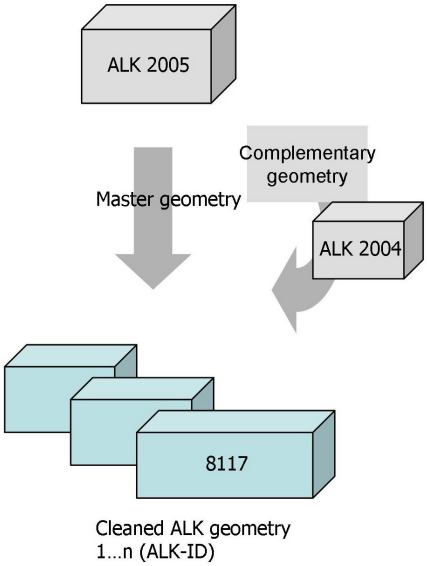


Figure 3. Workflow for preparation of ALK geometry

For determining the actual LULC type multi-spectral remote sensing imagery was used. Four SPOT 5 pan-sharpened (5 m ground sample distance; GSD) color images (scene reference: 53/251, 55/251, 53/252, 55/252), level 1A, acquired between September 2 and 6, 2004, have been obtained. For the task, SPOT 5 data provide sufficient spectral resolution, three bands ranging from the green visible light to the near infrared electromagnetic spectrum. Spatial accuracy could be improved significantly by co-registering the data to an existing orthophoto mosaic (0.25 m GSD) and orthorectifying using a DEM (5 m GSD). As the 5m-DEM did not completely cover the entire area, detected voids were filled using 30m-DEM data (see Fig. 4).

For orthorectification we have parameterised the orbital pushbroom model as implemented in the Leica Photogrammetry Suite. Mosaicking was performed using breakpoints. Finally we clipped the complete SPOT 5 mosaic to the boundaries of the respective administrative district, considering a 500 m buffer around the district outline. The workflow is outlined in Fig. 5.

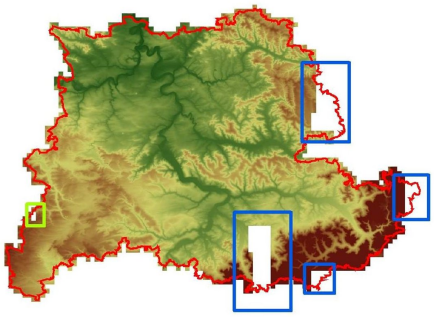


Figure 4. Mosaicked 5 m DEM of Stuttgart Region. Missing tiles (indicated by rectangles) were filled by available 30 m DEM data.

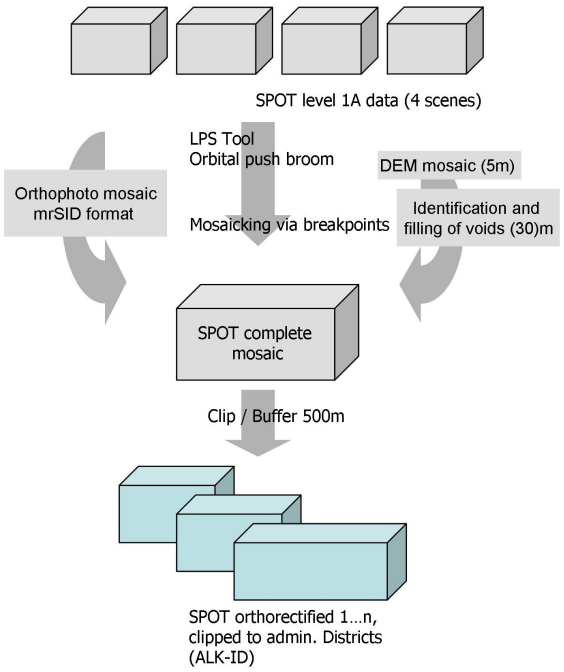


Figure 5. Workflow for orthorectification of the SPOT 5 data: (1) Co-registering to an existing orthophoto mosaic (2) Orthorectification using the DEM mosaic (3). Clipping of the orthorectified SPOT 5 mosaic to the boundaries of the respective administrative district

The resulting SPOT ortho-products showed very high spatial accuracy, the root mean square error of displacement was considerable below 1 pixel. Atmospheric correction has not been applied due to the high quality of the original data material, where no atmospheric disturbances were detectable, and the successful histogram balancing while mosaicking.

Within the framework of the BIMS project, we focused on open areas and forest. Settlements were of no concern. For masking out settlement areas, we used digital ATKIS (administrative topographical cartographical information system) data.

2.3 Methods

We used an adaptive per-parcel approach to treat cadastral parcels individually. The method differs from a 'classical' per-field (De Wit and Clevers, 2004) or parcel-based (Ozdarici and Turker, 2005) approaches in such a way that heuristics need to be found to (1) tell mandatory outlines from redundant ones, and (2) introduce new boundaries wherever needed. The critical information for this task is obtained from the underlying multispectral satellite data using object-based image analysis (OBIA, Lang and Blaschke, 2006). Cognition Network Language (CNL) enables to address single objects and their specific behaviour. CNL is a modular programming language in the Definiens Software environment, which supports the development of complex, reproducible and adaptable rule sets (cf. Tiede and Hoffmann, 2006).

In a first step the digital ALK data were used as pre-defined boundaries to perform parcel-based segmentation. Resulting image objects correspond to the cadastre units (Fig. 6). Settlement areas were masked out through a spatial selection process using the respective ATKIS data layer.

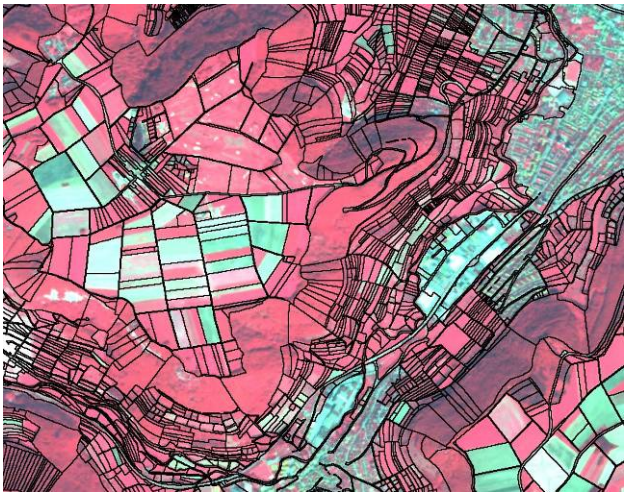


Figure 6. Subset of Spot 5 data set overlaid by ALK data. Settlements were masked out.

2.3.1 Object splitting

In a cyclic object modeling process the produced image objects were checked in terms of spectral homogeneity. In this modeling process each individual object was compared with the spectral values of the underlying image data. In case the standard deviation of the spectral values exceeded a threshold (> 10 in the green band) the object has been marked as a candidate for the splitting operation. Whenever splitting was required, we applied a region-based, local mutual best fitting segmentation (Baatz and Schäpe, 2000) within this object. Elongated objects from the ALK data set like streets or tracks were also selected and split since compact objects were required.

For this domain-specific segmentation (Tiede et al., 2006) we used the following parameters: Scale parameter (SP) = 50; Shape weighting (SW) = 0.5; Compactness weighting (CPW) = 0.5. New objects were generated, which were embedded in the original cadastral object. These new objects are now corresponding to the LULC information derived from the satellite data. Cadastral boundaries, not representing LULC change, are still present (cf. Fig. 7).

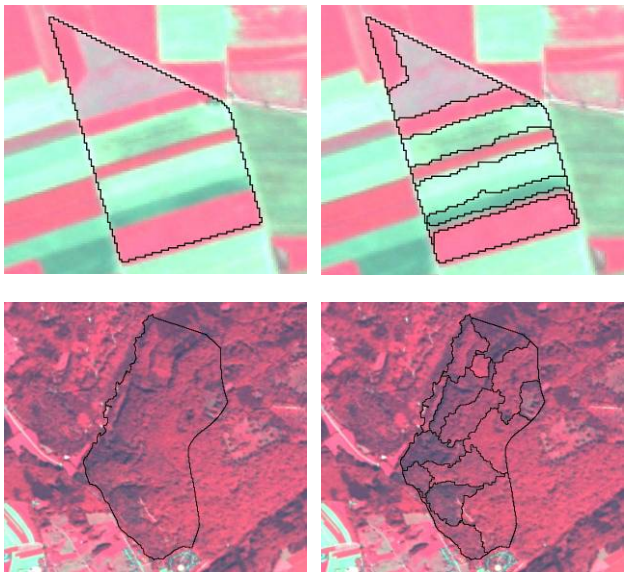


Figure 7. ALK objects before (left) and after (right) the automatic splitting process. Top row: agricultural fields; bottom row: forest.

2.3.2 Object merge

To get rid of redundant boundaries an object merge algorithm has been developed. Objects with similar spectral information were grouped into homogeneity classes. The objects in each class were forced to be merged. In contrast to a simple dissolve operation this approach makes it possible to control the merging process: To avoid fairly elongated objects a compactness threshold (CPW = 0.9) and a maximum size (approx. 50 ha) threshold were applied. Cadastral boundaries not corresponding to LULC change were only kept in some cases to limit uncontrolled size and shape growth in the object merge process (cf. Fig. 8)

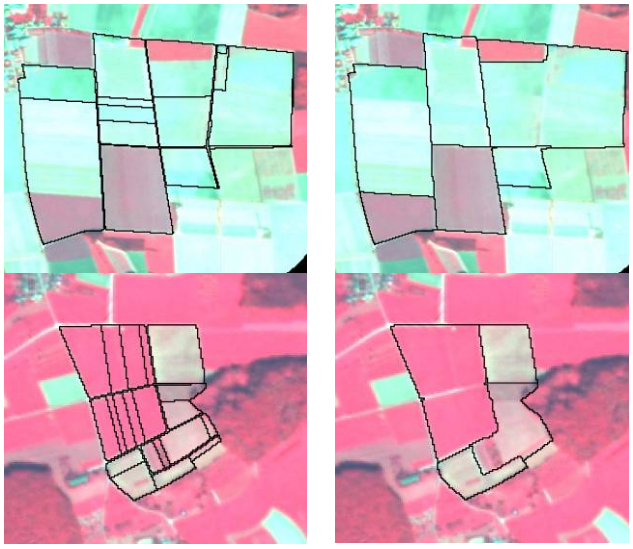


Figure 8. ALK objects before (left) and after (right) the automatic merge and cleaning process of homogeneous objects

2.3.3 Object cleaning

Due to the different resolutions introduced by combing SPOT and cadastral data, small objects of the cadastral map could not be incorporated in the modelling process. In addition, in some cases sliver polygons occurred. The shift in scales was accommodated in a way that very small objects (slivers) were dismissed and a minimum size (2 ha) of generated objects was considered. Figure 9 shows an overview of the entire workflow.

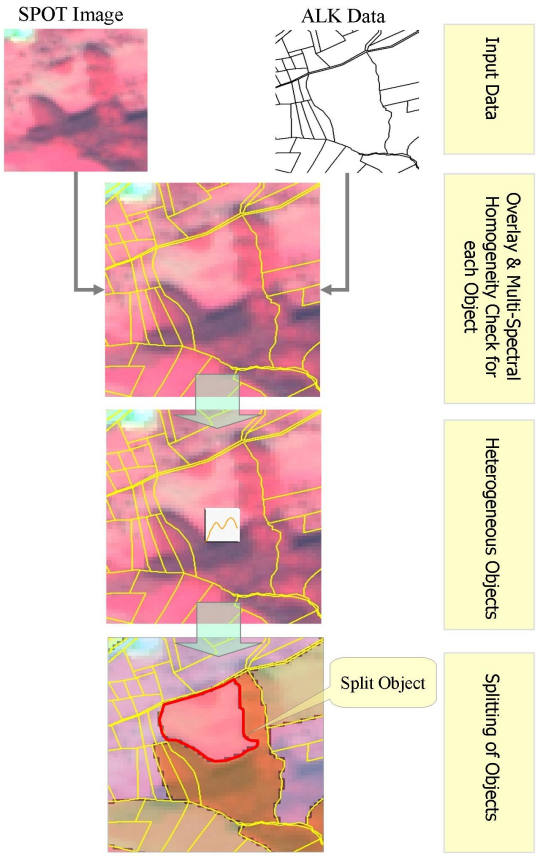


Figure 9. Processing workflow

3. RESULTS AND DISCUSSION

The results of the object modeling process fulfilled the given requirements of the project by representing homogenous objects in terms of LULC but sharing cadastral boundary, wherever LULC is changing. Insofar the results of our study have completely met the demands of the client or user. In terms of transferability, we applied the same approach to all administrative districts in the Stuttgart Region with only minor adaptations of parameters in the north western part of the study area.

The following bar chart (Fig. 10) shows the decrease of the number of units for the whole study area by performing object splitting, merge, and cleaning. Whereas in about one fifth (20-25 %) of the cadastral units new boundaries have been introduced, the majority of units were merged due to spectral similarity. This accounted for an overall strong decrease in the number of units, from approximately 1 million units (without urban areas) to less than 220,000.

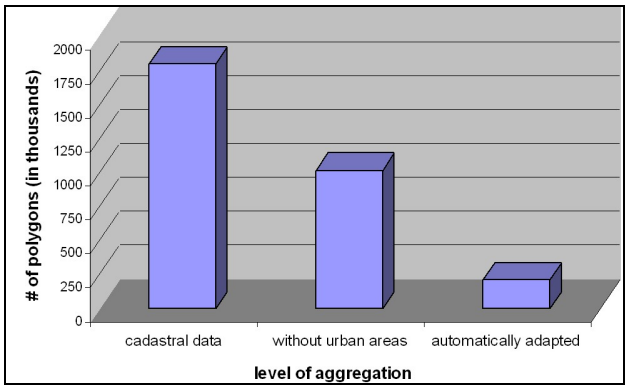


Figure 10. Decrease of the number of units by performing object splitting, merge, and cleaning.

An evaluation of the generated units was carried out by experts for habitat mapping, who used the extracted LULC types as base units for biotope complexes. In case the LULC types were not matching the given criteria for homogenous objects, the borders were changed manually. We used the expert evaluations for validating the approach and carried out an "accuracy assessment" for a subset, which covers about 583 km^2 (~16 % of the study area). The amount of cadastral units (without urban area) in this subset has been reduced from 87,162 to 14,291 automatically derived homogeneous LULC units. After the expert evaluation the final amount of units has again risen to 16,265. Table 1 provides an overview about the lengths of introduced, removed and retained boundaries, compared to the cadastral data and the final evaluation. More than 96 % of removed boundaries were correctly removed, which corresponds to 12,734 km. About 86 % of the newly introduced boundaries were correct (1,510 km) and nearly the same value (86.7 %) applies to the correctly retained boundaries (6,531 km). Missing boundaries, which could not be automatically derived, were digitized manually by the experts summing up to a total length of 348 km.

	Length [km]	**%**
Introduced boundaries	**1,745**	
Correctly introduced boundaries	1,510	86.5
Incorrectly introduced boundaries	235	13.5
Removed boundaries	**13,258**	
Correctly removed boundaries	12,734	96.0
Incorrectly removed boundaries	524	4.0
Retained boundaries	**7,531**	
Correctly retained boundaries	6,531	86.7
Incorrectly retained boundaries	999	13.3
Missing boundaries (manually introduced)	**348**	

Table 1. Evaluation of the results for a subset covering around 16 % of the study area (without urban areas).

The high amount and accuracy of correctly removed boundaries shows the potential of the approach to considerably reduce manual work and to obviously increase cost-efficiency substantially.

With regard to the integration of different data sources, there are methodological problems attached to the combination of scales. In other words, minor limitations occurred due to different spatial detail introduced by the combination of SPOT image data and cadastral vector data. The overall accuracy of the optimized vector objects and also the newly introduced boundaries correspond to the spatial image resolution of the SPOT data. Actually, when dealing with cadastral objects, i.e. features with the most detailed outlines, image data with a finer spatial resolution should be used. In our case, a fast and cost-efficient update of LULC types was requested and for this task SPOT image data were considered sufficient enough in terms of spectral and spatial information.

4. CONCLUSION AND FUTURE PERSPECTIVE

The introduced method demonstrates the performance for a semi-automated optimisation of cadastre-level units and can be extended to related research problems. There is high demand of actual map information world-wide in particular in densely populated countries like Germany. The main LULC data in Germany, ATKIS, covers all digital geo-data in a representation equivalent to a scale of about 1:25.000. The DLM (digital landscape model) as part of ATKIS and equivalent to a topographic map sheet needs a frequent update, because it is used in many planning processes. In 1997 the first complete DLM25/1 was established for entire Germany. An update DLM25/3 is planned for the year 2007. At the moment this process comes very close to a halt due to lack of labour. Using very high resolution space borne image data with sufficient spectral range, an update could be performed comparatively fast and with reliable results. At the moment huge efforts are likewise made for updating the historic CORINE LULC data. To this end, the delineations from the mapping campaign carried out in 2000 (Keil et al. 2005) are compared with recently acquired SPOT and IRS image data. Adaptation of our method could lead to some remarkable time and cost savings, because only objects with a significant change will be automatically selected and further examined by an interpreter.

The presented method can also be transferred to different geographical regions; huge demands for the update of map information are almost anywhere on Earth. When thinking about topo-sheet information in less developed countries, the absence of reliable LULC information becomes an issue. Usually there is some historic LULC information available, but usually several decades old and available as printed maps only. After digitizing object boundaries from these maps, an update could be performed based on recent high resolution remote sensing image data and the method described above.

Upcoming research will focus on the update of ALK cadastral data. As mentioned before, this data set is the most accurate spatial data available for Germany. But keeping the data updated is difficult and the combination of existing cadastral ALK data with recent remotely sensed imagery could help tackling this problem. Extremely high resolution airborne scanner data from airborne sensor systems like Vexcel Ultracam, ADS40, DMC and HRSC-AX will be considered for this task. This type of image data is more and more available and provides both: an extremely fine spatial resolution of up to 5 cm pixel size and reliable geometric accuracy in the range of about +/- one pixel (see Moeller, 2003). This new data material

when being used for cadastral updates, make methods as the one descried nearly indispensable.

REFERENCES

Baatz, M. and A. Schäpe, "Multiresolution Segmentation – an optimization approach for high quality multi-scale image segmentation", in Angewandte Geographische Informationsverarbeitung XII, Strobl, J., Blaschke, T., Griesebner, G. (eds.), Wichmann: Heidelberg, 2000, pp. 12-23.

De Wit A. J., J. G. Clevers (2004): Efficiency and accuracy of per-field classification for operational crop mapping. International Journal of Remote Sensing. 68(11), pp. 1155-1161.

Keil, M., Kiefl, R., Strunz, G. (2005): CORINE Land Cover 2000 - Germany. Final Report. German Aerospace Center, German Remote Sensing Data Center Oberpfaffenhofen, July 2005.

Lang, S., T. Blaschke (2006) Bridging remote sensing and GIS - what are the most supportive pillars? In: Proceedings of the 1st International Conference on Object-based Image Analysis, July 4-5, 2006 in Salzburg. CD-ROM

Moeller, M. (2003): Urbanes Umweltmonitoring mit digitalen Flugzeugscannerdaten, Book with CD, Wichmann, Karlsruhe, 126 p.

Ozdarici A., M. Turker, Comparison of different spatial resolution images for parcel-based crop mapping. ISPRS Workshop, Commission II, WG2, Spatial/Spatio-Temporal Data Mining (SDM) and Learning, November 24-25, Ankara, Turkey, CD.

Schumacher, J. and J. Trautner (2006): Spatial Modeling for the purpose of regional planning using species related expert knowledge. The Biotope Information- and Management System of Stuttgart Region (BIMS) and its deduction from the Information System on Target Species in Baden-Württemberg In: Buhmann, E., Ervin, S., Jørgensen, I., Strobl, J. (eds.): Trends in Knowledge-Based Landscape Modeling. Wichmann-Verlag, Heidelberg

Tiede, D., C. Hoffmann (2006): Process oriented object-based algorithms for single tree detection using laser scanning data"; EARSeL-Proceedings of the Workshop on 3D Remote Sensing in Forestry, 14th-15th Feb 2006, Vienna, 162-167.

Tiede, D., S. Lang, C. Hoffmann (2006): Supervised and forest type-specific multi-scale segmentation for a one-level-representation of single trees. In: Proceedings of the 1st International Conference on Object-based Image Analysis, July 4-5, 2006 in Salzburg. CD-ROM

ACKNOWLEDGEMENTS

This study has been carried out within the project Biotope Information- and Management System (BIMS), financed through the Verband Region Stuttgart (contact: Mrs. Weidenbacher) for the purpose of a regular update of the regional plan. We thank Jens Schumacher from the *Gruppe für ökologische Gutachten* for fruitful discussions during the course of the project and his effective project management.

DETECTION OF POSE CHANGES FOR SPATIAL OBJECTS FROM PROJECTIVE IMAGES

Boris Peter Selby [a], Georgios Sakas [b], Stefan Walter [a], W. -D. Groch [c], Uwe Stilla [d]

[a] MedCom GmbH, Medical Imaging, Rundeturmstr. 12, 64283 Darmstadt, Germany
[b] Cognitive Computing and Medical Imaging, Fraunhofer IGD, Fraunhoferstr. 5, 64283 Darmstadt, Germany
[c] Fachbereich Informatik, University of Applied Sciences, Haardtring 100, 64295 Darmstadt, Germany
[d] Photogrammetry and Remote Sensing, Technische Universitaet Muenchen, Arcisstr. 21, 80333 Muenchen, Germany

KEY WORDS: Pose estimation, Change detection, Projection, Image registration, Mutual Information, Landmark segmentation

ABSTRACT:

Numerous fields of application effort the detection of pose changes for 3 dimensional objects in six degrees of freedom (6 DoF). Automatic procedures that exploit 2D images for the detection of pose changes can be used for example for tracking object movements, for quality control or for the verification of the alignment of patients in radiation treatment devices. In this contribution we present two different solutions for the detection of pose changes that base on the comparison of two 2D images resulting from the projection of an object in the new pose and a 3D volume of the same object in a known reference alignment. Whereas for the first solution we use an object where we can clearly extract landmarks useable as reference positions for the determination of the object's alignment, we provide a second solution for objects where these landmarks cannot be extracted, which is involved automatically if necessary. In this case grey value based pose estimation is conducted by registering the computationally projected reference 3D volume to the 2D images. As reference data for the object with known alignment, CT slices will be used, as they are provided for the alignment of patients in radiation treatment devices. Two X-ray images of the same object in an unknown pose can then be compared to the reference data to determine the respective pose change, which may consist of 3 rotations and 3 translations. Using both approaches to determine patient misalignments in treatment devices shows, that both methods result in highly accurate pose detections and that the second method, despite being less accurate and more time consuming, is an appropriate solution in cases where landmark detection fails.

1. INTRODUCTION

1.1 Motivation

Modern particle beam based radiation treatment techniques for tumours allow accurate application of the treatment dose onto carcinogen tissue with an accuracy much better than 1.0 mm and therefore require an accurate alignment of the patient in the treatment facility (Verhey et al., 1982). Common strategies like tracking of external markers or fixation of the patient's body do not suffice the requirement of high set-up precision and are not feasible whenever internal tumours are to be irradiated, because of possible movements of the treatment target relative to the outer body shape.

Today it is common practice in image guided radiotherapy to align patients manually in the treatment device according to visual evaluation of reference images as X-rays and CT volumes that allow an estimation of the patient's misalignment (Thilmann et al., 2005).

During this time consuming procedure, the alignment of the respective body region may change, which leads to unknown set-up errors and degrades the results of the treatment. Besides that, a manual alignment correction can hardly be done for 6 DoF, because rotational misalignments can hardly be detected visually in the imaged objects.

To overcome these problems an approach for the automatic determination of alignment errors is used, which is based on the comparison of the position of internal landmarks whenever fiducial markers are available.

The use of fiducial markers can be advantageous in many cases, because a marker based procedure allows determining a pose correction without being influenced by surrounding tissue, for example for pose correction of the eyeball.

However, fiducial markers are not always present or detectable. Because of the invasive character of the maker application, marker attachment is not possible for many anatomical regions. In these cases, the respective alignment of the target region has to be estimated by comparison of either natural landmarks or other image properties. Because detection of natural landmarks, especially for soft shaped objects, is not very reliable, we use an approach that estimates the pose by comparison of the grey value distribution in the projected images, which are in the case of radiation treatment X-ray images and the reference data volume, here a CT dataset of the involved body region. To be able to achieve best results for all cases, the second approach is involved automatically if the landmark-based procedure is not able to deliver acceptable results.

1.2 Aims

Our aim is to provide fast and stable algorithms that allow detecting changes of the spatial alignment of an object in respect of a reference position, using 2D projections of the object. These known changes can be used to realign the object, to achieve a correct placement. We do not intend to restrict the

procedure to any known object geometry, but limit it to rigid transformations, because of the inability of correcting object deformations.

Two different solutions are provided, because usage of landmarks comes with several advantages, but is not applicable in all cases. If the procedure using landmarks does not lead to proper results, the alternative approach shall be used to accomplish the pose estimation. The decision which of the two procedures is to be used shall be automated.

The radiation therapy field of application is used for the implementation and test of the procedures, because high accuracy and reliability are of special importance in this scope. Using high-resolution images we aim to achieve accuracies better than 1.0 mm and 0.5° for the detection of pose changes. Besides that, volumetric data as well as projective images of respective objects are available in this scope and specific fixation devices, as for example stereotactic frames, can be used to validate the results in a controlled environment.

1.3 Overview

This contribution consists of two main parts. Part one introduces methods for pose estimation that base on fiducial markers in two projections of an object and a 3D volume dataset. The markers used in this work are tantalum clips attached to an eyeball. We use 4 or 5 clips, which are resided in a CT dataset of the said eyeball and 2 X-ray images of the eye in different alignment. To be able to conduct an automatic detection of the pose change, the landmark positions are extracted from the 3D and the 2D image data. Then an inverse projection of the 2D positions is performed to be able to compute the transformation between the resulting points and the reference markers from the volumetric dataset. This is done by a rigid registration of the point-sets that has to be tolerant against single inaccuracies from the landmark detection.

If the first approach fails or leads to inconsistent results, the second approach is conducted automatically.

The second approach provides a solution for cases where no proper landmarks can be found. Then the volume dataset is projected into the imager planes of the X-ray beamlines for different virtual object poses. Optimisation of grey value distribution based image comparators gives the alignment of the reference dataset. The modification of the reference volume alignment results in the change of the object's pose relative to the original pose.

1.4 References to related work

A solution for X-ray based pose estimation is given in (Bhunre et al. 2007). In this approach, a single 3D model is used to estimate the pose of a bone, visible in X-ray images. As the optimal shape for the model can vary from patient to patient, a single model does not suffice the demand for a highly flexible solution. Besides that, the model-based solution requires a segmentation of the respective object in the X-ray image. As X-ray images can be noisy and the same anatomical objects may appear different, depending on the X-ray beam's energy, it is not possible to rely on a segmentation, especially if, as in our case, the goal is to detect pose changes for any rigid object, without restricting the detection to bones.

The approach described in (Frahm et al. 2004) uses a Harris Corner Detector to extract features from several images obtained from different camera positions. Through comparison of these features, the pose of an object can be estimated. This approach is not applicable in the case of X-ray images, where detected corners may be placed anywhere inside the object, because of the nature of the X-ray imaging, not mapping the surface onto the image detector, but the integral of absorptions of the ray on its way through the object. For X-ray images it is not possible to determine whether a visible feature is located on the surface, within or at the back of an object.

In (Tang et al. 2000) a feature based method is proposed, basing on the comparison of features in one single X-ray image to known feature positions in 3D space. Accuracies of about 1 mm and 2° could be reached. Using one single X-ray image implies, that a point to line registration has to take place. The accuracy of the estimation for the distance of the 3D object from the X-ray source depends on the beam width of the X-ray beam pyramid. Because shifts in direction of the beam can only be determined by size changes of the projected 2D image, the accuracy of the results is restricted on how accurate the image-scaling factor could be determined. In a typical radiation machine device, where the source – detector distance is up to 3 m and more, much higher accuracies can be reached, using two X-ray images.

2. METHODS

2.1 Landmark based pose estimation

The first approach is based on the comparison of artificial landmarks.

2.1.1 Identification of landmarks

Detection of landmarks in 3D data
To enable very fast detection, clips are segmented in the CT data using different levels of volume resolution. The CT is resampled to a 4D pyramid containing several instances of the volume, each with a different resolution. The search for a clip starts at a low-resolution level. As soon as a potential clip voxel is found the search continues at a higher resolution. To determine if a voxel belongs to a clip, two Hounsfield thresholds are used. A voxel between those thresholds is considered to belong to a clip. As soon as a clip is found, the thresholds are adapted, assuming that upcoming clips are of the same material and will be represented by similar Hounsfield values. If no clip can be found, the thresholds are modified and the search continues (Fig. 1).

Figure 1. Landmark detected in consecutive slices

Detection of landmarks in 2D projections
For detecting the landmarks in the X-ray images, two different approaches are combined. Using a Harris Corner Detector (Harris et al., 1988) potential clip corners are identified. Then the convex hulls for the point-sets are determined, as the shape of the eye clips will be convex after projection onto the X-ray panel plane. The area A_{poly} of a convex polygon is calculated by equation 1:

$$A_{poly} = \sum_{i=1}^{N-2} A_{\Delta}\left(P_1, P_{i+1}, P_{i+2}\right) \quad (1)$$

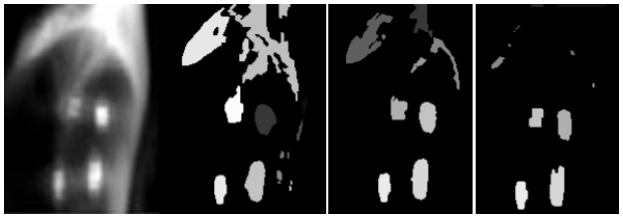
where N = number of polygon corners
P_i = ordered corners of the polygon
A = area of the triangle

In a next step, segmentation is performed inside the area of the convex polygons. A grey value threshold T is used to identify potential clip pixels. The area A of the identified pixels is determined by equation (2):

$$A = sy \sum_{i=y0}^{y1} \left(sx \sum_{j=x0}^{x1} t(i,j) \right)$$
$$t(i,j) = \begin{cases} 1 & for \quad I(i,j) \geq T \\ 0 & else \end{cases} \quad (2)$$

where $y0, y1, x0, x1$ = bounding box of the polygon
sx, sy = pixel size on image plane
$I(x, y)$ = pixel intensity

If the quotient of A and A_{poly} becomes larger than a certain threshold, the polygon is considered to belong to a fiducial marker. If the total number of resulting marker objects does not correspond to the expected number of markers, the detection is resumed, using a modified threshold T (Fig. 2).



(a) (b) (c) (d)

Figure 2. X-ray with 4 landmarks (a); Iterative refinement of areas identified as landmarks (b, c, d)

2.1.2 Back-projection

Inverse projection
To compare the markers detected in the projective images with the reference markers from the spatial dataset, an inverse projection is performed for corresponding pairs of both images. Based on a known geometric set-up of the imaging devices (Fig. 3), the inverse projection is done by calculating the intersection points of rays from the X-ray source to the centre of the segmented marker in the respective projection plane.

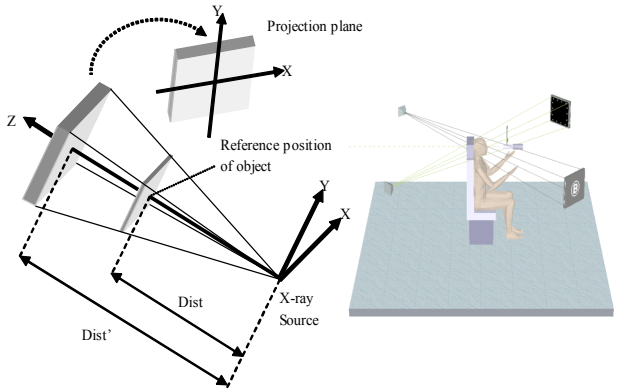


Figure 3. Imaging device (left); geometric set-up of imaging devices (right)

Handling of redundant results
Because it is not always clear which clips correspond to each other, it is possible that one clip becomes a member in several back-projection results. This is the case, if several clips are projected onto a horizontal line in the plane of one flat panel (Fig. 4).

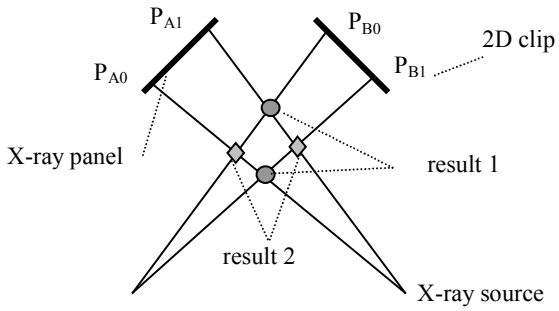


Figure 4. Back-projection with two possible results

All possible resulting point-sets B_i are kept as preliminary results. In the following registration procedure only the point-set, which can be mapped onto the marker positions A derived from the volume data is used to derive the final transformation for pose detection.

2.1.3 Point-set registration

Registration
Several registrations are performed, one for each possible back-projection result with the reference positions. In each registration, two sets of points in the 3D space are used to calculate 3 shifts and 3 rotations, which map one point-set onto the other as good as possible. The remaining mapping error can serve as an indicator for the quality of the calculated alignment deviation. We use a Downhill Simplex optimisation (Press et al., 1992) to minimize the error for the misalignment between back-projected point-set B_i and the original landmark positions A. The error metric bases on the undirected Hausdorff Distance $H(A,B_i)$ (Huttenlocher et al. 1993).

The optimisation may suffer from local minima of the error function (Fig. 5).

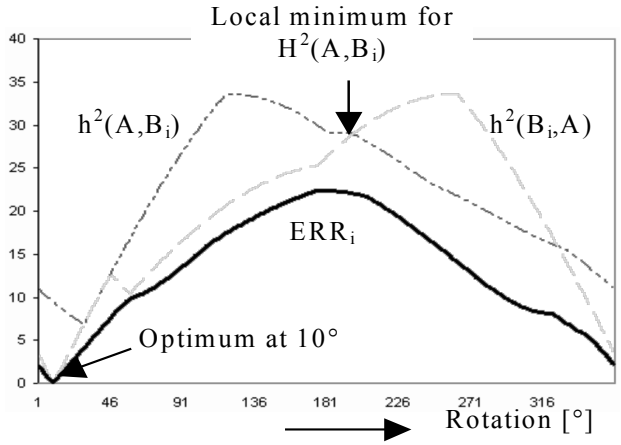


Figure 5. Comparison of error functions

The correct alignment detection for figure 5 is at 10°. The squares of the directed Hausdorff Distances $h(A,B_i)$ and $h(B_i,A)$, as well as the square of the undirected Hausdorff Distance $H(A,B_i)$, which is the maximum of the directed distances, have local minima. To avoid these, we calculate an error ERR_i for each combination of A with B_i as the square sum of a fractional undirected Hausdorff Distance of rank 1 to K, where K denotes the minimum of the number of landmarks detected in either the reference data or the projected images, as shown in equation 3:

$$
\begin{aligned}
ERR_i &= \sum_{k=1}^{K} \mathbf{max}\left(h_k(A,B_i), h_k(B_i,A)\right)^2 \\
h_k(A,B_i) &= kth_{a\in A}\ \mathbf{min}_{b\in Bi}\left(\|a-b\|\right) \\
h_k(B_i,A) &= kth_{b\in Bi}\ \mathbf{min}_{a\in A}\left(\|a-b\|\right) \\
K &= \mathbf{min}\left(N_A, N_{Bi}\right)
\end{aligned}
\qquad (3)
$$

where N = number of points
h_k = directed Hausdorff Distance of rank k

After minimization of the ERR_i values only the transformation for

$$ERR_{best} = \mathbf{min}\left(ERR_1,...,ERR_n\right) \qquad (4)$$

is kept for further calculations.

Acceptance of results
The result of the alignment detection is accepted, if the standard deviation $_{AB}$ between the closest members of the final point-sets A and B_{best} used for the optimisation lies beneath a threshold given in the program configuration as the maximal accepted error. If the result is not accepted, grey value based pose estimation is conducted.

2.2 Grey value based pose estimation

The grey value based pose estimation is conducted if the landmark based approach fails because of the lack of detectable landmarks, which lead to a good match with the 3D reference data.

2.2.1 The algorithm

The procedure for the grey value based pose estimation projects the reference volume A computationally onto the detector planes of the imaging devices, using a pose correction C. The results are two projections of the volume that depend on a current pose correction. For each we calculate a value Q for the quality of the match with the respective X-ray image B, acquired with the object in the current, unknown pose. The quality values are combined. Minimizing the negative combined matching quality by modification of the pose correction C gives the final pose change (Fig. 6).

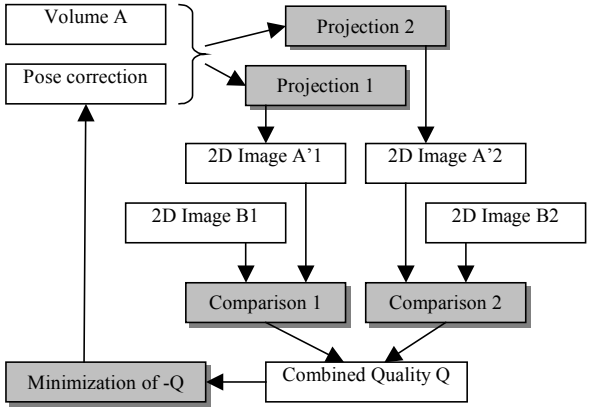


Figure 6. Algorithm for grey value based six degrees of freedom registration

2.2.2 Projection of the volume

The volume is projected onto an imager plane by a ray tracing technique, where rays to the X-ray source are computed for a sub-set of pixels, covering ¼ of the receptor area (Fig. 7).

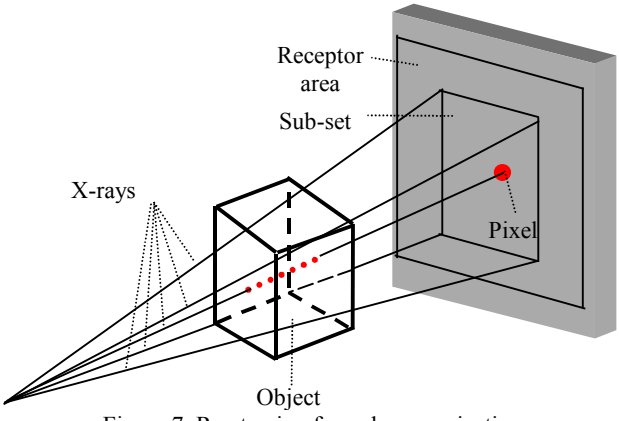


Figure 7. Ray tracing for volume projection

To speed-up the process, we stop ray tracing as soon as summing up the density values of the volume leads to a saturation of the respective pixel intensity. The pixel value is calculated by summation of the voxels, intersected by the ray. The voxels are weighted by their absorption coefficient, which can directly be obtained from the CT data. To improve the quality of the resulting image, trilinear interpolation can be applied as the ray passes through the volume. However, we used

nearest neighbourhood interpolation to improve the performance and because we did not intend to produce visual results of high quality.

2.2.3 Image comparison and optimisation

There exists a wide range of grey value based image comparators in the scope of registration. As methods like cross-correlation or usage of difference images are not applicable for images that differ in much more aspects than contrast and intensity, we decided to use mutual information as image correlation measure (PLUIM et al., 2003).

Figure 8 shows five different joint histograms, where each axis stands for the grey values of one of the images.

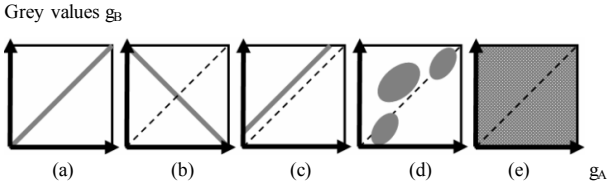


Figure 8. Joined histograms for images A and B: a) identical images; b) inverse images; c) A darker than B; d) partly different images; e) images without correspondences

A joint histogram is built up by reading the grey values of both images at the position of two overlaid pixels and incrementation of the histogram cell by one at the respective coordinates, defined by the two grey values. The mutual information value *MI* is then calculated by equation 5:

$$MI(A,B) = H(A) + H(B) - H(A,B)$$
$$H = -\sum_{g=0}^{G} p_g \ln p_g \qquad (5)$$

where G = largest grey value
p = probability for occurrence of the grey value in the image, based on the distribution in the histogram
H = entropy of either one of the image histograms or the joint histogram

Optimisation of the rigid transformation *T* for the current object pose in 6 DoF is done by minimization of a negative quality value -*Q* given in equation 6:

$$-Q = -\sum_{i=1}^{N} MI_i^{\,2} \qquad (6)$$

where MI_i = Mutual Information value for the image pair *i*
N = Number of images (here 2)

We minimize -*Q* by the downhill simplex method for 6 pose parameters (3 shifts and 3 rotations).

2.3 Results

2.3.1 Landmark based pose estimation

For all tests a standard PC has been used. Tests have been performed using CT datasets (0.2 mm slice distance, 250 slices) and X-rays of a pig's eye attached with 4 and 5 tantalum clips of 2.5 mm in diameter (Fig. 9).

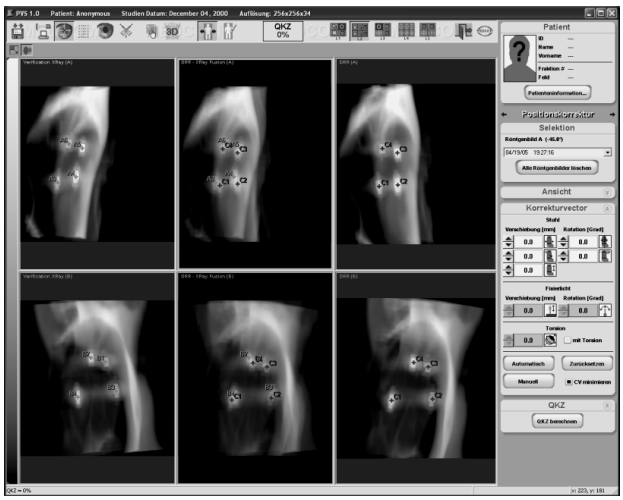


Figure 9. Images of pig's eye with 4 clips: X-rays (left); Fused (centre); DRRs (right)

The X-ray equipment has been calibrated with geometric accuracy of about 0.25 mm. In all cases it was possible to detect all landmarks and to perform a correct mapping of the back-projected marker positions to the reference data (Tab. 1).

Number of Landmarks	Pose change		Calculation error		Time
	shift	rotation	shift	rotation	
4	2.0 mm	2.0°	0.2 mm	0.1°	1sec
4	5.0 mm	10°	0.3 mm	0.1°	1sec
4	10 mm	20°	0.2 mm	0.2°	2sec
5	2.0 mm	2.0°	0.2 mm	0.1°	2sec
5	5.0 mm	10°	0.1 mm	0.1°	3sec
5	10 mm	20°	0.2 mm	0.2°	3sec

Table 1. Pose estimation errors for landmark based approach

2.3.2 Grey value based pose estimation

In this case, tests have been performed using a CT dataset with 295 slices of 0.8 mm slice distance and X-ray images of a human scull. No landmarks have been attached to the scull (Fig. 10).

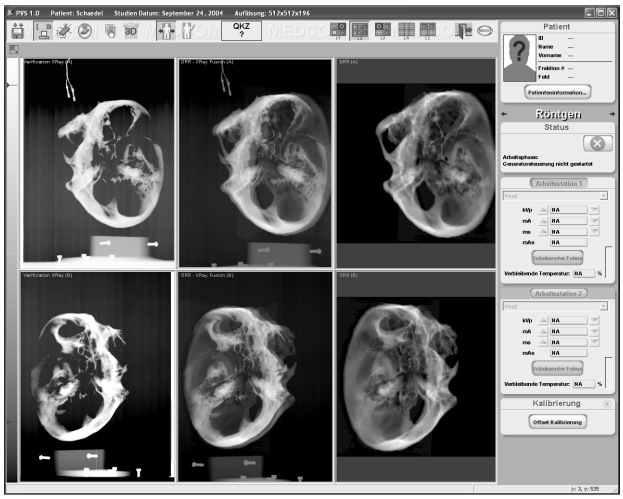


Figure 10. Images of a human scull: X-rays (left); Fused (centre); DRRs (right)

Starting with the landmark based approach the procedure recognized that it was not possible to find a consistent set of landmarks. The automatic procedure for the grey value based pose estimation was started.
For all used X-ray images the grey value based method was able to detect the respective pose change. In table 2 the results are shown for a number of given pose deviations (for better comparability, we chose the same initial poses as for the landmark based procedure).

Pose change		Calculation error		Time
shift	rotation	shift	rotation	
2.0 mm	2.0°	0.9 mm	0.4°	62sec
5.0 mm	10°	1.1 mm	0.7°	98sec
10 mm	20°	2.3 mm	0.9°	113sec

Table 2. Pose estimation errors for grey value based approach

3. DISCUSSION

Both presented methods are able to provide reliable and accurate detection of pose changes. The major disadvantage of the landmark-based method is that it is only applicable if some detectable markers are present.

Regarding the results in tables 1 and 2, the advantages of the landmark based approach become apparent:

- Pose estimation is much more accurate even if we consider that the CT resolution was higher in the case of the landmark based approach;
- The calculation can be done in a few seconds on a standard PC;

Performance of the grey value based approach could be improved, if faster rendering algorithms would be used for the volume projection. However, combination of the two approaches is an ideal solution to exploit the advantages of either method, whenever it is possible.

For the radiotherapy field of application, each method provides an enormous advance in accuracy compared to manual alignment methods.

4. CONCLUSIONS

We presented two different procedures for image based pose estimation. A landmark-based approach has been tested with 2.5 mm tantalum clips, which could be detected relatively easy. Further efforts have to be done to assure a correct detection for the wide variety of applicable metal clips. Under adverse circumstances, as when clips are occluded by other clips or by bony structures of the skull, not all clips can be detected, which results, depending on the total number of clips used, in results less accurate.

If the landmark-based approach failed, grey value based pose detection is started automatically and leads to acceptable results. However, the grey value based approach was less accurate and more time consuming, but whenever no landmarks can be detected, the grey value based approach is an optimal solution to this problem.

REFERENCES

Bhunre, P. K.; Leow, W. K.; Howe, T. S. 2007. Recovery of 3D pose of bones in single 2D X-ray images. IEEE Workshop on Applications of Cumputer Vision 2007. pp. 48.

Frahm, J.-M; Köser, K.; Koch, R. 2004. Pose Estimation for Multi-Camera Systems. Proceedings of Deutsche Arbeitsgemeinschaft für Mustererkennung Vol. 26: pp. 27-35.

Harris, C.; Stephens M. 1988. A combined corner and edge detector. Procs of the 4th Alvey Vision Conference: pp. 147-151.

Huttenlocher, D. P.; Klanderman G. A.; Rucklidge W. J. 1983. Comparing images using the Hausdorff Distance. IEEE transactions on pattern analysis and machine intelligence 15 (9): pp. 850-863.

Pluim, J. P. W.; Maintz, A.; Viergever, M. A. 2003. Mutual registration based registration of medical images: a survey, IEEE Transactions on medical imaging Vol. XX.

Press, W. H.; Teukolsky, S. A.; Vetterling, W. T.; Flannery B. P. 1992. Numerical Recipes in C 2. Cambridge University Press.

Tang, T. S. Y.; Ellis, R. E.; Fichtinger, G. 2000. Fiducial registration from a single X-ray image: A new technique for fluoroscopic guidance and radiotherapy. Springer Lecture Notes in Computer Science Vol. 1935: pp. 502-511.

Thilmann, C.; Nill, S.; Tücking, T; Höss, A.; Hesse, B.; Dietrich, L.; Bendl R.; Rhein, B.; Häring, P.; Thieke, C.; Oelfke, U.; Debus,J.; Huber, P. 2005. Correction of patient positioning errors based on in-line cone beam CTs: clinical implementation and first experiences. International Journal of Radiation Oncology Biology Physics Vol. 63, Part 1, pp. 550-551.

Verhey, L. J.; Goitein, M.; McNulty, P.; Munzenrider, J. E.; Suit, H. D. 1982. Precise positioning of patients for radiation therapy. International Journal of Radiation Oncology Biology Physics. Vol. 8, Part 2, pp. 289-294.

SEMANTICALLY ENHANCED PROTOTYPES FOR BUILDING RECONSTRUCTION

Dirk Dörschlag, Gerhard Gröger and Lutz Plümer

Institute of Geodesy and Geoinformation
University of Bonn, Germany
Meckenheimer Allee 172, 53115 Bonn
(doerschlag, groeger, pluemer)@ikg.uni-bonn.de

KEY WORDS: building reconstruction, 3d, semantics, formal grammars, minimum description length, ontology, level of detail, constraints

ABSTRACT:

We present a system for automatic building reconstruction, combining the strengths of grammars to generate varying building models and the principle of minimum description length [MDL] to evaluate results and to control the search process. The reconstruction process is guided by the level of detail, starting with a coarse level and is stepwise improved towards highly detailed models. On each level of detail, the corresponding components are identified by prototypes, provided by the building ontology. The matching between input surfaces and prototypes is supported by constraints representing relevant topological and geometrical relations between these surfaces. When employing MDL, usually an optimal coding is required. In contrast, we use an asymptotic optimal coding which is easier to generate since no a priori knowledge is needed. Instead, a larger amount of input data has to be processed to achieve comparable results.

1 INTRODUCTION

During the last years a lot of activities in the field of 3D city models could be recognized. A lot of cities, especially in Germany like Berlin, Hamburg or Stuttgart are nowadays owner of digital 3D city models and provide these for other users. There are free viewer tools like Google Earth or Aristoteles3D providing an intuitive interface for accessing this data. But not only the number of 3D city models and visualization tools increases significantly, there are also several new standards for city and building models, e.g. the city geography markup language (CityGML) (Kolbe et al., 2005, Gröger et al., 2006) and the Industry Foundation Classes (IFC) (Eastman, 1999).
The first one was developed within the geographic information community and is being standardized within the Open Geospatial Consortium (OGC), one of the most important organizations in this field. The second one was developed within the architectural community. They are intended to provide interfaces to interchange these models, including the semantics and the topology belonging to the geometric models. At the moment, several projects are using these standards to specify their input and interchange format. One of these is part of the testbed for OGC Web Services Phase 4 (OGC, 2006) of the open geospatial consortium which is evaluating the use of service oriented architectures for 3D city models to achieve better response capabilities in the context of homeland security. Another project is initiated by the government of North Rhine-Westphalia (Germany). It currently uses a similar web service architecture to fulfill the requirements within the noise prevention program of the European Union. Each of these projects requires both, a detailed geometry and assigned semantic information. So nowadays city models do not only have to look good they also have to be smart to enable advanced analytic processing. Due to the requirements of users, projects and applications, methods are needed to acquire 3D city models. Requirements for these methods are:

1. they have to be automatic to enable the generation and update of large-area models within short time frames
2. they need to reconstruct both, geometric and semantic properties of urban objects on a high level of detail
3. and they should generate quality information for each reconstructed object.

Within the literature on automatic building reconstruction from different data sources, several different methods could be recognized. They could be differentiated by the focused level of detail, the use of semantic information during the reconstruction process and the used input data. For 3D city models, these 3 levels of detail (LoD) are commonly used: LoD1 is the well-known blocks model, LoD2 adds roof structures and LoD3 completes LoD2 by including balconies, roof structures like dormers and chimneys (Kolbe et al., 2005). (Haala, 2005) uses 2D ground plans of buildings and digital surface models (DSM) to derive LoD2 buildings. The method proposed in (Fischer et al., 1998) and (Kolbe, 1999) uses pairs of aerial stereo images and semantically founded geometric constraints to reconstruct LoD2 building models. (Grau, 2000) proposes a multi step method using semantic networks to generate first LoD2 building hypothesis and extending these in a second step to LoD3 models by adding e.g. dormers. Those models are evaluated by least square matching with terrestrial stereo images. Both (Brenner and Haala, 1998) and (Vosselman and Dijkman, 2001) employ 2d ground plans, DSMs and some volume primitives to derive LoD2 building model hypotheses. Both approaches differ in the used evaluation method. The first uses least square matching while the second one uses the minimum description length principle, which will be considered in detail in section 2.4. A method similar to (Brenner and Haala, 1998) is presented by (Stilla and Jurkiewicz, 1999). Beside these automatic methods there exist several approaches for interactive, semi-automatic reconstruction of buildings, e.g. (Grün and Wang, 1999, Rottensteiner, 2001, Gülch et al., 1999). Since the focus of this paper is on the automatic reconstruction of the semantical and geometrical properties of urban objects, these approaches are not considered any further. In our approach we combine the strengths of spatial grammars to generate building models, the principle of minimum description length as evaluation function to control and guide the search process and we employ a multi scale approach starting with a coarse level of detail and stepwise refinement towards highly detailed models. We used building ontology for grammar rule design and the definition of our levels of detail. A widely known type of grammar in the context of plant model

generation are L-Systems as discussed in (Prusinkiewicz and Lindenmayer, 1990). This type of grammar was used by (Parish and Mueller, 2001) to generate simple building models. Within the paper (Müller et al., 2006) other types of grammar evolved in the context of architecture and design, such as set grammars (Stiny, 1982) and split grammars (Duarte, 2002, Wonka et al., 2003), are used to generate typical, artificial, fictitious building models for certain contexts, e.g. how the ancient Rome may have looked like (Müller et al., 2005). A crucial problem in building reconstruction is the evaluation of alternative modeling possibilities. A method to deal with this problem is the principle of minimum description length (MDL) (Grünwald et al., 2005). This criterion considers both, the goodness of fit between the model and the data and the complexity of the used model. The use of MDL as evaluation function in our approach is described in section 2.4.

2 METHODS

The method presented in this paper aims at the field of automatic building reconstruction. In contrast to others, we combine the strengths of grammars to generate building models, the principle of minimum description length as evaluation function to control and guide the search process and we employ a multi scale approach starting with a coarse level of detail and stepwise refinement toward highly detailed models. We use building ontology for grammar rule and symbol design and the definition of our specific levels of detail. Important aspects we focus on are formal grammars, the constraint graphs used and finally the model selection criterion MDL.

2.1 Input data

Input data for our approach are terrestrial laser scans of a single suburban building supplemented by an aerial laser scan of the same building.

This cloud of noisy 3D points needs to be preprocessed to remove outliers and to reconstruct the interrelationship between the points. Given the points and the corresponding plane, the boundaries of polygons have to be derived. These may have also interior boundaries.

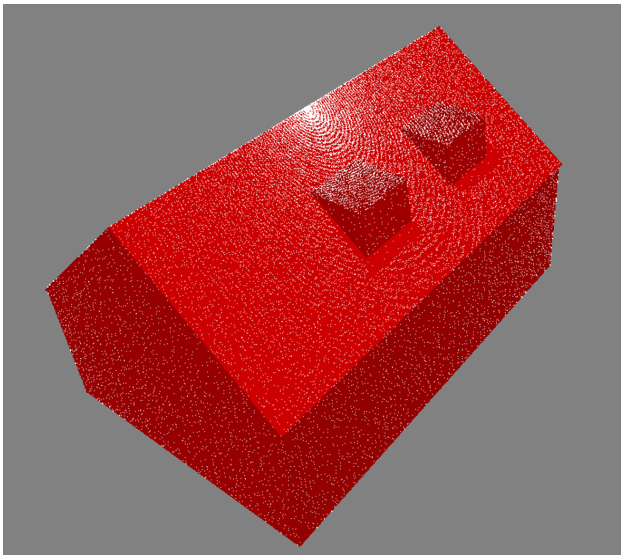


Figure 1: point cloud from a synthetic laser scan and the available 3D polygons

The planes are derived by applying a RANSAC procedure as described in e.g. (Wahl et al., 2005) to the point cloud. After the preprocessing step, we obtain the following:

1. 3d planar polygons, each with one outer boundary
2. the assignment of 3D points to corresponding polygons
3. the assignment of planes to polygons. For each polygon, the primary components and the equation of this planes are stored

Due to the characteristics of laser scanners, the edges of buildings are not observable directly. Instead, they have to be derived by intersection the corresponding planes. This will have consequences for the presented approach, which will be discussed in more detail in section 2.2.

An impression of the input data is shown in figure 1.

2.2 Constraint graph and prototypes

Within any geometric representation of objects, there exists additionally several geometric and topological constraints between modeling primitives like nodes, edges and faces.

The most important observation in this context is that there exist constraints which are invariant against certain types of transformations like scaling, translation and rotation of objects. E.g. given an object consisting of two planes which are parallel, if one of the transformation mentioned before is applied to both planes, then these two still remain parallel. This characteristic of constraints is very important and one of the main reasons why a certain constraint is used in the following.

The modeling primitives and the constraints can be represented in a graph structure. Within such a structure, the modeling primitives are represented by nodes and the constraints are represented as edges between the two nodes they apply to. This graph structure is called constraint graph in the following. This concept was used by (Kolbe, 1999) to match geometric shapes within a 2D context and is part of the weak CSG primitive concept introduced by (Brenner, 2004). In contrast to both papers, the constraint graphs used and discussed here are always embedded in the 3d space and describe e.g. geometric constraints between 3D faces like parallelism between planes.

A constraint graph representation is derived for the given input data. This graph is called data constraint graph (DCG). Within the derived structure any node represents 3D polygons, and the edge represent the ascertainable constraints. An extract of the 3D data constraint graph for a scanned L-shaped building is shown in figure 2. In this figure, any edge represents a 3D polygon and the constraint holds for the polygons or the planes they belong to or both.

The different compartments of the ontological structure of buildings, like storeys or dormers, have to be linked to geometry observable within the input data. These compartments have to be bounded by closed and topologically correct surfaces. Following the idea of the constructive solid geometry, that complex solid can be represented by a set of parameterized instances of solid primitives and a set of boolean operations on them, and be represented in a tree structure (CSG tree), for any compartment typical simple solid were identified. This association of a geometric primitive and a semantic interpretation is called semantically enhanced prototype or prototype for short. An extract of the 3D data constraint graph for a prototype with a cuboid as geometric primitive is shown in figure 3. In this figure, any edge represents a 3D polygon and the constraint holds for these polygons or the planes they belong to or both. Due to the fact that the geometric primitives of the prototypes are decomposable into modeling primitives and a set of constraints that holds between them, it is possible to derive a constraint graph representation for any prototype. The semantic interpretation of a prototype furthermore allows to predict the constraints that have to be introduced if one

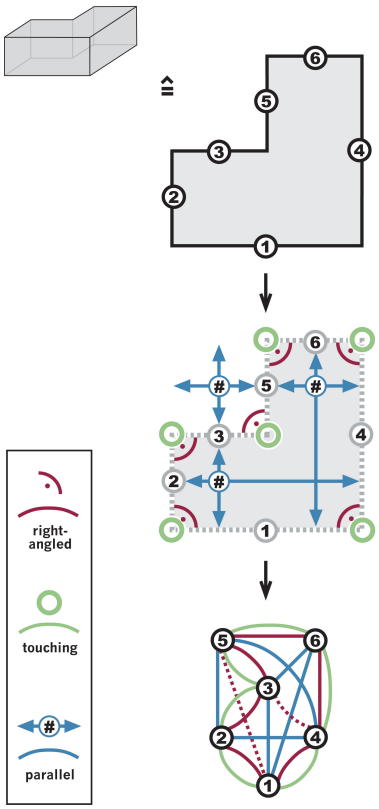


Figure 2: Constraint Graph reconstructible for the wall surface polygons of a house

or more prototype constraint graphs (PCG) have to be integrated into an existing aggregate of PCGs. Because of the special capabilities of constraints mentioned above, each of these prototype constraint graphs (PCG) could be compared with subgraphs of the DCG. If both match, the identified part of the DCG represents an instance of the geometric primitive associated with this PCG and the parameter values for this primitive could be estimated by using the geometry associated with the DCG nodes. Figure 4 illustrates this matching procedure. In this figure, any node represents a vertical 3D polygon and the constraint holds for these polygons or the planes they belong to or both.

One important observation that will be from interest later is that this holds for the matching of more complex constraint graphs with the DCG, too.

Due to the uncertainties of the input data the methods used to derive the data constraint graph (DCG) are modified to be able to handle these uncertainties. Other effects of the uncertainties and the general setting during data acquisition influence the calculation of the matching quality between the prototype constraint graphs and the data constraint graph. The handling of these influences will be discussed in section 2.4, where the principle of minimum description length (MDL) as evaluation function will be presented.

2.3 A grammar for building generation

During the last decades, the connection between grammars and spatial design was addressed by several research groups. Within this section a brief introduction to grammars is given. Several spatial grammars, like set grammars (Stiny, 1980, Stiny, 1982), shape grammars (Stiny, 1980) and structure grammars (Carlson et al., 1991), are available and discussed in the context of the generation of spatial objects.

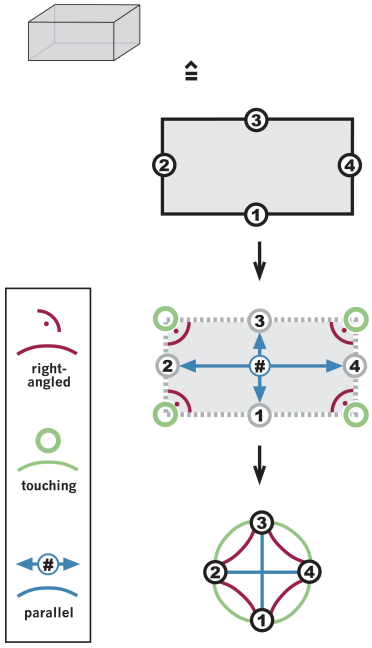


Figure 3: constraint graph within a solid primitive of a cuboid

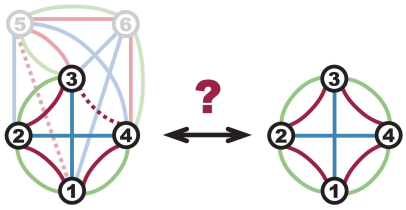


Figure 4: Mapping the constraint graph of the cuboid into the one of the house

The grammar used in our approach (called BG grammar) is a context-free grammar as defined by (Chomsky, 1959). This was extended to an attributed grammar in the way given by (Knuth, 1968, Knuth, 1971). The following definitions are the basis for the grammar we developed:

Definition 2.1 A **formal grammar** G consists of the quad-tuple (N, T, S, P). N is the finite set of nonterminal symbols, T the finite set of terminal symbols with $T \cap N = \emptyset$, $S \in N$ a distinguished start symbol and P a finite set of production rules of the form $(T \cup N)^* N (T \cup N)^* \rightarrow (T \cup N)^*$[1]. The alphabet V of the grammar G is defined with $V = N \cup T$. The language of a formal grammar $G = (N, T, P, S)$, denoted as $L(G)$, is defined as all those strings over T that can be generated by starting with the start symbol S and then applying the production rules in P until no more nonterminal symbols are present.

Definition 2.2 A **context-free grammar** is a formal grammar in which the left-hand side of each production rule consists of only a single nonterminal symbol.

Definition 2.3 An **attributed grammar** is a context-free grammar, were for any symbol $x \in V$ with $V = N \cup T$ a set

[1] S^* terms any final concatenations of the set S including the empty symbol ϵ, S^+ terms any final concatenations of the set S without the empty symbol ϵ

of attributes $\alpha(x)$ exists. For every production rule $p \in P$ there exists a set of semantic rules $R(p)$ of the form $x_i.a = f(x_j.b, ..., x_k.c)$, were $x_i, ..., x_k$ are the occurring symbols within the production rule and $a...c \in \alpha(x)$. For each x within a derivation at least one semantic rule $R(p)$ is applicable.

After recapitulating the basic notions, we now are in a position to present our grammar for the reconstruction of buildings. First we define how the result of the reconstruction process is represented:

Definition 2.4 A **reconstructed constraint graph (RCG)** is a constraint graph , which consists of a set of PCGs connected at common nodes, and two partial functions v, e mapping the vertices of the RGC to the vertices of the DCG, and the edges of the RCG to the edges of the DCG.

The RCG represents the building reconstructed so far and consists of PCGs as subgraph, where each PSG represents a prototype. The functions v associates faces in the RCG with the faces observed in the data, while the function e assign the information whether a constraint in the RSG is observable in the data. Since not all faces and not all constraint in the RCG are observable in the data, both v and e have to be partial functions. The grammar which produces a RCG is a special attributed grammar called BG grammar (building generating grammar) which is specified as follows:

Definition 2.5 A **BG grammar** is a attributed grammar, were the set N of nonterminal symbols is is partitioned in the set PCG of prototype constraint graphs, the set J of junctions and a start symbol $\{S\}$. The set T of terminal symbols is is partitioned in the set TJ of terminal junctions and the set TP of prototypes. The production rules have one of the following three forms:

1. $S \rightarrow PCG$
2. $PCG \rightarrow TP(J)^*$
3. $J \rightarrow TJ$
4. $J \rightarrow PCG$

A junction in the set J represents a part of the hull of the building reconstructed so far, which is denotes a discrepancy between the RCG and the DCG. A junction indicates a missing prototype, which is usually identified in one of the next production steps. If a junction can not be assigned finally, it is replaced by a symbol from the set TJ by applying a rule of the third form.
The reconstruction process using a BG grammar is illustrated now by applying it to the L-shaped building which was already given in figure 2. The process starts with the matching of a PCG (see bottom of figure 3) with the DCG (see bottom of figure 2), by applying a production $S \rightarrow PCG_c$ of the first form. PCG_1 represents the constraint graph of the first detected prototype, e.g. a cuboid. The result is a RCG reflecting the detected polygons and constraints. The corresponding derivation tree is depicted in figure 5. The current string produced by the grammar so far is PCG_1. To this string, a rule $PCG_c \rightarrow cRJ$ of the second form is applied, where c is a cuboid. The resulting string is c_1RJ, where the RJ indicates a missing prototype. In the next step, a rule $RJ \rightarrow PCG_c$ is applied, which yields the string c_1PCG_c. Finally, the rule $PCG_c \rightarrow c$ completes the process. The generated string is c_1c_2, where the geometric information is represented in the attributes of c_1 and c_2. If during the reconstruction

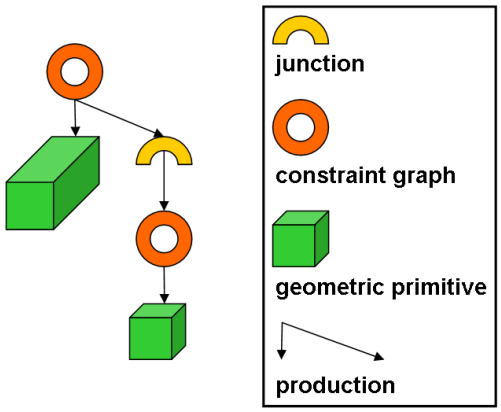


Figure 5: Derivation tree for the first reconstruction level of a L shaped building

process more than one area is not observable, more than one junction is needed. In the rule of the second form, this is reflected by using the star notation.
In the general case typically several rules are applicable to one derived string. Here a selection method is required, which chooses the "best" rule. This selection method is topic of the following section.

2.4 MDL for decisions

As already stated in section 2.3, the process of applying production rules is nondeterministic, since in one step more than one rule may be applied. It is possible to exhaustively generate all possible building models, but this of course is too time consuming. Thus decision mechanisms are required. Within the literature, several selection criteria are discussed (Akaike, 1974, Schwarz, 1978, Rissanen, 1978). For our approach we have chosen the principle of minimum description length [MDL] (Grünwald et al., 2005). The suitability of MDL for building reconstruction was already demonstrated by (Kolbe, 1999, Vosselman and Dijkman, 2001). MDL is an information theoretic model selection criterion. It incorporates the goodness-of-fit between the observed data and the model, and the complexity of the fitted model. Especially the last point is important for the intended use, since we are operating on biased real world data and do not want to transfer the bias into model parts. The formulation of MDL used within our approach is due to (Vosselmann, 1992):

$$\hat{M} = \arg\max_{M_i} I(D; M_i) - I(M_i) \quad (1)$$

where $\hat{M}$ is the selected model, $I(D; M_i)$ the mutual information between data D and model M_i and $I(M_i)$ the information of the model M_i. This formulation of the MDL criterion provides several advantages for the use within our approach. The most important one is, that wild card assignments do not have any effect on the criterion (Vosselmann, 1992). This means, they neither support the mapping between the model and the data nor contradict it. A wild card assignment is required for this approach because the input data cannot be assumed as complete.
The data is not complete because parts of the surfaces of the prototypes are occluded by other prototypes, a situation which leads to the junctions defined within the grammar section (2.3). Another reason for their occurrence lies in the surveying situation. For each of these cases there exist nothing mappable for some compartments within the model, which then will be mapped to virtual objects called wild cards.
Within our approach we use MDL as a criterion to measure the

quality of the mapping between the models generated by our grammars and the input data. One possible way is to derive the probabilities of occurrence $P(x)$ of any character x of an alphabet X, since the information $I(x)$ is defined as (Cover and Thomas, 2006):

$$I(x) = -logP(x) \tag{2}$$

Within the constraint graphs mentioned in section 2.2 the probabilities of occurrence for any edge and node in all occurring graphs have to be derived. Due to the large variety of buildings and a high impact of regional characteristics of building types, it is a difficult and cumbersome process to learn the required information from training data. Since any set of training data is only related to a specific region and not globally valid, these values are always biased. Under the assumption of optimal coding the probability can be replaced by the code length $L_C(x)$ of x encoded with the code C (Cover and Thomas, 2006):

$$L_C(x) = I(x) = -logP(x) \tag{3}$$

Up to this point the problem of deriving the a priori probabilities correctly is the same as in equation 2, because it is required to build an optimal code. A possibility to face this problem is the use of asymptotic optimal codes. One compression technique producing asymptotic optimal codes is the Lempel-Ziv-Welch-compression introduced by (Welch, 1984). A code is asymptotic optimal, if the redundancy approaches zero whenever the source code length tends to infinity (Lelewer and Hirschberg, 1987). While LZW-compression is a codebook based compression technique, an optimization of the initial codebook can be used to reduce the source code length required for initial learning. An initial codebook for the approach discussed within this paper contains the different geometric components and constraints forming the constraint graphs and it contains the constraint graphs of the primitives in use. These elements have to be entered in that way that the most frequent element appears first and the most infrequent appears last.
Within our approach the LZW-compression will be used to derive the required code length $L_C(x)$ of the model and its matched parts.

2.5 System architecture

In the last sections we have presented the components of the building reconstruction process. Now these components are integrated to obtain the whole procedure.
The process is composed of several steps, which correspond to scales. Starting with the coarsest scale, form one step to another the scale becomes more detailed. Usually we consider four scales: In the coarsest scale no. 1 the storeys except attics are detected. In scale no. 2 the attics are reconstructed, in scale no. 3 the larger building characteristics like dormers and balconies, and in scale no 4 doors and windows are detected.
For each scale, the components defined so far are specialized. A BG grammar is specified for each scale, obtaining different grammars BG_{scale} which differ in the prototypes and their semantics. Furthermore, the data constraint graph is specialized for each scale. Specific scale-dependent criteria are used to derive these constraint graphs.
The scale oriented procedure is depicted in figure 6; The procedure starts with scale 1 by matching the specialized PCG_{scale} with the specialized DCG_{scale}. Then the rules of the grammar BG_{scale} are applied as described in section 2.3, until there is no nonterminal symbol left. Now the scale is incremented and the procedure is repeated, until all scales are considered.

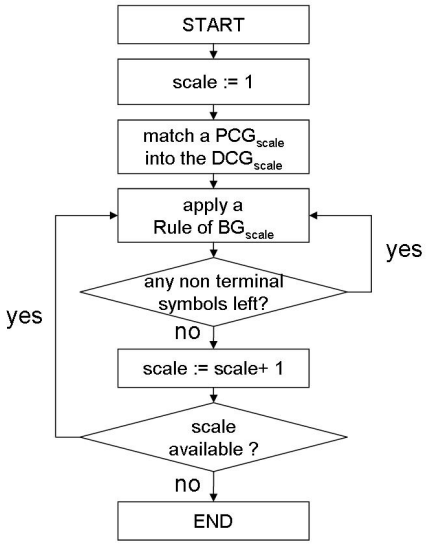


Figure 6: The building reconstruction procedure.

3 CONCLUSIONS AND FUTURE WORK

In this paper we have presented the outline of a procedure for automatic building reconstruction. It enables the extraction of semantic building models from heterogeneous input data which passed a preprocessing step. While the input data could be points from laser scanning or existing building models without semantic information, the preprocessing step produces 3D planar polygons and connected planes from the input data. The reconstruction itself is a multi step process where each step corresponds to a level of detail. Each reconstruction step uses a specific set of symbols, production rules and semantic rules. To guide the generation process and reduce the search space, the principle of minimum description length is employed to select the best alternative and it is used as a termination criterion. These steps are embedded in a control structure, which decides which step has to be performed next. Semantic information, gathered from ontologies are used to define the objects and get a priori information about their specific attributes. The system offers a mechanism to reconstruct buildings and their structural elements on a high level of detail in a generic and flexible way. Future work will include the implementation and evaluation of the presented approach. Another aspect is the extension of the set of available solid primitives within the grammars to be able to reconstruct e.g. pillars or downpipes. More reconstructible semantical parts of buildings have to be specified. Another point of future work will be the derivation of typical value ranges for the attributes and spatial extends of the reconstructed components. This knowledge seems to be helpful to improve the efficiency of the outlined process.

4 ACKNOWLEDGEMENTS

This research was supported by the German Research Foundation (Deutsche Forschungsgemeinschaft, DFG) as part of the bundle project entitled ”Abstraction of Geographic Information within the Multi-Scale Acquisition, Administration, Analysis and Visualization” (FO 180/10-1).

REFERENCES

Akaike, H., 1974. A new look at the statistical model identification. IEEE Transactions on Automatic Control 19(6), pp. 716–723.

Brenner, C., 2004. Modelling 3d objects using weak csg primitives. nternational Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences 35(3), pp. 1085–1090.

Brenner, C. and Haala, N., 1998. Fast production of virtual reality city models. IAPRS 32, pp. 77–84.

Carlson, C., Woodbury, R. and McKelvey, R., 1991. An introduction to structure and structure grammars. Environment and Planning B: Planning and Design 18(4), pp. 417 – 426.

Chomsky, N., 1959. On certain formal properties of grammars. Information and Control 2, pp. 137–167.

Cover, T. M. and Thomas, J. A., 2006. Elements of Information Theory. 2 edn, Wiley-Interscience.

Duarte, J., 2002. Malagueira Grammar - towards a tool for customizing Alvaro Siza's mass house at Malagueira. PhD thesis, MIT School of Architecture and Planning.

Eastman, C. M., 1999. Building Product Models: Computer Environments, Supporting Design and Construction. CRC.

Fischer, A., Kolbe, T. H., Lang, F., Cremers, A. B., Förstner, W., Plümer, L. and Steinhage, V., 1998. Extracting buildings from aerial images using hierarchical aggregation in 2d and 3d. Computer Vision and Image Understanding: CVIU 72(2), pp. 185–203.

Gülch, E., Müller, H. and Läbe, T., 1999. Integration of automatic processes into semi-automatic building extraction. In: Proceedings of ISPRS Conference "Automatic Extraction Of GIS Objects From Digital Imagery".

Grau, O., 2000. Wissensbasierte 3D-Analyse von Gebäudeszenen aus mehreren frei gewählten Stereofotos. PhD thesis, Universität Hannover.

Gröger, G., Kolbe, T. H. and Czerwinski, A. (eds), 2006. OpenGIS City Geography Markup Language (CityGML), Implementation Specification Version 0.3.0, Discussion Paper, OGC Doc. No. 06-057. Open Geospatial Consortium.

Grün, A. and Wang, X., 1999. Cybercity modeler, a tool for interactive 3-d city model generation ,. In: Fritsch and Spiller (eds), Photogrammetrische Woche 1999, Photogrammetrische Woche 1999.

Grünwald, P. D., Myung, I. J., Pitt, M. A., Balasubramanian, V., Lanterman, A. D., Hanson, A. J., Fu, P. C., P-Vitányi, Barron, A., Liang, F., Foster, D. P., Stine, R. A., Yamanishi, K., Rissanen, J., Tabus, I., Comley, J. W., Dowe, D. L., Jörnsten, R., Yu, B., Kontkanen, P., Myllymäki, P., Buntine, W., Tirri, H., Lee, M. D., j. Navarro, D., Charter, N. and Su, Y., 2005. Advances in Minimum Description Length. MIT Press.

Haala, N., 2005. Multi-Sensor-Photogrammetrie - Vision oder Wirklichkeit? PhD thesis, Universität Stuttgart.

Knuth, D. E., 1968. Semantics of context-free languages. Theory of Computing Systems 2(2), pp. 127–145.

Knuth, D. E., 1971. Top-down syntax analysis. Acta Informatica 1(2), pp. 79–110.

Kolbe, T. H., 1999. Identifikation und Rekonstruktion von Gebäuden in Luftbildern mittels unscharfer Constraints. PhD thesis, Hochschule Vechta.

Kolbe, T. H., Gröger, G. and Plümer, L., 2005. Citygml – interoperable access to 3d city models. In: Geo-information for Disaster Management. Proc. of the 1st International Symposium on Geo-information for Disaster Management.

Lelewer, D. A. and Hirschberg, D. S., 1987. Data compression. ACM Computing Surveys 19(3), pp. 261–296.

Müller, P., Vereenooghe, T., Ulmer, A. and Gool, L. V., 2005. Automatic reconstruction of roman housing architecture. In: International Workshop on Recording, Modeling and Visualization of Cultural Heritage, Balkema Publishers (Taylor & Francis group).

Müller, P., Wonka, P., Haegler, S., Ulmer, A. and Gool, L. V., 2006. Procedural modeling of buildings. In: Proceedings of ACM SIGGRAPH 2006 / ACM Transactions on Graphics (TOG), ACM Press, Vol. 25number 3, pp. 614–623.

OGC, 2006. Ogc web services phase 4. Technical report, Open Geospatial Consortium. http://www.opengeospatial.org/projects/initiatives/ows-4 (last visited 30.03.2007).

Parish, Y. and Mueller, P., 2001. Procedural modeling of cities. In: E. Fiume (ed.), Proceedings of ACM SIGGRAPH 2001, ACM Press / ACM SIGGRAPH, pp. 301–308.

Prusinkiewicz, P. and Lindenmayer, A., 1990. The Algorithmic Beauty of Plants. Springer.

Rissanen, J., 1978. Modeling by the shortest data description. Automatica 14, pp. 465–471.

Rottensteiner, F., 2001. Semi-automatic extraction of buildings based on hybrid adjustment using 3D surface models and management of building data in a TIS. PhD thesis, Technische Universität Wien.

Schwarz, G., 1978. Estimating the dimension of a model. Annals of Statistics 6(2), pp. 461–464.

Stilla, U. and Jurkiewicz, K., 1999. Reconstruction of building models from maps and laser altimeter data. In: P. Agouris and A. Stefanidis (eds), Integrated spatial databases:Digital images and GIS, Springer, pp. 34–46.

Stiny, G., 1980. Introduction to shape and shape grammars. Environment and Planning B 7, pp. 343–361.

Stiny, G., 1982. Spatial relations and grammars. Environment and Planning B 9, pp. 313–314.

Vosselman, G. and Dijkman, S. T., 2001. 3d building model reconstruction from point clouds and ground plans. International Archivs Photogrammetry and Remote Sensing XXXIV part 3/1, pp. 339–345.

Vosselmann, G., 1992. Relational matching. Vol. 628, lect. not. comp. sci. edn, Springer.

Wahl, R., Guthe, M. and Klein, R., 2005. Identifying planes in point-clouds for efficient hybrid rendering.

Welch, T. A., 1984. A technique for high-performance data compression. Computer 17(6), pp. 8–19.

Wonka, P., Wimmer, M., Sillion, F. and Ribarsky, W., 2003. Instant architecture. ACM Transactions on Graphics 22, pp. 669–677.

METHODS FOR AUTOMATIC EXTRACTION OF REGULARITY PATTERNS AND ITS APPLICATION TO OBJECT-ORIENTED IMAGE CLASSIFICATION

Luis A. Ruiz, Jorge A. Recio, Txomin Hermosilla

Department of Cartographic Engineering, Geodesy and Photogrammetry
Polytechnic University of Valencia. Camino de Vera s/n. 46022-Valencia (Spain)
laruiz@cgf.upv.es

KEY WORDS: Regularity patterns, object-oriented classification, image analysis, semivariogram, Hough transform, Fourier analysis.

ABSTRACT:
Detection and quantification of regularity patterns are important structural aspects for object-oriented classification of images for geo-databases updating. Four image processing methods are analysed and evaluated for this purpose: semivariogram analysis, the Hough transform, the histogram of minimum distances, and Fourier space descriptors. In addition, several features are extracted from each method and evaluated for classification of regular and non regular parcels in a rural environment. The classification has been performed by using the C5 algorithm, based on data mining techniques. A total of 276 objects have been evaluated using the cross-validation method. After selecting the most discriminant features, a land use object-oriented classification has been performed, which includes some spectral and textural features in the model. The results show that the features based on the semivariogram and the Hough transform are the most efficient for detecting regularity patterns. The combination of the three groups of features (spectral, textural and structural) clearly improves the classification of parcels, which is encouraging for automated land use cartography updating.

1. INTRODUCTION

A strategic issue brought on by the evolution of the geoinformation systems is the application and development of new methods that allow for an efficient generation and update of geographic databases by means of integrating different types of data. In this sense, aerial and high resolution satellite images play an important role, since they can be acquired with a high frequency, they offer a variety of spectral information - including infrared bands- and the image processing methods are evolving and being improved in order to automate some tasks that have traditionally been done by interpretation and field work.

Usually, classification techniques are applied for updating cartographic databases from images, and very often object-oriented classification methods are used to avoid errors related to the borders of landscape elements. This means that each object is classified independently, from different features that are extracted in a specific manner. The limits of the objects can be defined by image segmentation, using cadastral units or other existing georeferenced databases, which changes the traditional *per-pixel* approaches based on *one pixel-one value*.

In the characterisation of objects in the image, different types of features can be used, such as spectral, textural, and/or structural. The first are directly based on the values of the spectral bands, or on indices or combinations derived from them. Textural features attempt to describe the spatial relationships of the data values within an object; and the structural features provide information about specific patterns or arrangements of landscape elements contained in the object. These are more related to the way in which the humans interpret and understand the scenes. In this context, a relevant property that helps us to describe the landscape is the presence, degree and type of regularity patterns that provide the final configuration of an object. Therefore, the definition of variables or features extracted from images that give us a quantification of the regularity patterns is important to improve the automatic classification of objects, with the final aim of increasing the efficiency of the processes to update land cover information systems.

The objectives pursued in this work are the development and extraction from images of quantitative parameters or indicators that allow for the identification of regularity in agricultural objects (e.g. cadastral units, parcels, etc.), the evaluation of these parameters and the selection according to their efficiency for the classification of the objects. An application of the methods is performed using aerial images from a rural environment in the Mediterranean area of Spain.

2. PRE-PROCESSING OF DATA

The data used for this study were 0.5 meters resolution digital aerial orthoimages acquired in August 2005 using the DMC (*Digital Mapping Camera*). This is a CCD sensor with three bands in the visible part of the electromagnetic spectrum (0.4-0.58 μm, 0.50-0.65 μm, and 0.59-0.675 μm), one in the NIR (0.675-0.85 μm) and a panchromatic band. In addition, the definition of the objects to classify was based on the vectorial limits obtained from regular cadastral units or parcels from the area of *Castellón*, on the Mediterranean coast of Spain. A set of 276 rural parcels was selected for the test, representing different types of land use/land cover, such as *Citrus* orchards, other crops, forest, shrub, fallow and barren soil. Since the principal aim was to detect the presence of regular patterns, all the objects were pre-classified as regular or non regular, maintaining a balanced proportion of both types.

With the exception of the feature extraction method based on the analysis of the semivariogram, the other three methods proposed use pre-processed binary images in which the position of trees has been estimated. This is because the principal factor of regularity in rural landscapes is due to the relative position of the trees in the terrain. The location of the trees inside an object

is based on the *local maximum filtering* method (Pouliot, 2002; Nelson, 2005; Wulder, 2000), which is based on the assumption that reflectance is highest at the tree apex and decreases towards the crown edge (Wulder, 2000). Moving a kernel over the image, trees are found when the central value in the kernel window is higher than all other values. The scene illumination has an important influence on local maxima position, displacing their position from the real apex location. This displacement has not factual effects because it equally affects to all the maxima located.

In this study, since most of regular features present in the images used were due to different crop arrangements, a local maximum method was applied over NDVI images using a circular kernel with variable diameter size, ranging from 9 pixels to 23 pixels. The size of this circular neighbourhood was determined as the position of the first maximum value of the semivariogram curve, which is computed for each particular object. This position is in accordance with the mean tree separation in the parcel, ensuring that a tree, but no more than one, is present inside the window, assuming a regular distribution of the trees in the parcel. Additionally, a minimum threshold was defined in order to avoid the selection of maximum points in non-vegetated areas. In those parcels with a low NDVI mean value, the tree search consists of local minimum finding over the NIR band. This variation is applied to locate young trees recently planted. The result is a binary image, where each located tree is represented by a pixel (see examples in figures 2 and 3).

3. METHODS FOR EXTRACTION OF INDICATORS OF REGULARITY PATTERNS

Four methods were used to extract regularity indicators: Analysis of the semivariogram, the Hough transform, the histogram of minimum distances between trees, and Fourier space descriptors. The first uses the red band as input, while the last three use the binary image containing the estimated position of trees as input data.

3.1 Analysis of the Semivariogram

The semivariogram quantifies the spatial associations of the values of a variable, and measures the degree of spatial correlation between different pixels in an image. The experimental semivariogram is defined as:

$$\gamma(h) = \frac{1}{2N}\sum_{i=1}^{N}\left[Z(x_i) - Z(x_i + h)\right]^2$$

where $Z(x_i)$ represents the gray level for a generic pixel in the location x_i; N is the number of pixels considered; and h is a vector that represents the distance between pixels in a particular direction. Several authors have used information extracted from the semivariogram to incorporate texture into image classification (Miranda et al., 1998; Carr and Miranda, 1998; Chica-Olmo and Abarca, 2000; Maillard, 2003; Durrieu et al., 2005).

One object-specific semivariogram was computed for every parcel. Figure 1 shows some examples of curves associated with different types of land use, and their relationship with the regularity of the landscape elements. Several parameters were measured to extract a set of features: the position of the first maximum and its value, the slope between the first maximum and the fist minimum, and the slope between the first minimum and the second maximum. In addition, other parameters were used as reported by Durrieu et al. (2005).

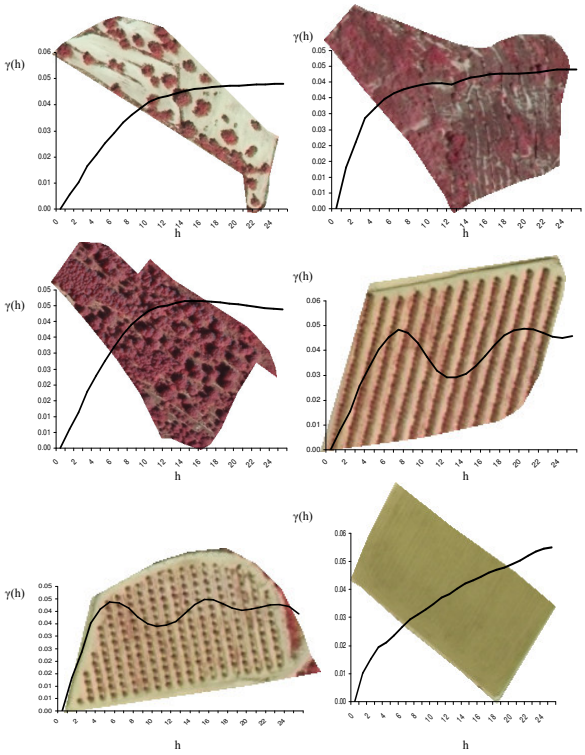


Figure 1.- Examples of six parcels with different use, and their respective experimental semivariogram *γ(h)* superimposed.

3.2 The Hough Transform

This method is based on the transformation of the coordinates of points from a cartesian image space (X,Y) to a polar coordinate space (ρ, θ), where ρ represents the distance from the origin to a point, and θ its angle with respect to the X axis. The points in cartesian space correspond to sinusoids in polar space, and a line in this space is defined by a point where several sinusoids are intersected (figure 2).

The histogram of angular values ranging from 0º to 180º is computed, the two maxima corresponding to the principal directions or alignments of trees in the parcels when some regularity in their spatial arrangement exists (figure 2). A total of 15 parameters related to the regularity of the distribution of trees were defined based on this transformation and the histogram of orientations. Some of them, selected upon the criteria described in section 4, are specified in table 1.

In addition to the variables for classification, this method allows for the determination of the distance between trees, following the two principal directions (see detail in figure 3). This information can be useful for inventories, as well as to discriminate different species of agricultural trees (e.g., *Citrus*, olive trees, etc.).

3.3 Histogram of distances

This method consists of the computation of all distances among points extracted after the application of the local maxima method, and the selection of the minimum distance for each point. Then, a histogram of all the minimum distances is created for each parcel. The mean of this histogram provides information about the average separation of the elements, while

its standard deviation gives information about the presence of regularity in the distribution. If the standard deviation is high, the points are expected to be randomly distributed, but if it is low, it means that the points are located at a similar distance from each other, inferring some regularity pattern. The skewness and kurtosis of the distribution were also initially considered, but they did not show any relevance after the evaluation.

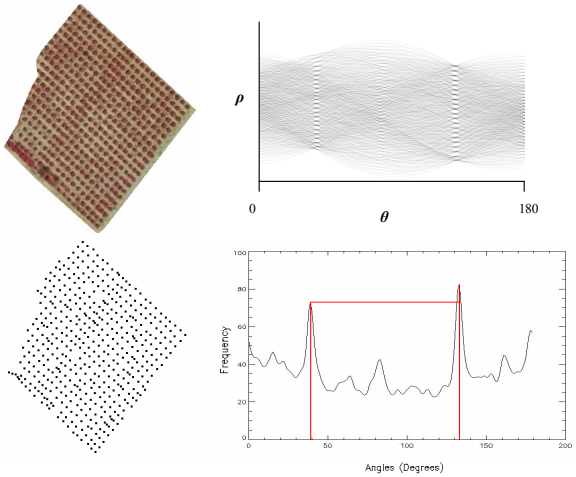


Figure 2.- First column: A parcel of *Citrus* trees (above) and their location after the application of the local maxima method (below). 2nd column: Representation of points on the space defined by Hough coordinates θ and ρ, and the histogram of the orientations (θ), with the two principal directions enhanced.

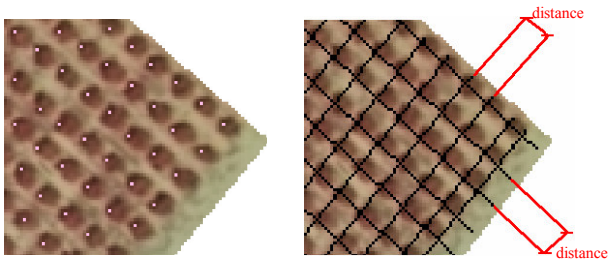


Figure 3.- Detail of identification of tree positions (left); extraction of the 2 principal directions of tree alignnment and the regular distance between trees (right).

3.4 Fourier Space Descriptors

After the application of the 2D Fourier transformation of an image, the Fourier spectrum reveals the existence of regular patterns in an organised manner. Based on the concept of the descriptors proposed by Gonzalez and Woods (1993), the regularity parameters have been extracted as follows: First, the direction of the maximum value of $S(\theta)$ is obtained, being

$$S(\theta) = \sum_{r=1}^{R} S_r(\theta)$$

where r is the radius from the origin of the frequency space and θ is the angle of the direction with respect to the X axis, which ranges from 0º to 180º. Once θ_{max} is obtained, a profile of the Fourier spectrum values is defined starting at the origin and following that direction (figure 4, left). Then the local maxima values of $S_{\theta_{\max}}(r)$ are computed (figure 4, right), representing the frequencies at which the main regular patterns occur.

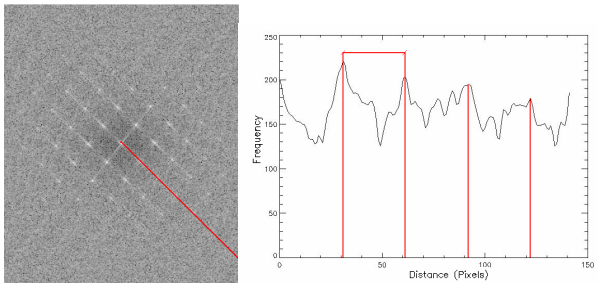


Figure 4.- Fourier spectrum of the image from figure 2 (left) and the profile of the direction $S_{\theta_{\max}}(r)$ of maximum change,.

The extraction of the maxima of these positions and the standard deviations of the differences among them in the frequency domain, provides a valuable information about the existence of regularity patterns in the images, being these the features extracted (table 1).

4. SELECTION OF VARIABLES AND CLASSIFICATION

4.1 Pre-selection of Variables by multivariate analysis

In the case of the methods based on the Hough transform and the semivariogram analysis, a high degree of redundancy existed in the total number of features extracted initially. Therefore, a statistical selection based on the stepwise discriminant analysis was applied over these two methods in particular. This was done in order to select a reduced number of features containing the most significant information relative to the regularity patterns in the objects.

The results of this selection process, in which all 276 parcels were used, are shown in the graph of figure 5. The overall accuracy obtained in the classification of regular and non regular patterns is represented for each of the four methods tested, as a function of the successive increase in the number of variables included in the classification model.

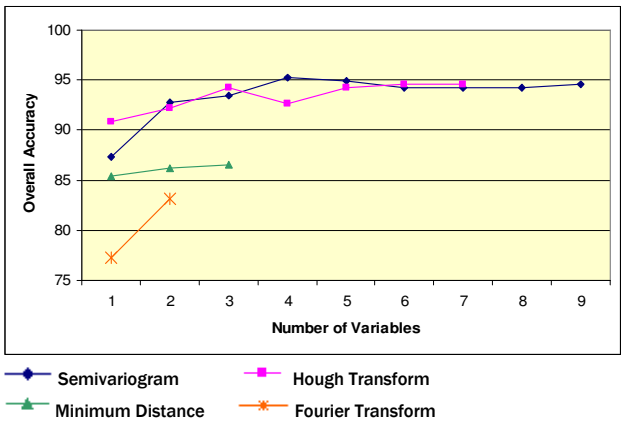


Figure 5.- Overall accuracy in the classification of regular/non regular parcels obtained using the four different sets of structural features tested.

By using the methods based on the semivariogram and the Hough transform, the selection of 4 or 5 variables is sufficient to obtain approximately 95% accuracy, while the methods based on the Fourier transform and in the minimum distance are less efficient, with an accuracy of about 85%, but using only 2 and 3 variables, respectively. Considering these results obtained by the discriminant analysis method, a combined set of 14 variables was selected from the initial set of features. These variables are described in the table 1.

Semivariogram Variables
Slope of the normalized semivariogram between the first maximum and the first minimum
Slope of the normalized semivariogram between the first minimum and the second maximum
$Smp3 = 1 - \left(\frac{\gamma(h_{max_2})}{\gamma(h_{max_1})} \right)$ Decay/increase of the semivariogram cycle*
Position of the first maximum
$Gp5 = \frac{\gamma(h_3) - 2\gamma(h_2) + \gamma(h_1)}{2h^2}$ Concavity/Convexity value at h_2 (variability in short distances)*
Hough Transform Variables
Proportion of points included in the principal direction with respect to the total points
Angular difference between the principal directions
Standard deviation of distances between points included in the principal direction
Proportion of points on the secondary direction with respect to the total aligned points
Minimum Distance Variables
Maximum frequency of the minimum distance between points normalized by the total area
Skewness of the distribution of minimum distances
Kurtosis of the distribution of minimum distances
Fourier Transform Variables
Normalized mean of the maxima detected in the direction of maximum change
Standard deviation of the distances between maxima in the same direction

(*) *From Durrieu et al., 2005.*

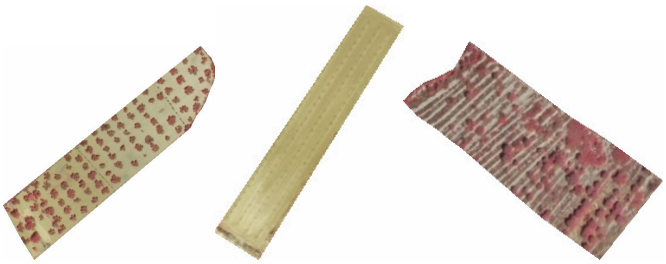
Table 1.- Group of variables describing regularity pre-selected after the application of the stepwise discriminant analysis method.

4.2 Decision tree binary classification

A decision tree is a set of conditions organized in hierarchical structure in such a way that the assignation of a class to an object can be determined following the conditions fulfilled by the object. The goal is to learn how to classify objects by analyzing a set of training samples whose classes are known. Classes are mutually exclusive labels. The objects are represented as vectors that give the numerical values of a collection of properties or features. Learning input consists of a set of such vectors, each belonging to a known class, and the output consists of a mapping from attribute values to classes.

A decision tree can be constructed from a set of rules by a *divide and conquer* strategy. A test with mutually exclusive outcomes is used to partition the training set into subsets that are more homogeneous than the initial set. For each potential test, the impurity degree of the generated subsets is calculated, and the test which generates the most homogeneous subsets is selected. The algorithm iterates until the subset elements belong to the same class. The algorithm employed is known as C5 and its splitting criterion is the *gain ratio*. The gain ratio criterion (Quinlan, 1996) assesses the desirability of a test as the ratio of its information gain to its split information, and the split with maximum gain ratio is selected.

A binary classification (regular/non regular) was made by means of the C5 algorithm and using the 14 selected features previously described. From the 276 parcels tested, only 6 were misclassified: 4 regular were classified as non regular, and 2 non regular as regular. Examples of these errors are shown in figure 6. Usually, the errors are due to the presence of unclear regular patterns (case of parcel *b*, containing very small aligned trees), or the presence of pseudo-patterns (case of parcel *c*, with terraces).



a) b) c)

Figure 6.- Examples of misclassification of objects with the decision-tree using the selected structural features. a) and b) regular objects classified as non regular. c) Non regular objects classified as regular.

These results show that the set of proposed features contains fundamental information about the distribution of patterns in images, allowing for the discrimination of regular and non regular parcels. In order to go further in the process of automated classification, another test was made including these structural features in a land use object-oriented classification process, as described in the next paragraph.

5. APPLICATION TO OBJECT-ORIENTED LAND USE CLASSIFICATION

An important application based on the use of structural information extracted from images is the automated classification of images for updating land use databases. Using the same set of cadastral parcels, an object-oriented classification of land use in a rural environment was made, considering the following 5 classes: *Citrus* orchards, Olive and/or carob trees (*Ceratonia siliqua*), Bare soil, Forest, and Shrub. In addition to the 14 structural features selected (table 1), 6 spectral features and 9 texture features were used as initial variables for the generation of the decision tree. The spectral features used were: the mean values of the NIR and red bands, and their standard deviations, as well as the mean NDVI and its standard deviation. The texture features used were based on the first order histogram (skewness and kurtosis), some of the texture features proposed by Haralick et al. (1973), that are extracted from the grey level co-occurrence matrix (uniformity, entropy, contrast, correlation and inverse difference moment), and the mean and standard deviation of the *edgeness* parameter. A complete description of the texture variables, as well as the selection process applied, is reported by Ruiz et al. (2004). The combination of these three groups of features and their object-oriented extraction is analysed by Recio et al. (2006).

The decision tree was generated by the C5 algorithm, using the described set of variables and the 5 classes described. The evaluation was performed using the cross-validation method, dividing the 276 objects into 10 equally sized sets, and repeating the same process ten times. Each time, 9 sets were used as training samples to compute the decision tree, and the other set was used for testing purposes. The results are shown in table 2. The overall accuracy was 89.9%, which means that the global information extracted from the images seems to be very efficient for the automated classification of such landscapes.

	Olive & Carob tree	*Citrus* orchards	Bare soil	Forest	Shrub	Total
Olive & Carob tree	56				5	61
Citrus orchards		104	5			109
Bare soil		2	25			27
Forest		1		27	5	33
Shrub	3	1		6	36	46
Total	59	108	30	33	46	276

Table 2.- Error matrix of the object-oriented classification of 5 different types of rural parcels using spectral, textural and structural features. Columns represent classified values and rows reference values. (Overall accuracy: 89.9%)

Some errors were due to the presence of very young orange trees (see example in figure 6b), that were confused with bare soil. In some cases, olive trees without a regular arrangement were classified as shrub. Another type of error was due to the difficulty in distinguishing between shrub and forest that sometimes occurs in the photointerpretation of the images. In general, it seems that most of the errors were produced in parcels where the visual interpretation of the type of cover has a high degree of difficulty.

6. CONCLUSIONS

An exhaustive set of structural features for the analysis of regularity of image objects, based on four different approaches, has been proposed. After the statistical selection of the most relevant features from the set, those extracted from the experimental semivariogram of each object, and those based on the Hough transform resulted to be particularly efficient for the discrimination of parcels containing elements with a regular spatial distribution pattern, usually due to the linear arrangement of trees in agricultural fields. Using a decision tree classification method based on the C5 algorithm, the final rate of error was only 2.2%, which shows that the regularity can be efficiently characterised using these image processing methods.

The results are particularly interesting for the application in automated land use object-oriented classification of images. In this sense, a test has been performed for the classification of aerial image objects defined by cadastral units, including spectral, texture and structural features extracted from the images. The overall accuracy obtained for a problem of 5 classes is of 89.9%, which is encouraging for the use of these techniques in more complex problems.

Finally, the application of object-oriented classification using a combination of complementary variables is a promising method in order to advance in the incorporation and standardisation of automated methods for updating geo-databases and map production.

REFERENCES

- Carr, J.R., Miranda, F.P., 1998. The semivariogram in comparison to the co-occurrence matrix for classification of image texture. IEEE Transactions on Geoscience and Remote Sensing, 36 (6), 1945-1952.
- Chica-Olmo, M., Abarca-Hernández, F., 2000. Computing geostatistical image texture for remotely sensed data classification. Computers & Geosciences, 26, 373-383.
- Durrieu, M., Ruiz, L.A., Balaguer, A. Analysis of geostatistical parameters for texture classification of satellite images. Procs. of the 25th EARSEL Symposium: Global Developments in Environmental Earth Observation from Space (ISBN: 9059660420): pp. 11-18, 2005.
- Gonzalez, R.C, Woods, R.E., 1993. Digital Image Processing. Addison-Wesley Publishing.
- Haralick, R.M., K Shanmugam and Dinstein, 1973. Texture features for image classification. IEEE Trans. on Systems, Man. and Cibernetics. SMC-3 : pp. 610-622.
- Maillard, P., 2003. Comparing texture analysis methods through classification. Photogrammetric Engineering & Remote Sensing, 69 (4), 357-367.
- Miranda, F.P., Fonseca, L.E.N., Carr, J.R., 1998. Semivariogram textural classification of JERS-1 SAR data obtained over a flooded area of the Amazon rainforest. International Journal of Remote Sensing, 19 (3), 549-556.
- Nelson, T., Boots, B. and Wulder, A., 2005. Techniques for accuracy assessment of tree locations extracted from remotely sensed imagery. Journal of Environmental Management 74, pp. 265–271.
- Wulder, M., Niemann, K. O. And Goodenough, G. D. 2000. Local maximum filtering for the extraction of tree locations and basal area from high spatial resolution imagery. Remote Sensing of Environment, 73:103-114.
- Pouliot, D.A. King, D.J., Bell, F.W., and Pitt. D.G. ,2002. Automated tree crown detection and delineation in high-resolution digital camera imagery of coniferous forest regeneration. Remote Sensing of Environment, 82(2-3):322-334.
- Quinlan, J.R. 1996, Improved use of continuous attributes in C4.5 Journal of Artificial Intelligence Research 4, 77-90
- Recio, J.A., Ruiz, L.A., Fdez-Sarría, A., Hermosilla, T., 2006. Integration of multiple feature extraction and object oriented classification of aerial images for map updating. II Recent Advances in Quantitative Remote Sensing. Torrent (Valencia), pp 391-396.
- Ruiz, L.A, Fdez-Sarría, A., Recio, J.A., 2004. Texture feature extraction for classification of remote sensing data using wavelet decomposition: A comparative study. International Archives of Photogrammetry and Remote Sensing. Vol. XXXV, part B, pp. 1109-1115.

ACKNOWLEDGMENTS

This research was partially funded by the Spanish Ministry of Education and Science and the FEDER, in the framework of the projects CTM2006-11767/TECNO and CLG2006-11242-C03-03/BTE. We would like to thank the Instituto Cartográfico Valenciano for the support and data provided to complete this research.

EXTRACTION OF LANDCOVER THEMES OUT OF AERIAL ORTHOIMAGES IN MOUNTAINOUS AREAS USING EXTERNAL INFORMATION

Arnaud LE BRIS, Didier BOLDO

Institut Géographique National (IGN) - Laboratoire MATIS
2-4 Avenue Pasteur
94165 SAINT-MANDE Cedex - FRANCE
arnaud.le-bris@ign.fr, didier.boldo@ign.fr

KEY WORDS: Image classification - Landcover extraction - Data fusion - Uncertainty management - National base map - Landcover-knowledge based interpretation - Mountainous landcover extraction - MAP classification method

ABSTRACT:

In mountainous areas, the landcover extraction out of orthoimages through semi-automatic classification is limited by several factors (such as large shadow areas, radiometric similarities between different themes, inhomogeneous radiometry among regions of the same class...). Image information is not sufficient to separate the different classes. Nevertheless, good results can be obtained by dividing each landcover class "*c*" into two subclasses "*c in shadow*" / "*c not in shadow*" and introducing external information in the classification process. This information can be an older or more generalized database, geographic knowledge concerning the links between relief and landcover, or prior information concerning shadows. This external knowledge is then interpreted in terms of *a priori* probabilities and merged with radiometric information from the image in a *MAP* per region classification process. Besides, the results can also be improved by the use of combinations of channels calculated from the initial image.

1 INTRODUCTION

The French National Mapping Agency (IGN) is designing a new 1/25 000 national base map. This new map must be produced from the IGN's national digital databases through a fully digital process, which must be as automatic as possible. Nevertheless, some legend items are not present in the IGN's digital databases whereas they are necessary to obtain a correct map. In the special case of mountainous areas, important landcover information lacks from the IGN's digital databases. More precisely, information about the themes rocks, screes and glaciers is not available whereas their representation is necessary in mountainous area maps. As a consequence, this missing information must be gotten from other sources. It can be extracted either from present maps or from aerial images. As the cartography of these missing landcover themes is not up to date in present maps, the second solution has been chosen (Le Men et al., 2002). Furthermore, it offers the possibilty to be used afterwards for map updating.

To sum up, the chosen solution consists in extracting landcover information out of aerial orthophotos through a supervised classification method in two steps : the images are firstly segmented (by the tool presented in (Guigues et al., 2006)) in homogeneous regions which are then classified knowing statistical radiometric models of the classes previously computed from training data (as described in (Trias-Sanz and Boldo, 2005)). The orthoimages come from the IGN's orthoimages database, which contains digital colour orthophotos with 3 or 4 (red - green - blue - near infrared) bands and with a 50cm resolution. To obtain a landcover classification of the whole area, the six following classes are defined : lakes - forest - pasture - rocks - screes - glaciers.

This landcover supervised classification problem could seem quite easy since few themes are seeked. However, it is perturbed by several phenomena such as important radiometric variations or shadow areas, so that image information is not sufficient to separate the different classes. A previous study (Le Men et al., 2002) has shown it was possible to improve the results by correcting the radiometry in shadow zones and by introducing external knowledge in the classification process. Nevertheless, this correction of radiometry was limited by uncertainties on the DTM and on images'capture time. Therefore, a simpler variant of this method without radiometric corrections has been tested in a new study. It is presented in this paper. The results obtained with this new method are equivalent to these obtained with the old one and are suitable for the base map's purposes.

In this paper, the different radiometric phenomena perturbing the classification will firstly be presented. Secondly, solutions consisting notably in taking knowledge from external sources into account, in taking shadows into account by the means of classes divided into two "shadow/non shadow" subclasses and in using combinations of derived channels computed from the orthoimages'bands will be proposed. The classification method and the way external information is taken into account will then be developped. In last section, experimental results will be presented.

2 PROBLEMS

In mountainous areas, an automatic land-cover extraction is perturbed by several phenomena causing misclassifications : image information is not sufficient to separate some classes.

2.1 Shadow areas

There are very large shadow areas in the images because of the strong variations of the relief as on figure 1. The landcover themes concerned by shadows are mostly rocks, screes and glaciers. As the radiometry of pixels belonging to a same landcover class is obviously completely different whether they lie in shadow or not, it is necessary to take the shadow areas into account in order to obtain a correct classification of the whole zone.

2.2 Radiometric variations inside a class

The radiometry of pixels belonging to a same class greatly varies from a part of the image to another. These variations can be :

Figure 1: Example of large shadow area : the radiometry of pixels of a same class is different whether they lie in shadow or not

- "natural" - It means related to changes, for instance in vegetation or geology inside the image area.
- related to illumination variations - In mountainous areas, these variations are very important because of the rough relief : the sky illumination and the ground illumination can greatly vary from a point to an other as on figure 2. The problems of shadows mentioned above are an extreme situation of illumination effects.

Figure 2: Example of illumination variations due to the relief

- "artificial" - The classified orthoimages are in fact a mosaic of orthorectified aerial photographs which have not been captured at the same time (or even the same day). Moreover, they have undergone several radiometric treatments (such as image dodging...) which have sometimes increased variations of radiometry inside a same class from a part of the orthoimage to an other.

2.3 Classes with similar radiometry

Distinct classes can have similar radiometric distributions, as for example some screes (especially riverbed screes) which are almost as light as glaciers or lakes which are often difficult to distinguish from rocks in shadow, or even rocks and screes. Besides, this phenomenom is increased by the variations of radiometry inside a same class previously described.

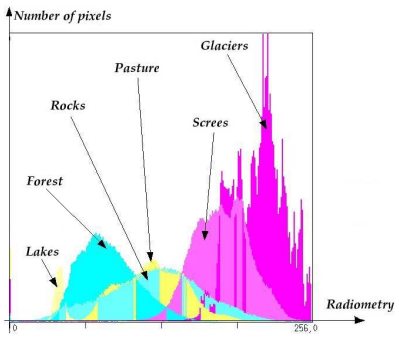


Figure 3: Image histogram for "value" channel.

3 PROPOSED SOLUTIONS

3.1 Shadow/non shadow classes

As it was said in the previous section, it is necessary to take into account the shadow areas to obtain a correct classification of the whole image.
A first way to achieve this could consist in correcting the radiometry in shadow areas (after having detected them). However, in the present case, this correction would be limited by several uncertainties : the orthoimage to treat is a mosaic of merged aerial orthophotographs which have undergone several (sometimes hand-made) unknown radiometric treatments and of which the precise time of capture is no more available. Moreover, the accuracy of the available digital terrain model is about 10-20m in mountainous areas. Nevertheless, a method to deal with these uncertainties to correct the radiometry in shadows areas was proposed and used in a previous study (Le Men et al., 2002).
As the main problem with shadows is the fact that the radiometric model of a class will be completely different in shadow and in light, the chosen solution consists in dividing each class "*C*" in two classes "*C in shadow*" and "*C in light*" so that two distinct models are obtained for each landcover theme. At the end of the classification process, the two subclasses "*C in shadow*" and "*C in light*" are merged.
However, the approximate time of image capture and the DTM can be used to give each pixel a probability of being in shadow (see 3.2.3).

3.2 Introduction of external knowledge in the classification process

Because of the important radiometric variations inside a class and the radiometric similarities between distinct classes, the image information is not sufficient to obtain a correct classification. A previous study has shown that the introduction of external knowledge in the classification process could improve the results. Furthermore, it can help to obtain a more generalized result, which is usefull in the present mapping context where only zones with a cartographic meaning are required. This information is interpreted as *a priori* probabilities in a MAP classification method (described in 4.2) (Trias-Sanz, 2006) (Le Men et al., 2002).

3.2.1 Information related to the relief In mountainous areas, the landcover is strongly related to the relief, it means to altitude, slope and orientation. Those variables are easily computed from the DTM. As a consequence, knowing geographic information such as the lowest and highest limits of the landcover themes or the influence of orientation on landcover (especially glaciers and forest), it becomes possible to define for each landcover theme a probability model function of those variables. Such a model is proposed in (Le Men et al., 2002) from (Elhai, 1968) and (Lacambre, 2001). It consists of two distinct models P_{alti} and P_{slope} (respectively one for altitude and another for slope information) made of the following simple piecewise linear functions drawn in figure 4.
As the orientation has an important influence only on forest and glaciers, this is taken into account only for these themes. Probability model knowing altitude, probability model knowing slope and influence of orientation are merged by the following formulas :

P_{model}(*forest|altitude=h,azimuth=az,slope=s)*=
P_{alti}(*forest|altitude=h+s* $\cdot cos(az)) \cdot P_{slope}$(*forest|slope=s)*
P_{model}(*glacier|altitude=h,azimuth=az,slope=s)*=
P_{alti}(*forest|altitude=h-s* $\cdot cos(az)) \cdot P_{slope}$(*glacier|slope=s)*
P_{model}(*t|altitude=h,azimuth=az,slope=s)*=
P_{alti}(*t|altitude=z)* $\cdot P_{slope}$(*t|slope=s)* for another theme *t*

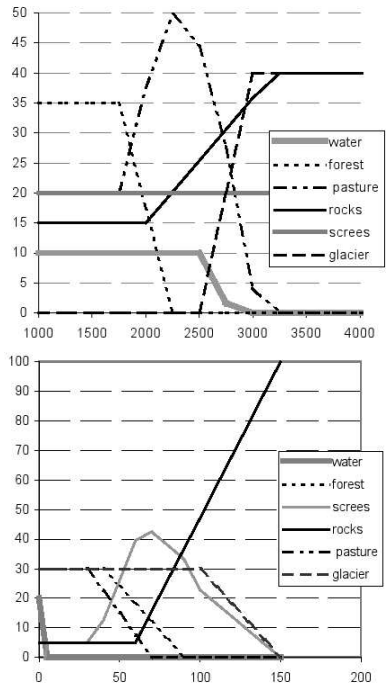


Figure 4: Prior probability (in %) to find the different landcover themes knowing altitude (in meters) and slope (in %)

The final probability *P(theme=to|altitude,orientation,slope)* is equal to $\frac{P_{model}(theme=to|altitude,orientation,slope)}{\sum_{t \in legend} P_{model}(theme=t|altitude,orientation,slope)}$
For example, the probability to find rocks knowing the DTM is shown on figure 5.

3.2.2 Other databases Knowledge from other available data bases can also be used.
In the present case, prior information from CORINE Land Cover 2000 (CLC2000) database was introduced in the classification process. This digital landcover European geographic database has been made from satellite images (captured in the year 2000) by means of photointerpretation (CORINE Land Cover, last visited on the 31st of January 2007) (Bossard et al., 2000).

Its scale (1/100 000) is smaller than the one of the national base map (figure 6). As a consequence, this information must be considered as imprecise since the CLC2000 regions are too generalized and may contain distinct themes of the base map classification.
Furthermore, the semantic precision is different from the one of our six items legend. CLC2000 is sometimes more precise - for instance, in CLC2000, different kinds of forest are discriminated - or on the contrary less precise - for example, CLC2000's "rocks" class includes rocks and screes. It also offers "intermediate" themes such as a sparse vegetation class which concerns screes, pasture, rocks... Therefore the introduction of prior information from CLC2000 in the classification process must deal with those uncertainties. That's why CLC2000 is interpreted in terms of probability with an empirical probability model : for each CLC2000 item $T_{CLC2000}$ and each classification class $T_{classif}$, a probability value $P(T_{classif}|T_{CLC2000})$ is empirically fixed. Several such models have been tested.
The geometric uncertainties have not been taken into account since no general rule (such as "a CLC2000 class A region always goes over a CLC2000 class B region") has been noticed.
An older database could also be used (in an updating context for instance). In this particular case, the uncertainty comes from evolution and also mostly lies in the geometry of the regions. As a consequence, it could be modeled by a probability function such as a regression or a progression function.

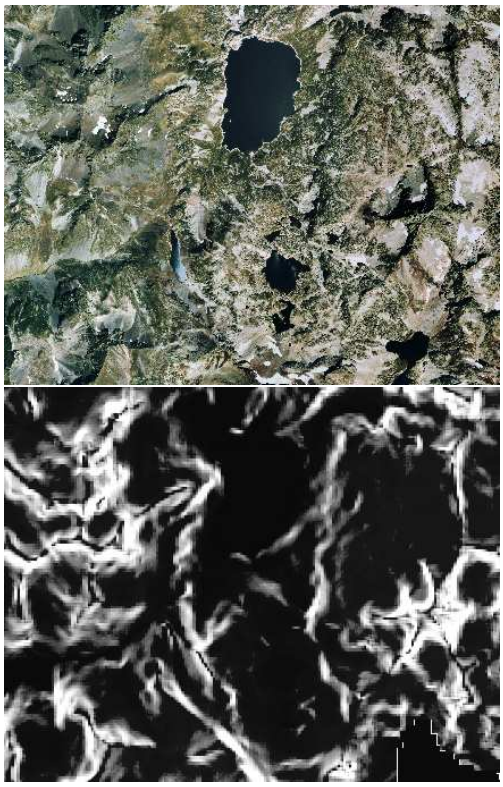


Figure 5: Prior probability to find rocks knowing the DTM

Figure 6: CLC2000 is a smaller scale database with a different nomenclature

3.2.3 Shadow/non shadow prior information Even though the available information is not sufficient to precisely detect shadows, it is possible to use it to compute a prior probability for each pixel of the image to lie in shadow knowing the DTM (figure 7). The exact capture times of the different aerial photographs merged in the orthoimage are not precisely known, but the beginning and final time of data capture of all these images are known. So a probability for each image pixel to be in shadow can be computed with the following method :
- Every five minutes between the beginning and the end of the images'acquisition, the Sun's position is computed and then the

shadows are estimated knowing the Digital Terrain Model : a pixel (it means a point of the DTM) is in shadow if it is hidden from the Sun by another point of the DTM.
- In the end, the probability that a pixel is in shadow during the data acquisition interval is computed as the number of five minutes periods (of this interval) during which it lies in shadow divided by the total number of five minutes periods of this interval. No special care is taken concerning the DTM precision.

Figure 7: Shadow prior probability knowing DTM

3.3 Derived channels

New channels can be computed from the original bands of the orthoimage (Trias-Sanz, 2006). They can be texture channels, indexes or simple functions of the original channels. The classification result can be improved using a combination of these derived channels. For instance, $NDVI$ (Normalized Difference Vegetation Index) channel is very usefull to discriminate vegetation. That's why several combinations of derived channels have been tested.

4 METHOD

The orthoimage is first segmented into homogeneous regions. These regions are then classified through a *MAP* classification algorithm taking into account external knowledge as prior probabilities.

4.1 Segmentation

First of all, the image must be segmented into homogeneous landcover regions. This is achieved thanks to the multi-scale segmentation method described in (Guigues et al., 2006) and (Guigues, 2004). This tool allows to compute a pyramid of segmentations of the image. Each level of this pyramid corresponds to an alternative between detail and generalization. This pyramid is then cut at a level empirically chosen to obtain a suitable image partition. The choice of this level is a compromise between detail and the size of regions since on one hand, in an over segmentation, some regions will be too small to have meaning and will be at risk to be misclassified whereas on the other hand, in a too coarse segmentation, wide regions will contain different landcover items. In the present case, the goal is to obtain a quite generalized result and the main difficulty was to find a level allowing to discriminate distinct themes (such as glaciers and rocks) in shadow without any over-segmentation of lit zones. More particularly, lit forest areas tend to be easily over segmented since they contain many small shadows due to differences of height between trees. Nevertheless, the use of downsampled images decreases this over-segmentation problem. Moreover, this reduces the computing time.

4.2 Classification

The segmentation's regions are then classified by the classification tool presented in (Trias-Sanz, 2006) and (Trias-Sanz and Boldo, 2005). This tool works in two steps :

1. Model estimation from training data captured by an operator
   First, for each class, the best parameters of several statistical distributions (such as gaussian, laplacian laws but also histograms (raw or obtained by kernel density estimation)...) are computed to fit to the radiometric n-dimensional histogram of the class (with n number of channels used for the classification). Then the best model is selected thanks to a Bayes Information Criterion which allows to choose an alternative between fit to data and model complexity.
2. Classification : The image can then be classified knowing the probability model of the radiometry of the different classes. Several per pixel and per region classification methods are proposed in (Trias-Sanz, 2006).

In the present case, a *MAP* per region classification algorithm is used. Such a method allows to take easily into account external information (from relief, from CLC2000 and concerning the shadow probability in the present case) as prior probability. With this classification method, the label $c_o(R)$ given to a region R is its most probable class according to the radiometric model previously estimated and to prior probabilities. Hence, $c_o(R)$ is the class c that maximizes the following function :
$\prod_{i\ extern\ information\ source} (P_i(c(R)=c))^{a_i} \cdot$
$\left(\prod_{pixel\ s\in R} P_{radiometricmodel}(I(s)|c(s)=c)\right)^{\frac{1}{Card\ R}}$
with $I(s)$ standing for the radiometry vector of pixel s, $c(z)$ meaning region or pixel "z's class" and $P(c(z)=c)$ standing for the probability for pixel or region z to belong to class c. the a_i terms stands for weight parameters balancing the different prior probability sources.

5 TESTS AND RESULTS

Tests were carried out on two zones : the first one is located near Saint-Christophe-en-Oisans (in the Alps) where only an old 3-bands argentic scanned orthoimage with many radiometric problems was available. All the landcover themes of the classification are present there. This was already the test zone of the previous study (Le Men et al., 2002), so it was interesting to compare the new results to the ones previously obtained.
The second test zone has been chosen in the Pyrenees, in the neighborhood of the Pic du Midi d'Ossau since 4-bands orthoimages (captured by a digital camera) are available there. No real glacier (just small remaining snow regions) is present in this zone, but all the other classification items are present there.

5.1 Tests on several parameters

5.1.1 Channels combination Many channels combinations have been tested. Those tests have shown that several channels combinations give quite good and almost equivalent results. Value, hue and NDVI (when infrared band is available) channels is one of the channels combinations giving the best results. The three Karhunen-Loève color space channels give good results too (Wang et al., 2003). Value, hue and a log-opponent chromaticity channel (Berens and Finlayson, 2000) has also allowed to obtain good results on the Alps'test zone.
Those tests have also shown that the use of texture channels does not improve the classification and tends to bring too generalized results and misclassifications between distinct themes having similar texture.

5.1.2 Prior information about shadows The tests have shown that this information is useful to prevent misclassifications such as the ones between lakes and rocks in shadow (as on figure 8). It also helps to discriminate glaciers in shadows from other themes and prevents the algorithm to classify every dark regions of the image as “rocks in shadow”.

Figure 8: Results without (on the left) and with (on the right) prior information about shadows (rocks in red, screes in pink, pasture in yellow, water in blue)

5.1.3 Prior information from the relief and CLC2000 The results obtained without introducing prior information in the classification process are very noisy and bad with many confusions (such as dark pasture and forest, or glaciers and light screes...). The introduction of prior probabilities considerably improves the classification results.
The balance between the different information sources has been tested too. Good results (see table 1) have been obtained with the following weights : 1 for image information, 0.75 for relief information and 0.25 for CLC2000 information (and 1 for shadow probability). A too important strength given to CLC2000 knowledge leads of course to too generalized results.
Concerning CLC2000, several prior probability models have been tested too and a convenient model for the two test zones has been found.

5.2 Final results and evaluation

The results were visually (on the whole image) and numerically (on smaller test zones in the image) evaluated. Numeric evaluation consists in computing confusion matrices by comparing test data captured by an operator to the classification result. Nevertheless, it is difficult to evaluate the obtained results since even a human operator can find it hard to discriminate landcover themes in some parts of the test zones. Moreover, concerning the particular case of glaciers, they can be covered by screes and therefore classified as screes (by the algorithm and a human operator).

Concerning the first test zone, obtained results are equivalent to the ones obtained by the previous study with little improvements in particular parts of the zone. A score of almost 67% well classified pixels (after aggregation of shadow/non shadow classes) among the test data has been obtained (see table 1). For comparison, a score of less than 55% well classified pixels has been obtained without prior knowledge. These results could seem quite bad but the part of the image where those zones are located is very difficult to classify, even for a human operator (see figure 9). Besides, the obtained results are sufficient for the new base map purposes. On the second test zone, better results have been obtained since almost 90% of the pixels of the test data have been well classified (table 1, figures 10, and 11). In this case, it must be said that test data zones are scattered in the whole image and are not so many as in the Alps zone. These better scores are also explained by the fact that there are less problems with the radiometry of the digital orthoimage used here than with the radiometry of the scanned argentic orthophoto available on the Alps’ test zone. In addition, the *NDVI* channel is available here and is usefull to discriminate vegetation and non vegetation regions. Nevertheless, prior information has been necessary to prevent many misclassifications and to limit noise (for instance very small rocky zones in a wide pasture zone).

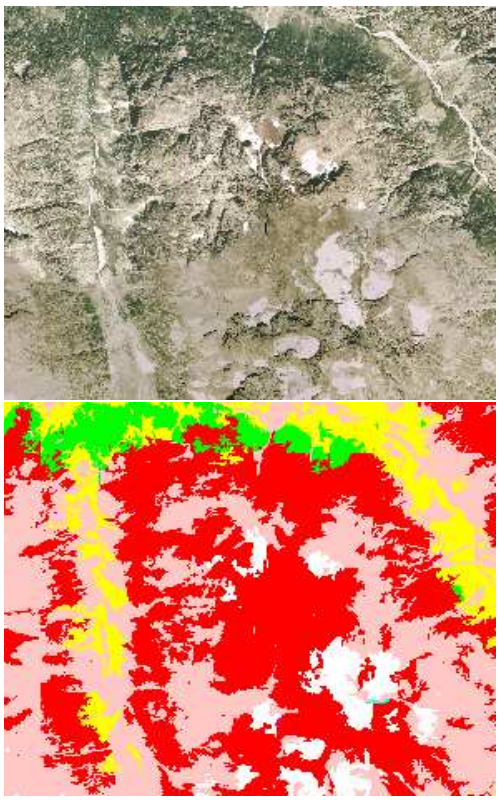


Figure 9: Example of classification in the Alps’test zone (rocks in red, screes in pink, pasture in yellow, forest in green, glaciers in white)

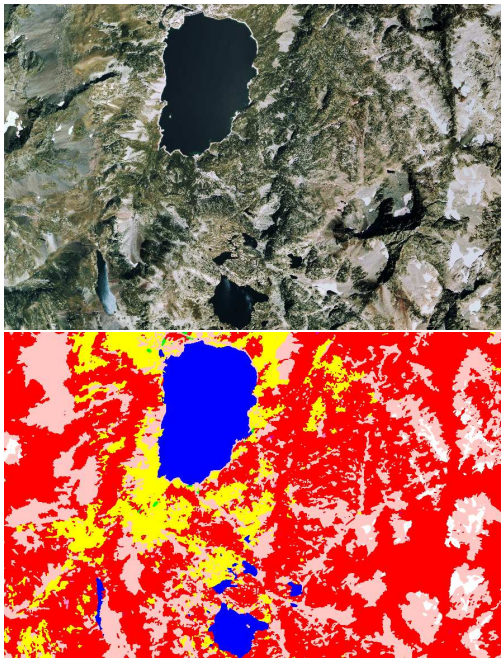


Figure 10: Example of classification in the Pyrenees’test zone (rocks in red, screes in pink, pasture in yellow, forest in green, glaciers in white, water in blue)

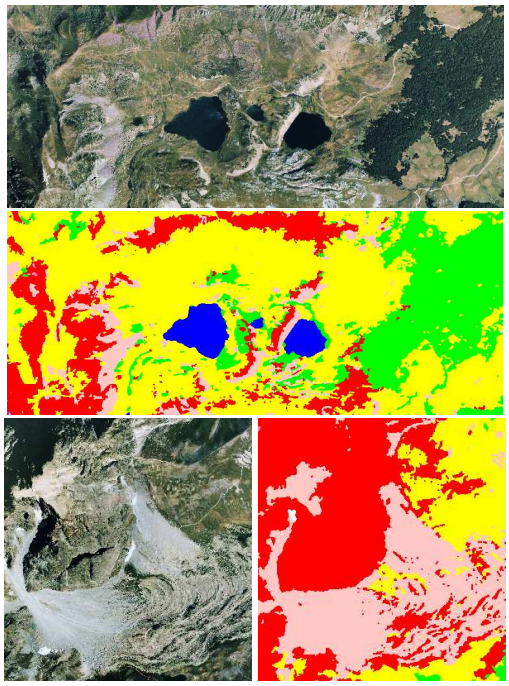


Figure 11: Example of classification in the Pyrenees'test zone (rocks in red, screes in pink, pasture in yellow, forest in green, glaciers in white, water in blue)

Table 1: Results obtained on the test zones (in %) with prior information: *us-ac* is the user accuracy and *pr-ac* is the producer accuracy.

	Pyrenees		*Alps*	
	with prior information			
	us-ac	*pr-ac*	*us-ac*	*pr-ac*
Water	100,0	76,4	/	/
Forest	89,0	95,8	81,9	65,2
Pasture	96,3	85,1	71,1	52,2
Rocks	71,0	87,2	76,4	69,9
Screes	88,0	83,1	54,7	73,3
Glaciers	98,3	72,2	58,6	69,5
Well classified pixels	87,4%		67,0%	
	without prior information			
Well classified pixels	75%		55%	

A visual evaluation has not revealed important errors on both test zones, and most of the regions of the classification have cartographic meaning (which is important in our mapping context).

5.3 Computing time

The tests have shown that the segmentation of orthoimages corresponding to a French base 1/25 000 map would take almost 6 hours with a single technical computer. The classification step would last almost 2 hours. It must be said that the initial 50cm resolution image has been previously downsampled to a 2m resolution image before the different computations. This allows to reduce the computing time and to obtain better segmentation results.
However, the computing time can yet be reduced, since before the start of the downsample - derived channels computation - segmentation - classification steps, the whole image is splitted into smaller images that are treated alone before being merged at the end of the computations. As a consequence, these small images can be treated in parallel on a cluster.

6 CONCLUSIONS AND FUTURE WORK

The obtained results are suitable for the new base 1/25 000 map's purposes since they are as precise and as generalized (after a local fusion of the smallest remaining regions without cartographic meaning).
The tests have shown that the introduction of external information from the relief or another database is necessary and to what extent this improves the result. They have also allowed to test how it is possible to use a more generalized database with a different legend, such as CLC2000 in the classification process. Tests have also proved that correcting the shadows is not necessary since dividing each class into "shadow/non shadow" subclasses is sufficient even though prior knowledge about the shadows'localization improves the results.
The method presented in this paper will now be tested on a new and more precise DTM in order to know to what extent it improves the results. It will also be tested on a new test zone (in the Alps) where all the landcover themes of the classification are present and where new 4-bands orthoimages (captured by a digital camera) are available.

REFERENCES

Berens, J. and Finlayson, G., 2000. Log-opponent chromaticity coding of colour space. In: Proc. of the International Conference on Pattern Recognition, Vol. 1, IAPR, Barcelona, Spain, pp. 206–211.

Bossard, M., Feranec, J. and Otahel, J., 2000. Corine land cover technical guide - addendum 2000. technical report no 40. Technical report, European Environment Agency.

CORINE Land Cover, last visited on the 31st of January 2007. http://www.ifen.fr/donIndic/Donnees/corine/presentation.htm.

Elhai, H., 1968. Biogéographie. Armand Colin.

Guigues, L., 2004. Modeles Multi-Echelles pour la Segmentation d'Images. PhD thesis, Ecole Doctorale Sciences et Ingenierie de l'Universite de Cergy-Pontoise.

Guigues, L., Cocquerez, J.-P. and Le Men, H., 2006. Scale sets image analysis. International Journal of Computer Vision 68(3), pp. 289–317.

Lacambre, A., 2001. Aléas et risques naturels zn milieu montagnard; apport et limites d'un systme d'information gographique. PhD thesis, Universite Paris 4, Paris, France.

Le Men, H., Trevisan, J. and Boldo, D., 2002. Automatic extraction of landcover themes on digital orthophotos in montainous area for mapping at 1/25k. In: Proc. of the ISPRS Commission II, Xi'an, China.

Trias-Sanz, R., 2006. Semi-automatic high-resolution rural land cover classification. PhD thesis, Universite Paris 5, Paris, France.

Trias-Sanz, R. and Boldo, D., 2005. A high-reliability, high-resolution method for land cover classification into forest and non-forest. In: H. Kalviainen, J. Parkkinen and A. Kaarna (eds), Proc. of the Scandinavian Conference on Image Analysis (SCIA), Lecture Notes in Computer Science, Vol. 3540, Springer, Joensuu, Finland, pp. 831–840.

Wang, Z., Ziou, D. and Armenakis, C., 2003. Combination of imagery - a study on various methods. In: Proc. of the IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Toulouse, France.

ASSESSING THE 3D STRUCTURE OF THE SINGLE CROWNS IN MIXED ALPINE FORESTS

A. Barilotti *, F. Sepic, E. Abramo, F. Crosilla

Dept. of Georesources and territory, University of Udine, Via del Cotonificio 114, 33100, Udine, Italy
andrea.barilotti@uniud.it, sepic@dimi.uniud.it, elabramo@tin.it, fabio.crosilla@uniud.it

Commission I/2, III/2, III/4, III/5, IV/3

KEY WORDS: LiDAR, Tree extraction, Morphological analysis, Region Growing, Crown delineation, Forest typology

ABSTRACT:

A method to automatically detect the tree crowns shape is presented in this paper. The study site is located in some mountainous parts of Friuli Venezia Giulia characterized by coniferous, mixed and broad-leaved forests with different population densities. The method, developed in an open source environment, is based on mathematical morphology operations that assess the cartographical position of the trees, as well as the height of the trees. Starting from single-extracted trees, a segmentation algorithm makes it possible to classify the laser point data as a subset of crown points. Then, the crowns are delineated by circular polygons centred on the geometric laser point barycentre. To enhance the quality of the calculated crown parameters (area, depth of insertion, volume), a statistical analysis of the height (z) frequency distribution was performed which allows the re-filtration of the low vegetation (border or under-canopy vegetation). The results have been validated using topographic total station data surveyed in situ, in 13 forestry sample plots with a total of about 550 reference trees. Considering the ecological diversity (complexity) of the chosen plots, the paper shows a high correlation between plot data and laser scanning extracted data, particularly in the coniferous areas, underlining the possibility of extending the fields of research to the study of the dominated vegetation under canopy.

1. INTRODUCTION

Monitoring of the forestry ecosystem is a current topic in the wooded resources sustainability debate. To characterize the vegetation from an ecological state and biomass content point of view, a detailed knowledge of the single tree population is needed. The assessment of such parameters is critical in terms of field operations and time needed. In this context, aerial laser scanning (LiDAR) is a promising survey technique for forestry inventories because of its capacity to directly assess the three dimensional structure of the forest due to the high point number of sampling per surface. Part of the research activities were carried out as a part of the INTERREG IIIA Phare/CBC Italia-Slovenia project entitled "Cadastral map updating and regional technical map integration for the GIS of the regional agencies by testing advanced and innovative survey techniques" at the University of Udine and in collaboration with the Geodetic Institute of Ljubljana. The research is focused on the use of Laser scanning data in the forestry field. In this context, the work was centred on the development of informative methodologies and algorithms to automatically assess the parameters characterizing the three dimensional structure of the single trees. The experiments have been carried out using original software developed in an open source environment (Beinat, Sepic, 2005) that allows the management of the laser point clouds. On the basis of this software, a specific tool of algorithms for forested areas has been implemented through which we can extract information about:

- the position, the number and the height of the single trees;
- the shape and the area of the single crowns (Barilotti, Sepic, 2006).

The data processing and the development of innovative algorithms for filtering, classification and modelling of laser scanning data are still being developed (Hyyppä et al., 2004). Further effort overall in the forestry field is needed because of the natural complexity of the single trees shape. Following the approach presented by some authors the determination of the population density and the crown shape over vast areas can be carried out by integrating laser scanning rasterized data with high resolution aerial images (Weinacker et al. 2004, Hyyppä et al., 2005). Other authors underline the advantages of using a direct analysis of the point clouds, avoiding interpolation on regularized grid of data (Tiede et al., 2005). As far as the study of three dimensional crown shape is concerned, the assessment of the insertion height is one of the most difficult parameters to assess. Due to this difficulty, some authors derive this geometric attribute from the LiDAR-extracted tree height, using empirical models of linear correlation (Pitkänen et al., 2004). However, this approach is only valid in a local setting and cannot be generalized because the crown shape varies depending on many factors: forest typology, population density, tree species, management type, soil typology etc. Moreover, the field survey of the crown parameters is not an easy procedure in terms of costs and time/operator needs and the results are not objective to determine. Starting from these considerations, the implementation of auto-adaptive methods to assess the crown three dimensional parameters is presented in this paper. Particular attention has been paid to verifying the quality of the results in different study plots in Alpine latitudes.

2. MATERIALS

The study areas are located in some mountain sectors of Friuli Venezia Giulia Region (N-E Italy) essentially characterized by coniferous forests (spruce, spruce-fir), broad-leaved forests (beech) and mixed forests. Within these areas some sub-zones

* Corresponding author

of interest have been located and geo-referenced using topographic total station and GPS. This has allowed the precise and accurate determination of the coordinates of 13 circular forestry plots (transects) with radius ranging between 12 and 25 meters. The forestry characteristics of the studied plots are reported in Table 1.

Plot ID	n° of trees /ha	Area (m^2)	Management type	Age	Composi-tion
FOA	663	450	stand	mature	mixed
FOB	531	450	stand	mature	mixed
MBA	619	450	stand	mature	mixed
MBB	1525	450	stand	juvenile	spruce
MBC	575	450	stand	juvenile	spruce
MBD	463	2000	stand	mature	spruce
PRB	840	450	stand	Juvenil e/adult	spruce
PRC	752	450	stand	Juvenil e/adult	spruce
SAA	336	2000	stand	mature	beech
TUA	538	700	conversion	juvenile	beech
TUB	862	450	conversion	juvenile	beech
TUC	553	450	conversion	juvenile	beech
VBA	1105	450	stand	juvenile	spruce

Table 1 – Summary of the geo-referenced forestry plot characteristics. Considering the different management type, age and composition of the 13 transects, 6 different forestry situations can be found.

These characteristics, describing the general ecological structure of the forests, give us an initial idea about the difficulty of characterizing the population with laser scanning and help to understand the expected morphometry of the single trees. The principal forest typologies studied are shown in Figure 1.

Figure 1 – Pictures of 4 different typologies of transect: juvenile stand spruce (MBC plot, upper-left); mature stand spruce (MBD plot, upper-right); juvenile converted beech (TUA plot, lower-left); mature stand beech (SAA plot, lower-right).

A field measuring campaign, performed within the 13 georeferenced transects, allowed to obtain detailed information on morphology and structure of each tree. Using a topographic total station we measured:

– The cartographic position of all trees (diameter at breast height more than 5 cm);
– The crown extension (4 sampling points for each one).

The crown base height was measured using portable instruments (length and angle). The diameter at breast height was also measured. The total data surveyed in situ using topographic instruments covers approximately 550 tree points and 2200 crown points. As far as the laser data is concerned, the principal characteristics of the dataset are reported in Table 2.

Plot_ID	Period of survey	N° of echoes	Local point density
FOA	november	F&L	2 pt/m2
FOB	november	F&L	2 pt/m2
MBA	june	Multiple	6 pt/m2
MBB	june	Multiple	7 pt/m2
MBC	june	Multiple	8 pt/m2
MBD	june	Multiple	10 pt/m2
PRB	november	F&L	1,5 pt/m2
PRC	november	F&L	1,5 pt/m2
SAA	october	F&L	4 pt/m2
TUA	may	F&L	2 pt/m2
TUB	may	F&L	2 pt/m2
TUC	may	F&L	2 pt/m2
VBA	june	Multiple	5 pt/m2

Table 2 – Summary of the laser data characteristics for each forestry transect.

As shown in the table, the datasets were surveyed in different periods. This must be taken into consideration, especially in the case of beech forests. In fact, as already shown (Barilotti et al 2006), the capacity of the laser beam to penetrate through the canopy depends on the presence (TUA, TUB, TUC plots) or non presence (SAA plot) of foliage cover. Some datasets were detected using a multiple pulse instrument (Optech ALTM 3100) that increases the capacity to sample the intermediate layers of the vegetation. In these cases we have plots with higher sampling points (5-10 pts/m^2) than those surveyed with a First & Last pulse laser scanner (Optech ALTM 3033; low density: 1.5 – 2 pts/m^2). The flight altitude was about 1000 m above ground and the laser beam divergence was 0.2 mrad according to the different survey campaigns.
The combination of the plot characteristics in terms of forest typology and laser metadata will be useful in order to understand the strengths and the limits of applying laser technology in forestry.

3. METHODS

A complete processing chain has been developed, starting with raw laser points as input data and ending with derived tree parameters for each single tree. The procedure is composed of a series of elaborations and transformations that can be schematically related to the following methodological aspects:

- Pre-processing of the raw laser data;
- Application of mathematical morphology algorithms, following a single tree approach, to extract the canopy apexes;
- Identification of the laser points belonging to the single crowns by means of a cluster analysis algorithm;
- Low vegetation sub-clustering using a local filtering method.

3.1 Pre-processing

The implemented step relating to the laser data pre-processing consists of an original algorithm that eliminates the points corresponding to the laser beam reflections under canopy from the dataset. The algorithm executes a first triangulation (Delaunay) of all points, then analyzes the height (z) difference between the vertexes of each triangle. Those vertexes whose height difference is greater than a threshold value (according to the minimal height of the forest) are eliminated. This allows the creation of a Digital Surface Model (DSM) without points under canopy and therefore introduces a higher degree of DSM adhesion to the external forest surface.

3.2 Tree extraction

The method proposed for the tree extraction is based on the morphologic analysis of the laser point distribution. To this aim the Top Hat algorithm, whose formulation is relative to the image elaboration theory (Serra, 1982), was implemented. Independently from the image typology, this mathematical function allows the extraction of the highest elements in the scale of the represented values (Andersen et al., 2001, Barilotti et al, 2005).
Extending the Top Hat concept directly to the pre-filtered point cloud, the method allows the detection of the set of points belonging to the top of the crown, avoiding the interpolation on raster images. The spatial position (x, y, z coordinates) of the apexes on the laser data can be obtained. It is assumed that the x,y coordinates of such apexes correspond to the cartographic position of the single trees. In some cases, because of the small height differences between nearest points belonging to the same crown, more than one apex can be marked for each tree. In order to diminish this kind of error, a checking algorithm that identifies and corrects the erroneously classified apexes (often localized into the crown edges) was introduced. The algorithm compares the height value of each extracted apex to the nearest laser points, using an opportune (user defined) search radius. If a point with a greater height value is found inside the searching window, it becomes the new apex.
Ground filtering is not an essential requirement in order to apply the morphological analysis, but it is however necessary to calculate the tree height. The filtering procedure has been done in this work using the software Terrascan™. The tree height is therefore calculated as the difference between the height value of the apex and the corresponding ground height. In accordance with the "National Forest Inventories" it is easy to exclude those uncertain apexes (trees) whose height is less than the given threshold (Barilotti, Turco, 2006).

3.3 Cluster analysis

In order to identify the single crowns a region growing algorithm was implemented. Starting from the apexes previously extracted, the algorithm classifies the vegetation points according to the criteria defined below:

- If the points located in the proximity of the starting apex are lower (height difference) than a fixed threshold, these are marked as belonging to the same cluster;
- When the same laser point is marked as belonging to different apexes (this is particularly true when the forest is characterized by close vegetation), the algorithm associates the point to the nearest apex;
- For each marked apex, the same procedure is iteratively applied.

An example of clustered data is given in Figure 2. The image highlights the auto-adaptive nature of the method. As can be noticed, the cluster shape (crowns) is not predefined and is closely related to the local morphology.

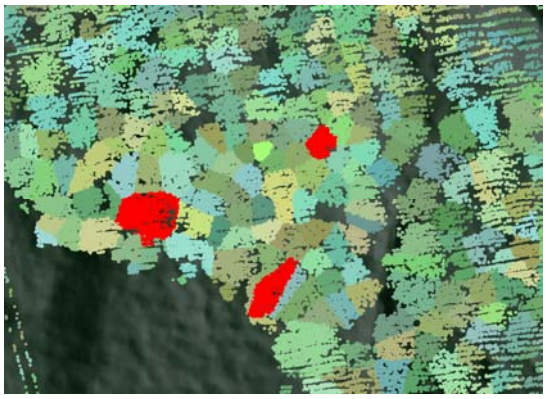


Figure 2 – Example of clustered laser data in a coniferous transect (MBD). The red coloured clusters emphasize the differences in terms of shape of the same species (spruce) in dependence of the ecological state.

3.4 Cluster re-filtering

Three different clusters are highlighted in red in Figure 2. One represents an example of a tree in a close forest while the others are located along the border. Even though the same species is present in the dataset shown, the resulting crown shape is very different in the three cases. The result is highly dependent on the efficiency of the tree extraction process.
Moreover, because of the presence of low vegetation in the dominated layer of the forest, a non optimal restitution of the crown geometry (area, base height) can be observed. This situation is outlined in the series of images in Figure 3 that show the three trees coloured in red of the previous figure isolated and visualized in a frontal view.
The height frequency distribution of the clustered points is reported (blue line) in the same sequence. The values are related to 1 meter spaced out classes along the x-axis. A specific analysis tool to automatically calculate the frequency distributions and the relative interpolating curves (red line) was implemented for each cluster.
As shown in the images (Fig. 3), the interpolated curves are very different in the three cases. The first kind of regression curve, in particular, indicates that the related tree is well clustered. On the contrary, the other curves (cases 2 and 3) indicate the presence of anomalies (higher point density in the lower classes of points) in the height frequency distribution. Such anomalies are evidently caused by the presence of dominated vegetation (understorey) and are not present when a mono-storey forest is surveyed. Starting from these considerations, it is therefore possible to characterize the interpolating curve through the study of the analytical function. Thus, we can use the height difference between the minimum

and the maximum points of the interpolated curve to find and define an automatic threshold of cluster re-filtering.

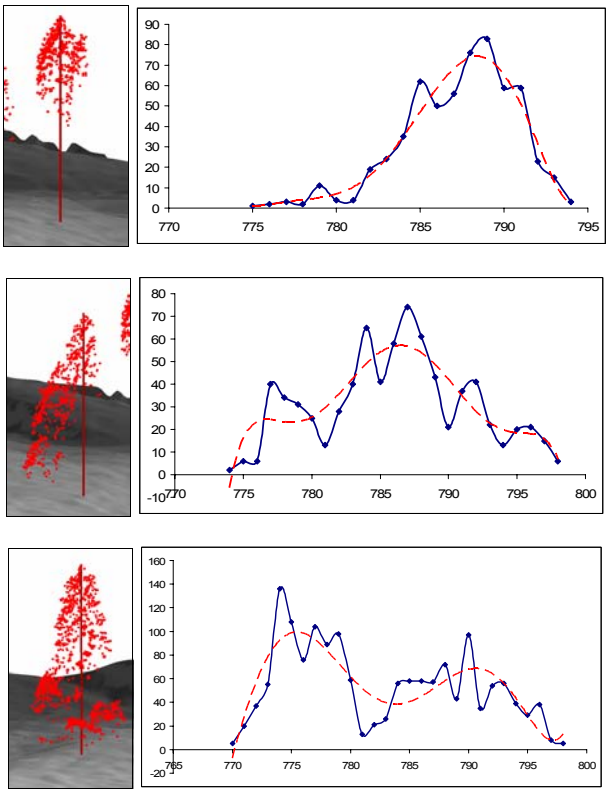


Figure 3 – Cases of clustered trees located in different positions into the forest. From the top to the bottom respectively: tree within close vegetation, two trees clustered together and tree with low vegetation. On the right the relative height frequency distribution of clusters is reported (blue line). For each frequency diagram, the 6° degree of the polynomial distribution is calculated (red line) and then it is used to re-filter the sub-clusters.

3.5 Crown delineation

The crowns are delineated using polygon circles whose parameters (centre and radius) are calculated analysing the planimetric coordinates of the points belonging to the clusters. The barycentre of the point distribution is assumed as being the centre of the crown. In most cases the centre of the circle does not correspond to the coordinates of the respective apex. This is particularly true in the case of beech forest. Each circle is drawn using a radius (r) equal to the following equation:

$$r = (X_{Max} - X_{min} + Y_{Max} - Y_{min}) /4$$

The equation allows the calculation of the average radius of the cluster distribution. The crown depth is calculated as the difference between the maximum height and the minimal height of the points belonging to the cluster. Moreover, the crown base height is calculated in terms of difference between the tree height and the crown depth (C_depth). The crown volume can be determined using the following equation:

$$C\_vol = CHM - [C\_area * (tree\_height - C\_depth)]$$

where C_area is the area of the delineated crown and CHM (Canopy Height Model) is derived by subtracting the height value of the DTM (Digital Terrain Model) at each pixel from the height value of the DSM.

4. RESULTS

The experimental results obtained by applying the methodological scheme shown above, were integrated into a G.I.S. (Geographic Information System) in order to create a database of forestry interest. The final result of the elaborations consists in two shapefiles for each input dataset (raw laser data) which contains information about trees and crowns summarized as follows:

- Cartographic position and height of the detected trees;
- Crown base height (alternatively the crown depth), crown area and volume.
- The field survey data is loaded into and managed by the same G.I.S. in order to compare and to check the quality of the LiDAR data extracted. The results are shown below.

4.1 Tree extraction

In Table 3 the results of the comparison between field trees and laser extracted trees are reported.

Field trees				LIDAR extracted trees			
Plot_ID	**Tot (ø>5)**	**Dominated**	**Dominants**	**Total Extracted**	**False positive**	**Total error**	**Correctly extracted (%)**
FOA	30	10	20	19	1	-2	90
FOB	24	8	16	15	3	-4	75
MBA	28	9	19	19	6	-6	68
MBB	69	37	32	19	0	-13	59
MBC	26	5	21	17	0	-4	81
MBD	91	11	80	77	4	-7	91
PRB	38	5	33	21	3	-15	55
PRC	34	5	29	27	6	-8	72
SAA	66	3	63	61	9	-11	83
TUA	38	18	20	42	17	5	75
TUB	39	12	27	33	3	3	89
TUC	22	1	21	19	2	-4	81
VBA	50	14	36	23	0	-13	64

Table 3 – Summary of field tree numbers and of the relative LiDAR extracted trees for each study plot.

In the table the trees whose diameter at breast height is significantly smaller to the surrounding ones are considered “dominated”. However, the individuals whose crown does not reach the top of the canopy were measured during the field campaign. Moreover, the apexes which are located 3 meters beyond the field surveyed trees are considered “false positives”. This is not generally a big error and could be further reduced applying more constraining parameters to the morphological analysis. We have to consider, however, that the method was applied in an automatic way (the same input parameters were used), independently from the typology of the forest and of the laser data differences. In any case, the forestry tool makes it possible to define and to optimize such parameters according to the previously mentioned variables. The percentage of correctly

extracted trees varies meaningfully depending on the structure of the different forestry plots examined. Juvenile forests, with a high population density and a high percentage of small diameters, highlight the difficulty of using laser technology to characterize the population well. In these cases, underestimation is evident in terms of "dominated" trees. On the contrary, the results seem to improve significantly when the forestry plot is mature and mono-storey structured (even-aged). In this case, the percentage of extracted trees reaches high values in coniferous forests (80-92%) as well in broad-leaved forests (83%), meaning that the most interesting part of the forest (from an above ground biomass content point of view) is extracted anyway. The tree height value is calculated using the maximum height of the laser points (apexes). As far as this parameter is concerned, the method does not introduce relevant underestimations, which are possible using different approaches based on rasterized data.

4.2 Crown delineation

Table 4 reports the difference between the crown base height values measured on site and those extracted from the laser data. As far as this parameter is concerned, the correlations correspond to the well-extracted trees. In these cases, we isolated and analyzed the base height values connected to the different species within each plot, as the table shows.

ID	Plot_ID	Species	Cnt	Min	Max	Ave	SD
1	FO_A_B	Beech	4	-12,77	-0,07	-4,10	5,86
2	FO_A_B	Spruce	41	-13,27	11,00	-0,25	5,12
3	MB_A_B	Fir	8	-2,29	6,50	2,05	2,96
4	MB_A_B	Beech	4	-20,19	10,86	-5,52	12,7
5	MB_A_B	Spruce	21	-19,80	8,59	-0,74	7,43
6	MB_C	Spruce	17	-4,10	0,41	-1,07	1,28
7	MB_D	Fir	14	-16,00	9,15	-2,53	6,49
8	MB_D	Spruce	55	-18,57	4,32	-2,84	5,46
9	PR_B_C	Spruce	56	-15,93	1,71	-3,70	3,58
10	SAA	Beech	65	-9,97	12,03	-0,47	4,51
11	TU_A_B_C	Beech	79	-19,37	15,50	-5,54	8,65
12	VBA	Spruce	31	-15,84	11,32	-0,62	6,93

Table 4 – Summary of the crown base height analysis. The results are reported in terms of different species surveyed within each transect.

Excluding from the analysis process those sub-plots where the number of trees per species (Cnt) is insufficient to perform statistical analysis (row 1, 3 and 4 in Table 4), the result show:

- The minimum and maximum differences of base height for each forestry transect reach high values, respectively as a result of the difficulty of the laser beam in penetrating the canopy (negative values) and of the presence of outlayers (positive values);
- The average values (ave) are always negative, suggesting the tendency of the laser to overestimate the base heights (it follows that the crown depth and the volume are underestimated);
- The worse valuation concern those plots characterized by juvenile broad-leaved forests (ave of TU_A_B_C = -5,54), while there is an improvement in mature broad-leaved transect (ave of SAA = -0,47). However, this last area was surveyed in the absence of leaf cover;
- The best results correspond to the coniferous transect and don't seem to be affected by the age of the forest.

The average and the standard deviation values for those plots which are statistically significant are shown in graph form in Figure 4. The graph highlights that the average values of crown base height have a small overestimation (negative values) in most transects while the standard deviation has a high range of values.

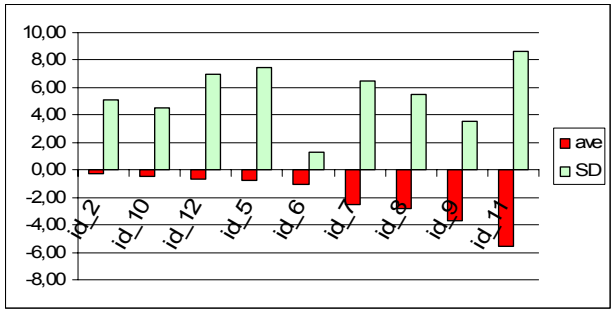


Figure 4 – Graphic visualization of the average and standard deviation of the values reported in Table 4 for those plots where the species number is significant.

The same analysis (average difference and standard deviation) was considered from a tree species point of view, without considering the plot characteristics. The values reported in Table 5 underline what can be expected in terms of crown estimation if a vast area of coniferous or broad-leaved forests is surveyed. While coniferous areas (spruce and fir) show little underestimation of the base depth (- 1,95 m < Ave < - 0,87 m), the base depth for beech trees is underestimate more (Ave < - 4 m).

ID	SPECIES	cnt	Min	Max	Ave	SD
1	Spruce	227	-19,80	11,32	-1,95	5,34
2	Beech	146	-20,19	10,86	-4,89	5,72
3	Fir	22	-16,00	9,15	-0,87	5,83
4	Pine	5	-13,93	11,51	2,62	9,85
5	Larix	4	-13,04	8,61	-1,05	10,32
6	Maple	3	0,83	8,69	4,01	4,14
7	Ash	2	0,79	2,04	1,42	0,88

Table 5 – Comparison of the crown base values between field surveyed and laser extracted data. The differences are summarized considering the different tree species.

Moreover, the application of the methods to multiple pulse surveyed data (cfr Table 2) doesn't seem to give better results, compared to the first & last data. The application of the re-clustering method previously mentioned would help to remove those return pulses due to the presence of low vegetation, like the example in Figure 5.

The quality of the crown area values depends substantially on the validity of tree extraction method. The qualitative comparison between the crown values measured on site and the clustered laser data correspond, especially in the case of the dominant vegetation layer. The reliability of the crown parameter estimates generally improves when laser point

density is increased (2 pts/m^2 is the inferior limit of useful density for the described morphologic approach) but, in any case, this is influenced by the quantity of extracted trees.

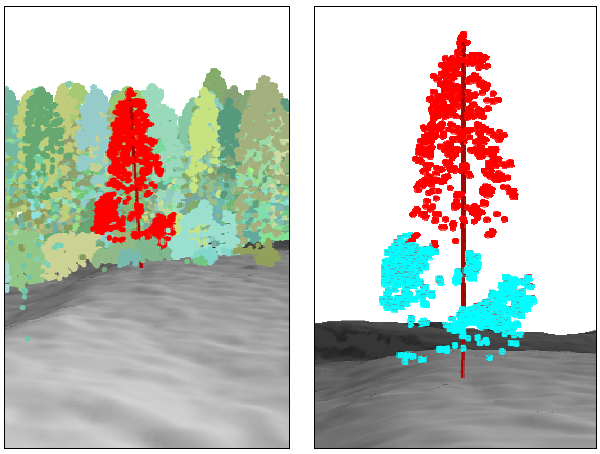


Figure 5 – Example of cluster re-filtering method to locally isolate the low vegetation (light-blue coloured points in the figure on the right) under dominant trees.

5. CONCLUSIONS

An innovative method of laser scanning data processing to automatically determine the crown parameters is proposed.
The Top Hat algorithm, implemented in Open Source environment, was used in order to determine the position of the tree apexes. Afterwards, using an original clustering algorithm, the single crowns were isolated and delineated. The results of the elaborations, opportunely integrated in GIS environment to create a database for the forestry sector, provide detailed information on the three-dimensional structure of the trees.
A field survey campaign in some mountainous geo-referenced plots highlighted the optimal performances of the method as far as the positioning and counting of the dominant trees (the main source of forestry biomass), in both coniferous and broad-leaved forests is concerned. Otherwise, a further work has to be done to improve the detection of the dominated vegetation.
As far as the correctly extracted trees are concerned, considering the difficulty of detecting the three dimensional tree parameters on site, the implemented method for crown delineation (particularly referred to the crown base height estimation) showed better results in the case of coniferous trees than in the broad leaved trees. As a matter of fact, in the latter case, the percentage of the laser beam penetration through the canopy is little because of the presence of very close vegetation.
Finally, a new method of cluster analysis, useful in filtering and isolating the dominated vegetation under canopy, was implemented. This last topic has to be verified in detail in the future, considering laser scanning data surveyed within multi-storey forestry plots.

REFERENCES

Andersen, H.E., Reutebuch, S.E., Schreuder, G.F., 2001. Automated Individual Tree Measurement through Morphological Analysis of a LIDAR-based Canopy Surface Model. *Proceedings of the first International Precision Forestry Cooperative Symposium*, Seattle, Washington.

Barilotti, A., Turco, S., Ciampalini, R., 2005. Misurazione automatica di singoli alberi attraverso analisi morfologiche su dati laser scanning. *Atti della 9° Conferenza nazionale ASITA*, Catania (Italy), 15-18 novembre 2005.

Barilotti, A., Turco, S., Alberti, G., 2006. LAI determination in forestry ecosystem by LiDAR data analysis. *Proceedings International Workshop 3D Remote Sensing in Forestry*, pp. 248 - 252, Wien, 14-15 Feb. 2006.

Barilotti, A., Turco, S., 2006. A 3-D GIS for the sustainable management of forest resources. *Proc. of the 4th Meeting of IUFRO Working Party 8.01.03*, Pattern and Processes in Forest Landscapes - Consequences of human management, pp. 349 - 354, Locorotondo (Italy), 26-29 Sept. 2006.

Barilotti, A., Sepic, F., 2006. Delineazione automatica delle chiome in diverse tipologie forestali attraverso analisi di dati LiDAR. *Atti della 10° Conferenza Nazionale ASITA*, Bolzano (Italy), 14-17 Nov. 2006

Beinat, A., Sepic, F., 2005. Un programma per l'elaborazione di dati Lidar in ambiente Linux. *50° Convegno Nazionale della Società Italiana di Fotogrammetria e Topografia*, Mondello, (Italy), 29-30 Jun. 2005.

Hyyppä, J., Hyyppä, H., Litkey, P., Yu, X., Hagrré, H., Rönnholm, P., Pyysalo, U., Pitkänen, J., and Maltamo, M., 2004. Algorithms and methods of airborne laser scanning for forest measurements. *International Archives of Photogrammetry, remote Sensing and Spatial Information Sciences,* Vol. XXXVI - 8/W2.

Hyyppä, J., Mielonen, T., Hyyppä, H., Maltamo, M., Yu, X., Honkavaara, E., Kaartinen, H., 2005. Using individual tree crown approach for forest volume extraction with aerial images and laser point clouds. *ISPRS WG IIIA, V/3 Workshop "Laser scanning 2005"*, Enschede, Sept. 12-14.

Pitkänen, J., Maltamo, M., Hyyppä, J., 2004. Adaptive methods for individual tree detection on airborne laser based canopy height model. *International Archives of Photogrammetry, remote Sensing and Spatial Information Sciences,* Vol. XXXVI - 8/W2.

Serra, J., 1982. Image analysis and mathematical morphology 2. *Theretical advances,* Academic press, London.

Tiede, D., Hochleitner, G., Blaschke, T., 2005. A full GIS-based workflow for tree identification and tree crown delineation using laser scanning. *IAPRS, Vol XXXVI, Part 3/W24,* Vienna, 29-30 Aug. 2005.

Weinacker, H., Kock, B., Heyder, U., Weinacker, R., 2004. Development of filtering, segmentation and modelling modules for LiDAR and multispectral data as a fundament of an automatic forest inventory system. *International Archives of Photogrammetry, remote Sensing and Spatial Information Sciences,* Vol. XXXVI - 8/W2, 2004.

ACKNOWLEDGEMENTS

This work was carried out as a part of the research activities supported by the extension of the INTERREG IIIA Italy-Slovenia 2003-2006 project "Cadastral map updating and regional technical map integration for the Geographical Information Systems of the regional agencies by testing advanced and innovative survey techniques".

A SUPERVISED APPROACH FOR OBJECT EXTRACTION FROM TERRESTRIAL LASER POINT CLOUDS DEMONSTRATED ON TREES

Shahar Barnea [a], Sagi Filin [a], Victor Alchanatis [b]

[a] Dept. of Transportation and Geo-Information, Civil and Environmental Engineering Faculty, Technion – Israel Institute of Technology, Haifa, 32000, Israel - (barneas, filin)@technion.ac.il
[b] Institute of Agricultural Engineering, The Volcani Center, Bet Dagan, 50250, Israel - victor@volcani.agri.gov.il

Commission III, WG III/4

KEY WORDS: Object Recognition, Feature Extraction, Terrestrial Laser Scanner, Point Cloud, Algorithms

ABSTRACT:

Terrestrial laser scanning is becoming a standard for 3D modeling of complex scenes. Results of the scan contain detailed geometric information about the scene; however, the lack of semantic details is still a gap in making this data useable for mapping. In this paper we propose a framework for object recognition in laser scans. The 3D point cloud, which is the natural representation of scanners outcome, is a complex data structure to process, as it does not have an inherent neighborhood structure. We propose a polar representation which facilitates low-level image processing tasks, e.g. segmentation and texture modeling. Using attributes of each segment a feature space analysis is used to classify segments into objects. This process is followed by a fine-tuning stage based on graph-cut algorithm, which takes into consideration the 3D nature of the data. The proposed algorithm is demonstrated on tree extraction and tested on 18 urban scans containing complex objects in addition to trees. The experiments show the feasibility of the proposed framework.

1. INTRODUCTION

We address in this paper the problem of object extraction from 3D terrestrial laser point clouds. Such extraction becomes relevant with the growing use of terrestrial laser scanners for mapping purposes and for the reconstruction of objects in 3D space. Object extraction from terrestrial laser scanners has indeed been a research topic in recent years, ranging from reverse engineering problems, to building reconstruction, and forestry applications. In most cases a model driven approach is applied, where domain knowledge about the sought after object shape drives the reconstruction and recognition process. Rabanni (2006) models industrial installations by making use of predefined solid object model properties. Bienert et al. (2006) propose an ad-hoc approach for tree detection based on trimming the laser data at a certain height to separate the canopy from the ground and searching for stem patches. Such approaches cannot be generalized to other objects, and usually assume well defined shape of the sought after objects.

Alternative approaches, which can still be categorized as model driven, involve generating a database consisting of diverse instantiations of 3D objects. Upon the arrival of a new unseen data, they search for a good matching score between regions in the new data and the database objects. The matching score is usually calculated via key-features and spatial descriptors. Such models are reported in (Huber and Hebert, 2003; Huber et al., 2004) that show good results while using the spin image based descriptors, Frome et al. (2004) that introduce 3D shape and harmonic shape contexts descriptors for the recognition, and Mian et al. (2006) that present a matching score which is based on robust multidimensional table representation of objects. These methods require the generation of a massive object instantiations databases and are relatively specific to the modeled objects. As such they can hardly be considered applicable for natural objects and data arriving from terrestrial scans. Another approach, which is model driven as well, is based on the extraction of primitives (points, sticks, patches) and modeling inter-relation among them as a means to recover the object class. Pechuk et al. (2005) propose the extraction of primitives followed by mapping the links among them as cues for the recognition part. This is demonstrated on scenes containing a small number of well defined objects with relatively small number of primitives (e.g., chair, table).

Differing from model driven approaches we examine in this paper the possibility to extract objects form highly detailed geometric information using a small number of training data and with limited domain knowledge. We demonstrate this approach on tree detection primarily because of the shape complexity of trees. The approach we propose is based on 3D geometric variability measures and on learning shape characteristics. The proposed method begins with segmentation of the scans into regions which are then being classified into "object" and "not-object" segments. This classification generates a proposal of candidate objects that are then being refined. As we show, the choice of descriptive features makes the classification part, which is the core of the proposed model, successful even when based on a relatively small training.

2. METHODOLOGY

2.1 Data Representation

When dealing with range data, most approaches are applied to the point cloud in 3D space aiming to recover the 3D relationship between scans. The hard task is to calculate the descriptive information in the irregularly distributed laser point cloud. Nonetheless, as the angular spacing is fixed (defined by system specifications), regularity can be established when the data is transformed into a polar representation (Equation 1)

$$(x, y, z)^T = (\rho \cos\theta \cos\varphi, \rho \cos\theta \sin\varphi, \rho \sin\theta)^T \qquad (1)$$

with x, y and z the Euclidian coordinates of a point, θ and φ are the latitudinal and longitudinal coordinates of the firing direction respectively, and ρ is the measured range. When transformed, the scan will form a panoramic range image in which ranges are "intensity" measures. Figure 1 shows range data in the form of an image where the x axis represents the φ value, $\varphi \in (0, 2\pi]$, and the y axis represents the θ value, $\theta \in (-\pi/4, \pi/4]$. The range image offers a compact, lossless, representation, but more importantly, makes data manipulations (e.g., derivative computation and convolution-like operations) simpler and easier to perform.

2.2 Segmentation

The transformation of the data panoramic range image allows the segmentation of the data using common image segmentation procedures. Recent works (e.g., Russell et al., 2006) have demonstrated how the application of segmentation processes for recognition tasks yields promising results both related in relation to object class recognition and to correct segmentation of the searched objects. Before segmenting the range images comes a data-cleaning phase that concerns filling void regions and the removal of isolated range measurements. Void regions are mainly the result of no-return areas in the scene (e.g., the skies) or object parts from which there is no reflectance. Isolated ranges appear detached from the ground and will relate to noise, leaves, or other small objects. No return regions are filled with a background value (maximal range), and for "no-reflectance" regions, ranges are assigned by neighboring objects. In Figure 1 the "no return" and the "no-reflectance" pixels marked with red.

For segmentation we use the Mean-Shift segmentation (Comaniciu and Meer, 2002), an adaptation of the mean-shift clustering algorithm that has proven successful for clustering non-parametric and complex feature space. The mean shift segmentation performs well in identifying homogeneous regions in the image. As can be seen in Figure 2, because of surface continuity and the general smoothness that characterize range data a tendency to join bigger regions into a single surface may exist. The algorithm can be controlled by two dominant parameters, the kernel size and permissible variability (range) within the segment. Tuning the variability to a small magnitude was useful in extracting "tree" segments (which are vertically dominant objects) as independent segments in the data. We note that even though under-segmented regions can be seen in other parts of the scan, this has little relevance to us.

2.3 Feature Space

The current part concerns isolating the tree related segments from the rest via classification. To perform the segment classification, a set of descriptive features for each of the segments should be computed. To keep the framework as general as possible we limit our search to low-level features. The sought after features should describe both the internal textural characteristics of the segment and characteristics of its silhouette shape. To keep the description simple, we seek a small set of descriptive features for characterizing the object. Limiting the set of features is useful for avoiding dimensionality related problems as well as overfitting concerns. The features we choose, consist of i) the sum the first-order derivatives, ii) absolute sum of the first-order derivatives, iii) the cornerness of the segment. These features (denoted *f1*, *f2* and *f3*) are computed per segment (L_i) as follows

$$\begin{aligned} f_1(L_i) &= \sum \left(d_\varphi(L_i) + d_\theta(L_i) \right) \\ f_2(L_i) &= \sum \left(|d_\varphi(L_i)| + |d_\theta(L_i)| \right) \\ f_3(L_i) &= \sum cornerness(L_i) \end{aligned} \quad (2)$$

with d_φ and d_θ the first-order derivatives of the polar image in the directions of its two axes. Since all three features involve summation and therefore are area dependent, they are normalized with respect to the segment area.

Analyzing the chosen features, the following observations can be seen. The first two features measure texture characteristics within the segmented area. Since trees have high range variability in all directions, the first feature should have low values (positive and the negative values cancel one another), while the second feature yields high values. The third feature, measures "cornerness" value for the area of the segment and its silhouette. For cornerness measure we use a corner operator we term min-max. The min-max operator considers points as corners when having "strong" gradients in all directions. In another formulation this can be stated as – a point is considered a corner even if the strength of the smallest gradient projection is big enough. With this formulation, corner detection can be seen as a min-max problem, by looking for the gradient projection in the minimal direction as the measure for the point "cornerness" (Cn). We leave the full mathematical development outside this text, due to space limitations, and present the formula for the cornerness measure in Equation (3)

$$Cn(\varphi_0, \theta_0, \alpha^*) = \sqrt{\sum W(\varphi - \varphi_0, \theta - \theta_0) \cdot \left(\left(\frac{d\rho}{d\varphi}\right)^2 \frac{\sqrt{T^2+1}+1}{2\sqrt{T^2+1}} + \left(\frac{d\rho}{d\theta}\right)^2 \frac{\sqrt{T^2+1}-1}{2\sqrt{T^2+1}} \pm \pm 2 \frac{d\rho}{d\varphi}\frac{d\rho}{d\theta}\frac{T}{2\sqrt{T^2+1}} \right)} \quad (3)$$

with

$$T = \frac{\sum W(\varphi - \varphi_0, \theta - \theta_0) \cdot 2\left(\frac{d\rho}{d\varphi} \cdot \frac{d\rho}{d\theta}\right)}{\sum W(\varphi - \varphi_0, \theta - \theta_0) \cdot \left(\left(\frac{d\rho}{d\varphi}\right)^2 - \left(\frac{d\rho}{d\theta}\right)^2 \right)}$$

$$\alpha^* = \frac{1}{2}\tan^{-1}(T)$$

and W, a Gaussian window. The weighting function can be applied by simple convolution over the image and derivatives by φ and θ can be easily computed numerically. Generally, because of their complex shape and depth variability, tree related segments will tend to have high cornerness values.

In computing gradients, the need to control the varying object-to-background distances arises. The potential mixture between object and background may arise from the 2D representation of the 3D data, and may lead to very steep gradients when the background is distant, or shallower ones for closer ones. To handle this we erode the border pixels and do not sum their derivative value, thereby keeping the texture measures to "within" the segment only. Additionally, we trim the magnitude of possible derivative by a threshold to eliminate background effects, so that backgrounds that are closer and farther from the object (which is irrelevant for the classification task) will have the same contribution to the derivatives related features.

The three features as calculated for the segments of the demonstration scan are presented in Figure 3. One can see that tree related segments have average values with f_1 (in this sub-figure the most negative values is black and the most positive is white), and relatively high values both in f_2 and in f_3 (bright).

2.4 Classification

The computation of the features for each segment in the training set allows the creation of the feature space. Such feature space is illustrated in Figure 4 via three projections and an isometric view. The four views show the separablity of the tree and non-tree classes as achieved through these features. Green dots are segments that were marked as "trees", red dots are "not-tree" segments. As can be seen in Figure 4 even tough the two classes are separated, the data do not follow the classical form of two, well separated, hyper-Gaussian distributions. We therefore apply a non-parametric method for classification, using the k-Nearest Neighbors (k-NN) algorithm. Our choice is motivated by its simplicity and efficiency, but we note that other methods may prove suitable as well. The k-NN model is based on evaluating cardinality of a sample (unseen data) compared to the neighborhood in the training data. Following the extraction of the k nearest neighbors for the data sample, a voting procedure among them is performed. If more than h class I segments are within this subset, the unseen segment is recorded belonging to class I if not, class II is recorded. The k-NN model is greatly affected by the distance measures between elements, particularly when the different axes measure quantities in different units and scales. Because of the different measures we use, great differences are expected in scale and distribution, motivating the need to normalize the data. For normalization we use the whitening (Mahalanobis) transformation (Duda et al., 2000) that transforms data into the same scale and variance in all dimensions. If $\mathbf{X}$ is a training set of size Nx3, with N the number of segments distributed with $\sim\{\mu, \mathbf{\Sigma}\}$; using the SVD, $\mathbf{\Sigma}$ can be factored into $\mathbf{\Sigma}=\mathbf{U}\mathbf{D}\mathbf{V}^T$, where $\mathbf{U}$ is orthonormal, $\mathbf{U}\mathbf{V}^T=\mathbf{I}$, and $\mathbf{D}$ a diagonal matrix. The transformed $\mathbf{X}$ is calculated by:

$$\mathbf{X}' = (\mathbf{D}^{-1/2}\mathbf{U}^T\mathbf{X}^T)^T \quad (4)$$

with $\mathbf{X}'$ the transformed set. Following the whitening transformation the data is distributed with zero mean and unit variance in all three dimensions of the feature space. Distance measures in this space become uniform in all directions.

Figure 1. Top: Polar representation of terrestrial laser scans; the horizontal and vertical axes of the image represent the values of φ, θ respectively and intensity values as distances ρ (bright=far). "No-return" and "no-reflectance" pixels are marked in red. Bottom: panoramic view of the scanned scene acquired by a camera mounted on the scanner.

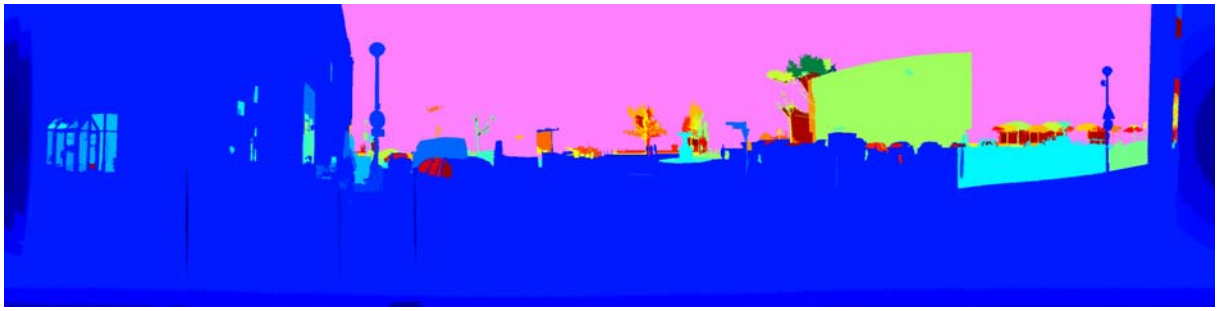


Figure 2. Results of the data segmentation using the mean-shift algorithm.

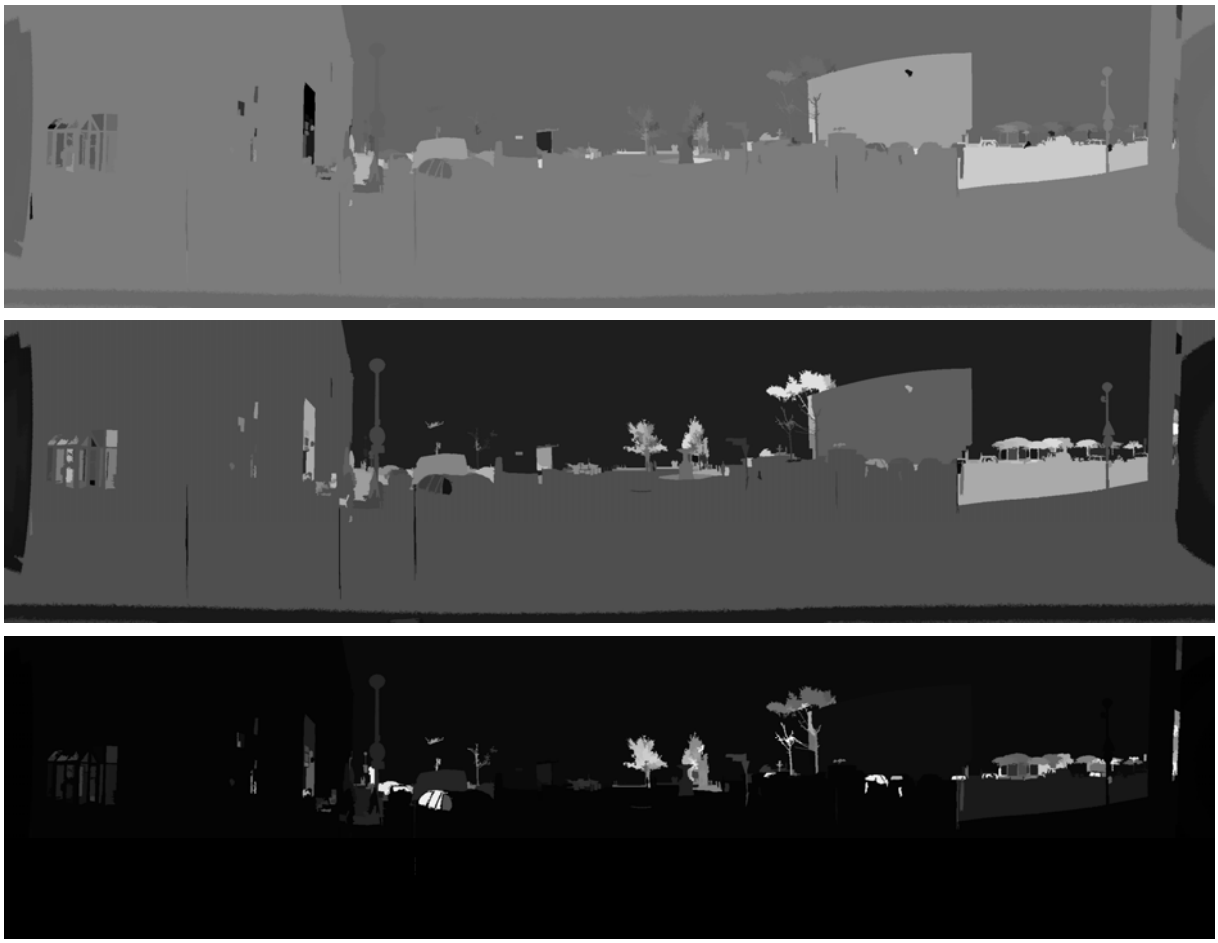


Figure 3. Segments weighted score for the three proposed features. Top: sum the first-order derivatives, middle: absolute sum of the first-order derivatives, bottom: the cornerness of the segment.

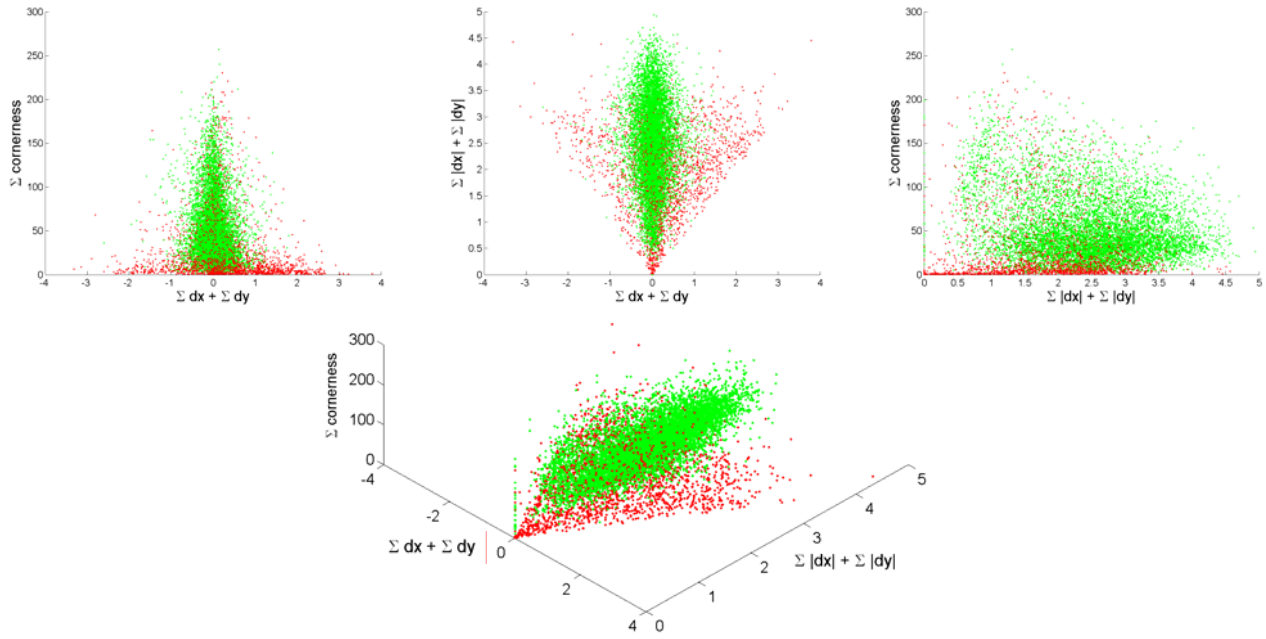


Figure 4. Four views of the feature space. The experiment contained 12351 segments that were manually classified. In green tree related segments and in red non-tree related segments.

The k-NN framework depends on number of neighbors checked (k), and on the cardinality parameter (h). Bigger k will make the model more general (when more samples are used to decide more information is weighted in) but less accurate (the extreme is where all samples are always used as neighbors). The choice of h affects the accuracy of the classification model. Setting h to a too small value, the model can become error prone, setting h too strictly, the number of false positives will decrease but on the expense of a large number of false negatives. An optimal value for h can be based on many considerations; our choice is based on finding a value that leads to the highest level of accuracy (ACC) as defined by

$$ACC = \frac{\text{True-Positive} + \text{True-Negative}}{\text{Positive} + \text{Negative}} \quad (5)$$

Such values can be derived by experimenting with different values for k and h. For each such trail a confusion matrix, C, is recorded

$$C \equiv \begin{bmatrix} true\ positive & false\ negative \\ false\ positive & true\ negative \end{bmatrix} \quad (6)$$

and the one with the highest accuracy value (Eq. 5) determines both the h and k parameters.

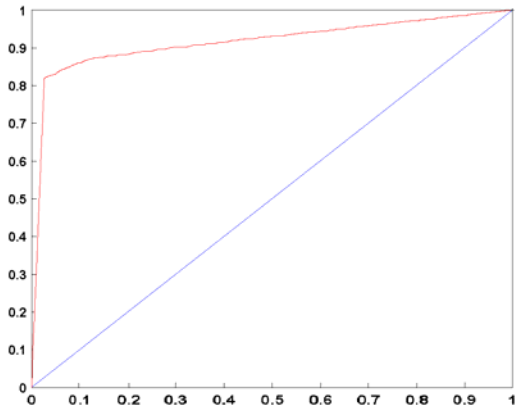


Figure 5. The ROC curve of the K-NN classifier.

2.5 Fine Tuning

So far, regions that have been identified via segmentation in 2D space have been classified as either trees or non-trees. Some of these segments are in fact sub-segments of the same tree (different part of the canopy or the stem), some segments may be a mixture of tree the background, and some segments may hold tree characteristics but are in fact non-tree objects. The fine-tuning phase aims linking segments that are part of the same tree, reducing to a minimum the number of false alarm detections, and separating mixture segments into object and background. Generally, this can be described as a split and merge problem among segments. We approach it differently by weighting the inter-relation between the individual points, so that neighboring points (by 3D proximity measures) will indicate potentially tight relations and therefore stronger utility in their link. The refinement phase revolves around an energy function of the form:

$$E = E_{data}(labeling) + E_{smooth}(labeling) \quad (7)$$

with E the total energy, E_{data} the energy related to the "wish" of laser point to maintain its original classification, and E_{smooth} the "wish" of highly connected points to have the same label. Labeling here refers to the binary value of the classified point in the point cloud and not to the outcome of the classification process. This energy function can be modeled by a graph, where each point in the cloud, i, is a vertex (V_i), and additionally, two more vertices, a source (s) and the sink (t) are added. The E_{data} elements are modeled through the weights assigned to edges linking each point and the source and each point and the sink. Each point (P_i) can have values of 0 or 1, depending on the output of the classification process. The weights on the edges are set according to

$$\begin{aligned} w(s, v_i) &= |p_i - \alpha| \\ w(v_i, t) &= 1 - |p_i - \alpha| \end{aligned} \quad (8)$$

with α the possible error in assigning a point. For representing the E_{smooth} part we search for the nearest neighbor point, j, for each point i in the cloud, and for each such pair (i,j) we build a link between the v_i and v_j whose weight is the inverse to the 3D Euclidian distance between the two points (the search for the nearest neighbor is performed via the Approximate Nearest Neighbor (ANN) method, (Arya et al., 1998)). Following the preparation of the graph, a graph-cut algorithm (Ford and Fulkerson, 1962) is applied to find the minimal cut (and the maximal flow) of the graph which also minimizes the energy function. The outcome of the graph cut refinement algorithm is separating between "tree" and "non-tree" points.

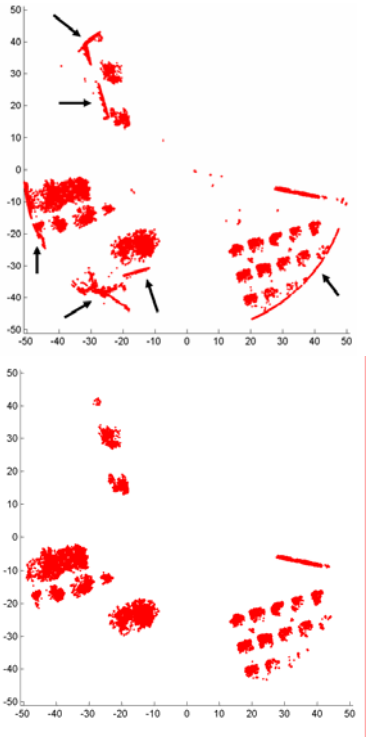


Figure 6. Fine tuning results, top: before, the arrows show areas that are not related to the object but lay on the background of it, bottom: the result of the execution of the algorithm. One can see how unwanted regions are filtered out.

3. RESULTS AND DISCUSSION

The algorithm was tested on 18 scans that were acquired in urban environment and, in addition to trees, contain cars, buildings, and other complex objects (see Figure 1). Each of the scans was segmented using the mean-shift segmentation; results

of a typical segmentation can be seen in Figure 2. For the experiment tree objects in those scans were manually marked and all related points were assigned as the ground-truth. In all, the eighteen scans generated 12351 segments (~700 segments per scan).

As was noted the k-NN classification model depends on the choice of k and h. Following the formation of the feature space those parameters were studied by letting k vary between 1-11 while for each k, potential values for h ranged from 1-k. The highest accuracy value that was recorded was found to be ACC=0.87 when using k=9 and h=5. The corresponding confusion matrix was

$$C = \begin{bmatrix} 0.7129 & 0.0347 \\ 0.0977 & 0.1547 \end{bmatrix}$$

All confusion matrices resulting from this experiment (for all k's and h's) were plotted on a ROC curve (Figure 5). The area under the ROC curve is 0.9192, which is an evidence for good classification.

Learning by example models usually require a large training set data. Because of the relatively limited number of the available scans we used leave-one-out cross validation experiments. For each scan the training feature space was recovered from the remaining 17 scans. In this experiment, the algorithm is tested in its holistic form, including the refinement phase. As a performance metric we use the percentage of correctly recognized tree points (true-positive), correctly recognized background points (true-negative). The performance of this procedure is

$$C = \begin{bmatrix} 0.053 & 0.029 \\ 0.005 & 0.913 \end{bmatrix}$$

leading to ACC=0.966. One can see that the results both the high level of success of the complete algorithm and the contribution of the refinement phase. This improvement is also demonstrated in Figure 6. Figure 7 offers the tree classification results in the range image. From Figure 6 one can see how the background objects that were wrongly classified as trees are now eliminated from the results. In addition to the filtering out of wrongly classified points, new points which are highly connected to the tree were added. The results also show how trees in different distances (resolution) and ones that are partially occluded were detected by the algorithm.

4. CONCLUDING REMARKS

The paper has demonstrated that detection of objects with high level of accuracy can be reached by learning object characteristics from a small set of features and a limited number of samples. The detection scheme has managed identifying trees both in different depths (scales) and ones that were partially occluded. The small number of false alarm detections indicates the appropriateness of the selected features for the recognition. Using additional features and slight adaptations, the proposed approach can be further extended to detect different objects like buildings, cars, and others as well.

5. ACKNWOLEDGEMENT

The authors would like to thank Dr. Claus Brenner for making the data used for our tests available.

6. REFERENCES

Arya, S., Mount D. M., Netanyahu N. S., Silverman R., Wu A., 1998. An optimal algorithm for approximate nearest neighbor searching. Journal of the ACM, 45, 891–923.

Bienert, A., Maas, H.-G., Scheller, S. (2006). Analysis of The Information Content of Terrestrial Laserscanner Point Clouds For The Automatic Determination Of Forest Inventory Parameters. Workshop on 3D Remote Sensing in Forestry, 14th-15th Feb 2006, Vienna

Comaniciu D., Meer. P., 2002. Mean shift: A robust approach toward feature space analysis. IEEE trans. PAMI, 24:603–19.

Duda, R. O. Hart P. E. and Stork D. G., 2000. Pattern Classification 2nd ed. Wiley, 2000.

Ford, L., Fulkerson, D., (1962). Flows in Networks Princeton Univ. Press.

Frome A., Huber, R. Kolluri, T. Buelow, and J. Malik (2004). Recognizing Objects in Range Data Using Regional Point Descriptors. in Proc of ECCV 2004, 3, 224-237

Huber, D. Kapuria, A. Donamukkala, R., Hebert, M. (2004) Parts-Based 3D Object Recognition. Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition, vol. 2, 82-89.

Huber D. and Hebert, M., (2003). Fully automatic registration of multiple 3D data sets. Image and Vision Computation 21(7):637-650.

Mian, A., Bennamoun, M., Owens, R., (2006). Three-Dimensional Model-Based Object Recognition and Segmentation in Cluttered Scenes. IEEE transactions on PAMI. 28(10), 1584-1601.

Pechuk M., Soldea O., Rivlin E., (2005) Function based classification from 3D Data via Generic and Symbolic Models. In AAAI, 950-955.

Rabanni T., 2006. Automatic reconstruction of Industrial Installations using point clouds and images. PhD thesis. NCG, publication on Geodesy 62.

Russell, C. B., Efros A., Sivic J., Freeman T. W., Zisserman A., (2006). Using Multiple Segmentations to Discover Objects and their Extent in Image Collections. in Proc. of CVPR 2006 2, 1605-1614.

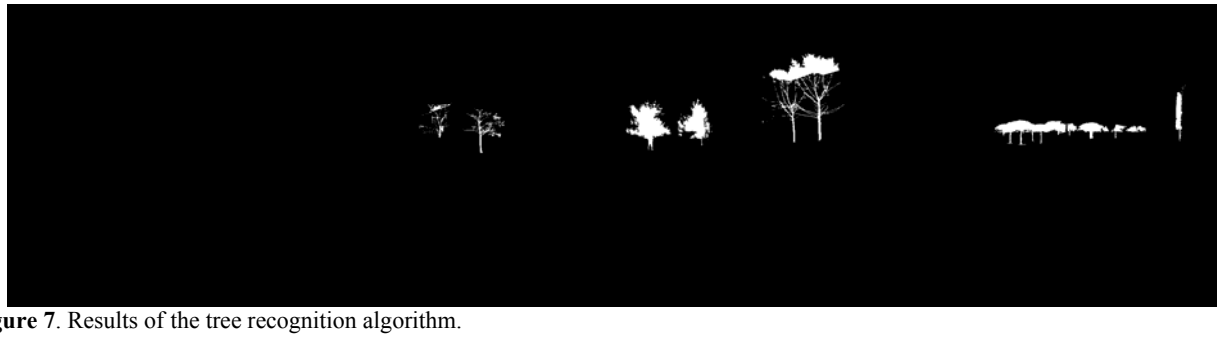


Figure 7. Results of the tree recognition algorithm.

AUTOMATIC ROAD EXTRACTION FROM REMOTE SENSING IMAGERY INCORPORATING PRIOR INFORMATION AND COLOUR SEGMENTATION

M. Ziems[a], M. Gerke[b], C. Heipke[a]

[a] Institut für Photogrammetrie und Geoinformation, Leibniz Universität Hannover, Nienburger Str. 1, 30167 Hannover, Germany - (ziems,heipke)@ipi.uni-hannover.de

[b] International Institute for Geo-Information Science and Earth Observation - ITC, Department of Earth Observation Science, Hengelosestraat 99, P.O. Box 6, 7500AA Enschede, Netherlands, gerke@itc.nl

KEY WORDS: GIS, Imagery, MSI, Road Extraction, Quality

ABSTRACT:
In this paper an approach to road extraction in open landscape regions from IKONOS multispectral imagery is presented which combines a line-based approach for road extraction with area-based colour segmentation. Existing road databases are used in two ways: firstly, to estimate scene dependent parameters of the line-based approach and secondly to exclude non road regions from the extraction, also exploiting the colour information. The images and reference data for the evaluation are identical to the information used in a recently conducted EuroSDR test on automatic road extraction. Therefore, the results can easily be compared with the published ones. It is shown that our new approach with reduced human interaction for the parameter optimisation obtains results similar to the best ones of the EuroSDR test.

1. INTRODUCTION

The road network is an important component of the infrastructure in every country. Therefore, up-to-date and accurate information on the road network is of vital importance. Today, short up-date-cycles and high quality digital road databases are requested. One means to satisfy these demands is to make use of digital remote sensing imagery for automated road network extraction, quality control and update.

Recently, a test on automatic road extraction algorithms has been carried out by EuroSDR (Mayer et al., 2006). The results show that automatic road extraction approaches for open landscape based on aerial images or high resolution satellite imagery are operational, although lacking quality and efficiency in some instances. Two issues were identified which contribute to enhanced results: the use of 1) colour information and 2) prior information for the extraction of roads. Although the latter aspect could not be verified by the test, because no prior information was offered, the authors of the EuroSDR test report made this general statement based on their experience and feedback from the participants. We can confirm this statement, because in own experiments the incorporation of existing road databases into the road extraction lead to better results compared to an approach where this data was not incorporated (Gerke et al., 2004).

In our current research presented in this paper we use pan-sharpened multispectral IKONOS imagery to delineate road centrelines in open and rural landscape areas. Prior information, derived from an existing road database, is used in two ways: firstly, to estimate scene dependent parameters of a line-based road extraction approach and secondly to exclude non-road regions from the extraction, also exploiting the available colour information.

The background of our research is given by the WiPKA-QS-Project, where the German authoritative topographic reference data set ATKIS-DLMBasis and the MGCP (Multinational Geospatial Coproduction Program) dataset being produced by the German Federal Armed Forces (Grelck and Müller-Grunau, 2005) are to be automatically verified using remote sensing imagery (Busch et al., 2004).

Results of our approach document that both aspects, namely the exploitation of available prior information and the use of colour for road extraction significantly enhance the overall performance, i.e. the correctness and completeness of the automatic road extraction.

2. RELATED WORK

In this section, a brief overview of existing road extraction approaches for open and rural landscape areas is given. The overview is not meant to be complete; it focuses on some aspects which are related to our present work and the approaches only represent a small part of the whole range of papers available. More complete overviews on road extraction approaches are presented e.g. in (Hinz and Baumgartner, 2003), (Hinz, 2004) and (Gerke, 2006). We first concentrate on the geometric road model which is used for the extraction, i.e.: line vs. area-based road extraction. The role of colour information is also discussed in this context. Secondly, the way other authors incorporate existing prior information is presented.

2.1 Line-based vs. area-based road extraction and use of colour information

In (Wiedemann, 2002, 2003) roads are modelled as linear objects in aerial or satellite imagery with a resolution of about 1 to 2m. The underlying line extractor is the one introduced in (Steger, 1998). The initially extracted lines are evaluated by fuzzy values according to attributes, such as length, straightness, constancy in width and in grey value. The final step is the grouping of the individual lines in order to derive topologically connected and geometrically optimal paths between seed points Seed points are end points of lines which reached a cost minimum in the evaluation score. The decision whether extracted and evaluated lines are grouped into one road object is taken based on a collinearity criterion, allowing for a maximum gap length and a maximum direction difference.

One main problem with the line-based approach is the tuning of parameters for line extraction. The parameters depend on the object and on context. For instance, a dark asphalt road in a desert environment can be delineated more reliably and accurately than a vegetated path between a cultivated environment, e.g. between two grassland fields. Moreover, the contrast conditions depend on the actual illumination conditions of the satellite scene. To compensate these variabilities of road extraction results, Hinz and Wiedemann (2004) described an extended approach that checks the final results by internal evaluation measures. This enables a prediction about the adaptation level of the used parameter set and may lead to alternative settings.

In (Zhang and Couloigner, 2006) an area-segmentation-based approach is developed. Firstly, the multispectral image is segmented using the unsupervised K-means clustering method. Road segments are filtered by evaluating the resulting segments based on shape descriptors. Finally, the road centrelines are retrieved from the grouped road segments using a skeletonisation method.

Also in (Mena and Malpica, 2005) colour information is used for the segmentation of the image. Mainly, the evidence delivered by three different statistical approaches is combined and leads to an enhanced segmentation of the imagery. The first source of evidence is given by a supervised classification, employing the Mahalonobis distance. The second source is obtained from the comparison of the colour distribution in the neighbourhood of individual pixels with the trained distributions. For this purpose, the Bhattacharyya distance is used. The last source of evidence is given by Haralick features derived from the co-occurrence matrix. The segments obtained after evidence fusion are then skeletonised to obtain the road centre axis.

In (Bacher and Mayer, 2005) and (Bacher, 2006), a combined approach is introduced. A pixel-based multispectral classification is used to generate a so-called roadclass-image. The training information for the supervised classification is obtained from a very strict initial road extraction according to Wiedemann (2002, 2003), and using additional parallel edge information (Baumgartner et al., 1999). The subsequent road extraction is then conducted according to Wiedemann (2002, 2003), but the roadclass-image is used to additionally evaluate the extracted lines.

One problem with the presented approaches based on area segmentation is that the estimation of the road centreline is only based on classified segments. Consequently, the accuracy of the resulting centreline is decreased (Mayer et al., 2006). In (Zhang und Couloigner, 2006) shape parameters are applied to filter road segments, i.e. to eliminate parking lots, buildings etc. from the segments. However, no image information is used at this stage, leading to misclassification of segments in some cases.
Bacher (2005, 2006) uses a line-based approach, but also exploits colour information for an enhanced classification of extracted lines. However, the sensitive parameter setting for line extraction is not further automated in this approach.

2.2 Incorporation of prior information

In (Doucette et al., 1999) the information from a coarse resolution road database is used to initialise road extraction based on a Neural Network approach, i.e. the database information is used to provide input samples. In (Bordes et al., 1997) and (Zhang and Baltsavias, 2002; Zhang 2004) road databases are used to specify the type of road and attributes such as the width. This information is employed to define hypotheses for the appearance of roads. Features extracted by means of image analysis are then used to evaluate these hypotheses.

One interesting result of these approaches is that although the representation of the database objects is very coarse, it is good enough for limiting the search space and making assumptions on the appearance of the objects due to the type given in the database.

The approach presented in (Mena and Malpica, 2005) also makes use of prior information as explained above. It is used to define the feature space for colour segmentation.

3. THE NEW APPROACH

From the brief overview of the related work, some lessons can be learned. If applicable, the line-model should be used. This is the case when the background objects are homogeneous. However, the problem of line extraction approaches is the tuning of parameter which is object and scene dependent. Furthermore, existing research shows that colour information is an adequate means to distinguish road objects and background.

This paper presents a new approach to road extraction in open landscape using IKONOS imagery. The approach has the following properties:

- It makes use of the line-model and extraction algorithm as proposed by Wiedemann (2002, 2003)
- GIS database objects are used as prior information
- Radiometric properties of the roads, in this case the contrast between road surface and background are trained for every individual scene using the prior information from the road database
- Colour information is used to segment background objects and exclude them from the extraction.

3.1 Training of radiometric parameters

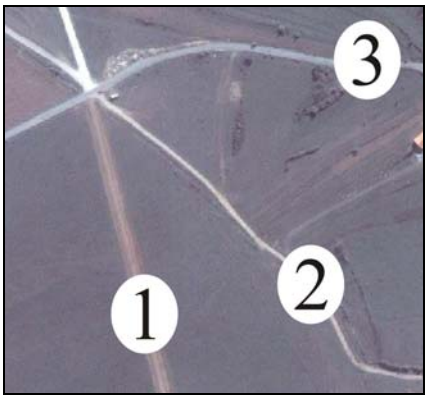


Figure 1: IKONOS sub-image with three different roads

Figure 1 shows an example of rural road objects with different surfaces. The surface appearance is correlated with the given road classes. Roads for traffic are mainly sealed and mostly consist of concrete or asphalt. In contrast, agriculturally paths often consist only of gravel or sand and are partly covered by vegetation. Additionally, the variety of possible local

backgrounds, i.e. cropland, grassland and wood, leads to a number of possible radiometric combinations for roads and their local background. Our assumption is that given a representative set of prior information on road objects in a particular (satellite) scene, the most critical parameters for line-based road extraction can be estimated automatically. The prior information is obtained from GIS road data. Appropriate information includes geometry, width, classification and the image co-registration of all given road objects.

The core of the parameter training algorithm is realized by a radiometric histogram analysis of road regions in high resolution one-channel imagery. In our application we use panchromatic imagery as well as the NDVI channel which was computed from the pan-sharpened MS channels. For each road object, which is registered in the GIS database, a specific image region is generated, based on the given road centre axis and the attribute *road width*. If no road width is available, a default value is used. The resulting region width is reduced to exclude most of the mixed pixels at the road border. Consequently, it is assumed that the majority of the region pixels belong to a road. Additionally, for each object a second and a third region (one to the left, the other to the right of the road region) are generated to refine radiometric information about the local background. Once again mixed pixels are excluded by a defined minimum distance from the analysed object. Figure 2 shows an example for a generated road region (blue) and the two regions which represent the roads local background (yellow).

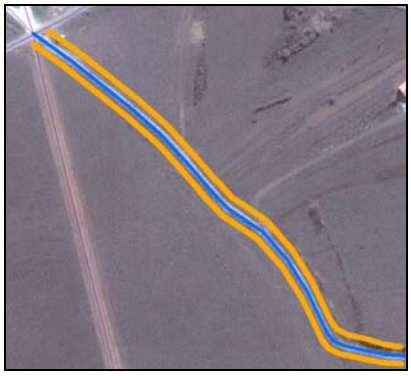


Figure 2: Buffer of road object (blue) and local background (yellow)

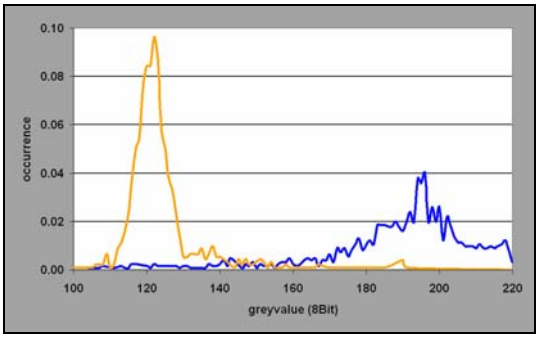


Figure 3: Histogram of a bright dirt road (blue) and its local background (yellow)

Figure 3 shows the histogram of the bright dirt road with homogeneous local background. In contrast, Figure 4 shows a homogeneous dark asphalt road with a typical heterogeneously local background. From the histograms this evident, that the parameters for the line extraction should be selected according to the road class. Thus, if information on road classes is available, the procedure described in the following is applied per road class.

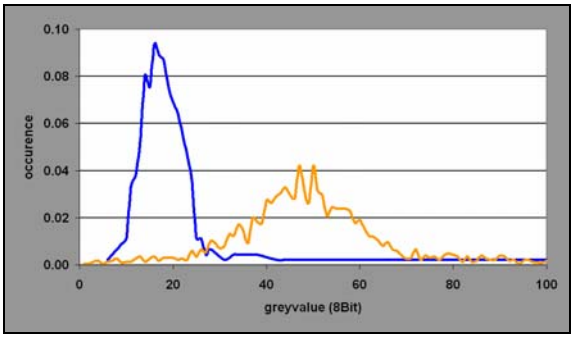


Figure 4: Histogram of a dark asphalt road (blue) and its local background (yellow)

Based on the calculated object specific histograms the following radiometric parameters are derived:

- *Homogeneity* along the line object assumed to be a road. The line extraction operator requires pixels with similar greyvalues along the road centreline to form a line object.
  We calculate the homogeneity as the standard deviation of greyvalues from the road histogram.

- *Contrast* between road and local background. The line extractor requires a minimum difference of pixel-greyvalues across the road to form a line object.
  Local background can consist of different objects. We look for the contrast between the road and the most similar background object, thus the smallest difference between the peak of the road pixels and a peak of the background pixels is required. The peak of the road pixels is computed as follows: first we only consider greyvalues which account for the top 80% occurrence in the histogram in order to eliminate possible disturbances. We use rank filtering to obtain this result. We then compute the median of the remaining road pixels as the desired peak value. In order to detect the background peak we apply the same rank filtering. Starting from the road peak we then find the closest local maximum. The difference between both peaks is the contrast.
  The contrast is used to decide weather the investigated road is brighter or darker than the background. The differentiation between dark and bright roads is used to classify the radiometric parameters into two categories. The line extractor separates between dark and bright line models. Accordingly, a dark–bright specific parameter derivation is useful.

- *Global threshold* is defined as the higher (dark roads) or the lower (bright roads) limit to generate a region of interest containing the roads.
  The threshold is extracted as the maximum value (dark roads) or the lowest value (bright roads) from the rank filtered road region histogram.

In terms of the line extraction algorithm, only plausible results are considered for further calculations. Therefore, the refined *Homogeneity* has to be smaller than the resulting *Contrast*.

Consequently, the three parameters are calculated per given GIS road object. The next task is to find the optimal values for a scene and – if available – a road class dependent road extraction. For this task, three sorted lists are created, one for every parameter where the individual values are listed. Geometric or attribute errors of the GIS database have to be considered in the parameter training algorithm. It is assumed that the parameters derived from wrong GIS road objects with plausible results only influences some of the top values of the sorted lists, therefore rank filtering is applied to derive the radiometric parameters. Other recent work like (Bacher 2005, 2006) uses this method with a similar aim.
The details of rank filtering depend on the overall GIS database quality. Empirical investigations showed that a rank of 10 % seems to be applicable. To guarantee the robustness of the system towards area wide incorrectness of GIS data a minimum number of training objects has to deliver plausible parameter values. Otherwise standard parameter settings are automatically used for the extraction algorithm.

3.2 Colour segmentation

The available IKONOS imagery contains four spectral bands: red, green, blue and infrared with a ground resolution of approximately 4m, but is pan-sharpened to a nominal resolution of 1m. The used line extraction approach of Wiedemann (2002, 2003) does not consider multispectral information directly. Therefore, advantageous channel combinations like NDVI or intensity are still used in the extraction step. A separated line extraction conducted in every available IKONOS channel is not applied. According to our experience the different colour bands provide no significant additional information. Nevertheless, this information is more valuable in an area-based approach. Therefore, a colour segmentation step is added, to enhance multispectral application. Thereby, a region of interest (ROI) is defined for the further road extraction step. The basic ambition is to determine the range of spectral signatures which together contain all possible spectral combinations of roads in a specific scene. The overall calculation is based on RGB colour space with 8 bit per channel. Although, the NIR channel would probably contribute valuable information, we make no use of this channel up to now. The usage of an alternative colour space, like HIS or HSV, offers to our experience no significant advantage for segmentation.

3.2.1 General approach

Similar to the parameter training approach, explained in the preceding section, roads from the given GIS database are used to define the training region. If available, the attribute *road width* is used to generate a region that contains all road objects. This training region is further classified using the attribute *road class*, if available, to enhance the spectral analysis. In contrast to the parameter training approach an object specific procedure is not applied. All pixels from the training region are transferred to RGB colour space. In this step, the number of occurrences of every spectral combination is registered. Based on these values, a histogram analysis is applied as described below, to build up different clusters in feature space. Next, all image pixels within ROIs are classified into road and non-road pixels.
Noise effects of the resulting image regions are reduced by erosion and the adjacent regions are connected. Moreover, simple shape descriptors are used to identify image regions which are not consistent with the road model, i.e. large crop fields. These large regions are sometimes spectrally similar to roads and are eliminated. Subsequently, based on the topological characteristics of the road network, all isolated region parts are also eliminated. The resulting ROI is extended along the borders to include the local background for the line-based road extraction in the image.

3.2.2 Histogram analysis

The definition of relevant clusters in feature space is difficult. Because of the limited road width, the pixels belonging to road objects are often heavily influenced by the spectral properties of the local background. Only for wider asphalt roads a clear differentiation from the background can be achieved from the training data. These roads are often characterised by a significant saturation in the blue band. Additionally, a lot of its pixels are influenced by low saturation objects like bright road markings or dark wheel tracks. In contrast, smaller dirt roads have no predictable spectral signature. Because of the described difficulties, the colour segmentation algorithm is currently reduced to a histogram analysis of the RGB-feature space including rank filtering.
Again, it is assumed that the GIS database used for the training is mainly correct. Consequently, the spectral signature of roads is supposed to occur frequently. A histogram analysis based on the RGB colour space is carried out in a similar manner just as described for the parameter training. The computed number of occurrences of every spectral combination gives evidence about the membership of scene specific road classes. Consequently, all occurring spectral combinations are sorted according to their frequency. Based on the sorted list, a heuristic threshold for rank filtering is used for the classification. This threshold depends mainly on the overall quality of the used GIS database. For the tested GIS database, an empirical determination has shown that the colour information from the top 80% of all analysed pixels represents the scene depended road surfaces.

4. RESULTS

In order to be able to compare our approach with the recently published results from the EuroSDR test on road extraction (Mayer et al. 2006), we use the same pan-sharpened IKONOS imagery and reference data for the evaluation. They depict some different scenes in the Kosovo. Our test is restricted to the rural hilly scenes of the EuroSDR test, shown in Figure 6a and 6b. The urban IKONOS scene is excluded from the following results because the overall approach is designed for open landscape.
GIS road data is necessary prior information for our method. To evaluate the impact originating from different input data quality, we created three different GIS road datasets per IKONOS image:

- SET_A: a “realistic” dataset: most of the digitised roads are correct and show a good positional accuracy, but some roads which are visible in the imagery are missing in the dataset, and some objects from the road database are not correct at all, i.e. they do not exist in the imagery. To simulate a realistic dataset, errors were manually introduced into the EuroSDR reference dataset (Figure 5-dotted red lines).
- SET_B: the EuroSDR reference dataset which is correct in every aspect (Figure 5-yellow lines)
- SET_C: a totally incorrect road data set.

Set_A reasonably simulates situations where a road database exists and is used for the training. Set_B is chosen to have an idea how well the correct dataset can be extracted if the same data is used for the training. Set_C is chosen to be able to conduct a sensitivity analysis, i.e. to test the approach with

useless prior information. Since both images have the same size, we simply chose as Set_C for a particular scene the Set_B of the other scene. The following results were obtained with geometric information from the different datasets. The attributes road width (cf. 3.1) and road class were not available in the EuroSDR reference dataset. Therefore, constant default values are used.

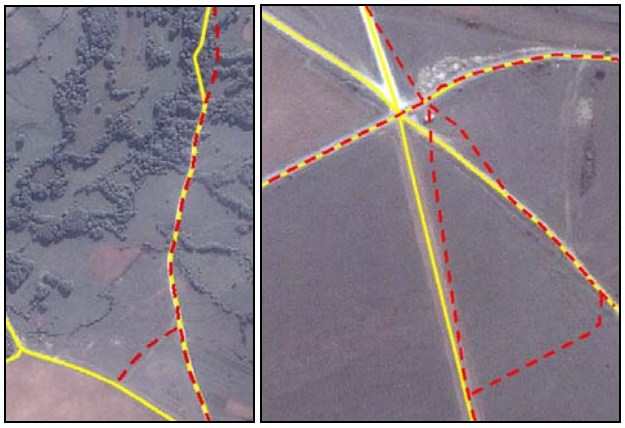


Figure 5: SET_A: realistic dataset (dotted red) and SET_B: reference dataset (yellow), exemplary zooms for tested imagery

The evaluation of the results is done applying the same reference data as used for the EuroSDR test. Additionally, the measures *completeness*, *correctness* and *RMS* are calculated using the same software as in the test, therefore the results can easily be compared. Firstly, only the results of radiometric parameter training are presented. Afterwards the results of the additional application of the colour segmentation step are shown.

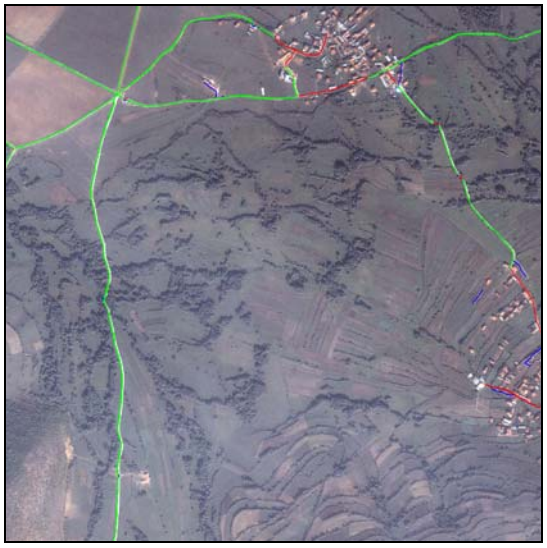
4.1 Training of radiometric parameters

The algorithm for parameter training and line extraction is separately applied to the NDVI and the intensity channel. Both channels provide complementary information for road extraction, as already shown in other research work. After line extraction, the results are fused and evaluated as a combined result. For Set_A we provide results using only the intensity channel and including also the NDVI channel. To guarantee the focus on radiometric parameters, the geometric parameters (for details see Wiedemann, 2002) are held constant for all used datasets and images.

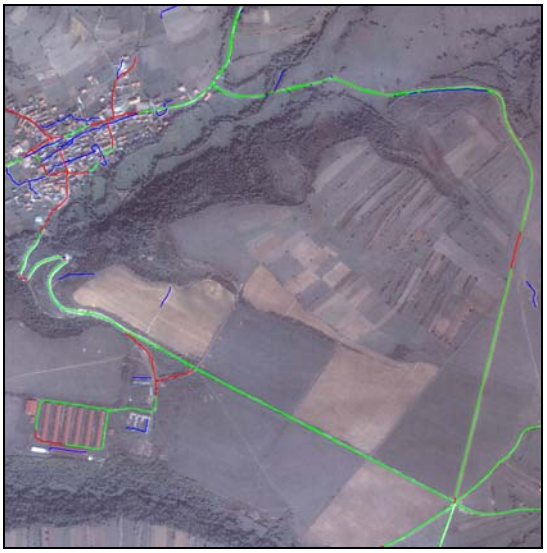
No	Name	Completeness	Correctness	RMS [pix]
IKONOS_3_Sub2				
1	Best_EuroSDR	0.85	0.91	1.59
2	Gerke_EuroSDR	0.75	0.52	1.35
3	SET_A_(noNDVI)	0.78	0.91	1.22
4	SET_A	0.83	0.90	1.22
5	SET_B	0.83	0.92	1.22
6	SET_C	(0.75)	(0.52)	(1.35)
IKONOS_3_Sub1				
7	Best_EuroSDR	0.81	0.87	0.97
8	Gerke_EuroSDR	0.80	0.65	1.53
9	SET_A_(noNDVI)	0.75	0.75	1.20
8	SET_A	0.77	0.76	1.42
9	SET_B	0.76	0.75	1.40
10	SET_C	(0.80)	(0.65)	(1.53)

Table 1: Evaluation Results. Grey rows are from (Mayer et al. 2006)

Table 1 contains the best result from the EuroSDR test together with those obtained in our investigations. Generally, the extraction results, focused on open landscape areas, are nearly complete and correct, shown in Figure 6a and Figure 6b.
Compared with Best_Euro_SDR (Karin Hedman) of IKONOS_3_Sub2 our RMS is smaller. The Best_Euro_SDR (Uwe Bacher) of IKONOS_3_Sub1 summarises a very good RMS and high completeness and correctness. This advantage compared to our approach is mainly caused by an enhanced extraction in the built-up area.
The use of NDVI is not necessarily advantageous because many roads are dirt roads, which are covered by vegetation.

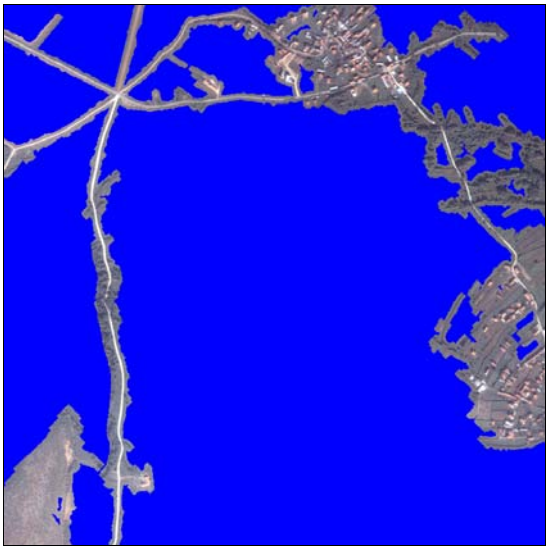


(a) IKONOS_3_Sub2

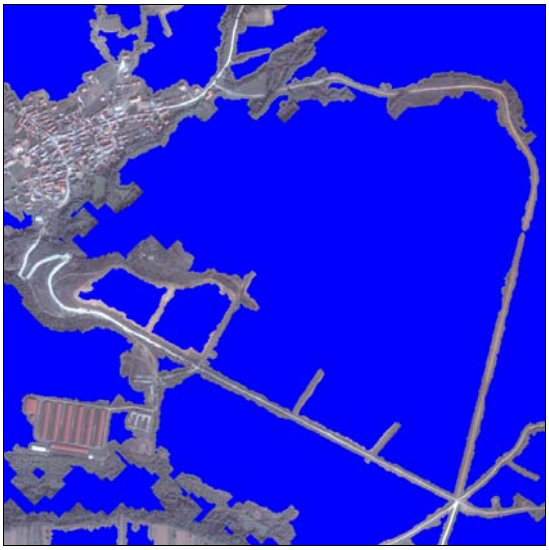


(b) IKONOS_3_Sub1

Figure 6: EuroSDR test images, SET_A-extraction results: Correctly extracted roads are given in green, incorrectly extracted roads in blue and missing roads in red.



(a) IKONOS_3_Sub2



(b) IKONOS_3_Sub1

Figure 7: ROIs, trained with SET_A

The small differences between the results from SET_A and SET_B show that the rank filter works efficiently. Therefore, realistic imperfect databases can be used for our problem. The parameter training algorithm has also registered the completely incorrect datasets (SET_C) and did not result in plausible parameters. Thus, useless prior information is detected automatically in our approach. In order to still extract roads the human operator has to manually select appropriate parameters. In this case, the results correspond with Gerke_EuroSDR, because in both tests the same parameters set where selected.

Compared with Gerke_EuroSDR in particular the correctness of the recent tests is enhanced. The main reason for this is the tuning of the global threshold parameter.

4.2 Colour segmentation

The ROI results for the line extraction algorithm based on colour segmentation are shown in Figure 7. Large areas with multispectral signature of trained roads are found within settlement areas. Therefore, urban areas are shown as compact ROIs. The generated regions contain nearly all road objects including local background - only the large background areas are excluded from line extraction.

No	Name	Completeness	Correctness	RMS [pix]
IKONOS_3_Sub2				
4	SET_A	0.82 (-0.01)	0.90 (±0.00)	1.22
5	SET_B	0.82	0.91	1.22
IKONOS_3_Sub1				
8	SET_A	0.77	0.79 (+0.03)	1.42
9	SET_B	0.76	0.78	1.40

Table 2: Evaluation results. Numerical values in brackets describe differences towards prior results.

The colour segmentation algorithm is used as a separate module for the whole road extraction process. Therefore, the overall result can be compared to the results presented in Table 1. The evaluation (Table 2) shows no significant enhancement. Only for IKONOS_3_Sub1, the correctness is slightly increased, because some incorrect line objects are excluded. However, in the IKONOS_3_Sub2 image a correct object is missing, because its multispectral combination was not sufficiently represented in the training dataset, resulting in a marginally smaller completeness value.

Because of the mainly bright roads and the dark background, the usage of the trained global threshold as shown with the former test and the calculation of ROI from colour segmentation achieve similar results. Thus, in this case the additional colour information does not seem to be necessary for road extraction.

As before the usage of realistic (SET_A) and perfect reference dataset (SET_B) achieves similar results.

5. CONCLUSIONS AND OUTLOOK

In this paper, we present two methods to incorporate prior information into a line-based road extraction algorithm. The first method aims at parameter estimation for the line extraction. The parameters being automatically tuned are the contrast between road and background, the homogeneity within the road objects and a global threshold for masking out non-road areas. The method turns out to be robust against errors in the available prior information and it was shown that the obtained results are better than results which have been achieved with the same road extraction algorithm and manually tuned parameters.

Multispectral colour information is not directly used within our basic method – the road extraction is applied to intensity or NDVI channel. The second refinement introduced here makes explicit use of the available colour information. Non-road areas are identified based on a statistical analysis of the RGB colour space and the available prior information on road data. The results obtained with this second method in combination with the first method do not significantly differ from the results where only parameter tuning was applied. This observation can be traced back to the fact that the background in the given images is relatively homogeneous and therefore the estimation of the global threshold as done with the first method nearly masks out the same region as the multispectral approach. Additional tests will further investigate this issue.

In the EuroSDR test to which we refer it was stated that the current automatic approaches available in the literature and implemented in some prototypes have gained already practical relevance, at least for open landscape regions. From a practical point of view, the refinements presented in this paper should contribute to an even better acceptance of a road extraction system, because the tiresome tuning of parameters is significantly reduced. A prerequisite, however, is that the database with the prior information contains all kinds and classes of desired roads.

The parameter tuning can be further enhanced, in particular regarding the geometric parameters for road extraction. For instance, to extract winding roads we needs, other parameters compared to the extraction of straight highways. The prior information from the GIS database can be used to determine the types of roads contained in the current scene and consequently the respective parameters can be optimised.

The colour segmentation method needs also to be enhanced in several regards. Three issues are in the focus of our ongoing work. In our method we do not use the infrared channel yet. In regions where the background of a road object is densely vegetated and the road surface itself is sealed, the incorporation of the infrared information is supposed to contribute valuable additional information; this was also shown by other participants of the EuroSDR test. Additionally, the radiometric information content of the used 8 bit image is significantly reduced compared to the original 11 bit images. In future we will make use of the complete radiometric information. Finally, we are currently investigating means for alternative clustering in RGB-(IR)-(frequency)-feature space.

The presented method is being used in a system for the assessment and updating of existing GIS databases (Busch et al., 2004, Gerke, 2006). It is expected that the use of both methods introduced in this paper will contribute to an increased efficiency and reliability of the developed system.

ACKNOWLEDGEMENTS

The test and reference data was made available by Helmut Mayer and Uwe Bacher. The work is funded by the Bundeswehr Geoinformation Office (AGeoBw), Euskirchen, Germany. We gratefully acknowledge this support. We would also like to thank the anonymous reviewer for their valuable comments.

REFERENCES

Bacher U., Mayer H., 2005. Automatic road extraction from multispectral high resolution satellite images. In: *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. XXXVI, Part B3/W24, pp. 29–34.

Bacher U., 2006. Automatische Straßenextraktion aus optischen Satellitenbildern hoher und höchster Auflösung, Dissertation, Universität der Bundeswehr München.

Baumgartner A., Steger C., Mayer H., Eckstein W., Ebner H., 1999. Automatic road extraction based on multi-scale, grouping, and context. *Photogrammetric Engineering & Remote Sensing*, Vol. 65, 7, pp. 777–785.

Bordes G., Giraudon G., Jamet O., 1997. Automatic road extraction from grey-level images based on object database. In *Integrating Photogrammetric Techniques with Scene Analysis and Machine Vision III* , Vol. 3072, pp. 110–118.

Busch A., Gerke M., Grünreich D., Heipke C., Liedtke C.-E., Müller S., 2004. Automated verification of a topographic reference dataset: system design and practical results, In: *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. XXXV, Part B2, pp.735–740.

Doucette P., Agouris, P., Musavi M., Stefanidis A., 1999. Automated extraction of linear features from aerial imagery using Kohonen learning and GIS data. In: *Lecture Notes in Compute Science*, Vol. 1737, pp. 20–33.

Gerke M., Butenuth M., Heipke C.,Willrich F.,2004. Graph supported verification of road databases. *ISPRS Journal of Photogrammetry and Remote Sensing*, 58(3-4), pp. 152-165

Gerke M., 2006. Automatic quality assessment of road databases using remotely sensed imagery. PhD thesis, Deutsche Geodätische Kommission. Reihe C, Dissertationen, Nr. 599.

Grelck H., Müller-Grunau G. 2005. MGCP – nur eine neue Abkürzung im GeoInfoDBw? In: *Geo Forum, Mitteilungen des Geoinformationsdienstes der Bundeswehr*, Vol 1/2005, pp. 23–24.

Hinz S., 2004. Automatische Extraktion urbaner Straßennetze aus Luftbildern. PhD thesis, Deutsche Geodätische Kommission. Reihe C, Dissertationen, Nr. 580.

Hinz S., Baumgartner A. 2003. Automatic Extraction of Urban Road Networks from Multi-View Aerial Imagery. *ISPRS Journal of Photogrammetry and Remote Sensing*, 58/1-2, pp. 83 - 98.

Hinz S., Wiedemann C., 2004. Increasing Efficiency of Road Extraction by Self-Diagnosis. *ISPRS Journal of Photogrammetry and Remote Sensing*, 70(12), pp. 979–986.

Mayer H., Baltsavias E., Bacher U., 2006. Automated extraction, refinement, and update of road databases from imagery and other data. *Report Commission 2 on Image Analysis and Information Extraction*, European Spatial Data Research - EuroSDR, Official Publication 50, pp. 217-280.

Mena J., Malpica J., 2005. An automatic method for road extraction in rural and semi-urban areas starting from high resolution satellite imagery. *Pattern Recognition Letters*, Vol. 26, pp. 1201–1220.

Steger C., 1998. An unbiased detector of curvilinear structures. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 20, 2, pp. 311–326.

Wiedemann C., 2002. Extraktion von Straßennetzen aus optischen Satellitenbildern. Deutsche Geodätische Kommission. Reihe C, Dissertationen, Nr. 551.

Wiedemann, C. 2003: External Evaluation of Road Networks. In: *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. XXXIV, Part 3/W8

Wiedemann C., Ebner H., 2000. Automatic completion and evaluation of road networks. In: *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. XXXIII, Part B3/2, pp. 979–986.

Zhang C., Baltsavias E.P, 2002. Improving cartographic road databases by image analysis. In: *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. XXXIV, Part B3, pp. 400-405.

Zhang, C. (2004). Towards an operational system for automated updating of road databases by integration of imagery and geodata. ISPRS Journal of Photogrammetry and Remote Sensing, Vol. 58, 3-4, pp.166–186.

Zhang Q., Couloigner I., 2006. Automated road network extraction from high resolution multi-spectral imagery. In: ASPRS 2006 Annual Conference, Reno, Nevada

RECTANGULAR ROAD MARKING DETECTION WITH MARKED POINT PROCESSES

Olivier Tournaire[1,2], Nicolas Paparoditis[2], Florent Lafarge[2]

[1]UMLV / OTIG, 5, Bd Descartes, Champs-sur-Marne 77454 Marne-la-Valle CEDEX 2 - France
[2]IGN / MATIS, 2-4, Ave Pasteur, 94165 S^t-Mandé - France

{olivier.tournaire;nicolas.paparoditis;florent.lafarge}@ign.fr

Commission III/5

KEY WORDS: Road-marks, aerial images, RJMCMC, Simulated annealing, Stochastic geometry

ABSTRACT:

We propose in this article an energy minimization based approach to detect dashed lines of road markings from very high resolution aerial images ($<$ 20 cm). The strategy we presented in a previous work was based on the grouping of image features (i.e. segments) extracted from the images (reference, n.d.), but suffers from a few problems such as occlusions mainly due to cars or changing in illumination conditions like shadows areas produced by trees or buildings or low contrast due to worn out markings.
In order to obtain a more robust and reliable extraction of those objects, we have to find another approach which can take into account those observations. In this context, we use marked point process which are random variables whose realization are configuration of geometrical shapes (rectangle in our case). We use an energy formulation based on both an external and internal terms. A RJMCMC sampler coupled with a simulated annealing is used to find the optimal object configuration according to the proposed energy. Some results are shown on various aerial images in urban areas.

1 INTRODUCTION

Automatic road extraction has been a challenging topic since many years in the geographic and photogrammetric communities (Mayer et al., 2006). Road markings can be an interesting object for their update or completion in rural as in dense urban areas. In fact, they are useful to extract topological information such as intersections or semantic ones (i.e. road functionality, number of circulations lanes ...).
With a different goal, some authors use road markings informations in their system. Those objects have been considered particularly interesting for autonomous navigation of land vehicles (Charbonnier et al., 1997). Indeed, road marks are invariant elements and can easily be recognized by terrestrial based camera sensors in order to provide visual landmarks. They can then be used in a matching process in order to find or match a vehicle trajectory.

In an aerial context, only few papers addressed this topic. This is most probably due to the fact that this kind of objects are quite small related to the ground sampling distance of the images used for road extraction systems. They are however very structuring notably in urban areas where accurate road detection stays a vast challenge.
Many approaches have been developed for the extraction of complex road networks. Some of them lie on clues detection such as road marks which is a very good indicator of the attendance of road. For example, (Zhang, 2003) tries to find zebra-crossings with a colorimetric analysis in order to obtain the main road direction. (Hinz and Baumgartner, 2002), with a radiometric model of dashed lines extract those objects for the same purpose, also introducing geometric regularity analysis of the detected elements. A graph approach is presented in (Steger et al., 1997) to group road marks patterns. A best-first-search is performed and leads to find an optimal path in the graph to group those features. Closer to what we present in this paper, (Lacoste et al., 2005) detects road networks with a stochastic approach using marked point processes (segments objects). One should note that the method is extensible to various network forms. RJMCMC is also of interest in terrestrial photogrammetry for facades features extraction and interpretation (Mayer and Reznik, 2006, Brenner and Ripperda, 2006).
The strategy we will address in this article is based on the same approach. Our aim is to provide an energetic modeling of the dashed road marks lines. We will use marked point processes modeled by rectangular patterns. Each point of the process will stand for a strip of the lines we propose to extract. The method is fully automatic and is based on an homogeneous mathematical framework which provides robustness to the process. Such a technique is particularly interesting for the following reasons:

- it allows a modeling using simple geometric objects (rectangle),
- it allows the introduction of prior knowledge related to the object layout, which is particularly interesting for the management of occlusions and low contrast,
- the process converges towards the optimal solution for any initial configuration.

To present our algorithm we first have an overlook on marked point process. Then, we propose an energy formulation based on both a data term which measures the coherence between the objects configuration and the image, and a regularizing term which takes into account some interactions existing between neighboring objects. A RJMCMC sampler coupled with a simulated annealing is used to find the optimal object configuration according to the proposed energy. Finally, results on Amiens downtown are presented.

2 MODEL DEFINITION

We now present how we build the energy of the model. Before going into details, we come back on essentials aspects of marked points processes.

2.1 Marked Point Processes

Point processes have been introduced in image processing by (Baddeley and Van-Lieshout, 1993) so as to detect an unknown number of objects in an image. In this theoretical frame, the problem is to locate the objects in the image and to measure them. Such modeling has been applied to numerous problematics in many research areas, as for example buildings reconstruction (Lafarge et al., 2006) or road extraction (Lacoste et al., 2005) from satellite or aerial imagery.
As show on figure 1, dashed lines are composed of several rectangular objects separated from each others with a fixed distance depending on the road type or functionality. The rectangles also have the same width for a given line. To describe the base object of our process, we need three parameters : its position (x_i, y_i) and its orientation θ_i (see figure 2). The width w and the length h of the rectangle are fixed according to the image GSD. In practice, h corresponds to 15 cm and ω is 1.5 m or 3 m according to the scene.

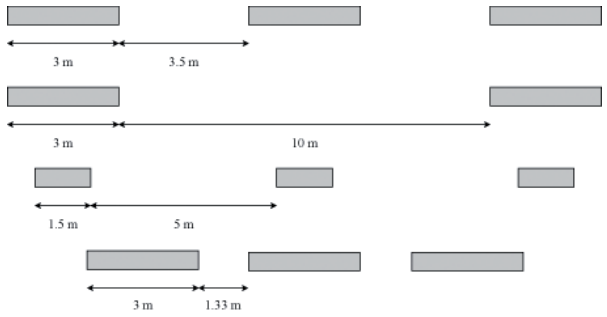


Figure 1: Geometric characteristics of the discontinuous lines objects'.

To summarize, an object of our marked process is distributed in the space $\mathcal{M} = [0; X_{max}] \times [0; Y_{max}] \times [0; \pi]$. $(x_i, y_i) \in [0; X_{max}] \times [0; Y_{max}]$ ... and θ_i stands in the interval $[0; \pi]$. A realization of our marked point process is an element of $\mathcal{M}^n$, $n \in \mathbb{N}$.

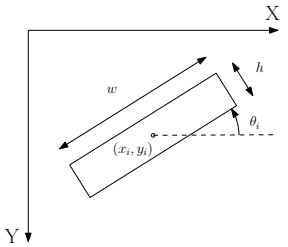


Figure 2: The object i: the point (x_i, y_i) and its associated mark θ_i.

2.2 Energy formulation

We aim at proposing a model for dashed lines detection of the road marking. We will thus have to build an energetic formulation of the modeling. The global energy $\mathcal{U}$ of the model is composed of two terms - a data attachment term and a regularizing term - following equation 1:

$$\mathcal{U} = \beta \mathcal{U}_{ext} + (1 - \beta)\, \mathcal{U}_{int} \tag{1}$$

Each term is described in the next paragraphs. First of all, we need to describe the base objects of the process and their associated marks.

2.2.1 Internal energy $\mathcal{U}_{int}$

The internal term allows to introduce prior knowledge concerning the object layout. This regularizing term is developed through interactions existing between neighboring objects. We first need to setup a neighborhood relationship.

Neighborhood
Two objects i and j are said neighbors if they verify the following equation (equation 2).

$$i \sim j \Leftrightarrow d\left(\overline{i}, \overline{j}\right) \leq r_{\mathcal{D}} \tag{2}$$

where $\overline{i}$ is the center of the object i and $r_{\mathcal{D}} = w + d_{inter-stripes} + \epsilon_{\mathcal{D}}$. $d_{inter-stripes}$ directly comes from the specifications and $\epsilon_{\mathcal{D}}$ is used to allow a tolerance on the distance between two consecutive stripes.

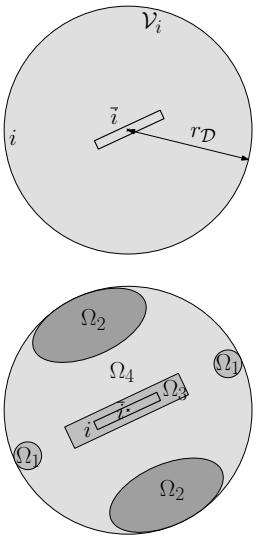


Figure 3: The neighborhood $\sim$ of an object i (top) and its different interest areas (bottom).

Interaction principle
The internal term is computed for pairs of objects in interaction. In other words, if x is a configuration of objects, the internal term is expressed as:

$$\mathcal{U}_{int}(x) = \sum_{i \sim j} \mathcal{U}_P(i, j) \tag{3}$$

We detail $\mathcal{U}_P$ in the following. Interactions have to be differentiate in accordance with the relative orientation and position of the objects inside the neighborhood (see figure 3 and equations 4 and 5). The neighborhood is thus divided into several areas: an attractive one (Ω_1) which is the union of two discs, a neutral one (Ω_2) composed of two ellipses and a repulsive one ($\Omega_3 \cup \Omega_4$).

$$\mathcal{U}_P(i, j) = \begin{cases} \mathcal{U}_P^*(i, j) & \text{if } |\theta_i - \theta_j| \leq \epsilon_\theta \\ \mathcal{A} & \text{otherwise} \end{cases} \tag{4}$$

$$\mathcal{U}_P^*(i, j) = \begin{cases} \mathcal{B} & \text{if } \overline{j} \in \Omega_1 \\ 0 & \text{if } \overline{j} \in \Omega_2 \\ \mathcal{C} & \text{if } \overline{j} \in \Omega_3 \\ \mathcal{D} & \text{if } \overline{j} \in \Omega_4 \end{cases} \tag{5}$$

If two neighbors objects have a high angular difference, they are penalized and are affected with a regularizing value of $\mathcal{A}$. Otherwise, four cases are distinguished. In order to highly penalize

objects overlapping, Ω_3 is defined as a rectangle surrounding the base object i. If an object j lies in this area, the relation between i and j is set to the maximum value. In the remainder of the repulsive area, the value ($\mathcal{D}$) is a little low in order to avoid disadvantage too much objects between circulation lanes (like directional arrows). In Ω_2, the neutral area, the value of the energy is set to 0. This is useful when the road is composed of parallel circulation lines as shown on figure 4.

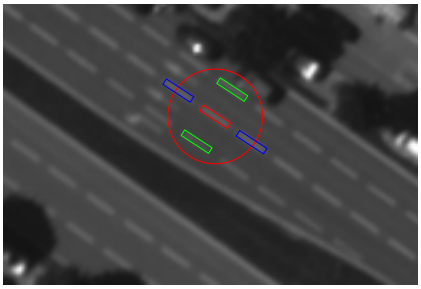


Figure 4: Illustration of the problem of parallel circulation lanes when the neighborhood is not parted with Ω_2. The base object i is the red one. Blue ones are located in the attractive area. If Ω_2 is not defined, the green objects would have been affected a high energy value.

Finally, if j stands in Ω_1, the objects i and j are considered to be attracted each other and they are affected with the lowest regularizing energy value. The closest $\overline{j}$ is from the center of Ω_1, the lowest the energy is. The energy is defined thanks to the function f (equation 6) inspired by a Laplace distribution:

$$f(x) = \frac{1}{2b} e^{-\frac{|x-t|}{b}} \tag{6}$$

2.2.2 External energy $\mathcal{U}_{ext}$

The data term indicates the likelihood of the objects of the model in relation with the image. With the same notations as in equation 3, we have:

$$\mathcal{U}_D(x) = \sum_{i \in x} \mathcal{U}^i_{int} \tag{7}$$

were $\mathcal{U}^i_{int}$ is the external energy of each object of the x configuration.
For dashed lines modeling, we simply have noticed that the strips forming a line are clear objects on a darker background. We also make the hypothesis that pixels within the objects and in their adjacent neighborhood are homogeneous. From this, the external energy is defined in terms of region analysis.
In order to avoid thresholds, we use a statistical criterion to decide if two regions are radiometrically distinct. This way, we define for an object two kinds of points: internal and external ones (see figure 5).

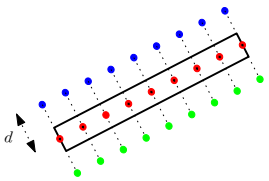


Figure 5: Inner and outer points of an object.

We thus have three sets of pixels obtained by slicing the rectangle. In order to test if the object is well positioned, we compute the Mahalanobis distance (Mahalanobis, 1936) between each set of outer points and the inner points. For two regions $\mathcal{R}_1$ and $\mathcal{R}_2$, it is given by equation 8.

$$d_{Maha.}(\mathcal{R}_1, \mathcal{R}_2) = n \left(\overline{p^{\mathcal{R}_1}} - \overline{p^{\mathcal{R}_2}}\right)^2 \left[\sum_{j \in \{\mathcal{R}_1, \mathcal{R}_2\}} \sum_{k=1}^{n} \left(p^j_k - \overline{p^j}\right)^2\right]^{-1} \tag{8}$$

where n is the number of pixels in a region, $\overline{p^{\mathcal{R}_1}}$ (*resp.* $\overline{p^{\mathcal{R}_2}}$) is the radiometric mean of the internal pixels (*resp.* external) and p^j_k is the radiometry of the pixel $k \in \{1, \ldots, n\}$ from the region $j = \{\mathcal{R}_1, \mathcal{R}_2\}$. The Mahalanobis distance follows a χ^2 law. In our case, it has two degrees of freedom. Thus, with a risk α, it is possible to obtain from the χ^2 distribution function a threshold which allows to decide if the regions we are currently testing are radiometrically dissimilar. From this, the external energy is given by equation 9.

$$\mathcal{U}^i_{ext}\left(d_{Maha.}(in, out), t_{\chi^2}\right) = -\frac{d_{Maha.} - t_{\chi^2}}{\sqrt{1 + \frac{\left(d_{Maha.} - t_{\chi^2}\right)}{2}}} \tag{9}$$

This function has a high derivative around t_{χ^2} and is null when the Mahalanobis distance is equal to t_{χ^2}. When inner and outer regions are dissimilar, the numerator takes a high positive value and the energy is then highly negative. The shape of the function is given on figure 6 where the x-coordinate is the Mahalanobis distance and the y one is the data energy value.

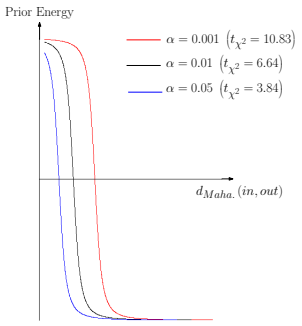


Figure 6: Data energy value for different error risks α.

2.2.3 Parameters settings

Physical parameters
This kind of parameters have a physical interpretation in the application. They are fixed according to the scene.
For example, w and $d_{inter-stripes}$ are used is the computation of the radius of the $\sim$ relation. They directly depend on the kind of dashed line we want to extract. The different possible values are defined from the specifications as show on figure 1.

Weights and thresholds
Data and regularizing terms are weighted one compared to the other, respectively with a factor β and $1 - \beta$. β is tuned by *"trial and errors"*. Let us say that the data term is around 60% to 75% of the total. A better way to obtain this value is discussed in the conclusion.
Some thresholds are also necessary in our model. The first one, ϵ_d, is set in order to have a tolerance on the radius of the $\sim$ relation. We need it because specifications are strong, but sometimes, they are not respected and the distance between two consecutive stripes can be a little bit more than what it should be. The other threshold, ϵ_θ, is used when computing internal energy. It leads to consider a particular relation between two stripes having a weak angular difference. In our application, we use $\epsilon_d = 0.5$ m pixels

and $\epsilon_\theta = \frac{\pi}{8}$.

3 OPTIMIZATION

We aim at finding the configuration of objects which minimizes the energy $\mathcal{U}$. This is a non convex optimization problem in a high and variable dimension space since the number of objects is unknown.

3.1 RJMCMC sampler

The Reversible Jump Markov Chain Monte Carlo (RJMCMC) algorithm (Geyer and Møller, 1994, Green, 1995) is well adapted to our problem. Several papers have shown the efficiency of the RJMCMC sampler for the marked point processes problems (Geyer and Møller, 1994, Lacoste et al., 2005). To do so, an non normalized density $h(.)$ is defined through the energy $\mathcal{U}$ thanks to the Gibbs relation:

$$h(.) = \exp -\mathcal{U}(.) \tag{10}$$

The RJMCMC sampler consists in simulating a discrete Markov Chain $(X_t)_{t\in\mathbb{N}}$ on $\mathcal{R}$ the space of the configurations having π as invariant measure (specified by the density $h(.)$) which performs "small jumps" between spaces of variable dimensions respecting the reversibility assumption of the chain. Propositions are based on "small jumps" which means only one object of the global configuration will be concerned by a new proposition. One of the main advantages of such a sampler is that the chain asymptotically converges towards π for any initial configuration X_0. It means that we do not need specific initial object configuration. The jumps are proposed according to various kinds of kernels Q_m specified in the following:

- **Birth and death** kernels allow to informally add / remove an object in / from the current configuration. These two transformations, which correspond to jumps in spaces of higher (birth) / lower (death) dimension, guaranty that the Markov Chain visits the whole configuration space. However, it is important to define relevant moves in order to speed up the convergence of the Markov chain.

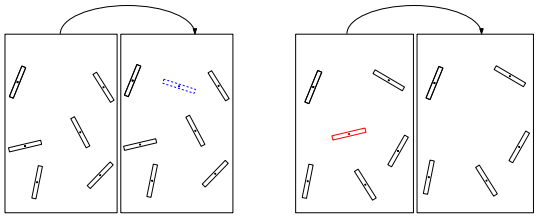


Figure 7: Birth uniform (left) and death uniform (right).

- **Birth and death in a neighborhood** kernels, introduced by (Green, 1995), consists in adding / removing an object in the neighborhood of a current object. In other words, it allows to propose objects in the areas of interest.

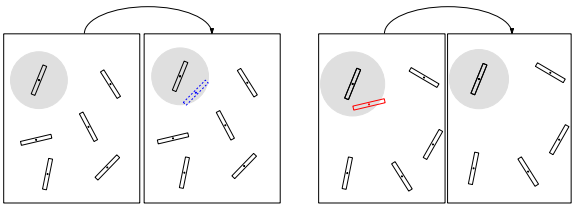


Figure 8: Birth (left) and death (right) in a neighborhood.

- **Perturbation** kernels are composed of rotation and translation moves. This kind of kernels is very useful for adjusting the positioning of the objects.

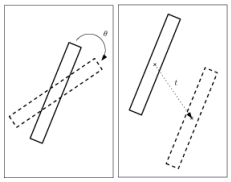


Figure 9: Rotation and translation propositions.

Let us summarized the RJMCMC sampler. At iteration t, if $X_t = x$:

- Choose the kernel $Q_i(x, .)$ with probability q_i
- According to Q_i, propose a new state y
- Take $x^{(t+1)} = y$ with probability:

$$\min\left(\frac{\pi(dy)}{\pi(dx)}\frac{Q_i(y,dx)}{Q_i(x,dy)}, 1\right)$$

- And take $x^{(t+1)} = x$ otherwise

Algorithm 1: RJMCMC sampler algorithm

3.2 Simulated annealing

A simulated annealing is used to ensure the convergence process: the density $h(.)$ is substituted by $h(.)^{\frac{1}{T_t}}$ where T_t is a sequence of temperatures which tends to zero as t tends to infinity. The simulated annealing allows to theoretically ensure the converge to the global optimum for all initial configuration x_0 using a logarithmic temperature decrease. In practice, we prefer using a geometrical decrease which is faster and gives an approximate solution close to the optimal one. The initial and final temperatures are estimated through the variation of the energy, using the work of (White, 1984).

4 RESULTS

We now present some results of our algorithm with aerial images acquired on the downtown of Amiens with a GSD of 25 cm.
Figure 10 shows the evolution of the process on a simple example and its ability to deal with curved structures. At the beginning of the algorithm (i.e. when the temperature is high - see 10 a)) the process explores the density modes. When the temperature decreases, the process begins to be selective (see 10 d) and e)) and the rectangles begins to be well located. At low temperature (see 10 g) and h)) the configuration is close to the optimal one: it consists in adjusting the parameters of the objects of the configuration.
Some statistics associated with the results of figure 10 are presented on figure 11. They show how does the configuration evolves and tends to converge.

Figure 12 shows an other example of the results obtained with our algorithm. Some details illustrate how the algorithm overcome difficulties of vehicles occlusions or shadows area due to trees. However, on this example, some rectangles are missing, and some others are detected where no pattern exist. We can certainly cope with those miss detection and false detection with tuning more precisely the weights of the both energy terms.

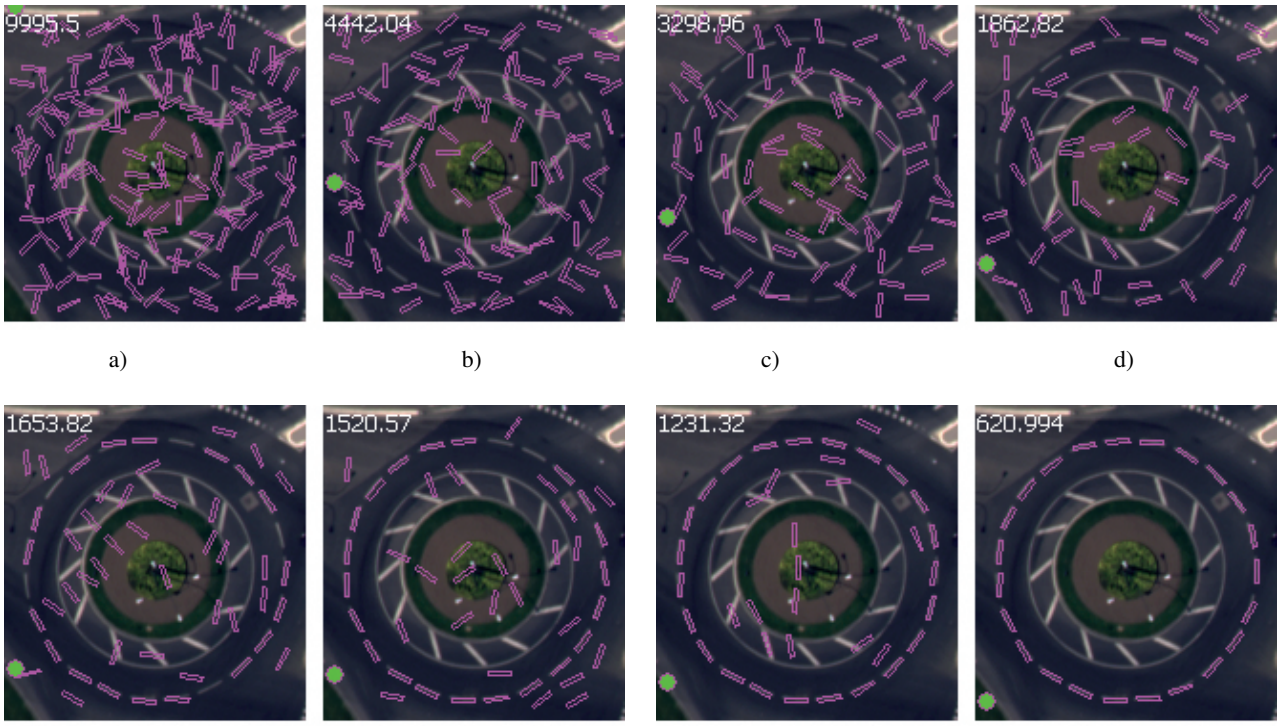


Figure 10: The algorithm in evolution (the figures correspond to the unormalized temperature used in the simulated annealing). Amiens - GSD 25 cm. From left to right and top to bottom : initial configuration - the firsts steps - a few objects find their position - more and more objects are well positioned - all objects are on road marks but a few are still present with a bad position - finally, all objects are well positioned and no more objects are misplaced.

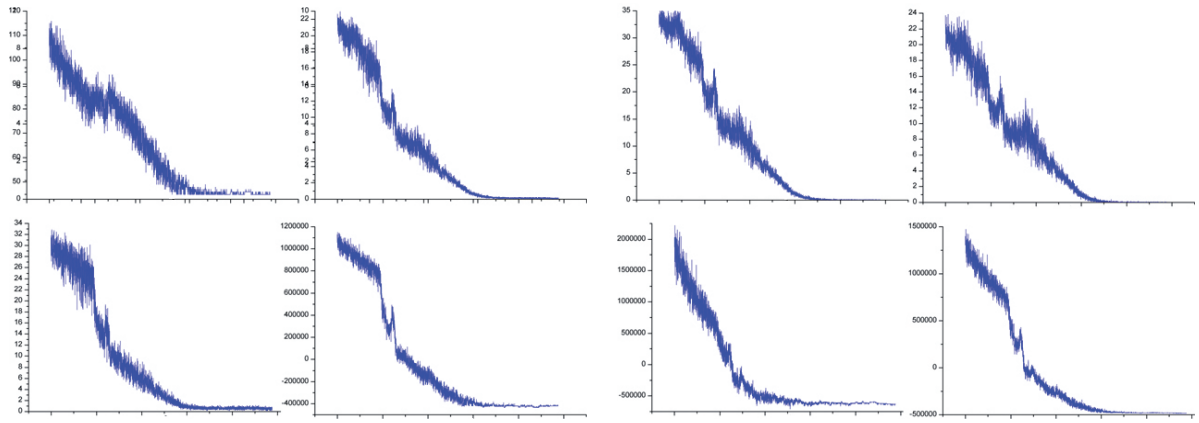


Figure 11: Statistics associated with the result show on figure 10. From left to right and top to bottom : number of objects - global acceptance rate - birth acceptance rate - death acceptance rate - perturbation acceptance rate - external energy - internal energy - total energy.

Figure 12: The algorithm in evolution. Amiens - GSD 25 cm

5 CONCLUSION AND FUTURE WORKS

We have presented in this paper an algorithm based on stochastic geometry to detect rectangular road marking. Marked point processes are well adapted to detect this kind of objects fully automatically and without an initialization. Using simulated annealing with a RJMCMC sampler for the model optimization allows us to choose randomly the initialization without any impact on the result. Prior knowledge lead to the detection of strictly linear or curved structures in aerial images. It also allows us to deal with total occlusions due to vehicles or partial ones due to shadows, thus bringing a high level of robustness to the model.
However, some issues have to be studied in the future. The first one will be to introduce in the regularizing term knowledge in order to take into account changes in the lines structure. Next, it will be interesting to try to find the weighting coefficients with an automatic algorithm approach such as *Expectation Maximization* (Dempster et al., 1977) instead of tuning it by *trial and error*. Finally, in order to reduce computation time, it could be interesting to develop data driven kernels allowing to reduce the search space. Investigations on this point are currently being studied.

REFERENCES

Baddeley and Van-Lieshout, M., 1993. Stochastic geometry models in high-level vision. Statistics and Images 1, pp. 233–258.

Brenner, C. and Ripperda, N., 2006. Extraction of facades using RJMCMC and constraint equations. In: International Archives of Photogrammetry and Remote Sensing, Vol. XXXVI - 3, pp. 155–160.

Charbonnier, P., Diebolt, F., Guillard, Y. and Peyret, F., 1997. Road markings recognition using image processing. In: IEEE Conference on Intelligent Transportation System, Vol. 1, pp. 912–917.

Dempster, A., Laird, N. and Rubin, D., 1977. Maximum likelihood from incomplete data via the EM algorithm. Journal of the Royal Statistical Society B39, pp. 1–38.

Geyer, C. and Møller, J., 1994. Simulation and likelihood inference for spatial point processes. Scandinavian Journal of Statistics 21, pp. 359–373.

Green, P., 1995. Reversible jump markov chain monte carlo computation and bayesian model determination. Biometrika 82, pp. 711–732.

Hinz, S. and Baumgartner, A., 2002. Urban road net extraction integrating internal evaluation model. In: ISPRS Commission III Symposium on Photogrammetric Computer Vision, Vol. XXXIV - Part 3A, Graz, Austria, pp. 163–168.

Lacoste, C., Descombes, X. and Zerubia, J., 2005. Point processes for unsupervised line network extraction in remote sensing. IEEE Trans. Pattern Analysis and Machine Intelligence 27(10), pp. 1568–1579.

Lafarge, F., Descombes, X., Zerubia, J. and Pierrot-Deseilligny, M., 2006. An automatic building reconstruction method : A structural approach using high resolution images. In: Proc. of IEEE International Conference on Image Processing, Atlanta - USA.

Mahalanobis, P., 1936. On the generalized distance in statistics. In: Proc. Nat. Inst. Sci. India, Calcutta - India.

Mayer, H. and Reznik, S., 2006. MCMC linked with implicit shape models and plane sweeping for 3D building facade interpretation in image sequences. In: International Archives of Photogrammetry and Remote Sensing, Vol. XXXVI - 3, pp. 130–135.

Mayer, H., Hinz, S., Bacher, U. and Baltsavias, E., 2006. A test of automatic road extraction approaches. In: International Archives of Photogrammetry and Remote Sensing, Vol. XXXVI - 3, pp. 209–214.

reference, A. A., n.d.

Steger, C., Mayer, H. and Radig, B., 1997. The role of grouping for road extraction. In: A. Gruen, E. Baltsavias and O. Henricsson (eds), Automatic Extraction of Man-Made Objects from Aerial and Space Images (II), Birkhäuser Verlag, Basel, Switzerland, pp. 245–256.

White, S., 1984. Concepts of scale in simulated annealing. In: Proceedings IEEE, International Conference on Computer Design, Port Chester, pp. 646–651.

Zhang, C., 2003. Updating of cartographic road databases by image analysis. PhD thesis, Institute of Geodesy and Photogrammetry, Zurich.

SPATIO-TEMPORAL MATCHING OF MOVING OBJECTS IN OPTICAL AND SAR DATA

S. Hinz[1], F. Kurz[2] D. Weihing[1], S. Suchandt[2]

[1]Remote Sensing Technology, Technische Universitaet Muenchen, 80290 Muenchen, Germany
[2]Remote Sensing Technology Institute, German Aerospace Center, 82235 Wessling
{Stefan.Hinz | Diana.Weihing}@bv.tu-muenchen.de ; {franz.kurz | steffen.suchandt}@dlr.de

KEY WORDS: Vehicle Detection, Vehicle Tracking, Traffic Monitoring, Traffic Parameters, Optical images, SAR images

ABSTRACT:

We present an approach for spatio-temporal co-registration of dynamic objects in Synthetic Aperture Radar (SAR) and optical imagery. Goal of this work is the performance evaluation of vehicle detection and velocity estimation from SAR images when comparing it with reference data derived from aerial image sequences. The results of evaluation show the challenges of traffic monitoring with SAR in terms of detection rates for individual vehicles.

1. INTRODUCTION

Increasing traffic has major influence on urban und suburban planning. Usually traffic models are utilized to predict traffic and forecast transportation. To derive statistical parameters of traffic for these models, data of large areas acquired at any time is desirable. Therefore, spaceborne SAR missions can be a solution for this aim. With the upcoming TerraSAR-X or RADARSAT-2 mission, SAR images up to 1 m resolution will be available. Additionally, the Dual Receive Antenna (DRA) mode enables the reception of two SAR images of the same scene within a small timeframe, which can be utilized for along-track interferometry.

In preparation of these missions, a variety of algorithms for vehicle detection and velocity estimation from SAR has been developed; see e.g. (LIVINGSTONE et al., 2002, GIERULL, 2004, MEYER et al, 2006). An extensive overview on current developments and potentials of airborne and spaceborne traffic monitoring systems is given in the compilation of (HINZ et al., 2006). It shows that civilian SAR is currently not competitive with optical images in terms of detection and false alarm rates, since the SAR image quality is negatively influenced by Speckle noise as well as layover and shadow effects in case of city areas or ragged terrain. However, in contrast to optical systems, SAR is an active and coherent sensor enabling interferometric and polarimetric analyzes making data acquisition independent from weather and illumination conditions. While the superiority of optical systems for traffic monitoring are in particular evident when illumination conditions are acceptable, SAR has the advantage of being illumination and weather independent, which makes it to an attractive alternative for data acquisition in case of natural hazards and crisis situations. Hence, validating the quality of SAR traffic data acquisition is crucial to estimate the benefits of using SAR in such situations. It is of particular importance to observe in which way fair detection results influence more generic parameters like mean velocity per road segment.

In this paper, an approach for evaluating the performance of detection and velocity estimation of vehicles in SAR images is presented, which utilizes reference traffic data derived from simultaneously acquired optical image sequences. While the underlying idea of this approach is naturally straightforward, the different sensor concepts imply a number of methodological challenges that need to be solved in order to compare the dynamics of objects in both types of imagery.

2. EFFECTS OF MOVING OBJECTS IN SAR IMAGES

As it is well known, the SAR principle of exploiting the platform motion to enhance the resolution in azimuth (i.e. along-track) direction by forming a long synthetic antenna causes image derogations when objects of the imaged scene move during RADAR illumination. The most significant effects are defocusing due to along-track motion and displacement due to across-track motion. Accelerations influence the imaging process in a similar way (see, e.g. (MEYER et al, 2006, SHARMA et al, 2006)). We briefly summarize the most important relations in the following. For a more extensive overview, we refer the reader to (MEYER et al, 2006; HINZ et al, 2007).

2.1 Along-Track Motion

To quantify the impact of a significantly moving object we first assume the point to move with velocity v_{x0} in azimuth direction. The relative velocity of sensor and scatterer is different for the moving object and the surrounding stationary world. Thus, along track motion changes the frequency modulation rate *FM* of the received scatterer response. Forming the synthetic aperture with a conventional Sationary World Matched Filter (SWMF, (BAMLER & SCHAETTLER, 1993; CUMMING & WONG, 2005)) consequently results in a blurring of the signal. The width Δt of the peak can be approximated by

$\Delta t \approx 2T_A \frac{v_{x0}}{v_B}[s]$ with T_A being the synthetic aperture time and v_B the beam velocity on ground. As can be seen, the amount of defocusing depends strongly on the sensor parameters. A car traveling with 80km/h, for instance, will be blurred by approx. 30m when inserting TerraSAR-X parameters (MEYER et al, 2006). However, it has to be kept in mind that this approximation only holds if $v_{x0} >> 0$.

2.2 Across-Track Motion

When a point scatterer moves with velocity v_{y0} in across-track direction, this movement causes a change of the point's range history proportional to the projection of the motion vector into the line-of-sight direction of the sensor $v_{los} = v_{y0}\sin(\vartheta)$, with ϑ being the local elevation angle. In case of constant motion during illumination the change of range history is linear and causes an additional linear phase trend in the echo signal. Correlating such a signal with a SWMF results in a focused point that is shifted in azimuth direction by

$t_{shift} = \frac{2v_{los}}{\lambda \cdot FM}[s]$ in time domain, respectively by $\Delta_{az} = -R\frac{v_{los}}{v_{sat}}[m]$ in space domain where λ is the carrier frequency, v_{sat} the satellite velocity and v_{los} the object velocity projected into the sensor's line of sight. In other words, across-track motion leads to the fact that moving objects do not appear at their "real-world" position in the SAR image but are displaced in azimuth direction – the so-called "train-off-the-track" effect. Again, when inserting typical TerraSAR-X parameters, the displacement reaches an amount of 1.5km for a car traveling with 80km/h in across-track direction. Figure 1 shows an example of the combination of both effects. Due to across track motion a car is displaced from its real-word position on the road (green arrow in Figure 1a). In addition, the car is defocused because of along track motion when processed with a SWMF (Figure 1b). If it was filtered with the correct reference signal, the point should be sharp as in Figure 1c.

Across-track motions not only influence the position of an object in the SAR image but also the interferometric phase in case of an along-track interferometric data acquisition, i.e., the acquisition of two SAR images within a short time frame with baseline Δl aligned with the sensor trajectory. The interferometric phase is defined as the phase difference of the two co-registered SAR images $\psi = \varphi_1 - \varphi_2$ and is proportional to motions in line-of-sight direction. Hence, the interferometric phase can also be related to the displacement in space domain:

$$\Delta_{az} = -R\frac{v_{los}}{v_{sat}} = -R\psi\frac{\lambda}{4\pi \cdot \Delta l}[m]$$

2.3 Accelerations

In the majority of the literature, it is assumed that vehicles travel with constant velocity and along a straight path. If vehicle traffic on roads and highways is monitored, target acceleration is commonplace and should be considered in any processor or realistic simulation. Acceleration effects do not only appear when drivers physically accelerate or brake but also due to curved roads, since the object's along-track and across-track velocity components vary on a curved trajectory during the Radar illumination. The effects caused by along-track or across-track acceleration have recently been studied in (SHARMA et al., 2006, MEYER et al., 2006). Summarizing, along-track acceleration results in an asymmetry of the focused point spread function, which leads to a small azimuth-displacement of the scatterer after focusing, whose influence can often be neglected. However, the acceleration in across-track direction causes a spreading of the signal energy in time or space domain. The amount of this defocusing is significant and comparable with that caused by along-track motion. Its influence for the following matching is however negligible since defocusing appears purely in along-track direction.

3. MATCHING CARS IN OPTICAL AND SAR DATA

The quality of SAR based traffic monitoring can be assessed for large areas when using simultaneously acquired aerial image sequences as reference data. Yet matching dynamic objects in SAR and optical data remains challenging since the two data sets do not only differ in geometric properties (Section 3.1) but also in temporal aspects (Section 3.2) of imaging.

3.1 Geometric co-registration

Digital frame images, as used in our approach, inhere the well-known radial perspective imaging geometry that defines the mapping [X, Y, Z] => [x_{img}. y_{img}] from object to image co-ordinates. The spatial resolution on ground (ρ_X, ρ_Y, cf. Figure 2) is mainly depending on the flying height H, the camera optics with focal length c and the size of the CCD elements (ρ_x, ρ_y). Whereas, SAR images result from time/distance measurements in range direction and parallel scanning in azimuth direction defining a mapping [X, Y, Z] => [x_{SAR}, R_{SAR}]. 3D object co-ordinates are thus mapped onto circles with radii R_{SAR} parallel aligned in azimuth direction x_{SAR}. The spatial resolutions (ρ_R, ρ_{SA}) of range and azimuth dimension are mainly depending on the bandwidth of the range chirp and the length of the physical antenna after SAR focusing.

To accommodate for the different imaging geometries of frame imagery and SAR, we employ a Digital Elevation Model (DEM), on which both data sets are projected. Differential rectification can then be conducted by direct georeferencing of both data sets, if the exterior orientation of both sensors is precisely known. In case the exterior orientation lacks of high accuracy – which is especially commonplace for the sensor attitude – an alternative and effective approach is to transform an existing ortho-image into the approximate viewing geometry at sensor position C:

$$[x_C, y_C] = f(p_{ortho}, X_{ortho}, Y_{ortho}, Z_{ortho})$$

where p_{ortho} is the vector of approximate transformation parameters. Refining the exterior orientation reduces then to finding the relative transformation parameters p_{rel} between the given image and the transformed ortho-image, i.e.

$$[x_{img}, y_{img}] = f(p_{rel}, x_C, y_C),$$

which is accomplished by matching interest points. Due to the large number of interest points, p_{rel} can be determined in a robust manner in most cases. This procedure can be applied to SAR images in a very similar way – with the only modification that, now, p_{ortho} describe the transformation of the ortho-images into the SAR slant range geometry.

The result of geometric matching consists of accurately geo-coded optical and SAR images, so that for each point in the one data set a conjugate point in the other data set can be assigned. However, geometrically conjugate points may have been imaged at different times. This is crucial for matching moving vehicles and has not been considered in the approach outlined so far.

3.2 Time-dependent matching

Frame cameras take snapshots of a scene at discrete time intervals with a frame rate of, e.g., 0.3 – 3Hz. Due to overlapping images, most moving objects are imaged at multiple times. SAR, in contrast, scans the scene in a quasi-continuous mode with a PRF of 1000 – 6000 Hz, i.e. each line in range direction gets a different time stamp. Due to the parallel scanning principle, a moving vehicle is imaged only once, however, as outlined above, possibly defocused and at a displaced position. Consequently, the two complementary sensor principles of SAR and optical cameras lead to the fact that the time of imaging a moving object differs for both sensors.

Figure 2 compares the two principles: It shows the overlapping area of two frame images taken at position C_1 at time t_{C1} and position C_2 at t_{C2}, respectively. A car travelling along the sensor trajectory is thus imaged at the time-depending object co-ordinates $X(t = t_{C1})$ and $X(t = t_{C2})$. On the other hand, this car is imaged by the SAR at Doppler-zero position $X(t = t_{SAR0})$, i.e. when the antenna is closest to the object. It illustrates that exact matching the car in both data sets is not possible because of the differing acquisition times. Therefore, a temporal interpolation along the trajectory is mandatory and the specific SAR imaging effects must be considered.

Temporal matching includes thus following steps:

- Reconstruction of a continuous car trajectory from the optical data by piecewise interpolation (e.g. between control points $X(t = t_{C1})$ and $X(t = t_{C2})$ in Figure 2). Alternatively, GIS road axes could be used if they were accurate enough.
- Calculation of a time-continuous velocity profile along the trajectory, again using piecewise interpolation.

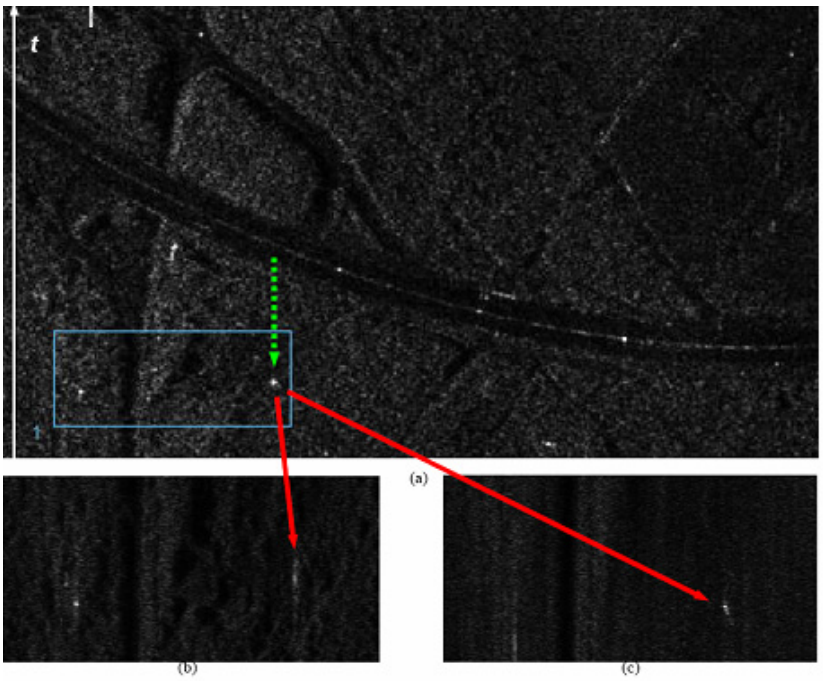


Figure 1. a) SAR image of a highway section with displaced car due to across track motion (green arrow). b) Detail of a): Defocused car when processed with a SWMF due to along track mostion. c) Same part, however, processed with a filter corresponding to the car's along track velocity. Now the car is imaged sharply while the background gets blurred.

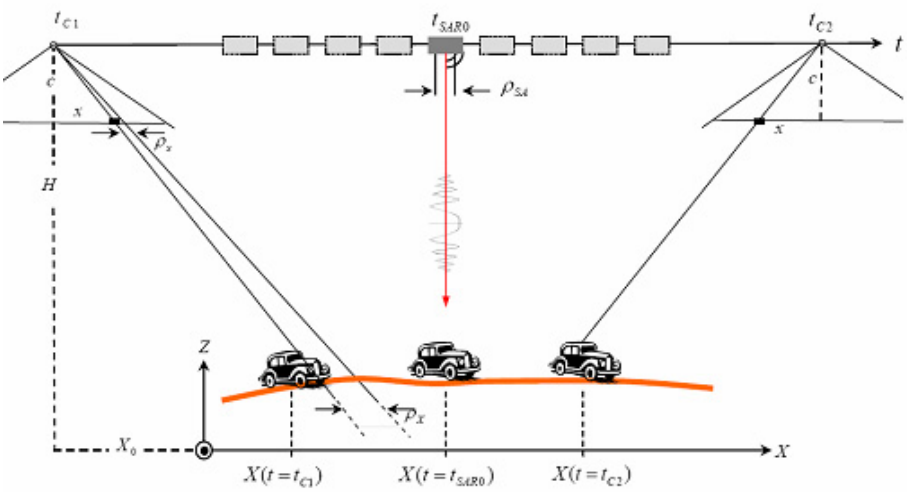


Figure 2: Moving objects in optical image sequence and SAR image in azimuth direction

- Derivation of a maximum velocity-variance profile. The velocity variance at the control points depends purely on the imaging and measurement accuracy (see Section 4.1). To propagate the variance into the interpolated regions, we employ a simple and empirically tested dynamic model defining that the variance between control points follows a parabolic shape as exemplified in the cut-out of Figure 3. This model accommodates the fact that velocity interpolation gets less accurate with grater distance to the adjacent control points. Together with the velocity profile, it defines an uncertainty buffer, i.e. a minimum and maximum velocity for each point along the trajectory.
- Transforming the trajectory into the SAR image geometry and adding the displacement due to the across track velocity component. In the same way, the uncertainty buffer is transformed.
- Intersection/matching of cars detected in the SAR image with the trajectory by applying nearest neighbour matching. Cars not being matched are defined as false alarms.

As result, each car detected in the SAR data and not labeled as false alarm is assigned to a trajectory and, thereby, uniquely matched to a car found in the optical data. Figure 3 visualizes intermediate steps of matching: a given highway section (magenta line); the corresponding displacement area color coded by an iso-velocity surface; a displaced track of a smoothly decelerating car (green line); and a cut-out of the displaced uncertainty buffer. Two cars correctly detected in the SAR image are marked by red crosses in the cut-out. The local RADAR co-ordinate axes are indicated by magenta arrows.

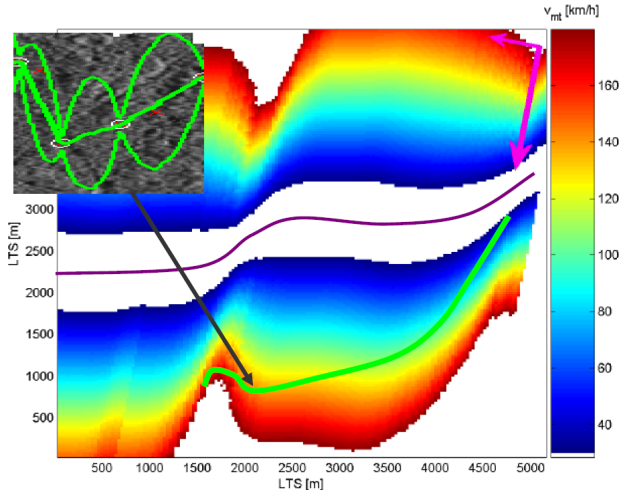


Figure 3. Intermediate steps of matching: highway section (magenta line), corresponding displacement area (color coded by iso-velocity surface), displaced track of a decelerating car (green line), local RADAR coordinate system (magenta arrows). Cut-out shows a detail of the displaced uncertainty buffer. Cars correctly detected in the SAR image are marked by red crosses.

4. ACCURACY AND VALIDATION

In order to validate the matching and estimate the accuracy, localization and velocity determination have been independently evaluated for optical and SAR imagery.

4.1 Accuracy of velocity estimation from optical images

The basic concept of determining the accuracy of vehicle measurements in optical images is the comparison of theoretically derived accuracies with empirical accuracies measured with airborne images of reference cars.

Vehicle velocity $v_{I2\text{-}I}$ derived from two consecutive co-registered or geo-coded optical images *I1* and *I2* is simply calculated by the displacement Δs over the time elapsed Δt. The displacement can be calculated through the transformed coordinate differences in the object space or by the pixel differences multiplied with a scale factor m in co-registered images.

$$v_{I2-1} = \frac{\Delta s}{\Delta t} = \frac{\sqrt{(X_{I2} - X_{I1})^2 + (Y_{I2} - Y_{I1})^2}}{t_{I2} - t_{I1}} = m \frac{\sqrt{(r_{I2} - r_{I1})^2 + (c_{I2} - c_{I1})^2}}{t_{I2} - t_{I1}}$$

where X_{Ii} and Y_{Ii} are object coordinates, r_{Ii} and c_{Ii} the pixel coordinates of moving cars, and t_{Ii} the acquisition times of images $i=1,2$.

Using factor m simplifies the calculation of theoretical accuracies, since the calculation is separated from the geo-coding process. Thus, three main error sources on the accuracy of car velocity are of interest: the measurement error σ_P in pixel units, the scale error σ_m assumed to be caused mainly by DEM error σ_H, and finally the time error σ_{dt} of the image acquisition time.

Figure 4 shows accuracies of vehicle velocities derived from positions in two consecutive acquired images based on calculation of error propagation. For this, different assumptions about the error sources must be made. The measurement error σ_P is defined as *1.0* pixel including co-registration errors, the time distance error σ_{dt} as *0.02s*, which corresponds to the registration frequency of the airplane navigation system, and finally a DEM error σ_H of 10m is assumed. The simulation in Figure 4 shows decreasing accuracy at higher car velocities and shorter time distances, as the influence of the time distance error gets stronger. On the other hand, the accuracies decrease with higher flight heights as the influence of measurement errors increases. Last is converse to the effect, that with lower flight heights the influence of the DEM error gets stronger.

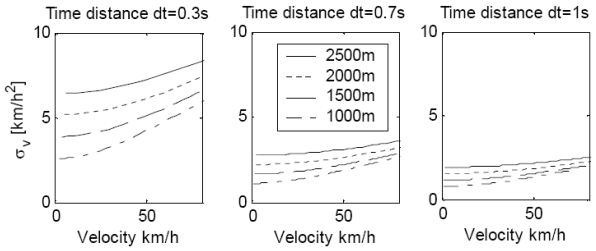


Figure 4. Accuracy of vehicle velocities derived from positions in two consecutive acquired images for three time differences 0.3s, 0.7s, and 1.0s. For each time distance, four airplane heights from 1000m up to 2500m and car velocities from 0 to 80 km/h were considered.

The theoretically calculated accuracies were validated with measurements in real airborne images and with data from a reference vehicle equipped with GPS receivers. The time distance between consecutive images was 0.7s. Exact assignment of the image acquisition time to GPS track times was a prerequisite for this validation and was achieved by connecting the camera flash interface with the flight control unit. Thus, each shoot could be registered with a time error less than 0.02s. Based on onboard GPS/IMU measurements, the images were geo-coded and finally resampled to a ground pixel size of 30cm.

Figure 5 illustrates the results of the validation for one car track. The empirically derived accuracies are slightly higher than theoretical values due to inaccuracies in the GPS/IMU data processing. Yet, it also shows that the empirical standard deviation is below 5km/h which provides a reasonable hint for defining the velocity uncertainty buffer in Section 3.2. The validation exemplifies on the other hand that vehicle accelerations cannot be derived from these image sequences with sufficient accuracy.

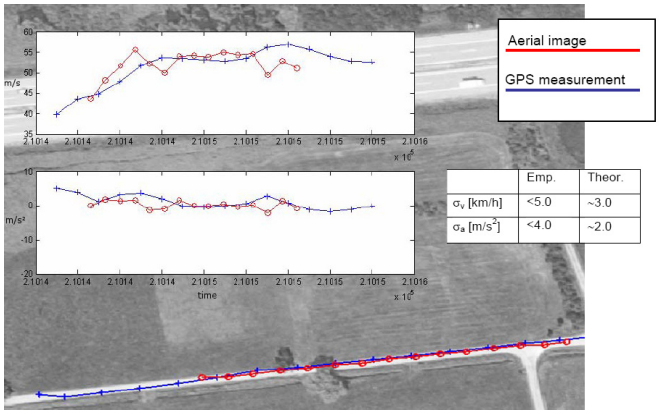


Figure 5 Vehicle positions (projected tracks), vehicle velocities (top figure), and accelerations (bottom figure) derived from airborne images and GPS measurements. Empirically measured and theoretically calculated accuracies are listed in the table.

4.2 Accuracy of velocity measurements in SAR images

Several flight campaigns have been conducted to estimate the accuracy of velocity determination from SAR images, thereby also verifying the validity of the above derived theory. An additional goal of the flight campaigns is to simulate TerraSAR-X data for predicting the performance of the extraction procedures. To this end, an airborne Radar system has been used with a number of modifications, so that the resulting raw data is comparable with the future satellite data. During the campaign 8 controlled vehicles moved along the runway of an airfield. All vehicles were equipped with a GPS system with a 10 Hz logging frequency for measuring their position and velocity. Some small vehicles were equipped with corner reflectors to make them better visible in the image. The experiments have been flown with varying angles between the heading of the aircraft and the vehicles. The vehicles have been driven with such velocities v_{Tn} that they approximately match traffic scenarios as recorded by satellites (see Table 1).

To estimate the accuracy, the predicted image position of a moving object is derived from the object's GPS position and its measured velocity and compared with the position measured in the image. The positions of displaced vehicles detected in the image (yellow dots in Fig 6) are compared with their true GPS-position (green dots) and the theoretical displacement computed from the GPS-velocities (red dots). As can be seen, yellow and red dots match very well, so that the theoretical background of detection and velocity estimation seems justified. Although there might be some inaccuracies included in the measurements (varying local incidence angle, GPS-time synchronization, etc.) the results show a very good match of theory and real measurements. As expected, target 7 is not visible in the image. This is due to the low processed band width (PBW) of only 1/10 of the PRF and the targets velocity. The across-track velocity of target 7 shifts the spectrum of the target outside of the PBW.

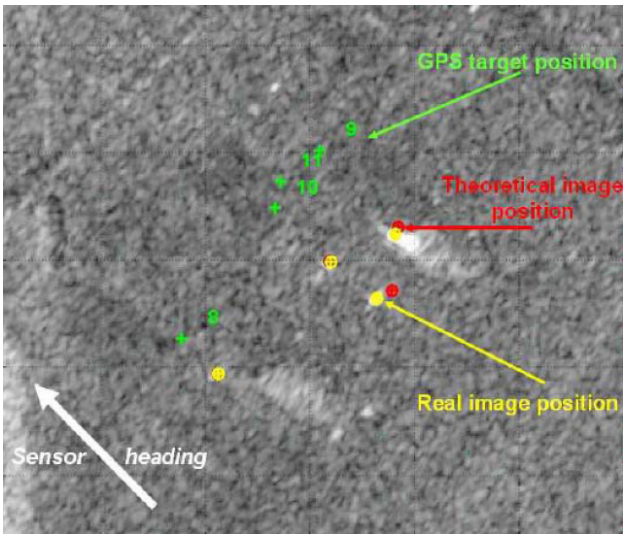


Figure 6. True GPS positions (green) of cars, displaced positions derived from GPS velocity (red), displaced position measured in the image (yellow).

To obtain a quantitative estimate of the quality of velocity determination SAR images, the velocity corresponding to the along-track displacement in the SAR images v_{Tn}^{disp} has been compared to the GPS velocity v_{Tn}^{GPS} (see Table 1). The numerical results show that the average difference between the velocity measurements is significantly below $1 km/h$. When expressing the accuracy of velocity in form of a positional uncertainty, this implies that the displacement effect influences a vehicle's position in the SAR image only up to a few pixels depending on the respective sensor parameters, as can be seen from Figure 6.

Target #	v_{Tn}^{GPS} [km/h]	v_{Tn}^{disp} [km/h]	Δv [km/h]
4	5.22	5.47	0.25
5	9.24	9.14	0.1
6	10.03	9.45	0.58
7	36.92	not visible	-
8	2.16	2.33	0.17
9	4.78	4.86	0.08
10	3.00	2.01	0.01
11	6.31	6.28	0.03

Table 1: Comparison of velocities from GPS and SAR

4.3 Matching results with real data

The matching approach has been tested on real data stemming from DLR's E-SAR and 3K optical system. The flight campaign aimed at monitoring a freeway nearby Lake Chiemsee, approx. 80 km in the south-east of Munich. The freeway is heading nearly in across-track leading to large displacements of the cars in the SAR image. During the flight, also optical images of the same scene have been acquired to enable the verification of the detection results. For ensuring error-free reference data, vehicle detection and tracking has been carried out manually. Some track sections are exemplified in Figure 7.

An existing modular traffic processor has been applied to detect vehicles in the SAR data automatically, see (SUCHANDT et al., 2006; WEIHING et al. 2007) for details. Different detectors (ATI, DPCA, likelihood ratio detector) are integrated for finding

vehicles and can be selected individually or can be combined. Figure 8 shows an example of vehicle detection with the likelihood ratio detector (WEIHING et al. 2007). Detected vehicles are marked with red rectangles at their displaced positions. The triangels represent the positions of these vehicles when backprojected to the assigned road, whereby their color indicates the estimated velocity ranging from blue to red (0 to 170 km/h). Having these detections projected back onto the road axis, it is possible to derive parameters describing the situation on the road and feeding them into traffic simulations and traffic prediction models.

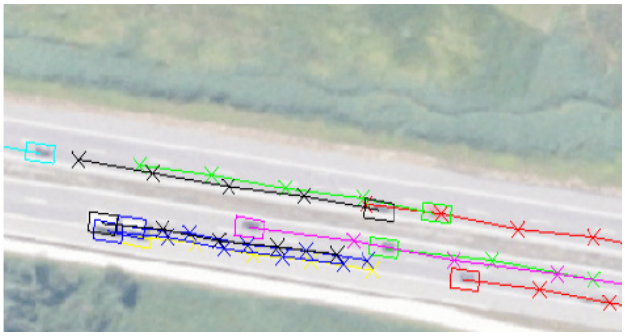


Figure 7. Example of vehicles tracked in optical image sequence

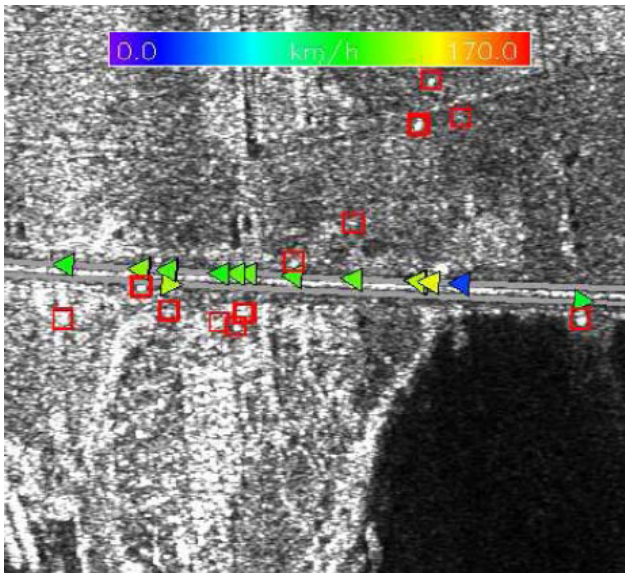


Figure 8. Cars detected in SAR image. Displaced position of detection (rectangle), backprojection onto road (triangle), estimated velocity (color of triangle).

The traffic data from the optical and the SAR system have been co-registered as described above to evaluate the performance of vehicle detection and velocity estimation. In Table 2 the traffic flow parameters derived from the detections with the likelihood ratio detector are compared to those estimated from the reference data. The vehicles moving on the upper lane from right to left are considered in this case. On the opposite lane only two vehicles have been detected which makes the derivation of reliable parameters impossible.

It can be seen from Table 2 that the detection rate is quite fair, as expected from other studies (e.g. (MEYER et al., 2006)). However, the results obtained for more generic traffic parameters are very encouraging, e.g. when comparing the values of the estimated mean of velocity, a good correspondence can be seen. Hence, even for a lower percent of detections in the SAR data, reliable parameters can be extracted. As has been shown in (SUCHANDT et al., 2006) one can derive, for instance, drive-through times for a road section from these data with high accuracy. Such information is highly useful for near-realtime traffic management since it allows to advising the drivers in choosing the best route.

Traffic parameters	SAR data	optical data
mean velocity	104 km/h	100 km/h
velocity range	29-129 km/h	81-135 km/h
number of vehicles	12	31
detection rate	39 %	100 %

Table 2: Traffic parameters for vehicles moving on the upper lane from right to left

5. SUMMARY AND CONCLUSION

In this article, an approach for spatio-temporal co-registration of dynamic objects in SAR and optical imagery has been presented. It was used to evaluate the performance of vehicle detection and velocity estimation from SAR images compared to reference data derived from aerial image sequences. The evaluation shows the challenges of traffic monitoring with SAR in terms of detection rate. However, the traffic *flow* parameters derived from these results show a good correspondence with the reference data, even for a low detection rate. Hence, traffic models can make use of such data to simulate and predict traffic or to even verify certain parameters of models.

REFERENCES

BAMLER, R. & SCHÄTTLER, B., 1993: SAR Data Acquisition and Image Formation, in: G. Schreier (Ed.), Geocoding: ERS-1 SAR Data and Systems, Wichmann-Verlag, 1993.

CUMMING, I. & WONG, F., 2005: Digital Processing of Synthetic Aperture Radar Data, Artech House, Boston, 2005.

GIERULL, C., 2004: Statistical Analysis of Multilook SAR Interferograms for CFAR Detection of Ground Moving Targets – IEEE Transactions on Geoscience and Remote Sensing **42**: 691–701.

HINZ, S., BAMLER, R. & STILLA, U., 2006: Theme issue "Airborne and spaceborne traffic monitoring". – ISPRS Journal of Photogrammetry and Remote Sensing **61** (3/4).

HINZ, S., MEYER, F., EINEDER, M. & BAMLER, R., 2007: Traffic monitoring with spaceborne SAR – Theory, simulations, and experiments. – Computer Vision and Image Understanding **106** (2/3): 231-244.

LIVINGSTONE, C.-E., SIKANETA, I., GIERULL, C., CHIU, S., BEAUDOIN, A., CAMPBELL, J., BEAUDOIN, J., GONG, S. & KNIGHT, T.-A, 2002: An Airborne Sythentic Apertur Radar (SAR) Experiment to Support RADARSAT-2 Ground Moving Target Indication (GMTI). – Canadian Journal of Remote Sensing **28** (6): 794–813.

MEYER, F., HINZ, S., LAIKA, A., WEIHING, D. & BAMLER, R., 2006: Performance Analysis of the TerraSAR-X Traffic Monitoring Concept – ISPRS Journal of Photogrammetry and Remote Sensing **61** (3/4): 225–242.

SHARMA, J., GIERULL, C. & COLLINS, M., 2006: Compensating the effects of target acceleration in dual-channel SAR-GMTI. – IEE Radar, Sonar, and Navigation **153** (1): 53–62.

SUCHANDT, S., EINEDER, M., MUELLER, R., LAIKA, A., HINZ, S., MEYER, F. & PALUBINSKAS, G., 2006: Development of a GMTI Processing System for the Extraction of Traffic Information from TerraSAR-X Data. – Proceedings of European Conference on Synthetic Aperture Radar: on CD.

WEIHING D., HINZ S., MEYER F., SUCHANDT S. & BAMLER R., 2007: An Integral Detection Scheme for Moving Object Indication in Dual-Channel High Resolution Spaceborne SAR Images. Proceedings of IEEE-ISPRS Workshop URBAN 2007, Paris, France, on CD.

A FORMAL MODEL AND MIXED-INTEGER PROGRAM FOR AREA AGGREGATION IN MAP GENERALIZATION

Jan-Henrik Haunert

Institut für Kartographie und Geoinformatik
Leibniz Universität Hannover
Appelstraße 9a, 30167 Hannover, Germany
jan.haunert@ikg.uni-hannover.de

KEY WORDS: GIS, Generalization, Land Cover, Abstraction, Automation, Modelling

ABSTRACT:

This paper presents a model and an optimization method for a problem that appears when reducing the scale of a topographic database. Such a database commonly contains areas of different land cover classes that define a planar subdivision. When reducing its scale, some areas become too small and need to be aggregated. In order to produce contiguous aggregates that are not smaller than a user-defined threshold, it is necessary to change the classes of some areas. As generalization intends to preserve the characteristic features of the map, we aim to change classes as little as possible. A second objective is to create simple, compact shapes. Based on a previous work that neglected this second objective, we define a more general problem in this paper that reflects both aims of generalization. The problem was proven to be NP-hard, meaning that it is unlikely to find an efficient solution. Therefore, we propose a mixed-integer program (MIP) and heuristics, which enable the production of near-optimal results. The paper concludes with the presentation of some results we obtained using our method.

1 INTRODUCTION

In topographic databases information about land use or land cover is commonly represented by areas that are assigned to different classes, such as settlement, water, or different kinds of vegetation. The areas in such a database collectively define a subdivision of the plane, i.e., overlaps and gaps are not allowed. Generalizing this kind of map requires algorithms for different problems (Bader and Weibel (1997)). A challenging task is the aggregation of areas, which aims to satisfy size thresholds for the target scale. In an earlier paper we proposed a method for this generalization problem based on mixed-integer programming - a technique for combinatorial optimization (Haunert and Wolff (2006)). This method ensures different kinds of constraints coming from the specifications of the data sets and produces solutions with minimum change of land cover classes. The results were promising, but it was observed that the resulting geometries were not compact. Figure 1 (left) shows an example of a map at the original scale and the result which was obtained according to the defined objective (right). The settlement in the result (red) contains a narrow isthmus that was created to satisfy the area constraint while expending a minimum cost for class changes. In order to avoid such complex shapes, we define compactness as additional objective in this paper. The possibilities for the application of different compactness measures are discussed.

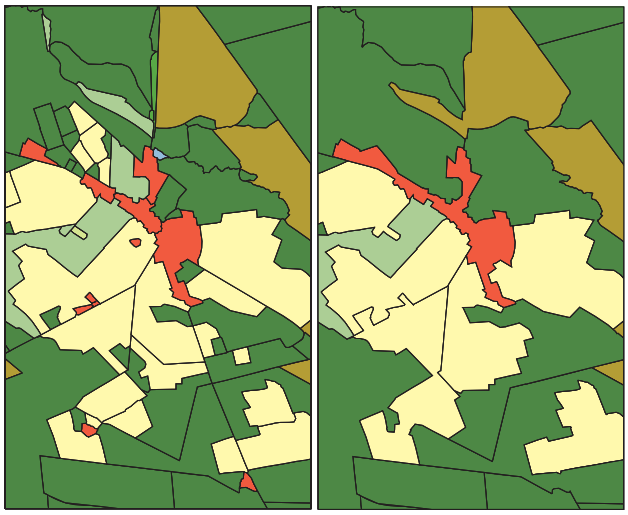


Figure 1: An example from the input data set at scale 1:50.000 (left) and a result for the scale 1:250.000 when minimizing changes of classes.

When designing optimization problems always two things need to be taken into account: The adequacy of the optimization objective and the possibility to solve the problem. Because of this, we concentrate on compactness measures that can be expressed by linear expressions. Regrettably, with this requirement, it is not possible to express size-invariant compactness measures. We discuss this deficit and its effects in detail. To cope with this, we add further requirements to the problem.

The paper is structured as follows: In Section 1.1 we discuss related work. Section 2 gives a formal problem definition and discusses the possibilities and difficulties to model compactness. In Section 3 we define a problem with additional requirements that allows to better express the cartographer's aim of generalization while getting along with the defined measures. In Section 4, we present our new mixed-integer program, results and an outline of an approach for the processing of large data sets. Finally we give a conclusion.

1.1 Related Work

The problem of area aggregation in map generalization has extensively been analyzed by researchers (Timpf (1998); van Smaalen (2003)). However, from an algorithmic point of view little success has been made in tackling its combinatorial nature. Different researchers have proposed iterative methods for the area aggregation problem. The following algorithm is described by van Oosterom (1995):

In each iteration the feature with lowest importance is selected. The selected feature is merged with a neighbor, which is chosen

according to a collapse function, and the next iteration is processed. The iteration can be terminated, if all areas satisfy the minimal dimension that is required for the target scale.

Many proposed algorithms are specializations of this general method. Jaakkola (1997) uses the method within a more comprehensive generalization framework for raster based land cover maps. Podrenek (2002) discusses preferences for merges, which reflects the collapse function. Generally, semantic similarity of classes, boundary lengths and area sizes are considered as criteria that need to be incorporated into the collapse function. The main problem with these iterative approaches is that consequences for future actions are not taken into account, when greedily selecting a neighbor. Therefore, we present a global approach in this paper.

Though there has not been any global optimization approach to area aggregation in map generalization, there exists a multiplicity of related problems that have been investigated by researchers. Especially, in the field of operations research, optimization methods for districting and aggregation problems have been developed. A typical application is the definition of sales districts presented by Zoltners and Sinha (1983). Their solution to find optimal districts is based on mathematical programming. As it is aimed to minimize distances between customers and stores, compactness is also aimed in their approach. We discuss the applied measure in Section 2.2.1. Other researchers have applied metaheuristics such as simulated annealing (Bergey et al. (2003)). The major disadvantage of these methods is the requirement for the definition of several tuning parameters, which are not inherent to the aggregation problem. Because of this, we concentrate on mathematical programming.

2 AGGREGATION PROBLEM

2.1 General Problem Statement

In this section, we first give a formal problem definition, which models the requirements and objectives of area aggregation in map generalization, and then explain this definition in detail. We simply refer to this problem as “Area Aggregation”.

Given

- a planar graph $G(V, E)$ with node weights $w : V \to \mathbb{R}^+$ and a coloring of nodes $\gamma : V \to \Gamma$, where Γ is the set of all colors, i.e. land cover classes,
- a function $\theta : \Gamma \to \mathbb{R}^+$, defining minimal allowed weights for colors,
- a function $d : \Gamma^2 \to \mathbb{R}_0^+$, expressing a distance between colors,
- a function $c : 2^V \times \Gamma \to \mathbb{R}_0^+$, defining the non-compactness of an aggregate,
- and a scalar weight factor $s \in [0, 1]$,

define a new coloring $\gamma' : V \to \Gamma$ of nodes and find a partition $P = \{V_1, V_2, \ldots, V_p\}$ of V, such that

- for each node set $V_i \in P$
  - the graph induced by V_i is connected,
  - all nodes in V_i receive the same new color $\gamma_i' \in \Gamma$, i.e., $\gamma'(v) = \gamma_i' \quad \text{for all} \quad v \in V_i$,
  - there is at least one node $v \in V_i$ with unchanged color, i.e., $\gamma'(v) = \gamma(v)$,
  - and V_i has total weight at least $\theta(\gamma_i')$,
- and the cost

  $s \cdot \sum_{v \in V} w(v) \cdot d(\gamma(v), \gamma'(v)) + (1 - s) \cdot \sum_{V_i \in P} c(V_i, \gamma_i')$

  is minimized.

The graph G is the dual graph of the planar subdivision. It contains a node for each shape and an edge between two nodes if the corresponding shapes share a common boundary. Node weights represent the sizes of areas. Land cover classes are represented by colors.

The defined requirements for connectivity and weight feasibility come from the specifications of data sets. Such specifications have been introduced as data standards by mapping authorities. To model minimal allowed area sizes that are defined for different land cover classes in the target scale, the weight threshold θ is defined as a function of color. Additionally, the requirement for a node with unchanged color in each part is introduced, to avoid that new classes pop up in the generalized map. Throughout this paper, such a node, which defines the color of an aggregate, will be referred to as *center*. Note that in this definition each node is a potential center. Figure 2 shows an instance of the problem and a solution, which is feasible according to the defined requirements. The partition P defines the shapes for the target scale, which can be obtained by geometrical union of shapes that correspond to the nodes contained in each element $V_i \in P$.

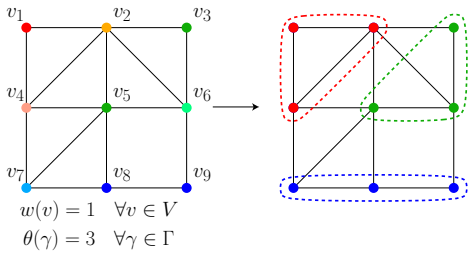


Figure 2: An instance of the aggregation problem (left) and a solution with $P = \{\{v_1, v_2, v_4\}, \{v_3, v_5, v_6\}, \{v_7, v_8, v_9\}\}$ (right).

The objective function expresses the cartographer’s preferences for different feasible solutions. Two different objectives can be identified: Firstly, it is aimed to change the original classes as little as possible. Secondly, compact shapes are preferred. To model these two aims, the two functions d and c are introduced, which are combined in a weighted sum. These functions need to be explained in detail.

The function d defines costs that are charged to change an area of unit size from one color into another. The values of this function could be given explicitly by a quadratic matrix with $|\Gamma| \times |\Gamma|$ elements. Generally, this matrix is not symmetric: Objects of rare land cover classes are often considered more important than others. It is unwanted to loose these objects; because of this, one will rather change a frequent class into a rare class than vice versa.

The function c defines a penalty being charged for the non-compactness of an aggregate, i.e., an area in the target scale that is defined by a subset of nodes and their new color (2^V refers to the power set of V, i.e., the set of all subsets of V). We assume that c attains high values for complex shapes, but we will simply

use the term compactness measure in the following. It is clear that explicitly expressing the values of this function is prohibitive due to limited time and storage space. However, no assumptions are made for this function here, in order to allow for different variations of the problem.

The complexity of this problem was investigated for the special case that compactness is neglected, i.e., for $s = 1$. Even for this case, the described problem was proven to be NP-hard, which means that it is very unlikely to find an efficient algorithm that reaches the optimal solution (Haunert and Wolff (2006)). This is still true, if $|\Gamma| = 2$, i.e., if the map only contains areas of two different classes. The NP-hardness justifies that we will later turn to mixed-integer programming and to heuristics.

2.2 Definition of Compactness

In the last section, the objective for compactness of shapes was expressed in a very general sense by the function c. This definition is specified here. Many different compactness measures have been proposed by researchers for the analysis of shapes. A detailed discussion is given by Maceachren (1985). The presented measures could be expressed as objectives to be optimized for the area aggregation problem.

However, before deciding for a specific model, it needs to be pointed out that this definition will influence the solvability of the problem. For this reason, we only consider those measures here that can be expressed by means of linear expressions, including integer and fractional variables. Additionally, we would like to bound the number of variables by a polynomial of low degree, e.g., quadratic in the input size. With this in mind, we discuss the possibilities to model compactness.

2.2.1 A Measure based on Distances to a Center A simple measure of compactness is defined by Zoltners and Sinha (1983). The aggregate defined by the set $V' \in 2^V$ contains a node $u \in V'$ whose corresponding shape defines the *geometrical center* of the aggregate by its centroid. The aggregate is considered to be compact, if the centroids of all other nodes in V' are close to this. To respect different sizes of areas, a penalty is charged for each node, which is equal to the product of the node's weight and the distance of its centroid from the geometrical center.

To apply this measure here, we claim that the geometrical center and the center according to Section 2.1 are defined by the same node. This definition slightly eases the problem, because less variables are needed to express the model. Nevertheless, it is reasonable, since it is undesired that nodes with unchanged color only appear at the margin of an aggregate. In other words, it is preferred, that nodes "gather around" a center of unchanged color. If there are several nodes with this color, then, among these, the center is defined by the node for which the overall penalty is minimal.

To formalize this measure, let $\delta : V^2 \to \mathbb{R}_0^+$ be the Euclidean distance between centroids of the shapes corresponding to two nodes. With this, we define the measure $c_1 : 2^V \times \Gamma \to \mathbb{R}_0^+$ as

$$c_1(V', \gamma') = \min \left\{ \sum_{v \in V'} w(v) \cdot \delta(v, u) \mid u \in V' \wedge \gamma(u) = \gamma' \right\} . \quad (1)$$

The function c_1 attains high values for complex shapes. Certainly, this measure only coarsely reflects the geometrical characteristic of a shape, since shapes are approximated by centroids. Because of this, the aggregate's perimeter is introduced as a second measure.

2.2.2 Measuring Compactness by the Perimeter of a Region The previously discussed iterative approaches to area aggregation in map generalization usually consider the length of boundaries as criterion when choosing a neighbor for merging (van Oosterom (1995)). To formalize the perimeter of an aggregate, let $\lambda : E \to \mathbb{R}^+$ be the length of the common boundary between two areas. Now, the perimeter $c_2 : 2^V \to \mathbb{R}_0^+$ becomes

$$c_2(V') = \sum_{e \in E'} \lambda(e) , \quad (2)$$

with E' being the set of edges incident to one node in V', i.e.,

$$E' = \left\{ \{u, v\} \in E \mid \left| \{u, v\} \cap V' \right| = 1 \right\} .$$

Similar to c_1, the compactness measure c_2 attains high values for complex shapes, which supposably have greater perimeters.

2.2.3 Discussion of Proposed Measures Both measures can result in side-effects, when being applied as global objectives. These need to be discussed. The measures c_1 and c_2 can result in two different biases:

1. When minimizing $\sum_{V_i \in P} c_1(V_i, \gamma_i')$, solutions with many small aggregates are preferred compared to solutions with few large aggregates. This is simply because average distances to centers are shorter for smaller aggregates.
2. When minimizing $\sum_{V_i \in P} c_2(V_i)$, solutions with few large aggregates are preferred compared to solutions with many small aggregates. In fact, when neglecting the objective for minimal color change, the globally optimal result would contain only one single aggregate, since in this case the total boundary length of the resulting partition would be minimal.

Both effects are due to the fact that the measures are not size invariant. In order to avoid these effects, the functions c_1 and c_2 could be normalized. However, we cannot satisfy the earlier claimed possibility for modeling the objective by means of linear expressions when using size invariant compactness measures.

It is important to note that aggregates will not become too small when applying c_1, since the size of each aggregate is bounded from below by the threshold θ. However, when applying the measure c_2 we run the risk of creating unintentionally large aggregates. To avoid this danger, we add additional hard requirements to the problem statement from Section 2.1. A detailed explanation of this method is given in Section 3.

3 AN APPROACH BASED ON PREDEFINED CENTERS

To avoid the creation of too large aggregates one could define an upper bound for the weights of the elements in the partition P, or a lower bound for the number of elements in P. Both definitions are probably too global and do not take local differences in the data set into account. Because of this, we chose another approach. It is based on a set of nodes that are predefined as centers. We give a general outline of this approach, formalize the modified problem and explain the definition of centers in detail.

3.1 Outline of Approach

In our previous paper (Haunert and Wolff (2006)), we proposed a heuristic that allowed for the elimination of certain variables. The idea was to fix relatively large areas as centers of aggregates. This

resulted in solutions with slightly higher values for the objective function, i.e., the total change of colors increased approximately by 10%. In the same way we subjectively perceived a decrease of quality.

Our first experiments with the proposed compactness measures, however, revealed that the defined objective function does not suffice to model our aim: Without fixed centers, the aggregates did not become compact enough when giving low weights to c_2 and the aggregates became unintentionally large for higher weights. The reason for this effect was explained in Section 2.2.3. Nevertheless, by fixing centers we did obtain nice and compact results. Since each aggregate can contain at most one center, the expansion of aggregates is limited. We assume that this model sufficiently reflects the aims of area aggregation in map generalization if the set of fixed centers is reasonably defined. Because of this, we include the previously defined heuristic in the problem statement.

3.2 Modified Problem Statement

The modified problem is defined as generalization of the problem Area Aggregation from Section 2.1. We refer to this problem as "Area Aggregation With Predefined Centers". In addition to an instance of Area Aggregation we require a set of predefined centers $C \subseteq V$ as input and define the constraints that for each node set $V_i \in P$

- at most one center is contained, i.e, $|V_i \cap C| \leq 1$ and
- if V_i contains a center $v \in C$, then all nodes $u \in V_i$ receive the color of the center, i.e., $\gamma'(u) = \gamma'_i = \gamma(v)$.

With the concept of predefined centres, the measure c_1 is generalized by the function $c_3 : 2^V \times \Gamma \rightarrow \mathbb{R}_0^+$:

$$c_3(V', \gamma') = \begin{cases} \sum_{v \in V'} w(v) \cdot \delta(v, u) & \text{if a node } u \text{ is in } V' \cap C\,, \\ c_1(V', \gamma') & \text{else, i.e., if } V' \cap C = \emptyset\,. \end{cases} \quad (3)$$

This simply means that if there is a predefined center in V', then the corresponding centroid defines the geometrical center of the aggregate, which is used to measure the compactness. Note that for $C = \emptyset$ the problem is the same as the original problem. Because of this, it is also NP-hard.

3.3 Definition of Centers

To define the set C, we recall that the application of c_1 as global objective is rather unproblematic, i.e., the influence on the size of aggregates is limited due to strict lower bounds for their weight. However, it was argued that the geometrical compactness is only coarsely reflected. We therefore propose a two-steps approach:

1. We solve the problem first, expressing compactness solely by the distances to a center, i.e., $c := c_1$. Presumably, such a solution is close to the result wanted by a cartographer.
2. Based on the resulting partition P, we define the set C to contain one node for each element in P. For this, we chose the center according to the measure c_1. With this definition, the problem can be solved a second time, this time applying the following combination of measures:

$$c := s' \cdot c_3 + (1 - s') \cdot c_2, \quad s' \in [0, 1] \quad (4)$$

The scalar weight factor s' is introduced to define a compromise of the two objectives c_3 and c_2. We discuss a solution of the problem in the next section.

4 A MIP FOR AREA AGGREGATION WITH PREDEFINED CENTERS

Different possibilities exist to model the aggregation problem as MIP. A difficult task is to express the connectivity of aggregates by means of variables and linear constraints. Williams (2002) and Shirabe (2005) have found different solutions for the problem of ensuring connectivity when selecting a subset of nodes from a graph. These approaches can be adopted in a straight forward way to model the aggregation problem, leading to a quadratic number of variables and constraints. Both methods have been implemented and tested using the software ILOG CPLEX 9.100 on a Linux server with 4 GB RAM and a 2.2 GHz AMD-CPU. In conclusion, the obtained running time was prohibitive – the largest instance that could be solved contained only 30 nodes. An improvement was made using a new MIP based on a single commodity flow model that requires only a linear number of variables and constraints (Haunert and Wolff, 2006). Still, without heuristics, it was not possible to process more than 50 areas.

Because of these experiences and the absence of existing approximation algorithms, heuristics need to be applied. Therefore we now define a more restrictive requirement for the connectivity of aggregates. This leads to an alternative MIP formulation. A similar approach was used by Zoltners and Sinha (1983) for the problem of optimally defining sales territories.

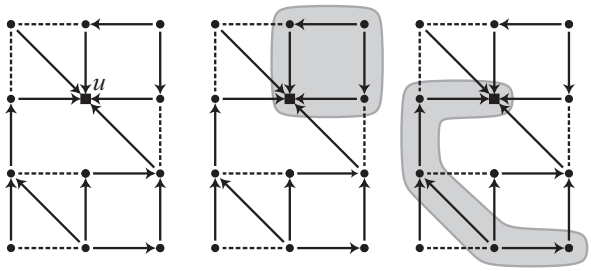
4.1 Connectivity based on Precedence Relationship

Evidently, in order to end up with connected parts, a node v can only be assigned to a distinct center u, if at least one of its neighbors is also assigned to u. However, it is important to note that this does not suffice. Consider two adjacent nodes being assigned to the same center: Both nodes will mutually satisfy their requirements without ensuring the connectivity to others, i.e., the problem with the neighbor relationship is that it contains cycles. To cope with this, we introduce the stricter, acyclic *precedence relationship*.

Given a graph $G(V, E)$ with edge lengths $\alpha : E \rightarrow \mathbb{R}^+$, a center $u \in V$ and a node $v \in V\,, u \neq v$, we define the set of predecessors of v with respect to center u as

$$\mathrm{Pred}_u(v) := \{w \in V \mid \mathrm{D}(u\,, w) < \mathrm{D}(u\,, v) \wedge \{v, w\} \in E\}\,, \quad (5)$$

with $\mathrm{D}(i\,, j)$ being the length of the shortest path in G from i to j using edge lengths α. Different possibilities exist for defining the edge lengths α. The definition which is applied here is based on the minimal size of a potential aggregate containing u and v. This definition is discussed in our earlier paper. An example for the precedence relationship with the setting of equal edge length is illustrated in Figure 3(a). Arcs are drawn from each node $v \in V$ to its predecessors $\mathrm{Pred}_u(v)$. The resulting directed graph is acyclic and the center u is the only terminal. For some edges of the adjacency graph, both incident nodes have the same distance to the center. These edges are displayed as dashed lines. However, when using non-uniform edge lengths, these cases are rare exceptions.



(a) Precedence relationship Pred_u.

(b) A feasible aggregate according to Pred_u.

(c) A connected aggregate that is not feasible according to Pred_u.

Figure 3: Precedence relationship with respect to center u (displayed as square) and feasibility of aggregates in the presented MIP.

The definition of the precedence relationship can be used to define a simple requirement that ensures connectivity of aggregates: A node may only be assigned to a distinct center if at least one of its predecessors with respect to this center is also assigned to it. The constraint clearly forbids disconnected aggregates since the center can always be reached from an assigned node via predecessors, i.e., without leaving the aggregate. Figure 3(b) shows an example which satisfies the requirement. However, by defining this requirement several connected aggregates will be excluded. An example for this is displayed in Figure 3(c). The aggregate does not contain any predecessor of the node located in the bottom right corner. Zoltners and Sinha legitimate the restriction of their model with their preference for compact sales districts: The non-feasible connected parts likely are non-compact. So, they can probably be excluded without loosing good solutions. As we also aim compactness, their model is a reasonable approach. However, since the optimal solution might be missed, we refer to it as heuristic. Comparisons with results that were attained with a model allowing for connectivity in a general sense, i.e., with our flow model, have shown that this heuristic only marginally affects the result. The attained results for the objective function were at most 5% worse than the optimum. The processing time, however was greatly reduced. As mentioned in the introduction, our tests were made without consideration of compactness. It is likely that the results are even closer to the optimum when defining compactness as additional objective.

4.2 MIP Formulation

In this section, we present a MIP hat models the requirements and objectives of the problems from Sections 2.1 and 3.2. We first introduce the formulation as a whole and then explain it in detail.

$$x_{uv} \in \{0,1\}\,, \quad \text{with } x_{uv} = 1 \text{ if node } v \in V \text{ belongs to center } u \in V\,.$$

$$y_{ue} \in [0,1]\,, \quad \text{with } y_{ue} = 0 \text{ if at least one incident node of } e \in E \text{ does not belong to center } u \in V\,.$$

Minimize

$$\begin{aligned} & s \cdot \sum_{u\in V}\sum_{v\in V} w(v)\cdot x_{uv}\cdot d(\gamma(v),\gamma(u)) \\ +(1-s)\cdot s' \cdot & \sum_{u\in V}\sum_{v\in V} w(v)\cdot x_{uv}\cdot \delta(v,u) \\ -(1-s)\cdot(1-s')\cdot & \sum_{u\in V}\sum_{e\in E} 2\cdot\lambda(e)\cdot y_{ue} \end{aligned} \tag{6}$$

subject to

$$\sum_{u\in V} x_{uv} = 1 \qquad \forall v \in V\,, \tag{7}$$

$$\sum_{v\in V} w(v)\cdot x_{uv} \geq \theta(\gamma(u))\cdot x_{uu} \qquad \forall u \in V\,, \tag{8}$$

$$\sum_{w\in \mathrm{Pred}_u(v)} x_{uw} \geq x_{uv} \qquad \forall u, v \in V : u \neq v\,, \tag{9}$$

$$\left.\begin{array}{rcl} y_{ue} & \leq & x_{uv} \\ y_{ue} & \leq & x_{uw} \end{array}\right\} \forall u \in V, \quad e = \{v,w\} \in E\,. \tag{10}$$

The binary variables x_{uv} define the solution of the problem: All nodes u with $x_{uu} = 1$ constitute the set of centers that define the color and geometrical centers of aggregates. To assign a node v to a center u, x_{uv} needs to be set to 1. The cost for the color change that is charged for this assignment is defined by the first term in Equation 6. The second term defines the cost for compactness according to Equation 1. The third term defines a benefit for each edge $e = \{v, w\}$ that is totally contained in one aggregate, i.e., there is a center $u \in V$ with $x_{uv} = 1$ and $x_{uw} = 1$. Auxiliary variables y_{ue} are defined to express this case. Giving a benefit for interior edges has the same result as charging a cost for the perimeter, as the objective was defined in Equation 2. The factor 2 is needed, as each edge belongs to the boundaries of two areas.

We now describe our set of constraints. Constraint 7 expresses that each node must be assigned to exactly one center. Constraint 8 does not have any effect for $x_{uu} = 0$, i.e., if u is not selected as center. For $x_{uu} = 1$ it ensures that the aggregate with center u is weight feasible, i.e., the threshold for the target scale is satisfied. Constraint 9 ensures connectivity according to the precedence relationship, as defined in Section 4.1: Node v can only be assigned to center u if there is also a predecessor w which is assigned to u. Finally, Constraint 10 is used to couple the variables x_{uv} and those of type y_{ue}: If one of the incident nodes of e, i.e., v or w, is not assigned to center u, then y_{ue} is forced to be 0 and no benefit will be given. Otherwise, the constraint defines that $y_{ue} \leq 1$. Since a benefit proportional to y_{ue} is given, y_{ue} will always take the value of its upper bound. Because of this y_{ue} will be 1 for edges included in aggregates. Thus we do not have to make y_{ue} explicitly a 0-1 variable, which usually speeds up MIP solvers.

Our MIP models both, the problem from Section 2.1 and the modified problem with predefined centers. In the second case, it is possible to simply define $x_{uu} = 1$ for all u in C. Additional variables can be fixed after this. To define a MIP without the restricting precedence relationship, one can simply replace Constraint 9 by formulations that have been presented by Williams (2002) and Shirabe (2005). These, however, require additional auxiliary variables. In our previous paper, we presented two additional heuristics that can be applied to speed up the processing. The first is to set $x_{uu} = 0$ for nodes u with very small weights, i.e., to exclude them from the set of potential centers. The second heuristic is to set $x_{uv} = 0$, if the distance between u and v is large. These heuristics have been formally defined and discussed in detail. We present results of this method and the addition of criteria for compactness in the next section.

4.3 Results

We used the presented formulation to express the problem as a MIP and solved it by application of standard branch-and-cut methods. The performance was similar to the MIP without the application of the compactness objective, which has extensively

been tested in our previous paper. With the presented heuristics it is possible to solve instances with 400 areas in modest time, i.e., less than one hour. Figure 4 shows the same sample as Figure 1, but this time the proposed measure for compactness was applied in combination with the objective for minimum class changes. The resulting aggregates are clearly more compact. As in the first example, the red settlement was saved by sacrificing smaller neighbors, but instead of building a narrow bridge to smaller areas of the same color, a neighbor on the right side was included, leading to a simpler shape. However, the resulting map certainly does not constitute a finished product. For example, one would need to apply a line simplification algorithm to remove further details. Nevertheless, it is perceived that the formulated optimization problem sufficiently models the aims of aggregation.

4.4 Processing Large Data Sets

In the presented form, the method is still not suitable for cartographic production, as the expense of time is too high. To process large datasets we have developed a heuristic approach (Haunert (2007)). The idea is to predefine each node $v \in V$ with $w(v) \geq \theta(\gamma(v))$ as center, i.e., those areas in the original scale that are sufficiently large for the target scale. Let G' be the sub graph of G that is induced by all other nodes, then the aggregation problem with the compactness measures from Section 2.2 can be solved independently for each connected component of G'. This fact allows to decompose the problem into smaller instances. However, for our data set, the resulting instances are still too large to be processed. We have solved this problem by definition of intermediate size thresholds, such that the number of predefined centers increases until the problem instances are manageable, i.e., do not contain more nodes than a user-specified number k. Via these intermediate scales, the target scale can be reached in several steps. We have shown that our method generalizes the existing iterative method of van Oosterom (1995), i.e., for $k := 1$ both methods are the same. However, for a complete map sheet of a topographic map, our method with $k := 200$ resulted in 20% less class change, 2% less cost for non-compact shapes and 8% less total cost.

5 CONCLUSION

We have proposed a new method for the aggregation of areas in a planar subdivision that takes compactness and class similarity into account and enables the application of mixed-integer programming. With this restriction, we could model compactness only by adding requirements that avoid the creation of too large aggregates. To define these requirements, we developed a two-step approach. First we apply a coarse measure of compactness for the definition of centers and second we create a high-quality map by applying a more sophisticated measure. The obtained results showed that this approach satisfies the aims of aggregation in map generalization. Due to the NP-hardness of the problem, heuristics needed to be introduced to solve instances of interesting size. We gave an outline of a heuristic approach that decomposes the problem into manageable instances.

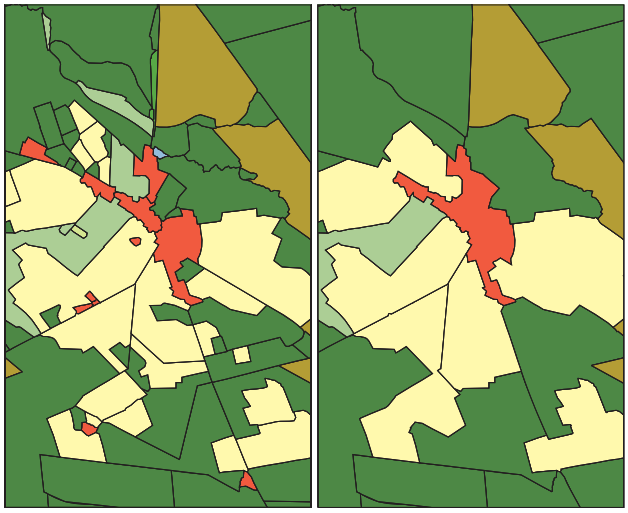


Figure 4: An example from the original map at scale 1:50.000 (left) and a result of the proposed method (right). The sample contained 200 areas in the original map. These were grouped into 32 aggregates. The processing took 9s.

References

Bader, M. and Weibel, R., 1997. Detecting and resolving size and proximity conflicts in the generalization of polygonal maps. In: Proc. 18th International Cartographic Conference, Stockholm, Sweden, pp. 1525–1532.

Bergey, P. K., Ragsdale, C. T. and Hoskote, M., 2003. A simulated annealing genetic algorithm for the electrical power districting problem. Annals of Operations Research 121, pp. 33–55.

Haunert, J.-H., 2007. Efficient area aggregation by combination of different techniques. In: Proc. of 10th ICA Workshop on Generalisation and Multiple Representation, 2–3 August 2007, Moscow, Russia.

Haunert, J.-H. and Wolff, A., 2006. Generalization of land cover maps by mixed integer programming. In: Proc. 14th Int. ACM Sympos. Advances in Geographic Information Systems (ACM-GIS '06), ACM Press, New York, pp. 75–82.

Jaakkola, O., 1997. Quality and Automatic Generalization of Land Cover Data. PhD thesis, Department of Geography, University of Helsinki.

Maceachren, A. M., 1985. Compactness of geographic shape: Comparison and evaluation of measures. Geografiska Annaler. Series B, Human Geography 67(1), pp. 53–67.

Podrenek, M., 2002. Aufbau des DLM50 aus dem Basis-DLM und Ableitung der DTK50 – Lösungsansatz in Niedersachsen. Karthographische Schriften Band 6, Kartographie als Baustein moderner Kommunikation, pp. 126–130.

Shirabe, T., 2005. A model of contiguity for spatial unit allocation. Geographical Analysis 37, pp. 2–16.

Timpf, S., 1998. Hierarchical structures in map series. PhD thesis, Technical University Vienna, Austria.

van Oosterom, P., 1995. The gap-tree, an approach to 'on-the-fly' map generalization of an area partitioning. In: J.-C. Müller, J.-P. Lagrange and R. Weibel (eds), GIS and Generalization - Methodology and Practice, GISDATA, Taylor & Francis, London.

van Smaalen, J. W. N., 2003. Automated Aggregation of Geographic Objects. PhD thesis, Wageningen University, The Netherlands.

Williams, J. C., 2002. A zero-one programming model for contiguous land acquisition. Geographical Analysis 34(4), pp. 330–349.

Zoltners, A. A. and Sinha, P., 1983. Sales territory alignment: A review and model. Management Science 29(11), pp. 1237–1256.

REPRESENTATION AND ANALYSIS OF TOPOLOGY IN MULTI-REPRESENTATION DATABASES

M. Breunig *, A. Thomsen , B. Broscheit, E. Butwilowski, U. Sander

IGF, University of Osnabrück, Kolpingstr. 7, 49069 Osnabrück, Germany –
{martin.breunig, andreas.thomsen, edgar.butwilowski, uwe.sander, bjoern.broscheit}@uos.de

KEY WORDS: Topology, multi-scale representation, geodatabase, MRDB, data modelling, LOD, abstraction.

ABSTRACT:

Multi-scale representation and analysis of topology is playing a growing role in Photogrammetric Image Analysis. However, the standardisation of multi-scale topological data models is still at its beginning. Furthermore, the multi-representation of geo-objects poses new challenges, resulting in the development of Multi-Representation Databases. In this article the realisation of a general model based on oriented hierarchical d-Generalised Maps to represent and analyse topology in MRDB is described in detail. The model can be used as a data integration platform for 2D, 3D, and 4D topology. Examples of elementary and complex topological operations for multiple representations are presented. An application example with 2D cartographic datasets from Hannover University shows the feasibility of the new approach. Finally, an outlook on future research is given.

1. INTRODUCTION

Multi-scale representation and analysis of topology is important for GIS and will also play a growing role in Photogrammetric Image Analysis. However, to our knowledge, the database representation of topology in different levels of detail (*LOD*) has not been investigated in detail.

Multi-representation of topology poses new challenges resulting in the development of *Multi Representation Databases (MRDB),* that manage discretely and continuously changing LOD. Although generalisation operations affect the topology of a spatial model, research about the representation and management of topology in MRDB is still at its beginning. In (Thomsen and Breunig, 2007), we propose some elementary and complex topological operations for a topological database toolbox based on oriented Generalized Maps (G-Maps).

In this paper, we investigate how oriented hierarchical G-Maps can be used to handle the topology of a digital spatial model at different levels of detail in a MRDB based on the object-relational model, providing a generic, application-independent approach. The method is general enough to support 2- and 3-dimensional models, as well as 2D-manifolds in 3D space.

2. RELATED WORK

Approaches for representing topology in 3D modelling have been examined by different authors (Mäntylä, 1988). For the representation of 3D-objects in GIS by 2D-manifolds, (Gröger and Plümer, 2005) propose “2.8-D maps”, that avoid the topological complexity of true 3D-Models. Cellular complexes, and in particular cellular partitions of d-dimensional manifolds (d-CPM) have been described to represent the topology of an extensive class of spatial objects by (Mallet, 2002). The topology of d-CPM can be represented by d-dimensional Cell-Tuple Structures (Brisson, 1993), respectively d-dimensional Generalized Maps (d-G-Maps) (Lienhardt, 1994). (Lévy, 1999) has shown that 3D-G-Maps have comparable space and time behaviour as the well-known DCEL and radial edge structures, but can be used for a much wider range of applications, allowing for a more concise code. Lévy also introduces hierarchical G-Maps (HG-Maps) for the representation of nested structures. 3-G-Maps are also applied e.g. in the geoscientific 3D-Modelling software GOCAD (Mallet, 1992, 2005). (Fradin et al., 2002) use 3-G-Maps to model and visualize architectural complexes in a hierarchy of multi-partitions. Finally, an interactive graphical G-Map-based 3D-modeller MOKA has been made available by the group of graphical informatics at Poitiers University (MOKA, 2006). (Meine & Köthe (2005)) have introduced the GeoMap, a related but less general concept based on half-edges, that integrates planar topology and geometry for raster image segmentation.

3. MULTI-SCALE REPRESENTATION AND ANALYSIS OF TOPOLOGY

Aggregation, simplification, elimination, displacement and typification are well-known generalisation transformations. Aggregation and elimination directly affect the topology of a map. Simplification may affect the interior structure of an object, whereas displacement may be employed in order to maintain topological consistency under a geometrical generalisation operation - e.g. if smoothing a river bend would leave a building on the wrong side. In a first step, we concentrate on the aggregation of contiguous cells by the

* Corresponding author.

application of sequences of Euler transformations, being aware that this approach covers only a selection of generalisation operations. In a second step, we will try to model the aggregation of disjoint cells using transformations of classifications/colourings of cellular complexes. Whereas the choice of the generalisation method is taken by the geoscientist, supported by specialised software (cf. Haunert & Sester, 2005), we focus on the representation of the given transformations and of the resulting relationships between LOD in the MRDB. Relationships between cells at different levels can be defined by explicit links, or by indicating the sequence of elementary operations that transform a cellular complex at scale *A* into a cellular complex at scale *B*. It is the task of the database software, to keep track of the incurred changes, and if possible to support *transitions* with *commit* and *rollback* operations.

3.1 Representation

3.1.1 Hierarchies of maps: For the representation of multi-scale topology, Lévy (1999) proposes Hierarchical G-Maps (HG-Maps): The Aggregation of neighbouring cells results in a classification of cells on the more detailed level *A*, each class being associated with one cell on the less detailed level *B*. It can be represented by an n:1-mapping from one level *A* to level *B*. As cells are merged, and interior boundaries disappear, the number of cell-tuples is reduced. The cell-tuples on level *B* can be associated with a selection of cell-tuples on the lower level *A*, or be identified with a subset of the latter. If the geometry of the remaining cell boundaries is not changed after the aggregation step, higher level cell-tuples may delegate their geometrical embedding (co-ordinates, lengths, angles etc.) to their counterparts on the lower level (**fig. 1**), so that a higher-level edge is geometrically represented by a sequence of lower-level arcs and vertices. Otherwise, links with a new higher-level geometrical embedding must be established.

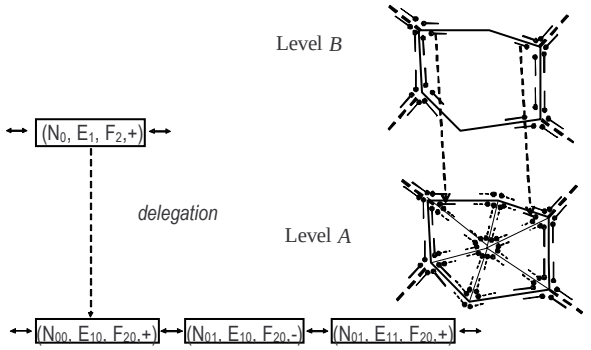


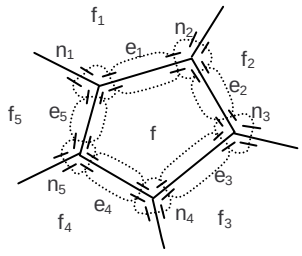
Figure 1. Generalisation by aggregation in a hierarchical 2-G-Map. Cell-tuples (darts) are symbolised by small pins.

3.1.2 Progressive Variation of LOD: Due to the necessity of keeping all levels of detail consistent with each other, any changes in an MRDB are first introduced at the greatest scale, and then propagated upwards using appropriate generalisation methods (Haunert & Sester, 2005). Carrying this "dynamic" approach a step further, we investigate the applicability of progressive meshes. The progressive triangulation method (Hoppe, 1996) uses two localised elementary operations, namely the "edge collapse" and its inverse, the "vertex split", to coarsen or to refine a triangle network incrementally in both directions, by successively applying a sequence of stored "delta" operations. This method is well suited for *progressive transmission*, as it can reduce the amount of data exchanged between a geo-database server and a local client (cf. Shumilov et al., 2002).

Generalized maps are abstract simplicial complexes, but Hoppe's method cannot be adapted: Although a d-cell-tuple is an abstract d-simplex, its d+1 components belong each to a different class defined by dimension, and therefore cannot be merged, like in an "edge collapse" operation on a triangle network. An analogous argument holds for the inverse "vertex split" operation. Instead, we investigate the possibility to use combinations of the Euler elementary split and merge operations on cells to model the transformation of topology induced by generalisation. Different from Hoppe's method, the progressive mesh transformation is controlled by the external generalisation method, and not by a given optimisation criterion. Note that the merge operations are applicable only in certain configurations and hence require supervision.

3.2 Analysis

The relational representation of d-G-Maps has been made persistent using an Object-Relational Database Management System (*ORDBMS*). Implementing a topological component for multi-representation databases (Thomsen and Breunig, 2007) we used 2D- and 3D-G-Maps with the ORDBMS PostgreSQL (PostgreSQL.org, 2006) in combination with the open source PostGIS (PostGIS.org, 2006).



(n1,**e1**, f, +)
(n1,e1, **f1**,-)
(n1,**e6**, f1,+)
(n1,e6, **f5**,-)
(n1,**e5**, f5,+)
(n1,e5, **f**, -)

(n1,e1, **f**, +)
(**n2**,e1, f, -)
(n2,e1, **f1**,+)
(**n1**,**e1**, f1,-)

(n1,**e1**,f,+)
(**n2**,e1,f,-)
(n2,**e2**,f,+)
(**n3**,e2,f,-)
(n3,**e3**,f,+)
(**n4**,e3,f,-)
(n4,**e4**,f,+)
(**n5**,e4,f,-)
(n5,**e5**,f,+)
(**n1**,e5,f,-)

Figure 2. 2-G-Map with darts and involutions, and cell-tuple representation of the orbits around node n_1, edge e_1 and face f.

3.2.1 Oriented Generalized Maps: An *oriented Generalized Map of dimension d (d-G-Map)* (Lienhardt, 1994) represents a cellular complex that is used as a discrete model of the topology of an orientable manifold of dimension d. It consists of a set of *darts*, d+1 transformations of the set of darts, α_i, $i = 0 \ldots, d$, that are *involutions* verifying $\alpha_i(\alpha_i(x)) = x$ (**fig. 2**). The involutions must further verify the condition that $\alpha_i(\alpha_{i+2+k}())$ is an involution for $k >= 0$. Subsets of darts that can be reached from a starting dart x_0 by any combination of involutions $\alpha_i \ldots \alpha_i$ are called *orbits*. We note them $orbit^d(i,\ldots,j, x_0)$ or $orbit^d_{i\ldots j}(x_0)$, where d is the dimension of the G-Map, the indexes $i, \ldots, j$ are a subset of $\{0,\ldots,d\}$, and x_0 is the starting dart. Certain orbits, namely those of the form $orbit^d(\ldots\sim k\ldots)$, that use all involutions except α_k, determine the k-dimensional cells of the cellular complex, i.e. nodes, edges, faces, and solids for dimension $d=3$ (**fig. 2**). In a d-G-Map, $orbit_{0\ldots d}(x_0)$ returns the connected component containing x_0. Orbits $orbit_i()$, and by Lienhardt's condition, orbits of the form $orbit_{i,i+2+k}()$ have a fixed length. Other orbits can be

implemented by single or nested programming loops, a small number of orbits however, are more complicated – they can be implemented recursively, returning the subset of cell-tuples as a collection of connected sequences possibly interrupted by discontinuities. For some topological operations, especially the solid split operation, we need *continuous loops* that generally are defined by the user, and not produced by an orbit. Different from linear iterators, orbits and loops are examples of *circulators* (Fabri et al., 1998) that can begin at any object in the circular sequence, and advance until the starting point is again encountered.

3.2.2 Realisation by means of an ORDBMS: Whereas G-Maps can be implemented focusing on the involution transitions represented e.g. as references between anonymous darts, we prefer the relational realisation to focus on the darts, which are represented by signed *d-cell-tuples* $(c_0, \ldots, c_d, +/-, \ldots)$ (**fig. 2**), cf. (Brisson, 1993), collected in the tables of an ORDBMS. The c_i are identifiers of cells of dimension *i*, i.e. nodes, edges, faces, solids. The identifiers of the *neighbour cells* c_inv_j, $0 \le j \le d$, are also attached to the cell-tuples. The involutions α_j are implemented as "switch" operations that transform the *cell-tuple key* $(c_0, \ldots, c_j, \ldots, c_d)$ into $(c_0, \ldots, c\_inv_j, \ldots, c_d)$, exchanging c_j and c_inv_j and then retrieve the corresponding cell-tuple record from the database.

Orbits and loops. By definition, an orbit $orbit_{i..j..}(ct_0)$ consists of the subset of cell-tuples that can be reached from ct_0 using any combination of $\alpha_{i,\ldots},\alpha_{j,\ldots}$ The components of dimension *k* where *k* is not contained in the set of indices i,…,j remain fixed, e.g. if $ct_0=(n,e,f,s)$, then $orbit_{012}(ct_0)$ leaves solid s fixed, and returns all cell-tuples of the form (*,*,*,s).
The implementation of the darts of a G-Map as cell-tuples in a relational DBMS is straightforward, the involutions can be implemented using queries or joins, supported by foreign keys and indexes, and iterators can be realised as database cursors, but a normal relational DBMS does not provide the equivalent of circulators, i.e. closed loops of undetermined, albeit finite, length. The representation of orbits therefore needs additional code controlling repeated database queries. As such implementations are not very efficient, we try to replace orbits by subset queries, wherever the circular arrangement is dispensable.
The trivial orbits of the form $orbit_i()$ can be treated like the corresponding involutions, and by Lienhardt's condition, orbits of the form $orbit_{i,i+2+k}()$ have a constant length of four and can be modelled by a limited number of queries or join operations. Whereas RDBMS do not support cyclic cursors that would correspond to circulators, result sets of queries can be ordered, e.g. the query:

```
SELECT * FROM celltuples
WHERE <condition>
ORDER BY face, edge, sign;
```

returns the retrieved cell-tuples ordered according to faces, in ordered pairs corresponding to the edges of the face boundary, although not in a cyclic arrangement. In some application cases, this may be sufficient. Whenever the orbit arrangement must be reproduced exactly, however, a true orbit can be implemented by stepwise executing the involution operations:

```
Start with
   node n0, edge e0, face f0, sign sg0, n_inv0, e_inv0, f_inv0;
i=0;
repeat {
  ++i;
  update j; /* j: selector of the next involution αj */
  case j {
     0: SELECT node as ni, edge as ei, face as fi, n_invi ...
          FROM celltuples
          WHERE ni = n_invi-1 AND ei= ei-1 AND fi= fi-1
     1: SELECT node as ni, edge as ei, face as fi, e_invi...
          FROM celltuples
          WHERE ni = ni-1 AND ei= e_invi-1 AND fi= fi-1
     2: SELECT node as ni, edge as ei, face as fi, f_invi ...
          FROM celltuples
          WHERE ni = ni-1 AND ei= ei-1 AND fi= f_invi-1  }
} until ni= n0 and ei=e0 and fi= f0;
```

We use a *selector* variable to determine the next transition step. This procedure can be modified to implement any closed loops in the G-Map, by attaching to the cell-tuples a selector variable the current value of which controls the choice of the next α_i transition.

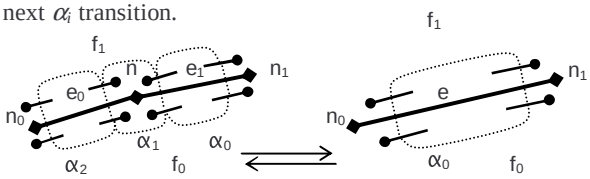


Figure 3. Merging two edges e_0, e_1 that separate faces f_0, f_1, by deletion of a node *n*.

3.2.3 Realisation of simple generalisation operations: At the present stage, we concentrate on basic split and merge operations, which serve to build more complex aggregation operations in 2D and 3D.

Merging two edges. The merging of two edges, i.e. 1-cells, by removal of an intermediate node is straightforward: consider a sequence $n_1\ e_1\ n\ e_2\ n_2$ consisting of nodes n_i and edges e_j. We wish to replace $e_1\ n\ e_2$ by a new edge *e*, hence we delete all cell-tuples $(n,\ldots)$ having *n* as node component, and in all cell-tuples containing e_1 or e_2 as edge component, we replace e_1 and e_2 by *e*. Then, we update all cell-tuples related to $(n_1,e_1,...)$ or $(n_2,e_2,...)$ by α_1 involutions. If node *n* and edges e_1, e_2 are not used elsewhere, we delete them as well (**fig. 3**). A necessary condition for the edge merge operation to be applicable is that there are only two edges incident with node *n*. This can be checked counting the length of an $orbit_{12}((n,e_1,f_1))$, or by counting the number of darts returned by a corresponding SQL query. In the following, we tacitly assume that whenever a sequence of edges without branches that separates two faces is to be submitted to a merge operation, it is first transformed into a single edge by a succession of edge merges.

Merging two faces. Let us consider the following situation:
Two faces f_1 and f_2 are separated by one edge *e* between nodes n_1 and n_2. By removing *e*, f_1 and f_2 are merged into one face *f* (**fig. 4**). Again, we first remove all cell-tuples containing edge *e*. Then in all cell-tuples containing f_1 or f_2, we replace these by *f*. Next, we replace f_1, f_2 by *f* in all cell-tuples relating f_1, f_2 by

α_2 involutions, and "repair" the involutions at nodes *n1* and n_2 replacing sequences of the form
$(n_i,e_x,f)\ \alpha_1\ (n_i,e,f)\ \alpha_2\ (n_i,e,f)\ \alpha_1\ (n_i,e_y,f)$
by $(n_i,e_x,f)\ \alpha_1\ (n_i,e_y,f)$ (**fig. 4**).

The face merge operation can be applied if none of the faces separated by e belongs to the outside ("universe") of the G-Map. Otherwise, it has to be verified that the operation doesn't produce a "bridge" configuration – a single edge incident on both sides to the outside, linking two connected parts of the G-Map. Though bridge configurations could be modelled in 2D using the orientation of the cell-tuples, we exclude them because they do not fit well with our definition of an involution as exchange of two distinct k-cells.

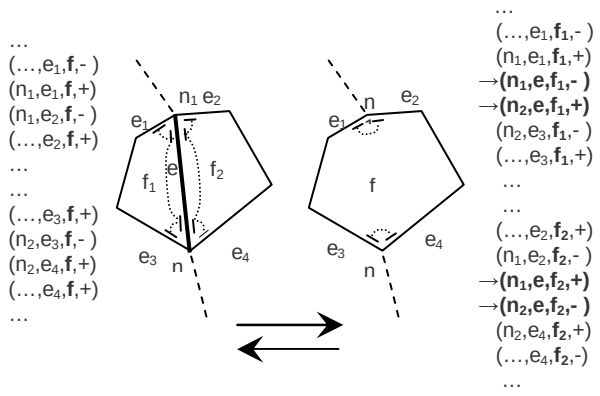


Figure 4. Merging two faces f_1, f_2 by removing edge *e*.

Splitting a solid. The inverse operations are decomposed analogously, exchanging the roles of insert and delete operations. Let us discuss the splitting of a solid *s* by the insertion of a separating face *f*, into two solids s_1, s_2 (**fig. 5**).
Besides set operations, splitting a 3d-cell requires the use of an $orbit_{012}()$. We start with the definition of a closed connected sequence of nodes and edges that define the contact – the seam – between the circumference of the face and the meshing of the inner surface of the solid. This can be done using a sequence of cell-tuples connected by α_0- α_1-, and α_2-involutions forming a closed loop. This seam location has to be defined by the user or by a client program, and the number of its nodes and edges must coincide with that of the boundary of *f*. The operation then consists of the following steps:
First, insert face *f*, and solids s_1 and s_2. Next, for each pair of cell-tuples situated on either side of the seam location, replace
$(n_i,e_j,f_k,s,+)\ \alpha_2\ (n_i,e_j,f,s,-)$
by a sequence
$(n_i,e_j,f_k,s_1,+)\ \alpha_2\ (n_i,e_j,f,s_1,-)\ \alpha_3\ (n_i,e_j,f,s_2,+)\ \alpha_2\ (n_i,ej,f_l,s_2,-)$.
Finally, starting from a cell-tuple $ct_0(n_i,e_j,f,s_1,+)$, use an $orbit_{012}(ct_0)$ to replace *s* by s_1 on every cell-tuple encountered, and all cell-tuples related by α_3-involutions. By the use of an $orbit_{012}()$, we assure that all cell-tuples $ct(.,.,.,s)$ selected for update are situated on the boundary of solid *s* and on one side of face *f*, independent of the value of the solid component. Next we repeat the same procedure starting with $(n_i,e_j,f,s_2,-)$, replacing *s* by s_2 on the other side of face *f*.
Obviously, such sequences can be implemented using the insert, delete and update operations of a relational database within a transaction. For the solid merge to be applicable, we have to check that there is no other contact between s_1 and s_2, and that none of the solids s_1 and s_2 is part of the outside of the G-Map. Otherwise, we have to check that no 3D-bridge configurations result, i.e. a single face incident on both sides with the same solid, or with the outside. The latter configurations can be avoided by first ensuring that none of the other neighbouring cells are part of the outside.

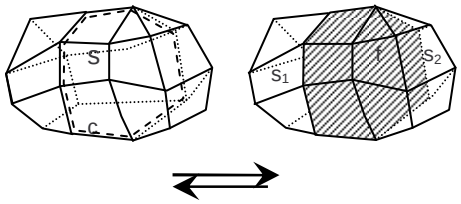


Figure 5. splitting a 3D solid *s* by the insertion of a 2D face *f*. The location of the seam is defined by the loop *c*.

A non-Euler operation. Geo-data from external sources cannot be expected to carry an explicit representation for their topology ready for representation as a G-Map. Rather, one of the first steps of the import of geo-data consists in extracting topological relationships that are implicit within the data. As an example, consider a land use map encoded as a *shapefile*: each parcel is defined by one or several polygons, that are not linked to each other, so that topologically each parcel is an island disconnected from the rest. In this particular case the vertex co-ordinates, however, of neighbouring polygons match exactly, so that it is possible to reconstruct the neighbourhood relationships between parcel boundaries by matching vertex co-ordinates. In the general case, we have to modify the geometrical matching criterion such as to accommodate small numerical fluctuations, e.g. resulting from digitisation.
We introduce the newly gained information into the G-Map by *sewing* corresponding cell-tuples, i.e. by establishing the α_i involution links. This operation starts with the merging of a pair of nodes from two neighbouring polygons. It is not an Euler operation, as the number of nodes is reduced by one, whereas edges and faces remain unchanged. The resulting configuration of two polygons having one point in common is theoretically admissible, but it poses practical problems, therefore we require it to be immediately followed by the merging of a second pair of nodes, and of the two edges joining the nodes to be merged. This second sewing operation, and any others following without interruption on the same boundaries, do not affect the Euler-Poincaré characteristic.

Integrity constraints. Whereas basic split operations do not affect the consistency of the G-Map, merging of cells may lead to singular and inconsistent configurations. As an example, consider a map of land use, comprising a number of parcels of identical land use *A* that surround one or more parcels of land use *B* (**fig. 6**). A complex merge operation that aggregates all cells of type *A* eventually results in a ring, which is multiply connected and hence is not consistent with the definition of a cell in a cellular complex. Another consequence is that the cells of type *B* could never be reached by an orbit starting from the outer boundary of a type *A* cell. We must therefore detect these configurations and stop the merging process such as to conserve two cells of type *A* separated by two bridging edges (**fig. 6**). The occurrence of a bridge configuration during a face merge operation is not detected by a change in the Euler characteristic of the G-Map subset defined by the class *A*.

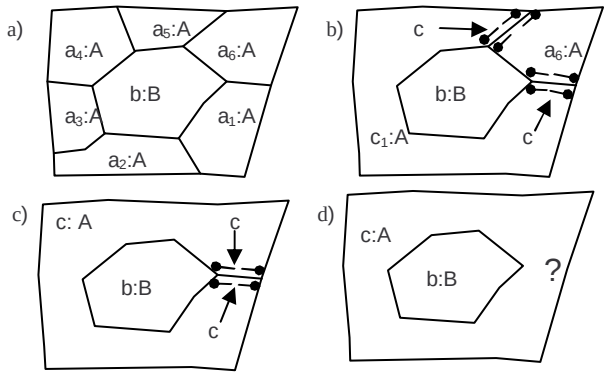


Figure 6. (a) A face *b* of class *B* is completely surrounded by faces a_i of a different class *A*. (b) Stepwise merging all cells of class *A* results in a bridge configuration (c) and finally in a ring-shaped cell (d).

It can be detected before the merge operation by verifying that the boundary between the cells to be merged is simply connected, or after the operation by searching for α_2-transitions that link two cell-tuples having the same face:

```
SELECT count(*)
    FROM celltuples
    WHERE <condition>  AND face_id= face_inv;
```

Any result different from 0 indicates an error. If the bridge configuration is detected after a merge operation, it can be corrected either using DBMS transition rollback, or by performing the inverse edge split operation.

The transition to a ring configuration (**fig. 6d**) can be detected by a change of the Euler characteristic *N-E+F*, where *N*, *E*, *F* are the numbers of nodes, edges and faces respectively. In fact, deleting the last bridging edge doesn't change *N* or *F*, but reduces *E* by one. Though irregular configurations can be avoided during a merge after classification, it is an inconvenience that in some cases contiguous cells of the same class nevertheless must be kept separate. To handle the connected components of a partition, (Fradin et al. 2002) use boolean flags to distinguish those α_i transitions that join cells of the same class, from α_i transitions that link different classes. Moreover, they implement multiple partitions using an array of flag bits associated with the α_i transitions, supporting several different classifications on the cells of the same basic G-Map.

Since their G-Map implementation is based on α_i transitions of darts, rather than on explicitly modelled cell-tuples, we cannot use this approach without modification.

It is possible, however, to adapt this feature to our cell-tuple-based representation by associating the flag array with the cell-tuple variables node_inv, edge_inv, etc. that define the α_i transitions, or simply use queries like the following one:

```
UPDATE celltuples ct1
SET face_class_flag= TRUE
WHERE  EXISTS(
  SELECT * FROM celltuples ct2
  WHERE ct2.face_id= ct1.face_inv
  AND ct2.face.class =  ct1.face.class );
```

Then, iterators can be derived from the α_i orbits that yield all cell-tuples associated with a given class *A*, its associated flag values indicating a position at a class boundary or in the interior. Using a nested query or a join, the corresponding relational query can directly return all cell-tuples belonging to a given class *A*, together with the associated flag for further processing (e.g. for skipping interior cell-tuples).

3.2.4 Realisation of complex generalisation operations: A Multi-Resolution Database (*MRDB*) of land use (Haunert & Sester, 2005) consists of a stack of maps at different scale and LOD, and a hierarchy of partitions of the map of highest LOD. In this example, the maps are encoded as *shapefiles*, and the aggregation hierarchy is represented by n : 1-relations between successive LODs that are stored in a table.

To establish the topological properties of the MRDB, at each LOD first the isolated polygons are sewed to form a partition of part of the map plane. Then, the classification induced by the lower LOD *B* and the aggregation table is used to "paint" the faces of the map at higher LOD *A*. The resulting partition of *A* can be used to introduce flags distinguishing inter-class α_i transitions from intra-class transitions.

In a next step, in order to reduce the amount of data and to establish a more detailed relationship between successive LODs, neighbouring faces of *A* that belong to the same class are merged wherever this is possible without violating the integrity of the G-Map. The result of this operation is an aggregated G-Map *A'* of *A* that, with a number of exceptions, corresponds to the G-Map *B*. At this stage, the user may intervene and modify the aggregation table in order to reduce the number of problematic configurations.

Let us now extend the hierarchical relationship between faces of *A* and *B*, and to establish relationships between nodes, edges and faces of *A'* and *B* respectively. As generalisation may have involved displacements, a simple comparison of co-ordinates is not a sufficient matching criterion. Instead we use the G-Map to find corresponding nodes in *A'* and *B* by comparing the configuration of their neighbourhoods. E.g. if a connected set of class *C* in the G-Map of *A'* corresponds to a face *f* in *B*, we can search for nodes on the boundaries of *C* and *f* that have similar neighbours. As *A'* has been developed from *A* by aggregation of cells, the nodes, and cell-tuples of *A'* correspond to a subset of those of *A*. Thus a finer correspondence between the topologies of *A* and *B* is established, than the initial aggregation hierarchy.

4. AN APPLICATION EXAMPLE

The Hannover Institute of Cartography (IKG) is investigating methods that generalise land use maps by an automatic aggregation of parcels using thematic and/or geometric criteria (Haunert & Sester, 2005). The resulting hierarchies of maps at different LOD are stored in a MRDB (Anders & Bobrich, 2004). The n:1 relationships between polygonal faces between different scales are represented in tabular form (**fig. 7**).

From a set of separate maps at different scales imported into PostGIS/PostgreSQL, we derive corresponding G-Maps. Topological consistency is checked and the Euler characteristic and some basic statistics are established. The n:1 relationship between maps at different LOD induces a classification of the cells of greater scale. Using the elementary merge operations described above, groups of cells of the same class are aggregated either until a 1:1 correspondence is established, or

until inconsistent configurations are detected. Thereafter, unnecessary nodes on the boundaries of the aggregated cells are eliminated while edges are merged. If no premature stop has been encountered, the 1:1 relationship between faces and aggregated cells is used to determine the relationships between edges, nodes, and in consequence cell-tuples. The resulting hierarchical G-Map represents the interrelations between the topologies at different LOD.

Figure 7. Application example by courtesy of J. Haunert, IKG Hannover University: a section of ca. 2 % of a digital map on land-use at three different scales.

5. CONCLUSION AND OUTLOOK

In this article the realisation of a general model based on oriented hierarchical d-Generalized Maps to represent and analyse topology in MRDB has been described in detail. The model can be used as a data integration platform for 2D, 3D, and 4D topology. Typical examples for elementary and complex topological operations for multiple representations have been presented and illustrated. An application example with 2D cartographic datasets from Hannover University showed the feasibility of the new approach. It can also be used to combine 2D maps and 3D models, the last-mentioned being the specialisation of the 2D map. The advantage of this approach is to have a single representation for describing 2D and 3D topology. In our future work we intend to focus on this aspect, e.g. in the context of 3D urban planning.

REFERENCES

Anders, K.-H., Bobrich, J., 2004. MRDB Approach for Automatic Incremental Update. In: *ICA Workshop on Generalisation and Multiple Representation,* Leicester.

Brisson, E., 1993. Representing Geometric Structures in d Dimensions: Topology and Order. In: *Discrete & Computational Geometry (9),* pp. 387-426.

Gröger, G., Plümer, L., 2005. How to Get 3-D for the Price of 2-D-Topology and Consistency of 3-D Urban GIS. *Geoinformatica, 9 (2),* pp. 139-158.

Fabri, A., Giezeman, G.-J., Kettner, L., Schirra, S., Schönherr, S., 1998. On the design of CGAl, the Computational Geometry Algorithms Library. *Research Report MPI-I-98-1-007,* Max-Planck-Institut für Informatik, Saarbrücken.

Fradin, D., Meneveaux, D., Lienhardt P., 2002. Partition de l'espace et hiérarchie de cartes généralisées. In : *AFIG 2002*, Lyon, décembre 2002, 12p.

Haunert, J.-H., Sester, M., 2005. Propagating updates between linked datasets of different scales. In: Proceedings *XXII Int. Cartographic Conference,* A Coruna, Spain July 11-16.

Hoppe, H., 1996. Progressive meshes. In: *ACM SIGGRAPH* 1996, pp. 99-108

Lévy, B., 1999: *Topologie Algorithmique - Combinatoire et Plongement.* PhD Thesis, INPL Nancy, 202p.

Lienhardt, P., 1994. Topological models for boundary representation: a comparison with n-dimensional generalized maps. In: *Computer Aided Design 23(1),* pp. 59-82.

Mallet, J. L., 2002. *Geomodelling.* Oxford University Press, 599 p.

Mallet, J.L., 1992. GOCAD: A computer aided design programme for geological applications. In: Turner, A.K. (Ed.): *Three-Dimensional Modelling with Geoscientific Information Systems*, NATO ASI 354, Kluwer Academic Publishers, Dordrecht, pp. 123-142.

Mäntylä M., 1988. *An Introduction to Solid Modelling.* Computer Science Press, 401 p.

Meine, H., Köthe, U., 2005. The GeoMap: A Unified Representation for Topology and Geometry. in: Brun, L., Vento, M. (Eds.): *Graph-Based Representations in Pattern Recognition*, Proc. GbR 2005, LNCS 3434, pp. 132-141, Springer, Berlin.

MOKA, 2006. Modeleur de Cartes. http://www.sic.sp2mi.univ-poitiers.fr/moka/ (accessed 21.03.2007).

PostGIS.org, 2006. http://postgis.refractions.net/documentation (accessed 21.03.2007).

PostgreSQL.org (2006): http://www.postgresql.org/docs (accessed 21.03.2007).

Shumilov, S., Thomsen, A., Cremers, A.B., Koos B., 2002. Management and visualisation of large, complex and time-dependent 3D objects in distributed GIS, In: Proc. *ACM-GIS 2002*, pp. 113-118.

Thomsen, A., Breunig, M., 2007. Some remarks to topological abstraction in multi representation databases. *Proc. Int. Workshop on Information Fusion and Geographical Information Systems IF&GIS´07,* St. Petersburg, 12p. (in print).

ACKNOWLEDGEMENTS

This work is funded by the German Research Foundation (DFG) in the project "MAT" within the DFG joint project "Abstraction of Geoinformation", grant no. BR 2128/6-1.

IMPLICIT SHAPE MODELS, MODEL SELECTION, AND PLANE SWEEPING FOR 3D FACADE INTERPRETATION

Sergej Reznik, Helmut Mayer

Institute of Photogrammetry and Cartography, Bundeswehr University Munich, Germany
Sergiy.Reznik|Helmut.Mayer@unibw.de

KEY WORDS: Facade Interpretation, Implicit Shape Models, Model Selection, Plane Sweeping, Markov Chain Monte Carlo

ABSTRACT:

In this paper we address the automatic 3D interpretation of facades from terrestrial image sequences making two novel contributions: First, we employ Implicit Shape Models (Leibe and Schiele, 2004) coherently for the detection as well as for the delineation of windows, allowing to learn the appearance of windows and their outline from training data. Second, we use model selection to choose the most appropriate model for the configuration of windows in terms of rows or columns. These components are complemented by plane sweeping for the 3D determination of the windows or rows / columns made up from them. Results show the feasibility of the approach.

1 INTRODUCTION

Facade interpretation from terrestrial wide-baseline image sequences has been a focus of research since the seminal paper of (Dick et al., 2004). They interpreted buildings in line with the trend in computer vision towards statistical generative models. Particularly they employ Reversible Jump Markov Chain Monte Carlo (RJMCMC) (Green, 1995) allowing to add and delete new parameters and therefore also objects. The results are impressive though restricted to a limited number of objects as the models are generated manually. A more geometric approach is taken by (Werner and Zisserman, 2002). They make use of the regular structure of buildings, particularly the existence of vanishing points. Specific geometric regularities such as the symmetries of dormer windows are used to obtain a high-quality textured model. Yet, the existence of these regularities is presumed to be known.

Our first main contribution of this paper lies in employing Implicit Shape Models – ISM (Leibe and Schiele, 2004) coherently for the appearance based detection as well as for the delineation of windows. While we used information of corners to delineate windows only on dark facades and employed black rectangles for bright facades in (Mayer and Reznik, 2006), we now delineate the outline of whole windows on any kind of facade via ISM.

The second main contribution can be seen as an inversion and at the same time extension of (Alegre and Dallaert, 2004) and (Brenner and Ripperda, 2006). We invert, as we do not split the facade, but rather detect and delineate objects and group the constituents into rows, columns, and finally also grids. We extend the above work as we employ model selection based on Akaike's Information Criterion (AIC) to compare different groupings. Basically, individual windows always lead to the best likelihood as they can adapt to the individual shapes of windows. Only by taking into account the lower number of parameters for rows, columns, etc., they will prevail. One particular contribution is to show how the likelihood term has to be interpreted to come up with meaningful results for our delineation of windows based on ISM. (Dick et al., 2004) have also used model selection, but to switch between different interpretation for windows, namely with and without an arc, etc.

We assume, that a wide-baseline image sequence is given, and employ (Nistér, 2004), which makes the reconstruction much more stable by additionally presuming that an (approximate) calibration is available. 3D Reconstruction leads to camera parameters and 3D points. From the latter we compute the facade planes via Random Sample Consensus – RANSAC (Fischler and Bolles, 1981). We orient the planes using the vertical vanishing points in the images, again employing RANSAC. All images looking at a particular facade are projected on its plane and combined using a consensus-based approach (Mayer, 2007) allowing to get rid of partial occlusions. We use a sampling distance of 1 cm to normalize the further processing.

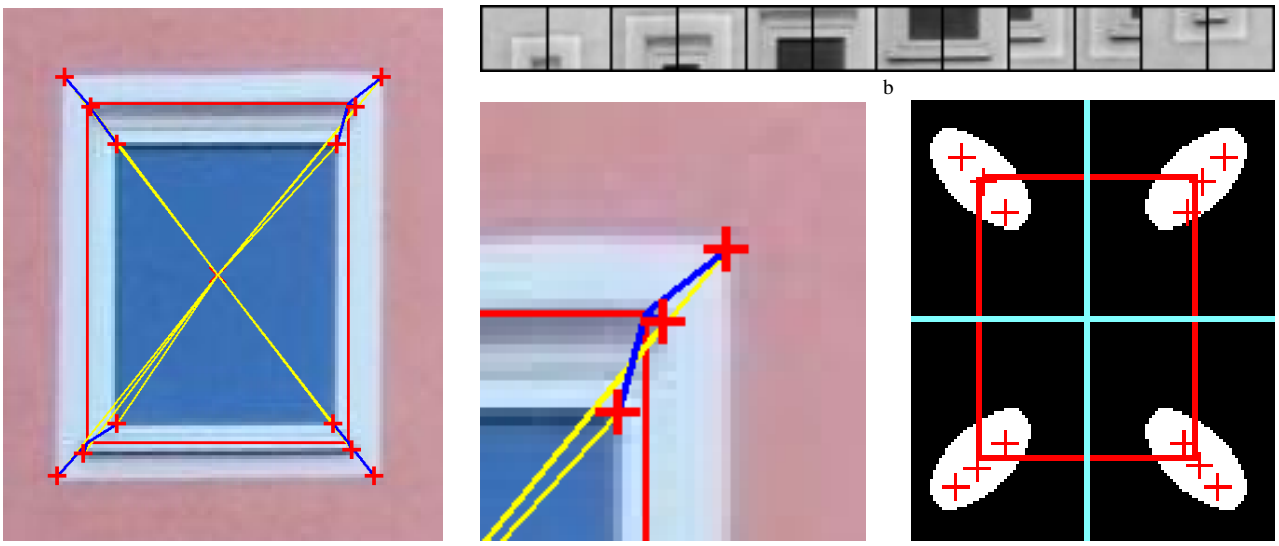
We first describe appearance based detection and delineation of windows on the facade plane images based on ISM in Section 2. Section 3 is devoted to model selection for the decision between a representation based on individual windows or rows or columns of windows. Plane sweeping leading to the determination of the depth, i.e., the 3D shape of windows, is described in Section 4. The paper ends with conclusions.

2 DETECTION AND DELINEATION OF WINDOWS BASED ON IMPLICIT SHAPE MODELS

We employ Implicit Shape Models – ISM (Leibe and Schiele, 2004) for the detection of windows, but also for the delineation of their outline.

For training we cut out image patches containing windows, in our case 120 windows of modern type. We note that none of the windows shown in our results is part of the training set and that we use the patches as well as their horizontally mirrored versions, making the algorithm more invariant to the viewing direction. The rectangular outlines of the windows are manually delineated (cf. (red) rectangle in Figure 1 a)). Only in elliptical areas around the corners of the outline (cf. Figure 1 d)) Förstner points (Förstner and Gülch, 1987) are extracted. The image patches around the Förstner points shown in Figure 1 b) are the basis for the appearance based detection of windows together with their arrangement relative to the center of the window computed from the manually delineated outline marked as yellow lines in Figure 1 a). For the delineation, the relation of the patches to the corners of the outline is used marked as blue lines in Figures 1 a) and c).

For the retrieval, i.e., for the detection of the windows, Förstner points are extracted with the same parameters as for training, but in the whole image (cf. Figure 4 a)). Patches around the points with a size of 35 pixels are then matched via cross correlation to all patches in the training data. If the cross correlation coefficient is above an empirically found threshold of 0.75, the match



a b c d

Figure 1: Training – a) Window with manually given outline of window (red rectangle), Förstner points at corners of window outline (red crosses) as well as their relation vector to the center of the window (yellow lines) and to their corresponding corner of the outline (blue short lines); b) Image patches around Förstner points; c) Detail of a) focusing on the relation of Förstner points to the corner of the window; d) Elliptical areas around window outline (white) where Förstner points are extracted.

is accepted and the vector relating the training patch to its center is used to generate a hypothesis for the center of the window in an initially empty accumulation image. The hypotheses are integrated via a Gaussian of the average size of the windows used for training and local maxima of the resulting function are regarded as hypotheses for windows. The patches which led to the maxima are hypotheses for corners of window outlines.

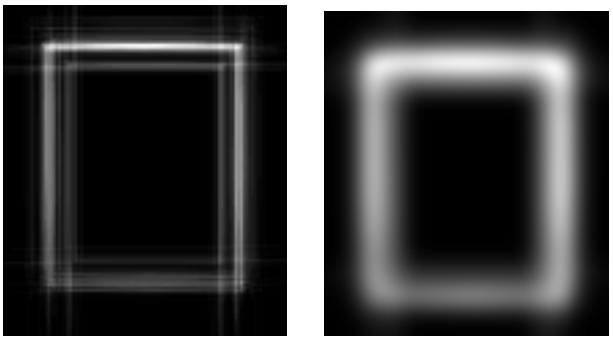
To precisely delineate the windows, we employ the relation between the centers of the training patches and the given outline of the windows marked as blue lines in Figure 1 a) and c). E.g., the point marked in red in the upper left corner of the dark window pane in Figure 4 a) has been matched by cross correlation to the training patch marked in red just left above the "b" of Figure 4 b). Figure 4 c) shows how the center of the patch marked by a thick red cross is related to the corner of the outline of the window marked by a small yellow cross. From the corner of the outline the two neighboring sides of the rectangle from the training data are drawn (cf. Figures 4 c) and d)). The result is a hypothesis for parts of the window outline.

The hypotheses for window outlines as in Figure 4 d) are accumulated over all points in the given image and all training patches. The result is a distribution for the window outline as in Figure 2 a) which is finally smoothed (cf. Figure 2 b)) and normalized by setting the largest value in the window to one.

Figures 3 and 5 give two results for distributions of window outlines. The distributions for the window outline are input to a Markov Chain Monte Carlo – MCMC (Neal, 1993) Maximum A Posteriori (MAP) estimation procedure. The employed prior punishes too small and too wide or too high windows. The likelihood function is the sum over the distribution along the window outline (e.g., cf. red line in Figure 6 a)).

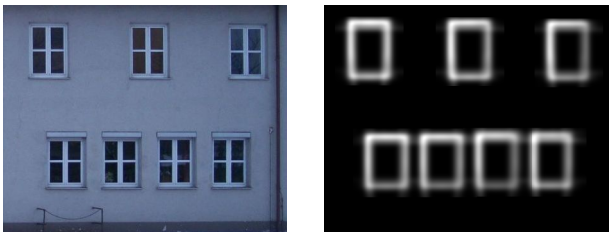
3 MODEL SELECTION: INDIVIDUAL WINDOWS, ROWS, AND COLUMNS

In the preceding Section we have described how to detect and delineate individual windows such as in Figure 7 a). Yet, windows are usually not arranged randomly, but in rows, columns, or grids.



a b

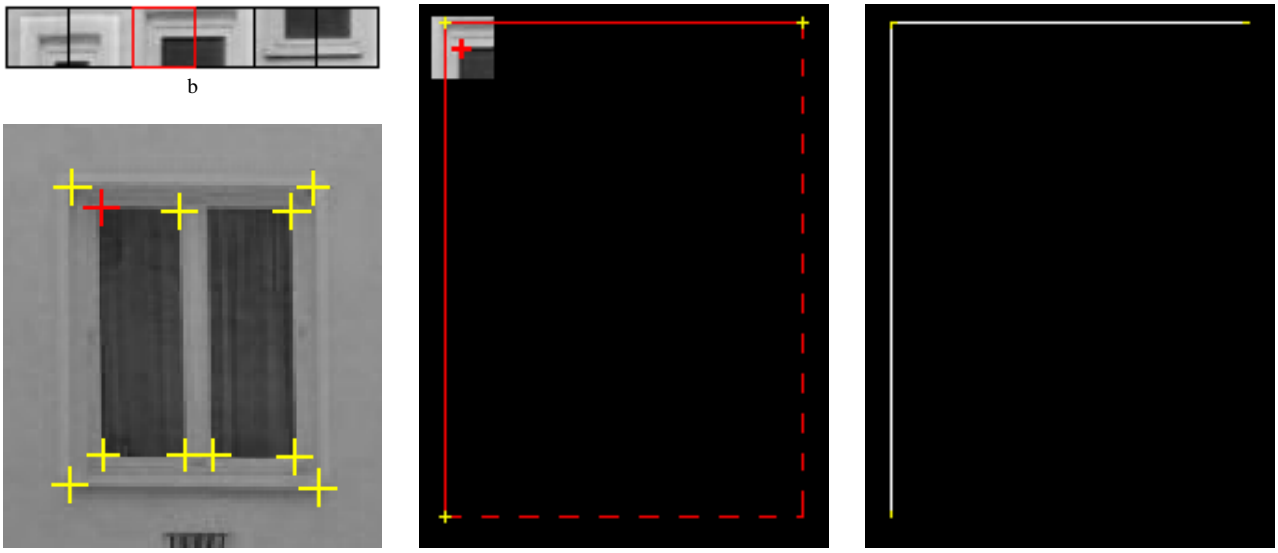
Figure 2: Distribution for window outline – a) accumulation; b) smoothing



a b

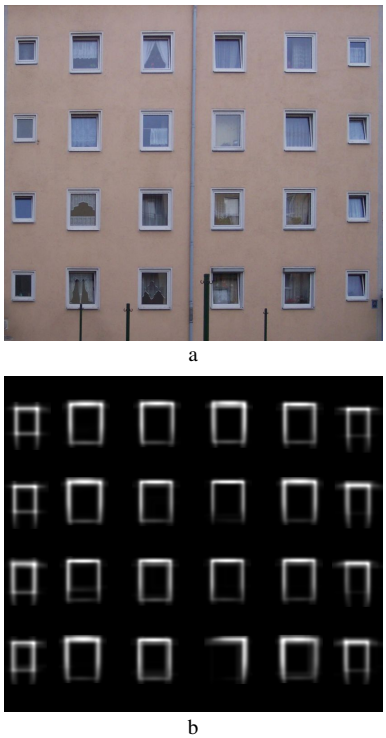
Figure 3: a) Facade and b) distribution for window outlines

Rows and columns, in this paper defined to have the same horizontal or vertical distance between windows of the same size, can be built by analyzing the horizontal or vertical arrangement. Yet, it is often not clear if one should represent a facade by means of individual windows or by rows or columns of windows. E.g., Figure 7 shows a configuration which can be represented adequately by means of columns, but not in terms of rows. Basically, in terms of an optimum fit described in the form of the likelihood always the individual windows will be preferred as they can optimally adapt to the data. Thus, one needs a way to reward arrangements of objects and one way to do this is to consider that they can be described by smaller number of parameters.



b a c d

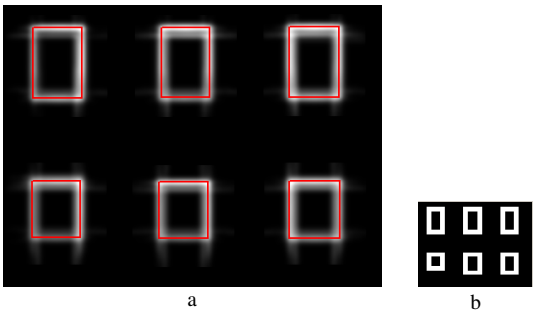
Figure 4: Retrieval – a) Förstner points; b) Training patches with the patch just left above the "b" being matched to the red cross at the upper left corner of the window pane in a); c) Relation of the center of the patch (red cross) to the window outline in the training data (left cross for position – lengths of sides from training data); d) Hypothesis for parts of the window outline



a b

Figure 5: a) Facade and b) distribution for window outlines

The above problem is thus regarded as a problem of model selection. Numerous ways have been devised to balance the complexity of a model, e.g., described by the number of parameters or their accuracy, on one hand and the adaptation to the data, i.e., the likelihood, on the other hand. Two well known are Minimum Description Length – MDL (Rissanen, 1978) and AIC – Akaike's information criterion (Akaike, 1973). A very good analysis of the relations of these two means as well as their characteristics, their strengths, and weaknesses can be found in (Schindler and Suter, 2006). For its simplicity and as we found it to work well for our application, we employ AIC, though recent work on composition



a b

Figure 6: Determination of the likelihood in the distribution for the window outline – a) Given image with outline in red; b) minimal size

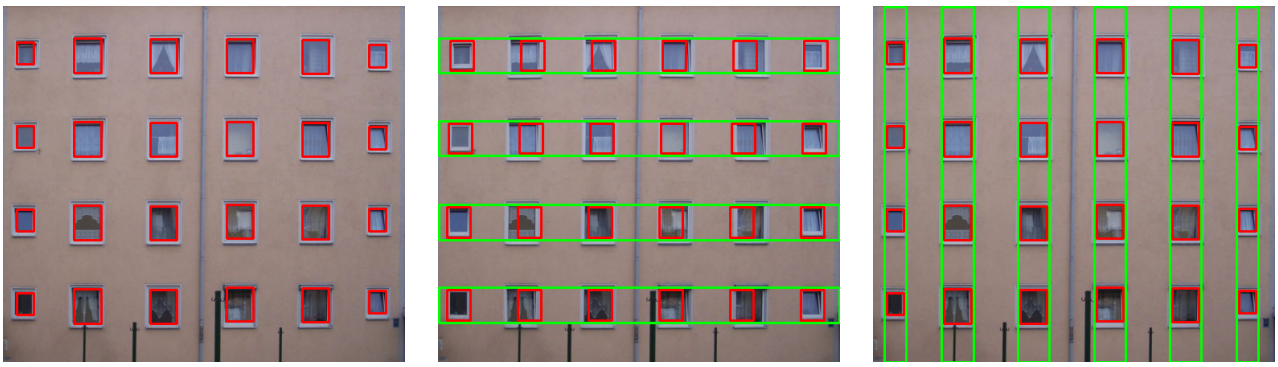
such as (Geman et al., 2002) prefers MDL. Particularly, we use

$$AIC = k - 2n \ln(L)$$

with k the number of the parameters of the model, n the number of observations, and L the likelihood of the outline. The number of parameters is four (width, height and center coordinates) for every individual window and five for a row or column (four parameters for window shape plus – horizontal or vertical – spacing). The likelihood is determined in the normalized distribution image described in Section 2 above by means of MCMC. Figure 6 a) shows how the distribution is sampled at one position with the outline given in red. The idea is that every boundary point gives one observation of the likelihood which are multiplied leading to the multiplication factor for the log-likelihood.

A couple of experiments led to experience that it is not sufficient to just sample the given distribution for windows. We concluded that one also has to reduce the determination of the likelihood to a minimal setup. Thus, we derived from the sampling theorem that for a window consisting of parallel lines the minimum size is a length of just above three pixels. We accordingly resample the distribution image to this minimum size (cf. Figure 6 b)) for the computation of the likelihood for AIC. (Note: For the delineation the original resolution is used to obtain a higher accuracy.)

Results for this procedure are given in Figure 8. For all three



a b c

Figure 7: Model Selection – Representation of facade by a) individual windows; b) by rows; c) by columns of windows, the latter consisting of windows with the same size and a constant horizontal or vertical spacing.

given as well as many other facades we tested our procedure on we selected the correct model. If there is an obvious structure on the facade, it is reflected in significantly different AIC values as in Figure 8.

4 3D RECONSTRUCTION VIA PLANE SWEEPING AND RESULTS

The result from the above procedure are the outlines of windows on the facade images possibly restricted to form horizontal rows or vertical columns. As we use image sequences as basis, we can also determine the 3D extent of the windows on the facade planes. To do so, we follow (Baillard and Zisserman, 1999) and (Werner and Zisserman, 2002) and employ plane sweeping, in this case in the direction of the normal of the individual facade plane. The determination of the depth for individual windows is based on the sum of the least squares differences between the projections of the plane into the individual images to their average image. This is computed for a meaningful range of depth values for windows and the result is the depth value for the minimum of the sum. For rows or columns we sum up the contributions of all images of a row or column at a particular depth.

Figure 9 shows four images of a sequence with seven images and Figure 10 the result for three manually coarsely marked facades. In Figures 11 and 12 further results are given showing in both cases two again coarsely marked facades respectively. While for the first three facades rows of windows were chosen by model selection, it decided for the two facades of the second example, that they are better described by means of columns, and for the two facades of the third example it selected individual windows, as the columns with different window sizes do not fit to our models. Please note that our rows and columns consist of windows with the same shape and a constant distance in either horizontal or vertical direction and we do the selection for the whole facade. The 3D reconstruction was done mostly reliably and accurately and led to the windows behind the facade marked by green rectangles which can be seen in Figures 10, 11, and 12.

5 CONCLUSIONS

We have presented two novel contributions for the interpretation of facades consisting of individual windows, i.e., no glass facades, from terrestrial image sequences, namely the coherent use of Implicit Shape Models for the delineation of windows and model selection based on Akaike's information criterion (AIC) for selecting between individual windows and rows and columns constructed from them. Combined with plane sweeping we obtain a 3D interpretation of facade planes including the windows.

Concerning future work we think into different directions. First, we need to do model selection for individual rows and columns in a more flexible way by using RJMCMC. Then, we want to create more detailed models of the windows including mullions and transoms, the appearance of both possibly learned in an appearance based hierarchy. On a more global level we want to integrate other objects such as doors on the ground level but also architectural details around windows possibly including their 3D structure as well as balconies. For the latter plane sweeping might be a solution for some shapes of balconies.

On a more global level we consider composition Systems (Geman et al., 2002) as an important theoretically sound basis for our hierarchical modeling ranging from the window details to grids made up of windows and other architectural objects. Another question is a statistically sound link between discriminative and generative modeling such as in (Tu et al., 2005).

ACKNOWLEDGMENTS

We want to thank Deutsche Forschungsgemeinschaft for supporting Sergej Reznik under grant MA 1651/10. We thank the anonymous reviewers for their helpful comments.

REFERENCES

Akaike, H., 1973. Information Theroy and an Extension of the Maximum Likelihood Principle. In: Second International Symposium on Information Theory, pp. 267–281.

Alegre, F. and Dallaert, F., 2004. A Probalistic Approach to the Semantic Interpretation of Building Facades. In: International Workshop on Vision Techniques Applied to the Rehabilitation of City Centres, pp. 1–12.

Baillard, C. and Zisserman, A., 1999. Automatic Reconstruction of Piecewise Planar Models from Multiple Views. In: Computer Vision and Pattern Recognition, Vol. II, pp. 559–565.

Brenner, C. and Ripperda, N., 2006. Extraction of Façades Using RJMCMC and Constraint Equations. In: International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. (36) 3, pp. 155–160.

Dick, A., Torr, P. and Cipolla, R., 2004. Modelling and Interpretation of Architecture from Several Images. International Journal of Computer Vision 60(2), pp. 111–134.

a – **C: 265**, R: 404, W: 359 b – C: 115, **R: 65**, W: 94 c - **C: 76**, R: 83, W: 87

Figure 8: Results for Model Selection using AIC – C: Columns, R: Rows, W: Individual Windows – selected model in bold

Figure 9: Images one, three, five, and seven of sequence Ostbahnhof-1

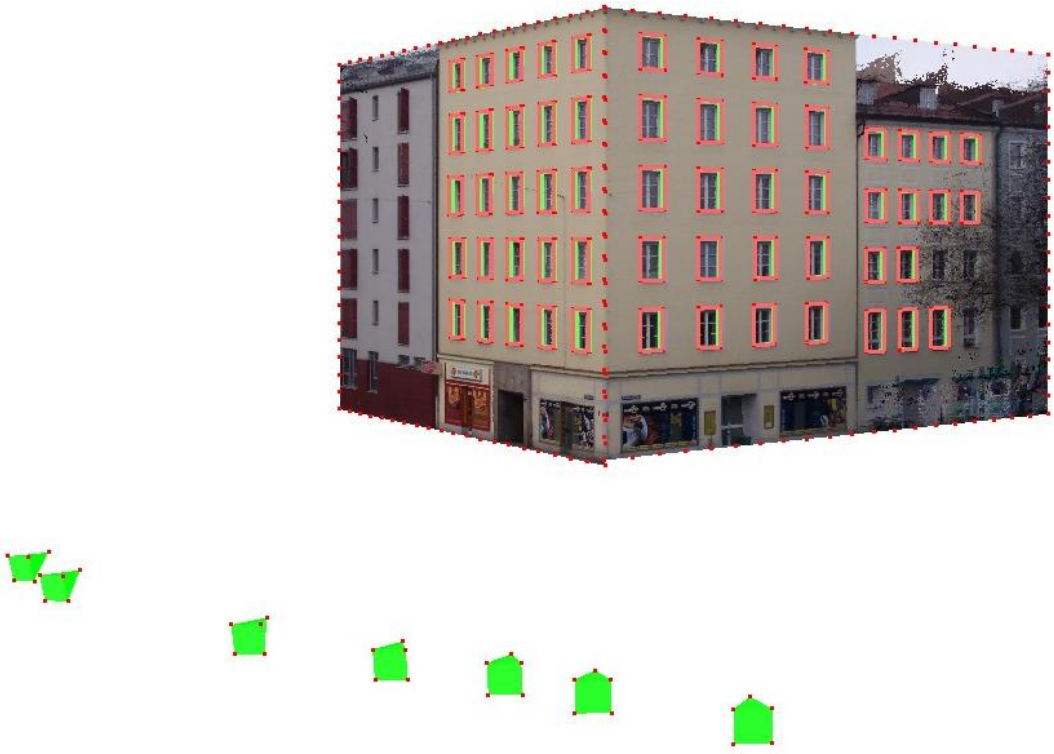


Figure 10: Result for sequence Ostbahnhof-1 (images cf. Figure 9) – Window outlines for three facades with rows of windows as red rectangles, 3D window positions as green rectangles, camera positions as green pyramids

Fischler, M. and Bolles, R., 1981. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. Communications of the ACM 24(6), pp. 381–395.

Förstner, W. and Gülch, E., 1987. A Fast Operator for Detection and Precise Location of Distinct Points, Corners and Centres of Circular Features. In: ISPRS Intercommission Conference on Fast Processing of Photogrammetric Data, Interlaken, Switzerland, pp. 281–305.

Geman, S., Potter, D. and Chi, Z., 2002. Composition Systems. Quarterly of Applied Mathematics LX, pp. 707–736.

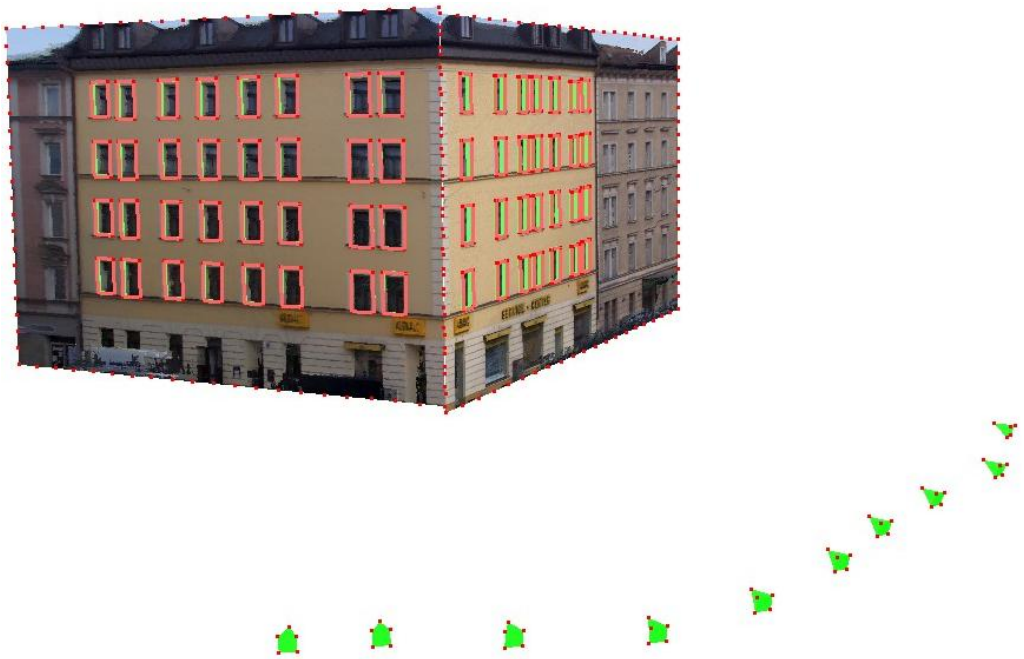


Figure 11: Result for sequence Ostbahnhof-2 with columns of windows constructed from ten images – explanation cf. Figure 10

Figure 12: Result for sequence Bordeaux-3 with individual windows constructed from eleven images – explanation cf. Figure 10

Green, P., 1995. Reversible Jump Markov Chain Monte Carlo Computation and Bayesian Model Determination. Biometrika 82, pp. 711–732.

Leibe, B. and Schiele, B., 2004. Combined Object Categorization and Segmentation with an Implicit Shape Model. In: ECCV'04 Workshop on Statistical Learning in Computer Vision, pp. 1–15.

Mayer, H., 2007. 3D Reconstruction and Visualization of Urban Scenes from Uncalibrated Wide-Baseline Image Sequences. Photogrammetrie – Fernerkundung – Geoinformation 3/07, pp. 167–176.

Mayer, H. and Reznik, S., 2006. MCMC Linked with Implicit Shape Models and Plane Sweeping for 3D Building Facade Interpretation in Image Sequences. In: International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. (36) 3, pp. 130–135.

Neal, R., 1993. Probabilistic Inference Using Markov Chain Monte Carlo Methods. Technical Report CRG-TR-93-1, Department of Computer Science, University of Toronto.

Nistér, D., 2004. An Efficient Solution to the Five-Point Relative Pose Problem. IEEE Transactions on Pattern Analysis and Machine Intelligence 26(6), pp. 756–770.

Rissanen, J., 1978. Modeling by Shortest Data Description. Automatica 14, pp. 465–471.

Schindler, K. and Suter, D., 2006. Two-View Multibody Structure-and-Motion with Outliers Through Model Selection. IEEE Transactions on Pattern Analysis and Machine Intelligence 28(6), pp. 983–995.

Tu, Z., Chen, X., Yuille, A. and Zhu, S.-C., 2005. Image Parsing: Unifying Segmentation Detection and Recognition. International Journal of Computer Vision 63(2), pp. 113–140.

Werner, T. and Zisserman, A., 2002. New Techniques for Automated Architectural Reconstruction from Photographs. In: Seventh European Conference on Computer Vision, Vol. II, pp. 541–555.

TOWARDS SEMANTIC INTERACTION IN HIGH-DETAIL REALTIME TERRAIN AND CITY VISUALIZATION

Roland Wahl **Reinhard Klein**

AG Computer Graphik
Institut für Informatik II – Universität Bonn
{wahl,rk}@cs.uni-bonn.de

KEY WORDS: DTM, DSM, city model, LOD, multiscale methods, realtime visualization, semantic interaction, ontology

ABSTRACT:

On the one hand, the extent, modeled detail and accuracy of virtual landscapes, cities and geospecific content is rapidly increasing. Realtime visualizations based on geometric levels-of-detail (LODs) allow the user to explore such data, but up to now, the methods of interaction are very low-level. On the other hand, we have semantic categories for the objects which are modeled in ontologies. We propose an approach which allows to combine the advantages of both, realtime visualization techniques and semantic hierarchies, in a single application without establishing an explicit link. That way, we can achieve semantic interaction without interfering with the rendering techniques which is crucial for performance on large datasets. Moreover, we are able to exchange geometric and semantic models independently of each other.

1 INTRODUCTION

Recent advances in digitization technology and reconstruction methods have lead to the availability of huge high-resolution 2.5d digital surface models (Hirschmüller, 2005). As sensors also record views at slightly tilted angles, to a certain degree also 3d reconstruction from aerial data is possible. Reconstructions from these aerial data will help to match and integrate data obtained from terrestrial sensors (Früh and Zakhor, 2003), so that in future we will face captured data sets of cities which bear high detail in full 3d, i.e. huge raw point clouds with spatial resolution in the range of single centimeters. These advances in data acquisition and fusion go side by side with progress in realtime rendering methods which become capable of visualizing the ever growing data sets in full detail.

Concepts in our mind tell us that something is a house, a church, a balcony or something else depending on its appearance and our experience. We are used to perceive things with high visual detail and at the same time think about them in the compressed form of semantic categories. For efficient human computer interaction a system must match these abstract concepts of our mind, i.e. we have to close the semantic gap. To this end data corresponding to different semantic entities are labeled accordingly, which results in a semantic model.

Currently, coming along with the increasing detail of the captured data we also observe an increasing demand for semantic models. Developments in 3d GIS lead to domain-specific ontologies which are also rapidly growing in that they add more and finer semantic categories and metadata (Kolbe et al., 2005). Semantic models based on such ontologies represent the underlying geometry in different simplified versions depending on the semantic level of detail (LOD). Details which are not semantically relevant for the model are neglected and at most represented by textures. This way the reconstruction and semantic modeling of parts of the scene that are not required for a specific ontology can be avoided which saves reconstruction time and costs. Therefore, much information contained inherently in the captured data sets is not mapped into the semantic domain and therefore not available in the semantic model. However, these details, although irrelevant from a semantic interaction point of view might still carry important cues. Examples would be the natural cover and topography in front gardens which are important for the rating of real estates, whereas on the semantic level cadastral data and building data would suffice.

Unfortunately, current terrain and city visualization systems that allow for 3d semantic interaction build on the geometry of the corresponding semantic models only and omit the additional information contained in the captured data. To achieve interactive performance, only parts of the terrain or city-models are rendered and in addition only coarse representations like extruded ground polygons on LOD1 or extruded ground polygons with roof structures on LOD2 are used (Gröger et al., 2004). Due to the rising amount of data higher LODs like LOD3 and LOD4 would require additional multi-resolution techniques to achieve interactive performance and are therefore currently used only selectively. Combining these semantic LODs with view-dependent geometric LODs is a non-trivial problem that is currently not solved, since the geometric and semantic hierarchy must be intertwined. While from a rendering point of view representing planar facades of several neighboring buildings as a single polygon with texture is appropriate, the semantic model requires the geometric representation to respect the borders of the houses.

If semantics and geometry were not kept separate, one possible way to handle this situation would be to choose a representation based on the semantic category. The problems with this approach are firstly, that the optimal representation is not necessarily consistent throughout a semantic class and secondly, that we have to decide for every element of the raw data which category it belongs to. Also semantic categories often overlap, are ambiguous or decompose geometric entities into non-trivial subparts.

Therefore, we suggest to use different geometric representations for the semantic and geometric hierarchies of terrain and city models. The geometric representation, in the following called *rendering model*, which is actually visualized is based directly on the captured data. The representation of the semantic entities, in the following called *interaction model*, which is only used for interaction purposes is built on reconstructed and modeled data, like the above mentioned semantic LODs. These separate representations of geometric and semantic data are joined on-the-fly in interactive rendering systems. This approach enables us to

implement semantically based interaction within high-resolution virtual worlds. Furthermore, it allows to combine interaction models for different ontologies with the same rendering model. Even on-the-fly exchange of ontologies can be achieved by loading different interaction models. This is especially useful as the ontology, the semantic LOD as well as the spatial extent of the semantic model can be selected dependent on the specific task. In addition, geometry data and semantic information are usually obtained and created by very different and separate processes as well as different people. Therefore, separate representations fit naturally in the corresponding graphics and GIS workflows and modeling of geometry as well as semantic information becomes substantially easier as none of them has to consider the intricacies hidden in the other, separate system. Last but not least, separate representations allow independent modification of either geometry or semantics at any time. This is especially of interest for update purposes where either the interaction model is further refined or the rendering model is updated, e.g. by adding new data.

In the following, we first concentrate on the rendering aspect. We discuss the problems arising during interactive visualization of high-definition terrain and city models, as well as the consequences for the rendering model in the next section. Then we discuss the interaction aspect in section 3 which describes several computer graphical methods that can be employed to connect the semantic data with the rendering model. This in turn leads to the definition of the interaction model. Some results are presented in section 4 before we come to a brief conclusion in section 5.

2 REALTIME TERRAIN RENDERING

2.1 High detail terrain and city models

For the purpose of photorealistic rendering we need a model which captures as much of the photometric and geometric details as could be perceived in the real world. Obviously, the perceivable detail depends on where the camera is placed in the virtual world. As long as the application shows the terrain from high altitude aerial views, an orthotextured digital terrain model (DTM) is appropriate. Closer to the ground, we need a digital surface model (DSM) in order to perceive the correct parallax and occlusion. Even high-detail textures mapped on a DTM will spoil the realism, due to the contradicting depth-cues (experience tells you that roofs are above ground, but the identical parallax suggests that the roof is at the same height as its surroundings).* For terrestrial or almost terrestrial views the 2.5d modeling approach suffers from systematic problems as facades and other steep parts of the geometry are not represented in orthotextures. Moreover, relevant geometry may be occluded from above. Thus, simply increasing the resolution of the 2.5d model will not suffice. Instead, we have to switch to a 3d model to be able to represent the scene with high precision from terrestrial perspectives as well.

2.2 Insufficiencies of terrain rendering

Multiresolution algorithms for fast rendering of large terrain data sets with viewpoint adaptive resolution have been an active area of research in the field of computer graphics for many years. Since giving a complete overview is beyond the scope of this paper, we refer to the surveys (Lindstrom et al., 1996, Pajarola, 2002).

Although there is a wide variety of methods regarding the details, the basic principles are always:

1. Choose a reasonable granularity for LODs. View-dependent simplification of individual primitives on-the-fly is more expensive than rendering a larger number of primitives. Batches containing a part of the model at the same LOD can be rendered more efficiently.

2. Use these batches for the choice of parts of the model which are visible for the camera (view-frustum culling) and the choice of required LOD.

3. As we deal with out-of-core datasets (i.e. data of such size that it breaks the bounds of physical memory of a computer) we need to implement a preprocessing stage where the data is processed into a hierarchy of batches, which can then be loaded on demand.

4. Choose effective compression algorithms for the data that lessen storage and bandwidth requirements for the model without introducing performance penalties during decompression.

We base our terrain and city rendering algorithm on the terrain rendering system presented by (Wahl et al., 2004) which is based on a quadtree data structure (see Fig. 1). The system has proven to be able to visualize very large terrain data sets efficiently and with high quality, e.g. data sets with a resolution up to a few centimeters for the aerial photography together with elevation models of about 1m, covering areas of hundreds of square kilometers, have already been visualized with realtime frame rates.

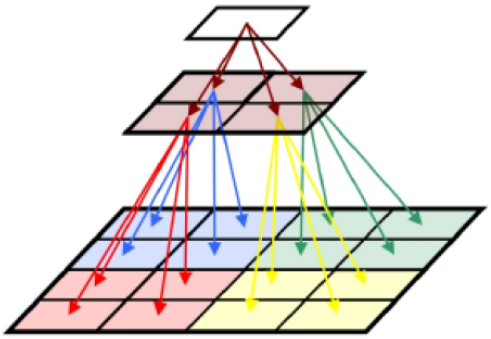


Figure 1: Quadtree layout of terrain rendering. Each part of the model belonging to a quadtree cell (called tile) has the same raster size independent of LOD.

In order to evaluate the different character of landscapes and cities over the scales, let us compare the number of primitives needed for a tile. In a city we may have about 20 buildings per block of size 100mx100m which is 2,000 buildings per km^2. If a tile of a square kilometer should be pixel correct we need $\sim$8m per pixel accuracy (1km per 128pixels) which means that every building needs to be present in the data. Even if each building is modeled by a hemicube with 5 faces only this leads to 20,000 triangles/tile just due to the buildings itself, not even accounting for the triangulation overhead at the floor and other features, like trees.

Let us now take a closer look at the root of the problem: In the presence of buildings and trees, the complexity in small tiles is comparable to that in alpine regions ($\sim$1,000),† but as opposed to the latter reducing the approximation accuracy does not lead to a smooth diminution of complexity. Instead, the complexity is rising to a maximum and finally breaks down when building

*This effect becomes even more striking if seen on stereo displays, where it is noticeable even in still images.

†As measured by (Wahl et al., 2004) using Hausdorff-distance based mesh approximations.

heights fall below the necessary accuracy. As due to the LOD scheme, the screen-size of a tile remains constant, this means we get more than 1 triangle/pixel, which indicates clearly that such a mesh is the wrong representation of the data.

This example demonstrates that the assumptions stating that the complexity density of representation is roughly independent of scale, which was made for terrain models, do not hold for city data and the scalability of classical terrain models is violated.

Moreover, in 2.5d models the representation of steep geometry most prominently at facades is inappropriate. That is even if terrain visualization did work, it would need to adapt to the incorporation of 3d data.

In summary we can state that there is already a fundamental difference between rendering huge terrain data sets which are modeled as DTM and rendering even comparatively small DSMs. Given a data accuracy and density in the range below meters yields a lot of complexity in otherwise harmless (i.e. flat) data sets. The reason for this is the disproportionate distribution of features to scales. On a 20m scale only hills, rivers, shores and mountains dominate the complexity, whereas on a single meter DSM even in flat regions every tree, bush, car, building etc. leaves its high-frequency fingerprint. As a result of such details a visualization of a single city on a meter scale can be more demanding regarding the level-of-detail scheme than visualizing planet earth on a 20–30 meter basis. Moreover, the representation of the surface as 2.5d is less appropriate for high detail scales than it was for classical terrain models.

2.3 The rendering model

Nevertheless, on coarser levels the geometry still keeps its DTM characteristics and therefore, current highly optimized terrain rendering techniques remain the first choice. In addition, most of the earth's surface is not yet modeled other than with a DSM. The question therefore is how to represent the detailed geometry. We want to decompose the data set into a terrain model and highly detailed 3d geometry like buildings and natural cover which are added in the visualization. This decomposition does not need to respect any kind of semantics. That is, for mere visualization performance we will not necessarily distinguish between what belongs to terrain and what not, but we want to automatically generate representations which consist of hybrid mesh, points or volume data, or any other representation which is applicable.

One way to represent such non-2.5d details is to represent them as point clouds. The rationale behind modeling with points is simple. Point clouds can be trivially extracted from any other boundary representation by surface sampling. Laser scanners, other range finders and stereo reconstruction generate points natively and the raw point cloud thus contains all details present in the data without any interpretation. An additional advantage of point clouds is that they easily represent high-frequency features. A small bump on a plane is modeled using a small pyramid in case of a triangle mesh. This however takes 4 points and 8 additional triangles (3 for the pyramid and 5 to maintain the mesh topology in the plane), whereas with points we could use a single point. Therefore point clouds have become an important means of modeling and as a consequence a lot of rendering techniques and LOD-schemes for points have been developed (Sainz and Pajarola, 2004).

Despite the apparent advantages of the point cloud representation, it leads to problems due to the high depth complexity of the city model (i.e. a high number of intersections of a half ray emanating from the view point with the model) if viewed at acute angles. In this case the number of point rendering primitives remains high although the projected screen size of the model is small, i.e. multiple points are drawn into the same pixel. In such a situation billboards, where the geometry is approximated by textured planes are advantageous (Décoret et al., 2003, Meseth and Klein, 2004). We therefore build our rendering model on a combination of point rendering with billboards as described in (Wahl et al., 2005).

3 INTERACTIVE VISUALIZATION

In this section we will discuss how semantic information can be integrated into a visualization. Of course, the mere fact whether semantic information is available or not does not affect the visual appearance. So, the degree of semantic enrichment of a model is measured by how the user is able to interact with the model. We want to achieve an interactive application which gives the user the impression that the machine has a similar concept of the objects which are in the scene.

Apart from standard interaction features:

- the application reacts on user input through navigation
- the application renders metainformation of selected objects onto the screen

we want to be able to perform *semantic* interaction:

- the application activates objects or visualizes metadata it has linked with objects on demand (picking)
- the application emphasizes objects in the visualization based on semantic information (highlighting)
- semantics-based navigation
- the application is able to temporarily delete objects from the scene, such that the camera looks through them
- the application synthesizes objects on demand

For semantic modes of interaction, the application needs a concept to translate user queries from a semantic level to a geometric level (used for rendering images) and vice versa.

3.1 Implicit modeling of semantics

As we do not want to interfere with the rendering model, we cannot store semantic information directly with the original geometry. We store the missing link between the 3d rendering model and the semantic categories and metadata in the interaction model. The interaction model is linked with the rendering model implicitly by means of its spatial reference. Although this is a common technique in the 2d case (e.g. mapping thematic layers on orthophotos or topographic maps), the extension to 3d is not trivial. In the 2d case, the mapping is easily achieved by reprojecting the data into the target's coordinate system and highlighting or picking can be performed by using image overlays. In the 3d case, a 2d overlay will not suffice, since it does not discriminate along the vertical axis (e.g. windows on different floors). However, generating a full 3d overlay is no option, as it requires exactly the semantic 3d model we wanted to get rid of. Our solution to this dilemma is to use proxy objects in the interaction model. For example, in the semantic model a house can be modeled by a cube although it is a detailed point cloud

in the rendering model. The link is established by performing a 3d (i.e. volume) intersection between the proxy geometry and the rendering model.

This approach effectively implements the relation between data objects and semantic entities and has the following advantages:

- There is no explicit link between the rendering model and the semantic model.
- There is no necessity to deal with the different LODs of the rendering model.
- The rendering model can be changed or replaced with no influence on the interaction capabilities.
- The interaction model can be updated dynamically without affecting the rendering, i.e. interaction models for different ontologies can be loaded on-the-fly.
- Direct compatibility with traditional 2d or 2.5d GIS data. Proxy geometry for such data can be trivially generated on-the-fly by extrusion.
- The intersection test with the proxy geometry can be evaluated on the graphics hardware (see section 3.3) which is very fast and easy to implement.
- Output sensitivity. The overhead is solely determined by the query complexity irrespective of the complexity of the rendered model. Especially, this means that without any semantic interaction we have no additional costs at all.

However, there are also some limitations:

- The rendering model must use accurate 3d coordinates, so that the volume intersection tests leads to the expected result. This constraint is obviously fulfilled by most geometry representations (triangles, points, voxels) but it does not hold for image based approaches.
- Skipping parts of the model based on semantic information (see section 3.4) is substantially more difficult to implement than with explicit links.
- As the user can interchange models, depending on the accuracy of these models and temporal changes in the meantime they can be inconsistent. This is unlikely for models derived from the same raw data sets and therfore rather a side-effect of interchanging models as a real disadvantage.

Alternatively to the implicit linking via volume intersection, we could also explicitly store the inverse relation between geometry and semantic entities. Instead of telling the geometry which semantic object it belongs to one would establish a mapping of semantic objects to geometric primitives. If this mapping is implemented with the ability to address sub-parts of a geometry (e.g. the part of a triangle belonging to a window), the model could be employed without constrainingthe simplification. The task during semantic interaction then consists of identifying the object by querying the inverse relation which is admittedly less efficient. Although the direct influence of the semantics on the preparation of the model is remedied, there is still a close explicit link between semantic and rendering model: In order to address the geometry of an object we need detailed information on how the model is represented. Moreover, a representation which lends itself to efficient rendering might still be very complex when we have to address parts for example in point clouds or billboards.

3.2 Interaction model

As mentioned earlier, the interaction model represents the semantic hierarchy and therefore its underlying ontology within the visualization application. It is composed of the actual semantic information i.e. categories and metadata as well as the proxy geometry. Apart from the difference in the geometric representation any type of semantic model can be directly used as interaction model, e.g. CityGML (Kolbe et al., 2005). Therefore, we will not discuss the semantic features here, but focus on the requirements for the proxy geometry.

Regarding the representation, the proxy geometry should consist of well-defined solids, ideally in form of an oriented boundary representation as these can be used directly for rendering. Apart from that there are three observations which influence the design of such proxies:

1. Interaction may take place on a coarser level-of-detail than visualization. Inaccuracies in the range of pixels would compromise the visual quality, but human interface devices are rarely placed with pixel accuracy.
2. Interaction models remain hidden from the user. As highlighting is not implemented as a 2d overlay, we can change the appearance of the model by evaluating the 3d proximity to the semantic model, thus the proxy geometry carrying the semantic information is an invisible layer.
3. Picking results are predictable even with coarse interaction models. The predictability of the picking is owed to the accuracy and simplicity of the proxies. Buildings and related objects in cities can mostly be well described by simple features like a small number of boxes or polyhedra. Moreover, an immediate feedback to the user can be used to verify whether the active object in the interaction model matches the intention of the user.

As a consequence, we do not need to model the proxies with the same detail as a representation intended for rendering. Actually, we just need to ensure that the object under consideration is within the volume and no other object intersects the volume. As we are dealing with models of bounded accuracy, this implies that we should inflate the proxy solids a bit in order to avoid that parts of the rendering model protrude the volume.

For performnce reasons, the proxy geometries should also be organized in a spatial hierarchy like the quadtree mentioned earlier. This will improve performance when a large number of highlighted objects is not within view and can therefore be skipped. Activating only parts of the interaction model, based on the specific application will further accelerate interaction in the case of very complex models.

3.3 Implementation issues

The implementation of semantic interaction based on implicit mapping using proxy geometry is straight forward. For highlighting, however, there is an optimized implementation which exploits graphics hardware.

The highlighting works as follows:

1. Render the whole scene. This sets the colors of the framebuffer (the buffer which is displayed on the screen) but also the corresponding z-buffer (an off-screen buffer which records the distance of each drawn pixel to the camera, used for correct occlusion).

2. Render the active object into the stencil buffer (a special-purpose auxiliary buffer which can be efficiently queried and updated during rendering):

   a) Disable depth-buffer and frame-buffer writes. Set stencil operation to increase stencil for pixels less distant than the z-buffer (z-pass) and render back-facing polygons of the proxy.

   b) Set stencil operation to decrease stencil on z-pass and render front-facing polygons of the proxy.

3. Now the stencil is positive in all pixels which have depth values within the proxy geometry. Activate frame-buffer write, blending for transparency and stencil test. Render screen-sized quad with the highlight color.

As only the colors are changed, it is clear that this process can be applied to any number of objects simply by iterating phases 2 and 3. In fact it suffices to simultaneously perform step 2 for all objects with the same highlight color.

Picking can either be implemented by using the highlighting feature to render unique IDs into an auxiliary buffer, which then tells which objects cover which part of the screen or by performing a z-readback (transferal of the depth-buffer from graphics hardware to main memory) which yields the 3d point at the given screen coordinates. This point can then be transformed into a spatial query to the semantic model.

3.4 Advanced interaction

While the picking and highlighting features are easy to implement because they only rely on information gathered in the hardware buffers of the graphics processing unit (GPU), tasks such as looking through (i.e. deleting) objects are more intricate. However, in many scenarios of future 3d-GIS applications such features may be especially worthwhile. Although omitting objects from rendering is a lot easier when we have direct access to the model, there are still possibilities to achieve the same result without touching the rendering model.

The problem with the transparency feature is that we need to know what would have been seen if the occluding (deleted) object was not there. Of course, we can apply our highlighting framework to determine which fragments [‡] of the image are the false occluders, but as our output consists only of 2d (color and depth) buffers there is no way to access information behind these. The obvious way to circumvent this problem is to modify the rendering such that only fragments not within the proximity of deleted objects are rendered. This can be achieved using the hardware-support on modern GPUs by either performing clipping of the rendering primitives in the geometry shader or by discarding fragments in the fragment shader based on an implicit representation of the proxy geometry. Clipping can be very efficient for convex polytopes as the in/out status can be established using a few half-space inclusion tests. The disadvantages of this approach are that it relies on computing capabilities only available on very recent hardware, the computation needs to be done every frame and the number of halfspace intersections will introduce a performance penalty, if many or complex objects are to be hidden. Preferably, the fragments should be discarded based on a single lookup. The stencil lookup however is insufficient, as it has no depth value. We can render the hidden proxies into an auxiliary z-buffer, but there we could only store one depth per fragment. In a 3d scene, however a high depth complexity (many primitives project to the same pixel) can be present especially when rendering terrain from almost terrestrial views. Therefore we would need to compute a list of depth intervals for each pixel which correspond to the different parts of the hidden proxies. Generating such lists is not feasible on current hardware. Therefore we propose to use a technique similar to shadow mapping (William, 1978). We make use of the fact, that city models still have a 2.5d character. Therefore, the complexity in vertical direction will generally be bounded by a small constant. So instead of storing the visibility information in the screen domain we will use a map-projection and store height values instead. The remaining z-complexity of the proxies' volume can be accounted for by storing multiple upper and lower contours. This approach has the additional advantage, that the visibility information remains the same for many frames and does not have to be recomputed. Recomputation is only necessary when and where objects change their invisibility status. In order to exploit this locality we will bind these auxiliary invisibility maps to the tiles of the geometric LOD-hierarchy.

Technically, the algorithm for hiding objects works in two parts:

Initialization of new invisible objects (if applicable):

- identify tiles of new objects
- initialize contour buffers
- set map projection
- render front/back faces in upper/lower contour buffer respectively

Scene rendering (every frame):

- reproject each fragment into map coordinates (gives height above ground)
- compare height to contour heights
- if height is within an upper/lower contour pair kill fragment

Note that again the additional complexity correlates with the query complexity (complexity of invisible objects) and does not depend on the scene, such that a true output-sensitivity is maintained.

4 RESULTS

We implemented the proposed scheme for semantic interaction within a realtime 2.5d terrain visualization framework to demonstrate the applicability. The rendering model is a $28cm$ resolution DSM colored with orthophotos of $7cm$ resolution. This model is courtesy of German Aerospace Center (DLR) – Institute of Robotics and Mechatronics and was derived by semi-global matching (Hirschmüller, 2005) from Vexcel Ultracam™ imagery. The semantic model was derived from "CityGML reference data on Pariser Platz" freely available for download[§]. The boundary representation of the solids modeling the semantic entities was used as the above mentioned proxy geometry.

Figure 2 shows a rendering of a part of Berlin with the highlighting feature. The extension of Hotel Adlon was selected by the user and is thus highlighted. In figure 3 the picking feature is demonstrated: As the mouse hovers over the image during realtime exploration the semi-transparent highlighting provides an

[‡] A fragment denotes the data necessary to generate a pixel in the frame buffer. This includes raster position, depth, color, stencil, etc.

[§] `http://www.3d-stadtmodell-berlin.de/`

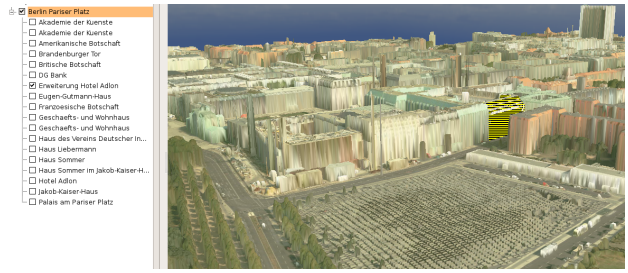


Figure 2: Screenshot from the rendering application showing a highlighted** feature of the semantic dataset within a visualization of the Berlin DSM.

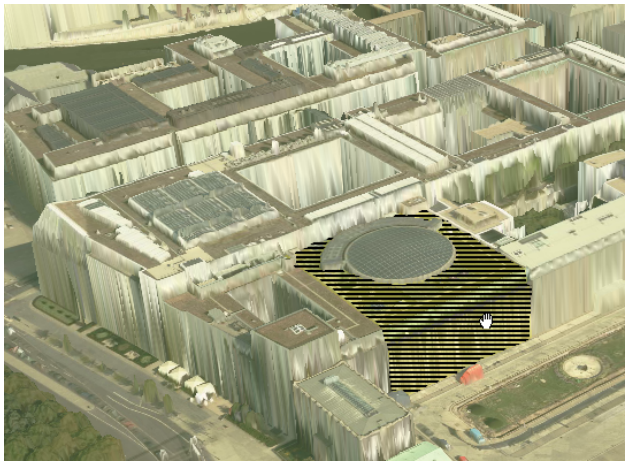


Figure 3: Screenshot from the rendering application. In picking mode semantic entities are highlighted** as the mouse hovers over them in the image. The user thus gets an instant feedback with which objects and on which semantic LOD he is about to interact.

instant feedback which objects are present in the active semantic category. With these objects we can interact by accessing meta-information or highlighting them as seen above. The glass roof of this building (Eugen-Gutmann Haus) is not highlighted because it is not within the volume defined by the semantic model which demonstrates that this mode of interaction is truly 3d.

5 CONCLUSION

In this work we introduced methods to allow semantic based interaction in highly detailed 3d terrain- and city models. In the proposed approach different ontologies can be used without changing the geometric LOD-hierarchy used for photorealistic rendering. In our opinion, the ability to switch the ontologies and thus the categories in which we think about the data is more important than the exact correspondence between geometric features and semantic entities. We have presented methods which allow interaction with photorealistic visualizations, so that the missing explicit link becomes unnoticable to the user.

5.1 Future Work

The ability to access detailed semantic information within a real-time photorealistic rendering framework is just a first step towards semantic interaction. With the combined strength of detailed semantic models and visually detailed instances there is a lot of place for new interaction paradigms especially concerning the modeling or synthetization of such scenes but also the exploration and visualization techniques.

Another direction of research will concern the flexibility of models. By now, all successful realtime rendering methods have to preprocess the data. Apart from the delay which is introduced by doing so, there is a conceptual difference whether the data needs to be static or it can be adapted on-the-fly.

**The black stripes were added to illustrate highlighting in the b/w hardcopy of the proceedings.

ACKNOWLEDGMENTS

This work was funded by the German Science Foundation (DFG) as part of the bundle project “Abstraction of Geographic Information within the Multi-Scale Acquisition, Administration, Analysis and Visualization”. We thank the Berlin Senate and Berlin Partner GmbH for making the CityGML model available. Finally, we also want to thank the reviewers for helpful comments.

REFERENCES

Décoret, X., Durand, F., Sillion, F. and Dorsey, J., 2003. Billboard clouds for extreme model simplification. In: Proceedings of the ACM Siggraph, ACM Press.

Früh, C. and Zakhor, A., 2003. Constructing 3d city models by merging aerial and ground views. IEEE Computer Graphics & Applications 23(6), pp. 52–61.

Gröger, G., Kolbe, T. H., Drees, R., Kohlhaas, A., Müller, H., Knospe, F., Gruber, U. and Krause, U., 2004. Das interoperable 3d-stadtmodell der SIG 3d der GDI NRW. `http://www.ikg.uni-bonn.de/fileadmin/sig3d/pdf/Handout_04_05_10.pdf` (15.07.2007).

Hirschmüller, H., 2005. Accurate and efficient stereo processing by semi-global matching and mutual information. In: Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, Vol. 2, pp. 807–814.

Kolbe, T. H., Gröger, G. and Plümer, L., 2005. CityGML – interoperable access to 3d city models. In: P. van Oosterom, S. Zlatanova and E. M. Fendel (eds), Proc. of the 1st International Symposium on Geo-information for Disaster Management.

Lindstrom, P., Koller, D., Ribarsky, W., Hodges, L., Faust, N. and Turner, G., 1996. Real-time continuous level of detail rendering of height fields. Proceedings of SIGGRAPH’96 pp. 109–118.

Meseth, J. and Klein, R., 2004. Memory efficient billboard clouds for BTF textured objects. In: B. Girod, M. Magnor and H.-P. Seidel (eds), Vision, Modeling, and Visualization 2004, Akademische Verlagsgesellschaft Aka GmbH, Berlin, pp. 167–174.

Pajarola, R., 2002. Overview of quadtree-based terrain triangulation and visualization. Technical Report UCI-ICS-02-01, Information & Computer Science, University of California Irvine.

Sainz, M. and Pajarola, R., 2004. Point-based rendering techniques. Computers & Graphics 28(6), pp. 869–879.

Wahl, R., Guthe, M. and Klein, R., 2005. Identifying planes in point-clouds for efficient hybrid rendering. In: The 13th Pacific Conference on Computer Graphics and Applications.

Wahl, R., Massing, M., Degener, P., Guthe, M. and Klein, R., 2004. Scalable compression and rendering of textured terrain data. In: WSCG, pp. 521–528.

William, L., 1978. Casting curved shadows on curved surfaces. Computer Graphics (Proceedings of SIGGRAPH ’78) pp. 270–274.

EFFICIENT SEMI-GLOBAL MATCHING FOR TRINOCULAR STEREO

Matthias Heinrichs*, Volker Rodehorst and Olaf Hellwich

Computer Vision & Remote Sensing, Berlin University of Technology, Franklinstr. 28/29, FR 3-1, D-10587 Berlin, Germany – (matzeh, vr, hellwich)@cs.tu-berlin.de

KEY WORDS: Photogrammetry, Area-based Image Matching, Semi-Global Matching, Trinocular Stereo, Similarity Measures

ABSTRACT:

This paper describes an efficient method for dense matching of two or three images. After some investigations in different similarity measures we propose a modification of Semi-Global Matching, which uses a simple energy function that implies piecewise smoothness but no ordering and gives promising results in practice. Our improvements include a symmetric and hierarchical matching strategy and allow an efficient generalization of the stereo matching problem to trinocular surface reconstruction. Finally, we present results for synthetic and for real images.

1. INTRODUCTION

The automatic three-dimensional (3D) reconstruction of an observed scene using digital images has been one of the core challenges in photogrammetry and computer vision for decades. Two or more images may be taken by different cameras at the same time (stereo) or by the same camera at different times (motion).

The key problem is how to find homologous image points which arise from the same physical point in the scene. If this *correspondence problem* is solved, depth information can be derived by triangulation using the orientation parameters of calibrated cameras.

The existing techniques for correspondence analysis can be distinguished by either matching some *features* (or relations between them) producing sparse depth information or matching all *pixels* in the images producing dense depth maps. For surface reconstruction tasks it is essential to compute dense depth maps using every pixel of the entire image.

Stereo image matching remains a difficult problem because of

- *Noise*, which arises from illumination variations and sensor noise during image formation,
- *Untextured regions* or *repetitive patterns*, which introduce ambiguities,
- *Occluded pixels* in one image, which should not be matched with pixels in the other image, as well as
- *Depth discontinuities* at object boundaries, which violate the spatial smoothness and ordering constraints.

1.1 Related Work

For a good overview and quantitative comparison of current two-frame stereo correspondence algorithms, we refer to (Scharstein & Szeliski, 2002), and for multi-view stereo reconstruction algorithms to (Seitz et al., 2006). The simplest algorithm compute stereo correspondence of image windows by searching along the corresponding epipolar line and match with the highest similarity (winner takes all).

The central problem of such *local* matching methods is to determine an optimal support region for each pixel. An ideal window is adaptive and should be bigger in homogeneous regions and smaller at depth discontinuities (Veksler, 2003). To reduce ambiguities, most algorithms make additional assumptions about the scene geometry (see Section 2.1).

Reasonable and commonly made assumptions are that the scene is piecewise smooth and order is preserved. These constraints on the depth map can be formulated into an objective function and directly optimized with various *global* methods.

- *Dynamic Programming* is one of the oldest optimization methods for stereo correspondence. The performance may reach state-of-the-art, if the vertical consistency between the scan lines is enforced (Lei et al., 2006).
- *Graph Cut* is based on the maximum-flow algorithm in graph theory (Kolmogorov & Zabih, 2001). The idea is to construct a specialized graph such that the minimum cut on the graph also minimizes the energy function. This optimization is applied to the whole image and not just one scan line.
- *Belief Propagation* formulates the stereo matching problem as a Markov network and solves it using Bayesian belief propagation to obtain the maximum a posteriori estimation (Klaus et al., 2006).

Unfortunately global methods are often time-consuming and memory exhausting. In addition most of them have problems preserving sharp discontinuities.

In this paper we propose an efficient method that is motivated by some recent work on Semi-Global Matching (Hirschmüller, 2005/2006). This approach uses a simple energy function (see Section 3) that implies piecewise smoothness but no ordering and gives promising experimental results in practice.

Our main contributions are investigations in different similarity measures (Section 2.2), the improvement by symmetric matching (Section 3.6), a hierarchical strategy (Section 4) and the efficient generalization of the stereo matching problem to trinocular reconstruction (Section 5). Finally, we present results for synthetic and for real images.

2. AREA-BASED MATCHING

Area-based matching is a widely used method for dense stereo correspondence. The similarity is computed statistically on the rectangular neighborhood (matching window) around the examined pixel. The algorithm searches at each pixel in reference image I_1 for maximum correlation in the horizontal image I_2 (and/or vertical image I_3) by shifting a small window pixel-by-pixel along the corresponding epipolar line.

2.1 Geometric Constraints

The time-consuming computation can be substantially simplified and accelerated by utilizing geometric constraints.

2.1.1 Disparity Limit

Assuming that the relative image orientation is known and that the smallest and highest displacement are roughly given, the search range $[d_{\min}, d_{\max}]$ of an image point in the reference view can be reduced to line segments in the corresponding images.

2.1.2 Normal Images

We assume that the epipolar lines of a horizontal image pair are parallel to the *x*-axis (*stereo normal case*) and those of a vertical image pair parallel to the *y*-axis. For the trinocular case using three images we recommend an L-shaped configuration. If the baselines of both stereo pairs have the same length, the horizontal and vertical displacements are identical.
Except for critical configurations, most images can be geometrically transformed so that the epipolar lines coincide with the same image rows and/or columns. This process is called *trinocular rectification* (Heinrichs & Rodehorst, 2006). After rectification it holds

$$\begin{aligned} I_1(x,y) &\approx I_2\big(x + D(x,y), y\big) \\ I_1(x,y) &\approx I_3\big(x, y + D(x,y)\big) \end{aligned} \tag{1}$$

where I_1, I_2, I_3 are rectified normal images, x is the column coordinate, y the image row coordinate and D is called disparity map. Please note that for differing image setups the horizontal and vertical disparities need to be scaled appropriately. The disparity at the current position (x,y) is inversely proportional to the depth of the scene.

2.2 Local Similarity Measures

In the following, popular similarity measures for calculating local image matching costs will be briefly described.

2.2.1 Difference Correlation (SSD, SAD)

A very simple but effective matching metric is the difference correlation. For each color pixel in $I = (R,G,B)$ the red, green and blue channels should be normalized by its intensity

$$I' = \frac{I}{R+G+B} \tag{2}$$

to compensate brightness and contrast variations. The color difference of two $n \times n$ image windows $a(x,y)$ and $b(x,y)$ with $N = 3n^2$ pixels can be defined by the *sum of squared differences* (SSD)

$$\rho_{SSD}(a,b) = \frac{1}{N}\sum_{i,j=1}^{n}\sum_{k=R,G,B}\big(a'_k(i,j) - b'_k(i,j)\big)^2 \tag{3}$$

or the slightly faster *sum of absolute differences* (SAD)

$$\rho_{SAD}(a,b) = \frac{1}{N}\sum_{i,j=1}^{n}\sum_{k=R,G,B}\big|a'_k(i,j) - b'_k(i,j)\big|. \tag{4}$$

Each measure should be normalized to a range between unity and zero, whereas unity means maximum similarity. The normalized correlation coefficient can be derived using

$$\rho'(a,b) = \frac{1}{1+\rho(a,b)}. \tag{5}$$

2.2.2 Cross-Correlation (NCC, MNCC)

The statistically based *normalized cross-correlation* (NCC) measures the linear relation between two image windows normalizing over all intensity changes.

$$\rho_{NCC}(a,b) = \frac{\sigma_{ab}}{\sqrt{\sigma_a^2 \cdot \sigma_b^2}}$$

$$\text{Covariance: } \sigma_{ab} = \frac{1}{N}\left(\sum_{i,j=1}^{n}\sum_{k=R,G,B} a_k(i,j)\cdot b_k(i,j)\right) - \bar{a}\cdot\bar{b}$$

$$\text{Variance: } \sigma_a^2 = \frac{1}{N}\left(\sum_{i,j=1}^{n}\sum_{k=R,G,B} a_k(i,j)^2\right) - \bar{a}^2 \tag{6}$$

$$\text{Mean: } \bar{a} = \frac{1}{N}\sum_{i,j=1}^{n}\sum_{k=R,G,B} a_k(i,j)$$

We precalculate the means $\bar{a}, \bar{b}$ and the means of squared intensities in order to significantly accelerate the computation. The definition of the *modified* NCC (MNCC)

$$\rho_{MNCC}(a,b) = \frac{2\cdot\sigma_{ab}}{\sigma_a^2 + \sigma_b^2} \tag{7}$$

handle homogeneous areas better by adding the two denominator variances instead of multiplying them (Egnal, 2000). Finally, we use

$$\rho'(a,b) = 0.5\cdot\big(\rho(a,b)+1\big). \tag{8}$$

to transform the correlation coefficient range to [0,1].

2.2.3 Mutual Information (MI)

Mutual information is a popular matching metric for images from airborne cameras (Hirschmüller, 2005) as well as inhomogeneous sensors like magnetic resonance imaging (MRI) and computerized axial tomography (CAT) scanners (Viola & Wells, 1997). A good description of MI can be found in (Kim, 2003; Pluim, 2000; Maes, 1997) and a comparison with MNCC in (Egnal, 2000). The major advantage of MI is its robustness against radiometric differences, i.e. non-Lambertian reflection properties and different gamma nonlinearities (Hirschmüller & Scharstein, 2007).
In (Hirschmüller, 2005) an improvement of pixel-wise matching costs based on MI is introduced. It requires a rough correspondence map, which can be computed hierarchically from the previous resolution. It suggests a random map for the lowest resolution to compute the MI for the next resolution. The intensity probabilities P_1 and P_2 are computed over all intensities i, j of all corresponding pixels $(\mathbf{x_1}, \mathbf{x_2})$ in images I_1, I_2.

$$\begin{aligned} P_1(i) &= \frac{1}{M}\sum^{\mathbf{x}_1} T\big[I_{1\mathbf{x}_1} = i\big] \\ P_2(j) &= \frac{1}{M}\sum^{\mathbf{x}_2} T\big[I_{1\mathbf{x}_2} = j\big] \end{aligned} \tag{9}$$

M is the number of correspondences and T is a Boolean function which returns 1 if the argument is true and 0 otherwise. The joint probability P_{ab} of the corresponding intensities is computed by

$$P_{12}(i,j) = \frac{1}{M}\sum^{\mathbf{x}_1\mathbf{x}_2} T\Big[\big(I_{1\mathbf{x}_1} = i\big) \wedge \big(I_{2\mathbf{x}_2} = j\big)\Big]. \tag{10}$$

Now the pixel-wise MI can be defined by

$$mi_{I_1,I_2}(i,j) = h_{I_1}(i) + h_{I_2}(j) - h_{I_1,I_2}(i,j) \tag{11}$$

where the entropy values h are defined by:

$$h_{I_1}(i) = -\frac{1}{M}\log\left(P_1(i) \otimes g(i)\right) \otimes g(i)$$
$$h_{I_2}(j) = -\frac{1}{M}\log\left(P_2(j) \otimes g(j)\right) \otimes g(j) \quad . \qquad (12)$$
$$h_{I_1,I_2}(i,j) = -\frac{1}{M}\log\left(P_{12}(i,j) \otimes g(i,j)\right) \otimes g(i,j)$$

The convolution $\otimes$ with a Gaussian g is applied for Parzen estimation (Kim 2003). This leads to a lookup table of 256×256 values for 8bit intensities. The MI matching costs can be computed by looking for the min-max values in this table:

$$\rho_{MI}(i,j) = \frac{mi_{I_1,I_2}(i,j) - mi_{\min}}{mi_{\max} - mi_{\min}}. \qquad (13)$$

For color images the three channels are computed separately and the MI costs for every pixel are calculated by the mean values of the three colors. This results in more stable matching than using only the joint intensity of all color channels.

3. SEMI-GLOBAL OPTIMIZATION

Generally, the calculation of local matching costs is ambiguous and a piecewise smoothness constraint must be added. In (Hirschmüller, 2005/2006) a very simple and effective method of finding minimal costs is proposed.

3.1 Energy Function

Semi-global matching (SGM) tries to determine a disparity map D such that the energy function

$$E(D) = \sum_{x,y \in I}\left(\left(1-\rho\left(a(x,y), b(x+D(x,y),y)\right)\right)\right.$$
$$+ Q_1 \sum_{i,j=-1}^{1} \mathrm{T}\left[\left|D(x,y) - D(x+i,y+j)\right| = 1\right] \qquad (14)$$
$$\left. + Q_2 \sum_{i,j=-1}^{1} \mathrm{T}\left[\left|D(x,y) - D(x+i,y+j)\right| > 1\right]\right) \quad \text{for} \quad i \neq j$$

is minimal. The first term calculates the sum of all *local matching costs* using the inverse correlation coefficient ρ (see Section 2.2) of the image windows a and b around the current position (x, y) and the related disparity in D. Explained intuitively, $E(D)$ accumulates the local matching costs with a small penalty $Q_1 = 0.05$ if the disparity varies by one from the neighboring disparities. If the disparity differs by more than 1, a high penalty $Q_2 \in [0.06, 0.8]$ is added. The actual value of Q_2 depends on the intensity gradient in the original image. Long gradients result in a low Q_2 while short gradients result in a high Q_2. This prevents depth changes in homogeneous regions. There are only two different penalties for the depth changes. First, Q_1 ensures that regions with a slightly changing depth are not penalized too hard. Second, if depth changes in the scene occur, the size of the discontinuity is not correlated to the penalty.

3.2 Optimization Strategy

Computing the minimum energy of $E(D)$ leads to NP-hard complexity, which is difficult to solve efficiently. Following (Hirschmüller, 2005), a linear approximation over possible disparity values $d \in [d_{\min}, d_{\max}]$ is suggested by summing the costs of several 1D-paths L towards the actual image location (x_1, y_1). A path L with $i = n \ldots 1$ steps is recursively defined as

$$\begin{aligned} L^n(x_n, y_n, d) &= 1 - \rho\left(a(x_n, y_n), b(x_n + d, y_n)\right) \\ L^i(x_i, y_i, d) &= 1 - \rho\left(a(x_i, y_i), b(x_i + d, y_i)\right) + \\ &\min\left(L^{i+1}(x_{i+1}, y_{i+1}, d-1) + Q_1,\right. \\ &\quad L^{i+1}(x_{i+1}, y_{i+1}, d), \\ &\quad L^{i+1}(x_{i+1}, y_{i+1}, d+1) + Q_1, \\ &\quad \left. L^{i+1}_{\min} + Q_2\right) \end{aligned} \qquad (15)$$

where the minimal costs over all disparities

$$L^i_{\min} = \min\left(L^i(x_i, y_i, d)\right) \qquad (16)$$

of the previous step are constant and must be computed only once. Changes of image positions from one recursive step to the next depend on the path direction $\mathbf{r} = (r_x, r_y)$ and the actual position:

$$\begin{aligned} x_i &= x_1 + (i-1) \cdot r_x \\ y_i &= y_1 + (i-1) \cdot r_y \end{aligned} \qquad (17)$$

Now, the final disparity map can be estimated using

$$D(x,y) = \min_d\left(\sum_{\mathbf{r}} L_{\mathbf{r}}(x,y,d)\right), \qquad (18)$$

where the number of accumulated paths should be ≥ 8.

3.3 Thresholding

We introduce a threshold for the local matching costs in order to penalize dissimilar candidates. If the correlation coefficient ρ is lower than a certain threshold, the local matching costs are set to a high constant value. If the minimal costs for the best matching candidate are higher than this value, the match is marked as invalid. Since MI provides statistical values which depend on the intensity distribution of the image a global threshold is not applicable to MI cost values.

3.4 Sub-pixel Matching

If the cost function has a local extreme at position $D(x, y)$, a sub-pixel disparity can be estimated by fitting a parabola curve to the correlation coefficients of the best match s_0 and its two neighbors s_{-1} and s_{+1}:

$$s_i = \rho_{SSD}\left(a(x,y), b(x + D(x,y) + i, y)\right) \qquad (19)$$

The interpolation is only valid for cubic functions like SSD (Hirschmüller, 2005). Therefore, s_i must be computed again if

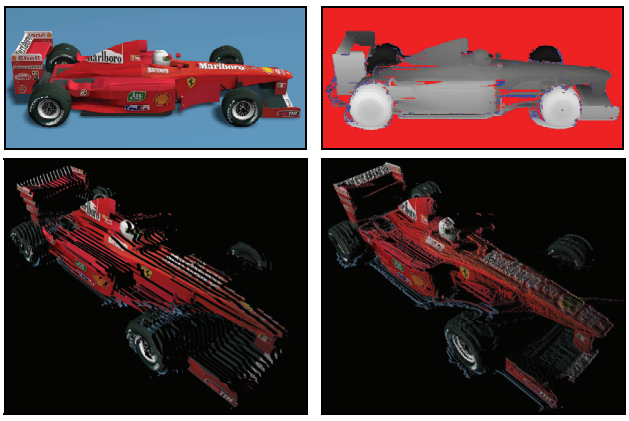


Figure 1. Sub-pixel improvements on the 3D reconstruction of a synthetically rendered Formula One car

different correlation functions were used before. Since the distance between the three coefficients is unity, they can be shifted symmetrically around the origin. The equation for sub-pixel disparity in x-direction is defined as

$$D_{sub}(x,y) = D(x,y) + \frac{s_{-1} - s_{+1}}{2 \cdot (s_{+1} + s_{-1} - 2 \cdot s_0)}. \quad (20)$$

3.5 Optimizing Memory Requirements

One disadvantage of SGM is the required space for all correlation values, which is needed to compute all non-horizontal trails $L_{\mathbf{r}}$. The memory for this buffer is $O(n^3)$ depending on the image width, height and disparity search range. We save memory by reducing the length of the trails.
Since the influence of previous L after a disparity discontinuity is very low, we need the complete path only for homogeneous areas. Except for trails along the epipolar lines, we limit the length of L to a small value (e.g. five). Therefore, the buffer size reduces to $O(n^2)$, which allows to process larger images.

3.6 Symmetric Matching

An important issue for image matching is the stability of the found correspondence. A correspondence can be unstable either due to an occlusion or because the image significance is very low, e.g. in homogeneous regions or periodic patterns. To enforce stability, we check the *left/right consistency* (LRC) of the bidirectional correspondence search.
A robust matching process should produce a unique result. On one hand, LRC detects most stereo errors and does not depend critically on thresholds. On the other hand, LRC does not report an error if the two matching directions mistakenly agree and it requires one extra matching process. Nevertheless, the computational expense is tolerable for many applications.
LRC leads to two disparity maps D_i, one for each image permutation. If the matched point in the second image points back to the original one in the first image

$$D_1(x,y) + D_2\left(x + D_1(x,y), y\right) \le 1, \quad (21)$$

the match is validated. Otherwise it is invalidated or in case of multi-image stereo matching, other permutations of the disparity map must verify this match. In addition, using the reverse direction guarantees that all matched points are one-to-one correspondences, because doubly matched points can verify only one location.

4. HIERARCHICAL APPROACH

This section describes the hierarchical approach of the matching process using image pyramids.

4.1 Building an Image Pyramid

Based on the original image resolution a number of reduced images are computed using a scale factor f_i. The search range can be scaled by f_i too, so that the computational complexity drops dramatically from the actual scale level to the next smaller one.

4.2 Hierarchical Processing

For the hierarchical approach the original images are resampled to different resolution layers. The layers are processed from the lowest resolution to the highest one. Only image points of the first layer have to be checked at every possible location within the search range.

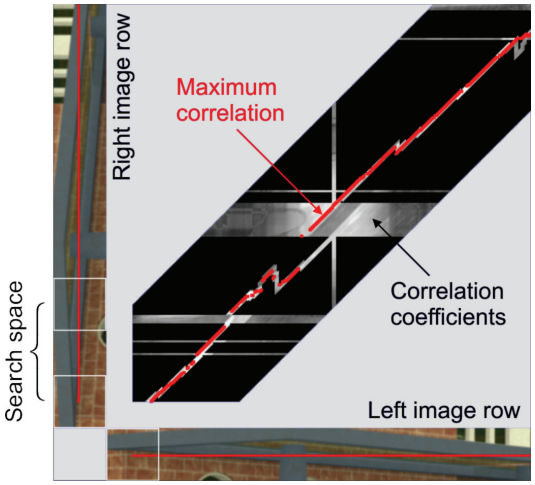


Figure 2. Sample of the reduced search space

To reduce the number of candidates in the succeeding layers, the potential information of the previous layer is used and refined. If displacement information from a previous layer is available, the number of candidates can be reduced by restricting the possible range. The valid candidates fulfill at least one of the following three criteria:

1. **Accuracy improvement:** The information from the previous layer has an accuracy of $\pm s \cdot f_i$, where s is the distance from one candidate to the next one and f_i is the scale factor from the previous layer to the actual one. Possible matches within this accuracy range must be checked.
2. **Unmatched points:** Points in the target image which are already matched in a previous layer should be excluded from further matching in order to avoid double matches.
3. **Edge preservation:** If points of the previous layer lie on a surface edge, the depth value of the associated points in the actual layer is bounded by the depths of the two neighboring points. It might happen that not all information is available.

Figure 2 illustrates this technique. The diagonal strip represents the search space of the original layer. Every column is the search space for a pixel position. The red line represents the selected correspondence. The thin diagonal stripe around the red line is the accuracy improvement from criterion 1. Vertical lines are unmatched positions in the previous layer. Therefore, the search space at these positions has to be analyzed completely to find possible new matches.
Horizontal lines represent unmatched points from criterion 2. The small vertical strips are caused by the edge preservation of criterion 3. Candidates in the black area are excluded by the hierarchical approach, which shows the efficiency of the proposed method. The search space is reduced to approximately 25% of the original size. After calculating the local costs for each candidate, our modified version of SGM calculates the best match for the given position.

5. TRINOCULAR MATCHING

In (Hirschmüller, 2005/2006), a multiple image technique is suggested which matches every image pair independently and selects the median of all disparities. This does not take advantage of the fact that every match of a single image pair leads to only one possible position in all remaining images. Thus, the position of a match candidate in a single image of the set is linked to certain computable positions in all other images

of the set. It is not necessary to search the whole range in every image pair independently. Since every candidate is linked to a specific candidate in all other images, the local matching costs of every pair between the candidate and the reference point can be simply averaged. This leads to two major advantages: First, the computing costs for all images are increasing only by a linear factor, while matching every pair would lead to a quadratic increase of the computational costs. Second, the cost function is more robust because local minima in a single set do not lead to outliers in the median selection technique. In addition, symmetric patterns parallel to one camera baseline are very unlike to be symmetric to the other baseline, too. Therefore, only the third image can provide sufficient information to find the right correspondence. More images are not integrated in this approach because the camera configuration gets too special and the common image region gets smaller with every additional image. We consider the trinocular case as a good compromise between stability and generality for most cases.

6. EXPERIMENTAL RESULTS

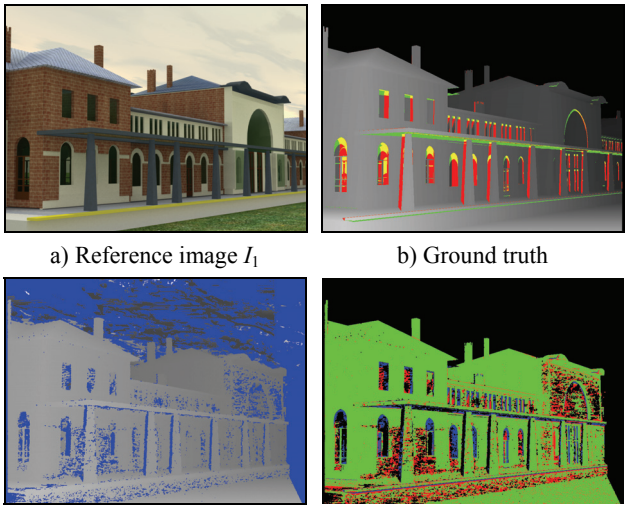
The proposed algorithm is tested on synthetic image triplets with ground truth and real image triplets from a regular 5MP digital camera.

6.1 Synthetic Images with Ground Truth

The rendered image in Figure 3 shows a train station with a resolution of 1280×1024 pixels. The image has many occlusions, homogenous regions and a lot of regular patterns like bricks. One of the three original images is presented in (a). Figure 3(b) illustrates the ground truth with horizontal occlusions (red), vertical occlusions (green) or both (yellow). The computed disparity map can be seen in (c), where blue pixels indicate inconsistencies or no match at all. The last picture (d) shows the color-coded evaluation: green pixels have an error less than one and false matches are marked red. Blue pixels represent false matches in occluded areas. Table 1 contain the quantitative evaluation results excluding the sky, which was not available in the ground truth data. The results describe the completeness, correctness and identified occlusions. The mean errors and standard deviations are given in the first two columns. The trifocal method is applied with bifocal fallback in regions which are seen only by two cameras. Most of the mismatched areas are due to interpolation of the dark and transparent windows. The unmatched areas result from homogeneous wall parts in the original image.

Method	Error [pix.]	Std. dev.	Match [%] Total	Match [%] Correct	Occ. [%]	Time [sec.]
SAD	1.17	2.72	79.9	78.8	39.9	89.5
SSD	0.95	2.40	80.6	83.2	44.3	98.8
NCC	1.06	2.73	80.7	83.2	42.5	93.6
MNCC	0.98	2.63	83.1	83.7	42.5	92.4
MI	0.78	2.13	83.5	85.8	50.0	86.5
One way	1.17	2.88	99.6	77.5	0.1	53.6
Symmetric	0.96	2.43	81.0	82.9	44.0	97.4
Bifocal	0.94	3.24	78.5	82.9	58.6	90.0
Trifocal	0.78	2.18	84.3	86.1	49.6	103.0

Table 1: Quantitative evaluation of the matching quality



a) Reference image I_1 b) Ground truth

c) Disparity map D d) Color-coded evaluation

Figure 3. Matching result using the synthetic image triplet with ground truth

6.2 Real Image Triplets

For real data, ground truth is often not available. The quality is therefore difficult to evaluate. However, some properties like perpendicularity or parallelism can be measured quite well. In Figure 4 the statue of King Friedrich I. of Prussia is reconstructed. Figure 4(a) shows the rectified reference image (2311×2200 pixel) and (b) the disparity map using a search range of 400 pixels. The illustrations 4(c) to 4(e) represent different views of the reconstructed 3D point cloud using parallel projection to enable the verification of perpendicularity and parallelism. Figure 4(f) shows a perspective projection to give an overview of the scene. The matching of this 5MP image triplet was completed in 12 minutes on a 2,4GHz dual core CPU.

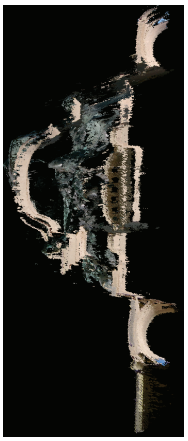
7. CONCLUSIONS

The proposed algorithm is a fast and effective adaptation of the SGM for multiple images. The hierarchical approach reduces the computational time by up to one quarter without any significant lost in accuracy. Additionally, we analyzed different cost functions. SAD and SSD perform very well on data with low noise and SSD outperforms SAD in every aspect, except speed. The use of the more robust (M)NCC is preferred for image data with higher noise and illumination changes, e.g. from a video sensor or scanned analog image, but generally has a weaker performance than MI. The optimized version of MNCC is even faster and detects more pixel than SAD/SSD with a minimal loss of accuracy. The advantage of MI is its ability to perform even in extreme circumstances where (M)NCC fails. Additionally, the pixel-wise matching of MI does not lead to blurred edges and it is the fastest technique. On the other hand this pixel-wise matching leads to some uncertainties in homogeneous regions, where window-based functions lead to more stable results. The matching of complex real images shows that stable matches are found in homogenous regions even if the path for SGM is limited to a short length. Combining the local costs of an image triplet to a single value stabilizes the matching especially in regions with repetitive patterns like bricks, grids or stripes.

a) Reference image I_1

b) Disparity map D

c) Textured point cloud



d) Top view

e) Side view

f) Perspective view

Figure 4. 3D reconstruction results on real image triplet of Friedrich I

REFERENCES

Egnal, G., 2000: "Mutual Information as a Stereo Correspondence Measure", Computer and Information Science MS-CIS-00-20, University of Pennsylvania, PA, USA, 8 p.

Heinrichs, M. and Rodehorst, V., 2006: "Trinocular Rectification for Various Camera Setups", Symp. of ISPRS Commission III - Photogrammetric Computer Vision PCV'06, Bonn, Germany, pp. 43-48.

Hirschmüller, H., 2005: "Accurate and Efficient Stereo Processing by Semi-Global Matching and Mutual Information", IEEE Conf. on Computer Vision and Pattern Recognition CVPR'05, Vol. 2, San Diego, CA, USA, pp. 807-814.

Hirschmüller, H., 2006: "Stereo Vision in Structured Environments by Consistent Semi-Global Matching", IEEE Conf. on Computer Vision and Pattern Recognition CVPR'06, Vol. 2, New York, NY, USA, pp. 2386-2393.

Hirschmüller, H and Scharstein, D., 2007: "Evaluation of Cost Functions for Stereo Matching", IEEE Conf. on Computer Vision and Pattern Recognition CVPR'07

Kim, J., Kolmogorov, V. and Zabih, R., 2003: "Visual Correspondence Using Energy Minimization and Mutual Information", IEEE Int. Conf. Computer Vision, 2003, Vol. 2, pp. 1033- 1040

Klaus, A., Sormann, M. and Karner, K., 2006: "Segment-based stereo matching using belief propagation and a self-adapting dissimilarity measure", IEEE Int. Conf. on Pattern Recognition ICPR'06, Hong Kong, China, pp. 15-18.

Kolmogorov, V. and Zabih, R., 2001: "Computing Visual Correspondence with Occlusions via Graph Cuts", Int. Conf. on Computer Vision ICCV'01, Vol. 2, Vancouver, Canada, pp. 508-515.

Lei, C., Selzer, J. and Yang, Y.H., 2006: "Region-Tree based Stereo using Dynamic Programming Optimization", IEEE Conf. on Computer Vision and Pattern Recognition CVPR'06, New York, NY, USA, pp. 2378-2385.

Maes, F., Collignon A., Vandermeulen D., Marchal G., and Suetens P., 1997: "Multimodality image registration by maximization of mutual information", IEEE Trans. Med. Imag., vol. 16, no. 2, pp. 187–198.

Pluim J.P.W., Antoine Maintz J.B., and Viergever, M. A., 2000: "Image Registration by Maximization of Combined Mutual Information and Gradient Information", IEEE Trans. Med. Imag., vol. 19, no. 8, pp. 809-814

Scharstein, D. and Szeliski, R., 2002: "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms", Int. Journal of Computer Vision, 47(1-3), pp. 7-42.

Seitz, S., Curless, B., Diebel J., Scharstein, D. and Szeliski R., 2006: "A comparison and evaluation of multi-view stereo reconstruction algorithms", IEEE Conf. on Computer Vision and Pattern Recognition CVPR'06, Vol. 1, New York, NY, USA, pp. 519-526.

Veksler, O., 2003: "Fast Variable Window for Stereo Correspondence using Integral Images," IEEE Conf. on Computer Vision and Pattern Recognition CVPR '03, Vol. 1, pp. 556-561.

Viola, P. and Wells, W.M., 1997: "Alignment by maximization of mutual information", International Journal of Computer Vision, 24(2), pp. 137-154.

3D SEGMENTATION OF UNSTRUCTURED POINT CLOUDS FOR BUILDING MODELLING

P. Dorninger [a,b] *, C. Nothegger [b]

[a] Vienna University of Technology, Institute of Photogrammetry and Remote Sensing, Gußhausstraße 27-29, 1040 Vienna, Austria - pdo@ipf.tuwien.ac.at

[b] Vienna University of Technology, Christian Doppler Laboratory for "Spatial Data from Laser Scanning and Remote Sensing", Gußhausstraße 27-29, 1040 Vienna, Austria - cn@ipf.tuwien.ac.at

KEY WORDS: Segmentation, Building Modelling, Point cloud, Matching, Laser scanning

ABSTRACT:

The determination of building models from unstructured three-dimensional point cloud data is often based on the piecewise intersection of planar faces. In general, the faces are determined automatically by a segmentation approach. To reduce the complexity of the problem and to increase the performance of the implementation, often a resampled (i.e. interpolated) grid representation is used instead of the original points. Such a data structure may be sufficient for low point densities, where steep surfaces (e.g. walls, steep roofs, etc.) are not well represented by the given data. However, in high resolution datasets with twenty or more points per square-meter acquired by airborne platforms, vertical faces become discernible making three-dimensional data processing adequate. In this article we present a three-dimensional point segmentation algorithm which is initialized by clustering in parameter space. To reduce the time complexity of this clustering, it is implemented sequentially resulting in a computation time which is dependent of the number of segments and almost independent of the number of points given. The method is tested against various datasets determined by image matching and laser scanning. The advantages of the three-dimensional approach against the restrictions introduced by 2.5D approaches are discussed.

1. INTRODUCTION

Numerous disciplines are dealing with the problem of geometric surface modelling from unstructured point clouds. These are, for example, computer vision, reverse engineering, or photogrammetry. In this article, we are studying segmentation for the purpose of building modelling from large, three-dimensional point clouds. Our ambition is the definition of a highly robust approach with respect to the method used for point sampling (i.e. image matching or laser scanning), measurement noise, scaling issues, and coordinate system definition implying three-dimensional applicability. Additionally, a low time complexity is required in order to process huge datasets.

The typical workflow from data (point) acquisition towards the final, geometrical surface model (e.g. triangulation or constructive solid geometry) consists of the following steps: Calibration (i.e. elimination of systematic effects), registration (in photogrammetric context often referred to as orientation), minimization of the influence of measurement noise (referred to as filtering or smoothing), and finally, surface modelling. Systematic errors should be eliminated by instrument calibration prior to the measurement (i.e. by the manufacturer) or a posteriori during data processing (if the behaviour of the systematic errors is known) or by self-calibration of the system (e.g. Lichti & Franke (2005) for terrestrial and Kager (2004) for airborne laser scanners). In this paper, we do not discuss calibration and registration, while the other two steps, namely random error (i.e. noise) elimination and modelling are considered. The starting point is therefore a set of points in three-dimensional Cartesian space without systematic but with random errors.

By means of segmentation, points with similar attributes are aggregated, thus, introducing an abstraction layer. This simplifies subsequent decision making and data analysis, as compound objects defined by multiple points are represented by segments. Higher level objects are easier to deal with compared to original point cloud handling, simplifying many applications such as object detection, recognition or reconstruction. Additionally, the data volume decreases significantly when handling the segments' parameters only instead of the original point cloud. Thus, the computation time of subsequently applied algorithms may decrease.

In general, the workflow for building reconstruction from point cloud data can be separated in three steps: *Building detection*, *determination of planar faces* representing individual roof planes and finally *model generation*. Point cloud segmentation as described within this article provides planar faces which can be used for the determination of building models thus, supporting the latter two steps. The first step, namely building detection, is not discussed in the following, as we assume to initiate building modelling from given, two-dimensional boundary polygons for each building. Numerous approaches for the automated determination of building boundaries can be found in literature. For example, Rottensteiner et al. (2004) and Haala & Brenner (1999) describe methods integrating laser scanning and multi-spectral image data, Maas & Vosselman (1999) introduced a triangle-mesh based approach demonstrated

* Corresponding author.

on laser scanning data and, Zebedin et al. (2006) describe a method based on matched points. Brenner (2003) gives an overview on automatic and semi-automatic systems for building reconstruction from image and laser scanner data.

Many approaches for the determination of planar faces for roof modelling from point clouds acquired from airborne platforms can be found in literature (e.g. Maas & Vosselman (1999), Lee & Schenk (2002), Filin (2004), or Vosselman & Dijkman (2001)). To reduce the complexity of the problem and to increase the performance of the implementation, often a resampled (i.e. interpolated) grid representation is used instead of the original points (e.g. Alharty & Berthel (2004), Rottensteiner et al. (2005)). Such a data structure may be sufficient for low point densities, where steep surfaces (e.g. walls, steep roofs, etc.) are not well represented by the given data. However, in high resolution datasets with twenty or more points per square-meter acquired by airborne platforms, vertical faces become discernible making three-dimensional data processing adequate.

Characteristics and acquisition of unstructured point clouds, basic characteristics of segmentation and the computation of highly robust, local regression planes are introduced in Section 2. Our segmentation algorithm is presented in Section 3, starting with a comparison to a grid-based approach our work was inspired by. In Section 4 results derived from image matching and laser scanning point clouds are presented. In Section 5, we discuss the characteristics of 2.5D and 3D segmentation and analyze the performance of our approach.

2. RELATED WORK

So far, we stated the applicability of our segmentation approach on unstructured point clouds, but without defining our understanding of the latter terminus. This is done in Section 2.1, followed by the definition of our understanding of segmentation in order to avoid misunderstandings (Section 2.2). The determination of highly robust, local regression planes for every given point is of crucial importance for our segmentation algorithm. Therefore in Section 2.3, the basic characteristics of the method used are presented.

2.1 Acquisition of Unstructured Point Clouds

We define an unstructured point cloud as a set of points, obtained from a random sampling of the object's surface in a three-dimensional way. No additional definition according to the reference frame (e.g. a reference direction) or the metric (e.g. scaling) are made. Numerous acquisition methods for the determination of such point clouds do exist. As we are studying the reconstruction of buildings in this article, we are concentrating on data acquisition methods from airborne (i.e. helicopter or airplane) and satellite borne platforms. The point determination is performed either directly through polar single point measurement (e.g. Lidar – light detection and ranging – often referred to as laser scanning or synthetic aperture radar (SAR) often performed by satellites) or indirectly (e.g. stereoscopic image matching). However, all these techniques deliver unstructured point clouds according to our definition. The segmentation results presented in this paper were derived from point clouds determined by image matching (IM) and airborne laser scanning (ALS), both acquired from airborne platforms.

2.2 Segmentation of Point Clouds

Segmentation refers to the task of partitioning a set of measurements in the 3D object space (point cloud) into smaller, coherent and connected subsets. These subsets should be 'meaningful' in the sense that they correspond to objects of interest (e.g. roofs, trees, power cables) or parts thereof (roof planes). Often, the segments are assumed to take the form of simple geometric primitives (e.g. planar patches). In this case segmentation and extraction of the primitives are typically performed simultaneously, rather than sequentially. We assume that the resulting segments $\boldsymbol{R}_i$ (a sub-set of points) of $\boldsymbol{R}$ (all N points) meet the following requirements (Hoover et al., 1996):

1. $\bigcup_{i=1}^{N} \mathbf{R}_i = \mathbf{R}$ (requires the definition of a rejection-segment)
2. $\mathbf{R}_i \bigcap \mathbf{R}_j = \{\}$, for $i \neq j \wedge 1 \leq i, j \leq N$
3. $\mathbf{R}_i$ is a connected component in the object space, for $1 \leq i \leq N$
4. $P(\mathbf{R}_i)$ == *true* for some coherence predicate P, $1 \leq i \leq N$
5. $P(\mathbf{R}_i \bigcup \mathbf{R}_j)$ == false for adjacent $\mathbf{R}_i$, $\mathbf{R}_j$ with $i \neq j \wedge 1 \leq i, j \leq N$

(1)-(2) state that the resulting segmentation is a partitioning of the original point cloud, i.e., a decomposition into disjunct sub-sets. The connectivity requirement (3), which is straightforward in image processing (since there, connectivity is defined in terms of the 4- or 8- neighbourhood on the pixel grid), requires a definition of neighbourhood (topology) for unstructured point clouds. This could be done, for example, by considering two points as neighbours if they are connected by an edge in the Delaunay triangulation; however, obtaining the Delaunay triangulation is computationally costly (in particular, for large point clouds). Hence, we use a criterion based on the Euclidean metric: $\boldsymbol{R}_i$ is a connected component if the distance of any point to its nearest neighbour in the segment is below a given threshold. The coherence predicate in (4) simply states that the points must lie on (or near) the same instance of a parametric primitive (in our case, a planar patch). Finally, (5) requires that the points belonging to two adjacent segments lie on two separate planes (otherwise, the segments are merged).

2.3 Robust Local Regression Planes

It can be assumed that the normal vectors (short: normals) of local regression planes of points belonging to a segment (i.e. plane) are almost identical. According to the given task, several requirements for the plane fitting algorithm are introduced. These are the capability to handle a high noise level, robustness at sharp surface features (e.g. planes intersecting at a common edge) and mode seeking behaviour due to the uncertainty of the distribution of the measurement errors which might not be symmetric.

Commonly used methods for regression plane estimation are, for example, moving least squares (e.g. Levin, 1998) or iteratively reweighted least squares (e.g. Hoaglin et al., 1983). Unfortunately, both methods are sensitive against non-symmetric point density. Additionally, the former is not robust against noise while the latter may handle noise well, but the result can be influenced by a single lever point (breakdown point of zero percent). The Random Sampling Consensus Algorithm (RANSAC) (Fischler & Bolles, 1981) is often

referred to in literature. But RANSAC does not consider statistics, behaves slow and has an insufficient breakdown point with respect to our task. Furthermore, RANSAC uses the object space (i.e. point position) only and cannot take additional parameter (e.g. local normals) into account. Therefore, we developed a method which fulfils the stated requirements as much as possible while achieving acceptable computation time as well. It is based on the Fast Minimum Covariance Determinant (FMCD) approach, described by Rousseuw & van Driessen (1999) for application in the field of data mining. Filin (2004) investigated a prior approach described by Rousseuw in the field of ALS-point classification and indicated it as too slow for that application. By introducing heuristics, the performance of this approach is sufficient for the given task.

3. METHODOLOGY

Our research has been inspired by a grid-based (in the following referred to as 2.5D) approach described by Pottmann & Wallner (1999). The method's application to building modelling was first described by Peternell & Pottmann (2002). We compare this method to our algorithm which is applicable on three-dimensional point clouds and behaves independent from the coordinate system definition. By means of a sequential implementation of the clustering algorithm used, the time complexity of the three-dimensional approach behaves better than the 2.5D-implementation by Pottmann et al.

3.1 Distance Measure between Planar Faces

Hierarchical clustering in the four-dimensional feature space defined by the local regression planes of the given points (Section 2.3) requires an appropriate distance measure. A plane in three-dimensional Cartesian coordinates is defined by

$$0 = a_0 + a_1 x + a_2 y + a_3 z \quad (1)$$

with a_1, a_2, a_3 representing the local unit normal vector of the plane and a_0, the normal distance of the plane to the origin of the coordinate system. To determine a three-dimensional measure of distance between two planes $A = (a_0, a_1, a_2, a_3)$ and $B = (b_0, b_1, b_2, b_3)$, we define the distance d over an area of interest Γ in the following way:

$$d_\Gamma(A,B)^2 = \int_\Gamma (c_0 + c_1 x + c_2 y + c_3 z)^2 \, dxdydz \quad (2)$$

where $c_n = a_n - b_n$

The integrand of (2) represents the squared difference of the orthogonal distances from a point to the two planes A and B. Thus, the integral over all squared distances within Γ can be interpreted as a mean squared distance between these planes, if it is normalized by the volume of Γ. As we consider squared differences, greater differences become a higher weight. But this does not matter, as we aim at the determination of similar planes (i.e. with small differences). The application of this measure in 2.5D space over a planimetric, rectangular Γ (as described by Peternell & Pottmann (2002)) obviously is dependent on the definition of the reference direction (in general z). The distance measures d for two planes (A, B) enclosing an angle of one degree, intersecting within the system's origin ($a_0 = b_0 = 0$) and rotating around the y axis are shown in Figure 1 (chain dotted line). The behaviour of a three-dimensional d (Eq. 2) with a bounding box (dashed line) and a bounding sphere (continuous line) are superimposed. While for the 2.5D case the distance measure becomes useless for inclination angles between 45° and 135° respectively 225° and 315° (the measure increases at least by a factor of two and becomes infinite in the worst case occurring if a plane is vertical), the box-based 3D measure is almost and the sphere-based measure really is constant.

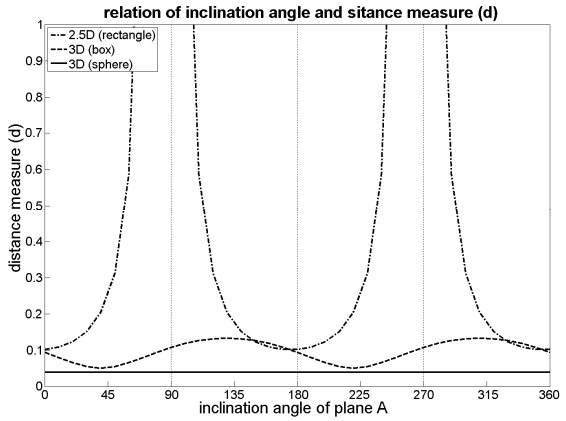


Figure 1. Relation of inclination angle against the horizontal xy-plane and different distance measures d demonstrated by two planes (inclined by 1°) rotated around the y-axis.

3.2 Implementation

A global clustering approach requires the evaluation of the distance measure d for each pair of given points. This results in a time complexity of $O(n^2)$ with n representing the number of points given. Considering datasets with millions of points, this is not feasible. Therefore, we replaced the determination of the distance matrix by a sequential evaluation of the given plane parameters. This reduces the time complexity to $O(m)$ with m representing the number of detectable planar faces.

The clustering in feature space is used to determine seed-clusters. For each seed-cluster, a region growing is performed. This is done by assigning points to the plane segment if they are within a normal distance band around the seed-cluster's plane and if the distance d between the points' normals and the seed-cluster's regression plane is smaller than a predefined threshold. If no more seed-cluster can be determined, the remaining points are assigned to the rejection class. Afterwards, a connected component analysis in object space and a merging of similar components considering feature and object space are performed.

Figure 2 shows the segmentation of a building from an unstructured point cloud. The distance measure used in feature space, the normal distance threshold in object space and the standard-deviation (σ) used to determine outliers during robust plane estimation have been set to 0.1m. Figure 2(a) and (b) show the determination of segment 1, (c) and (d) of segment 2, and (e) and (f) the final segmentation of the building including vertical walls. (a) and (c) show the points of the seed-clusters (large black dots). Points accepted in object and feature space are shown in light grey. Subsequently, a robust regression plane is fitted through these points using a 3σ threshold for outlier detection. The results of the plane fitting are shown in (b) and (d). Accepted points are shown in light grey; outliers in dark grey. The small light grey dots in (c) and (d) represent points already assigned to segment 1; black points have not been used so far.

Points accepted in object space but neglected in feature space are shown in (a) and (c) in dark grey. Algorithms taking into account object space only (e.g. RANAC) might use these points as support for the final plane determination. This may introduce problems for example along the intersection of two planes. As demonstrated by (c), numerous points along the gable of the roof were assigned to the plane defining the dormer. Without the normals' based decision criterion, these points might have been assigned to the dormer plane.

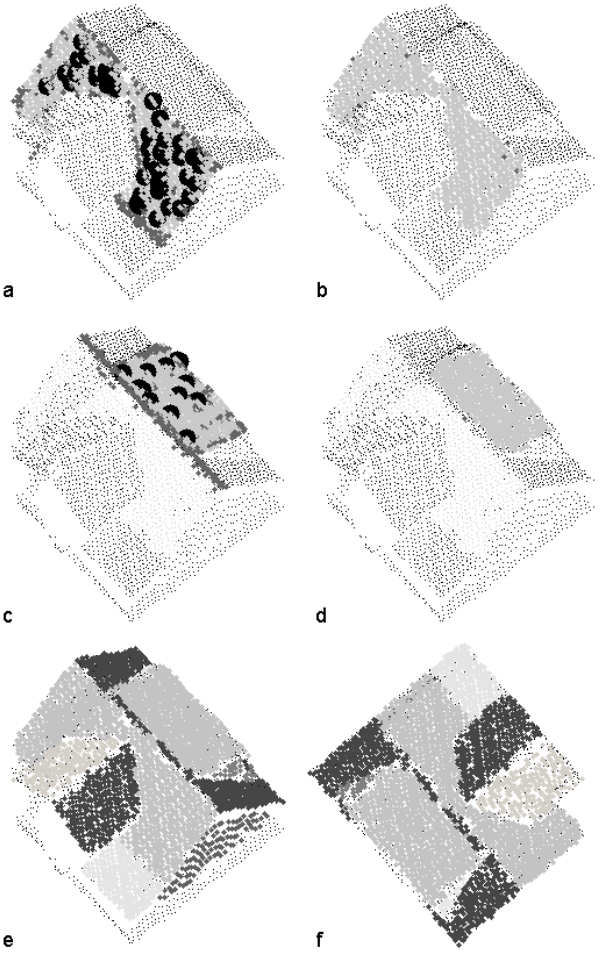


Figure 2. Segmentation of a building. (a) and (b) show the determination of segment 1; (c) and (d) of segment 2. (e) and (f) show the final segmentation including vertical walls.

4. APPLICATIONS

As mentioned in Section 2.1, digital image matching and airborne laser scanning are well suited for the determination of unstructured point clouds of a landscape. In the following, we present building models derived from segmentation results which were determined by the described segmentation algorithm. The building models were generated by piece-wise intersection of planar faces. Similar approaches can be found in literature (e.g. Maas & Vosselman (1999), Park et al. (2006)). In general, roof landscape modelling from airborne point clouds results in 2.5D descriptions of real world objects. This introduces restrictions with respect to the achievable level of detail. For example, roof overhangs can not be considered and vertical walls are defined at the eaves. Due to the high point density of the given dataset (~20 points per square-meter), numerous points at vertical façades have been determined (confer Figure 2), allowing to reconstruct the real position of the vertical walls (i.e. a planar representation of façades). Subsequently, roof overhangs can be modelled properly. The high point density was enabled by a helicopter borne platform at a low flight level. The large opening angle (±22.5°) of the scanner used (Riegl LMS-Z560i) (http://www.riegl.com) and multiple overlaps allowed the acquisition of the façade points.

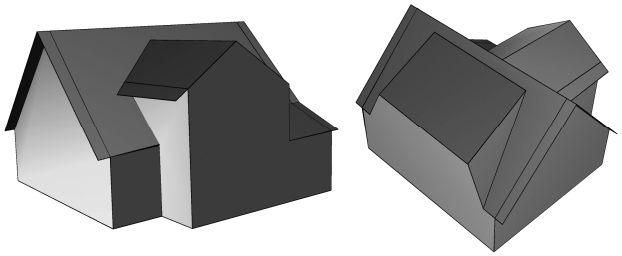


Figure 3. Detailed model of a building determined by piece-wise intersection of planar faces based on the segmentation shown in Figure 2.

The second example was derived from a matched point cloud. The images have been acquired by a *Vexcel UltracamD* (http://www.vexcel.com). The matching was done using the software *Match-T* from *Inpho* (http://www.inpho.de). Compared to the ALS data noise level is higher. Thus, a larger neighbourhood (i.e. more points) was used to determine the local regression planes. Hence, the minimum size of discernible faces is larger compared to the processed ALS data. Nevertheless, the segmentation and the building model derived from the matching points appear plausible (Figure 4), demonstrating the robustness of the segmentation with respect to the data acquisition method used.

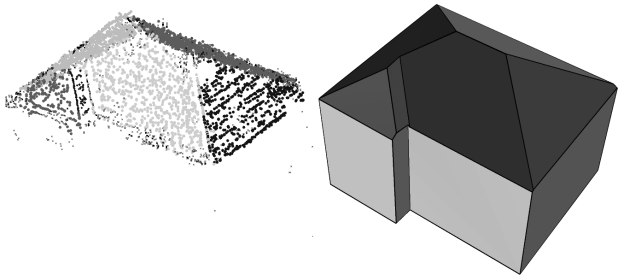


Figure 4. Segmentation of a matched point cloud (left) and building model (right).

5. DISCUSSION

Cadastral information is normally based on the polygonal boundaries of buildings at the terrain level. In most cases, this 2.5D polygon differs from the normal projection of the eaves on the terrain as most roofs have an overhang. 2.5D building modelling approaches cannot cope with this, making the resulting models unsuitable for gathering or updating cadastral information. However, we demonstrated that the determination of the real position of vertical walls from high density ALS data is possible (Figure 2) using a three-dimensional segmentation approach. The integration of this information in the modelling process is shown in Figure 3. In this example, it was not possible to determine the real position of all walls. In Figure 5 points on vertical walls (white lines) acquired by airborne laser scanning are shown. Obviously, the distribution of these points depends on the flight configuration. Thus, it can be influenced by an adequate flight planning.

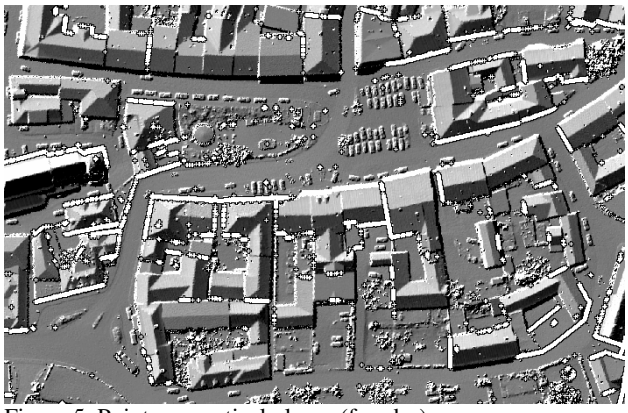


Figure 5. Points on vertical planes (façades).

To demonstrate the effect of data resampling (i.e. grid interpolation) on the segmentation of the point cloud, a regular grid (25 cm) was derived from the dataset shown in Figure 2 by means of linear prediction (σ = 5 cm). Figure 6 shows the segmentation of this grid. Several erroneous segments were found along the intersection lines of touching segments.

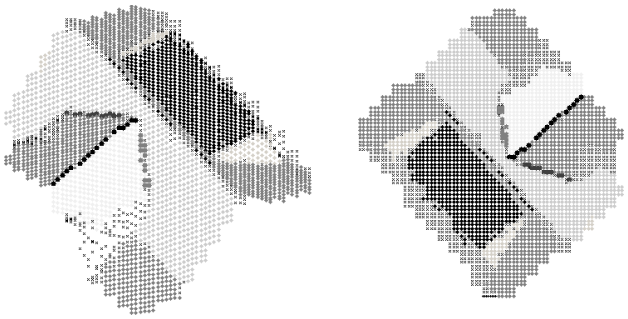


Figure 6. Segmentation of a 25 cm grid derived from the original point cloud. The smoothing of the grid interpolation causes erroneous segments.

Figure 7 shows the differences of the exposure and slope determined from the parameters of the planar faces derived from the original and the resampled point cloud (confer Figure 2 and 6). The differences of the planes 1 to 6 are not significant and seem to be randomly distributed, while the planes 7 and 8 show significant differences. These are the smallest faces (i.e. the are supported by the least number of points) representing the triangular faces of the huge dormer. As no reference data is available, we are not able to decide which parameters are the correct ones.

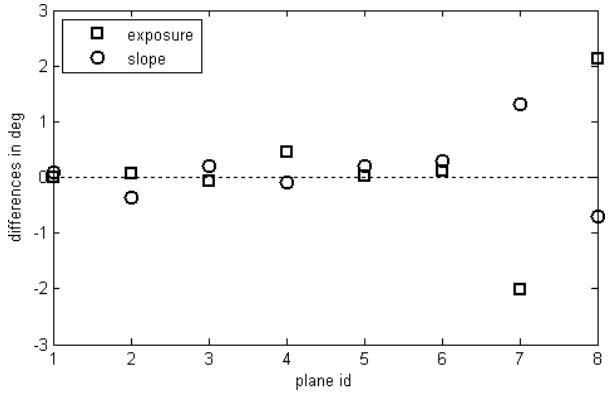


Figure 7. Differences of exposure and slope determined from the parameters of the planes derived from the original and the resampled point cloud.

The segmentation of buildings as presented in this paper including connected components analysis takes about a second. To investigate the time performance behaviour of the segmentation in more detail (not considering the connected components analysis), synthetically generated datasets were used. These datasets are three orthogonal prisms with rotation axes that are collinear with the axes of the coordinate system and thus, intersecting at the origin of the coordinate system. Every prism consists of n rectangular planes defined by m points. A normally distributed random noise was added in normal direction to each plane. Figure 8 shows such an object defined by 12 planar faces per prism. The original object is shown left while the segmentation result is shown right.

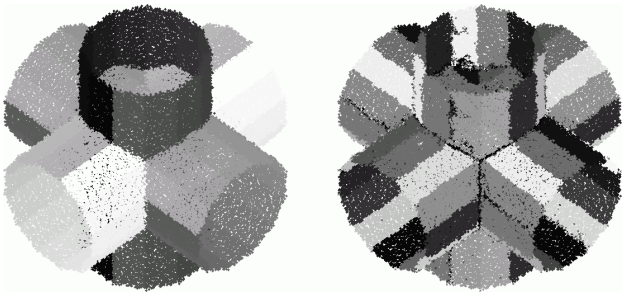


Figure 8. A synthetically generated dataset consisting of 36 planar faces, aligned in numerous directions with respect to the coordinate system. The left figure shows the original planar faces. The segmentation is shown right.

Increasing the number of object points defining similar objects (i.e. equal extension, 12 faces per prism, different point spacing (n=constant; m=increased)) by a factor of 150 (720 to 114,156 points) increases the computation time of the segmentation by a factor of 4 only (1.3 to 6.1 sec.). The computation time for the segmentation is shown as solid line in Figure 9 (left). Figure 9 (right) shows the relation of an increasing number of faces while the number of points is almost constant (n=increased; $m \cdot n$=30,000). In this example, the number of faces is increased by a factor of 6 (9 to 54 faces) while the computation time increases by a factor of 3 (1.1 to 3.2 sec.). The time for normal estimation is shown as dotted line in both diagrams. It scales almost linear by the number of given points.

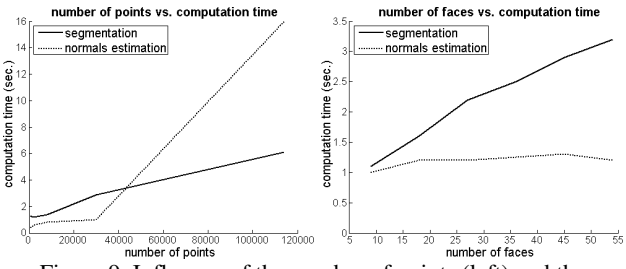


Figure 9. Influence of the number of points (left) and the number of faces (right) on the computation time.

6. CONCLUSIONS

Many building modelling approaches based on unstructured point clouds acquired from airborne platforms (i.e. airplane or helicopter borne) generate 2.5D representations of the real world objects for different reasons. On the one hand, most available datasets do not provide enough information to model vertical structures (i.e. façades) or even roof overhangs in a proper manner. On the other hand, approaches based on gridded

data structures are commonly used as they are likely to increase the performance.

However, currently available data acquisitions systems allow for the determination of numerous structures on vertical walls of buildings and similar objects by means of appropriate flight conditions (low flight level, great opening angle, multiple overlaps, …). Furthermore, the restrictions of 2.5D can be overcome by introducing real three-dimensional point cloud segmentation and subsequent modelling approaches. We presented such a segmentation approach in detail which was originally inspired by a 2.5D approach. By means of a smart implementation of the algorithm, the time complexity of the problem was reduced, thus resulting in lower computation time with respect to the original, 2.5D approach.

The segmentation algorithm was tested on point clouds acquired by laser scanning and image matching. The results, which appear reliable, were used for subsequent determination of building models. This was done using an approach based on piece-wise intersection of touching plane segments.

The presented comparison of 2.5D and 3D data processing demonstrated that the determined plane parameter may differ significantly, especially for small faces with less support points. These results should be analyzed further using reliable ground truth measurements.

REFERENCES

Alharthy, A., Berthel, J., 2004. Detailed building reconstruction from airborne laser data using a moving surface method. In: *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Istanbul, Turkey, Vol. XXXV, Part B3, pp. 213-218.

Brenner, C., 2003: Building Reconstruction from Laser Scanning and Images. Proc. ITC Workshop on Data Quality in Earth Observation Techniques, Enschede, The Netherlands, November 2003.

Filin, S., 2004. Surface classification from airborne laser scanning data. *Computers & Geosciences*, 30, pp. 1033-1041.

Fischler, M.A., Bolles, R.C., 1981. Random sample consensus: a paradigm for model fitting with application to image analysis and automated cartography. *Communications of the Association Computing Machinery*, 24, pp. 381-395.

Haala, N., Brenner, C., 1999. Extraction of buildings and trees in urban environments. *ISPRS Journal of Photogrammetry & Remote Sensing*, 54 (1999), pp. 130-137.

Hoaglin, D.C., Mosteller, F., Tukey, J.W., 1983. Understanding Robust and Exploratory Data Analysis, *Wiley Series in Probability and Statistics*, John Wiley & Sons Inc.

Hoover, A., Jean-Baptiste, G., Jiang, X., Flynn, P.J., Bunke, H., Goldgof, D., Bowyer, K., Eggert, D., Fitzgibbon, A., Fisher, R., 1996: An Experimental Comparison of Range Image Segmentation Algorithms, IEEE Transactions on Pattern Analysis and Machine Intelligence 18 (7), pp. 673-689.

Kager, H., 2004: Discrepancies Between Overlapping Laser Scanning Strips- Simultaneous Fitting of Aerial Laser Scanner Strips, In: "International Society for Photogrammetry and Remote Sensing XXth Congress", O. Altan (ed.), Vol XXXV, Part B/1 (2004), Istanbul, pp. 555-560.

Lee, I., Schenk, T., 2002: Perceptual Organization of 3D Surface Points, In: Proceedings of ISPRS Commission III, Symposium 2002, Graz, Austria.

Levin, D., 1998. The approximation of moving least-squares. *Mathematics of Computation*, 67, pp. 1517-1531.

Lichti, D.D., Franke, J., 2005: Self-calibration of the iQsun 880 laser scanner. In: Proceedings of 7th conference on optical 3-D Measurement Techniques, Vienna, Grün/Kahmen (Eds.), pp. 112–122.

Maas, H.-G., Vosselman, G., 1999. Two algorithms for extracting building models from raw laser altimetry data. *ISPRS Journal of Photogrammetry & Remote Sensing*, 54 (1999). pp. 153-163.

Park, J., Lee, I., Choi, Y., Lee, J.L., 2006. Automatic extraction of large complex buildings using lidar data and digital maps. In: *Proceedings of ISPRS Symposium of Commission III, WG III/3*, Bonn, Germany.

Peternell, M., Pottmann, H., 2002. Approximation in the space of planes - Applications to geometric modeling and reverse engineering. *Rev.R.Aca.Cien.Seria A.Mat (RACSAM)*, 96(2), pp. 243-256.

Pottmann, H., Wallner, J., 1999. Approximation algorithms for developable surfaces. *Computer Aided Geometric Design*, 16(6), pp. 539-556.

Rottensteiner, F., Trinder, J., Clode, S., Kubik, K., 2004: Building Detection by Dempster-Shafer Fusion of LIDAR Data and Multispectral Aerial Imagery. Proc. 17th International Conference on Pattern Recognition, Vol. 2, pp. 339-342.

Rottensteiner, F., Trinder, J., Clode, S., Kubik, K., 2005: Automated delineation of roof planes from lidar data. Proc. ISPRS WG III/3, III/4, V/3 Workshop "Laser scanning 2005", Enschede, the Netherlands, September 12-14, 2005.

Rousseeuw, P. J., van Driessen, K., 1999. A fast algorithm for the minimum covariance determinant estimator. *Technometrics*, 41, pp. 212-223.

Vosselman, G., Dijkman, S., 2001. 3D Building Model Reconstruction from Point Clouds and Ground Plans. in: International Archives of Photogrammetry and Remote Sensing, Volume XXXIV-3/W4, Annapolis, pp. 37-43.

Zebedin, L., Klaus, A., Gruber-Geymayer, B., Karner, K., 2006: Towards 3D map generation from digital aerial images. ISPRS Journal of Photogrammetry & Remote Sensing, Vol. 60, pp. 413-427.

ACKNOWLEDGEMENTS

This work was supported by *Vermessung Schmid* (http://www.geoserve.co.at) and funded by the *Christian Doppler Research Association* (http://www.cdg.ac.at). The ALS datasets have been provided by *Vermessung Schmid* in cooperation with *Bewag Geoservice* (http://www.geoservice.at). The matched point cloud was provided by *Meixner Vermessung ZT GmbH* (http://www.meixner.com).

2D BUILDING CHANGE DETECTION FROM HIGH RESOLUTION AERIAL IMAGES AND CORRELATION DIGITAL SURFACE MODELS

Nicolas Champion

MATIS Laboratory, Institut Géographique National
2, Avenue Pasteur. 94165 SAINT-MANDE Cedex - FRANCE
Firstname.Lastname@ign.fr

Commission IV/3

KEY WORDS: Change Detection, Buildings, 2D topographic database, DSM, DTM, Updating

ABSTRACT:

Updating 2D databases has become a crucial issue in most mapping agencies. Such a work traditionally starts out with a change detection phase. A subsequent update phase is then carried out to register changes in the up-to-date database. The first phase is by far the most costly and plodding, as it has until now required field or visual inspection (of orthophotos). The main goal of this paper is to present a new method for detecting changes in the building layer of a 2D cadastral database. This method aims at giving potential changes to a human operator for subsequent validation and update registration. In this paper, we propose a new workflow for the change detection process, by splitting it into 2 separate steps. The first step consists in verifying automatically buildings through a hypothesize-and-verify process: the initial description of the database is used to guide the change detection process. The second step consists in extracting new buildings from geometric considerations. In this paper, the method is tested and assessed in a densely built-up area. A specific methodology is firstly employed to estimate the best parameters to use in the system and also to characterise its performance. Results are secondly assessed and show the high potential of our system in such a context.

1 INTRODUCTION

In the past few years, most 2D topographic databases have been completed in developed countries. Most efforts in mapping agencies are now dedicated to the revision / update of such databases. Such a task is particularly time-consuming and tedious, as it is generally carried out manually, by visual inspection of an orthophoto to detect objects to be revised. Therefore, such a work is highly costly: (Steinnocher and Kressler, 2006) estimates that it can cost up to 40% of the whole cost entailed when generating the topographic database from scratch. Semi-automatic procedures also need to be developed. Such procedures are commonly split into 2 steps: in a first change detection step, input data (high resolution aerial or satellite images, laser scanning . . .) are given to an algorithm that then determines focalisation areas where a possible change has taken place; in a second update step, these focalisation areas are given to a human operator for validation and registration. Among all the objects contained in a topographic database, we will focus here on buildings, which play a crucial role in an increasing number of applications, especially in the production of 3D City models (Taillandier, 2005). As shown in Figure 1, a building change can be of several types: destroyed buildings as well as new buildings are obviously changes. Moreover, the modification or the deletion of a part of a building (caused either by a human activity or planimetric inaccuracies in the initial geospatial database) is a change and must be detected by the algorithm.

(1) **(2)** **(3)**

Figure 1: Update Problems. **(1) :** Destroyed buildings. **(2) :** New buildings. **(3) :** Planimetric inaccuracies.

1.1 Related works

Since the advent of high-quality digital aerial images and laser scanning, many researches have been carried out to detect changes in the building layer of a 2D digital database.
In Germany, the WIPKA project[1] has been launched to automatically verify a topographic database. Within this scope, a knowledge-driven approach (Busch et al., 2004) is proposed to verify area objects (settlement, industrial areas, cropland . . .) contained in the database: hints are collected from images for each object to be verified (top-down phase) and used to accept or reject the object (bottom-up phase). This study is all the more interesting as it shows that a change detection process is always data-dependent: the specifications of the database to be checked need to be taken into account before building the system design.
In (Steinnocher and Kressler, 2006), an object-based classification is implemented to support the update of existing land use databases. Orthophotos are firstly segmented and each object is classified into 4 classes (identical, plausible, questionable, new) by means of so-called evaluation matrices. Results are promising but show the difficulty to deal with objects assigned to a given class for legal and not physical reasons, typically administrative sections. (Matikainen et al., 2004) proposes a similar object-oriented classification based on laser scanning and digital aerial images but focus on the building theme only. Results are all the best as buildings are big.
(Olsen and Knudsen, 2005) proposes a hierarchical method: a coarse building mask is firstly computed from initial DSM and NIR images, and then refined with respect to object size and form. Eventually, it is compared to the database to update. The authors outline the necessity to compute a DTM from the DSM, as height features are very useful when detecting buildings.
Eventually, other methods exist in literature and could be used in a change detection process, even if they are not originally built for that purpose. For example, the method described in

[1] http://www.ipi.uni-hannover.de/html/forschung/laufend/wipka/wipka.htm, Last Visited: 2007-3-31

Figure 2: Semantic differences: one building can be split into 2 or more objects in the database.

(Rottensteiner et al., 2005) deals with building detection from aerial images and laser scanning. It is based on a hierarchical Dempster-Shafer classification. This method is being adapted to enter the framework of a new EuroSDR Building Change Detection project.

1.2 General Scheme

The main goal of this paper is to present a new method for change detection in the building layer of a 2D geospatial database from digital aerial images. Digital aerial images are chosen as primary input data, as they are most of time necessary for photogrammetric projects and therefore do not imply additional costs, contrary to other data like laser scanning. In our study, only significant changes (with a size larger than 25 m^2) corresponding to new, old, partially destroyed and enlarged buildings are considered. Other inconsistencies (typically, inaccuracies in planimetry) are considered as well.
In Section 2, input data are firstly described. In Section 3, our method is detailed. Section 4 is devoted to the presentation of the results and their evaluation. Eventually, forthcoming research axes are given in concluding remarks.

2 STUDY AREA AND INPUT DATA

The study area is located in Marseille, southern France and corresponds to a very dense urban area.

2.1 Input Data

In our study, RGB and IR aerial images are used, with a Ground Sample Distance (GSD) of 20 cm, a forward and a side lap of 60%. A correlation DSM with a GSD of 20 cm is then derived, with the method described in (Pierrot-Deseilligny and Paparoditis, 2006). Moreover, true RGB and IR orthophotos are computed. Eventually, a tree mask and a lawn mask are automatically derived, with the method described in (Iovan et al., 2007).

2.2 The database to update

The database to update is a cadastral map, composed of 3 layers: the building layer is here the object of the revision and is described now. As buildings are captured in the field by surveyors, their boundaries are given in the database by walls and not gutters (like in DSM or orthophotos). Moreover, as they are initially built for tax purposes, the limits of a building actually correspond to the limits of an ownership: when shared by 2 different homeowners, one physical building is systematically split into 2 objects (Figure 2). These 2 particularities have been taken into account when designing the system.

3 METHOD

A new workflow is proposed for this study: contrary to studies found in literature, the change detection process is here split into 2 separate steps. In a first (so-called automatic verification of the database) step, the initial scene description of the database is used to guide the detection of old and geometrically shifted buildings and to validate existing buildings. In a second (so-called detection of new buildings) step, a specific algorithm allows detecting new buildings. Combined together, these 2 steps perform a comprehensive change detection workflow.

3.1 Step I: Automatic Verification of the database

This first step is composed of 3 stages: clues are firstly collected for each object to be checked (1). A similarity measure is then computed (2) to give a final acceptance or rejection decision (3).

3.1.1 Features Extraction Following the recommendations found in (Olsen and Knudsen, 2005), robust geometric criteria are preferred to radiometric criteria, too dependent on illumination conditions, and not necessarily robust to recent buildings that are often built with various, non-conventionnal (and sometimes uncommon) material.
A large amount of geometric criteria can be found in litterature: objects height (Jordan et al., 2002), height textures based on the surface roughness (Rottensteiner et al., 2005), structural (form) features (Müller and Zaum, 2005), edges delimitation (Tarsha-Kurdi et al., 2006) . . . In our *vector* database, objects to be checked are well structured, as they are represented by their boundaries. Therefore, features based on edges (i.e. contours / height discontinuities) appear to be the best adapted.
In our work, contours are extracted in the initial DSM with the classical gradient operator (Deriche, 1987), followed by a hysteresis detection of local maxima in the direction of gradients, with a sub-pixel (0.5 pixel) accuracy. Sub-pixel local maxima are then chained and poligonized to obtained DSM contours.

3.1.2 Similarity Measure

Figure 3: Inner and Outer boundaries.

Selection of pertinent building boundaries As previously mentioned, the building boundaries in the database correspond to limits of ownerships. Therefore, as shown in Figure 3, boundaries are split into 2 categories: inner boundaries and outer boundaries. Inner boundaries (i.e. shared by 2 adjacent buildings) correspond to the intangible limit between 2 ownerships and they only seldom correspond to a physical (height, even radiometric) discontinuity. By contrast, outer boundaries (i.e. belonging to only one building) must have a corresponding height discontinuity in DSM. Moreover, boundaries covered by vegetation are not verifiable. Therefore, only pertinent boundaries (i.e. outer boundaries not covered by vegetation) are kept in the process for subsequent verification.

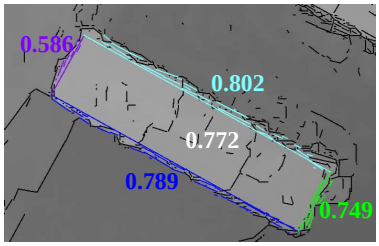


Figure 4: Description of the similarity measure. DSM contours extracted from the Canny-Deriche edge detector are pictured in black over the DSM. Each pertinent boundary is represented with its associated DSM contours and similarity measure. In white, the over-all similarity measure assigned to the building.

Definition of the similarity measure For each building to be checked, pertinent boundaries are selected. Then, as illustrated in Figure 4, for each boundary, DSM contours, located at a given distance from it and fulfilling a preset relative orientation with respect to it, are selected (top-down phase). A first measure is computed per boundary: it is based on the rate of coverage of selected contours on it. At the end, the building is assigned an over-all similarity measure that corresponds to a weighted mean of previous boundary measures.

More formally, let B be a building to be checked and b_j, a pertinent boundary (Refer to Figure 5 for an illustration). A Region Of Interest ROI_j is then defined for each pertinent boundary b_j, as a buffer given by its width d_0, centred on and aligned with b_j. The similarity measure SM is given by:

$$SM = \frac{\sum_{b_j \in B} \|b_j\| \, \rho\left(b_j, \left\{c_i : c_i \in ROI_j \text{ and } \left|\theta_i^j\right| \leq \theta_0\right\}_i\right)}{\sum_{b_j \in B} \|b_j\|} \quad (1)$$

where:

- c_i is a DSM contour
- θ_i^j is the relative orientation between c_i and b_j
- θ_0 is a preset relative orientation
- $\|\|$ gives the length of b_j
- ρ computes the coverage rate between b_j and selected contours

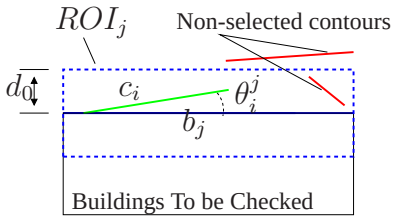


Figure 5: Similarity measure - Sketch.

In our application, d_0 and θ_0 are respectively set to 2m and 10°. Performance tests, similar to the one described in Section 4.1, show that these parameters are not critical.

3.1.3 Decision-making Module Once similarity measures are computed per building, the bottom-up phase is completed by a rule-based acceptance or rejection decision. Objects are here classified into 3 classes:

- "Destroyed" if $SM \leq T_L$
- "Modified" if $SM \geq T_L$ *and* $SM \leq T_H$
- "Validated" if $SM \geq T_H$

Objects contained in the 2 first classes are considered changes and are also given to a human operator for validation and update purposes (not described here). Remaining objects are considered unchanged. In our application, T_L is set to 0.1 and T_H is set to 0.61. T_H is by far the most important parameter in our system, as it fully determines "Change" from "No Change" objects. Performance tests are introduced in Section 4.1 and applied to estimate the best value to use.

3.2 Step II: Detection of new buildings

Once the buildings of the database have been verified, new buildings remain to be detected. For that purpose, a new workflow based on the initial DSM and a reliable and automatically computed DTM is proposed.

3.2.1 Automatic Estimation of a DTM As illustrated in Figure 7, a DTM is automatically derived from the initial DSM and the above-ground mask with the algorithm described in (Champion and Boldo, 2006). Note that the above-ground mask is composed of the initial tree mask and a building mask, directly derived from the database to update. Buildings, labelled as destroyed in Step I, are removed from the mask, as they potentially correspond to ground.

The algorithm used in our study belongs to surface-based algorithms: the DTM to reconstuct is supposed to be a regular surface (defined by some internal properties) and is estimated so that it best fits observation data (points out of the above-ground mask). A special attention has been paid to deal with outliers (above-ground points not present in the above-ground mask i.e. cars, street furniture and above all new buildings). As such points systematically deviate the DTM upwards, a module based on the M-estimator theory is integrated to the algorithm to filter them out: once outliers are removed, the final ground surface fits true ground points (inliers) and best reconstructs the true topographic surface, as shown in Figure 6.

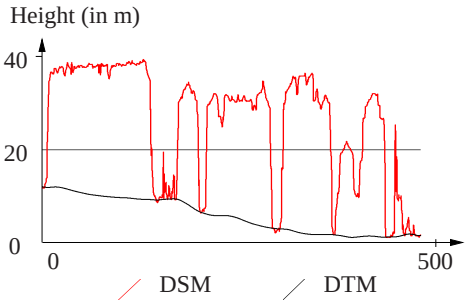


Figure 6: Profile along the red arrow (See Figure 7-1). The DTM calculated with our algorithm perfectly clings to lowest points in streets and courtyards.

3.2.2 Detection of new buildings A normalised DSM (nDSM) is then generated by substracting this DTM from the original DSM. Easy-to-use height thresholding techniques applied to this nDSM leads to the extraction of above-ground objects (in our application, the height threshold is set to 2m). Man-made structures and tree objects already present in the database to update or the tree mask are then filtered out and no significant above-ground objects are subsequently eliminated by morphological opening. Remaining objects correspond to new buildings (Figure 7-6).

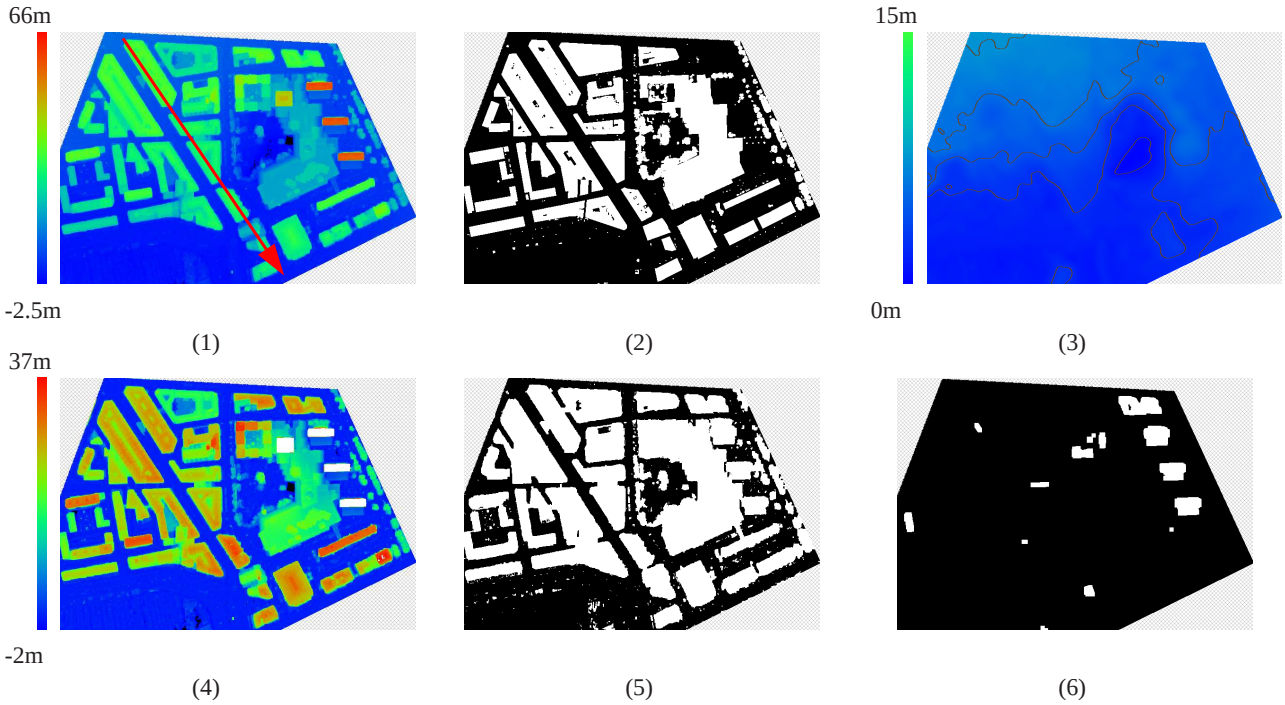


Figure 7: Detection of new buildings in Marseille. **(1) :** Initial DSM. **(2) :** Initial Above-ground Mask. $\{Trees \cup Buildings\} - \{Destroyed\ Buildings\}$ **(3) :** Automatically processed DTM (Contour lines are superimposed over the DTM, with a contour interval of 3m). **(4) :** Building Height. **(5) :** Processed Above-Ground Mask. **(6) :** New building Mask.

4 RESULTS AND EVALUATION

The database to update contains 256 buildings. 238 are still present in the database to update, 11 buildings correspond to modifications or planimetric inaccuracies and 7 buildings have been demolished. Moreover, 9 buildings have been built.
In this section, the methodology chosen to determine the best threshold T_H to use in the decision-making process is firstly described. Results are then given, assessed and discussed.

4.1 ROC Curves: How to optimise and characterise the performance of a system?

The outcome of our process is also a binary classification, in which buildings are labelled either as "Change" or "No Change". When comparing this classification to a reference classification (labels are here edited manually), 4 possible cases happen, as detailed in the 2×2 confusion matrix (Table 1).

Algo \ True	Change	No Change
Change	TP	FP
No Change	FN	TN

Table 1: Confusion Matrix

(Fawcett, 2004) proposes to evaluate a decision-making process by plotting its Receiver Operating Characteristic aka ROC curve. That comes down to plot the True Positive Rate (TPR) vs. the False Positive Rate (FPR), as the decision threshold is varied, where the TPR and FPR rates are respectively defined as:

$$TPR = \frac{TP}{TP + FN} \in [0; 1] \quad (2)$$

$$FPR = \frac{FP}{FP + TN} \in [0; 1] \quad (3)$$

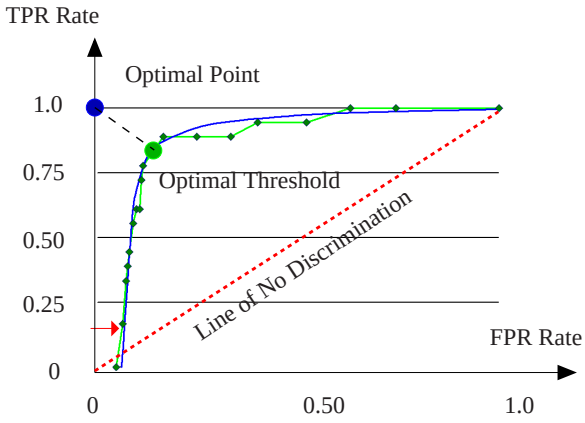


Figure 8: Performance Evaluation. In green, our experimental ROC curve. In blue, its corresponding trend curve.

In ROC curves (Figure 8), the so-called optimal point is located in the upper left corner and corresponds to a perfect classification, in which all changes are detected whitout any FP. Points located in the main diagonal (aka line of no discrimination) correspond to the result of a process that would randomly label buildings. Moreover, a test is said to be non-discriminative if its corresponding point in ROC space is situated below the main diagonal. Conversely, a test is all the more discriminative as its corresponding point is closer to the optimal point: the TP (benefit) / FP (cost) rate is then optimized.
To assess the performance of our system, the T_H threshold is also tuned from 0 to 1 and corresponding TPR and FPR rates are calculated and plotted in ROC space. As shown in Figure 8, the optimal threshold is easily derived and here corresponds to 0.61. Note that a small shift (highlighted with a red arrow) occur at the origin of the curve, which means the FPR rate is never null re-

gardless of the value of T_H. Such a characteristic is caused by both the database and the system design. Indeed, 11 unchanged buildings correspond to small structures in courtyards. All their boudaries are shared with other buildings and also considered inner boundaries in the selection phase (Subsection 3.1): in that case, the similarity measure is not computed; instead, an alert is systematically (and here wongly) sent.

4.2 Change Detection Results

The results of our change detection process is now presented (Figure 9) in a similar way as the so-called Traffic Light Paradigm (Förstner, 1994): destroyed buildings are highlighted in red, modified buildings in yellow and new buildings in orange. Concerning "No Change" objects, they are highlighted in green.

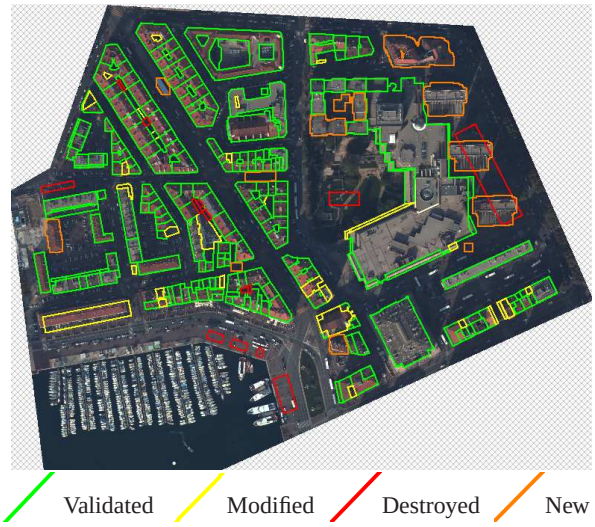


Figure 9: Change Detection Results in Marseille Test Area.

4.3 Evaluation and Discussion

4.3.1 Quality Measures As mentioned in (Rottensteiner et al., 2005), 2 quality measures are classicaly used to assess the results of a change detection process: the completeness and the correctness.

$$Completeness\ (TPR) = \frac{TP}{TP + FN} \in [0; 1] \quad (4)$$

$$Correctness = \frac{TP}{TP + FP} \in [0; 1] \quad (5)$$

As explained in (Heipke et al., 1997), these 2 measures respectively answer the questions: (1) How complete is the change detection? (2) How correct is the change detection? From a practical point of view, the completeness refers to errors kept in the final database, once updated. As for the correctness, it refers to the time lost by a human operator to check unchanged buildings. As expected in a change detection process, the FN rate must tend towards 0 (i.e. the completeness towards 1) whereas the FP rate must be as small as possible (i.e. the correctness as close to 1 as possible).

4.3.2 Quantitative Results The results of the evaluation are depicted in Figure 10 and also given in the confusion matrix (Table 2).

4.4 Discussion

As shown in Figure 11, the 2 non-detected changes (1) correspond to minor changes: the courtyard is not properly located in

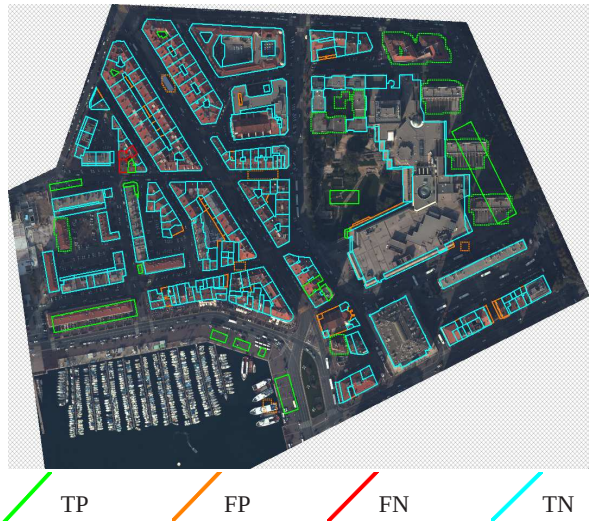


Figure 10: Evaluation in Marseille Test Area.

Algo \ True	Change	No Change
Change	25 [9.3%]	45[16.7%]
No Change	2 [≤ 1%]	197 [73.2%]
Completeness = 0.93 Correctness = 0.37		

Table 2: Confusion Matrix and Quality Measures

planimetry but validated.
Concerning false alarms (False Positive), they are of several types: they can correpond to the previously mentioned small structures in courtyards (2) or building-like structures, such as prefabs (3) or footbridges. Moreover, lots of false alarms correspond to inaccuracies / false correlation in the initial DSM. For example, the height of narrowest streets in shadows areas is sometimes overestimated and, at the end, considered new buildings by the algorithm (4). Nevertheless, this relatively high FP rate (approximately twice the number of factual changes) has to be put into perspective.
At first sight, this relatively high FP rate may appear to be a high limitation of our system. On one hand, it prevents us to consider a fully automatic change detection process. On the other hand, it systematically leads a human operator to check uselessly unchanged buildings. Nevertheless, the matter-of-fact / pragmatic approach employed in this study (that consists in splitting the change detection problem into 2 easier subtasks) allows checking only one quarter of the database (70/256 buildings) with very satisfying results: only 2 minor changes are not detected; moreover, the detection of new buildings, which remains the most important part when updating maps, is complete (9/9); eventually, the detection of difficult configurations (typically new buildings built at the same location as destroyed ones) is possible with our algorithm (Figure 11-6).
In the present (and initial) state of the development of our method, fully automatic change detection is therefore not achieved yet. However, our method can already be considered as an efficient interactive tool to support change detection and updating, and also to reduce the time-consuming aspect of such a work.

5 CONCLUSION AND FUTURE WORK

Our goal was to build a system to detect changes in the building layer of a 2D cadastral database. The system described here show

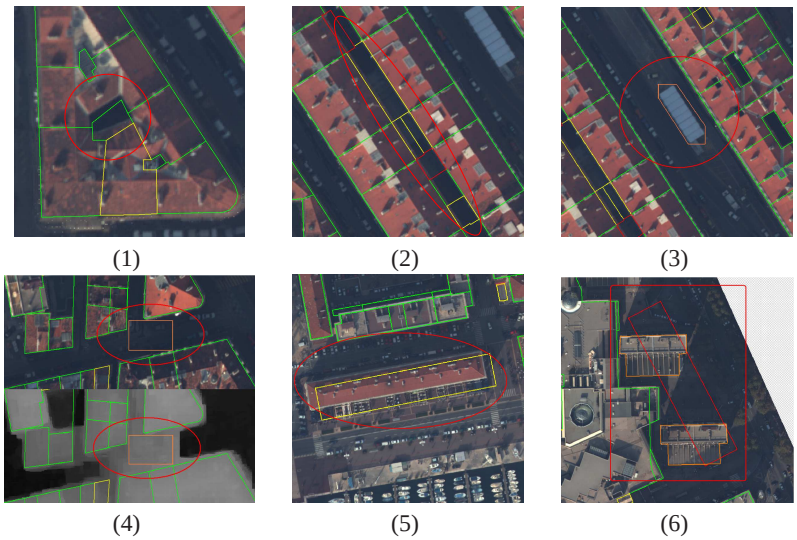


Figure 11: Evaluation Details. The colour code (Green / Yellow / Red / Orange) is the same as Figure 9. **(1) :** (FN) Inaccuracy in planimetry not detected. **(2) :** (FP) Internal structures. **(3) :** (FP) Building-like objects. **(4) :** (FP) Height inaccuracy in DSM. **(5) :** (TP) Inaccuracy in planimetry detected. **(6) :** (TP) Detection of destroyed and new buildings, even when located at the same place.

a very high potential, as all the factual changes (especially new buildings) in the study test area are detected, except for 2 minor changes. False alarms are almost caused either by building-like objects that do no need to be registered in the final up-to-date database (prefabs, footbridges ...) or height inaccuracies in the initial DSM.

We plan to test our method in a more challenging context, typically with high-resolution satellite imagery. Here again, the detection of new buildings does not appear as the critical point of the method. Tests have already been carried out with Pleiades imagery, with a GSD of 70cm (Durupt et al., 2006) and show that the processed DTM is accurate enough to extract, after appropriate filtering processes, new buildings correctly. Future work will also focus on improving the performance of the first step of the method. For that purpose, a similarity measure, computed between contours extracted from aerial images and buildings, is being considered and should be added to our decision-making process: the main challenge here remains to assign the right height to the 2D building boundaries (from noisy correlation DSM).

REFERENCES

Busch, A., Gerke, M., Grünreich, D., Heipke, C., Liedtke, C. and Müller, S., 2004. Automated verification of a topographic reference dataset: System design and practical results. In: International Archives of Photogrammetry and remote Sensing, Vol. XXXV, Part B2, Istanbul, Turkey, pp. 735–740.

Champion, N. and Boldo, D., 2006. A robust algorithm for estimating Digital Terrain Models from Digital Surface Models in dense urban areas. In: ISPRS Commission 3 Symposium on Photogrammetric Computer Vision (PCV06), Bonn, Germany.

Deriche, R., 1987. Using Cannys criteria to derive a recursively implemented optimal edge detector. International Journal of Computer Vision.

Durupt, M., Flamanc, D., Le Bris, A., Iovan, C. and Champion, N., 2006. Evaluation of the potential of Pleiades system for 3D city models production: building, vegetation and DTM extraction. In: Proceedings of the ISPRS Commission I Symposium.

Fawcett, T., 2004. Roc graphs: Notes and practical considerations for researchers. Technical report, HP Laboratories, USA.

Förstner, W., 1994. Diagnotics and performance evaluation in computer vision. In: Performance versus Methodology in Computer Vision, NSF/ARPA Workshop, IEEE Computer Society.

Heipke, C., Mayer, H., Wiedemann, C. and Jamet, O., 1997. Evaluation of automatic road extraction. In: International Archives of Photogrammetry and Remote Sensing, Vol. XXXII, pp. 47–56.

Iovan, C., Boldo, D. and Cord, M., 2007. Automatic extraction of urban vegetation structures from high resolution imagery and Digital Elevation Model. In: Proceedings of 4th International Symposium Remote Sensing and Data Fusion over urban areas (URBAN 2007), Paris, France. To Appear.

Jordan, M., Cord, M. and Belli, T., 2002. Building detection from high resolution Digital Elevation Models in urban areas. In: International Archives of Photogrammetry and Remote Sensing, Vol. XXXIV, Part B3, Graz, Austria.

Matikainen, L., Hyyppä, J. and Kaartinen, H., 2004. Automatic detection of changes from laser scanner and aerial image data for updating building maps. In: International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Istanbul, Turkey, pp. 434–439.

Müller, S. and Zaum, D. W., 2005. Robust building detection in aerial images. In: Proceedings of the ISPRS Workshop CMRT 2005, Vienna, Austria, pp. 143–148.

Olsen, B. and Knudsen, T., 2005. Automated change detection for validation and update of geodata. In: Proceedings of 6th Geomatic Week, Barcelone, Spain.

Pierrot-Deseilligny, M. and Paparoditis, N., 2006. An optimization-based surface reconstruction from Spot5-HRS stereo imagery. In: International Archives of Photogrammetry and Remote Sensing, Vol. XXXVI, Ankara, Turkey.

Rottensteiner, F., Trinder, J., Clode, S. and Kubik, K., 2005. Using the Dempster-Shafer method for the fusion of lidar data and multi-spectral images for building detection. Information Fusion pp. 283–300.

Steinnocher, K. and Kressler, F., 2006. Change detection. Technical report, EuroSDR Report.

Taillandier, F., 2005. Automatic building reconstruction from cadastral maps and aerial images. In: Proceedings of the ISPRS Workshop CMRT 2005, Vienna, Austria.

Tarsha-Kurdi, F., Landes, T., Grussenmeyer, P. and Smiegel, E., 2006. New approach for automatic detection of buildings in airborne laser scanner data using first echo only. In: ISPRS Commission 3 Symposium on Photogrammetric Computer Vision (PCV06), Bonn, Germany.

INSAR PHASE PROFILES AT BUILDING LOCATIONS

A. Thiele [a, *], E. Cadario [a], K. Schulz [a], U. Thoennessen [a], U. Soergel [b]

[a] FGAN-FOM, Research Institute for Optronics and Pattern Recognition, Gutleuthausstrasse 1, D-76275 Ettlingen, Germany - (thiele, cadario, schulz, thoennessen)@fom.fgan.de
[b] Institute of Photogrammetry and GeoInformation, Leibniz Universität Hannover, Nienburger Strasse 1, D-30167 Hannover, Germany - soergel@ipi.uni-hannover.de

Commission VI, WG III/4

KEY WORDS: Interferometry, SAR, Building Recognition

ABSTRACT:

The improved ground resolution of state-of-the-art synthetic aperture radar (SAR) sensors suggests utilizing SAR data for the analysis of urban areas. Building recognition however suffers from the consequences of the inherent side-looking viewing geometry of SAR, particularly occlusion and layover hinder the analysis. Usually extracted layover regions are not considered further in the recognition workflow. However, since layover regions contain information mixture of the building and its surrounding area, it is worthwhile to make efforts to understand such signal in order to tell different contributions of building façade and roof apart from those of other objects aiming at improvement of recognition.
Considering InSAR data conspicuous phase distributions at building locations are observable. The concerned area begins always at the building parts facing the sensor, in extreme cases even the entire data of the building area is inferred by layover. The characteristics of the phase profile in range direction depend on sensor and illumination properties as well as on geometric attributes of buildings. The processed interferometric phase information of a single range cell may result from superposition of several signal contributions of backscatters with same range distance to the sensor.
In this paper, a model is presented to calculate the expected interferometric phase values based on a given surface profile. The process takes into account that a mixture of several contributions defines the interferometric phase of a single range cell. Based on this model, simulations shall be included in an iterative analysis-by-synthesis approach for building recognition from multi-aspect InSAR data. The focus of the proposed model is to understand the impact of the building's geometry on the phase profiles, material properties are not considered in the present state. The assessment process of the simulated phase profiles is performed by comparison with real InSAR data, based on phase values of single range lines.

1. INTRODUCTION

The recognition of buildings from SAR data is often driven by analysis of magnitude images focussing on effects caused by the inherent oblique scene illumination, such as layover, radar shadow and multipath signal propagation (Simonetto et al., 2005), (Thiele et al., 2007). If InSAR data are provided, they are basically used for orthorectification and height estimation of the building hypotheses (Bolter, 2001), (Gamba et al., 2003). Due to the signal mixture in layover areas, at first glance the related InSAR elevation data often seem to exhibit arbitrary height values between ground and rooftop levels. Because of this fact, layover areas are often excluded from further recognition steps. In former research work conspicuous InSAR phase profiles at building locations were observable, which permitted the question of including this data in the process of building recognition and reconstruction. A prerequisite for exploiting layover signal is an understanding how different scattering objects contribute to the given InSAR data, which can be achieved for example based on simulation.
The analysis of InSAR phases in (Burkhart et al., 1996) was motivated by the task of building and tree extraction. The focus was on removal of noise and artefacts from the InSAR data. The appearance of the layover area in InSAR phase data at buildings was referred as "front porch". (Bickel et al., 1997) as well investigated mapping of building structures into InSAR heights and coherence data, and presented ideas to identify and avoid the layover problem. A study of the joint statistic of SAR images concentrating on layover areas was published in (Wilkinson, 1998).

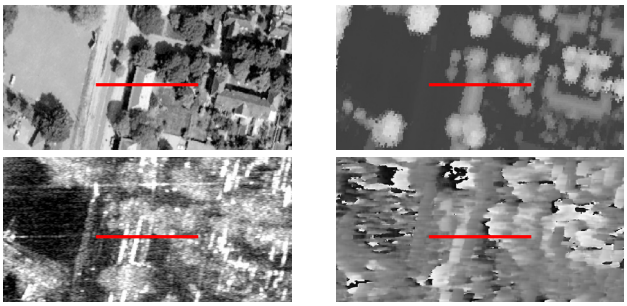


Figure 1. Appearance of gable-roofed building in orthophoto (u.l.), LIDAR DSM (u.r.), SAR magnitude image (l.l.), and InSAR phase image (l.r.); Red line marks range profile

In (Petit et al., 2000) an interferogram simulator was presented to reproduce and study coherence losses for steep gradient relief. The studies of (Cellier et al., 2006) are focused on segmentation of the borderline of the layover area as seen from the sensor, in order to determine the building's height. An interferometric mixture model using two contributors was developed and used to improve the estimation of the frontage height of an industrial building.

* Corresponding author. This is useful to know for communication with the appropriate person in cases with more than one author.

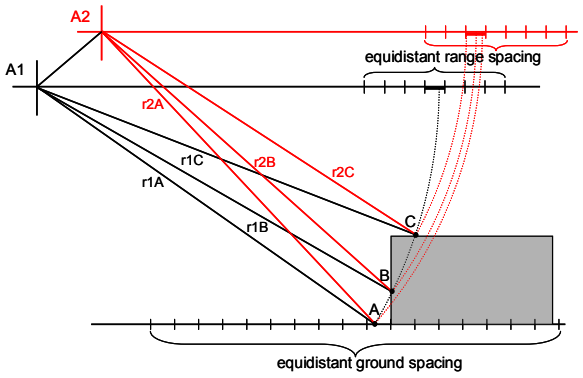


Figure 2. Contribution distribution of InSAR measurements at building location

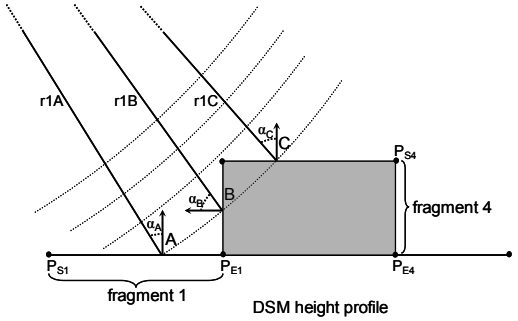


Figure 3. DSM profile of a flat-roofed building based on fragments; local incidence angle of radar signal

An analysis of the full layover area is necessary to clarify whether the InSAR phase profiles in range direction include useful information for the reconstruction of buildings. In Figure 1 an image example is given. The red line marks a profile, which is analyzed in the following.

The first step of studying the potential of phase profiles for recognition is simulation of interferometic phases, which takes all contributors into account. In this paper an appropriate phase mixture model is proposed. Results for different building types are presented and compared with real InSAR data. The analysis is mainly focused on the layover area.

2. PHASE PROFILE MODELLING

2.1 Definition and Implementation of Phase Mixture Model

For across-track SAR Interferometry two SAR images are required, which were taken separated by a baseline perpendicular to sensor tracks. After SAR processing the incoming signals gathered by antennas A1 and A2 are mapped into the related range/azimuth resolution cells (pixel) S_1 respectively S_2 of the complex SAR images. The signal phasor represented by a certain resolution cell is modelled to be the result of coherent superposition of contributions of every individual scattering object inside the related 3d volume. Equation 1 describes this superposition for a constellation depicted in Figure 2.

$$\begin{aligned} S_1 &= a_1 \cdot e^{j\varphi_1} \\ &= a_{1A} \cdot e^{j\varphi_{r1A}} + a_{1B} \cdot e^{j\varphi_{r1B}} + a_{1C} \cdot e^{j\varphi_{r1C}} \\ S_2 &= a_{2A} \cdot e^{j\varphi_{r2A}} + a_{2B} \cdot e^{j\varphi_{r2B}} + a_{2C} \cdot e^{j\varphi_{r2C}} \end{aligned} \quad (1)$$

where: a_{1n} , a_{2n} = magnitude of contributor n
φ_{r1n} , φ_{r2n} = phase of contributor n

In a simplified manner SAR can be described as distance measurement in horizontal cylinder coordinates with high resolution in radial (range) and azimuth coordinates, but poor resolution in off-nadir angle (elevation) direction. The latter is the reason for the layover phenomenon.

An example for a building is given in Figure 2. The point A (ground level), point B (building wall) and point C (building roof) have the same distance ($r_{1A} = r_{1B} = r_{1C}$) to antenna A1. But, related to antenna A2 the range distances differ ($r_{2A} \neq r_{2B} \neq r_{2C}$): point C is closest to the sensor, followed by point B and point A.

Since the layover effect is modelled to be independent from azimuth, but off-nadir angle θ rises over swath with increasing range, the simulation is carried out along range profiles.

From the phase difference of the two complex images the InSAR phase data is calculated. In the simulation different signal contributions at elevation discontinuities especially related to building locations are considered:

$$\begin{aligned} S &= S_1 \cdot S_2^* = a_1 e^{j\varphi_1} \cdot a_2 e^{-j\varphi_2} = a_1 a_2 \cdot e^{j(\varphi_1 - \varphi_2)} \\ &= (a_{1A} \cdot e^{j\varphi_{r1A}} + a_{1B} \cdot e^{j\varphi_{r1B}} + a_{1C} \cdot e^{j\varphi_{r1C}}) \cdot \\ &\quad (a_{2A} \cdot e^{-j\varphi_{r2A}} + a_{2B} \cdot e^{-j\varphi_{r2B}} + a_{2C} \cdot e^{-j\varphi_{r2C}}) \end{aligned} \quad (2)$$

The process of the phase profile modelling starts with the definition of the sensor parameters, e.g. wavelength, sensor altitude, antenna configuration, slant range resolution. From a given ground truth DSM range profiles are derived, in which linear patches are segmented (synthetic DSM). An example for a flat-roofed building is depicted in Figure 3. In the next step the synthetic DSM is split into DSM fragments in ground range of approximate constant gradient, with P_S (x_S,y_S,z_S) as start point and P_E (x_E,y_E,z_E) as end point. For each of these ground fragments the corresponding range cells are determined and the following quantities per range cell are calculated: local incidence angle α_i between radar signal path and normal vector n_i (equation 3), range distance difference Δr_i and phase difference $\Delta\varphi_i$ (equations 4).

$$\cos\alpha_i = \frac{(A_1 - P_i) \cdot n_i}{|(A_1 - P_i)| \cdot |n_i|} \quad (3)$$

$$\begin{aligned} \Delta r_i &= |A_1 - P_i| - |A_2 - P_i| \\ \Delta\varphi_i &= -2\pi \cdot \frac{\Delta r_i}{\lambda} \end{aligned} \quad (4)$$

where: P_i = point coordinate vector
A_1, A_2 = antenna vector
λ = wave length

The simulation is carried out in the range grid of the final interferogram. In this manner no co-registration of the two SAR images is required. The simulated interferometric phases are calculated by summing up all contributions of one range cell. Equation 5 describes the interferogram calculation for m backscatter contributors with the assumption of equal magnitude a:

$$S = m \cdot a \sum_m a \cdot e^{j\Delta\varphi_m} \quad (5)$$

Considering additionally the local incidence angle α_i leads to equation 6 for the interferogram calculation.

$$S_\alpha = \sum_m \cos\alpha_m \cdot a \quad \sum_m \cos\alpha_m \cdot a \cdot e^{j\Delta\varphi_m} \tag{6}$$

The process of phase profile modelling includes detection of shadow areas and flat earth correction. The shadow areas are modelled with a reflectivity of zero without implementing a noise contribution.

2.2 Examples of Simulated Phase Profiles

The influence of viewing geometry parameters is discussed for two simple building models: flat-roofed and gable-roofed. The first row in Figure 4 shows synthetic DSM profiles of a flat- (left) and a gable-roofed (right) building. The number of different contributors for a single range cell is given in the second row of Figure 4. The simulated phase profiles considering local incidence angle of radar signal are shown in the third row. In the fourth row additionally the component of flat earth correction is also considered at locations without backscatter contributors like shadow regions. In real InSAR data these regions are characterized by a random phase distribution, which is e.g. accounted in the model of (Wilkinson, 1998).

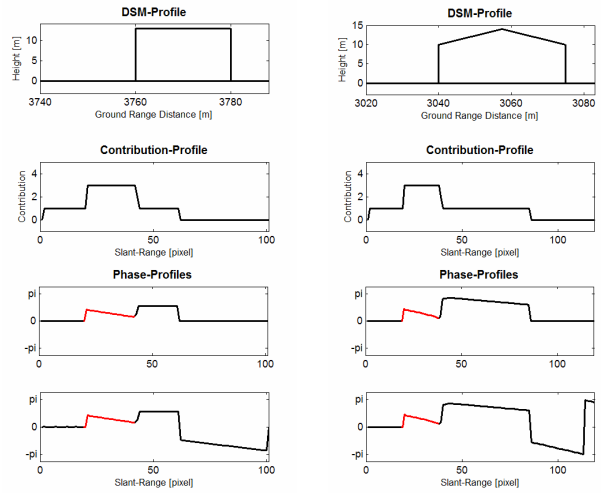


Figure 4. Simulation of phase profiles at flat- and gable-roofed buildings; Layover area marked red

The impact of the local incidence angle α of the radar signal is essential for the mapping of certain buildings. To emphasize this in Figure 5 simulated phase profiles of a gable-roofed building are given assuming off-nadir angle θ of 35° (left) respectively 50° (right). The profiles display the highest differences in front of the building at the layover area. The steepest off-nadir angle (35°) shows the highest difference (equation 3, 6).

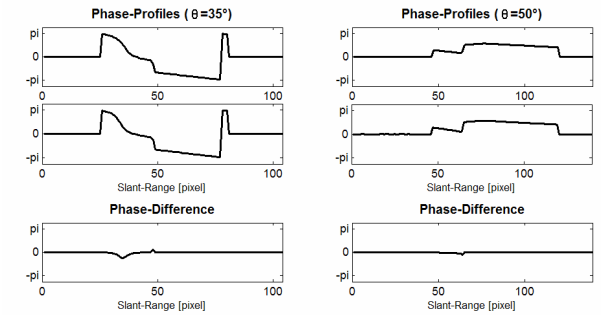


Figure 5. Simulation neglecting (top) and considering (middle) local incidence angle α, bottom: difference middle-top

Beside the local incidence angle of the radar signal other quantities affect the appearance of the InSAR phase profiles. That includes sensor parameters (e.g. range resolution and antenna configuration), illumination parameters (e.g. range distance and off-nadir angle), and building model parameters (e.g. height, width and roof type). The here presented studies are focused on the impact of illumination and building model parameters. In Figure 6 simulated phase profiles are shown for a flat-roofed building (10 m height and 40 m width) as a function of the off-nadir angle in the range from 20° up to 50°. The distribution of the phase information of the layover area as well as the roof part includes salient phase jumps to values lower zero. That is caused by the ambiguity height Δh_i defined by:

$$\Delta h_i = \lambda \frac{r_i \cdot \sin\theta_i}{B_\perp} \tag{7}$$

where: $B_\perp$ = perpendicular component of the baseline
r_i = range distance to position i
θ_i = off-nadir angle on position i

For the simulated configuration the ambiguity height is given from 8 m (20°) up to 24 m (50°). Accordingly, with a scaling from - π up to π and terrain definition at scale point zero, only the half of the elevation interval can be used for a positive display of building phases. Furthermore, the maximum phase value in the layover area is approximately building phase.

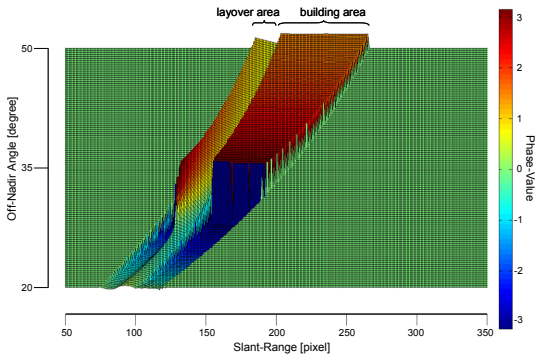


Figure 6. Changes in simulated phase profile for flat-roofed building by varying off-nadir angle

Analogue studies have been undertaken for a gable-roofed building (8 m eaves height, 12 m ridge height, 50 m width leading to 10 degree roof pitch). In addition building size respectively roof pitch are varied. The phase schemas are given in Figure 7. The influence of varying the off-nadir angle from 20° up to 50° is shown in Figure 7a. As mentioned before, a steeper off-nadir angle leads to a smaller 2 π unambiguous elevation interval (equation 7), which results in phase profile values lower zero.

The variation of the building size from 10 m up to 70 m yields to a change of the roof pitch from 40° down to 7°, the simulated profiles are depicted in Figure 7b. The first profile (at 10 m) reveals no exclusive phase signature of the roof at all, the entire roof signal is compound with other contributions in the layover area. With growing building width exclusive phase signature of the roof is observable, but after a certain building size the above mentioned phase wrapping takes effect. For small buildings this behaviour is not observed, because of the layover phenomenon. In comparison with Figure 7a is the 2 π unambiguous elevation interval for Figure 7b constant.

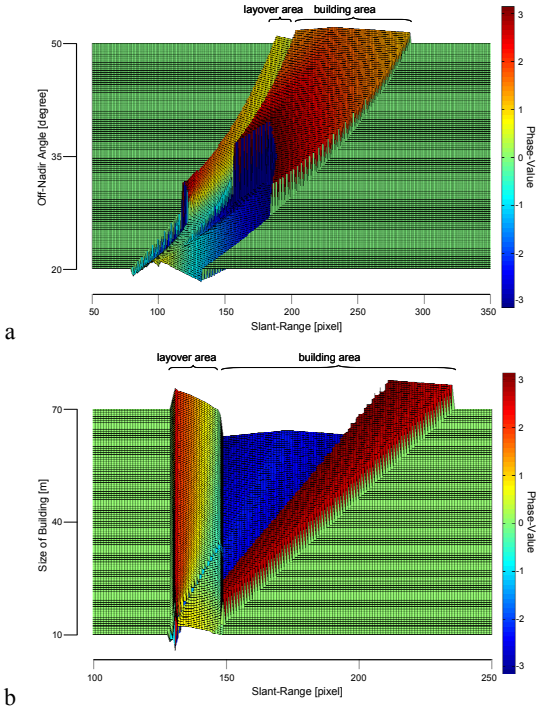


Figure 7. Changes in simulated phase profile for gable-roofed buildings by varying off-nadir angle (a) and building size (b)

This observation is confirmed by comparison with real InSAR data; especially in large urban areas with similar building inventory illuminated under different off-nadir angles.
Even if the building's height is smaller than the unambiguous elevation interval, this effect may occur due to suboptimal choice of the elevation interval borders in the InSAR processing. In such cases a phase shifting procedure is beneficial. This step contains phase shifting upwards by 2π for all phase values significantly below a threshold.

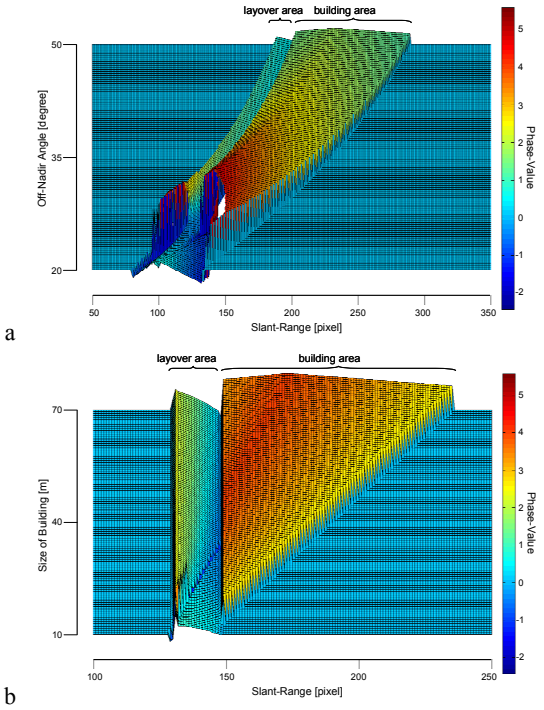


Figure 8. Changes in simulated phase profile for gable-roofed building by varying off-nadir angle (a) and building size (b) after phase shifting operation

The new results, including such a phase shifting, for both parameter variations (off-nadir angle respectively building size) are given in Figure 8. The described phase shifting is obviously helpful for the example of building size variation (Figure 8b). However, considering Figure 8a this procedure is less helpful, because the 2π unambiguous elevation interval connected to steep off-nadir angles is too small for an unambiguous visualisation of the phase values.

3. PHASE PROFILES IN REAL INSAR DATA

3.1 Calculation of Interferogram

The calculation of the interferometric phase values based on the SLC data of both antennas is done in slant range geometry. Due to the baseline established by the geometric separation of the antennas, the two SAR images have different range/azimuth coordinate grids. Therefore, co-registration is required. Since InSAR relies on the phase difference of the two given complex SAR images, sub-pixel accuracy is a prerequisite.
The subsequent interferogram generation includes multi-look filtering, followed by flat earth correction and phase centring. By phase centring a phase distribution with zero mean is achieved in the manner of the model results to make the comparison of both phase profiles possible. For some cases subsequent phase shifting is useful to reduce phase ambiguities at building locations.

3.2 Examples of Real Phase Profiles

The investigated SLC InSAR data set was produced by Intermap Technologies (Schwaebisch et al., 1999), and have a spatial resolution of about 38 cm in range and 16 cm in azimuth direction. The two X-Band sensors operated with effective baseline $B \approx 2.4$ m. The 2π unambiguous elevation span is about 16 m in close range and 20 m in far range. The mapped area is characterized by a mixture of residential and industrial building structures.
A small subset of this InSAR data set as well as the corresponding ground truth data (orthophoto, LIDAR DSM) are shown in Figure 9. The red line in the images signalizes the range profile chosen for the subsequently tests.

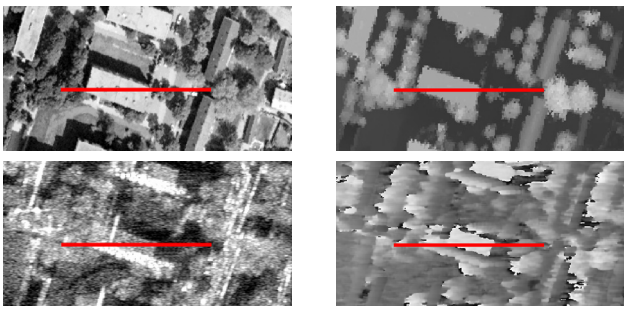


Figure 9. Appearance of flat-roofed building in orthophoto (u.l.), LIDAR DSM (u.r.), SAR magnitude image (l.l.), and measured InSAR phase image (l.r.); Red line marks range profile

For the investigation of typical measured phase profiles the influence of the multi-look window size was studied. The effects of window size [1x1] (single look), [3x3], [5x5] and [9x9] are illustrated for the chosen range profile in Figure 10 (the first image on top shows the corresponding LIDAR DSM profile).

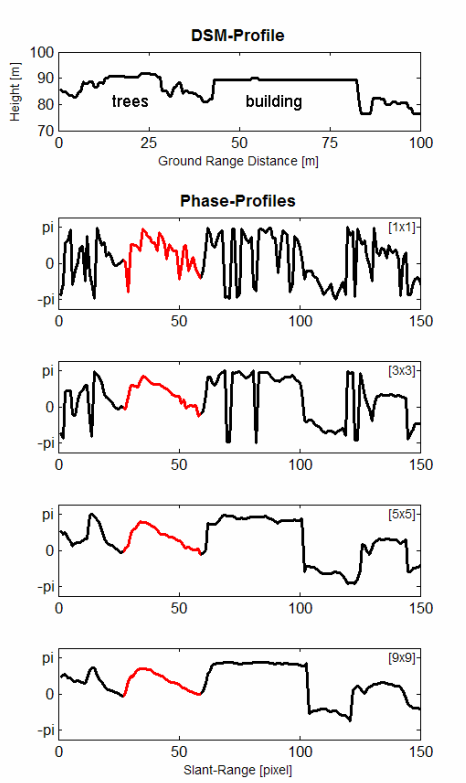


Figure 10. LIDAR DSM profile and corresponding measured InSAR phase profiles based on different multi-look parameters; Layover areas are marked red

Obviously, smoothing of the phase values to some extent is advisable, which can be confirmed with respect to the building recognition task from real InSAR data. For the following comparison between simulated and measured phase profiles the [9x9] interferogram was chosen.

In Figure 11 the measured InSAR phase profiles of a flat-roofed (left) and a gable-roofed (right) building are shown. The layover area (marked red) shows the previously discussed "front-porch" shape (Burkhart, 1996), because in this area the different heights of the contributors for the same range cell are mixed. After the initial maximum height value arising from dominate signal of the roof corner structure, usually a declining trend is observed towards the corner reflector spanned by the wall and the ground in front at the end of the layover region, which in theory should coincide with terrain level, but in reality sometimes larger elevation values are observed.

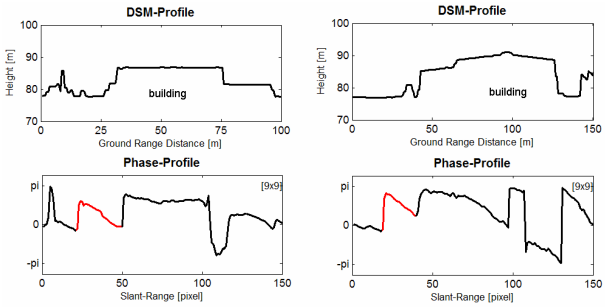


Figure 11. LIDAR DSM profiles and corresponding measured InSAR phase profiles; Layover area marked red

The typical distribution of InSAR profiles is not restricted to the investigated data set or sensor set-up. Similar phase contributions at building locations especially in the layover areas are observable also for other SAR sensors, baseline configurations and range resolutions.

4. COMPARISON OF PHASE PROFILES

The assessment of the simulated phase and the real InSAR interferograms is based on comparison of phase profiles in range direction. Therefore, a suitable LIDAR DSM profile of the scenery was selected and generalized (synthetic DSM) for the modelling process preserving geometrical dimensions and other key features (e.g. roof type of building). The modelled physical parameters (e.g. wavelength, length of baseline and sensor altitude) are extracted from log files of the investigated InSAR data set.

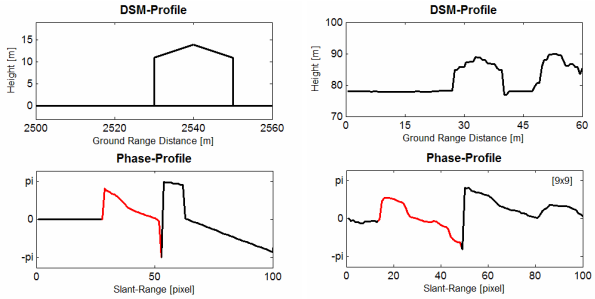


Figure 12. Synthetic DSM profile and simulated phase profile (left column); LIDAR DSM profile and measured InSAR phase profile (right column); Layover area marked red

The first comparison in Figure 12 shows the simple gable-roofed building also visualized in Figure 1. The simulation yields sharper edges and crisper contours. This is due to the generalisation effect of the model and the neglect of material properties.

The sensor-close part of the phase profiles match better then on the rear part. This difference is caused by phase noise from occlusion and interference from adjacent trees. The phase peaks below zero of both phase profiles could be compensated by the step of phase shifting. Focused on the layover area similarities are observable especially at the highest and lowest point of the area. The comparison of phase information in the shadow area is not reasonable, because the simulated phase profile only shows the flat earth component without apparent layover of the trees as observable in the measured phase profile.

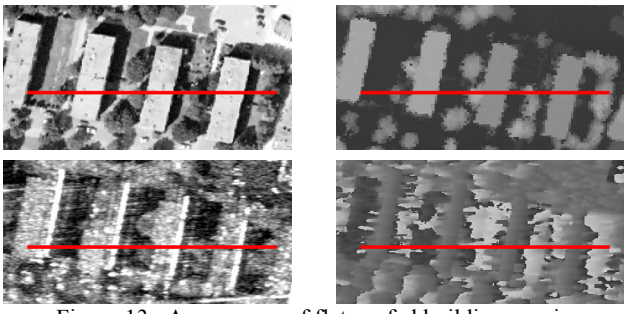


Figure 13. Appearance of flat-roofed building row in orthophoto (u.l.), LIDAR DSM (u.r.), SAR magnitude image (l.l.), and InSAR phase image (l.r.);Red line marks range profile

A more complex scenery is illustrated in Figure 13. The red line marks the location of the following investigated slant range profiles. The related synthetic and ground truth elevation data as well as simulated and measured phase profiles are depicted in Figure 14. The scene consists of a group of flat-roofed buildings of different height. Direct comparison of the phase

profiles is complicated by the interference between the neighbouring buildings and trees in-between. Furthermore, the interpretation is hampered by the 2π steps in the measured phase profile.

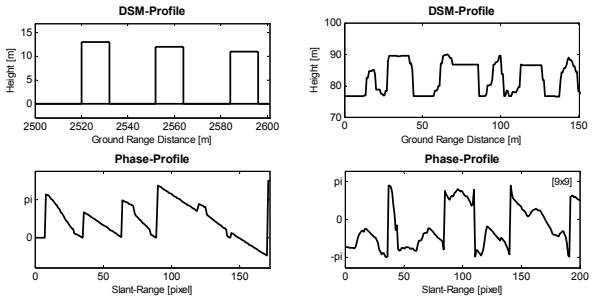


Figure 14. Synthetic DSM profile and simulated phase profile (left column); LIDAR DSM profile and measured InSAR phase profile (right column)

A similar configuration of gable-roofed buildings is given in Figure 15. The group of five buildings is characterised by four buildings of same and one of opposite orientation. In the generalised synthetic DSM, this building is replaced as a flat-roofed building (Figure 16).

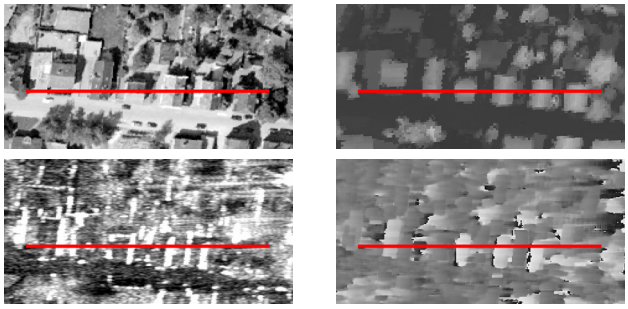


Figure 15. Appearance of gable-roofed building row in orthophoto (u.l.), LIDAR DSM (u.r.), SAR magnitude image (l.l.), and InSAR phase image (l.r.); Red line marks range profile

The simulated phase profiles in this case match even better the measured phase profile compared to the flat-roofed building row (Figure 14). The main reason is probably the absence of tall trees. Similarities are observable at significant points of the layover areas as well as in the phase distribution in the layover areas.

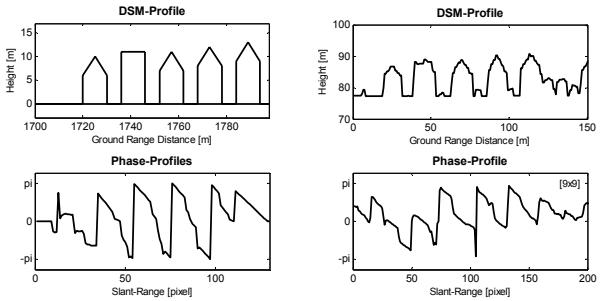


Figure 16. Synthetic DSM profile and simulated phase profile (left column); LIDAR DSM profile and measured InSAR phase profile (right column)

The sets of simulated and measured phase profiles reveal, how complex the analysis of the layover area is, especially if mutual interference caused by trees and closed neighbouring buildings have to be taken into account.

5. CONCLUSION

In this paper a model was proposed to simulate the interferometric phase values, based on a given surface profile. The simulation approach is tailored to the needs of an iterative analysis-by-synthesis approach for building recognition. The process takes into account that several scattering processes can contribute to the interferometric phase of a single range cell. Material properties are not considered, because such data is usually not available for the building recognition task. The influence of a number of different parameters was discussed. The model was verified by comparison with real InSAR data. The analysis focussed on the phase distributions in the layover area at building locations. The high correlation between simulated and real InSAR phase profiles fosters further investigations how such simulations can be best exploited in a frame work for building recognition and reconstruction. Furthermore an automatic phase unwrapping at building locations for disadvantageous 2π unambiguous elevation intervals is a crucial step and will be investigated in future work to improve the building reconstruction frame work.

6. REFERENCES

Bickel, D.L., Hensley W.H., Yocky, D.A. 1997. The Effect of Scattering from Buildings on Interferometric SAR Measurements. In: *Proceedings of IGARSS*, Vol. 4, 3-8 August, Singapore, pp. 1545-1547.

Bolter, R. 2001. Buildings from SAR: Detection and Reconstruction of Buildings from Multiple View High Resolution Interferometric SAR Data. Ph.D. dissertation, University Graz.

Burkhart, G.R., Bergen, Z., Carande, R. 1996. Elevation Correction and Building Extraction from Interferometric SAR Imagery. In: *Proceedings of IGARSS*, Vol. 1, 27-31 May, Lincoln, Nebraska, USA, pp. 659-661.

Cellier, F., Oriot, H., Nicolas, J.-M. 2006. Study of altimetric mixtures in layover areas on high-resolution InSAR images. In: *Proceedings of EUSAR*, Dresden, Germany, CD ROM.

Gamba, P., Dell'Acqua, F., Houshmand, B. 2003. Comparison and fusion of LIDAR and InSAR digital elevation models over urban areas. In: International Journal Remote Sensing, Vol. 24, No. 22, 20 November, pp. 4289-4300.

Petit, D., Adragna, F. 2000. A new interferogram simulator: 2SIR. Study of coherence losses for tortured reliefs. In: *Proceedings of SAR Workshop: CEOS - Working Group on Calibration and Validation*, ESA-SP Vol. 450, 26-29 October 1999, Toulouse, France, p.591.

Schwaebisch, M., Moreira, J. 1999. The high resolution airborne interferometric SAR AeS-1. In: *Proceedings of the Fourth International Air-borne Remote Sensing Conference and Exhibition*, Ottawa, Canada, pp. 540-547.

Thiele, A., Cadario, E., Schulz, K., Thoennessen, U., Soergel, U. 2007. Building Recognition from Multi-Aspect High Resolution InSAR Data in Urban Area. In: *IEEE Transactions on Geoscience and Remote Sensing, EUSAR Special Issue 2006,* in press.

Wilkinson, A. J. 1998. Synthetic Aperture Radar Interferometry: A Model for the Joint Statistics in Layover Areas. In: *Proceedings of COMSIG*, 7-8 September, Rondebosch, South Africa, pp. 333-338.

TOWARDS MASS-PRODUCED BUILDING MODELS

Luc Van Gool[a,b,*] **Gang Zeng**[a] **Filip Van den Borre**[b] **Pascal Müller**[a]

[a]ETH Zürich, (vangool, zengg, pmueller)@vision.ee.ethz.ch
[b]KU Leuven, (Luc.VanGool, Filip.VandenBorre)@esat.kuleuven.be

Commission III/5

KEY WORDS: Facade 3D Reconstruction, Procedural Modeling, Shape Grammar, Repetition Detection

ABSTRACT:

Interest in the automatic production of 3D building models has increased over the last years. The reconstruction of buildings, particularly their facades, is a hard subproblem, given the large variety in their appearances and structures. This paper discusses building facade reconstruction algorithms that process single images and exploit expectations about facade composition. In particular, we make heavy use of the repetitions that tend to occur, e.g. in windows and balconies. But this is only an example of the kind of rules found in recent architectural shape grammars. We distinguish between cases without and with substantial perspective effects in the input image. The focus is on the latter case, where also some depth layering in the facade can be performed automatically. We give several examples of real building reconstructions.

1 INTRODUCTION

The photogrammetry, vision, and graphics communities have already invested enormous efforts into the creation of 3D models from images. As a result, much progress has been made already. And the body of literature is still growing at a fast pace. Yet, we have to ask ourselves how far pure bottom-up approaches - and these constitute the vast majority so far - can bring us. Even when presented with a single photograph, people can often make stronger statements about the 3D structure of the objects in it than our best 3D modeling systems can generate from multiple views of the same scene. Obviously, people have strong expectations about the world, and an exquisite capacity to recognize the objects that populate it. Such knowledge is not brought to bear in most of our 3D acquisition systems. Probably the 3D modeling of faces is the one and foremost example where researchers have actually drawn heavily on expectations about that particular object class (Blanz and Vetter, 1999), and very successfully so.

Hence, there is a case for a wider object class specific extraction of 3D information. In this paper the focus is on the use of such strategies for the important class of buildings. Also here, the benefits of using prior knowledge about this class has been demonstrated already, *e.g.* in (Debevec et al., 1996, Dick et al., 2001). With the rampant growth in geo-applications like Google Earth or Microsoft Virtual Earth and the fast evaluation towards 3D GPS navigation systems, buildings form a class of objects that does indeed deserve special attention. The creation of 3D city models still is a very interactive procedure. Any advance in productivity for the creation of such models would be extremely timely. Here we propose methods for the mass production of 3D facade models, exploiting knowledge about their typical composition.

1.1 Overview of the work

In this paper we propose to rely on architecture-oriented shape grammars. The first steps of this approach have been laid in a 2001 paper by Parish and Müller (Parish and Müller, 2001). A more full-fledged grammar for buildings was proposed in our later work (Müller et al., 2006). There it was shown that these shape grammars can be used to efficiently generate models of existing buildings, or of virtual buildings of a particular style. In that work we have for instance used the building footprints at Pompeii to generate extensive 3D models of the site. This could be achieved largely automatically, once the footprints had been delineated manually from archaeological maps. The modeling of existing buildings required more intensive interaction from the user though. In more recent work (Müller et al., 2007), we started to use photographs to automatically derive grammatical rules that could be used to re-create the facades of buildings. In that work, we mainly detected repetitive structures in the facade, and automatically delineated its structural entities like windows or doors. Moreover, templates were fitted to such entities. The result was a far more compact 3D representation of the facade, yet with higher visual quality. A tool was presented to manually displace selected structures as a group with respect to the main plane of the facade. As an example, the tool allows the user to jointly select all windows, and to simultaneously put them a bit deeper than the facade. The size of the shift is interactively estimated by comparing its result with what can be seen in the photographs. In the work presented here, we exploit the repetition in facades to actually measure the depth difference for the repeated elements.

Although it would be useful to look into multi-view reconstructions, as most contributions in this area did (see refs in section 1.2), we stick to single-view analysis as in (Müller et al., 2007), because such data is much easier to come by as yet. This said, the nature of the images we used in (Müller et al., 2007) and in this paper are different in nature. Our previous work mainly focused on the use of oblique aerial imagery, i.e. the type most often used to create large city models for the moment, and ground-level imagery without strong perspective effects. Here we fully focus on close-range photogrammetry type of data, where images of facades have strong perspective distortions. The latter are becoming available at a high pace.

This leads to a two-legged strategy. If there is sufficient perspective in the image, then we propose the fully automatic strategy as laid out in this paper. If perspective effects are too weak, we resort to the earlier, slightly interactive strategy of (Müller et al., 2007). Camera focal length would tell what to do, or so could the position of vanishing points. The latter are extracted for both strategies anyway. Obviously, the vanishing point criterion is more directly related to the appearance of the building in the image.

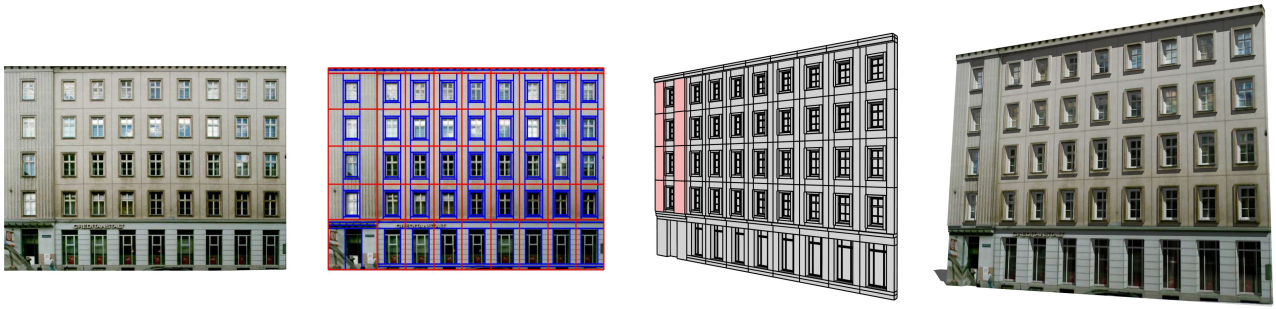


Figure 1: *The grammar based reconstruction can convert single facade textures of arbitrary resolution to semantic 3D models of high visual quality. Left: rectified facade image as input. Middle left: facade automatically subdivided and encoded as shape tree. Middle right: resulting polygonal model. Right: rendering of final reconstruction including shadows and reflections enabled by semantic information.*

The paper is organised as follows. In the remainder of this section, we give an overview of related work, to put these contributions into context. Then, section 2 recapitulates on the first strategy with weak perspective imagery, where the reader is referred to (Müller et al., 2007) for a more extensive account. Section 3 continues with the second strategy, where images have strong perspective effects. Section 4 concludes the paper.

1.2 Related Work

Shape Grammar Shape grammars were introduced by Stiny (Stiny, 1975) in the 70's as a formal approach to architectural design. They were successfully used for the construction and analysis of architectural design (Stiny and Mitchell, 1978, Koning and Eizenberg, 1981, Flemming, 1987, Duarte, 2002). The strategy of recent work was to simplify the geometric rules (Stiny, 1982), but to extend the derivation mechanisms (Parish and Müller, 2001, Wonka et al., 2003, Marvie et al., 2005, Müller et al., 2006). These shape grammars could be complemented by cellular textures (Legakis et al., 2001) to generate brick layouts and generative mesh modeling (Havemann, 2005) to generate facade ornaments. Many aspects and concepts of procedural architectural modeling are inspired by by L-systems (Prusinkiewicz and Lindenmayer, 1991), such as geometry sensitive rules (Prusinkiewicz et al., 1994), the incorporation of computer simulation (Mech and Prusinkiewicz, 1996) and artistic high-level control (Prusinkiewicz et al., 2001).

Building Facade Analysis Though limited, there is already some literature on the topic of building facade analysis. In practice, several systems still resort to semi-automatic methods (*e.g.* (Lee and Nevatia, 2003, Takase et al., 2003)). Generally, in these systems, a user is assisted by computer vision methods (Debevec et al., 1996) during modeling. This said, some automated processes have been proposed. Some of these make simplifying assumptions to get started. For example, Alegre and Dellaert (Alegre and Dellaert, 2004) as well as Brenner and Ripperda (Brenner and Ripperda, 2006) assume that windows basically correspond to dark rectangles. Others try to fit a limited set of rather complicated, parametrical models (Dick et al., 2001), or use detectors pre-trained for particular elements like windows (Mayer and Reznik, 2003). Finally, Lee and Nevatia (Lee and Nevatia, 2004) use a single ground-based image but their goal is restricted to windows.

Urban Reconstruction Urban reconstruction algorithms make use of a wide variety of input data, for example: ground-based facade images (Jepson et al., 1996, Debevec et al., 1996, REALVIZ, 2007, Lee et al., 2002, Dick et al., 2001, Wang et al., 2002), interactive editing using aerial images (Ribarsky et al., 2002), aerial images combined with ground-based panorama images (Wang et al., 2006), ground-based laser scans combined with aerial images (Früh and Zakhor, 2001), ground-based and airborne laser scans (Früh and Zakhor, 2003), ground-based laser scans combined with facade images (Karner et al., 2001), and laser scans, aerial images, and ground-based images (Hu et al., 2006). The problem is simplified if 3D data is available as depth displacements between elements (e.g. windows vs. walls) yield a strong, additional cue for their segmentation (Dick et al., 2004, Brenner and Ripperda, 2006, Schindler and Bauer, 2003).

2 STRATEGY 1: WEAK PERSPECTIVE

In this section we describe our strategy in case a facade image with small perspective effects is provided. Fig. 1 shows an overview of the procedure, from the input image on the left to the 3D result on the right. The procedure consists of four parts organized as *stages* in a pipeline. This pipeline transforms a single image into a textured 3D model including the semantic structure as a shape tree. We use a top-down hierarchical subdivision analogous to splitting rules in procedural facade modeling (Wonka et al., 2003, Bekins and Aliaga, 2005, Müller et al., 2006) (see Fig. 2). The following sections describe each of the four stages in this pipeline. Readers are referred to our Siggraph paper (Müller et al., 2007) for more technical details.

It is important to note that the facade image is rectified to a fronto-parallel view as a preprocessing step. We used an automatic rectification tool of our own implementation, which is a variant of the vanishing point based algorithm by Liebowitz and Zisserman (Liebowitz and Zisserman, 1998).

2.1 Determination of Facade Structure

The goal of this first stage is to detect the general structure in a facade and to subdivide it accordingly. The input is a single image and the output is a subdivision into floors and tiles. Additionally, we compute symmetry information so that we know

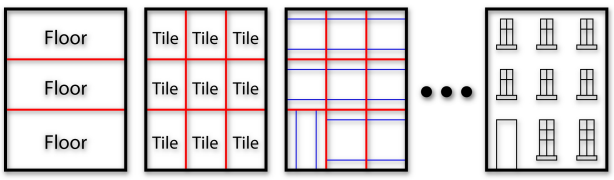


Figure 2: *Our system computes a hierarchical subdivision of facades. This subdivision scheme was successfully employed in the procedural modeling literature by various authors.*

Figure 3: *Left: the facade from Fig. 1 after removing the vertical symmetry. Right: further removing the horizontal symmetry yields the* Irreducible Facade. *Please note that we use the average pixel color for display purposes.*

for each pixel the location of corresponding pixels in symmetric tiles.

Firstly, we detect similar image regions using mutual information (MI). Secondly, based on the extracted repetitions, we create a data structure called *Irreducible Facade* (IF). An IF example is shown in the right image of Fig. 3). Its construction entails the determination of the splitting lines shown in Figs. 1(b) and 2 (only the thicker lines delineating the 'tiles' are meant here). The IF reduces the facade image to its essence, taking out all the repetitions. Although no longer visible, the IF encodes information about these symmetries that govern the floors and tiles (see Fig. 2). The aforementioned splitting lines are found through a global optimisation across all floors and tiles. This not only improves the robustness of the algorithm, but also guarantees that similar elements are split at corresponding positions.

2.2 Subdivision of Facade Tiles

At this stage we want to subdivide the detected tiles into smaller regions. We propose an algorithm which recursively selects the best splitting line in the region under consideration. See Fig. 4 for an example. This structure subdivision is a concept used in procedural modeling and will automatically create a hierarchy of elements. Such a hierarchy will be essential for further analysis, such as the generation of rules for a shape grammar.

Because individual tiles are noisy, the splitting algorithm exploits the knowledge about repetitions which is embedded in the IF. Fig. 5 left illustrates how noise makes the subdivision of individual tiles very unreliable. Therefore, the algorithm analyzes similar structures in other tiles to synchronize the derivation and in so doing, significantly improves the result (see Fig. 5 right).

2.3 Matching 3D Elements

Subdivision of facade tiles leads to a set of rectangular regions clustered into groups of similar regions. At this stage we want to match some of the architectural elements with 3D objects in a library. This is useful for the generation of high-quality geometric information and can provide some semantic interpretation. The solution has to fit the computer graphics modeling pipeline leading to two constraints: We need fast computation times and a general solution working for 3D models in a library.

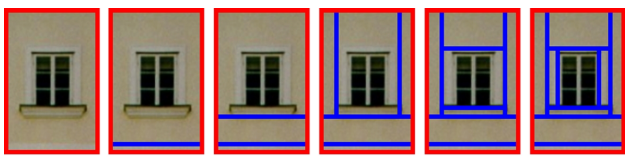


Figure 4: *In the second stage of the process, the tiles are hierarchically subdivided (illustrated as incrementally added lines). Each image represents one step of the subdivision.*

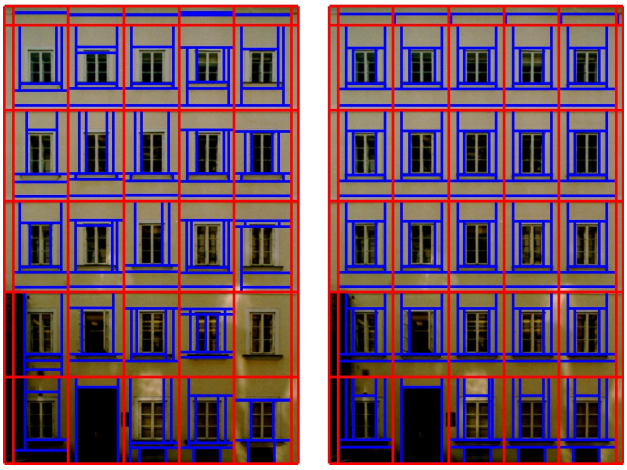


Figure 5: *To make the splitting process more stable, we make use of the previously detected tile repetitions. Left: subdivided tiles based on per-tile local split detection. Right: result if global split synchronization is added.*

2.4 Editing and Rule Extraction

At this stage of the pipeline, the resulting facade interpretation is encoded as a shape tree including fitted templates, but does not contain depth information for the relative positions of the different layers in which the facade and these templates ly. Therefore, simple editing operations are required to set the depth of the facade elements. The user can select clusters of elements and adjust their depth interactively. The added depth information is stored in the shape tree.

In the final step, we can encode the computed subdivision (i.e. the shape tree) as shape grammar rules (Bekins and Aliaga, 2005). The generated rules contain the hierarchical information and correct dimensions. As example, we present the rule set for the facade encoded as *CGA Shape* (Müller et al., 2006) in Fig. 6.

2.5 Discussion

A strength of this method is that it works well even for low resolution facade images, a challenge that has not been tackled previously. Even though the approach is robust in general, there are smaller and larger errors depending on the quality of the input image and input image complexity. Fig. 7 illustrates typical failure cases. The main problems for the fully automatic processing are heavy image noise or small irregular elements (e.g. several irregularly placed air conditioners outside of the window boundaries). In these difficult cases MI might be unable to detect repetitions (see stage 1). Also ground floors of commercial buildings are often problematic for MI due to their non-repetitive structure. As a consequence, vertical symmetries may be left undetected (even if the floors above consist of the same tiles). Another problem is posed by windows with prominent, thick frames. Furthermore, our approach assumes an orthorectified image as input. Strongly

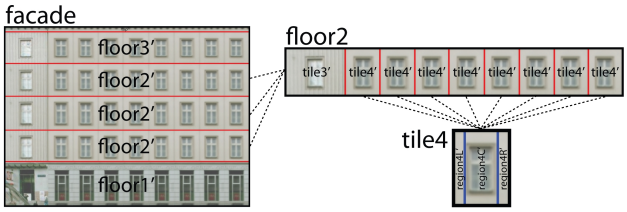


Figure 6: *The extracted shape tree can be automatically converted into a CGA shape grammar rule set.*

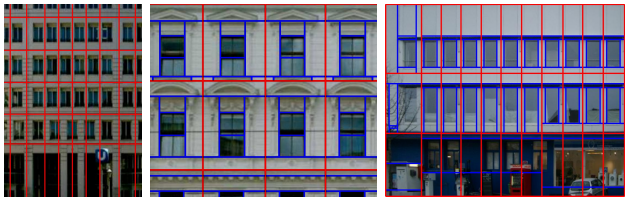


Figure 7: *Failure cases. Left: The facade structure detection cannot handle asymmetric patterns like mezzanines or non-aligned tiles. Middle: Thick window frames are wrongly interpreted as a split and the user has to reverse the split manually. Right: Worst case scenario consisting of a blurry texture with low contrast, a chaotic ground floor disturbing the MI-based repetition detection, and image noise caused by vegetation (left).*

protruding elements, such as balconies, violate this assumption and lead to incorrect tile subdivisions. To summarize, if our approach is applied on less repetitive architectural facades, we lose the structural support and run into the classic difficulties of edge detection i.e. the operations in section 2.2 will be less stable. Hence, we suggest using our technique only in urban areas with buildings of multiple storeys. Also, strong perspective effects complicate rather than help matters, as repetition detection will suffer more from differences due to multiple depth layers. Our second strategy, described next, exploits those very differences to automate the depth layering in stage 4 of the first strategy.

3 STRATEGY 2: STRONG PERSPECTIVE

Similar to the first strategy, the second one uses a single uncalibrated image and exploits the repetitions in typical facade structures. Different from the first strategy, the image here is supposed to show sufficient perspective effects and instead of interactively depth layering structures like windows, this is done automatically. First we summarize the main ideas and contributions behind the strategy. Then, we discuss them in more detail.

Relying on perfectly repeated elements would render the system fragile, especially in the presence of strong perspective effects. Nevertheless, we can hope that traces of repeated elements are found at some feature locations, if the spatial extent of these features is limited. Our method is based on a chain-wise similarity measure to robustly group these feature points. Each group provides evidence for potential repetitions. A group of feature points is also assumed to lie on a plane parallel to the facade to be reconstructed.

A new formulation is proposed to encode the interplay between repetition detection and shape recovery, *i.e.*the former provides clues for the latter, while the latter in turn produces 3D information (occlusion and depth differences) for the former. An energy functional captures the consistency between shape and image, the quality of repetition, and the smoothness. A graph-cut minimization globally optimizes the solutions for both the repetition detection and 3D shape recovery problems.

In contrast to the prior art, we are capable of reconstructing both windows and balconies, and we try to avoid using strong models for them, in order to keep the method sufficiently generic. The goal is also to deal with larger variations in appearance than what has been demonstrated so far.

3.1 Formulation and Overview

Given a single uncalibrated ground-based image $I(\mathbf{x})$ of a building facade, our goal is to reconstruct a three-dimensional shape

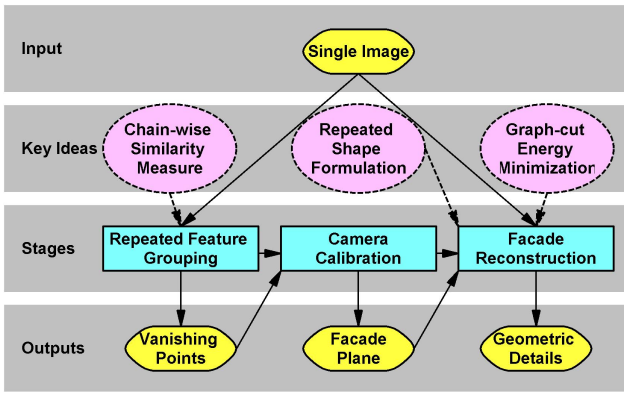


Figure 8: *Overview of the three stages of the proposed algorithm.*

$z(\mathbf{x})$ that is consistent with this input image. We assume that the facade contains multiple elements of the same type (*e.g.*similar windows or balconies) and that their appearances repeat in the horizontal and/or vertical directions along the facade plane. These two assumptions hold for most buildings. The exceptions are beyond the scope of this paper. Please note that we make no assumption on repeated element appearance or frequency.

Let $P = K[R|t]$ be the 3×4 camera matrix, where K and $[R|t]$ are the internal and external parameters. We can choose the world coordinate system such that $[R|t]$ is equal to $[I|0]$, thus we have $P = [K|0]$. Let $\mathbf{p}$ denote a unit vector representing the orientation of the facade plane, then the fact that repeated elements share the same depth layer parallel to the facade can be expressed in terms of $z(\mathbf{x})$, K and $\mathbf{p}$ as the following implicit function:

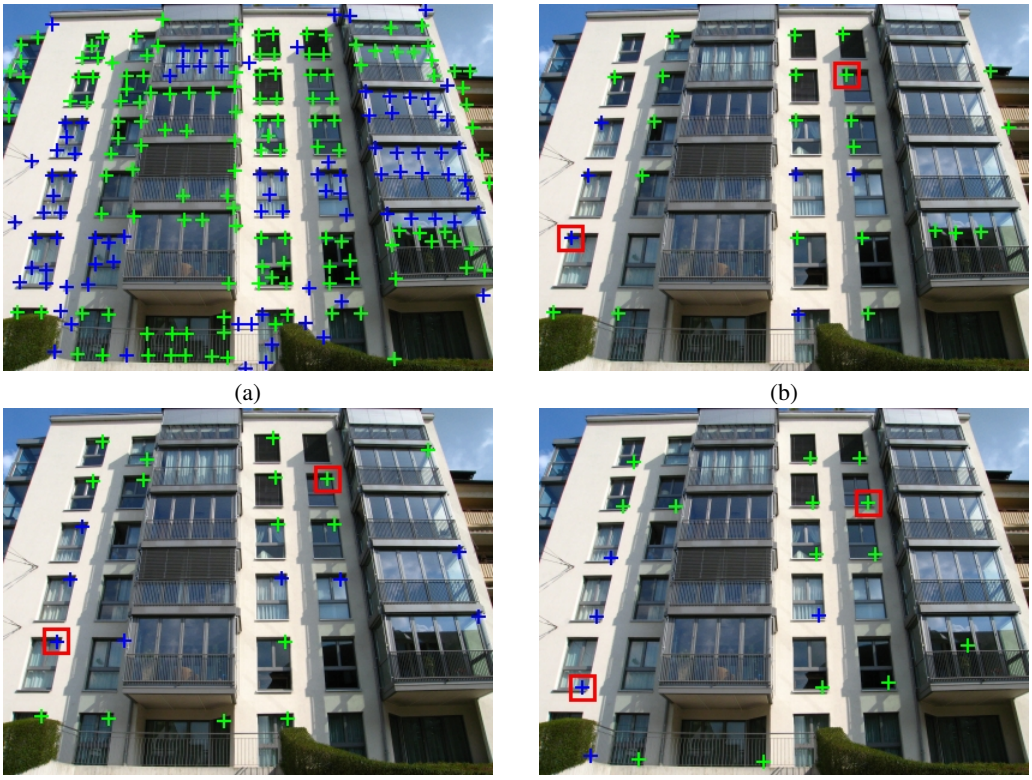
$$\mathbf{p}^T K^{-1} \left(\begin{pmatrix} \mathbf{x}_l \\ 1 \end{pmatrix} z(\mathbf{x}_l) - \begin{pmatrix} \mathbf{x}_r \\ 1 \end{pmatrix} z(\mathbf{x}_r) \right) = 0, \qquad (1)$$

where the pair $(\mathbf{x}_l, \mathbf{x}_r)$ are two arbitrary corresponding image points of the repeated elements.

In general, Eq. (1) is not easy to solve for those depths, as $\mathbf{p}$, K, $z(\mathbf{x})$ and point correspondences are all unknown. However, the following facts simplify the computation of this equation:

- Repetition is ubiquitous in facades and finding at least part of the repetitions ought to be feasible. We propose a robust matching method to provide several reliable corresponding pairs for Eq. (1);
- Considering the lines that link all pairs in the same group, two main vanishing points can be obtained corresponding to the vertical and horizontal directions of the facade plane.
- K can be determined by orthogonal vanishing points assuming that $K = diag(f, f, 1)$. This is acceptable under conditions specified later.
- $\mathbf{p}$ can be determined by a vanishing line and K;
- Once K and $\mathbf{p}$ are known, Eq. (1) becomes a linear equation in terms of $z(\mathbf{x})$, which can be optimized via a graph-cut minimization technique.

Based on these clues, we divide the whole system into the following three steps: In the first step (Sec. 3.2), repeated feature points are robustly detected and matched in groups; In the second step (Sec. 3.3), these groups are used to ease the computation of $\mathbf{p}$ and



(a) (b) (c) (d)

Figure 9: *Groups of feature points: (a) All detected feature points (Crosses with different intensities are used for visibility); (b)-(d) Three biggest groups by the chain-wise similarity measure. Although some feature points (*e.g.*see squares in (b)-(d)) are hard to match due to the different photometric or geometric transformations, they can be linked through additional evidence from intermediate patterns.*

K; In the last step (Sec. 3.4), an energy minimization scheme is designed to optimize both repetition $\{(\mathbf{x}_l, \mathbf{x}_r)\}$ and shape $z(\mathbf{x})$ densely. An overview of the whole system can be found in Fig. 8.

3.2 Repeated Feature Grouping

The first step is to define a feature detector and a robust similarity measure $C(\mathbf{x}_l, \mathbf{x}_r)$. We choose corners and a small square region around them (11×11 in the experiments) as our features. These regions are smaller than the areas matched with mutual information in our first strategy. The reason to keep the regions smaller is to make the matching more robust against the perspective effects dealt with here. Zero-mean normalized cross-correlation (ZNCC) can deal with some intensity changes (e.g. due to shadows), but is more stringent than MI, leading to fewer false matches between small regions. Thus we take $C(\mathbf{x}_l, \mathbf{x}_r) = ZNCC(\mathbf{x}_l, \mathbf{x}_r)$.

However, in order to group feature points into several types, the use of a robust pair-wise measure like ZNCC does not suffice. For effective grouping, a similarity measure is required to be an *equivalence relation* satisfying the following requirements:

- Reflexivity: $\mathbf{x}_i$ is similar to itself;
- Symmetry: If $\mathbf{x}_i$ is similar to $\mathbf{x}_j$, $\mathbf{x}_j$ is similar to $\mathbf{x}_i$;
- Transitivity: If $\mathbf{x}_i$ is similar to $\mathbf{x}_j$, and if $\mathbf{x}_j$ is similar to $\mathbf{x}_k$, $\mathbf{x}_i$ is similar to $\mathbf{x}_k$.

Unfortunately, the third requirement is not guaranteed to hold for the pair-wise similarity $C(\mathbf{x}_l, \mathbf{x}_r)$.

When comparing two feature points $\mathbf{x}_l$ and $\mathbf{x}_r$ of the same type, $C(\mathbf{x}_l, \mathbf{x}_r)$ may fail to achieve a high score due to the different photometric or geometric transformations. But important, additional evidence may come from intermediate patterns that are found, i.e. there exists a chain $\overrightarrow{\mathbf{x}_0\mathbf{x}_1\cdots\mathbf{x}_n}$ ($\mathbf{x}_0 = \mathbf{x}_l$ and $\mathbf{x}_n = \mathbf{x}_r$) in which subsequent elements have high similarity scores, even if the end nodes do not. This should encourage the system to group $\mathbf{x}_l$ and $\mathbf{x}_r$ into the same type.

3.2.1 Chain-wise Similarity The above observations have motivated us to introduce a chain-wise similarity $\hat{C}(\mathbf{x}_l, \mathbf{x}_r)$. The basic idea is to link up two feature points with the most gradually changing chain of elements, such that the pair-wise similarities between adjacent elements are high. This chain-wise similarity is then expressed as:

$$\begin{aligned} \hat{C}(\mathbf{x}_l, \mathbf{x}_r) &= \max_{\overrightarrow{\mathbf{x}_0\cdots\mathbf{x}_n}} \{\min_i \{C(\mathbf{x}_i, \mathbf{x}_{i+1})\}\} \\ \text{with} \quad & \mathbf{x}_0 = \mathbf{x}_l, \mathbf{x}_n = \mathbf{x}_r. \end{aligned} \tag{2}$$

In order to compute such a similarity measure, we embed the problem into a complete graph with the nodes being the feature points $\mathbf{x}_i$ and the edges among them having $C(\cdot)$ as weights. We then consider the spanning tree (ST), which is a graph containing all the nodes, but having no loops. With the maximal sum of its edge weights, the "maximum spanning tree" (MST) leads to an efficient computation of the chain-wise similarity $\hat{C}(\mathbf{x}_l, \mathbf{x}_r)$ for all pairs of nodes $\mathbf{x}_l$ and $\mathbf{x}_r$, *i.e.* to the path that leads to the maximum chain-wise similarity as just defined. Please note that this maximum spanning tree is similar to the usual minimum spanning tree but aimed at high edge weights.

With the chain-wise similarity $\hat{C}(\mathbf{x}_l, \mathbf{x}_r)$, the transitivity prop-

Figure 10: *Detected vanishing points: (a) With pair-wise similarity measure $C(\mathbf{x}_l, \mathbf{x}_r)$ and all feature points in Fig. 9(a), vanishing points are wrongly detected; (b)-(d) With chain-wise similarity measure $\hat{C}(\mathbf{x}_l, \mathbf{x}_r)$ and the three biggest groups in Fig. 9(b)-(d), vanishing points are correctly detected. The pairs consistent with vanishing points are linked with the superimposed lines.*

erty is satisfied as $\hat{C}(\mathbf{x}_l, \mathbf{x}_r) \geq \min\{\hat{C}(\mathbf{x}_l, \mathbf{x}_i), \hat{C}(\mathbf{x}_i, \mathbf{x}_r)\}$, and thus the grouping process of elements into types can be based on it. Given a threshold τ, if $\hat{C}(\mathbf{x}_l, \mathbf{x}_r) \geq \tau$, $\mathbf{x}_l$ and $\mathbf{x}_r$ are supposed to be of the same type, otherwise of different types. In the graph, this is equivalent to breaking certain branches of the MST, resulting in a subtree per element type.

Fig. 9 shows an example with its three biggest groups as detected by the chain-wise similarity measure. In general τ can be chosen more conservatively than the threshold used with a pair-wise measure. In all the experiments, we set $\tau = 0.9$.

3.3 Camera Calibration

Given a set of feature groups, the next step is to compute K and $\mathbf{p}$. Observing Eq. (1), it is interesting to examine the pair set,

$$S_{\mathbf{t}} = \{(\mathbf{x}_l, \mathbf{x}_r) : \begin{pmatrix} \mathbf{x}_l \\ 1 \end{pmatrix} z(\mathbf{x}_l) - \begin{pmatrix} \mathbf{x}_r \\ 1 \end{pmatrix} z(\mathbf{x}_r) = kK\mathbf{t}\}, \tag{3}$$

where k is a scaling factor and varies for different pairs and where, in the world coordinate system we use $S_{\mathbf{t}}$ to describe a set of vector pairs $(K^{-1} \begin{pmatrix} \mathbf{x}_l \\ 1 \end{pmatrix} z(\mathbf{x}_l), K^{-1} \begin{pmatrix} \mathbf{x}_r \\ 1 \end{pmatrix} z(\mathbf{x}_r))$ that share a common direction $\mathbf{t}$ for their difference. Once $S_{\mathbf{t}}$ is obtained, Eq. (1) can be converted into the form $\mathbf{p}^T\mathbf{t} = 0$. Two such equations are sufficient to solve $\mathbf{p}$.

On the other hand, in the image plane $S_{\mathbf{t}}$ manifests itself as a set of pairs $(\mathbf{x}_l, \mathbf{x}_r)$ that share a common vanishing point $K\mathbf{t}$. Vanishing points in turn can be used to solve for K: Two vanishing points $\mathbf{v}_1$ and $\mathbf{v}_2$ with perpendicular directions satisfy

$$\mathbf{v}_1^T \omega \mathbf{v}_2 = 0, \tag{4}$$

where $\omega = (KK^T)^{-1}$ is the absolute conic in the image.

In order to obtain vanishing points with perpendicular directions, we consider a group of feature points detected in Sec. 3.2. The feature points are corresponding points on the repeated elements. Searching these is based on our assumption facade structures repeat in the horizontal and vertical directions along the facade plane. This however, also implies repetitions in diagonal directions, which are of no further import to our analysis. For the detection of the horizontal and vertical vanishing points, we prefer using the repetition groups with the highest number of matched features. Within such group, we look for the two vanishing points supported by largest number of feature pairs. Fig. 10 shows an example of vanishing point detection with (b-d) and without (a) grouping information. The chain-wise similarity measure links the feature points in groups even if their appearances differ, and it produces more consistent pairs than the pair-wise ZNCC measure does. Hence, vanishing point detections with groups are more reliable than those without groups. Fig. 10(b-d) show the dominant vanishing points for the corresponding feature groups in Fig. 9(b-d).

In general, three couples of vanishing points with perpendicular directions are sufficient to solve K with the assumptions of no skew and square pixels. However, for a building facade image, the vanishing point for the third, perpendicular direction – the depth direction of the building – is often very difficult to extract, if possible at all. As a result, we simplify the internal camera model further to $K = diag(f, f, 1)$. This model assumes square pixels, as before, but adds the assumption that the principal point is known. Based on Eq. (4), the only unknown parameter f can then be solved.

Once K is obtained, $\mathbf{p}$ can be easily computed by the vanishing line that connects the two vanishing points, $\mathbf{p}^T = (\mathbf{v}_1 \times \mathbf{v}_2)^T K$, where $\times$ represents the vector or cross product.

3.4 Facade Reconstruction

Given K and $\mathbf{p}$, Eq. (1) becomes a linear equation in terms of $z(\mathbf{x})$. The last step is to design an energy minimization scheme to optimize $z(\mathbf{x})$. In order to achieve the goal, we first define a consistency measure to describe "how good $z(\mathbf{x})$ is".

We consider the group with the most feature points. Based on our assumption, the points in this group lie on a plane parallel to the facade. To fix the scale, we let $(\mathbf{p}^T, 1)$ denote this plane. Note that the resulting 3D construction will therefore come at a certain scale, which probably is not the correct one. Our final result is only defined up to an unknown scale. Thus, for each feature point $\mathbf{x}_i$ in this group, its depth value Z_i can be estimated. Please note that these $\{Z_i\}$ will not act as hard constraints when optimizing $z(\mathbf{x})$ (*i.e.*$Z_i = z(\mathbf{x}_i)$ does not always hold). $\{Z_i\}$ are only used to estimate the 3D *transformation vectors* introduced in the following paragraphs.

Suppose we are given a pair of corresponding feature points $(\mathbf{x}_l, \mathbf{x}_r)$. Corresponding points close to these two corresponding feature points can be joined by identical 3D transformation vectors:

$$\begin{aligned} \begin{pmatrix} \mathbf{x}_L \\ 1 \end{pmatrix} z(\mathbf{x}_L) \quad &- \quad \begin{pmatrix} \mathbf{x}_R \\ 1 \end{pmatrix} z(\mathbf{x}_R) \\ \equiv \begin{pmatrix} \mathbf{x}_l \\ 1 \end{pmatrix} Z_l \quad &- \quad \begin{pmatrix} \mathbf{x}_r \\ 1 \end{pmatrix} Z_r. \end{aligned} \tag{5}$$

Considering the inverse problem, the corresponding point $\mathbf{c}_{l,r}(\mathbf{x})$ of $\mathbf{x}$ and its depth value $z(\mathbf{c}_{l,r}(\mathbf{x}))$ can be determined from this 3D transformation vector, *i.e.*

$$\begin{aligned} \mathbf{c}_{l,r}(\mathbf{x}) &= \frac{\mathbf{x}z(\mathbf{x}) - \mathbf{x}_l Z_l + \mathbf{x}_r Z_r}{z(\mathbf{x}) - Z_l + Z_r}, \\ z(\mathbf{c}_{l,r}(\mathbf{x})) &= z(\mathbf{x}) - Z_l + Z_r. \end{aligned} \tag{6}$$

Based on these equations, we define two measures to describe the consistency with the input image and the quality of repetition, respectively:

$$\begin{aligned} e_{image,l,r}(\mathbf{x}) &= 1 - |C(\mathbf{x}, \mathbf{c}_{l,r}(\mathbf{x}))|, \\ e_{repeat,l,r}(\mathbf{x}) &= |z(\mathbf{x}) - z(\mathbf{c}_{l,r}(\mathbf{x})) - Z_l + Z_r|. \end{aligned} \tag{7}$$

3.4.1 Energy Minimization We minimize an energy functional of the form:

$$E_{total}(z) = E_{image}(z) + \beta E_{repeat}(z) + \gamma E_{smooth}(z). \tag{8}$$

The first term enforces the consistency between the observed image and the synthesized shape

$$E_{image}(z) = \sum_{\mathbf{x}} \min_{l,r} e_{image,l,r}(\mathbf{x}), \tag{9}$$

where $\min_{l,r}$ takes the minimum value from all potential matching points, i.e. all points found at a displacement corresponding with one of the 3D transformations coming out of repetition detection.

The second term assesses the quality of the repetition

$$E_{repeat}(z) = \sum_{\mathbf{x}} \sum_{l,r} |C(\mathbf{x}, \mathbf{c}_{l,r}(\mathbf{x}))| e_{repeat,l,r}(\mathbf{x}), \tag{10}$$

where $|C(\mathbf{x}, \mathbf{c}_{l,r}(\mathbf{x}))|$ gives more weight when the repetition quality is high.

The third term imposes smoothness. Since $\mathbf{p}$ is known, an intuitive idea is to measure the variation along $\mathbf{p}$. We define $z_{\mathbf{p}}(\mathbf{x})$ as

$$z_{\mathbf{p}}(\mathbf{x}) = \mathbf{p}^T K^{-1} \begin{pmatrix} \mathbf{x} \\ 1 \end{pmatrix} z(\mathbf{x}), \tag{11}$$

which is the distance to the plane $(\mathbf{p}^T, 0)$ in the world coordinate system. Thus the third smoothness term can be defined as

$$E_{smooth}(z) = \sum_{\mathbf{x}} ||\nabla z_{\mathbf{p}}(\mathbf{x})|| \cdot (1 - ||\nabla \bar{I}(\mathbf{x})||), \tag{12}$$

where $\nabla = (\frac{\partial}{\partial x}, \frac{\partial}{\partial y})$ is the gradient operator and $\bar{\nabla}$ represents normalised gradient magnitude, i.e. the maximum gradient value in the image is put to one and the other values are scaled accordingly. The effect of multiplying with $1 - ||\nabla I(\mathbf{x})||$ is to make the smoothing edge-preserving. Smoothing is stronger in homogeneous regions than near intensity boundaries.

3.4.2 Graph Cuts The success of graph-cut optimizations in similar domains has motivated us to embed our energy minimization problem (in Eq. (8)) into a graph, and use the classic max-flow/min-cut algorithm to obtain the optimal solution. Kolmogorov and Zabih (Kolmogorov and Zabih, 2004) give a characterization of what energy functions can be minimized using graph-cuts, and they also provide a graph-construction method. Readers are referred to their paper for more detailed information.

In the following paragraphs, we follow their approach and focus on the proof that validates our energy minimization problem, *i.e.* we convert the energy functional in Eq. (8) into a binary form which is graph-representable, *i.e.* each term $E^{i,j}$ satisfies the following condition

$$E^{i,j}(0,0) + E^{i,j}(1,1) \leq E^{i,j}(0,1) + E^{i,j}(1,0). \tag{13}$$

$\alpha-$expansion Although $z(\mathbf{x})$ is a continuous function and cannot be represented by binary variables, we can convert it for the $\alpha-$expansion operation: Any configuration $z_\alpha(\mathbf{x})$ within a single $\alpha-$expansion of the initial configuration $z(\mathbf{x})$ can be encoded by a binary function

$$\triangle z(\mathbf{x}) = \begin{cases} 0, & \text{if } z_\alpha(\mathbf{x}) = z(\mathbf{x}); \\ 1, & \text{if } z_\alpha(\mathbf{x}) = z(\mathbf{x}) \frac{\alpha}{z_{\mathbf{p}}(\mathbf{x})}. \end{cases} \tag{14}$$

Given Eq. (11) the α label defines a plane

$$\mathbf{p}^T K^{-1} \begin{pmatrix} \mathbf{x} \\ 1 \end{pmatrix} z_\alpha(\mathbf{x}) = \alpha \tag{15}$$

with orientation $\mathbf{p}$ and distance to the origin α. Let $z_\triangle(\mathbf{x})$ denote a configuration defined by $\triangle z(\mathbf{x})$. Then, we have the energy of binary variables,

$$\begin{aligned} \triangle E_{total}(\triangle z) &= \triangle E_{image}(\triangle z) + \alpha \triangle E_{repeat}(\triangle z) \\ &+ \beta \triangle E_{smooth}(\triangle z), \end{aligned} \tag{16}$$

where

$$\begin{aligned} \triangle E_{image}(\triangle z) &= E_{image}(z_\triangle) \\ \triangle E_{repeat}(\triangle z) &= E_{repeat}(z_\triangle) \\ \triangle E_{smooth}(\triangle z) &= E_{smooth}(z_\triangle). \end{aligned} \tag{17}$$

The first term $\triangle E_{image}(\triangle z)$ depends on only one variable, and

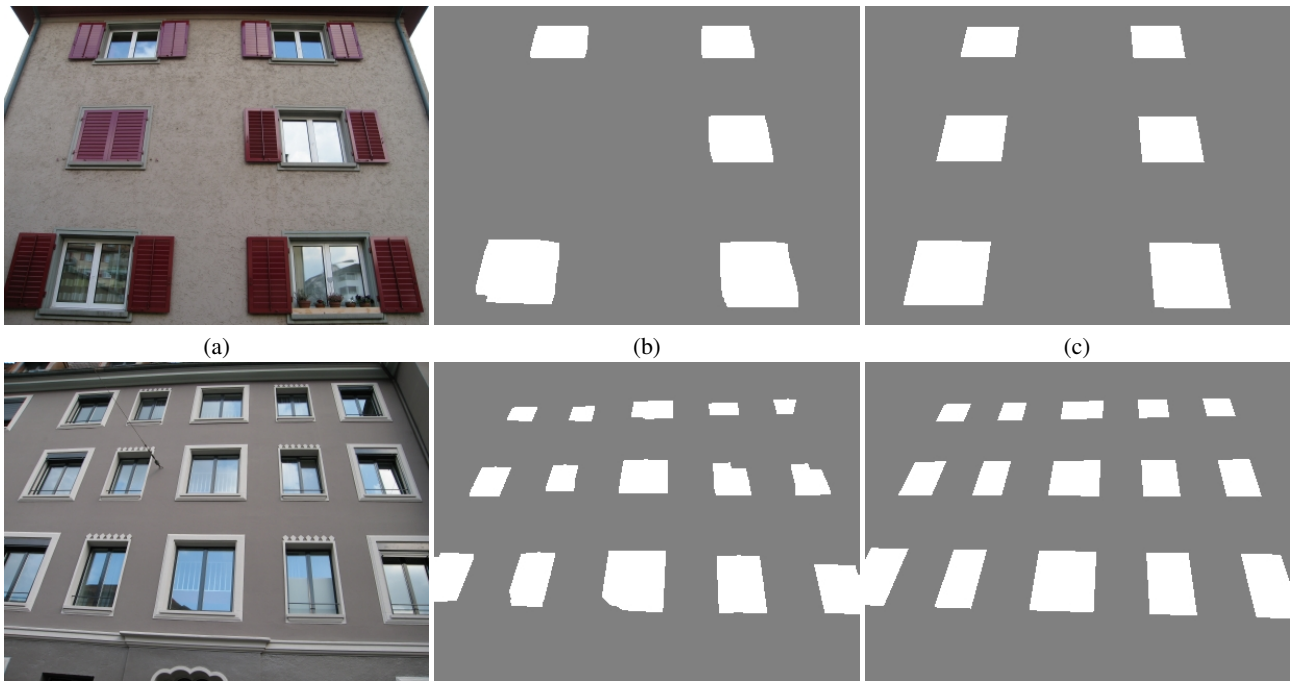


Figure 11: *Experimental result: (a) Input image; (b) Optimal shape by the graph-cut minimization; (c) Final shape after the refinement step.*

thus it is graph-representable.

For the second term $\triangle E_{repeat}(\triangle z)$, let's consider a single term $|C(\mathbf{x}, \mathbf{c}_{l,r}(\mathbf{x}))|e_{repeat,l,r}(\mathbf{x})$ in equation (10). Based on the fact that $e_{repeat,l,r}(\mathbf{x}) \equiv 0$ when (x) and $\mathbf{c}_{l,r}(\mathbf{x})$ have same label, we have $\triangle E^{\mathbf{x},\mathbf{c}(\mathbf{x})}_{repeat}(1,1) = 0$, and it can be proven that $\triangle E^{\mathbf{x},\mathbf{c}(\mathbf{x})}_{repeat}(0,0) \leq \triangle E^{\mathbf{x},\mathbf{c}(\mathbf{x})}_{repeat}(0,1) + \triangle E^{\mathbf{x},\mathbf{c}(\mathbf{x})}_{repeat}(1,0)$. Therefore, condition (13) holds.

For the third term $\triangle E_{smooth}(\triangle z)$, we have $\triangle E^{\mathbf{x},\mathbf{c}(\mathbf{x})}_{smooth}(1,1) = 0$ and $\triangle E^{\mathbf{x},\mathbf{c}(\mathbf{x})}_{smooth}(0,0) \leq \triangle E^{\mathbf{x},\mathbf{c}(\mathbf{x})}_{smooth}(0,1) + \triangle E^{\mathbf{x},\mathbf{c}(\mathbf{x})}_{smooth}(1,0)$. It is also graph-representable.

3.4.3 Shape Prior Windows and balconies often have rectangular shapes. It is not straightforward to directly add such prior constraints into the graph-cut minimization, since we have no information on the element locations before the minimization process starts. Therefore, we enforce the shape prior in a second refinement step. The element locations can then be based on the optimal shape produced with graph-cuts. The goal is also to align vertical and horizontal boundaries. Moreover, since the facade orientation $\mathbf{p}$ is known, we can add connecting planar patches orthogonal to the facade at steep transitions between different depths.

In practice, this refinement can be easily done by first summing up $z_{\mathbf{p}}(\mathbf{x})$ (in Eq. (11)) along the vertical and horizontal directions and then taking the positions of maximal variation of the sum as the element boundary positions. Figs. 12(b)-(c)&13(b)-(c) compare the element shapes before and after the refinement. Please note the connection between the balconies (in black) and wall (in gray).

3.5 Results

Implementation Details Feature points have been selected with the Harris corner detector. The threshold, τ, for classifying the feature points in Sec. 3.2 was fixed to 0.9. The β and γ parameters in Eq. (8) are set to 0.125 and 0.25, resp. The other parameters are all determined automatically by the system.

Experimental Results We show four experimental results to demonstrate the quality of the repetition detection and facade reconstruction. The first two are shown in Fig. 11. They contain windows with reflections. The first example has both open and closed windows of the same type, while the second example has two window types with different width as shown in Fig. 11(a). Such variations make the repetition detection harder. By combining repetition detection and shape recovery into the same framework and performing joint optimization via the graph-cut minimization, the proposed algorithm robustly detects the window regions by repeated feature points and depth differences as shown in Fig. 11(b). In the first example a window is missing due to the lack of depth difference. Although the window blind is a clue for a human, it can be regarded as wall texture and thus is hard to detect. The second example demonstrates the ability of our algorithm to handle different ratios between width and height. Please note the two partially open windows. By adding the prior knowledge of element shapes and layout, the boundaries of the final results are more accurate as shown in Fig. 11(c). The running time of the whole process for these two experiments are about 300 seconds on a Pentium4 3.2GHz machine.

Fig. 12 shows a third experimental result. Again, the windows vary greatly in their appearance and their detection is far from trivial. Moreover, balconies present another kind of building elements and often occlude other elements (*e.g.*windows or doors). Fig. 12(b) shows the optimal depth by the graph-cut minimization. Almost all of the windows are detected, except the top-left and top-right ones, due to the strong occlusions in both cases. There are some noticeable errors on window frames, *i.e.*the window frames are sometimes wrongly detected due to their thin shapes. Fig. 12(c) shows the final depth by adding shape priors. Please note the correction of the two missing windows and the added connections between the balcony fronts and the wall.

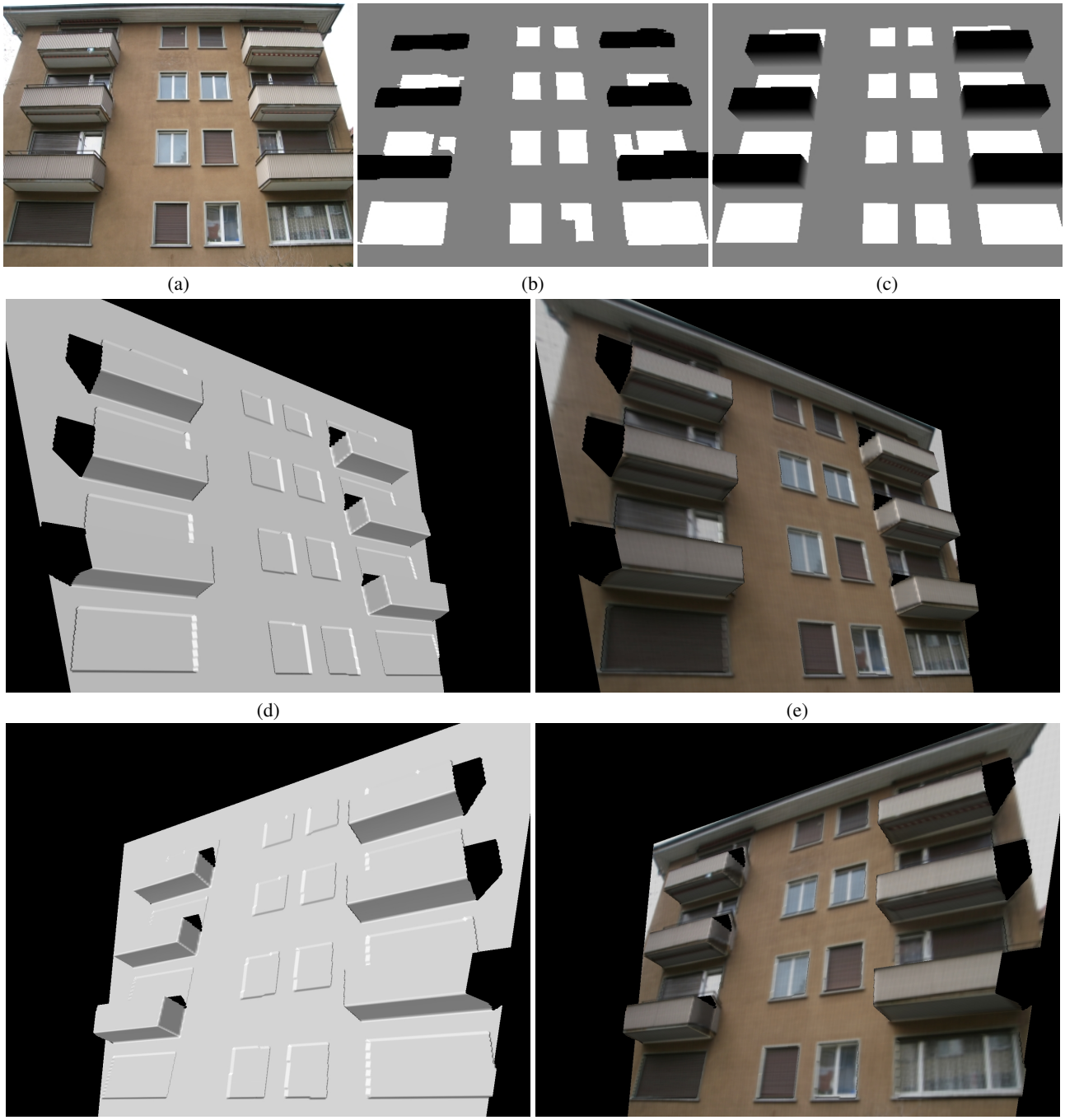


Figure 12: *Reconstruction result: (a) Input image; (b) Optimal shape $z_{\mathbf{p}}(\mathbf{x})$ obtained by the graph-cut minimization; (c) Final shape after the refinement with rectangular shape priors; (d) 3D surfaces in new viewpoints; (e) 3D surfaces with texture in the same viewpoints.*

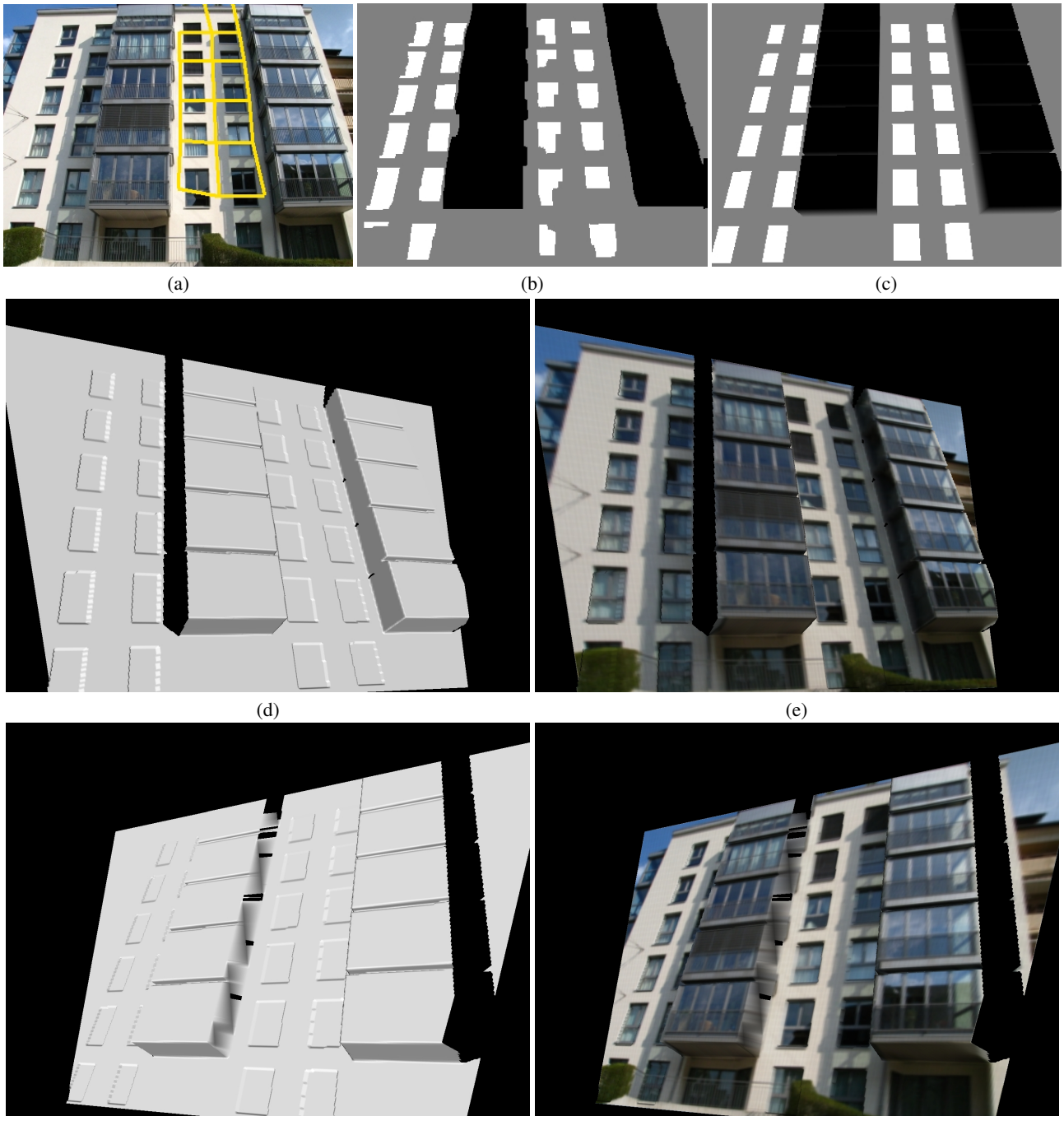


Figure 13: *Reconstruction result: (a) Input image; (b) Optimal shape $z_{\mathbf{p}}(\mathbf{x})$ obtained by the graph-cut minimization; (c) Final shape after the refinement with rectangular shape priors; (d) 3D surface in a new viewpoint; (e) 3D surface with texture in the same viewpoint; As comparison, the yellow lines in (a) shows the results of the state of the art repetition detection by Hays* et al. *(Hays et al., 2006).*

Fig. 12(d)&(e) show the 3D shape without and with texture from two new viewpoints.

Fig. 13 shows a fourth experimental result with many windows. The balconies are vertically connected. It contains several lighting effects, such as shadows, highlights, transparency and so on. Plants, window curtains, and blinds let the windows appear quite different from each other. Fig. 13(b) shows the optimal depth by the graph-cut minimization. There are some small errors on windows although the main parts are robustly detected. Fig. 13(c) shows the final depth by adding shape priors. The errors are all corrected, and occluded windows are detected. Fig. 13(d)&(e) shows the 3D shape without and with texture from a new viewpoint. The running times of the whole process for the two latter experiments are about 700 seconds.

Finally, we show a comparison with the state of the art in repetition detection. One of the detected groups of Hays *et al*. (Hays et al., 2006) is shown in Fig. 13(a). Their approach exploits the regular distribution of repeated elements (or texture). It is a generic method and has been designed for general purposes. The detected result contains almost all windows on the right hand side, but no connection is established with the similar windows on the left, as elements are supposed to be contiguous. Also, since neither 3D information nor shape knowledge is encoded, a meaningful element is often separated into parts, which belong to different repeated patterns. Compared with this work, our method is designed for the special purpose of repetition detection of building elements. Repetition detection provides information for shape recovery, while the latter in turn produces additional information for the former. Our detected results in Fig. 13(b)&(c) contain both sides of windows although they are not connected. The boundaries and depths are determined, and occlusions are also handled.

3.5.1 Discussion The proposed approach has been tested with various images of different qualities and conditions. We summarize the issues raised from these experiments as follows: Firstly, in order to speed up the whole process, we need to resize the image into 640×480 for all our experiments. A high resolution image requires too long a time for optimization. On the other hand, too small a resolution cannot provide sufficient information to distinguish different depth layers.

Secondly, readers may have noticed some errors in feature point grouping in Fig. 9. The situation is a bit like with RANSAC. Too many outliers or too many missing inliers - difficult to quantify in general terms - may cause failure to recover from such flaws.

Finally, the images should be taken with short focal lengths, so that there are strong perspective effects, conveying good depth information. This said, keeping a complete building in the field of view often imposes such choice.

4 CONCLUSION

In this paper, we have proposed two strategies for the efficient 3D modeling of facades. Both relied on prior knowledge of architectural structures and used a single image as input. The first could deal with single images, even if they show little perspective. But the price to pay is a need for limited interaction. In the second strategy, the full process is automatic, but it requires sufficiently strong perspective effects to be present in the image.
The automatic selection between the strategies still remains to be implemented. In the future, we would also like to extend the strategy further, and deal with multiple views in case they are available. Again, such system would then have to automatically adapt its strategy to the nature of the input.
For the moment, we have mainly considered simple repeat rules of the CGA Grammar. In the future we hope to extract more sophisticated rules from the imagery, and to describe the result of the image analysis as a set of CGA grammatical rules. This will lead to very compact building representations.

REFERENCES

Alegre, F. and Dellaert, F., 2004. A probabilistic approach to the semantic interpretation of building facades. In: International Workshop on Vision Techniques Applied to the Rehabilitation of City Centres.

Bekins, D. R. and Aliaga, D. G., 2005. Build-by-number: Rearranging the real world to visualize novel architectural spaces. In: IEEE Visualization, IEEE Computer Society, p. 19.

Blanz, V. and Vetter, T., 1999. A morphable model for the synthesis of 3d faces. In: SIGGRAPH, pp. 187–194.

Brenner, C. and Ripperda, N., 2006. Extraction of facades using rjMCMC and constraint equations. In: Photogrammetric Computer Vision, pp. 155–160.

Debevec, P. E., Taylor, C. J. and Malik, J., 1996. Modeling and rendering architecture from photographs: A hybrid geometry- and image-based approach. In: SIGGRAPH, pp. 11–20.

Dick, A. R., Torr, P. H. S., Ruffle, S. J. and Cipolla, R., 2001. Combining single view recognition and multiple view stereo for architectural scenes. In: ICCV, pp. 268–274.

Dick, A. R., Torr, P. H. S. and Cipolla, R., 2004. Modelling and interpretation of architecture from several images. International Journal of Computer Vision 60(2), pp. 111–134.

Duarte, J., 2002. Malagueira Grammar – towards a tool for customizing Alvaro Siza's mass houses at Malagueira. PhD thesis, MIT School of Architecture and Planning.

Flemming, U., 1987. More than the sum of its parts: the grammar of queen anne houses. Environment and Planning B 14, pp. 323–350.

Früh, C. and Zakhor, A., 2001. 3d model generation for cities using aerial photographs and ground level laser scans. In: CVPR (2), IEEE Computer Society, pp. 31–38.

Früh, C. and Zakhor, A., 2003. Constructing 3d city models by merging aerial and ground views. IEEE Computer Graphics and Applications 23(6), pp. 52–61.

Havemann, S., 2005. Generative Mesh Modeling. PhD thesis, TU Braunschweig.

Hays, J., Leordeanu, M., Efros, A. A. and Liu, Y., 2006. Discovering texture regularity as a higher-order correspondence problem. In: A. Leonardis, H. Bischof and A. Pinz (eds), ECCV (2), Lecture Notes in Computer Science, Vol. 3952, Springer, pp. 522–535.

Hu, J., You, S. and Neumann, U., 2006. Integrating lidar, aerial image and ground images for complete urban building modeling. In: 3DPVT, IEEE Computer Society, pp. 184–191.

Jepson, W., Ligget, R. and Friedman, S., 1996. Virtual modeling of urban environments. Presence 5(1), pp. 72–86.

Karner, K., Bauer, J., Klaus, A., Leberl, F. and Grabner, M., 2001. Virtual habitat: Models of the urban outdoors. In: Third International Workshop on Automatic Extraction of Man-Made Objects from Aerial and Space Imaging, Ascona, pp. 393–402.

Kolmogorov, V. and Zabih, R., 2004. What energy functions can be minimized via graph cuts? IEEE Trans. Pattern Anal. Mach. Intell. 26(2), pp. 147–159.

Koning, H. and Eizenberg, J., 1981. The language of the prairie: Frank lloyd wrights prairie houses. Environment and Planning B 8, pp. 295–323.

Lee, S. C. and Nevatia, R., 2003. Interactive 3d building modeling using a hierarchical representation. In: HLK, IEEE Computer Society, pp. 58–65.

Lee, S. C. and Nevatia, R., 2004. Extraction and integration of window in a 3d building model from ground view image. In: CVPR (2), pp. 113–120.

Lee, S. C., Jung, S. K. and Nevatia, R., 2002. Automatic integration of facade textures into 3d building models with a projective geometry based line clustering. Comput. Graph. Forum.

Legakis, J., Dorsey, J. and Gortler, S. J., 2001. Feature-based cellular texturing for architectural models. In: SIGGRAPH, pp. 309–316.

Liebowitz, D. and Zisserman, A., 1998. Metric rectification for perspective images of planes. In: CVPR, IEEE Computer Society, pp. 482–488.

Marvie, J.-E., Perret, J. and Bouatouch, K., 2005. The fl-system: a functional l-system for procedural geometric modeling. The Visual Computer 21(5), pp. 329–339.

Mayer, H. and Reznik, S., 2003. Building facade interpretation from image sequences. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences 36(3/W24), pp. 55–60.

Mech, R. and Prusinkiewicz, P., 1996. Visual models of plants interacting with their environment. In: SIGGRAPH, pp. 397–410.

Müller, P., Wonka, P., Haegler, S., Ulmer, A. and Gool, L. J. V., 2006. Procedural modeling of buildings. ACM Trans. Graph. 25(3), pp. 614–623.

Müller, P., Zeng, G., Wonka, P. and Gool, L. V., 2007. Image-based procedural modeling of facades. Proceedings of ACM SIGGRAPH 2007 / ACM Transactions on Graphics.

Parish, Y. I. H. and Müller, P., 2001. Procedural modeling of cities. In: SIGGRAPH, pp. 301–308.

Prusinkiewicz, P. and Lindenmayer, A., 1991. The Algorithmic Beauty of Plants. Springer Verlag.

Prusinkiewicz, P., James, M. and Mech, R., 1994. Synthetic topiary. In: SIGGRAPH, ACM, pp. 351–358.

Prusinkiewicz, P., Mündermann, L., Karwowski, R. and Lane, B., 2001. The use of positional information in the modeling of plants. In: SIGGRAPH, pp. 289–300.

REALVIZ, 2007. Realviz ImageModeler V4.0 product information. http://www.realviz.com.

Ribarsky, W., Wasilewski, T. and Faust, N., 2002. From urban terrain models to visible cities. IEEE Computer Graphics and Applications 22(4), pp. 10–15.

Schindler, K. and Bauer, J., 2003. A model-based method for building reconstruction. In: HLK, IEEE Computer Society, pp. 74–82.

Stiny, G., 1975. Pictorial and Formal Aspects of Shape and Shape Grammars. Birkhauser Verlag, Basel.

Stiny, G., 1982. Spatial relations and grammars. Environment and Planning B 9, pp. 313–314.

Stiny, G. and Mitchell, W. J., 1978. The palladian grammar. Environment and Planning B 5, pp. 5–18.

Takase, Y., Sho, N., Sone, A. and Shimiya, K., 2003. Automatic generation of 3d city models and related applications. In: International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, pp. 113–120.

Wang, L., You, S. and Neumann, U., 2006. Large-scale urban modeling by combining ground level panoramic and aerial imagery. In: 3DPVT, IEEE Computer Society, pp. 806–813.

Wang, X., Totaro, S., Taillandier, F., Hanson, A. and Teller, S., 2002. Recovering facade texture and microstructure from real-world images. In: Proc. ISPRS Commission III Symposium on Photogrammetric Computer Vision, pp. 381–386.

Wonka, P., Wimmer, M., Sillion, F. X. and Ribarsky, W., 2003. Instant architecture. ACM Trans. Graph. 22(3), pp. 669–677.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge support by the European Commission, through IST project Cyberwalk, and the Flemish Fund for Scientific Research, FWO. The authors also thank Peter Wonka, Simon Haegler, and Andreas Ulmer for their contributions to the weak perspective case.

Keyword Index (ISPRS)

Keyword Index (Complementary)

Author Index