A NEW FRAMEWORK FOR AUTOMATIC BUILDING DETECTION ANALYSING MULTIPLE CUE DATA

Petra ZIMMERMANN
Swiss Federal Institute of Technology, Zurich, Switzerland
Institute for Geodesy and Photogrammetry
petra@geod.baug.ethz.ch

KEY WORDS: Buildings**,** Feature Extraction, Object-oriented, Recognition

ABSTRACT

This paper describes a new framework for building recognition in aerial images. The proposed framework integrates several low-level image-processing algorithms to derive from different cues as colour, texture, edges and colour edges elevation data features to detect and recognise buildings. We developed an object-oriented Java based system for scene interpretation and reconstruction. Our system provides a set of classes for feature extraction, data storage, reasoning and visualisation, including user interaction. The core is an easy-to-expand repository of basic features and algorithms. Each step contains a possible exchange between data and processing in 2D and 3D to improve results on these levels. Due to its implementation in Java there is no restriction for future distributed applications over networks. We present a system to derive spatial data in 2D and 3D, based on independent components, knowledge and control able to distinguish between man-made objects like roads, buildings, bridges and places and natural objects like trees, vegetation on the ground. Results of building detection in a new data set from Zürich are presented.

1 INTRODUCTION

The number and variety of applications based on 3D data of terrain with or without buildings increases rapidly, since the necessary hardware becomes more powerful, costs more reasonable and visualisation faster. Many efforts in the area of building detection and reconstruction (Grün et al., 1999, Gülch et al. 1998, Henricsson, 1996) supported this development. Due to conditions in the particular application, terrain models and city models in 3D have to fulfil different requirements concerning resolution and accuracy. Building reconstruction and visualisation in 3D requires the previous careful detection of buildings. Present systems detect and reconstruct buildings in manifold ways, either through manual measurement, semi-automatic methods or for well defined situations automatically (Grün et al. 97).

1.1 AMOBE I and AMOBE II

After the successful completion of the first phase of the AMOBE project at ETH Zurich (Henricsson, 1996), focusing on reconstruction of buildings from multiple-view aerial imagery based on edges and planar patches attributed by colour information, we are now involved in a follow-up project AMOBE II, concentrating on remaining problems. AMOBE II is extended by building detection by integrating knowledge, basic models and multiple cues like colour, texture, colour edges, shadow and elevation data. AMOBE II will also improve building reconstruction (Scholze et al. 2000) by integrating these cues, treat problems caused by occlusions and focus on complex scenes and building details. In this paper we present feature-based, region-and boundary-based image segmentation based on the cues colour, DSM, texture combined with edges applied to building detection.

The first phase of AMOBE I proved how useful the strategy to process early in 3D instead of processing in 2D could be, and also the need for more extensive use of different cues became visible (Henricsson, 1998). In contrary to semi-automatic systems (Grün et al. 1999) where an operator marks buildings, or other systems that know about building locations through databases or maps (Niederöst et al. 2000), the AMOBE II system is designed to be able to detect automatically buildings and derive useful information to support reconstruction. Not only for building detection also for building reconstruction multiple cues and data sources are incorporated to improve the results (Park et al., 2000), (Zhang et al., 2000).

We present our concept of a "toolbox" of algorithms for different cues and two examples of tools for building recognition in detail, a short description of the single cues and the first results with our new dataset.

2 DESIGN OF A NEW SYSTEM

We decided to “redesign” the AMOBE I system to make it extensible and went from the former procedural language to an object-oriented software design. Also the processing is no longer like in a flow diagram fixed, but adaptive through the information derived from multiple cues and the class structure. Our system provides a set of classes for feature extraction, data storage, reasoning and visualisation, including user interaction. The core is an easy-to-expand repository, "toolbox", of basic features and algorithms. Each step enables an exchange between data and processing in 2D and 3D to improve results on these levels.

Common systems are often based on only few cues. The requirements of our system are that it should be able to-integrate as many cues as necessary and possible, further it should be able to

- Adapt algorithms and parameters depending on scale, density of buildings, degree of details etc.
- integrate models and knowledge
- integrate user interaction, visualise results
- process top-down and bottom-up
- integrate control mechanisms
- and be easy-to-expand through additional algorithms

2.1 Motivation

The idea behind this new system is to develop a framework for extracting as much as possible basic robust features from different type of data or different cues. These basic features should be extracted and stored for all further processes, that means for building detection, recognition and reconstruction processes in 2D and 3D (Table 1). As basic cues we assume colour, texture, colour edges, grey-level edges, shadow and elevation information from either DEM or DSM. Generally we consider buildings as "blobby" regions with long straight edges and high homogeneity in colour and texture domain.

Cue	**Region- or edge-based**	**Derived information for detection**	**Derivable features for recognition**	**Objects that could be recognised**	**Information for reconstruction**
Colour (RGB, HSV, L*a*b*)	Region-based Boundary information edge-based	Colour classification, homogeneous areas, similar areas	Roofs, trees, roads as regions and also the boundaries	*Homogeneous colour:* Roofplanes, roads, vegetation (Hue green) *Inhomogeneous colour:* small objects, different materials	regions for region-based matching boundary edges with information about the flanking region colour
Grey-level Edges Colour Edges	Edge-based	Grey-level gradient Gradient in the colour channels, gives additional edges	Long straight edges, useful for reconstruction Small edges	*Long straight edges:* Large (man-made) objects, ridges, boundary of roofs, roads, *Small edges:* Trees, cars, small objects	edges for edge-based matching
Texture	Region-based	Homogeneous regions: entropy, homogeneity, correlation, uniformity direction, contrast,	May differentiate between inhomogeneous vegetation areas and homogeneous man-made objects, additional attributes for segmented regions and edges	*Homogeneous texture:* Roofplanes, roads, water, bridges, shadow from buildings on e.g. roads *Inhomogeneous texture:* Vegetation, small objects, shadow from trees	additional information for region- or edge-based matching
DTM	Region-based	Slope, aspect	Gives underlying terrain, useful for blob extraction	Gives underlying terrain, useful for blob extraction	basic coarse terrain characteristics
DSM	Region-based	Blobs = regions with high elevation compared to their neighbourhood	Coarse location and shape information of roofs, e.g. ridges, size of these objects	*High elevation:* Trees, bridges, buildings, noise *Low elevation:* Roads, cars, places, lawn, shadow, small buildings	gives coarse model of buildings, ridges, main axes, shape, extend, useful as constraint in matching
Shadow	Region-based Edge-based	Colour classification in HSV space and elevation in the neighbourhood	Occlusions useful for further reconstruction to avoid errors	Occlusions useful for further reconstruction to avoid errors	indicates "dangerous" region where edges and regions may be distorted through shadows

Table 1: Applied cues and data and the derived information for building detection object recognition and building reconstruction

2.2 The new Dataset of Zürich Höngg

We used the well-known Avenches dataset (Mason et al., 1994) and the new AMOBE II dataset of Zürich (Zimmermann, 2000). This dataset covers an area nearby the centre of Zürich (Switzerland) and the ETH Hönggerberg (Figure 1). The region comprises the old centre of the quarter Höngg, different residential areas with different types of buildings (flat roofs, hip roofs etc.), other man-made objects like places, bridges and streets, different kinds of vegetation (forest, single trees, meadow, vineyard, gardens etc.), and the Limmat river. The region is situation in the Limmat Valley so that the mapped area has a strong slope.

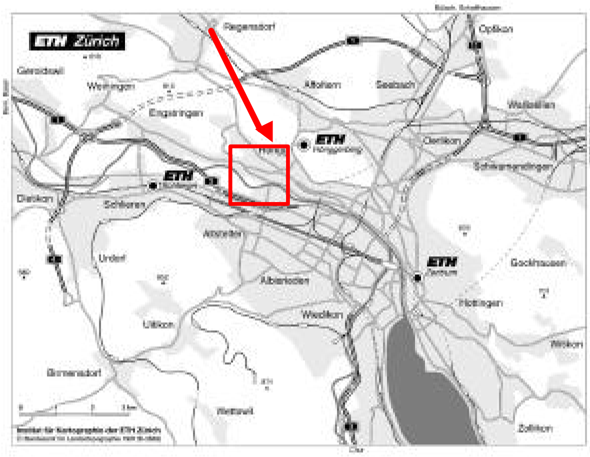

Figure 1: Map of Zürich and the situation of the aerial images of the Zürich Höngg dataset

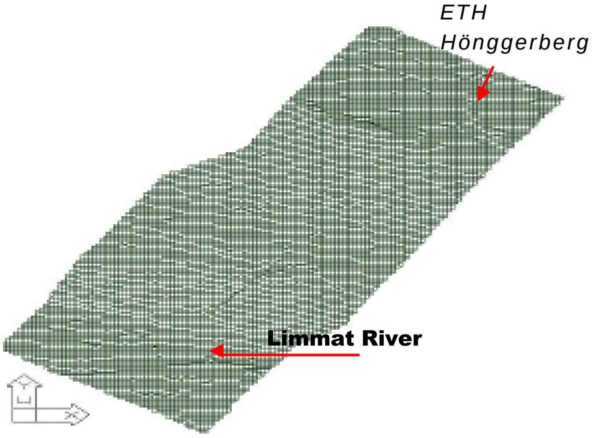

Figure 2: Manually measured DTM data shown as regular grid

The data is based on a 2x2 image block collected over Zürich at an image scale of 1:5000. It consists of two models from two neighbouring strips flown directly over the quarter Höngg. The 23cm x 23cm colour photographs were scanned at 14 µm (Figure 3). This photography was flown 1050 m over ground with 65% forward and 45 % sidewards overlap. A raster digital terrain model (DTM, Figure 2) without building or tree elevation data and a digital surface model (DSM) (Figure 4) were generated to support the processing.

Figure 3: One aerial image, No 59 of the new dataset

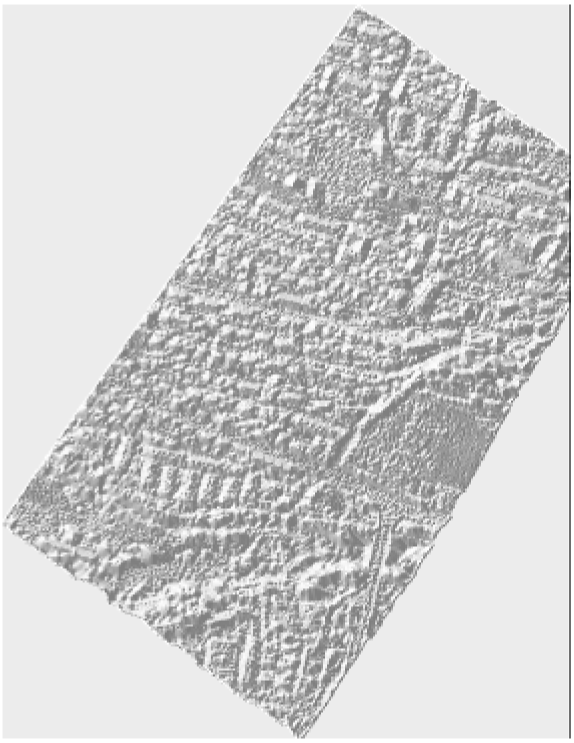

Figure 4: DSM of Image Pair 58/59 computed by the commercial software Virtuozo

2.3 Elements of the system

The system derives in each step very general information or features, then there is an instance - a very small agent-like unit, that knows the "application and requirements", so is able to give parameters, rules and the order of combining the modules. As a result attributed or new features with information about quality of results are stored. That means a system similar to a "unit construction set" with connectors able to derive general information and featured, able to exchange information with other modules of the same level or from others. Our system provides a basic set of Java classes to import and export elevation data, images and vector data results, for visualisation and interaction. The core is easy to expand for new features or for new algorithms and consists in classes for image and elevation data processing and feature extraction (edge detectors, watershed segmentation, colour segmentation, morphological filters etc.) and classes to store and process the derived features. What we call features are Points2D/3D, Lines2D/3D, Polygons2D/3D etc. with information about used cues, algorithms, parameters and quality of the feature. By combining features derived from different views or cues we get additional attributes. This data structure enables us to combine 2D and 3D information in each step, to process top-down and bottom-up, and to integrate a knowledge base or models in each step. For detailed building extraction we may use the same algorithms, same data structure, and same features as for coarse building detection by choosing different models, rules and knowledge. Control and exchange while processing is realised through a set of classes able to connect between the basic classes, an operator can replace single classes.

3 SEGMENTATION METHODS FOR AERIAL IMAGES AND DSMS

One goal of AMOBE II is the automatic building extraction including coarse 3D description from coloured aerial images and DSM or DTM data. The system tries to derive as many as necessary useful „low level“ information with according accuracy and to fuse these information (area, pixel or edge based). We implemented different segmentation methods, edge-based, region based and hybrid methods for DSM data and aerial imagery. Here we present one method to DSM segmentation and one method to aerial colour image segmentation.

3.1 Elevation – Blob extraction

We use the DTM and DSM data to derive general features as aspect and slope of the terrain (Skidmore 1989). A surface can be approximated using a bivariate quadratic function: $z = ax^2 + by^2 + cxy + dx + ey + f$ (1)

The change of elevation in x and y direction can be used to identify the direction and magnitude of the steepest gradient. These two parameters can be found by taking the partial first order derivatives of (1) above, with respect to x and y. The slope can be computed by combining the two component partial derivatives:

$$\frac{dz}{dxy} = \sqrt{\frac{dz^2}{dx^2} + \frac{dz^2}{dy^2}} \qquad (2)$$

This can be transformed and written as

$$slope = \arctan(\sqrt{d^2 + e^2}), \qquad (3)$$

$$aspect = \arctan(\frac{e}{d}). \qquad (4)$$

The DSM is segmented by extracting regions with high slope and aspect and if available checked also by subtracting the DTM data. So we get coarse "blobs", regions that give the location of objects that have higher elevation than the surrounding terrain. These objects may be buildings, bridges or trees etc. (Figure 5). The blobs are stored with their boundary polygon information, the slope and aspect information inside and average slope and aspect along the boundary (Figure 6,7). The accuracy of the position and shape of the blob is heuristically set to a lower value than in the other algorithms.

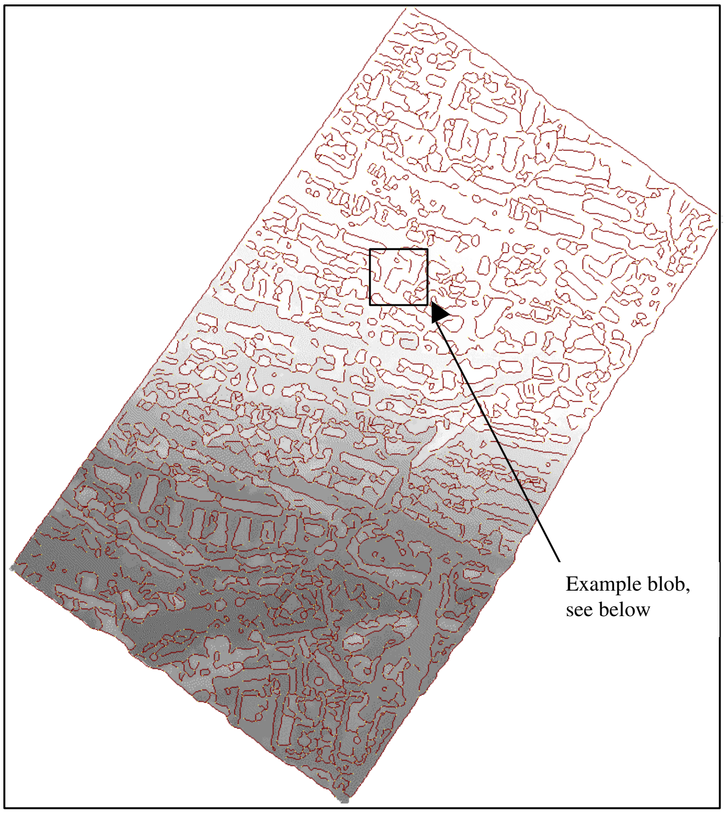

Figure 5: Extracted blobs through DSM information

3.2 Elevation – Blob extraction, coarse Roof Modelling

We use the DSM to derive coarse models of the buildings containing coarse shape and extend and the number and direction of the main axes. Given the aspect values within a blob we can get the direction of inclination of the roofs (Figure 8), strong change in slope and aspects indicates a ridge line (Figure 9), also, if there is a change in aspect a break-line is expected. Ridgelines are derived by computing the second derivatives: then a ridge point lies on a local convexity that is orthogonal to a line with no convexity or concavity, which means:

$$\text{Ridge } \frac{\partial z^2}{\partial x^2} > 0\,,\ \frac{\partial z^2}{\partial y^2} = 0 \tag{5}$$

For further reconstruction steps each blob with its position, an average height of the recognised ridge lines and the roof boundaries with their position and height, and the flanking regions information concerning slope and aspect is stored as "BasicRoofModel" (Figure 10). Due to noise in the DSM and low resolution no small or weak ridges and also no details can be recognised.

Figure 6: aerial image

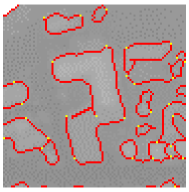

Figure 7: DSM blob, projection as image

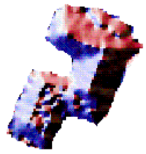

Figure 8: example of aspect of a blob

Figure 9: slope of a selected region

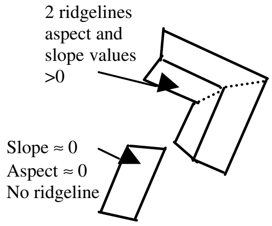

Figure 10: scheme of ridges and boundary lines

The geometric accuracy of the positions of the ridgelines is weighted very low. The extracted blobs with their information and ridgelines are projected back to the aerial images to get an approximate location of objects with elevation above ground.

3.3 Grey-level Edges and Colour Edges

Long straight edges may indicate the location of man-made objects and are directly applicable for edge-based matching. We extract edges in colour imagery, focussing on long straight edges. Edge preserving smoothing is applied to the input image, and then the Canny gradient operation derives the gradient information, which is used by a hybrid region-and edge based classification. In colour space -depending on the algorithm RGB, HSV or L*a*b* -we apply gradient filtering in each colour channel to derive colour edges to get slight improvements in edge detection compared to grey-level edge detection. To merge the information we just overlay the edges from the singe channels. Edges detected in all channels get a detection attribute with a weight of 3, whereas edges that are detected only in one channel get a weight of 1, each time multiplied with normalised gradient strength. Edge information is stored as Polygon2D with Endpoints of the single Lines2D, the detection attribute, the accuracy of linefitting of each Line2D and the distance and angle to the linked Line2D in the edge polygon, whether the polygon is closed, curved or straight is also stored. In processing other cues additional information is added e.g. colour attributes or entropy of the flanking regions.

3.4 Colour

Homogeneous coloured regions are assumed to belong to the same regions. We apply colour segmentation to derive homogeneous regions in different colour spaces: Saturation from HSV space helps to discriminate between shadow and

Figure 11: aerial image of dataset Höngg

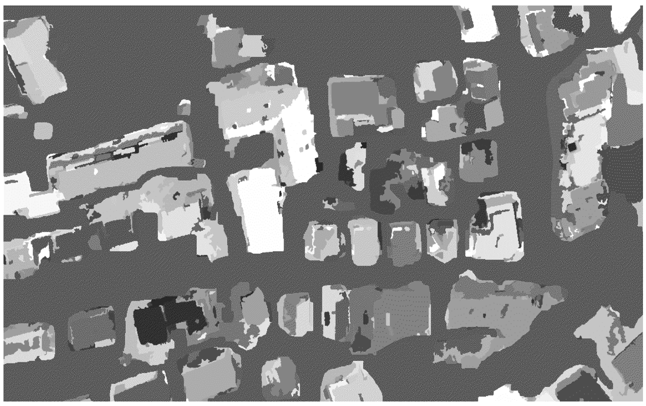

Figure 12: segmented blob regions

non-shadow regions, Hue is used to support separation of vegetation. The presented algorithm computes a low-level automatic colour-based segmentation. It fuses colour and edge features (see Figure 12) in region-growing process to derive homogeneous colour regions with accurate boundaries (Figure 15). Colour is the major segmentation feature of this algorithm edge information (Figure 15) is as a secondary cue implemented because the boundaries of colour areas may not be accurate due to noise, too low resolution etc. edge information is implemented in the process of segmentation to win accuracy. This combination uses complementary information: colour based merging works well on smoothed and quantized images while edge detection finds high frequency information in images (Figure 12). Input data for this processing is an aerial image (Figure 11), optional an edgemap if available instead of computing it in this step, and if available the blob data. Segmentation of only colour data will lead to misclassifications e.g. from roads and roofs of similar materials. Figure 13 shows one example - the flat roof in the bottom part is due to its photometric properties classified to be similar to the road on the left side. Through to the blob information the classification region is reduced. The processing steps are:

- the image is quantized in an perceptually uniform colour space (CIE L*u*v* or HSV) to a limited number of colours using clustering (k-means)
- the image is smoothed with a non-linear median filter, the edgemap is extracted by Canny operator from every colour channel, strong and weak edges are labelled
- iterative clustering is done, two neighboured regions with the smallest colour distance are continuously merged until the difference is larger than a given threshold
- only non-edge pixels are processed and labelled, edge pixels are not merged into any region, regions clearly separated by long lines will be prevented to be connected or merged
- recursively common boundaries of adjacent regions are removed
- after merging all non-edge pixels, edge pixels are assigned to the neighbour region according to the colour similarity (threshold)
- refinement through multiple repetitions, small regions can be eliminated
- perimeter, bounding box, number of neighbour pixels and edges are computed
- segmented regions must be homogeneous, checked by texture comparison

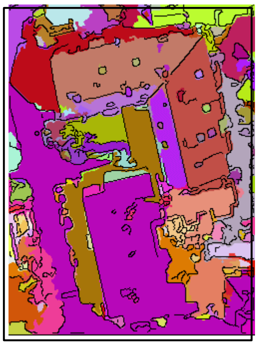

Figure 13: edgemap and segmentation without blobs

Region boundaries are stored as an Polygon2D, with an attributes for colour range in the regions, number of merged regions, number of edges with their attributes (4.3), perimeter, bounding box, number of neighbour pixels and their texture attributes (see 4.5). The quotient of region pixels to perimeter gives a measure of compactness.

3.5 Texture

To guarantee the spatial homogeneity of each segmented region and to get a criterion to differentiate between e.g. trees and buildings, or roads and low vegetation (Table 1), texture information is integrated and also segmented. We use the well-known definitions of texture from Haralick at al. 1973 and derive 6 texture parameter: contrast, correlation, direction, entropy, homogeneity and uniformity. Pure texture segmentation gives a only a coarse segmentation, so we use texture segmentation only as auxiliary tool to check colour segmentation and get texture parameters for the segmented regions.

Figure 14: shadow areas

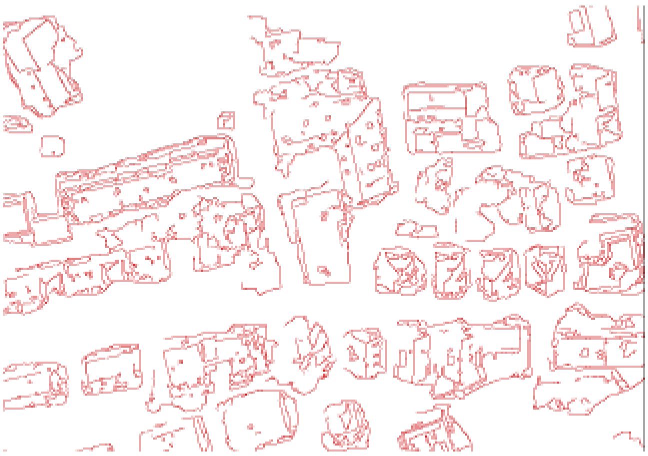

Figure 15: edgemap used for classification

3.6 Shadow

Shadow causes problems, e.g., in line following and segmentation. Shadow is no cue as the previous explained cues; it has to be recognised through combining several basic cues. We use the HSV colour space colour segmentation (Figure 14), edge information and information from the DSM and information about the date and time to derive areas occluded by shadow. At the moment our results have to be improved, because shaded parts of roofs and trees are still misclassified due to their saturation value.

4 REALIZATION OF BUILDING DETECTION

For building detection we extract coarse regions, store them with their bounding box, image primitives, boundary and attributes derived from different cues (Figure 16, 17). The buildings are detected through fusion of multiple cues as colour, edges and texture derived in different colour spaces and DSM data, see Table 1. In a first step we use edge detectors, texture filters and classification algorithms to derive independent low-level image primitives from grey-level and colour images. Edge detection gives us edges with the brightness or colour information in the neighbourhood, colour and texture filtering provides homogeneous textured or coloured regions.

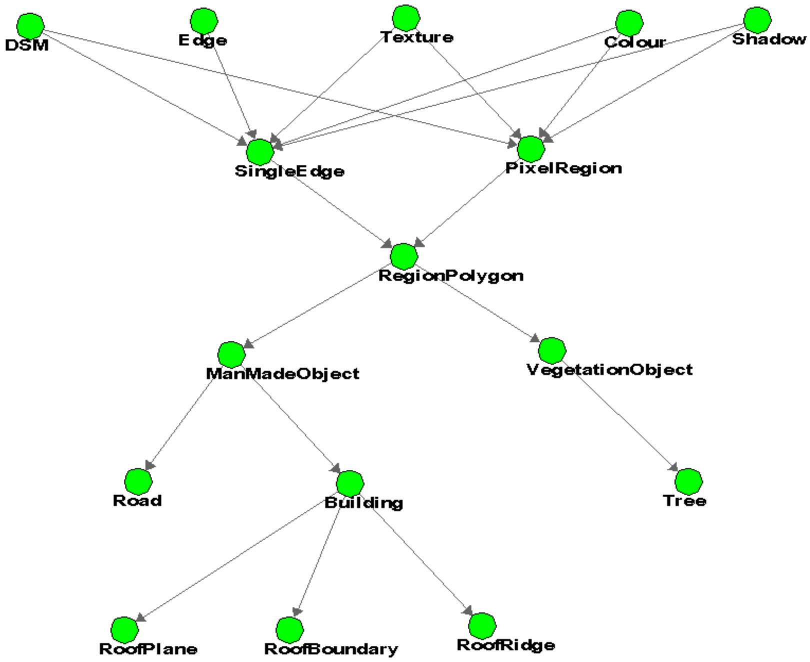

Figure 16: Process of extraction of features for building detection

A DSM is used to identify objects with elevation above ground, derive their coarse shape, height, values of slope and the main axes. For the later fusion of these primitives we derive the quality of the primitives dependent on the cue and the algorithm and store it with the primitive data. A knowledge base defines expected characteristics of man-made objects and buildings e.g. minimal size, homogeneous texture, minimal/maximal height, edge shape and length and supports the separation of man-made objects with elevation from natural objects. Region attributes e.g. texture and colour are incorporated implicitly in classification algorithms to fuse these attributes. Region segmentation is done with different colour channels, elevation and texture information with thresholds for merging and weights for different channels. Region-based and 2D edge information is fused in a region-refinement step providing homogeneous regions with exact boundaries e.g. roof planes, outlines and ridgelines. Depending on the thresholds, roof details could be computed. For later reconstruction steps objects are stored as polygons with attributes about quality, channels, thresholds and algorithms which were used. As a result of building detection the system knows about the location of buildings, relevant edges and relevant regions with their attributes. The resulting attributed image primitives can be used directly to derive 2D and 3D information and give constraints for the further reconstruction step. A control mechanism provides rules (preferences) to weight the primitives in fusion. If there is e.g. a region with high elevation detected in the DSM and in the same region were extended edges detected, this could be an indicator for a building and the system will look there for homogeneous regions. But if it detects extended edges but no elevation, the region might be a road.

4.1 Putting it all together

The size of the assigned attributes is chosen with respect to their relevance for building extraction through all algorithms and normalised to values between 0 and 1. Different cues get different weights. According to Table 1 and Figure 16 we compute for each region dynamically whether a region could belong to a building or not through a basic Quasi-Bayes Network for each region, provided with a set of initial probabilities computed by the regions attributes. An “BuildingRecognitionAgent” gets initial preferences, a table of values for states (=values) of single attributes (Table 2, e.g. “Segment is inside/outside DSM blob” = 1/-1) and decides hierarchically through the set of attributes whether he accepts a region as “BuildingRegion” or as “Non-BuildingRegion”. In Figure 17, e.g. segment B fulfils in each step the preferences of the agent also for segment A; segment C and D will be rejected. A GUI (graphical user interface) allows the user to visualise the results, check single results and their attributes.

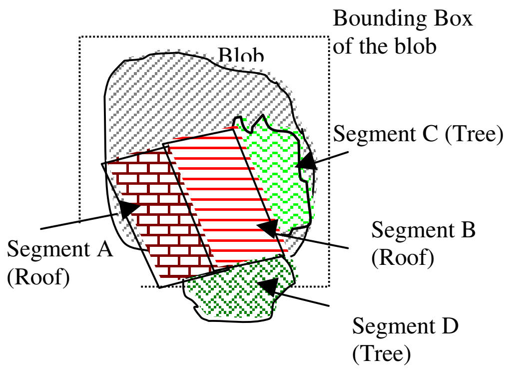

Figure 17: Examples for several segments

Criteria within Blob Bounding Box	Preferences
Range of hue values	Small
Average of hue values	Not green, blue
No of merged regions	Small
No of long straight edges	High
No of “holes” in polygon	Few
Magnitude of edge gradients	High
Homogeneity in texture	High
Edge direction and distance to ridge line	Small
Distance of centre of gravities from blob and segment	Low
No of pixels overlapping blob	High
Quotient perimeter to No of pixels	Low

Table 2: Predefined preferences of the BuildingRecognitionAgent

5 CONCLUSIONS AND OUTLOOK

We showed possible improvements for building detection through combining multiple cues, colour segmentation, edge detection, texture segmentation and blob detection. The resulting attributed image primitives may be used directly to further derive 3D information, e.g. to perform and constrain edge matching with colour edges. We have to test our algorithm for the whole area of the new Zürich Höngg dataset including the old centre of the quarter. The results of our subset datasets showed good results, the detected blobs contained 95% of the buildings compared to a manually measured dataset and visual inspection, within these blobs about 40% of the long strong ridgelines could be found by computing slope and aspect. At the moment blobs indicate the presence of buildings as a first guess, an additional check should be done to detect buildings that could not be detected by blobs. Merging similar colour regions may produce misclassifications and has to be improved, also the last step in combining the features by has to be improved. Our main goal is to refine this system for processing aerial imagery of dense settled areas and also to able to extract building details, this should be able with the same algorithms but adjusted parameters in an iterative processing. Results from point and edge matching (Zhang et al., 2000), (Park et al., 2000) will be integrated.

ACKNOWLEDGEMENTS

The city of Zürich collected the Dataset. Maria Pateraki contributed by measuring the DSM with the software Virtuozo, DEM and reference data were measured with software from Xinhua Wang through his wife Ms. Fu.

REFERENCES

Grün, A., E. Baltsavias, O. Henricsson (Eds.) 1997. Automatic Extraction of Man-Made Objects from Aerial and Space Images (II), Birkhäuser Verlag, Basel.

Grün, A., Wang, X., 1999. CyberCity Modeler, a tool for interactive 3-D city model generation, Proceedings of German Photogrammetry Week, Stuttgart, Germany.

Gülch, E.,Müller, H., Läbe, T., Ragia L. 1998. On the Performance of Semi-automatic Building Extraction, IAPRS Vol 32, Part 3/1, pp. 331-338.

Haralick, R.M., Shanmugam, K. and Dinstein, I. 1973. Textural features for image classification. IEEE Transactions on Systems, Man, and Cybernetics, 3(6), pp.610-611.

Henricsson, O. 1996. Analysis of image structures using color attributes and similarity relations, PhD thesis No. 11663, ETH Zurich.

Henricsson, O. 1998. The Role of Color Attributes and Similarity Grouping in 3-D Buiding Reconstruction, Computer Vision and Image Understanding, Vol. 72, No. 2, November, pp. 163-184.

Mason, S., Baltsavias, E., Stallmann, D., 1994. High Precison Photogrammetric Data Set for Building Reconstruction and Terrain Modelling, Internal Report, Institute of Photogrammetry and Geodesy.

Niederöst, M., 2000. Reliable reconstruction of buildings for digital map revision, accepted for IAPRS, Vol. XXIII, Amsterdam 2000.

Park, H.J, Zimmermann, P. , 2000. Colour Image Matching for DTM generation and house extraction, accepted for IAPRS, Vol. XXXIII, Amsterdam 2000.

Scholze, S.,Moons, T., Ade, F.,Van Gool, L., 2000. Exploiting colour for edge extraction and line segment stereo matching in high resolution aerial imagery, accepted for IAPRS, Vol. XXXIII, Amsterdam 2000.

Skidmore, A.K., 1989. A comparison of techniques for calculating gradient and aspect from a gridded digital elevation model, Int. J. Geographical Information Systems, 3 (4): 323-334; London, New York, Philadelphia.

Zhang, C., Baltsavias, E. 2000. Knowledge-based image analysis for 3D edge extraction and roadreconstruction, accepted for IAPRS, Vol. XXXIII, Amsterdam 2000.

Zimmermann, P. 2000. Description of the Dataset of Zürich Höngg, A new high precision dataset for building reconstruction and terrain modeling, Internal Report, Institute of Geodesy and Photogrammetry, ETH Zürich.