A SEMI-AUTOMATIC IMAGE-BASED MEASUREMENT SYSTEM

A. Reiterer

Institute of Geodesy and Geophysics, Engineering Geodesy
Vienna University of Technology
Gusshausstr. 27-29 / E128-3, A-1040 Vienna
alexander.reiterer@tuwien.ac.at
http://info.tuwien.ac.at/ingeo/

KEY WORDS: Image-based Measurement System, Videotheodolite Measurement System, Knowledge-Based System, Image Processing, Interest Operator.

ABSTRACT

Image-based measurement systems have been used in various surveying applications for several years. Unfortunately, most of such systems required artificial targets defining object points. To overcome this restriction the texture on the surface of the object can be used to find interesting points. However, well-trained "measurement experts" are required to operate such a measurement system. In order to make such systems easy to use even for non-experts, we extend it by a knowledge-based component which supports the operator. Automatic decision making can be done on the basis of features extracted from the images; we use histogram and Haralick features. We have conducted extensive experiments with the knowledge-based system on about 120 pictures showing different kinds of buildings. The system yields good results and shows a reasonable performance. The relative small number of necessary rules would permit to implement the whole knowledge base as a embedded system in the videometric system. We report on the architecture and functionality of the respective knowledge-based system, its development stage and the promising results obtained in experimentation.

1 INTRODUCTION

Videotheodolites, the combination of CCD cameras and motorized theodolites, have been working successfully in several areas of high precision 3D measuring for nearly twenty years. High precision online 3D measuring is required for many applications, among others: monitoring of displacements (buildings, produced workpieces, and others), quality control for production lines or hazardous site surveying. In the past online 3D object measuring by means of image-based measurement systems required artificial targets defining the points on the objects to be monitored. To overcome these restrictions at the Vienna University of Technology several research projects have been executed.

The key element of the first developed system (Roic, 1996) was image processing software that supports the operator to find "natural targets". The operator has to choose the image processing steps and to analyse whether the processed images can be used as targets. The result of this first development step was a non-automatic (interactive) measurement system.

The objective of the second step was to develop a semi-automatic measurement system. This was realized by using videotheodolites in a master and slave mode. The master-theodolite scans the object while the slave-theodolite tracks it by automatically searching for homologous regions. Two scanning methods were developed: scanning with a *point detection algorithm* (Mischke et al., 1997) and scanning with different *grid-line methods* (Kahmen et al., 2001).

Recently, research interest in the area of image-based measurement systems has been increased. Most notably the works done by Walser et al. (2003), Walser (2004), Wasmeier (2003) and Topcon (2006). The central topic of all these image-based measurement systems is the calculation of 3D object coordinates from 2D image coordinates for subsequent processing steps.

Our measurement system is a combination of different components: sensors (videotheodolites used as image sensors), a computer system, software (control system, image processing and decision system) and accessories. Image sensors used to capture data are two videotheodolites Leica TM3000V. A videotheodolite has a CCD camera in its optical path. Images of the telescope field are projected onto the camera's CCD chip. It is possible to project the images from the telescope's image plane to the CCD array or to switch to a special wide-angle optical system ($9\times12°$) to produce a general view of the object. The wide-angle view is only used for approximate target finding. The horizontal and vertical axes carrying the telescope and the CCD cameras are driven by motors, which are controlled by a computer. More details about the image sensors can be found in (Mischke et al., 1997).

The disadvantage of such online measurement systems is the requirement for a well-trained "measurement expert" who has to have certain skills and experience to properly handle the complex system. From image capturing to point measuring a series of actions and decision makings have to be performed. Reliable automatic or semi-automatic object surveying will be only possible if all the knowledge about the measurement system is available and included in a suitable decision system.

The main goal of our current development step is to automate different decision makings (up to now done by the user) in the course of the measurement process. This is done by the integration of a *knowledge-based decision system* [1].

In this paper we describe the development of a decision system which supports the operator when making the following decisions:

[1] Programs which emulate human expertise in well defined problem domains are called knowledge-based systems (Stefik, 1998). The advantages of knowledge-based systems in comparison with conventional programming languages, such as Delphi, Fortran and C++ are: (1) the knowledge about the problem domain is separated from general problem-solving knowledge (makes it easier for the knowledge engineer to manipulate this knowledge); (2) not only "hard" knowledge can be represented, but also "loose" knowledge (useful and potentially very profitable); (3) experts-knowledge, existing very often in form of rules, can be captured in this form without converting into forests of data definitions and procedures.

- selection of suitable image preprocessing algorithms,
- selection of suitable point detection algorithms (*interest operators*) and
- point filtering.

Due to the complexity of these tasks, fully automatic decision-making is not operational. For this reason our approach is an automatic decision-system with integrated user-interaction. The application of the developed measurement system is focused on building (facades) displacement-monitoring. The extension to other object types is envisaged.

In an image-based measurement system (such as described here), all decisions have to be done on the basis of the captured image or on values which represent this image. Using the whole image as input is not suited for an online measurement system (because of processing time). For this reason we use appropriate values as input for the decision system. The process for extracting these values from the image is called *image analysis* and will be content of the next section.

2 IMAGE ANALYSIS

2.1 Image and object features

In our work the goal of image analysis is to extract information needed as input for the knowledge-based decision system. Image analysis is one of the most critical tasks and bottlenecks in the processing chain of an online working system; low calculation effort is a basic requirement. For the image analysis procedure we use the *statistical techniques*. Additionally we collect some *object features* by different user-queries. *Statistical techniques* characterize texture by the statistical properties of the grey-levels of the points comprising a surface. These properties are computed from the grey-level histogram by simple mathematical routines. *Statistical techniques* have low calculation effort and are therefore suitable methods for an online system, by which fast execution of image analysis is necessary. We use two types of *statistical image analysis techniques*: *histogram feature* and *Haralick feature extraction*.

Histogram Features: The histogram of an image is a plot of the grey-level values versus the number of pixels at that value. It can be utilized to generate a class of image features (histogram features). The shape of an image histogram provides a large amount of information about the character of the image; e.g. a narrowly distributed histogram indicates a low-contrast image while a bimodal histogram suggests regions of different brightness (Pratt, 1978).

Based on the histogram, several features can be formulated. We use *mean* (M_1), *variance* (M_2) and *skewness* (M_3) to describe the shape of the image histogram. *Mean* is correlated to the brightness of the image; *variance* (M_2) is a measure of the average distance between each of a set of grey-levels and their mean value and is therefore correlated to the contrast of the image; *skewness* (M_3) is a measure of the symmetry of distribution of grey-levels around their mean and gives information about the balance of bright and dark areas in the image. Details about the used histogram features can be found in (Pratt, 1978; Sonka et al., 1999).

Haralick Features: Haralick et al. (1993) proposed 13 measures of textural features which are derived from the co-occurrence matrices, a well-known statistical technique for texture feature extraction. Texture is one of the most important defining characteristics of an image. The grey-level co-occurrence matrix is the two dimensional matrix of joint probabilities $p(i,j)$ between pairs of pixels, separated by a distance *d* in a given direction *r*. It is popular in texture description and builds on the repeated occurrence of some grey-level configuration in the texture. We generate the co-occurrence matrices and consequently the Haralick features for a distance $d = 1$ and given directions $r = 0°$,$45°$, $90°$, $135°$; additionally we calculate the dedicated average values. Therefore we receive 13×5 Haralick Features.

Additional object features: As mentioned above, the application of the developed measurement system is focused on building displacement-monitoring, especially on the monitoring of facades. In addition to the extracted image features (histogram features and Haralick features), further information (of "non measurable" nature) about the image respectively object is collected by different user-queries (e.g.: *What kind of type is the facade?* or *Are there any reflections on the object?*). For each question several answers are available; e.g. for the type of fassade: *old building facade, new building facade, brick-lined facade and steel-glass facade*[2]; or for the strength of existing effects: *none, slight, middle, strong, very strong*.

2.2 Abstraction of image features

To make the extracted numerical image features more suitable for the knowledge-based decision system we use a special fuzzification/abstraction procedure. This procedure translates the input values (image features) into linguistic concepts, which are represented by abstraction ("fuzzy") sets. This technique is not a fuzzification in terms of definition; we use only non-overlapping spring membership functions. The use of such an abstraction procedure permits us to write decision rules in terms of easily-understood word descriptors, rather than in terms of numerical values.

All these collected values form the *working memory* (WM) and are the input for the knowledge-based decision systems. WM is a collection of *working memory elements*, which itself are instantiations of a *working memory type* (WMT). WMTs can be considered as `record` declarations in PASCAL or `struct` declarations in C. An example of a WMT is as follows:

```
(deftemplate Stat_Moments
  (slot nr (type INTEGER))
  (slot M1 (type FLOAT))
  (slot M1_f (type SYMBOL)
    (allowed-symbols v_low low mid high v_high))
  (slot M2 (type FLOAT))
  (slot M2_f (type SYMBOL)
    (allowed-symbols v_low low mid high v_high))
  (slot M3 (type FLOAT))
  (slot M3_f (type SYMBOL)
    (allowed-symbols v_low_pos low_pos mid_pos
      high_pos v_high_pos very_low_neg lown_neg
      mid_neg high_neg very_high_neg)))
```

Stat_Moments is a WMT consisting of seven *slots*, namely *nr*, *M1* (mean), *M1_f* (fuzzy value of mean), *M2* (standard deviation), *M2_f* (fuzzy value of standard derivation), *M3* (skew) and *M3_f* (fuzzy value of skew) respectively. The type of each slot here is INTEGER, FLOAT or SYMBOL. SYMBOL means that a

[2] We have created these four facade types since most of existing buildings (for the central european culture) can be characterized by these types.

symbol can be stored in the slots. The allowed symbols for each of the slots are defined with "allowed-symbols". Type checking is performed during runtime in order to guarantee that the content of a slot satisfies its definition.

After having explained the basic elements for the decision-making process in the next sections we will describe the three developed sub-systems (a system overview is shown in Figure 1):

- image preprocessing,
- point detection,
- point filtering.

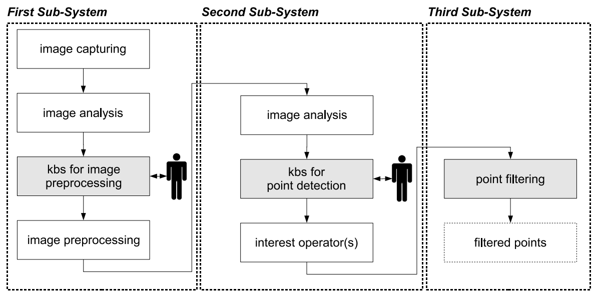

Figure 1. System overview and data flow.

3 IMAGE PREPROCESSING

A necessary precondition for the successful application of algorithms for finding *interesting points*, is the "quality" of the image. Image preprocessing operators work on the lowest level of abstraction, input and output are intensity images. Furthermore, such operators do not increase the image information content, but image preprocessing helps to suppress information that is not relevant to the specific image processing or analysis task.

The following image preprocessing methods have been implemented: histogram equalization, grey-level scaling (image brightening / darkening), median filtering, gauss filtering, edge detection (Sobel, Prewitt and Roberts operator) and thresholding. These algorithms are very simple and widely known in the field of image processing; details about them can be found in (Pratt, 1978; Sonka et al., 1999).

The knowledge-based decision system has to choose a single algorithm or a combination of algorithms for image preprocessing (including the necessary parameters) in order to improve the image for the subsequent application of *interest operators*. This choice is based on the extracted image features. At critical processing steps (e.g., edge detection, median filtering) the user has the possibility to overrule the system decision.

The knowledge which was required to be included in this part of the knowledge base was obtained in different ways: from technical literature (Pratt, 1978; Sonka et al., 1999), previous projects (Kahmen et al., 2001; Mischke et al., 1997) and from experiments. The acquired knowledge was converted into "*If–Then-Statements*" (rules) followed by coding them for the used development tool[3].

[3]The knowledge-based system has been carried out in CLIPS, a productive development tool which provides a complete environment for the construction of *rule- and object-based systems* (Clips, 2006)

The knowledge base (the part for the knowledge-based image preprocessing system) is divided into three components, which are (1) rules for the choice of suitable algorithms for image preprocessing, (2) rules for the predefinition of necessary parameters and (3) rules to define the order of the algorithms.

An example of a very simple rule (for image brightening) is shown in the following:

```
(defrule brightening
   (Stat_Moments (M1_f v_low | low))
   (Stat_Moments (M3_f mid_pos | high_pos | v_high_pos))
=>
  (assert (state (brightening yes))))
```

A rule is divided into two parts, namely the *lefthand side* (LHS) and the *righthand side* (RHS) with "⇒" separating both parts. In the LHS, we formulate the preconditions of the rule, whereas in the RHS, the actions are formulated. A rule can be applied (or *fired*), if all its preconditions are satisfied; the actions specified in the RHS are then executed. In our example here, we check whether there is a WME of type *Stat_Moments* where the contents of *M1_f*-slot is "v_low" or "low" and the one of slot *M3_f* equals "mid_pos", "high_pos" or "v_high_pos".

The used development tool (CLIPS) contains algorithms for the *matching phase*, i.e., the phase where all rules are checked against all working memory elements. The result of this matching phase is the *conflict set*, which includes all rule instances "ready to be fired". A *conflict resolution strategy* selects one rule instance which is actually fired.

Up to now only a small number of image preprocessing algorithms has been implemented. Therefore the knowledge base could be kept propositionally simple and thus easily modifiable and extensible.

We will provide a better understanding of this part of the system by means of an example. The image in Figure 2a shows a highly underexposed image.

(a) (b)

Figure 2. The underexposed image before and after image preprocessing.

M_1	M_2	M_3
55.600/v.low	18.390/v.low	-0.7580/v.high neg.

Table 1. Extracted image features (Histogram Features) for Figure 2a.

	0°	45°	90°
H_1	0.004/high	0.003/high	0.004/high
H_2	21.088/v.low	37.731/v.low	20.636/v.low
H_5	0.443/high	0.374/mid.	0.463/high
H_9	2.797/low	2.919/low	2.781/low

Table 2. Extracted image features (part of the Haralick Features) for Figure 2a.

First of all image analysis has to be done. The relevant image features and their fuzzy values are listed in Table 1 and 2. Due to the calculated image features the knowledge-based system chooses a 3×3 *median filtering* [4] and a *grey-level scaling* (*image brightening*). Now, the knowledge-based system gives the user the possibility to overrule this decision (the user has the option to remove the *median filter* from the *task-list*). For our example we assume that the system decision remains unchanged. The processed image is shown in Figure 2b. The application of *grey-level scaling* results in a brighter image with increased contrast; the *median filter* has reduced the noise by smoothing the image. Useful details, like edges and corners, are now (after image preprocessing) visible. If the user decides to remove the *median filter* from the *task-list* the resulting image is nearly the same but contains more noise.

4 POINT DETECTION

After having improved the visual appearance of an image by image preprocessing, point finding in the image can follow. Processing algorithms which extract *interesting points*[5] are called *interest operators* (IOPs). They highlight points which can be found easily by using correlation methods. There is a huge number of *interest operators* (Förstner et al., 1987; Harris et al., 1988; Moravec, 1977; Paar et al., 2001), however no *interest operator* is suitable for all types of desired point detection. For this reason we have implemented different *interest operators* in our system.

Schmid et al. (2000) differentiate three categories of *interest operators*: (a) *Contour based methods* extract contours in the image and search for maxima curvature or inflexion points along the contour chains; (b) *Intensity based methods* compute measurements directly from grey values that indicate the presence of *interesting points*; (c) *Parametric model based methods* fit a parametric intensity model to the signal.

The algorithms implemented in our system are *intensity based methods*. These methods go back to the development done by Moravec (1977). His detector is based on the auto-correlation function of the signal. It measures the grey value differences between a window and a window shifted in the four directions parallel to the row and columns. An interest point is detected if the minimum of these four directions is superior to a threshold (Schmid et al., 2000). Today there are different improvements and derivatives of the *Moravec* operator. Among the most well-known are the Förstner and the Harris operator, which represent two methods implemented into our system. Additionally, we have integrated the *Hierarchical Feature Vector Matching (HFVM) operator*, a development of the *Joanneum Research* in Graz (Austria). A listing of the mathematical derivation and description of the *interest operators* can be found in literature (Förstner et al., 1987; Harris et al., 1988; Moravec, 1977; Schmid et al., 2000).

As we have explained in the introduction the application of the developed measurement system is focused on monitoring of building facades. For such a process the facade has to be modeled by choosing points in such a way that they characterize the object. In a subsequent process step these points can be used for object reconstruction or monitoring of movements and distortions. Building facades elements (e.g. edges, windows, beams) can be represented by simple line geometry. Therefore the process of object modeling can be reduced on capturing points along such lines respectively intersections of them. The knowledge-based choice of suitable *interest operators*, the order of application and the necessary parameters are fitted according to this.

The knowledge to be included in this part of the decision system was obtained by extensive *knowledge engineering*. Only few evaluation methods for point detection (resp. description) algorithms can be found in the literature, cf. (Baker et al., 1999; Bowyer et al., 1999; Mikolajczyk et al., 2004; Schmid et al., 2000), among which the work by Schmid et al. (2000) on interest operators (IOPs) is of particular importance for this paper. The prevailing methods for evaluation are largely based on subjective evaluation based on visual inspection and ground-truth verification, as well as on objective criteria such as repeatability of information content for images. A drawback of these methods is that they neither account for the formation of the point cloud detected by a point detection algorithm, nor for its localization accuracy. Furthermore, human interpretation limits the complexity of the image used for evaluation.

For these reasons we combine several methods for the evaluation of *interest operators*: visual inspection, ground-truth verification on the basis of good and bad areas (defined by the user), and a new developed evaluation method. The novel criterion is based on distances between sets of points and can be used as a complementary technique to the existing evaluation methods. This technique allows to compare point detection algorithms very easily and, moreover, in an objective but strongly application-oriented way. We used about 120 images of building facades for the evaluation. These images are uniformly distributed over different facade-types. Additionally to these evaluation methods a runtime analysis was done. Details about the whole evaluation process can be found in (Reiterer et al., 2006).

The collected evaluation results are the basis for the formulated rules; also this rule base is divided into three groups of rules (see Section 3). An example of a simple rule (for the Förstner operator) is shown in the following (part of the rule):

```
(defrule foerstner
  (or (or (and (1_Haralick (H1_0_f low | v_low | mid))
               (5_Haralick (H5_0_f low | v_low | mid)))
          (and (1_Haralick (H1_90_f low | v_low | mid))
               (5_Haralick (H5_90_f low | v_low | mid))))
                ... ))
=>
  (assert (iop (foerstner yes))))
```

In the following we will continue the example shown in Section 3. After image preprocessing the image properties have changed, so that, before the knowledge-based system chooses a suitable algorithm for finding *interest points*, image analysis has to be done again. The resulting image features for the image shown in Figure 2b are listed in Table 3 and 4.

M_1	M_2	M_3
126.983/mid.	41.337/mid.	-0.761/high neg.

Table 3. Extracted image features (Histogram Features) for Figure 2b.

On the basis of the implemented rules the following *interest operators* are selected[6]: the Förstner operator (with $q_{min} = 0.2$, $W_{min} = 360$ and $R = 3$) and the Harris operator (with $\sigma = 1.0$, $N_0 = -0.04$ and $corn_{min} = 0.0018$). Note: To undertstand the selection by the rules please compare the above shown rule.

[4] Median filtering was selected due to the existence of noise in the image – this noise was detected by the Haralick Features (Haralick et al., 1993) and by user-queries (see Section 2.1).

[5] By "*interesting point*" we mean any point in the image for which the signal (the grey values of image pixels) changes two-dimensionally.

[6] A description of the parameter can be found in (Förstner et al., 1987; Harris et al., 1988).

	$0°$	$45°$	$90°$
H_1	0.001/mid.	0.001/mid.	0.001/mid.
H_2	68.037/low	126.912/mid.	61.911/low
H_5	0.363/mid.	0.278/low	0.383/mid.
H_9	3.305/mid.	3.467/mid.	3.283/mid.

Table 4. Extracted image features (part of the Haralick Features) for Figure 2b.

In Figure 3a it can be seen that *interest points* are generally detected on the regular structure of the object, only a small number of isolated single points are detected inside these "structure lines". These points result from local grey-level differences, like "fault-pixels". A more problematic area is the glass window, where many *interest points* are caused by reflections. Changes of parameter values for the *interest operators* would remove the undesirable points on the glass windows, as the desired *interest points*, too (the grey-level differences in this area are the same as those of the "structure lines" of the facade). Such undesirable points can only be eliminated by a suitable point filtering technique.

5 POINT FILTERING

As we have described above, the knowledge-based system chooses a suitable *interest operator* on the basis of the extracted image features. In normal cases not only one *interest operator* will be selected, but a group of suitable algorithms. Therefore in the course of finding interest points, more than one *interest operator* will be applied. This results in single lists of *interest points*.

The point filter, which will be described in the following, has the task to unite the single lists, to weight each points according to certain rules and to remove points respectively point groups with a certain weight.

Point reduction is necessary since, in spite of choosing suitable algorithms for image preprocessing and for *point detection*, many undesirable points are detected. Point detection has to be done in such a way, that the extracted points characterize the object in a suitable form. In case of facades the elementary structure can be represented by a simple line geometry. Points detected apart from this line structure (e.g. points inside glass windows) are undesirable and not useful for subsequent process steps, like object reconstruction or deformation analysis.

The filtering process will be done by means of two methods: (1) point filtering on basis of defined rules (knowledge-based), (2) point filtering on basis of user interaction (user-based).

5.1 Knowledge-based point filtering

The knowledge-based part of point filtering is based on (a) the number of *interest operators* which detect the same point and (b) the "property-parameters" obtained from the corresponding *interest operator*.

The *first criterion* is very simple, but effective. The point filter scans all point lists (one point list for each applied *interest operator*) and weights each point in correspondence with the number of *interest operators*, from which this point has been detected. In practice this is a search routine which finds points with the same co-ordinates in different point lists. The weights are fixed on the basis of this simple coherence. The fundamental idea behind this filtering (weighting) process is that important points are detected by more than one *interest operator* (in that case that more than one has been applied).

The *second criterion* is based on "property-parameters" (available in our implementation) obtained from the corresponding *interest operator* for each point. On the basis of these returned values we can formulate several rules for point filtering; in the simplest case thresholding will be used. Points with returned values less than the threshold values get a different weight from points with returned values greater than or equal to the thresholds. In a following process points with low weight can be removed.

5.2 User-based point filtering

The developed interactive point filter allows the user to choose the points or point clouds to be removed. This selection process is realized by means of a graphical user interaction. The user has to draw a rectangular window in the graphical output. Points inside these selected windows will be removed. In a final step the user has the possibility to preserve only points respectively point groups with a certain weight (weighted by the knowledge-based part of filtering).

The resulting *interest points* for our example are shown in Figure 3b.

(a) (b)

Figure 3. (a) *Interest points* detected with the Förstner and the Harris operator; (b) Final result after image preprocessing, application of *interest operators* and point filtering.

It should be noted that the whole developed point filtering technique is a very simple method, but an effective one with a lot of potential for future extensions of the system.

6 DISCUSSION, CONCLUSION AND OUTLOOK

In this work a new decision support system for an online videotheodolite-based multisensor system has been described. The main task of our development has been the automation of different decision makings in the course of the measurement process. The decision support system has been realized with a knowledge-based approach. The separation of domain-knowledge from the reasoning mechanism, one of the fundamental concepts of such an approach, leads to the biggest advantages in comparison to conventional software: the system is easily modifiable and extensible.

To restrict the development process our measurement system has been focused on monitoring of building facades. This object type offers a wide range of structures and can be represented by simple line geometry. Integrated knowledge, examples and simulations have been fitted according to this.

The decision process is based on numerical/fuzzy values which represent the decisive image features. For an online system a fast execution of the image analysis process is necessary – in our system this is done by statistical feature extraction techniques.

The developed system presents a basic approach for an automated online videotheodolite-based multisensor system. In such a system, the degree of automation can be very high, whereas by decision-making, human interaction remains an important part of the workflow even though the amount of decisions done by the user can be reduced considerably to a minimum.

There are still many possibilities to improve the operability of the system. The most interesting and important ones for the future are:

- Improvement of the developed image preprocessing system by integrating more algorithms, like *canny edge detection* or the *Wiener noise reduction filter*.
- The implemented knowledge-based point detection process could be improved by expanding the usability of the *interest operators*.
- The developed point filter has a lot of potential for an improvement of the system. Rules which do not only filter single points (by means of threshold values), but also regard the constellation of *interest points*, like filtering of points inside a glass window, could be implemented.

Beside the improvement of the system, the degree of automation for the whole system should be increased by integrating other sensors in the measurement process. A suggestive extension could be the integration of 3D laser scanners. The data of the different sensors have to be merged by a special data fusion process, which could be knowledge-based. Such a system provides an immense number of 3D data, both from the videotheodolite system and from the laser scanner. This point cloud may be reduced by filtering, even if not very effective. A new approach could build on cognitive vision.

Such a new measurement system would benefit from the efficiency of the 3D laser scanner, from the image information captured by the videotheodolites, and from the automation of decision processes basing on cognitive vision. The result would be a (semi) automated measurement system which is able to act and react in a known environment to unknown situations.

REFERENCES

Baker, S. and Nayar, S., 1999. Global Measures of Coherence for Edge Detector Evaluation. In: Proceedings of the Conference on Computer Vision and Pattern Recognition, Colorado, pp. 373-379.

Bowyer, K., Kranenburg, C. and Dougherty, S., 1999. Edge Detector Evaluation Using Empirical ROC Curves. In: Proceedings of the Conference on Computer Vision and Pattern Recognition, Colorado, pp. 354-359.

CLIPS-Project-Page, 2006. http://www.ghg.net/clips/CLIPS.html.

Eiter, T. and Mannila, H., 1997. Distance Measures for Point Sets and Their Computation. Acta Informatica, 34/2, pp. 109-133.

Förstner, W. and Gülch, E., 1987. A Fast Operator for Detection and Precise Location of Distinct Point, Corners and Centres of Circular Features. In: Proceedings of the ISPRS Conference on Fast Processing of Photogrammetric Data, Interlaken, pp. 281-305.

Haralick, R.M. and Shapiro, L.G., 1993. Computer and Robot Vision, 1st Edition, Addison-Wesley Publishing, New York.

Harris, C. and Stephens, M., 1988. A Combined Corner and Edge Detector. In: Proceedings of the 4th ALVEY Vision Conference, University of Manchester, England.

Kahmen, H., Niessner, A. and De Seixas, A., 2001. 3D Object Sensing Using Rotating CCD Cameras. In: Digital Image Analysis, Springer, New York, pp. 465-487.

Mikolajczyk, K., Tuytelaars, T., Schmid, C., Zisserman, A., Matas, J., Schaffalitzky, F., Kadir, T. and Van Gool, L., 2004. A Comparison of Affine Region Detectors. International Journal of Computer Vision.

Mischke, A. and Kahmen, H., 1997. A New Kind of Measurement Robot System for Surveying of non Signalized Targets. In: Optical 3-D Measurement Techniques IV, Herbert Wichmann, Karlsruhe.

Moravec, H., 1977. Towards Automatic Visual Obstacle Avoidance. In: Proceedings of the International Joint Conference on Artificial Intelligence 584.

Paar, G. and Bauer, A., 1996. Cavity Surface Measuring System Using Stereo Reconstruction. In: Proceedings of SPIE Conference on Intelligent Robots and Computer Vision XV, Boston.

Paar, G., Rottensteiner, F. and Pötzleitner, W., 2001. Image Matching Strategies. In: Digital Image Analysis, Springer, New York, pp 393-410.

Petrou, M. and Bosdogianni, P., 1999. Image Processing – The Fundamentals. 1st Edition, John Wiley and Sons, New York/Chichester/Brisbane/Toronto.

Pratt, W.K., 1978. Digital Image Processing. 1st Edition, John Wiley and Sons, New York/Chichester/ Brisbane/Toronto.

Reiterer, A., Eiter, T., 2006. A Distance-Based Method for the Evaluation of Interest Point Detection Algorithms. International Conference on Image Processing (ICIP), in review.

Roic, M., 1996. Erfassung von nicht signalisierten 3D-Strukturen mit Videotheodoliten. PhD Thesis, Vienna University of Technology.

Schmid, C., Mohr, R. and Bauckhage, C., 2000. Evaluation of Interest Point Detectors. In: International Journal of Computer Vision 37/2, pp. 151-172.

Sonka, M., Hlavac, V. and Boyle, R., 1999. Image Processing, Analysis, and Machine Vision. 2nd Edition, PWS Publishing, New York.

Stefik, M., 1998. Introduction to Knowledge Systems. 2nd Edition, Kaufmann, San Francisco.

Topcon-Webpage, 2006. The GPT-7000i, http://www.topcon-europe.com.

Walser, B. and Braunecker, B., 2003. Automation of Surveying Systems through Integration of Image Analysis Methods. In: Optical 3-D Measurement Techniques VI-I, Zurich, pp. 191-198.

Walser, B., 2004. Development and Calibration of an Image Assisted Total Station. PhD Thesis, ETH-Zurich.

Wasmeier, P., 2003. The Potential of Object Recognition Using a Servo-tacheometer TCA2003. In: Optical 3-D Measurement Techniques VI-II, Zurich, pp. 48-54.