

U. Stilla<sup>1</sup>

## *Map-aided structural analysis of aerial images*

A syntax-oriented method for a map-aided analysis of structures in aerial images is proposed. First the map must be analysed in order to obtain a suitable representation of its knowledge content. A special kind of graph, a so-called image-description graph, is the result of this map analysis. The knowledge of the map, represented on different description levels, is used to control the search process during the image analysis. Based on this knowledge, expectations for attribute values of image objects are defined. Generated objects are assessed relative to the expectations of the map and the object model. A set-oriented selection method is applied to deduce the processing priority using these two assessments. Expected objects are preferably processed for building up more complex objects. Thus the map-aided analysis can be used to reduce the processing time for a verification task.

### 1. Introduction

The presented work is part of the research project "Analysis of aerial and satellite images for automatic determination of the ground sealing of urban areas". The aim of this project is to attain a *context-oriented* automatic image analysis, which comprises in addition to multispectral and texture classification, an image structure recognition by using two- and three-dimensional models. For context information we use cartographic representations of the image area (Fig. 1).

In the first step of the project the objects already known from the map are being verified in the aerial image. In a second step it is intended to classify those structures which could not be verified by using the map.

In the field of pattern recognition, knowledge-based methods are increasingly used for the analysis and description of aerial imagery (Nagao and Matsuyama, 1980; McKeown et al., 1985; Nicolin and Gabler, 1987). A special group are structure-oriented hierarchical methods, which build up structure hierarchies by composing complex structures from less complex structures. Using this approach the analysis proceeds step by step, with constant reference to the patterns being analysed, producing intermediate results of increasing degrees of abstraction.

This paper describes the procedure of verifying objects with the aid of structure analysis and the way the content of the map corresponding to the aerial image is used.

Initially, the map must be analysed in order to obtain a suitable representation of its knowledge content. The result of the *map analysis* is then used to control the following *aerial image analysis*. For the automatic analysis of complex structures as they occur in aerial images, we propose a syntax-oriented structural image analysis.

### 2. Blackboard-based production system BPI

For structure analysis of complex scenes the blackboard-based production system for image understanding (BPI) (Lütjen, 1986) is suitable.

#### 2.1. Production system

A production system consists of three basic components: a database, a set of production rules and a control unit. A production rule, or production, is a statement in the form: IF this *condition* holds, THEN this *action* is appropriate. Execution of the action will result in a change of the data contained in the data base. A control unit controls the system's activity and has the special task of deciding which production with satisfied condition part to fire next.

The process of building up more complex structures from less complex structures using such productions can be described by a rewriting system.

<sup>1</sup>Research Institute for Image Processing and Pattern Recognition (FGAN/FIM), Eisenstockstr. 12, 76275 Ettlingen, Germany.

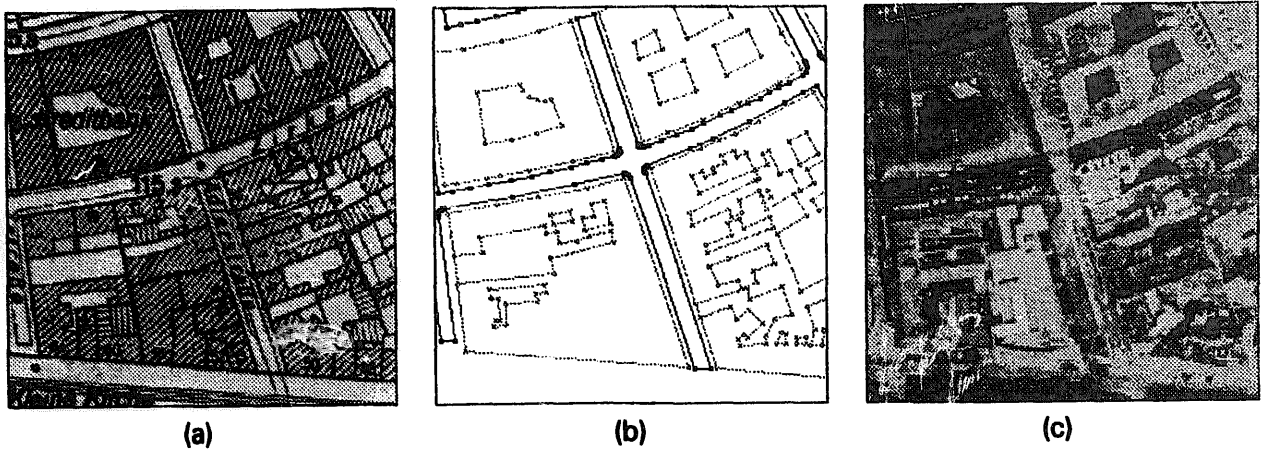


Figure 1. Town section. (a) Mapsheet DGK5. (b) Digitized map. (c) Aerial image. [(a) and (b) Permission granted by Landesvermessungsamt, Baden-Württemberg, Aussenstelle Karlsruhe. (c) Permission granted by Stadtverwaltung Karlsruhe.]

With reference to formal languages the rewriting system may be determined by a Grammar  $G$ . A formal grammar is defined by a 4-tuple:

$$G = (S, V_n, V_t, P)$$

where  $S$  is a set of start symbols (target objects),  $V_n$  is a set of non-terminal symbols (partial objects),  $V_t$  is a set of terminal symbols (primitive objects) and  $P$  is a set of rewriting rules (productions). The objects are attributed and represent certain structures. The productions determine how a given set of objects is transferred into a set of more complex objects.

On the analogy of string grammars we may say that an image content is parsed (bottom up) by the process of image analysis. Instead of examining the concatenation as done by parsers for string grammars, we examine the topologic or geometric relation of objects in the condition part of a production. Therefore, a production rule is written in the form:

$$P_i : X \wedge Y \odot \xrightarrow{i} Z$$

This means that, if an object of type  $X$  and an object of type  $Y$  fulfil the predicate  $\odot$ , then an object specific generative function  $\xrightarrow{i}$  is carried out which produces an object of type  $Z$ .

Starting with primitive objects, a target object can be composed step by step using the productions repeatedly. The general interaction of productions and the stepwise transfer of objects into objects with a higher abstraction level can be depicted by a production net (e.g. Fig. 5). The compositions

for the *actual* objects (instances) are recorded with the aid of pointers and can be illustrated by a derivation graph (e.g. Fig. 6).

## 2.2. Blackboard architecture

The blackboard architecture was first suggested by Newell (1962) and the basic idea can be demonstrated by a metaphor (Velthuisen, 1992): "Imagine several specialists with different expertise required to solve a problem cooperatively. The specialists are gathered around a blackboard for communicating intermediate results and hypotheses. As soon as data has been written on the blackboard by one of the specialists, other specialists may decide that they can use that information to contribute to the problem-solving process. These specialists in turn write their own findings on the blackboard, thus providing possibly interesting information for other specialists to continue the problem-solving process. The process so continues until either a solution to the problem is found or no specialist is able to proceed on any of the data present on the blackboard."

The basic components of a blackboard system are displayed in Fig. 2. In the blackboard system so-called *knowledge sources* are given the role of specialists and a global database is used as *blackboard*. A *control unit* given the role of a discussion leader is added to the system.

In the BPI system the global database is realized in the form of an associative memory and the control unit contains a priority-ordered queue. The knowledge sources are implemented by inde-

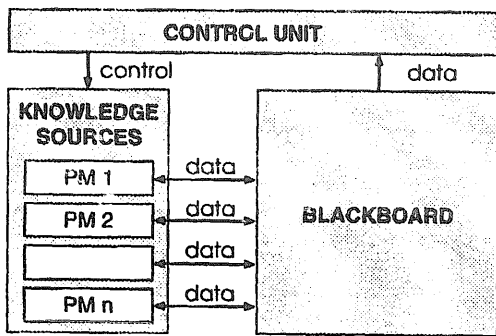


Figure 2. Blackboard architecture.

pendent processing modules (PMs) which communicate exclusively via the blackboard. All intermediate results (partial objects) remain stored in the blackboard during the analysis.

2.3. Associative memory

An associative memory, realized by special hardware, is used to get fast access to object sets. The memory concept which provides the database in an inverted manner is demonstrated in Fig. 3.

A bit matrix is used to assign attribute values to the objects. The objects (marked by an object index) are registered in the columns and the attribute values (e.g. length = 31, length = 32, length = 33) are registered in the rows. The bitstring (row) attached to an attribute value indicates to which objects this attribute value is assigned. A

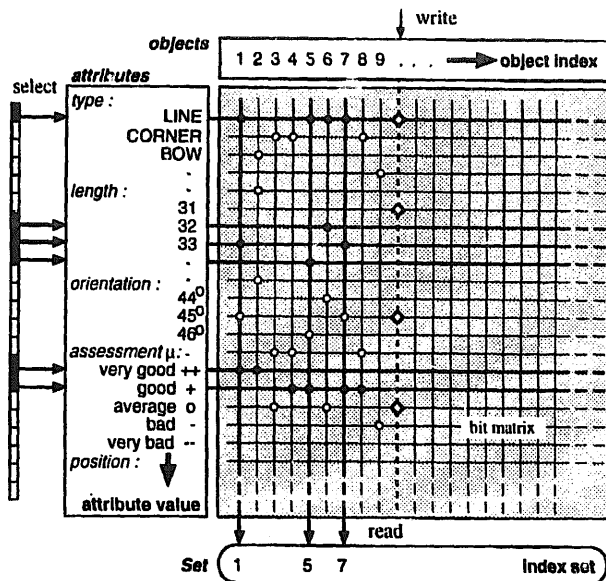


Figure 3. BPI system: associative memory.

new object is written into a free column, setting bits corresponding to the attribute values (e.g. objects of type LINE with length = 31, orientation = 45°, and assessment = average). Object sets with certain features are selected with attribute values, attribute intervals and set operations. Thus all objects may be read in a very efficient way and be available in an index set (e.g. all objects of type LINE with length >31 and with an assessment of good (+) or very good (++)).

2.4. Data flow in the BPI system

The data flow in the BPI system is illustrated in Fig. 4. Initially the primitive objects (X) are stored

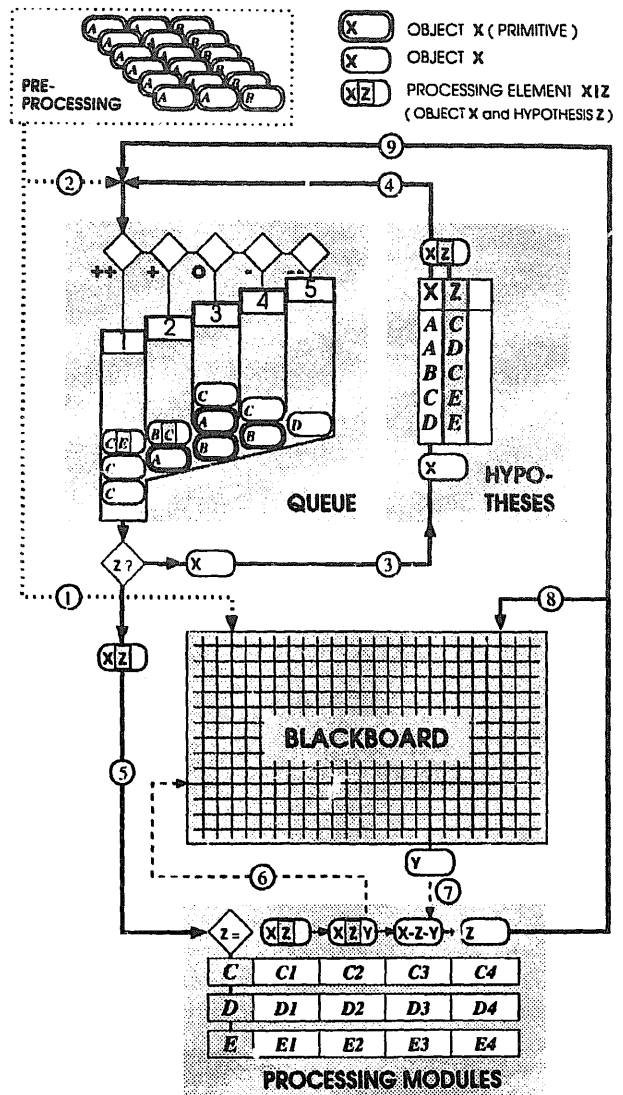


Figure 4. BPI system: data flow.

in the blackboard together with their attribute values (1). Furthermore they are entered as assessed *elements* in the priority-ordered queue (2). Then the data-driven analysis starts with the processing of elements, which are ordered according to their assessment in the queue.

The first element of the queue is removed and it is tested whether a hypothesis *Z* is attached to this element. If this is not the case (3) one or more hypotheses are attached to this element and these pairs of element hypotheses (*processing elements*) are inserted in the queue again (4). For example, for an object *LINE* which gets the hypotheses *part\_of\_LONGLINE* and *part\_of\_STRIPE* two processing elements will be queued.

If a processing element (*X | Z*) is removed from the queue (5) a hypothesis test will be started by triggering a specific verification program (processing module). This module tries to confirm the hypothesis by searching objects (*Y*) in the blackboard which fulfil a specific predicate  $\odot$  (6). If the hypothesis is confirmed (7) a new object (*Z*) will be created and its attribute values will be calculated. Then this object is stored in the blackboard (8) and additionally an element is entered in the queue (9) according to the assessment.

Thus, in different regions of the image, compatible objects grow to more complex objects. The production of partial objects proceeds until the set of nonprocessed elements is empty, or the target objects are constructed. This problem-solving technique is similar to assembling a jigsaw puzzle (Nii, 1986). The image analysis can be understood as a search process which looks for a compatible configuration of objects.

### 3. Map analysis

The source of context information is a digital map (DGK5) with different map coverages (object classes) available in a GIS. A section of the map-sheet DGK5 is shown in Fig. 1a and the manually digitized coverages *street* and *building* are shown in Fig. 1b. The base of the map analysis is represented by a set of straight lines (Fig. 1b). The structure of the digital map is analysed in order to appropriately provide the context knowledge for the aerial image analysis.

A simple production system with eight productions ( $P_1-P_8$ ) (Table 1) is given as an example of the analysis of the object class *street*.

TABLE 1

Set of productions

$P_i$	Objects <i>X, Y</i>	$\odot$	$\xrightarrow{i}$	Object <i>Z</i>
$P_1$	$L \wedge L$	--	$\xrightarrow{1}$	L
$P_2$	$L \wedge L$	-- /-	$\xrightarrow{2}$	SL
$P_3$	$SL \wedge SL$		$\xrightarrow{3}$	S
$P_4$	$S \wedge S$	--	$\xrightarrow{4}$	S
$P_5$	$S \wedge S$	$\angle$	$\xrightarrow{5}$	C
$P_6$	$C \wedge S$	- +	$\xrightarrow{6}$	C
$P_7$	$C \wedge C$	$\in$	$\xrightarrow{7}$	NC
$P_8$	$NC \wedge NC$	$\in$	$\xrightarrow{8}$	NC

- collinear, prolongable
- /- collinear, not prolongable, length > L
- || parallel to each other
- $\angle$  a certain angle to each other
- + ends in
- $\in$  consistent, comprises the same objects

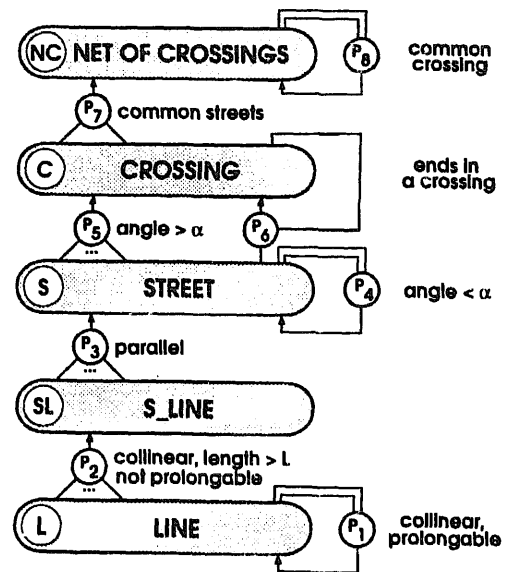


Figure 5. Production net.

Starting with the object primitives *LINE* (L), the objects *S\_LINE* (SL), *STREET* (S), *CROSSING* (C) and finally the object *NET\_OF\_CROSSINGS* (NC) can be built up in the map by applying the productions ( $P_1-P_8$ ) repeatedly.

The connection between the used object concepts and the productions of Table 1 is visualized in a production net (Fig. 5). With the design of a production net for pattern recognition the selection of suitable object concepts and predicates shall cause that the number of generated instances decreases with increasing complexity of concepts. Assuming that, the production net acts as a filter for struc-

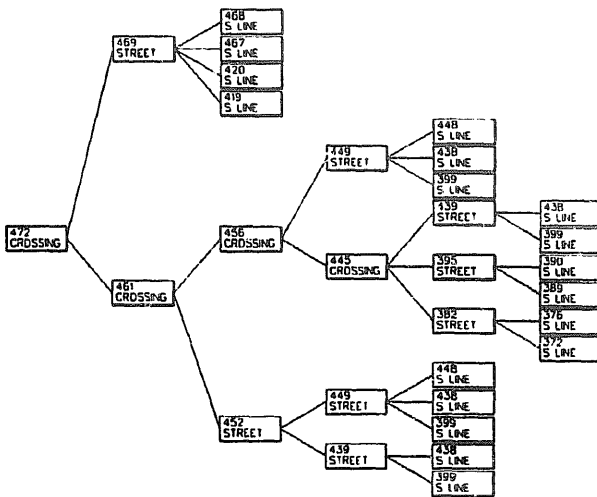


Figure 6. Derivation graph of CROSSING 472.

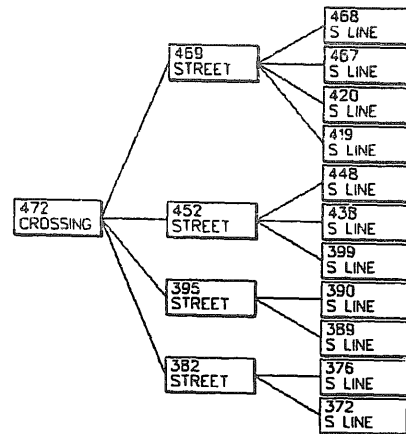


Fig. 9Aa Fig. 9Ab Fig. 9Ac

Figure 7. Image description graph of CROSSING 472.

tures. An unfavourable choice of object concepts and predicates may lead to a *combinatorial explosion* in the search space (Ledermann and Vajda, 1985).

The results of the map analysis are one or more target objects with their corresponding derivation graphs. A portion of a derivation graph is displayed in Fig. 6 for the instance CROSSING 472 in form of the derivation tree with the objects CROSSING, STREET and S\_LINE. Objects LINE are not displayed for reasons of clarity. The generation order of the instances is indicated by the consecutively numbered index.

As all intermediate results (partial objects) are recorded, the actual process of the analysis is reflected in the derivation graph. This graph, however, is of no interest for a *general* description of the map.

The following analysis of the derivation graph results in a reduced graph for the image description, the so-called image description graph (Fig. 7). This graph describes the contents of the map on different levels (image description levels) with the aid of attributed objects and relations among the objects. Figure 9A shows the objects of the image description graph using the image description levels CROSSING (Fig. 9Aa), STREET (Fig. 9Ab) and S\_LINE (Fig 9Ac).

To use the geometric information of the map, a registration between digital map and aerial image is accomplished by determining matching points manually and estimating the parameters of the affine transformation.

#### 4. Image analysis

The analysis of the aerial image is carried out similarly to that of the map, e.g. an object CROSSING is built up by the objects LONGSTRIPE, STRIPE, LONGLINE and LINE. The primitive objects LINE are generated in a preprocessing stage.

##### 4.1. Preprocessing

In the preprocessing stage the scanned image data is transferred into a sequence of binary images, applying a multiple thresholding method. In each binary image the contour lines of the segments are detected and approximated by straight lines using a dynamic split algorithm (Ramer, 1972). These straight lines are stored as primitive objects LINE (Fig. 8) with an assessment  $\mu$  in

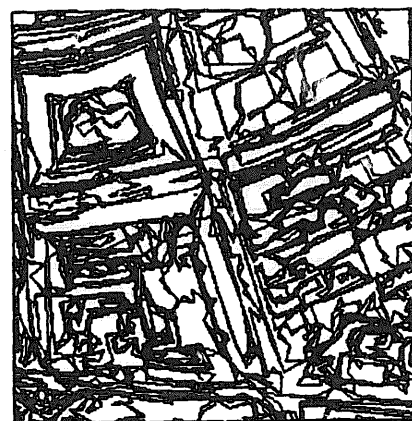


Figure 8. Primitive objects LINE (see Fig. 1c)

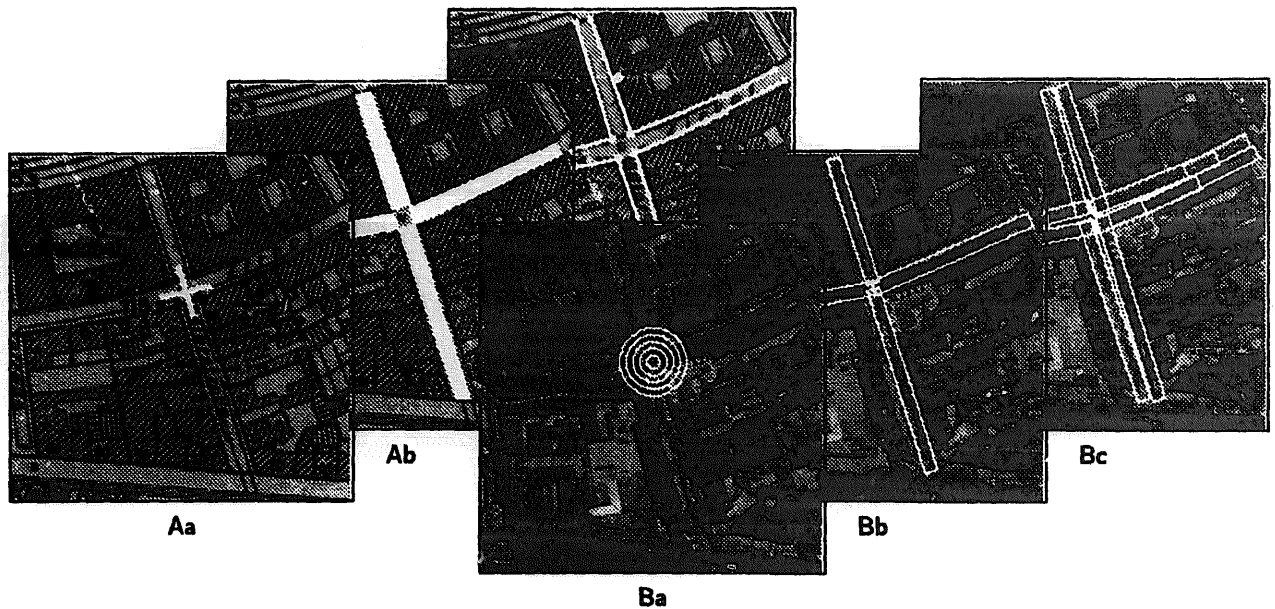


Figure 9. (A) Objects of the image description graph on description levels: (a) CROSSING, (b) STREET, (c) LINE. (B) Derived expectation ranges (expectation areas).

the blackboard. This assessment is deduced from the approximation quality.

#### 4.2. Object models

The object model defines the object concept and may incorporate a set of model parameters, tolerance parameters and assessment parameters. When a hypothesis is tested, objects are searched which fulfil the model conception with a given tolerance. For this reason search ranges are marked in the attribute space. E.g. a processing element LINE carries the hypothesis *part\_of\_LONGLINE*. For the examination of the hypothesis search areas are constructed around the endpoints of triggering LINE and a search interval for orientation is defined. Associatively those objects LINE are selected (see Fig. 3) which have a similar orientation, and one endpoint close to an endpoint of the triggering object LINE. Accordingly, for the hypothesis "part\_of\_STRIPE" objects LINE with similar orientation are searched in a given distance (Stilla and Jurkiewicz, 1991; Fügner et al., 1992).

#### 4.3. Assessment

A newly generated object receives an assessment  $\mu$  (model similarity) by comparing attribute values to the model concept. For example, an ob-

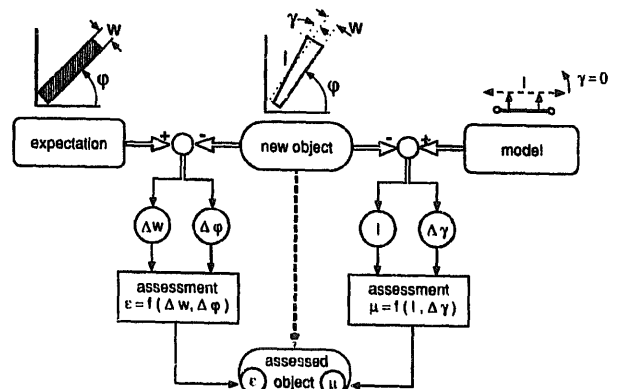


Figure 10. Assessment of a newly generated object STRIPE.

ject STRIPE is assessed with respect to the length  $l$  and the angle deviation  $\Delta\gamma$  to a parallel edge structure (Fig. 10).

By developing new productions we generally attempt to create object-specific assessment functions without *inheritance*. That is, for a newly generated object only the match of the ideal model concept is being assessed, without considering the assessment of the object parts. For example, an object STRIPE can be constructed of objects LINE assessed *very bad* (—). If these objects match perfectly in parallel formation with a given distance the object receives the assessment *very good* (++).

#### 4.4. Control by expectation

The knowledge of the map is used as a supportive aid for the following aerial image analysis. Applying map knowledge in general does not influence the results of the analysis (the target object generated) but merely the processing sequence, namely structures in the aerial image corresponding to map structures are given higher queuing priority.

Based on the knowledge of the image description graph, expectations are defined for attribute values of the objects to be generated in the aerial image. For example, using the presence of objects STREET in the map information, the objects LONGSTRIPE or STRIPE with a special width, orientation, and position are to be expected in the aerial image. Deviations between the attribute values expected from the map and those calculated from the aerial image may occur due to noise. Similar to the model concept an assessment  $\epsilon$  (expectation similarity) concerning the expectation (attribute values) is derived for a newly generated object. Thus, for an object STRIPE the assessment  $\epsilon$  is calculated on the basis of the difference  $\Delta w$  to the expected width  $w$  and the difference  $\Delta\varphi$  to the expected orientation  $\varphi$  (Fig. 10). If the registration is sufficiently exact, the difference to the expected position  $p$  may also be considered.

For the determination of the assessment  $\epsilon$ , tolerance classes are imposed and corresponding intervals of the attribute values (expectation ranges) are defined. Different expectations (from *very high* (+ +) to *very low* (- -)) are assigned to the intervals in order to place a value. An expectation range for the attribute position may be depicted graphically with an area (expectation area).

If a crossing is expected, the attribute position is evaluated to nested expectation areas by means of tolerance classes (+ +, +, 0, -, - -). Given equal probabilities of deviation towards any direction, the borders of the expectation areas form concentric circles. (Fig. 9Ba). Expectations are assigned in ascending order, where the inner circle area gets a *very high* (+ +) expectation and the outer ring gets a *very low* (- -) expectation. The borders of expectation areas with *very high* (+ +) expectation for the positions of middle endpoints of objects STRIPE (STREET) are displayed in Fig. 9Bb and for the endpoints of objects LONGLINE (S\_LINE) in Fig. 9Bc. The expectation ranges are marked in a context

space, which exists beside the object space (Fig. 3) and contains the same attributes. This permits an associative access to sets of elements which possess an expected attribute value.

The priority of the processing will be deduced from the assessment  $\mu$  and  $\epsilon$  by means of a set-oriented selection procedure (Lütjen et al., 1987). The control unit selects from the probably large number of nonprocessed elements those elements, which have the best assessment with respect to as many criteria as possible. Depending on the confidence in the map and in the image the assessments  $\mu$  and  $\epsilon$  can be weighted differently and the scales can also be shifted against each other. If the confidence in the map is very low or a map is not available the analysis may run only with knowledge of the object models.

The image analysis can be stopped for a verification task, if the objects to be examined have been found in the aerial image with sufficient correspondence. Termination criteria are to be defined depending on task and requirement.

## 5. Results

As image data aerial photographs (scale 1:6300) of an urban area (city of Karlsruhe) were used. The transparencies are scanned (RGB and B/W) with a resolution of 25  $\mu\text{m}$  per pixel. For this analysis only the B/W-images were examined.

The preprocessing was done with 16 thresholds, where for image sections of dimension 1024  $\times$  1024 pixel approximately 100,000 area segments and approximately 70,000 primitive objects were generated.

The aim of the analysis was to verify a road network with intersections. Figure 11 shows a partial result of the aerial image analysis. A verified crossing is displayed by the object CROSSING (Fig. 11a), or using lower description levels, by the objects STRIPE (Fig. 11b) and LONGLINE (Fig. 11c) with which the CROSSING was generated.

The comparison of an image analysis with or without map shows that using the map knowledge for the verification reduces the processing time significantly.

## 6. Outlook

In the presented procedure, the map knowledge is only used to accelerate the process of

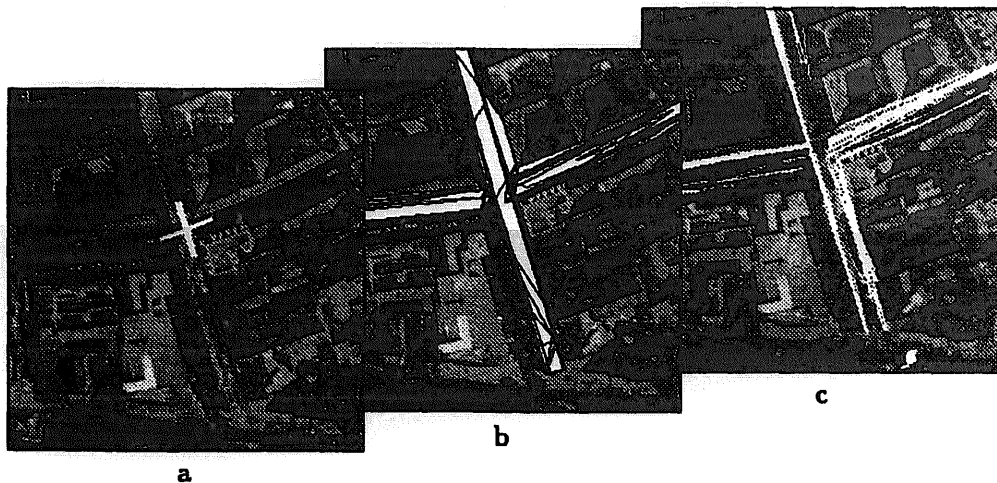


Figure 11. Result of image analysis on description levels: (a) CROSSING, (b) STRIPE, (c) LONGLINE.

verification. Using map knowledge to adjust model parameters specific to every expected object is the subject of further research.

The determination of tolerance parameters, which defines the search ranges, is usually done intuitively by the programmer. In related work with structural image analysis using BPI, the automatic adaptation of these parameters is examined. Therefore a closed loop system applying evolution strategies is proposed (Füger et al., 1994).

### Acknowledgements

This project is funded by the Deutsche Forschungsgemeinschaft (DFG) and is cooperatively done with the Institut für Photogrammetrie und Fernerkundung (IPF), University of Karlsruhe.

### References

- Füger, H., Jurkiewicz, K., Lütjen, K. and Stilla, U., 1992. Ein wissensbasiertes System für die automatische Bildanalyse. In: L.W. Fritz and J.R. Lucas (Editors), International archives of photogrammetry and remote sensing, International Society for Photogrammetry and Remote Sensing (ISPRS), XVIIth Congress, Washington, Bethesda, MD, ASPRS, Vol. XXIX, Commission III, pp. 167–172.
- Füger, H., Stein, G. and Stilla, U., 1994. Multi-populations evolution strategies for structural image analysis. IEEE Conference on Evolutionary Computation (ICEC'94), Orlando, Vol. 1, pp. 229–234
- Ledermann, W. and Vajda, S., 1985. Combinatorics and Geometry. Wiley, Chichester.
- Lütjen, K., 1986. BPI: Ein Blackboard-basiertes Produktionssystem für die automatische Bildauswertung. In: G. Hartmann (Editor), Mustererkennung 1986, 8. DAGM-Symposium. Springer, Berlin, pp. 164–168
- Lütjen, K., Füger, H., Greif, H.J. and Jurkiewicz, K., 1987. Auswahlverfahren für die wissensbasierte Bildauswertung mit dem Blackboard-basierten Produktionssystem BPI. In: E. Paulus (Editor), Mustererkennung 1987, 9. DAGM-Symposium. Springer, Berlin, pp. 290–294
- McKeown, D.M., Harvey, W.A. and McDermott, J., 1985. Rule-based interpretation of aerial imagery. IEEE PAMI, 7: 570–585.
- Nagao, M. and Matsuyama, T., 1980. A Structural Analysis of Complex Aerial Photographs. Plenum, New York.
- Newell, A., 1962. Some problems of basic organization in problem-solving programs. In: M.C. Yovits, G.T. Jacobi and G.D. Goldstein (Editors), Proceedings 2nd Conference on Self-Organizing Systems. Spartan Books, pp. 393–423.
- Nicolin, B. and Gabler, R., 1987. A knowledge-based system for the analysis of aerial images. IEEE Trans. Geosci. Remote Sensing, GE-25: 317–329.
- Nii, H.P., 1986. Blackboard systems. AI Mag., 7: 38–53; 82–106
- Ramer, U., 1972. An iterative procedure for the polygonal approximation of plane curves. CGIP, 1: 244–256.
- Stilla, U. and Hajdu, A., 1994. Map-aided structural analysis of aerial images. In: H. Ebner, C. Heipke and K. Eder (Editors), Spatial Information from Digital Photogrammetry and Computer Vision. ISPRS Commission III Symposium, Munich. SPIE, Bellingham, Vol. 30, Part 3/2, pp. 769–775.
- Stilla, U. and Jurkiewicz, K., 1991. Objektklassifikation mit einem blackboardorientierten Inferenzmechanismus. FIM/FGAN, Ettlingen, FIM-Bericht Nr. 230.
- Velthuisen, H., 1992. The Nature and Applicability of the Blackboard Architecture. PTT Research, Leidschendam.