FORSCHUNGSINSTITUT
FÜR INFORMATIONSVERARBEITUNG UND MUSTERERKENNUNG
FORSCHUNGSGESELLSCHAFT FÜR ANGEWANDTE NATURWISSENSCHAFTEN E.V. (FGAN)

Structural 3D-Analysis of Aerial Images with a Blackboard-based Production System

Preprint

U. Stilla, E. Michaelsen, K. Lütjen

Birkhäuser: In press

Ascona Workshop 1995
Automatic Extraction of Man-Made Objects from Aerial and Space Images
A. Gruen, O. Kuebler (Editors)

April 24 - April 28, 1995
Ascona, Switzerland

Ettlingen - 1995

Structural 3D-Analysis of Aerial Images with a Blackboard-based Production System

Uwe Stilla, Eckart Michaelsen, Karl Lütjen

Research Institute for Image Processing and Pattern Recognition (FIM/FGAN)
Eisenstockstr. 12, D-76275 Ettlingen, Germany
Ph.: +49 7243 99252, Fax: +49 7243 99229
e-mail: usti@gate.fim.fgan.de

Abstract

A model-based method for the analysis of structures in aerial images is described. The knowledge about object structures is represented by a set of productions. Together with the object concepts these productions form a net. Such production systems are implemented in a blackboard architecture. The database is realized in form of an associative memory. A simple example net is used to explain the 3D-analysis and the detection of a high building. The result of the image analysis and partial results obtained by a derivation graph are displayed. Problems with the associative access are discussed and an extended hardware concept is proposed.

1 Introduction

The presented work is part of the DFG 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.

In the field of pattern recognition, knowledge-based methods are increasingly used for the analysis and description of aerial imagery (Nagao & Matsuyama, 1980)(McKeown et al., 1985) (Nicolin & 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 analyzed, producing intermediate results with increasing degrees of abstraction.

A context-oriented recognition of objects may use different information sources, for instance: (i) internal context information e.g. neighbourhood relations to objects of the background (Stilla & Jurkiewicz, 1991), (ii) external context information e.g. a map corresponding to the image (Stilla & Hajdu, 1994), (iii) information from differend images or sensors e.g. stereo vision. Additional knowledge could be used in different ways and on different abstraction levels for the image analysis.

This paper describes the procedure of verifying objects with the aid of at least two images. Our emphasis lies more on the image understanding aspect and evidence fusion than exact measuring of size and coordinates of objects in photogrammetric sense.

2 Blackbord-based Produktion System BPI

For structure analysis of complex scenes the *b*lackboard-based *p*roduction system for *i*mage understanding BPI (Lütjen, 1986) is used.

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 *condition* holds, THEN *action* is appropriate. The execution of *action* will result in a change of the data contained in the data base. The control unit controls the overall system 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. With reference to formal languages the rewriting system may be determined by a Grammar G. Such 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 including S (partial objects), V_t is a set of terminal symbols (primitive objects) and P is a set of rewriting rules (productions). Attributes are assigned to the objects, which represent certain structures. The productions determine how a given set of objects is transferred into a set of more complex objects.

In analogy to string grammars we may say that an image content is parsed (bottom up) by the process of image analysis. Instead of examining concatenation as is 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 may be written in the form:

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

This means that, if an object of type X and an object of type Y fulfil the relation $\odot$, 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. Similar to tabular parsing methods (e.g. in Aho & Ullman, 1972) all intermediate results (partial objects) remain stored in the database during the analysis.

For example a simple rewriting system is determined by

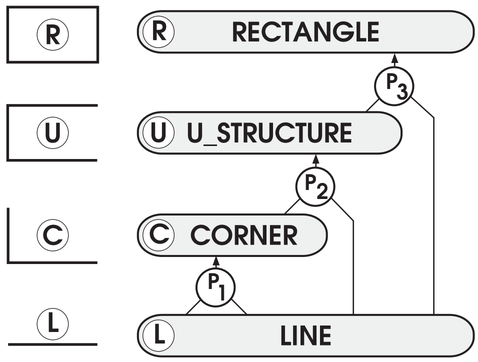
$$G_{Example} = (\{R\}, \{R, U, C\}, \{L\}, \{P_1, P_2, P_3\}).$$

According to $G_{Example}$ and starting with the primitive objects L, the partial objects C, U, R are composed using the productions P_1, P_2, P_3 (see Fig. 1a), object R representing the target object.

P_i :	Objects X,Y			$\odot$	$\xrightarrow{i}$	Object Z
P_1 :	L	$\wedge$	L	∟	$\xrightarrow{1}$	C
P_2 :	C	$\wedge$	L	⊏	$\xrightarrow{2}$	U
P_3 :	U	$\wedge$	L	□	$\xrightarrow{3}$	R

∟	right-angled
⊏	u-structure
□	rectangular

a



b

Fig. 1: Productions. a) Table of productions, b) Production net

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 (Fig. 1b). The compositions for the *actual* objects (instances) are recorded with the aid of pointers and can be illustrated by a derivation graph.

After the analysis, derivation graphs of the target objects can be constructed and used to explain the results. Thus, the subset of primitives, which represents the target objects can be determined. If we compare this subset with the set of primitives, we may say that the production net acts like a filter.

2.2 Blackboard Architecture

In the BPI-System the productions are implemented in a blackboard architecture (Newell, 1962; Nii, 1986). Generally, a blackboard architecture consists of a global database and a set of knowledge sources, which communicate only via the blackboard. In BPI the global database is stored in an associative memory. Knowledge sources are constructed as independent object specific processing modules, which examine a *condition* and execute an *action* of a production.

Systems with a blackboard architecture are essentially *data driven* (Velthuijsen, 1992). One or more hypotheses "*part_of* a more complex object" are attached to an object. An object-hypothesis pair (processing element) triggers a processing module to verify the hypothesis. The hypotheses arise from the production net, so that the analysis proceeds in a goal-directed manner. A control unit, containing a priority ordered queue of processing elements, is added to the blackboard system. Further details concerning the dataflow of the BPI-System are described in previous papers (Lütjen, 1986; Stilla & Hajdu, 1994)

2.3 Associative Memory

An associative memory, realized by special hardware, is used to get fast access to object sets (Davis & Lee, 1986). The memory concept which provides an inverted database is demonstrated in Fig. 2.

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 lines. The bitstring (line) attached to an attribute value indicates to which objects this attribute value is assigned.
A new object is *written* into a free column, placing bits corresponding to the attribute values. In Fig. 2 the four diamond-shaped bits indicate an object of type LINE with length=*31*, orientation=*45°*, and assessment=*average*.

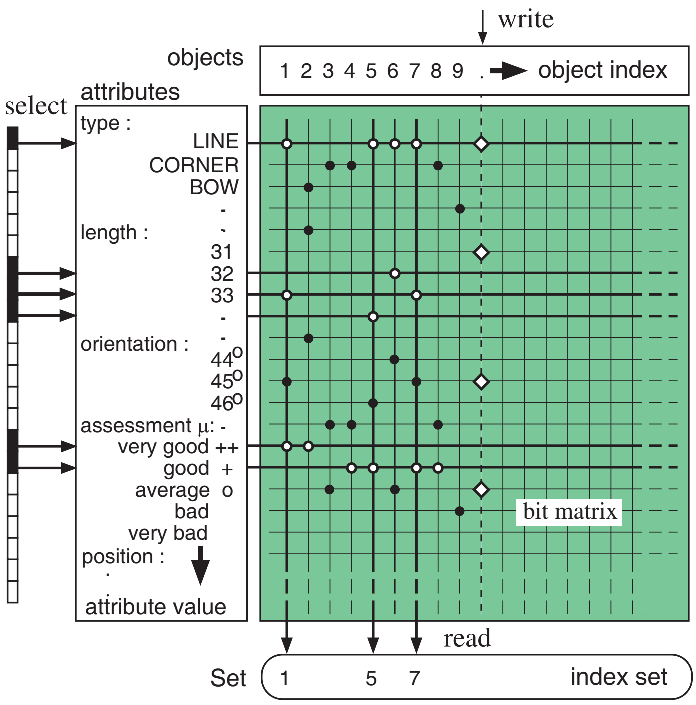


Fig. 2: BPI system: Associative memory

Object sets with certain features are *selected* with attribute values, attribute intervals and set operations (e.g. all objects of type LINE with length >*31* and with an assessment of *good (+)* or *very good (++)*). The hardware *reads* out an index set containing all objects satisfying the selected attributes.

3 Object Model

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 for a triggering processing element (object-hypothesis pair), objects are searched which fulfil the model conception with respect to the triggering object and a given tolerance. For this, search ranges R are marked in the attribute space.

Let us suppose a triggering object LINE carries the hypothesis *part_of*_CORNER. In order to test this hypothesis a search area R_A is constructed around an endpoint e_t=(x_t, y_t) of the triggering object and a search interval R_φ for orientation is defined. The search area R_A is bounded by a circle or simply by a square (Fig. 3). Associatively the set S of those objects type LINE is selected, which have a right-angled orientation φ and one endpoint e=(x, y) close to an endpoint of the triggering object LINE.

$$
\begin{aligned}
R_A &= \{(x,y)\,;\,(x_t-\Delta x \le\; x\; \le x_t+\Delta x)\ \wedge\ (y_t-\Delta y\le\; y\;\le y_t+\Delta y)\}\\
R_\varphi &= \{\varphi\,;\,\varphi_t+90^o-\Delta\varphi\le\;\varphi\;\le\varphi_t+90^o+\Delta\varphi \quad \mathrm{mod}\ 180^o\}\\
S &= \{o\,;\,\mathrm{type(o)}=\mathrm{LINE}\}\ \cap\ \{o\,;\,e(o)\in R_A\}\ \cap\ \{o\,;\,\varphi(o)\in R_\varphi\}
\end{aligned}
$$

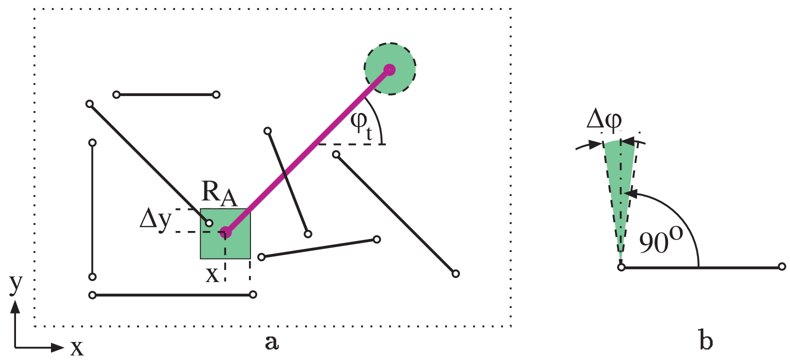


Fig. 3: Search ranges for object CORNER. a) search area, b) search interval

By applying the production P_1 (Fig. 1) repeatedly to a given set of n objects LINE a set of m objects CORNER is produced. In general $n(n-1)/2$ pairs of objects LINE have to be formed and tested with the relation of P_1. If the geometric relations make up a strong constraint m should be much smaller than n. In this case the number of tests can be reduced if only those object pairs are examined which are likely to fulfil the relation. Attaching a hypothesis to each object LINE n hypothesis tests have to be carried out. This means, that the computation effort using the associative access decreases from a quadratic dependency of n to a linear dependency of n.

4 Production Net

To explain the interaction of such model-based productions in a production net, a simple model for the 3D-anlysis of aerial images is chosen. Descriptions of more complex object models can be found in (Stilla & Jurkiewicz, 1991). As an example we select a model for the detection of multi-storied building. Using this model we assume that a multi-storied building can be described as a cuboid and that the cuboid surfaces (the same wall and roof surface) are visible in images from different view-points. These two perpendicularly oriented areas form an edge in space (CUBOID_EDGE). The height of this edge is a typical feature for high buildings and can be used to discriminate this object from other objects.

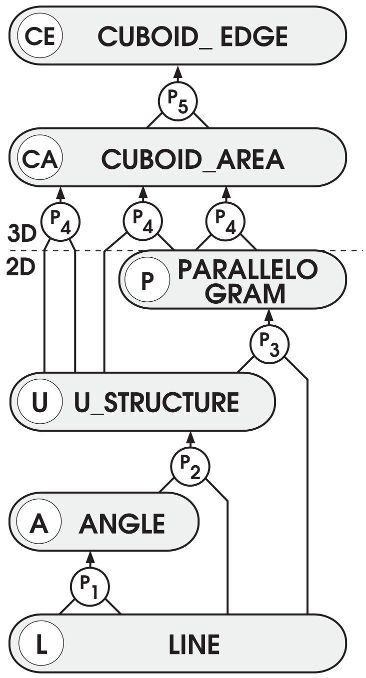


Fig. 4: Production net for object CUBOID_EDGE

A production net for the object CUBOID_EDGE is depicted in Fig. 4. The analysis distinguishes between a 2D- and 3D-scene analysis. At first the 2D-analysis is carried out independently in different images. Starting with the object primitives LINE, the objects ANGLE, U_STRUCTURE and PARALLELOGRAM can be built applying the productions (P_1-P_3). 2D-objects form the primitives of the following 3D-analysis.

The 3D-analysis attempts to find pairs of 2D-objects (U_STRUCTURE or PARALLELOGRAM) which are projections of the same 3D surface. This is done by selecting pairs and examining rays originating at the centre of the projection and passing through the vertices of the 2D-objects. On ideal conditions rays through corresponding object vertices of two images will intersect in the 3D-space.

Due to image noise, processing errors and inaccurate camera parameters the rays may not intersect exactly. Hence, the minimal distance between the rays is calculated. The 2D-objects will be called *not corresponding* if this distance between the rays of pairs of vertices is greater than a given threshold. Additionally, a model-based plausibility check ist carried out (e.g. regarding the position: object points must not be under the earth's surface or behind the camera).

If 2D-objects of different images correspond, the object CUBOID_AREA is generated (P_4). IF objects CUBOID_AREA are oriented in such a way that the surface normals are right-angled and located such that vertices are neighbouring, then a target object CUBOID_EDGE is generated (P_5).

5 Results

Aerial photographs (scale 1:40000, f=153.57 mm) of an urban area (city of Karlsruhe) were used as image data. The transparencies are scanned (B/W) with a resolution of 25 μm per pixel. An exact determination of the camera parameters for each image was carried out by determining matching points manually and calculated iteratively (resection in space).

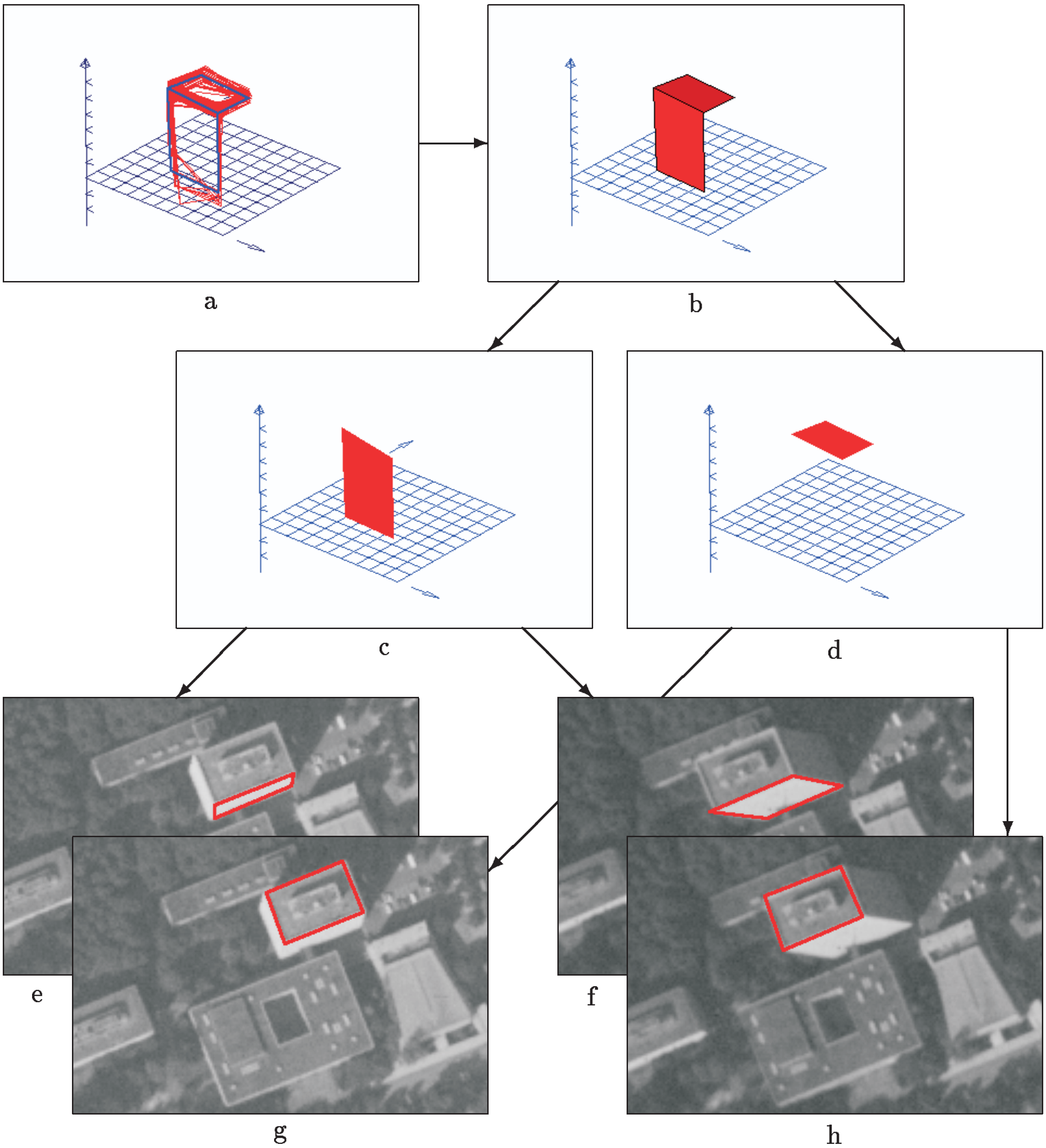


Fig. 5: 3D-analysis. a) set of objects CUBOID_EDGE, b-h) partial results

In the preprocessing stage the scanned image data are 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. These straight lines are stored as primitive objects LINE, attributed with length, orientation, and endpoint coordinates, in the blackboard.

Partial results of a scene analysis are displayed in Fig. 5. The set of objects CUBOID_EDGE, which are found in an expected area is displayed in Fig. 5a. From this set one object is selected (Fig. 5b). The previous objects, which have been obtained by the derivation graph are displayed in Fig. 5c-d (CUBOID_AREA) and in Fig. 5e-h (PARALLELOGRAM).

6 Discussion

Both the development of new models for complex structures and the computationally expensive testing of production nets tends to be quite time consuming. In order to translate the productions into a program code of process modules in an efficient way, software tools are required. On the one hand the programming is supported by a set-oriented programming language for the BPI-system, on the other hand an interactive explanation module (WEEK) is available for debugging and displaying sets and relations of objects. The processing time is reduced by special WPP(word parallel processing)-hardware developed for set-oriented associative access.

In some cases, a geometric condition can not be transformed in such a way, that a direct associative access is possible, for example if all objects LINE have to be determined, whose prolongation intersect a triggering object LINE. To access these objects, the possible intersection points have to be calculated first (Fig. 6a). A similar problem is given in 3D-space when searching for pairs of rays, whose minimum distance is smaller than some threshold (Fig. 6b).

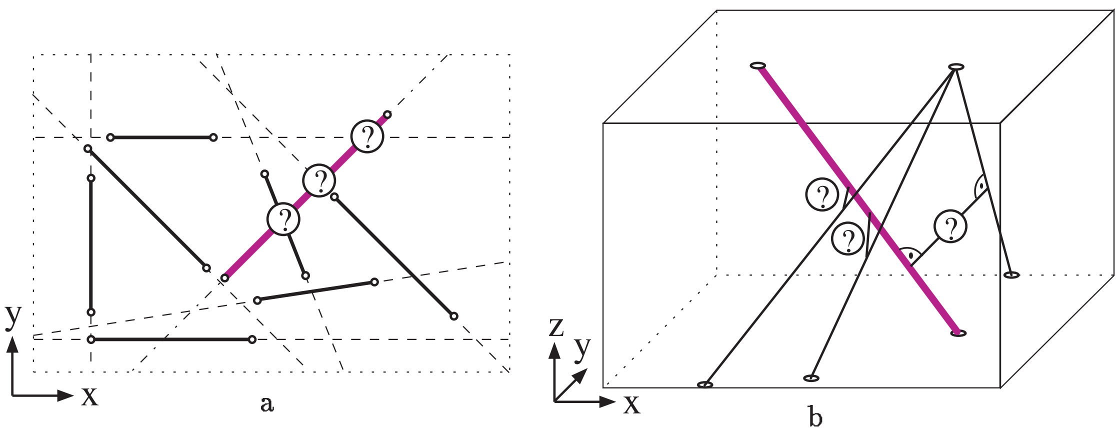


Fig. 6: Access to temporary attributes. a) intersection, b) minimum distance

Hence, the concept for the associative access has to be extended and a new hardware has to be developed. The extended concept for the associative memory ist displayed in Fig. 7. It works in the following way: A function to calculate temporary attributes (e.g. euclidean distance d, intersection (x,y), etc.) has to be selected. The arithmetic operations have to be carried out with a parameter set, which is determined by attribute values of the triggering object. For the calculation, each object has its own processor, which runs with the same program code. For this task SIMD(single instruction multiple data)-processors are suitable. The *selected* objects send the values of specified attributes to their processors and the temporary attributes are calculated for these objects. The results are *written* into the blackboard and available for associative access.

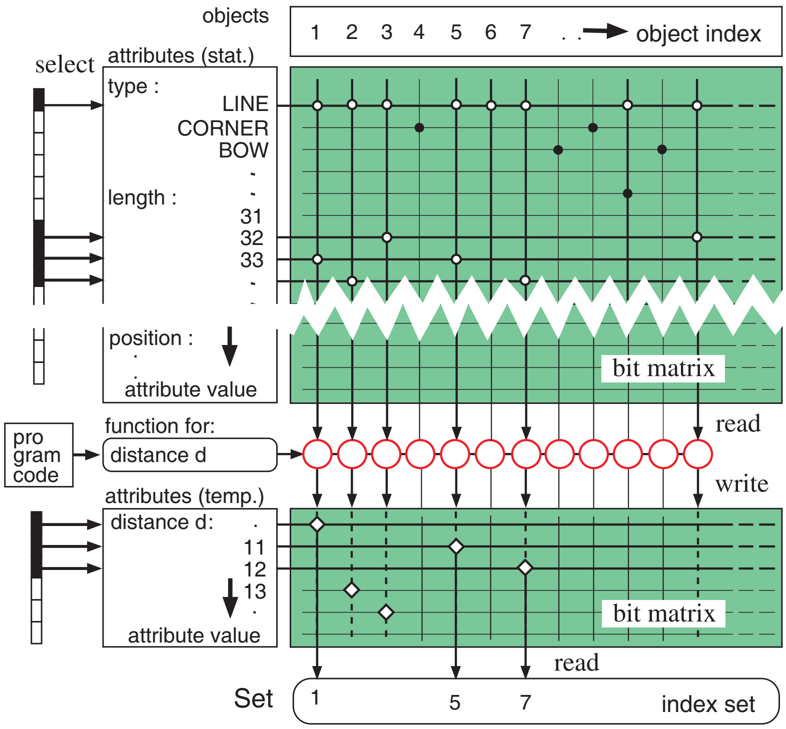


Fig. 7: Extended concept for an associative memory

In the example shown in Fig. 3 objects LINE with neighbouring endpoints to an endpoint of triggering object LINE are searched. Instead of constructing a search area around the endpoint e_t=(x_t, y_t) the distance function $d^2 = (x\text{-}x_t)^2 + (y\text{-}y_t)^2$ is selected and the SIMD-processors are loaded with the program code. All objects LINE respond to the attribute d^2 with their temporary attribute values. The set of objects LINE for which the condition $d_i^2 < d_{max}^2$ holds is selected associativly. Instead of describing search areas or search volumes by sets of pixels or voxels a more *analytical* description by separating hyperplanes and set operations can be used. For each hypothesis test the extended memory concept permits a fast transformation of attributes in a suitable coordinate system.

Acknowledgements

This project is funded by the Deutsche Forschungsgemeinschaft (DFG) and is carried out in cooperation with the Institut für Photogrammetrie und Fernerkundung, University of Karlsruhe.

References

Aho Av, Ullmann JD (1972) *The theory of parsing, translation, and compiling. Vol I: Parsing.* London: Prentice-Hall

Davis WA, Lee DL (1986) *Fast search algorithms for associative memories.* IEEE Transactions on Commputers, C-35: 456-461

Füger H, Stein G, Stilla U (1994) *Multi-populations evolution strategies for structural image analysis.* IEEE Conference on Evolutionary Computation (ICEC'94), Orlando, Vol I, 229-234

Lütjen K (1986) *BPI: Ein Blackboard-basiertes Produktionssystem für die automatische Bildauswertung.* In: Hartmann G (ed) Mustererkennung 1986, 8. DAGM-Symposium. Berlin: Springer, 164-168

McKeown DM, Harvey WA, McDermott (1985) *Rule-based interpretation of aerial imagery.* IEEE PAMI, 7: 570-585

Nagao M, Matsuyama T (1980) *A structural analysis of complex aerial Photographs.* New York: Plenum

Newell A (1962) *Some problems of basic organization in problem-solving programs.* In: Yovits MC, Jacobi GT, Goldstein GD (eds) Proceedings second conference on self-organizing systems, Spartan Books, 393-423

Nicolin B, Gabler R (1987) *A knowledge-based system for the analysis of aerial images.* IEEE Trans. on Geosc. and Rem. Sens., vol. GE-25: 317-329

Nii HP (1986) *Blackboard systems.* AI Magazine, 7: 38-53, 82-106

Stilla U, Jurkiewicz K (1991) *Objektklassifikation mit einem blackboardorientierten Inferenzmechanismus.* Ettlingen: FIM/FGAN, FIM-Bericht Nr. 230

Stilla U, Hajdu A (1994) *Map-aided structural analysis of aerial images.* In: Ebner H, Heipke C, Eder K (eds) Spatial information from digital photogrammetry and computer vision. ISPRS Commission III Symposium, Munich. Bellingham: SPIE, Vol. 30, Part 3/2, 769-775

Stilla U, Quint F, Sties M (1995) *Analyse von Luft- und Satellitenbildern zur automatischen Ermittlung der Bodenversiegelung städtischer Siedlungsbereiche.* Ettlingen: FIM/IPF, DFG-Zwischenbericht II

Velthuijsen H (1992) The nature and applicability of the blackboard architecture. Leidschendam: PTT Research