Bericht 41

Context-Guided Analysis of Scenes with Moving Objects

R. Bertelsmeier, B. Radig

IfI-HH-B-41/78

Hamburg
April 1978

CONTEXT-GUIDED ANALYSIS OF SCENES WITH MOVING OBJECTS

R. Bertelsmeier, B. Radig

ZUSAMMENFASSUNG

Beschrieben wird ein Verfahren, dass es gestattet, Objekte in einer Folge von Fernsehbildern zu identifizieren. Die Bilder werden digitisiert und segmentiert. Die resultierende Relationalstruktur besteht aus Orten, Konturen und Bereichen, die Eigenschaften haben und zwischen denen Beziehungen bestehen. Objektmodelle sind als Relationalstrukturen definiert. Ein hierarchisches Verfahren versucht, die Modelle zu verifizieren; dabei wird jeweils ein Baum von Teilmodellen benutzt, dessen Wurzel das Objektmodell ist. Erlaubte Konfigurationen von Objekten werden durch ein Szenenmodell beschrieben. Es wird im Verlaufe der Interpretation von Bildern aus der Folge durch Verwerten von Interpretationsergebnissen verfeinert.

ABSTRACT

An algorithm is presented which identifies objects in a sequence of TV-images. The images are digitized and segmented. The resulting relational structure consists of points, contours, and regions with relations between them and properties for each of them. Object models are defined as relational structures, too. To verify a model, a hierarchical process employees a tree of submodels whose root is the object model. Possible configurations of objects are prescribed by a scene model. Its accuracy increases by incorporating knowledge from the interpretation results of previous frames.

1. RELATED WORK

In the past there has been considerable effort to describe scenes with various complexity. It began with some heuristic programs designed for scenes of the "Blocks World" [1,2]. The next step was the formulation of explicit models[3] and the filtering approach of Waltz[4]. Yakimovsky[5], Ohlander[6], and Shirai[7] extended the scope of scene analysis to more complex scenes which are some limited subset of real world scenes.

The necessity to develop more formal methods for segmenting, modeling and interpreting scenes was stressed by Barrow et al.[8] at the same time.

The description of moving objects in image sequences of natural scenes was neglected up to now. One reason may be the enormous mass of data presented by real TV pictures, another the still unsolved problem of reliable and flexible analysis of complex static scenes. Chow and Aggarwal[9,10] reported some work on tracking a few simple objects (polygonal or curvilinear outline) in sequences of scenes. Potter tried to segment pictures into stationary and moving sections by comparing pictures, taken from a temporal sequence, with respect to their graylevel properties [11]. Nagel et al. [12] extended the method of comparing low level attributes of an image sequence in order to get a description of stationary and nonstationary image components. From this description they want to generate object models without using scene dependent knowledge. Badler[13] and Tsotsos[14] withdraw from the jungle of segmenting real world images by starting from an ideal symbolic description of pictures in terms of objects, properties, and relations. Backed up by this perfect human "low level" process their system develops an english like description in motion verbs and attributes. Lately Price[15] reported on comparing pictures to describe changes in the observed scene whose descriptions are based on regions and their attributes.

2. SYSTEM APPROACH

In our approach we describe scenes represented as a sequence of TV-frames which contain objects in motion. It is based on the assignment of a new property to the spatial context: the variation in time. So recognition of moving objects may benefit from the conceptual methods developed to utilize object context in static scene analysis. Moreover assuming a sufficiently slow varying scene configuration, its prediction is used to guide the analysis for the second and following frames. This results in a more efficient and more confident interpretation of the whole scene sequence.

Our system, which is now in the implementation phase, provides simple real objects (boxes, balls, flowers, jumping jack , etc.) as input for a high resolution TV-frontend of a process computer which is a satellite of our main computer. The digitized picture is segmented, the result is stored as a relational structure - the sketch - of primitive elements: points, regions, and boundaries. Two-dimensional models of objects are formulated as a hierarchy of submodels, which are coupled at the lowest submodel level to the primitives of the sketch. The configuration of model objects is described by a scene model, which allows to formulate time-variations of properties of and relations between objects. The interpretation is done by a general match process, which finds the common substructures of model objects and sketch. So even imperfect representations of real objects in the sketch can be recognized. The hierarchical organisation of the model objects cuts down the combinatorical explosion.

The first frame is interpreted bottom up instantiating substructures and finally top model objects. A consistent - with the scene model - configuration of object instances (a scene instance) serves as starting point for the top down search in the next frame. If the scene configuration does not change, this technique will obviously give fast interpretation of all subsequent frames. If there are some changes, however, the system attempts to explain properties and relations which are not immediately verifyable (e.g. position) by assuming motion. If the motion hypothesis fails the system may reinterpret (partially) the current or previous frames. On subsequent observation of the evolving scene, a scene instance will become more confident and the interpretation process speeds up. So the continuous variations of spatial context caused by the motion of objects is used for the efficient recognition of objects by deriving a sequence of relevant scene descriptions, refining the instance of a scene model. The temporal development of object configuration is used to determine velocity properties and to get an insight into the three-dimensional structure of the scene (occlusion).

3. SYSTEM STRUCTURE

3.1 Definitions

In order to describe the components and their tasks in our system we introduce some nomenclature to avoid confusion in naming real world objects and their various representations in the system.

Picture: 256 by 191 array of pixels.

Pixel: Mean gray level G (0<=G<=255) and its variance at the position i,j (0<=i<=255, 0<=j<=190). Mean and variance are computed from 2x3 readings at high resolution (512x573) from the digitized video signal of a commercial video camera.

Scene: Part of the miniworld, which is seen by the camera.

Miniworld: Set of all configurations of a finite number of simple real world objects.

Real object: Opaque object with simple contours and surfaces.

Sketch: Relational structure: carrier elements are position, region, contour, straight line, and arc; relations are unary (properties) and binary (spatial context).

Picture object: Subset of the picture representing a real object at pixel level. The camera may generate different picture objects from the same real object in different frames due to noise and varying illumination.

Sketch object: Substructure of the sketch representing a picture object.

Model object: Relational structure to describe a prototype of a class of sketch objects, which represents the same real object.

Model: Set of model objects and a hierarchy of mappings between them, defining a forest of constructions from the primitive elements (terminals) to the top model objects (roots). Each hierarchical level may introduce new relations, which are computable using relations at lower levels. The relations at terminal level are computed from the sketch.

Top model object: Does not serve as a part of another model object.

Primitive element: Relates the bottom level of the model hierarchy with the sketch.

Scene model: Structure of scene objects, which describes acceptable configuratons and motion states of scene objects.

Scene object: Is linked to a top model object and provides the facility to describe the variation of relevant properties and spatial relations in time for this object.

3.2 TASKS

During the process of interpretation, different tasks can be identified. This leads to a decomposition of the system and serves as a frame for its modular implementation:

1. Segmentation of the picture into regions and boundaries. Regions are defined as the interior of closed boundaries (contours). The segmentation is based on Yakimovsky´s approach [23].

2. Symbolic representation of segmentation which results in a relational structure, the sketch.

3. Formation of sketch objects by searching for structural similarity between model objects and substructures of the sketch.

4. Spatial context verification using the scene model. This eliminates instances of top level objects, and solves ambiguities which arise when sketch objects have been bound to several model objects in the hierarchical forest.

5. Verification of temporal context. By observing subsequent pictures of the scene a hypothesis of object motion can be generated, which in turn may aid the interpretation of temporarily and partially occluded objects or objects with similar properties.

The system executes steps 1. to 4. in the bottom up interpretation phase of the first frame of a sequence. The top down interpretation of following frames executes steps 1. and 2. , but in reverse order steps 3. and 4. descending from instantiated top model objects down to the sketch. After that, step 5. refines the instance of the scene model, eliminating misinterpretations and altering the confidence of some object properties. Modules 1. and 2. are called the sketch generator, modules 3., 4., and 5. the interpreter.

3.3 CONSTRAINTS

There are several assumptions and constraints which motivated us to develop the concepts described in this paper.

1. A single real object can have different representations as a picture object. The reasons are:
   - Different projections of a 3-dimensional object onto the 2-dimensional camera image-plane.
   - Noise (graylevel and scan jitter).
   - Changing illumination of the scene.
   - Varying reflectivity of moving object surfaces.

2. By the same token subsequent pictures can be very different in spite of the same configuration of the same real objects. This makes it extremely difficult to analyze the motional properties of picture objects solely by comparing gray level representations.

3. The transformation of the picture into the sketch is unidirectional and (hopefully) preserves meaning and context of picture objects while mapping them to sketch objects. All further processing will be based on the sketch only.

4. Subsequent pictures have the same content; they contain equivalent picture objects in a slightly changed configuration, if the sampling rate is high compared with the velocity of

picture objects. Therefore a sequence of scene interpretations reflects incrementally changing configurations.

4. STRUCTURE OF THE MODEL

The model has the following structure:

Scene model

Hierarchy of
model objects

Procedures

4.1 HIERARCHY OF MODEL OBJECTS

This part of the model defines the primitive elements, a hierarchy of model objects and their parts, and a network of constructions, which are mappings between model objects, their parts and the primitive elements. The structure is similar to the one proposed by Barrow et al. [8].

The primitive elements are a special class of model objects. Instances are found by special algorithms associated to these structures, which search in the sketch for instances of these primitive elements. An instance of a model object is defined in def.5.2. A relational structure is associated to each model object or part of it.

Def.4.1 <u>Relational</u> <u>structure</u>:

A finite set T, the carrier, with a finite set of functions F, and a set of values W

$$F = \{ f \mid f: T^n \dashrightarrow W_f \}$$

$$T^n = T \times . . . \times T \qquad , W_f \subset W$$

The domains of the functions f define relations as subsets of

$T^n \times W$.

Carriers of all model objects in the hierarchie contain elements of the following type:

Def.4.2 <u>Carrier of model objects</u>:

TM = {point, line, arc, region}

TM is the type definition of primitive elements.

Properties are defined:

Def.4.3 <u>Properties E</u>:

$$E = \{E_a \mid E_a: TM \dashrightarrow W_a,\ a=1...A\}$$

W_a is the range of values for property E_a.

We use:

X:	POINT	-->	[0,255]	
Y:	POINT	-->	[0,190]	coordinates
XS:	REGION	-->	[0,255]	
YS:	REGION	-->	[0,190]	center of mass
FL:	REGION	-->	[0,48896]	area
G:	REGION	-->	[0,255]	mean gray level
GS:	REGION	-->	[0,MAX]	variance
KS:	LINE			
or	ARC	-->	[0,MAX]	edge strength
GPHI:	LINE	-->	[0,360]	angle to x-axis
GL:	LINE	-->	[1,MAX]	length of line
KPHI:	ARC	-->	[0,360]	arc
KR:	ARC	-->	[0,MAX]	radius
MPX:	ARC	-->	[0,MAX]	
MPY:	ARC	-->	[0,MAX]	center coordinates.

Relations are defined:

Def.4.4 <u>Relations R</u>:

$$R = \{R_b \mid R_b: TMxTM \rightarrow W_b,\ b=1...B\}$$

W_b is the range of values for R_b.

We use:

KF:	REGIONxREGION	-->	[T,F]	contact between regions
TV:	REGIONxREGION	-->	[T,F]	tot. enclosed region
AP:	LINExPOINT			
or	ARCxPOINT	-->	[T,F]	start point
EP:	LINExPOINT			
or	ARCxPOINT	-->	[T,F]	end point
RA:	REGIONxLINE			
or	REGIONxARC	-->	[T,F]	boundary
GLA:	LINExLINE	-->	[T,F]	equal length
WK:	LINExLINE	-->	[0,360]	angle between lines
KD:	POINTxPOINT	-->	[T,F]	point identity

All relational structures for model objects have to be formulated in terms of the carrier TM, the properties and relations defined above. LINE and ARC are used for the description of shape. POINT is used to link the symbolic description to absolute positions in the picture. REGION describes surfaces of picture objects. The interpreter has the facility to combine two or more regions to a new instance of the primitive element REGION to overcome the situation when one surface of an object is segmented into several regions.

To assure the efficency of the matching process and to allow different control strategies the hierarchiy of model objects is formulated:

Def.4.5 <u>Model object</u>:

$$MO^d = (T^d, E^d, R^d) \qquad d=1...D$$

Def.4.6 <u>Carrier of a model object</u>:

$$T^d = \{TR_i{}^d \mid TR_i{}^d \in TM,\ i=1...N(d)\}$$

Def.4.7 Properties of a model object:

$$E^d = \{(Ea, TR_i{}^d, Wa_i) \mid a \in [1,A],\ i \in [1,N(d)]\}$$

Def.4.8 Relations of a model object:

$$R^d = \{(Rb, TR_i{}^d, TR_j{}^d, Wb_{ij}) \mid b \in [1,B];\ i,j \in [1,N(d)]\}$$

Def.4.9 Construction:

$$K^d = \{K_{fc}{}^d \mid K_{fc}{}^d : T^f \dashrightarrow T^d;\ f,d \in [1,D],\ c=1...C(f,d)\}$$

$C(f,d)$ = multiplicity of MO^f in MO^d

Def.4.10 Partial construction:

$K_{fc}{}^d : (T^f \dashrightarrow T^d)$

AND (ALL $E_a \in E^f$: $E_a(K_{fc}{}^d(T_i{}^f)) \sim E_a(T_i{}^f)$

AND (ALL $R_b \in R^f$: $R_b(K_{fc}{}^d(T_i{}^f), k_{fc}{}^d(T_j{}^f)) \sim R_b(T_i{}^f, T_j{}^f))$

The equivalence ~ is defined as a fuzzy relation.
The mapping

$$K_{fc}{}^d : T^f \dashrightarrow T^d$$

is in analogy to a monomorphism.
The purpose of the partial construction is to preserve all properties and relations of the partial model object.

Def.4.11 Top model object:

A model object MO^d is a top model object if there is no partial construction

$$K_g{}^f : T^g \dashrightarrow T^f \qquad \text{with } g=d$$

Because our main goal was to develop a scheme for utilizing spatial context and the description of motion properties of objects, we restricted the complexity to simple ones like boxes, balls, flowers, jumping jacks, etc.. Each of them is modelled by a top model object.

Model dependent procedures calculate values for relations which are not part of the sketch.

4.2 SCENE MODEL

The scene model serves to represent possible configurations of sketch objects and to define the current context of sketch objects by establishing spatial relations between model objects. The scene model has a static and a dynamic part. The static part is formulated as a network of constraints on the spatial relations between model objects. It reflects the starting configuration of objects in a sequence of scenes. The nodes in this network are linked to top model objects.

Formal definition:

Set of nodes N = {object}
Set of times TIME = {t1, . . . ,tn} , t ordinal number of picture

Properties:

X:	NxTIME	-->	[0,255]	
Y:	NxTIME	-->	[0,190]	position of scene object
F:	NxTIME	-->	[0,48896]	area
VX:	NxTIME	-->	[0,MAX]	
VY:	NxTIME	-->	[0,MAX]	velocity
G:	NxTIME	-->	[0,255]	mean gray value
NAME:	N	-->	{ball, flower, . . .}	

Relations:

D:	NxNxTIME	-->	[0,MAX]	distance
K:	NxNxTIME	-->	[T,F]	contact
S:	NxNxTIME	-->	[T,F]	surrounding
RL:	NxNxTIME	-->	[0,15]	relative position
IS:	NxHIERARCHY	-->	[T,F]	link with a top model object
ORIS:	NxHIERARCHY	-->	[T,F]	alternative links

The static part of the scene model is used for evaluating the maximal consistent (with respect to relation and property constraints) interpretation of the scene, selecting one of the many (possibly inconsistent) interpretations of single model objects. This interpretation is transformed to a relevant current scene model by instantiating all verified nodes of the static part of the scene model. The current scene model then serves as a guide for subsequent analysis by activating a top down interpretation of model objects.

Subsequent picture interpretations are compared and changing values of properties and relations are stored as time ordered lists of values in the frame of the current scene model.

Abstractions on the time behaviour of values of properties and relations may easily be implemented as algorithms working on these lists. Because these values are bound to named nodes of the current scene model, the motional properties of single objects can be determined.

5. INTERPRETER

The interpreter has different tasks:

1. Search for instances of the primitive elements in the sketch.

2. Search for instances of the model objects as substructures of the sketch.

3. Evaluate the maximal consistent interpretation of the scene.

4. Trace the development of scene interpretations and evaluate the time context.

5.1 SKETCH-OPERATOR

The sketch-operators are procedures which analyze boundaries and regions in the sketch. They enlarge the sketch when they find arcs and straight lines and collect regions to compound regions, by inserting descriptors for these entities into the sketch. Instances of the corresponding primitive elements are built and linked to these descriptors. Sketch operators also generate instances of all points (coordinate pairs) which have been used in connection with other primitive elements.

5.2 GENERAL MATCHER

All models at non-terminal levels of the hierarchy are matched by an algorithm, which is independent of the model in question. The matcher finds largest common substructures of relational structures. To generate an instance of a model object we look for a mapping from the model to the sketch, which preserves properties and relations of the model object. Simple graph isomorphism does not work efficiently [17,18,19], therefore the matching is done in a hierarchical manner, trying to combine instances of model objects to those of more complicated objects at a higher level. The possible combinations are defined by the network of constructions. It may be traversed bottom up when for the first picture the property and relation constraints are not known very well, or it can be used for a top down recursion where the search is guided by the knowledge of previous interpretations.

Def.5.1 Interpretation of a model object:

$$I^d = \{I_t^d \mid t=1...T(f)\} \quad d \in [1,D]$$

Def.5.2 Instance:

$$I_t^d = (T^d, S_t^d) := ((TR_1^d, S_{t_1}^d), \; . \; . \; . \; , (TR_{N(d)}^d, S_{t_{N(d)}}^d))$$

T^d carrier of model object MO^d
S_t^d sketch object

Def.5.3 Sketch object:

$$S_t^d = (S_{t_1}^d, \; . \; . \; . \; , S_{t_{N(d)}}^d)$$

$S_{t_i}^d$ carrier element of the sketch

The sketch object is a subset of the sketch defined by the mapping

$$I^d : T^d \text{ --> SKETCH}$$

This mapping assures that all relations and properties of the model object are valid in the associated sketch object. The procedure MATCH finds all instances of higher level objects in two steps:

1. Binding

All sketch objects to be instances of submodel objects are checked for conservation of properties and internal relations. That filters out a set of candidates

$$CAND = \{(T_h{}^d, S_t{}^h) \mid$$

$$t \in [1,T(h)],\ h \in \{g,f\},\ ALL\ E_a \in E^d : E_a(T_h{}^d) \sim E_a(S_t{}^h)\}$$

$$T_h{}^d = K_h{}^d(T^h)$$

2. Verification

Sketch objects to serve as parts of the object under construction are selected by verifying the relations between them which are prescribed by the object model. Following Ambler [15], the problem is transformed into a process of finding cliques in a graph the nodes of which are all elements of CAND. A bidirectional binary relation COMP is defined between nodes. COMP is true if all relations between the model elements are valid also between the associated elements of the sketch. Maximal cliques in this graph correspond to combinations of sketch objects which are accepted as an instance of a model object. As clique finder we use a modified version of the Bron-Kerbosch algorithm [20]. Assigned to each instance is a confidence value which depends on the similarity of properties and relations between model and sketch object. Poor interpretations are eliminated during the CAND or COMP process, which have a model dependent decision threshold, depending on the subobject confidence and the properties and relations which are checked. It is apparent that the matcher can find partial matches and should be able to cope with imperfect data of real images.

5.3 SCENE INTERPRETER

The scene interpreter looks for the maximal consistent interpretation of the scene model. Using the same formalism as in the match process all (not necessarily unique) instances of top model objects are stored as nodes of a consistency graph. So far no special relations between these objects have been used. They contain the labels of all interpreted regions. The relation COMP between the nodes computes pairwise consistency of labels and regions using the constraints stored in the scene model. Maximal cliques in this graph are consistent combinations of instances of top model objects. The largest maximal clique(s) are the accepted configuration(s). The selected interpretation is stored as an instance of the scene model and defines the current scene model.

5.4 MOTION ATTRIBUTES

The last instance of the scene model is used for an activation of the model objects, which are linked to the scene model by IS or ORIS pointers. They should be found in a reduced searchspace, if the semantics of this scene is similar to that of the previous one. The subsequent current scene models are stored in one common structure which was set up by the first instance. That is done by collecting the time-dependent values for properties and relations in lists of (value,time) pairs attached to the nodes of the current scene model. These nodes have also pointers to the attached sketch objects. The motional properties of sketch objects are obtained by evaluating the property and relation lists of the associated nodes of the current scene model. Velocity properties can easily computed and rely not only on changing local properties (i.e. position) but also on the whole network of changing relations.

6. SEGMENTATION

The symbolic description of the picture, the sketch, is stored as a relational structure. In addition to the definitions in the model there are the following types of elements, properties and relations:

1. Carrier

TS = {{POINT, BOUNDARY, REGION },{LINE, ARC, REGION}}

The first subset is produced by the segmentation process, the second by the sketch-operators.

2. Properties

KK:	BOUNDARY	--> [CHAINCODE]	
KS:	BOUNDARY	--> [0,MAX]	Strength of boundary

3. Relations

AP:	BOUNDARYxPOINT	--> [T,F]	
EP:	BOUNDARYxPOINT	--> [T,F]	
RA:	REGIONxBOUNDARY	--> [T,F]	
TK:	LINExBOUNDARY	--> [T,F]	
or	ARCxBOUNDARY	--> [T,F]	link
TF:	REGIONxREGION	--> [T,F]	link

The TK and TF relations are designated to link the nodes representing compound regions, arcs, and straight lines to boundaries and regions. The segmentation is produced by a region grower. On the output of this step above properties and relations are computed and stored in the sketch.

The segmentation is unidirectional. No model information is used to guide the segmentation process, so it is tuned to generate rather to much than to few boundaries and regions.

7. CONCLUSION

The structure of a model based system, which utilizes both spatial and temporal context in the same formal structure has been introduced. The sketch generator produces a symbolic description of TV images having a resolution of 512*573 in less than 5min CPU-time and 60k core on a DEC-KI-10. The implementation is fully described by Kraasch and Zach in [22]. The structure of the model, the model editor and translator, and the interpretation process is extensively discussed in [21].

Future work will include a substantial refinement of the vocabulary of primitive elements, a bidirectional segmentation process, and the development of a system structure using parallel processes to enable a more flexible matching and flow of control. The goal underlying these refinements is to describe more complex scenes. The implementation, now, is rather straightforward. Programming languages used are PASCAL and SAIL. The programs for sketch-operators, model object hierarchy, and model object interpreter are in the test phase and work for simple scenes for which the models have been written.

REFERENCES

[1] ROBERTS "Machine perception of three-dimensional solids" In Optical and Electrooptical Information processing, (Tippet et al., eds.) MIT Press, 1965, p.159-197

[2] GUZMAN "Computer recognition of three-dimensional objects in a visual scene" (Dissertation) MIT Report MAC-TR-59, 1968

[3] WINSTON "Learning structural descriptions from examples" in The Psychologie of Computer Vision (Winston, ed.) McGraw-Hill, New York, 1975

[4] WALTZ "Generating semantic descriptions from drawings of scenes with shadows" MIT Report, AI-Lab, TR-271

[5] YAKIMOVSKY "Scene analysis using a semantic base for region growing" STANFORD Report AIM-209, 1973

[6] SHIRAI "Edge finding, segmentation of edges and recognition of complex objects" Proceedings 4th IJCAI, Tbilisi 1975, p.674-681

[7] OHLANDER "Analysis of natural scenes" Computer Science Department, Carnegie-Mellon University, Ph.D Thesis, April 1975

[8] BARROW et al. "Some techniques for recognizing structures in pictures" In Frontiers of Pattern Recognition, (Watanabe ed.), AP, New York 1972, p.1-29

[9] AGGARWAL, DUDA "Computer analysis of moving polygonal images" IEEE Transactions on Computer, Vol. C-24, 1975, p.966-976

[10] AGGARWAL, CHOW "Computer analysis of planar curvilinear moving images" University of Texas at Austin, Report 1976

[11] POTTER "Scene segmentation by velocity measurements obtained with a cross-shaped template" Proceedings 4th IJCAI, Tbilisi 1975, p.803-810

[12] NAGEL "Formation of an object concept by analysis of systematic time variations in the optically perceptible environment" Institut für Informatik, Universität Hamburg, Bericht No. 27, 1976, Computer graphics and image processing (in press).

JAIN, MILITZER, NAGEL "Separating non-stationary from stationary scene components in a sequence of real-world TV-images" IJCAI-77, 612-618 and IfI-HH-B32/77 (March 77), Institut für Informatik, Universität Hamburg

[13] BADLER "Temporal scene analysis: Conceptual descriptions of object movements" University of Toronto, Technical Report No. 80, 1975

[14] TSOTSOS "A prototype motion understanding system" University of Toronto, Technical Report No. 93, 1976

[15] PRICE "Change detection and analysis in multi-spectral images" Department of Computer Science, Carnegie-Mellon University, Ph.D Thesis, 1976

[16] AMBLER et al. "A versatile system for computer controlled assembly" Artificial Intelligence 6(1975), p.129-156

[17] VORNEIL "An efficient algorithm for graph isomorphism" JACM 17(1970), p.51-64

[18] HAYES-ROTH "Patterns of induction and associated knowledge acquisition algorithms" Computer Science Department, Carnegie-Mellon University, May 1976

[19] ULLMAN "An algorithm for subgraph isomorphism" JACM 23(1976), p.31-42

[20] BRON, KERBOSCH "Finding all cliques of an undirected graph" CACM 16(1973), p.575-577

[21] BERTELSMEIER "Struktur eines Kontext-gelenkten Systems zur Beschreibung von Bildfolgen mit bewegten Objekten" Institut für Informatik, Universität Hamburg.

[22] KRAASCH, ZACH "SERF - Ein sequentieller Segmentierungsalgorithmus" IfI-HH-M-59/78 Institut für Informatik, Universität Hamburg

[23] YAKIMOVSKY "Boundary and object detection in real world images" Proceedings 4th IJCAI, Tbilisi 1975, p. 695-704 and J.ACM 23(1976) 599-618

1 Monien, B.
Beziehungen zwischen Zeitkomplexitaetsklassen und Kellerautomaten mit Zaehlern
X/71, 36 p

2 Kupka, I., Wilsing, N.
A Formal Framework for Dialog Languages
XII/72, 64 p

3 Kupka, I., Wilsing, N.
Syntax und Semantik des Dialogsprachenkonzepts HDL
I/73, 54 p

4 Kupka, I., Wilsing, N.
An Approach to Formal Semantics of Dialog Languages
/73, 46 p

5 Brunnstein, K., Schmidt, J. W.
Teilnehmersysteme
II/73, 167 p

6 Monien, B., Weicker, R.
Ueber die Effizienz von Tabulator-Turingmaschinen mit beschraenkter Markenzahl im Vergleich zu anderen Maschinenmodellen
X/73, 34 p

7 Woehl, K.
PREVAN - ein Vorcompiler zur Vektoranalysis
X/73, 67 p

8 Monien, B.
Komplexitaetsklassen von Automatenmodellen und beschraenkte Rekursion
II/74, 95 p

9 Nagel, H.-H.
On Processes, High Level Languages and Structured Programs for Computer Networks
V/74, 30 p

10 Fleischhauer, G.
Beziehungen zwischen den Zeitkomplexitaetsklassen verschiedener Automatenmodelle
IX/74, 42 p

11 Grosse-Lindemann, C.-O., Nagel, H.-H
Postlude to a Pascal-Compiler Bootstrap on a DECSystem-10
X/74, 28 p

12 Hoepner, M.
Ueber den Zusammenhang von Szilardsprachen und Matrixgrammatiken
XII/74, 119 p

13 Friesland, G., Sengler, H.-E.
Zur Uebertragung von Compilern durch Selbstcompilation am Beispiel des Pascal-Compilers
XII/74, 190 p

14 Monien, B.
Rekursive und grammatikalische Charakterisierung der Exponentialzeitsprachen und ein neuer Beweis des

15 Heibey, H.-W.
Ein Modell zur Behandlung mehrdimensionaler Strukturen unter Beruecksichtigung der in ihnen definierten Lagerelationen
V/75, 140 p

16 Heibey, H.-W., Lutterbeck, B., Schueler, U., Sengler, H.-E.
Nichttechnische Auswirkungen bei der EDV-Anwendung
V/75, 64 p

17 Hoepner, M.
Durch Mischoperationen definierte Sprachfamilien
VII/75, 26 p

18 Kupka, I., Oberquelle, H., Wilsing, N.
An Experimental Language for Conversational Use
IX/75, 105 p

19 Kudlek, M.
Reducing Operators for General Formal Systems
XI/75, 21 p

20 Opp, M.
Charakterisierungen erkennbarer Termmengen in absolut freien universellen Algebren
I/76, 109 p

21 Schaeffler, G.
Stochastische OL-Systeme
II/76, 66 p

22 Juers, J.
Indexgrammatiken mit eingeschraenkter Regelanwendung
II/76, 39 p

23 Nagel, H.-H.
Experiences with Yakimovsky's Algorithm for Boundary and Object Detection in Real World Images
III/76, 23 p

24 Wittig, T.
Semantische Analyse von Saetzen zur Erfassung eines Sachverhaltes
V/76, 137 p

25 Brunnstein, K., Lewandowski, U., Schmidt, J. W.
Rechner fuer den Schuleinsatz - Ein Schulrechnerkatalog
X/75, 260 p

26 Albrecht, M.
Automatische Erkennung von Organgrenzen in Leber-Szintigrammen und Erfassung von Parametern zur Beschreibung des Leber-Abbildes im Szintigramm
III/76, 82 p

27 Nagel, H.-H.
Formation of an Object Concept by Analysis of Systematic Time Variations in the Optically Perceptible Environment
VII/76, 36 p

28 Oberquelle, H.
Grundbegriffe zur Beschreibung von Dialogen und dialogfaehigen Systemen

29 Krieg, B.
Petrinetze und Zustandsgraphen
I/77, 100 p

30 Bruegge, B., Gisch, B., Kahl, T., Linde, H., Mittelstein, M., Westphal, H.
Concurrent PASCAL - Compiler fuer Kleinrechner
XII/76, 190 p

31 Schmidt, J. W.
Some High-Level Language Constructs for Data of Type Relation : An Investigation Based on PASCAL Extensions
I/77, 18 p

32 Jain, R., Militzer, D., Nagel, H.-H.
Separating Non-stationary from Stationary Scene Components in a Sequence of Real World TV-Images
III/77, 18 p

33 Nagel, H.-H.
Analysing Sequences of TV-Frames : System Design Considerations
III/77, 44 p

34 Valk, R.
Self-Modifying Nets
VII/77, 36 p

35 Hoepner, M., Opp, M.
Renaming and Erasing in Szilard Languages
V/77, 37 p

36 Ullrich, G.
Der Entwurf von Steuerstrukturen fuer parallele Ablaeufe mit Hilfe von Petri-Netzen
XII/76, 268 p

37 Schmidt, J. W.
Type Concepts for Database Definition : An Investigation Based on Extensions to PASCAL
V/77, 22 p

38 Wahlster, W.
Zur Repraesentation von vagem Wissen in natuerlich-sprachlichen Systemen der kuenstlichen Intelligenz
VII/77, 190 p

39 Damm, F.-E.
Entwurf einer Dialogsprache zum computergestuetzten Ableiten und Beweisen
V/77, 142 p

40 Bonk, M.
Rudimentaere Klassen
III/77, 69 p

41 Bertelsmeier, R., Radig, B.
Context-Guided Analysis of Scenes with Moving Objects
IV/77, 18 p

42 Neumann, B.
Identifikation von gestoerten Objektansichten unter Verwendung geradliniger Konturapproximation

43 Materna, W.
Ein allgemeines Verfahren zur Konstruktion von repraesetativen Modellen
XII/77, 187 p

44 Dreschler, L., Nagel, H.-H.
Using ´Affinity´ for Extracting Images of Moving Objects from TV-Frame Sequences
II/78, 20 p

45 Froelich, R.
Zwei neue Verfahren zur Aufwandsminderung diskreter Simulation von Rechensystemen
II/78, 157 p

46 Fischer, H., Jarke, M., Meyer, D., Schmidt, J. W., Ullmer, W.
A Structered Framework for the Implementations of Relations
II/78, 22 p