

AN ATN MODEL FOR 3-D RECOGNITION OF SOLIDS IN SINGLE IMAGES

H. Tropf, I. Walter

Fraunhofer-Institute for Information and Data Processing
Sebastian-Kneipp-Str. 12-14
D 7500 Karlsruhe 1/FR Germany

ABSTRACT

Augmented Transition Network (ATN) models have been developed in the field of natural language understanding. In this paper, an ATN model for image analysis is discussed; it is described for the 3-D recognition of randomly oriented solids with known geometry. Due to its perspicuity and generative power, ATN appears to be well suited to incorporate the geometrical knowledge at hand. Mechanisms for relating 2-D images to 3-D objects are handled by using the inverse representation: calculating 2-D model instantiations of a 3-D geometric model. The ATN controls an analysis-by-synthesis search procedure that is based on hypothesis generation and verification. A state-space search technique is used. Identical subspaces are recognized, the same is true for subspaces containing similar hypothetical model instantiations. The procedure copes with degraded image data. Introducing general semantic knowledge and knowledge of the domain fits harmonically into the ATN framework.

1. Introduction

In this paper the use of an Augmented Transition Network (ATN) model for image interpretation is discussed; a procedure for the 3-D recognition of randomly oriented rigid bodies with known geometry is described.

ATN models are mainly used in natural language understanding and speech recognition [1]. Here, we argue that the advantages of the ATN formalism, namely its perspicuity and generative power, can also be brought to bear in the domain of patterns and models with more than one dimension. Syntactic approaches to pattern recognition are thought of as being less appropriate for 3-D object recognition "since it is not obvious how to incorporate in it mechanisms for relating 2-D images to 3-D objects" [2]. It is shown in this paper that such mechanisms can harmoniously be embedded into the ATN framework by doing the inverse: calculating 2-D instantiations from a 3-D model.

* The research reported in this paper has been supported by the Bundesministerium für Forschung und Technologie under contract No. 413-5839-08IT10270. Responsibility for the content is with the authors.

The approach is based on the idea of analysis-by-synthesis[3]. Knowledge of the pattern generation is stored in the model. Hypothetical model instantiations are referred to as prototypes. Prototypes are created. If the similarity between a prototype and the actual data to be analyzed exceeds some threshold, the prototype is considered the model instantiation for the actual pattern data. Otherwise it is laid aside and another prototype is taken into consideration. As a rule, a problem arises by the tremendous amount of prototypes to be considered. A solution is an analysis-by-synthesis search procedure as sketched in Fig. 1: Prototypes are partially built up. Partial prototype data are compared with pattern data in order to compute a similarity measure to guide the selection of the partial prototype which is to be developed in the next step. The Fig. 1 feedback loop is essentially a process of hypothesis generation and verification: based on the partial prototype selected and using the constraints imposed by the model, hypotheses are generated to be verified by the pattern data.

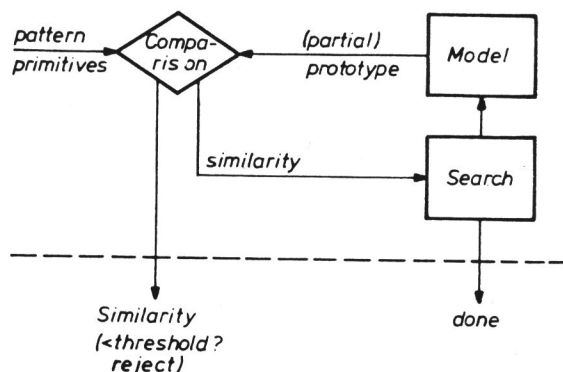


Fig. 1: Analysis-by-synthesis search procedure

We suppose that the pattern is represented as a list of primitives (extracted in a preprocessing stage) that are associated with geometrical properties (position, orientation etc.). The ATN approach presented here is an extension of a 2-D model described earlier [4] dealing with corner

primitives which later has been extended by straight line segment and circular arc primitives [5]. Additionally, points can be incorporated as primitives which can be either the central points associated with circular arcs (ellipses in 2-D) or simply corner points ignoring the direction of the associated edge lines.

The basic idea for the recognition of rigid bodies is as follows: when following a path through the transition network from the initial state to a goal state, any CAT-arc (see section 2.2) is associated with the placement of a primitive in 3-D space. Thus, the number of degrees of freedom of an object model is more and more constrained until the object is fixed in 3-D. After fixation, the rest is hypothesis verification by checking pattern primitives at clearly defined places of the image plane.

The following example is convenient for explanation. For the sake of simplicity, ambiguities are at first neglected in the following. We assume that only point primitives are used and parallel projection is provided; point occlusion is neglected. After the fixation of the first point P_1 in 3-D, the second point P_2 to be fixed must lie on the surface of a sphere whose radius is known. Therefore the second pattern primitive must lie within the corresponding circle in the image plane. Having chosen such a point, we know (neglecting ambiguity) the axis P_1 - P_2 . The object has now the freedom to rotate around P_1 - P_2 . The third point P_3 (not lying on P_1 - P_2), must lie on a circle around the P_1 - P_2 axis which in projection is an ellipse. Having selected a pattern primitive point close enough to the ellipse, the object is fixed. The remainder of the path is verification.

After verification, the data are smoothed in order to get a more precise interpretation. In this respect, the method is similar to the RANSAC (random sample consensus) approach to model fitting, in which estimates of the model parameters are generated using as small an initial data set as feasible; after verification, an improved fit by applying a smoothing technique is computed [6].

In the following, the main components of our system for the recognition of solids with known geometry are described: model and search.

2. Model

The model is decomposed into a purely geometrical model and a generative model. The geometrical model describes the object which is assumed to be in a randomly chosen position in 3-D space. The generative model is needed for controlling the analysis-by-synthesis procedure.

2.1 Geometrical model:

Rather than relying on a set of characteristic views, we use a single geometrical description of

the object. We aim to use a description as usual in the CAD field. Thus, the teaching phase for industrial part recognition could be reduced to merely typing in the generative model with associations to some specified geometrical model elements. As a first step, we have chosen a polyhedron description such that any (hidden-line)-3-D view of the part can be calculated. Many real workpieces fall into the polyeder class or can be described sufficiently well in terms of planar or slightly bent surfaces with straight edges.

2.2 Generative model:

The generative model is represented as an Augmented Transition Network. It describes several alternatives of model-to-pattern primitive association sequences. We use a somewhat more general representation than in our early approach [4]: instead of describing a sequence of specified model primitives (for example corner1, edge3,...), we rather prescribe the category of primitives (for example corner, edge, ...).

The network is augmented with a set of registers holding arbitrary information to keep track of the model-to-pattern primitive associations carried out. From these associations, information is deduced about how the position in 3-D space is constrained and hence what constraints are imposed on the pattern primitives to be selected further. The registers contain also supplementary information necessary to perform additional tests on the primitive data and to collect the information for the final image description.

The ATN consists of states, arcs, a dictionary, named registers, actions, and conditions.

Some comments on the arcs, the dictionary, and the named registers are given.

Arcs: we use the usual CAT-, JUMP-, SEEK-, and SEND-arcs (following Winograd's notation [7]). Additionally, CATEGORY-PHANTOM-arcs are defined as transitions by introducing a phantom pattern primitive.

Dictionary: It consists of the following parts:

structural knowledge	geometrical models geometrical formulas
semantic knowledge	will contain - overall knowledge of the world (e.g. constraints contributed by shadows according to Waltz's rules [8]) - special knowledge about the described world (e.g. restrictions on model subset selection)

Named registers: among other things they contain

- pointers to dictionary entries
- parameters

An example

The transition network model for the recognition

of solids assumes parallel projection. On its highest level the ATN has the form shown in Fig. 2. The recognition of a solid is done in three steps. In the FIX-phase (MOD-1), the object has to be fixated in 3-D space. It yields the hypothesis of the solid's position and orientation. In the VER-phase (MOD-2), the hypothesis established in the FIX-phase is verified. In the SMOOTH-phase (MOD-3), the data are averaged to obtain a more precise result.

CAT-arcs can be replaced by SEEK-arcs if convenient. To cope with broken pattern lines, for instance, an edge arc can be replaced by an edge network. For edge fragment composition the procedure described in [9] is appropriate and easy to incorporate into the ATN framework. Especially in the verification phase one could even think of invoking a special line finder or local feature detector. As the line is conjectured to be in a well constrained neighbourhood, line following should be possible even in highly corrupted image data.

3. Search

The search is based on the state-space search technique [10]. In the following, nodes are referring to the search tree, states and arcs are referring to the ATN. Registers are associated with nodes.

Each node N is associated with one state S_N , its successor nodes N_i , $i=1,2,\dots,n_N$ are associated with one successor state of S_N . Thus, a path from the root R to a goal node is associated with a path through the ATN.

With each node N a subspace of the six dimensional position and orientation space is associated. This is implicitly done by a set of model-to-pattern primitive associations $[P_i, p_j]$ (P_i : model primitive, p_j : pattern primitive) that are encountered in the registers when following a path from the root R of the search tree to the node N . Along this path, the position and orientation space is successively reduced.

Two basic restrictions are imposed on the $[P_i, p_j]$ associations:

- RES1: P_i and p_j are of the same category (e.g. both are edges)
 RES2: On any path from the root to a goal node, each model primitive is encountered exactly once.

Additional primitive selection restrictions will be introduced below.

Expanding a node N :

S_N CAT S_N'

1. Select a set P of model primitives (the category being prescribed by the arc) so that the primitive selection restrictions are not violated.
2. For each $P_i \in P$ calculate the geometrical con-

straints imposed on the pattern primitives to be associated with P_i (including visibility, which can always be done in the VER-phase).

3. For each p_j satisfying the constraints (within some tolerance), a successor node with state S_N' and association $[P_i, p_j]$ is created. If there is ambiguity about the interpretation of the position and orientation subspace reduction introduced by the association, all possibilities are stored in an alternative register. When necessary (may be further down in the tree), for each possible interpretation a successor node is created accordingly.

S_N CAT-PHANTOM S_N'

Step 1 as above

2. For each $P_i \in P$ calculate the geometrical properties of the phantom pattern primitive p_j .
 3. For each P_i , create a successor node with state S_N' and association $[P_i, p_j]$.
- JUMP, SEEK, and SEND arcs produce one successor node.

Missing primitives are coped with because the FIX-network incorporates many alternatives of fixation. For instance, in the Fig. 2 example pattern points $[P_1, P_2, P_3]$ or $[P_3, P_5, P_6]$ could be chosen for fixation.

We use a best first strategy based on a value f which is a measure of similarity that evaluates how well the (partial) prototype fits the associated model data. Among other things, it takes into account how many phantom primitives are associated with visible model primitives.

How to avoid doing the same thing twice?

The goal is to restrict as heavily as possible the primitive selection in order to speed-up the search.

- A) The procedure described so far does not take into account the case that the same interpretation results when using different sequences of the same $[P_i, p_j]$ -associations. The approach we actually use is a modification of an idea found in [11].

It consists of defining a unique sequence for any subset of $[P_i, p_j]$ -associations. An obvious choice is to impose a monotony restriction on a quality measure $q(j)$ of the pattern primitives within a category. $q(j)$ is calculated in the preprocessing stage and can take into account, for example, the length and contrast of an edge, or the sharpness of a corner. The restriction applies within the FIX- and VER-phase independently. (Note: the restriction is not applicable for all ATN structures; in the above example it can be used).

- B) An additional source of unnecessary work stems from the fact that the same (or about

the same) position and orientation hypothesis results from different sets of $[P_i, p_j]$ associations. To avoid this, at the end of the FIX-phase a hold list of such hypotheses is updated. The arc is inhibited if the hypothesis - within some tolerance t - already exists.

At this point, an obvious idea should be outlined: Clustering approaches have been reported to be robust even under poor conditions (till now for 3-degrees of freedom) [12]. Why not use the hold list for clustering the position-orientation hypotheses if clusters emerge? It seems promising: for undistorted data, the procedure runs in some depth first manner; for highly ambiguous data, where many tree branchings are introduced, more hypotheses are calculated and clusters may arise that are helpful for guiding the search.

Note: Method B) no longer ensures that the optimal solution will be found; speed-up is gained at the expense of suboptimality where the error induced (by method B) is held within a known bound depending on threshold t .

Additional general speed-up techniques are optionally used; their effectivity has been demonstrated for one-way transition diagrams and 2-D models [13].

4. Discussion

We have shown that the ATN framework is well suited to incorporate special geometrical knowledge as well as general semantic knowledge for image analysis tasks. The difficulty of relating 2-D images to 3-D models is avoided by doing the inverse: calculating 2-D prototypes from 3-D models. This inversion means putting the burden on search; we use a state-space search technique. Here, much speed-up is gained - without getting gross errors - by recognizing not only identical subspaces but even subspaces that contain similar prototypes. Incorporating cluster techniques fits well into the framework and seems promising in highly ambiguous data.

For coping with missing pattern primitives, phantom primitives are created where possible. Instead of handling the pattern data directly, the procedure is actually dealing with prototypes which comply with the a priori knowledge. (Depending on the similarity measure chosen, the procedure decides upon a prototype with not too many phantom elements that should be visible). The final image interpretation is derived from the model instantiation rather than from pattern data.

Similarly, general knowledge such as line labelling rules can be incorporated; illegal pattern situations do not necessarily lead to a dead end because legal phantom elements can be created. Due to this error tolerance, introducing strict yes-or-no rules will enhance the performance even when dealing with distorted pattern data.

Special knowledge such as problem dependent relational constraints can be introduced perspicuously in the form of ATN conditions, they may be represented as a relational constraint network installed in the dictionary.

The main difference between our approach and those reported in the blocks world literature is that our approach allows coping with heavily distorted image data. This is the case with real world workpieces (see Fig. 3). Initial topology tests as described in Roberts [14] are not very useful here because they are too susceptible to data distortion. We may use objects with a less constrained geometry. Searching for consistent labellings in line drawings usually does not provide enough information in industrial applications. Thus the results of the blocks worlds research alone are not sufficient for our purposes. However, they are used to assist the recognition process, through rules incorporated into the ATN dictionary.

As a starting point, our ATN-based system for the recognition of solids provides a good base of an image analysis program to be successively augmented by more general semantic knowledge and knowledge of the domain. Programs are being implemented.

REFERENCES

- [1] W.A. Woods: Transition Network Grammars for Natural Language Analysis. *CACM*, 13:10 (1970), 591-606
- [2] A. Rosenfeld: Image Analysis: Progress, Problems, and Prospects. *Proc. ICPR-82*, Munich, Germany, Oct. 1982, pp 7-15
- [3] M. Halle, K.N. Stevens: Analysis by Synthesis. *Proc. Seminar on Speech Compression and Processing*. W. Wathen-Dunn, L.E. Woods (Ed.), Vol. 2 Paper D7. Bedford, Mass., Sept. 1959
- [4] H. Tropf: Analysis-by-Synthesis Search for Semantic Segmentation - Applied to Workpiece Recognition. *Proc. ICPR-80*, Miami Beach, FLA, Dec. 1980, 241-244
- [5] P. Rummel, W. Beutel: A Model-Based Image Analysis System for Workpiece Recognition. *Proc. ICPR-82*, Munich, Germany, Oct. 1982, 1014-1017
- [6] M.A. Fischler, R.C. Bolles: Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *CACM*, 24:6 (1981), 381-395
- [7] T. Winograd: Language as a Cognitive Process. Vol. 1: Syntax. Addison-Wesley Publishing Company. Reading, Mass. 1983
- [8] D. Waltz: Generating Semantic Descriptions from Drawings of Scenes with Shadows. P.H. Winston (Ed), *The Psychology of Computer Vision* N.Y., Mc Graw-Hill 1975, pp 19-92
- [9] W. Hättich: Recognition of Overlapping Workpieces by Model Directed Construction of Object Contours. *Digital Systems for Industrial Automation*, Vol. 1, No. 2-3 (1982), 223-239

- [10] N.J. Nilsson: Principles of Artificial Intelligence. Tioga Publ. Co., Palo Alto, CA, 1980
- [11] W.A. Woods: Optimal Search Strategies for Speech Understanding Control. Artificial Intelligence, 18:3 (1982) 295-326
- [12] T.G. Fang et al.: Three Dimensional Object Recognition Using a Transformation Clustering Technique. Proc. ICPR-82, Munich, Germany, Oct. 1982, 678-681
- [13] W. Hättich, W. Schwerdtmann, H. Trops: Experience with Two Hybrid Systems for the Recognition of Overlapping Workpieces. Proc. ICPR-82, Munich, Germany, Oct. 1982, 670-673
- [14] L. Roberts: Machine Perception of Three-Dimensional Solids. G. Tippet (Ed), Optical and Electro-Optical Information Processing. Cambridge, Mass.: MIT Press, 159-197

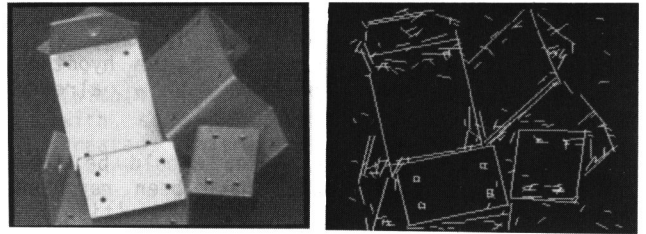
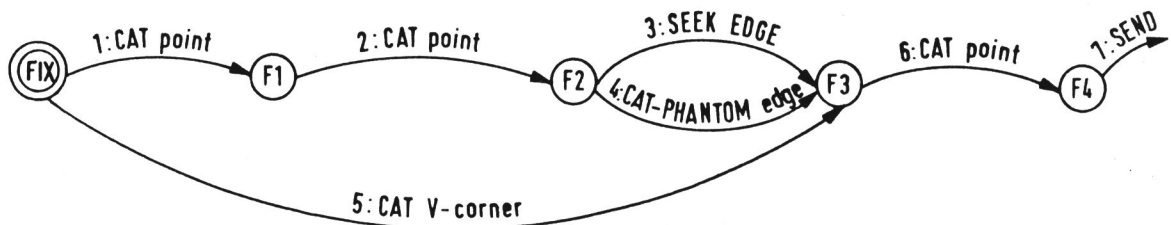


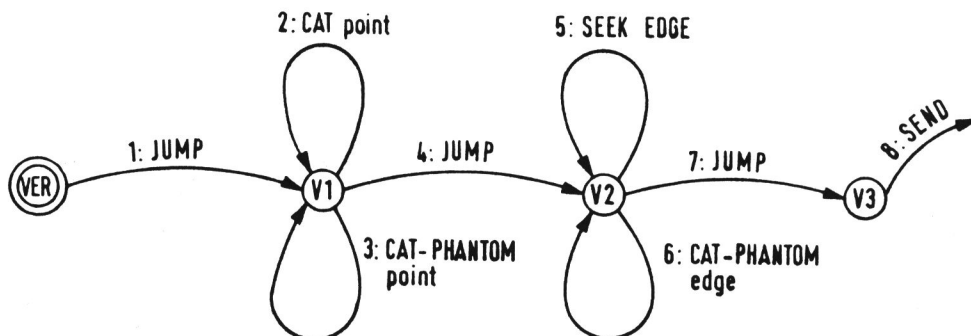
Fig. 3: Scenes under consideration



Highest level of the ATN-example. The FIXation subnetwork yields a position-orientation hypothesis of a 3-D object, which is verified in the VERification subnetwork. The JUMP-arc action smooths the data to obtain a more precise result.



Fixation by 3 points (including a verification edge between two points) or by corner plus point. The CAT-PHANTOM-arc is followed if the verification edge is not found.



Verification with points and edges. The CAT-PHANTOM-arcs are followed, if the point (edge) primitive is invisible or missing. The SEND-arc is followed if the similarity is greater than a threshold.

Fig. 2: An ATN example supposing parallel projection (notation see [7])