RECOGNITION IN 2D IMAGES OF 3D OBJECTS FROM LARGE MODEL BASES USING PREDICTION HIERARCHIES

J. Brian Burns Leslie J. Kitchen

Computer and Information Science Department
University of Massachusetts
Amherst, Massachusetts, 01003

Abstract

An object recognition system is presented that it designed to handle the computational complexity posed by a large model base, an unconstrained viewpoint and the structural complexity and detail inherent in a single view. The design is based on two ideas. The first is to compute descriptions of what the objects should look like in the image, called *predictions,* before the recognition task begins. This reduces actual recognition to a 2D matching process, substantially speeding up recognition time for 3D objects (with manageable storage overhead). The second is to represent all the predictions by a single, combined IS-A and PART-OF hierarchy called a *prediction hierarchy.* The nodes in this hierarchy are partial descriptions that are common to views and hence constitute shared processing subgoals during matching. Many of the problems encountered with large model bases and complex models are reduced by subgoal sharing: projections with similarities explicitly share the representation and recognition of their common aspects. The original contribution of this paper is the automatic compilation, from a 3D model base, of a prediction hierarchy that can be used to recognise objects. A prototype system based on these ideas is demonstrated using a set of polyhedral objects and projections from an unconstrained range of viewpoints.

1. Introduction

Object recognition is a central aspect of the process of understanding visual information, helping us to relate what we are seeing to what we have experienced in the past. In spite of much progress in this area, there are crucial problems that have not received adequate attention. One problem is that of representing information about 3D objects in a way that makes matching them to 2D image data efficient and reliable. That is, the geometric analysis required to relate an arrangement of 2D image features to the structure and pose of 3D objects should be sufficient for recognition and yet not involve massive amounts of computation during the time-critical recognition task. Another problem is ensuring that the storage and time complexity grows only slowly with respect to the size of the model base and the complexity of the models. We emphasize efficiency for model-based vision because of the remarkable ability of humans to rapidly recognize a large number objects from a range of viewpoints [Biederman85]. Also, while there are other sources of information that seem to be important, specifically scene context [Biederman85, Weymouth86] and model-independent understanding of 3D structure [Marr 82], these useful cues may quite often be unavailable, unreliable, or merely a first step towards a full interpretation.

Techniques for relating 3D model information to 2D image data can be partitioned into two basic approaches: prediction cycling and pre-recognition view analysis. In the former, represented by [Brooks81], the system iteratively cycles through a process of *prediction!,* deciding which projected model structure to search for in the image and what it looks like; *observation,* searching for image data that match the prediction; and *back constraining,* using additional properties of the matched data to further constrain the possible 3D poses and structural variations in the object. This approach could be computationally inefficient for large model bases since the prediction step would involve computing what is common about the projections of a large and possibly complex class of objects from a range of viewpoints. Similarly, manipulation of partially constrained poses during the back-constraining step can be involved [Lowe85].

In the alternative approach, *pre-recognition view analysis,* all expectations of what to look for in the image are generated before the actual recognition task. Recognition then becomes a 2D matching problem followed by object pose analysis and verification. The characteristic-view based schemes of Chakravarty [82], the property spheres of Fekete [84], the SCERPO system of Lowe [85], the principal views of Cooper [87] and the image-based descriptions of the VISIONS system developed in [Weymouth86] all roughly follow this method. Additionally, this approach has similarities with the photometric stereo interpretation system of Ikeuchi [87].

Another basic idea incorporated into our design is the use of IS-A and PART-OF hierarchical representations ([Marr82], [Brooks8l], [Mulder85] and [Weymouth86]). We developed a single, combined IS-A and PART-OF hierarchy called a *prediction hierarchy*. The nodes in this hierarchy are partial descriptions that are common to views and hence constitute shared processing subgoals during matching. Many of the problems encountered with large model bases and complex models are reduced by subgoal sharing: projections with similarities explicitly share the representation and recognition of their common aspects.

The original contribution of this paper is the automatic compilation, from a 3D model base, of a prediction hierarchy that can be used to recognize objects. A prototype system based on these ideas is demonstrated using a set of polyhedral objects and projections from an unconstrained range of viewpoints. A fuller treatment can be found in [Burns87].

2. Overview

The problem currently being studied is the compilation and use of a prediction hierarchy to recognize polyhedral objects using straight-line segments detected in the image. The actual objects used to demonstrate the design are shown in Figure 1. The objects have differences and similarities in various dimensions and part of the problem is to take advantage of both. The similarities in their visual structure, such as occurrences of parallelograms or of certain types of line junctions, must be utilized by the recognizer to make the search for a match efficient. The differences in visual structure, such as height-to-width proportions, must be utilized to discriminate between the objects. Addition-

This research was supported in part by the by the Air Force Office of Scientific Research under contract number AFOSR-86-0021. Thanks to Al Hanson, Ed Riseman Michael Boldt and John Dolan for comments, support and help with the figures.

Fig. 1. 3D Objects uued to demonstrate prediction hierarchy compiler and matcher *{tall box, cube, triangular prum, house, tetrahedron).*

ally, the variations in visual appearance caused by variations in the camera must be taken into account while doing the structural analysis.

The camera geometry used is as follows. The *viewpoint* of the camera is taken to be the position of the image origin in the object coordinate system. Currently, the projection is taken to be *normal perspective* [Brooks8l], i.e. orthographic projection with scaling. Also, since the objects are expected in any pose, all predictions generated are invariant to re-scaling, translation and rotation in the image plane. This means that there are only two degrees of camera variation that have a significant effect on the projections being described by the predictions, the two angular components of the viewpoint that sweep out a *viewing sphere* about the object.

A top-level view of the algorithm is as follows-

- Compile the prediction hierarchy from the set of 3D models given, before the recognition task begins. The hierarchy is compiled by starting with a small set of simple and very general structural predictions and then iteratively searching for commonly occurring combinations or specializations of these predictions across all objects and views, eventually isolating predictions that characterize the projections of fairly specific objects from a range of views

- Match predictions to input image. During the recognition phase, look for matches between segment descriptions and actual sets of image segments by a combined search of the prediction hierarchy and image data base. The hierarchy is used as an organized network of recognition subgoals.

- Refine the pose estimate. For each promising match, calculate the pose more precisely given the image-model mapping and an initial estimate of the pose implied by the prediction matched (i.e., some typical viewpoint from the set of those that could satisfy the prediction matched). For pose estimation refinement, the iterative method described in [Lowe85| is used

The integration of these three processes could be more sophisticated. However, the pressing issue at this point concerns the basic design of these steps; investigating further how they might fit together is outside the scope of this paper

Following a description of the nature of the predictions and their representation (Section 3), the processes that compile them and use them for matching are discussed (Sections 4 and 5).

3. Predictions and Their Representation

A *prediction* is a statement concerning some structural aspect of the image of an object. For example, this may be as simple and general as an assertion that there exists a pair of parallel segments in the projection; or as complex as a description of an image unique to some object. A prediction is represented here as a relational graph; the elements in the graph are projected straight-line segments. The relations associated with arcs in the graph mutually constrain the orientations, positions and sizes of pairs of segments.

A relation between a pair of projected segments used in the predictions is represented by ranges of four relational measures, u, v, a and s. Call one segment *s* and the other 8_2 The vector (u,v) is the position of segment 8_2 relative to si: it is the displacement of an endpoint of ^{8}2 from an endpoint of *s-* measured along *s* and normal to s_1 divided by the length of *s\.* The angle between them, a, is measured counterclockwise from S_1 to 8_2; and *s* is the relative scale or length ratio of 8_2 over 8_1 A relation is defined as an *extent box,* i.e. a set of ranges in u, *v, a* and 8, in order to capture the variation over ranges in viewpoint. For instance, projecting a pair of parallel object segments over all possible viewpoints will generate a set of measurements that have a single value (zero) in the a dimension but some extent in the others. Similarly, the family of projections of a pair of object segments that share an endpoint can be represented by a relation that has the value zero in both position components *(u,v).* A relation between projected segments is considered useful if it is valid over a wide range in viewpoints and its extent box is small in volume (for example, consider the two view-invariant relations just mentioned) The latter property is important if the relation is to help characterize an object's projection with a specificity sufficient to discriminate the object from a large number of other objects and from chance arrangements of image segments. Although invariant relations are clearly useful [Lowe85], they alone are in general not enough to fully characterize projections For instance, proportions are often strong characterizations of object structure, but the length measurement ratios that represent them are often not strictly view invariant For example, the tall box in Figure 1 has a height to width ratio that is significantly different from the cube over a large range in views, no other property can be used to differentiate them

It should be clear from the above discussion that a prediction may be valid only over a restricted set of views for a given object A prediction *instance* is a set of model segments, a mapping from the model segments to the segments of the prediction's relational graph and the range of viewpoints from which the prediction is valid for these segment bindings For a given model base, each prediction has a set of such instances and a *cumulative visibility,* the total area of all their visibility regions on the viewing sphere, across all objects.

Any given prediction in the *prediction hierarchy* is implicitly some relational graph, but explicitly it is almost always described as some combination or specialization of other predictions (see Figure 2). In such cases, it is a *derivative* of the other predictions A prediction is a *specialization* of another if it can be described by adding new relations or narrowing the extent boxes of existing ones. For example, the square can be described by adding a relation between the bottom and side segments of the more general parallelogram prediction that constrains their ratio of length to one. A prediction is a *combination* of other predictions if it can be described as a conjunction of these other predictions. Predictions may be combined in various ways, depending on the segment mappings between the whole prediction and its parts. See, for example, the triangular prism prediction of Figure 2. The mappings are collectively called the *arrangement* of the combination.

4. Prediction Hierarchy Compiler

The compiler attempts to build a prediction hierarchy that is composed of predictions that are commonly occurring combinations or specializations of other predictions. An added prediction should occur commonly enough that its satisfaction tends to dichotomize the instances of the prediction it is derivative of; and thus allow the recognizer to efficiently search for matches to objects and views. A useful way to build such a structure is by starting with a small set of simple and very general structural predictions and then iteratively searching for the commonly occurring combinations or specializations, eventually isolating predictions that characterize the projections of specific objects. By using this iterative construction approach we limit the combinatorial complexity, and hence processing time, to a manageable

```
r(define parallelogram (s1 s2 s3 s4)
   (and (coincident sl-head s2-head)
        (coincident s2-tail s3-head)
        (coincident sl-tail s4-head)
        (coincident s3-tail s4-tail)
        (parallel sl-tail s3-tail)
        (parallel s2-tail s4-tail)
        ))
 (define triangle (si s2 s3)
   (and (coincident sl-head s2-tail)
        (coincident s2-head s3-tail)
        (coincident s3-head s1-tail)
        ))
 (define triangular-prism (s1 s2 .. 68
   (and (parallelogram s1 s2 s3 s4)
        (parallelogram s5 s2 s6 s7)
        (triangle 3 7 8)
        ))
 (define square (s1 s2 s3 s4)
   (and (parallelogram s1 s2 s3 s4)
        (same-size 1 2)
        ))
```

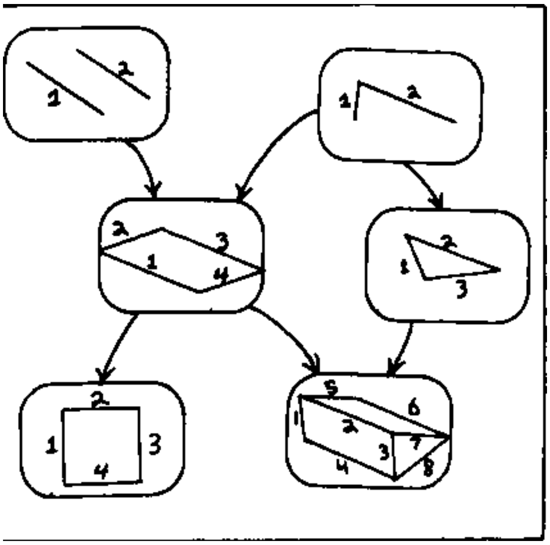

Fig. 2. Predictions as specialisations or combinations of other predictions.

level. This is because the system is never performing subgraph isomorphism analysis over large graphs it is always comparing combinations of small numbers of parts

For the experiments reported here, *parallelism* and *endpoint coincidence* are used as the initial set of simple and general predictions; and the iterative process is stopped when all of the predictions without derivatives (the ones at the top of the hierarchy) are either associated with single objects or, if they are satisfied by projections of more than one object, there are no deBcribable differences between their projections The prediction hierarchy compiler implemented iteratively combines predictions and then follows this up with specialization of the prediction to discriminate between objects that cannot be distinguished by parallel and end-point coincidence relations alone.

For each iteration of the combination process, the system isolates frequent combinations by (1) finding predictions that often appear together in the same projections and then (2) characterizing and representing the arrangement of the combination. The co-occurrences are found in the following fashion. All instances of all predictions are stored in data structures called *visibility maps.* There is a visibility map for each object; the maps are arrays of cells indexed by the two viewpoint parameters, making a discrete sampling of the view sphere about the object. Each cell lists prediction instances visible from the associated viewpoint and object; and with each prediction is a list of cells that contain it. To find frequent co-occurrences between some prediction P and other predictions, the system looks for predictions that appear in the same cells as P and accumulates the total number of cells for each that do. To keep the number of combinations analysed down to a manageable size, we consider only co-occurring *pairs* in whose arrangements the part-to-whole mappings overlap for at least one whole segment.

The combinations selected during a given iteration are those that are frequently occurring. Additionally, a heuristic is used to throw out predictions whose satisfaction during recognition creates dead ends in the matching process. The details of this heuristic can be found in [Bums87]. Very briefly, it dictates that the compiler adds a given prediction P only if, when P is satisfied by the projection of some object, that satisfaction almost always leads to a match of a derivative of P that is unique to that object.

The prediction hierarchy compiler design was tested on the models in Figure 1. The resulting hierarchy is shown in Figure 3. There are six levels of the hierarchy; the average path length between the initial nodes and goal nodes (object matches) is 3.9. The total number of nodes (predictions represented) is fifteen. Considering that the hierarchy is capable of being used to distinguish five objects from almost all viewpoints (with an average of 8 segments per view) — and the predictions are represented efficiently as combinations of simpler ones, this appears to be a reasonable result.

The iterative combination left the tall box and cube objects indistinguishable. This was corrected by the specialization process by adding segment length-ratio relations between two pairs of segments. This conjunction of two proportional relations was satisfied by the tall box projections over most of the view sphere, and satisfied by none of the projections of the cube.

5. Recognition

The object recognizer finds correspondences between segments detected in a given image and the segments of some model in the model base by an organized search of the prediction hierarchy and image data base. Like parsing and other interpretation problems, there are many ways to perform such a search (e.g., top-down, bottom-up or some combination of both). A fairly straight-forward method of search was implemented for the purpose of demonstrating the usefulness of the prediction hierarchy. The search proceeds in a bottom-up fashion by iteratively selecting a previously satisfied prediction (an image-prediction match), attempting to find additional evidence in the image to satisfy its derivative predictions (i.e., testing new constraints on already matched image segments for specializations and searching for parts for combinations), and storing any new image-prediction matches for further expansion. Figure 4 shows the results of this matching process using the hierarchy in Figure 3 and a synthetically generated set of image segments.

6. Conclusion

An object recognition system is presented that is designed to handle the computational complexity posed by a large model base, an unconstrained viewpoint and the structural detail inherent in a single view by extensive view analysis and the organization of predicted data into a PART-OF/IS-A structure called a *prediction hierarchy.* The original contribution of this paper is the automatic compilation, from a 3D model base, of a prediction hierarchy that can be used to recognize polyhedral objects. This

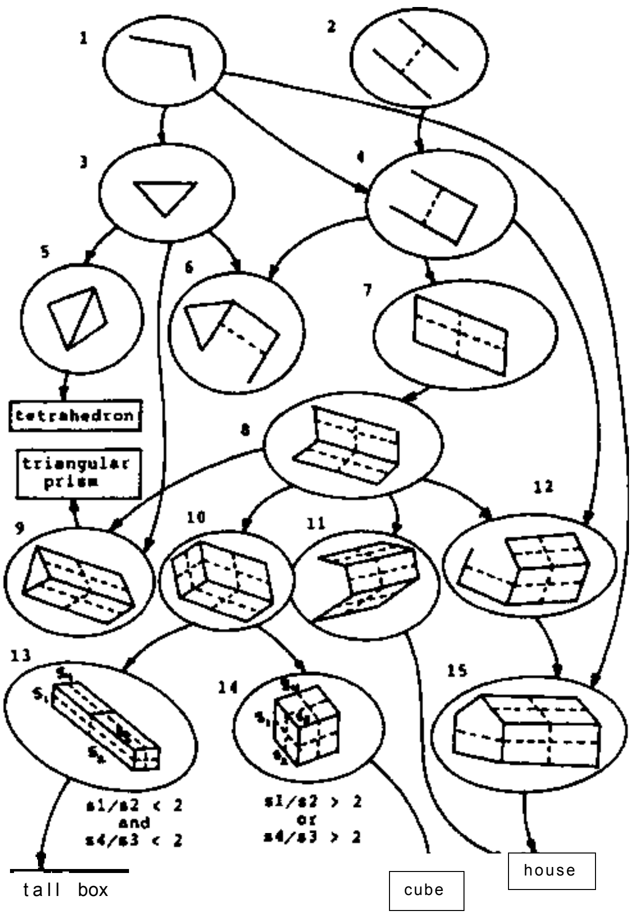

Fig. 3 The resulting prediction hierarchy compiled from views of the objects in figure 2. The node* represent predictions and arrows indicate combination and specialisation links. The predictions are represented graphically by segments and dashed lines for parallel relations.

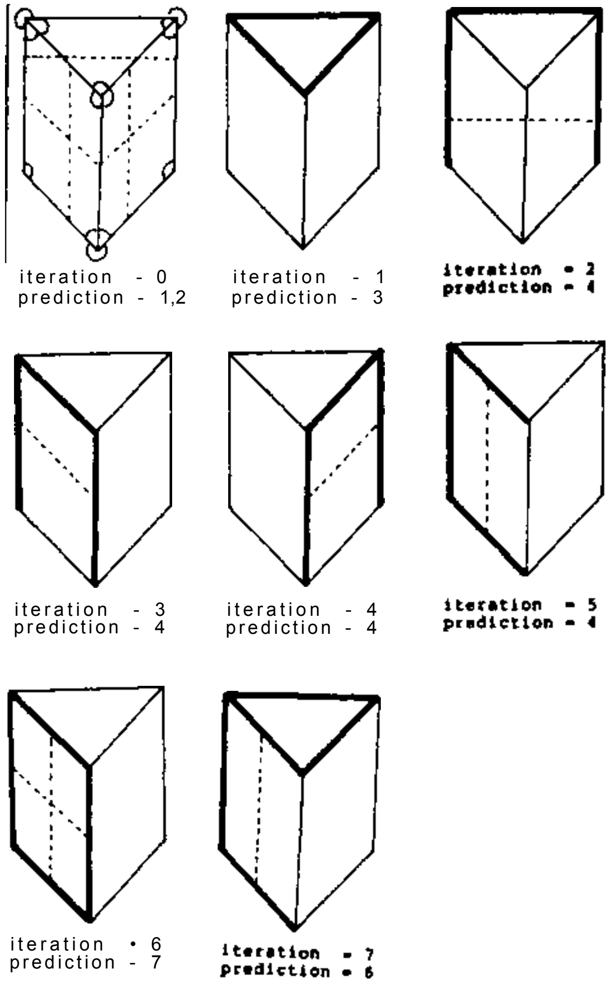

Fig. 4 Example run of the matcher using the prediction hierarchy in figure 10. The matcher is initialised (iteration=0) by finding all instances of the initial predictions (coincidence and parallel). The matcher then iteratively searches for matches between combinations and specialisations of already matched predictions and the image. A prediction unique to the triangular prism object was matched to the image at iteration 7.

research has been done in conjunction with studies of the visual properties of continuous surfaces |Callahan85] for the purpose of recognizing objects with curved parts and convexities. Further experiments will involve larger model bases, more complex models, and image noise. Current design refinements are centered on the cost/benefit analysis of adding a prediction to the hierarchy and the matching control strategies ([Weymouth86] and [Draper87]), including the use of image context for match selection.

References

1. Biederman, I., "Human Image Understanding: Recent Research and a Theory", CVGIP, vol. 32, pp. 29-73, 1985.

2. Brooks, R.A., "Symbolic Reasoning Among 3-D Models and 2-D Images", Artificial Intelligence, vol. 17, pp. 285-348, 1981.

3. Burns, J. and L. Kitchen, "Recognition in 2D Images of 3D Objects from Large Model Bases Using Prediction Hierarchies", forthcoming tech. rep., COINS Dept., Univ. of Mass. at Amherst.

4. Callahan, J. and R. Weiss, "A Model for Describing Surface Shape", CVPR, p. 240, 1985.

5. Chakravarty, I. and H. Freeman, "Characteristic Views as a Basis for Three-Dimensional Object Recognition", Proceedings of the SPIE, Vol. 338: Conf. on Robot Vision, Arlington, 37-45, 1982.

6. Cooper, P. and S. Hollbach, "Parallel Recognition of Objects Comprised of Pure Structure", DARPA Image Understanding Workshop, pp. 381-391, 1987.

7. Draper, B., R. Collins, J. Brolio, J. Griffith, A. Hanson, and E. Riseman, "Tools and Experiments in the Knowledge-Based Interpretation of Road Scenes", Proc. of the DARPA Image Understanding Workshop, 1987.

8. Fekete, G. and L.S. Davis, "Property Spheres: a New Representation for 3-D Object Recognition", IEEE Workshop on Computer Vision: Representation and Control, Annapolis, pp. 192-204, 1984.

9. Ikeuchi, K. "Precompiling a Geometrical Model into an Interpretation Tree for Object Recognition in Bin-picking Tasks", DARPA Image Understanding Workshop, pp. 321-339, 1987.

10. Lowe, D., "Visual Recognition from Spatial Correspondence and Perceptual Organisation", Proc. IJCAI-9, pp.953-959, 1985.

11. Marr, D., Vision, W.H. FVeeman, 1982.

12. Mulder, J.A., "Using Discrimination Graphs to Represent Visual Knowledge", University of British Columbia Laboratory for Computational Vision, T.R. 85-14, 1985.-

13. Weymouth, T.E., "Using Object Descriptions in a Schema Network for Machine Vision", Ph.D. Thesis, U. Mass., Amherst, 1986.