

3-D recognition via 2-stage associative memory

Randal C. Nelson
Department of Computer Science
University of Rochester

January 16, 1995

Abstract

We describe a method of 3-D object recognition based on two stage use of a general purpose associative memory and a principal views representation. The basic idea is to make use of semi-invariant objects called *keys*. A key is any robustly extractable feature that has sufficient information content to specify a 2-D configuration of an associated object (location, scale, orientation) plus sufficient additional parameters to provide efficient indexing and meaningful verification. The recognition system utilizes an associative memory organized so that access via a key feature evokes associated hypotheses for the identity and configuration of all objects that could have produced it. These hypothesis are fed into a second stage associative memory, which maintains a probabilistic estimate of the likelihood of each hypothesis based on statistics about the occurrence of the keys in the primary database. Because it is based on a merged percept of local features rather than global properties, the method is robust to occlusion and background clutter, and does not require prior segmentation. Entry of objects into the memory is an active, automatic procedure. We have implemented a version of the system that allows arbitrary definitions for key features. Experiments using keys based on perceptual groups of line segments are reported. Good results were obtained on a database derived from of approximately 150 images representing different views of 7 polyhedral objects.

1 Introduction: Philosophy and History

Those not doomed to repeat history may skip directly to section 2.

1.1 The Visual Recognition Problem

Object recognition is probably the single most studied problem in machine vision. It is also one of the most ill defined. The standard intuitive definition typically involves establishing a correspondence between some internal model of an object, and 2-D patterns of light produced by an imaging system. Attempts to formalize this notion however, generally lead to problem statements that are either unsolvable, or so restrictive as to be practically useless. For example, a statement such as “the ability to determine which members of an arbitrary set of objects contributed to the formation of a particular image, with no restrictions on environment, viewpoint, or lighting” is easily shown to represent an impossible task. Highly restricted versions such as “the ability to distinguish single images of an arbitrary group of ten polyhedral shapes with fewer than 100 faces and differing from each other by at least distance x in shape metric Z taken from an arbitrary viewpoint given Lambertian reflectance, point illumination, isolated presentation, and at least N pixels on target” tend to be unsatisfying, and even these generally contain unresolvable instances that are not easy to characterize.

There remains a pervasive intuition, stemming from human subjective visual experience, that visual recognition works, and that the bad cases, even in the general statement, are somehow pathological. This leads to a belief that a problem statement of the sort “the ability to recognize an image of an arbitrary normal object from any natural viewpoint in any reasonable environment most of the time” is sensible. The difficulty, of course, is making scientific sense of words such as “normal”, “natural”, and “reasonable”.

The operative word in the above paragraph is “works”, because it leads to the question “works for what?”. There is considerable evidence, from thirty years of research that “what” is not arbitrary establishment of correspondences between abstract object models and images. One idea is to interpret “works” in the context of a particular problem or problem area. The subjective terms in the previous paragraph can then be given meaning. A natural viewpoint is thus one that is expected to occur in the context of the application, a normal object is one whose identification is important, and a reasonable environment is one in which the application must be carried out. We have called this a behavioral approach [27]. In this context, recognition appears less as a process of solving a geometric/optical puzzle and more as a matter of using sensory information to get at stored state that permits the system to successfully interact with the environment, however success is defined. In other words, recognition is the evocation of memory.

1.2 Recognition as Memory Access

In this paper, we take the position that the phenomenon of recognition, rather than representing the establishment of a correspondence between an object in the world and some abstract model, should be viewed as sensory keyed access to a memory that is part of a behaving system. In this view, memory is any stored information that the system uses in

order to interact competently with the world. Such information can be procedural, analogical, or declarative, and at any level of abstraction. Thus we view evoking a particular activity on the same level as accessing a stored picture of an object or producing a sentence describing a visual scene. All three examples use sensory data to evoke stored information that is behaviorally relevant to the situation.

Viewing recognition as a process of memory access has several advantages. First, it frees us from what might be called the tyranny of the model. What we are referring to are situations in which a predetermined notion about the representation drives the development of the system and its applications rather than the other way around. For example, observing that object boundaries in an image often occur as long, straight structures, and noting the similarity to the mathematical concept of a line segment may be useful as far as it goes, but it does not imply that the next structuring element to try should be quadratic curves or ellipses. This is a prime example of false generalization from mathematics. A memory-based view does not, of course exclude the use of model correspondence, but neither does it restrict us to such a formalism if the application does not require it.

Second, taking a memory-based view can allow us to recognize and profit by relationships with other visual processes not generally thought of as recognition, for example, visual stabilization, and homing [28]. More generally, it provides a functional definition of recognition that ties in well with the behavioral notion of intelligence that has been gaining currency recently. It also provides a direct connection to learning and experience. Learning has traditionally been considered as orthogonal to recognition, which allowed the question of model acquisition to be finessed. If recognition is viewed as memory access, then the question of how the information in the memory was acquired and organized is substantially more immediate, forcing system designers to deal with it up front.

Finally, taking a memory-access approach forces us to develop a usable sensory-keyed memory architecture and access tools for it. The power and implications of such tools are not always apparent at the outset. For example, during the development of the system described here, we needed an associative memory that could be keyed by different types of sensory input - the original idea being to make use of multiple feature types. It is widely believed that raw sensory input (e.g. individual pixels), is not likely to be an effective memory key, and hence the system we developed allowed for preprocessing of the sensory data. It was only after implementing the system that we realized that passing sense data through the associative memory was effectively preprocessing it, and the result could be fed back in as a key. This observation was the basis for the second-stage use of the associative memory as an evidence accumulator that we describe below, but the process is general. The idea of recurrent association is, in fact, well known both in psychology and classical AI, but we had not considered using it in this context until we had the tool in our hands.

1.3 Previous Work

Addressing vision from the standpoint of behavior and memory is not a new idea. In fact, prior to the development of electronic computers, behavioral description was the only avenue available for investigating the phenomenon of vision. This precomputational work culminated in a series of books by Gibson [13; 14; 15] who advanced the central postulate that vision was essentially a modality that allowed biological systems to react to invariants

in the structure of the world. What Gibson overlooked, however, was the complexity of computing the visual invariants used as primitives. The first influential theory of computational vision, due to [23], essentially defined vision as the problem of determining what is where, and focussed almost entirely on the computational and representational aspects of the problem. Marr's theory of vision essentially described a staged computational architecture leading from image, to primal sketch, to 2-1/2 D sketch to invariant object centered descriptions. The processing hierarchy however, was static, and provided no structure for incorporating behavioral constraints. Moreover, the final, critical step, from 2-1/2 D image to object centered representation proved problematical, suggesting that something important was missing.

What distinguishes the recent interest in behavioral vision from the historical efforts is the commitment to establish it within a computational framework. The resurgence of interest in the behavioral aspects of intelligence is perhaps most clearly illustrated by the subsumption architecture proposed by Brooks [8; 9]. This paradigm is rather rigorously Gibsonian in that it explicitly disavows the notion of internal representation, relying instead on purely reactive strategies. The architecture works quite well for implementing low-level behaviors such as walking using simple sensors; however it now seems clear that higher level behaviors and more sophisticated sensory modalities such as vision, require some form of representation.

Recent work on active vision [1; 10; 3; 4] has focussed on how directed control of the sensor characteristics (e.g. eyes, or tactile receptors) can simplify the process of obtaining the desired information. Most work to date has focussed on the effect of the ability to move the sensor [11; 36] or dynamically change an internal focus of attention [31]. Work on purposive or behavioral vision [27; 26; 20] attempts to take the context of the task explicitly into account.

Much previous work in 3-D vision has focussed on model-based systems, on which there is a large literature. Notable recent examples are [22; 21; 19; 16]. Besl and Jain [5] give a survey of older work. The indexing techniques used in several of these systems have been recently analyzed, and the sensitivity of the techniques to various perturbations determined [17; 18]. Most model acquisition strategies have focussed on CAD-like techniques. The models are either entered by hand, or via a geometric reconstruction. There is a huge literature on shape from X (motion, stereo, shading etc.); however the goal in all this work has been to extract three-dimensional primitives such as point, line or plane descriptors, rather than functional representations.

There is little work on direct acquisition of 3-D representation from visual exploration, or on implicit representation of 3-D structure, though some research on recovery of explicit models from range sensors has been done [32; 34]. [6] address the problem of refining world and object representations during navigation by a mobile robot. [35] discuss a rigid body representation that is implicitly encoded in linear combinations of three views, and thus in principle automatically acquirable, but don't actually do it. Similarly, [33] propose a neural architecture for learning aspect-graph representation, but don't apply it to real objects. Work on automatic acquisition of 2-D representation has been more successful, since all the information is present (e.g. [7; 2; 12]) We expect these techniques to be useful here, since we essentially propose utilizing a collection of augmented 2-D representations.

There is some recent work that has a memory-based flavor too it. Rao and Ballard [30] describe an approach based on the memorization of the responses of a set of steerable

filters centered on, or located at key points of an object. Mel [24] takes a somewhat similar approach using a database of stored feature vectors representing multiple low-level cues. Murase and Nayar [25] find the major principal components of an image dataset, and uses the projections of unknown images onto these as indices into a recognition memory. The common theme behind all these approaches is extraction of a medium-sized set of parameters that are expected to vary slowly with changes in orientation, lighting, etc. followed by indexing into some set of examples organized by the same indices. Three dimensional structure is handled by using multiple examples. Our approach is along similar basic lines, but we index on robustly extractable parts and add a second indexing stage in order to accumulate evidence.

2 The Method

2.1 Overview

This paper describes an object recognition method based on two stage use of a general purpose associative memory, and a principal views representation of three dimensional objects. What we describe here is the application of the technique to the recognition of rigid 3-D objects, but the underlying principles are not dependent on rigid geometry, and we anticipate extending the system to handle non-rigid and statistical objects as well.

The approach makes use of semi-invariant objects we call *keys*. A key is any robustly extractable part or feature that has sufficient information content to specify a configuration of an associated object plus enough additional parameters to provide efficient indexing and meaningful verification. Configuration is a general term for descriptors that provide information about where in appearance space an image of an object is situated. For rigid objects, configuration generally implies location and orientation, but more general interpretations can be used for other object types. Semi-invariant means that over all configurations in which the object of interest will be encountered, a matchable form of the feature will be present a significant proportion of the time. Robustly extractable means that in any scene of interest containing the object, the feature will be in the N best features found a significant proportion of the time.

The basic idea is to utilize an associative memory organized so that access via a key feature evokes associated hypotheses for the identity and configuration of all objects that could have produced it. These hypothesis are fed into a second stage associative memory, keyed by the configuration, which maintains a probabilistic estimate of the likelihood of each hypothesis based on statistics about the occurrence of the keys in the primary database. The idea is similar to a multi-dimensional Hough transform without the space problems. In our case, since 3-D objects are represented by a set of views, the configurations represent two dimensional transforms. Efficient access to the associative memories is achieved using a hashing scheme.

The approach has several advantages. First, because it is based on a merged percept of local features rather than global properties, the method is robust to occlusion and background clutter, and does not require prior segmentation. This is an advantage over systems based on principal components template analysis, which are sensitive to occlusion and clutter. Second, entry of objects into the memory is an active, automatic procedure. Essentially, the system

explores the object visually from different viewpoints, accumulating 2-D views, until it has seen enough not to mix it up with any other object it knows about. Third, the method lends itself naturally to multi-modal recognition. Because there is no single, global structure for the model, evidence from different kinds of keys can be combined as easily as evidence from multiple keys of the same type. The only requirement is that the configuration descriptions evoked by the different keys have enough common structure to allow evidence combination procedures to be used. This is an advantage over conventional alignment techniques, which typically require a prior 3-D model of the object. Finally, the probabilistic nature of the evidence combination scheme, coupled with the formal definitions for semi-invariance and robustness allow quantitative predictions of the reliability of the system to be made.

2.2 General associative memory

We have been advocating a memory-based approach to recognition, but have not described just what is meant by the term. To some extent, all recognition algorithms employ memory in the recognition process, if only in the form of stored 3-D models. What distinguishes a memory-based approach, is that memory-lookup operations account for a large proportion of the computation employed in the recognition procedure (as opposed to, say, geometric operations or graph search).

Since our approach is based on an efficient associative memory, one of the first steps was to design and implement such a memory and verify that it satisfies our requirements. The basic operation we need is partial match association over heterogeneous keys. More specifically, we want a structure into which we can store and access (key, association) pairs where the key and association objects may be any of a number of disparate types. Associated with each object type employed as a key is a distance metric. The ideal query operation takes a reference key and returns all stored (key, association) pairs where the key is of the correct type and within a specified distance of the reference key in the appropriate metric. In practice, this ideal may have to be modified somewhat for efficiency reasons. In particular, highly similar association pairs may be merged in storage, and we may place a bound on the number of associations that are returned for any given query, or on the maximum separation that can be handled.

Our overall approach to the design of the memory was leave it as flexible as possible. In the current implementation, the memory is just a large array of buckets each of which can hold a variable number of (key, association) pairs. This allows a number of different access schemes to coexist. In particular, hashing, array indexing and tree search can all be implemented efficiently. Associated with each key type are functions defining a distance metric and a search procedure for locating keys in the memory. Thus if a certain key type has an efficient indexing method, it can be implemented for this type, rather than using a uniform but less efficient policy. This allows a large amount of flexibility in the system, and also permits new key types to be added efficiently in a modular fashion.

2.3 Key Features

The recognition technique is based on the the assumption that robustly extractable, semi-invariant keys can be efficiently recovered from image data. More specifically, the keys

must possess the following characteristics. First, they must be complex enough not only to specify the configuration of the object, but to have parameters left over that can be used for indexing. Second, the keys must have a substantial probability of detection if the object containing them occupies the region of interest (robustness). Third, the index parameters must change relatively slowly as the object configuration changes (semi-invariance). Many classical features do not satisfy these criteria. Line segments are not sufficiently complex, full object contours are not robustly extractable, and simple templates are not semi-invariant. We believe that features with the necessary properties can be found for a large number of situations. It may be necessary, however, to take the particular task and context into consideration. For example, in some applications, color cues may be sufficient. In others, where it is important to recognize orientation, shape features may be more important.

One conflict that must be resolved is that between feature complexity and robust detectability. In order to reduce multiple matches, key features must be fairly complex. However, if we consider complex features as arbitrary combinations of simpler ones, then the number of potential high-level features undergoes a combinatorial increase as the complexity increases. This is clearly undesirable from the standpoint of robust detectability, as we do not wish to consider or store exponentially many possibilities. The solution is not to use arbitrary combinations, but to base the higher level feature groups on structural heuristics such as adjacency and good continuation. Such *perceptual grouping* processes have been extensively researched in the last few years.

The use of semi-invariance represents another necessary compromise. From a computational standpoint, true invariance is desirable, and a lot of research has gone into looking for invariant features. Unfortunately, such features seem to be hard to design, especially for 2-D projections of general 3-D objects. We settle for semi-invariance and compensate by a combination of two strategies. First, we take advantage of the statistical unlikelihood of close matches for complex patterns (another advantage of relatively complex keys). Second, the memory-based recognition strategy provides what amounts to multiple representations of an object in that the same physical attribute of the object may evoke several different associations as the object appears in different configurations. The semi-invariance prevents this number from being too large. Possible keys for recognition of rigid 3-D objects include robust contour fragments, feature normalized templates, keyed color histograms, and normalized texture vectors.

Our current implementation is designed to recognize 3-D polyhedral objects on the basis of their shape, using a set of 2-D views as the underlying representation. This particular context derives from a robot assembly system we are implementing that serves off shape and geometric relationships between parts. The prototype manipulates a set of polyhedral pieces, which have no distinguishing markings aside from their shape. The vision system requirements are an ability to recognize which of several parts is present in a scene, and to localize important geometric features of the part. A shape-based description thus seemed appropriate. The keys we chose for this application are based on chains of line segments, variously referred to in the literature as polylines or supersegments. In particular, we first run a line segment finder on the image, and then extract perceptual groups of three segments whose properties are consistent with the hypothesis that they form a section of a 3-D boundary. We call such groups *3-chains*. The base segment of a 3-chain provides enough information to determine the 2-D configuration of any view of which it might be a part. In

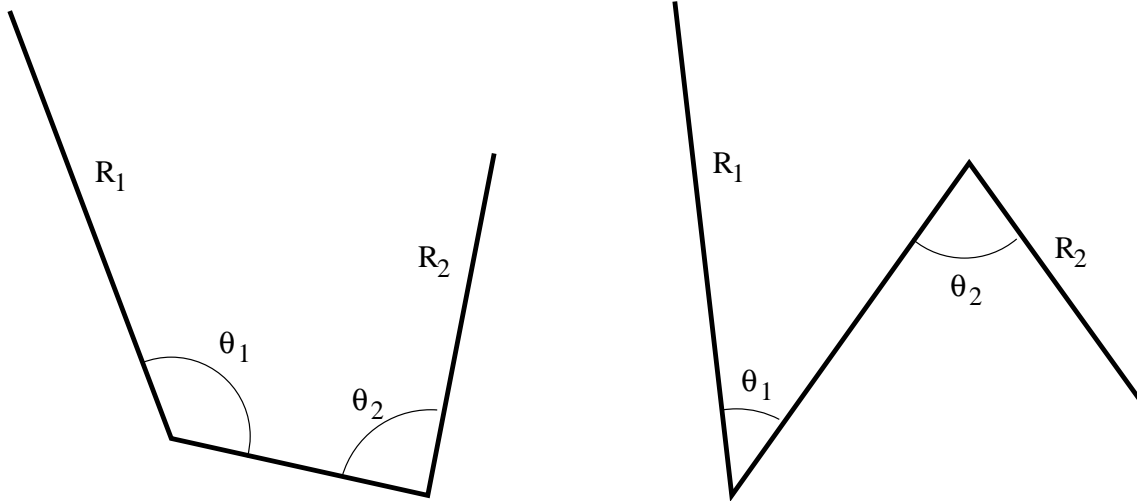


Figure 1: Examples of 3-chains showing invariant angles and length ratios

addition, associated with each 3-chain are two angles and two length ratios, which are absolute invariants for rigid 2-D transformations, and semi-invariant for 3-D rigid transformation of the projected object (see Figure 1). This use of segment chains is somewhat similar to the structural indexing of Stein and Medioni [12].

3 Recognition Algorithms

The basic recognition procedure consists of four steps. First, potential key features are extracted from the image using low and intermediate level visual routines. In the second step, these keys are used to access the associative memory and retrieve information about what objects could have produced them, and in what relative configuration. The third step uses this information, in conjunction with geometric parameters factored out of the key features such as position, orientation, and scale, to produce hypotheses about the identity and configuration of potential objects. Finally, these hypotheses are themselves used as keys into a second stage associative memory, which is used to accumulate evidence for the various hypotheses.

In the first step, the extraction of key features, one of the most significant issues involves the idea of active or selective processing. In general an object recognition algorithm will perform better if the search domain is restricted — that is, if the system is handed a region of interest thought to be more or less filled by some recognizable object, rather than simply being asked what recognizable objects occur in the scene. This is essentially the what/where dichotomy. The design of attentional operators which efficiently tag regions of high interest on the basis of low-level processing is consequently an important subject. A number of global or task-independent cues have been suggested, including gray-level blob detection, closed contour analysis, color contrast analysis, motion segmentation, and scale entropy measures. When incorporated into an autonomous system, such interest measures must be provided. For the purposes of testing the recognition system, we can have a user provide a window.

Such considerations raise the question, however, whether task-independent cues are always what is desired. All the previously mentioned cues already make an implicit assumption about the task - namely that we are looking for discrete physical objects. Perhaps the pre-attentive cues should be tailored to the task. Carried to its logical extreme, this idea suggests tailoring an attentional mechanism to respond best to a particular object, or even a particular configuration of an object. This is essentially the where task in the what/where dichotomy. The associative memory system proposed can be used to solve this task as well as the forward what task. One interesting approach is to use a reverse association to obtain the keys likely to be associated with an object or class of objects. These could then be used to design an optimal filtering process.

In the final step, an important issue is the method of combining evidence. The simplest technique is to use an elementary voting scheme - each piece of evidence contributes equally to the total. This is clearly not well founded, as a feature that occurs in many different situations is not as good an indicator of the presence of an object as one that is unique to it. An evidence scheme that takes this into account would probably display improved performance. The question is how to evaluate the quality of various pieces of evidence. An obvious approach in our case is to use statistics computed over the information contained in the associative memory to evaluate the quality of a piece of information. Having said this, it is clear that the optimal quality measure, which would rely on the full joint probability distribution over keys, objects and configurations is infeasible to compute, and we must use some approximation.

A simple example would be to use the first order feature frequency distribution over the entire database, and this is what we do. The actual algorithm is to accumulate evidence proportional to $\log(1 + 1/(kx))$ where x is the probability of making the particular matching observation as approximated from database statistics, and k is a proportionality constant that attempts to estimate the actual geometric probability associated with the prediction of a pose from a key. The underlying model is that the evidence represents the log of the reciprocal of the probability that the particular combination of features is due to chance. The procedure used makes an independence assumption which is unwarranted in the real world, with the result that the evidence values actually obtained are serious overestimates if interpreted as actual probabilities. However, the rank ordering of the values is fairly robust to distortion due to this independence assumption. Since only the rank ordering enters into the decisions made by the system, we are more comfortable with the scheme than might be expected.

Once a well founded evidence combination scheme is defined, the use of multi-modal information is relatively simple to implement. All that needs to be done is to define a new key type, and hook in the various routines needed to implement it. Issues of relative importance are subsumed by the evidence combination scheme. This easy use of multiple source of information was a primary factor in choosing to look at memory-based recognition. Characterization of the usefulness of the various key classes in different applications is an important piece of information for the integration system.

4 Model Acquisition

In the preceding discussion we have assumed that the associative memory already existed in the requisite form. However, one of the primary attractions of a memory-based recognition system is that it can be trained efficiently from image data. The basic process of model acquisition is simply a matter of providing images of the object to the system, running the key detection procedures on these images, and storing the resulting (key, association) pairs. The number of images needed may vary from one, for simple 2-D applications, to several tens for rigid object recognition, and possibly more for complicated non-rigid objects. This procedure has a number of advantages over existing schemes. First, it does not require a pre-existing 3-D model; just access to imagery of the object. It is even possible to use imagery that contains occluded or cluttered views of the object, though this requires some modification of the process. Second, the process is efficient. It essentially runs in time proportional to the number of pairs stored in memory. This is in contrast to many learning algorithms that scale poorly with the the number of stored items. Note that the term model acquisition is actually something of a misnomer since the representation of a particular object is typically distributed over many memory locations, and there is not necessarily any single structure that might be called a model gluing together all the parts.

Despite the simplicity of the basic procedure, there are a number of interesting research issues associated with the model acquisition process. First, the basic process may be modified somewhat. The most obvious modification would be to merge duplicate or near duplicate entries in the memory in order to conserve space. This is easily done by accessing a key in the database and examining the current associations before storing a new one.

A rather more interesting issue has to do with a tie-in to active vision. The necessary information for a two dimensional object can theoretically be obtained from a single image. In contrast, several views are needed for rigid 3-D objects. One way of getting these is simply for a human operator to guess at how many views are needed and from what angles, provide them, and then test the resulting system to make sure it is reliable. A more interesting approach however, is to use an active agent to explore the object and acquire the necessary views automatically. The basic idea is for the agent to examine the object in various configurations, adding new information whenever its current memory does not recognize the object. Doing this randomly would yield reasonably good performance quickly, however finding low-probability problematical configurations could take a long time. There is considerable room in this case, for knowledge directed exploration. for instance, the system could make use of knowledge that end-on views of highly elongated objects are apt to cause trouble. For non-rigid objects, the problem is even more interesting. Here, we have the possibility of not only active exploration for model acquisition, but for recognition as well. For example, in the case of a highly articulated object with no distinctive rigid subparts, an approach to recognition would be to attempt to manipulate it into a canonical form (e.g. stretched out) which would be more easily recognizable.

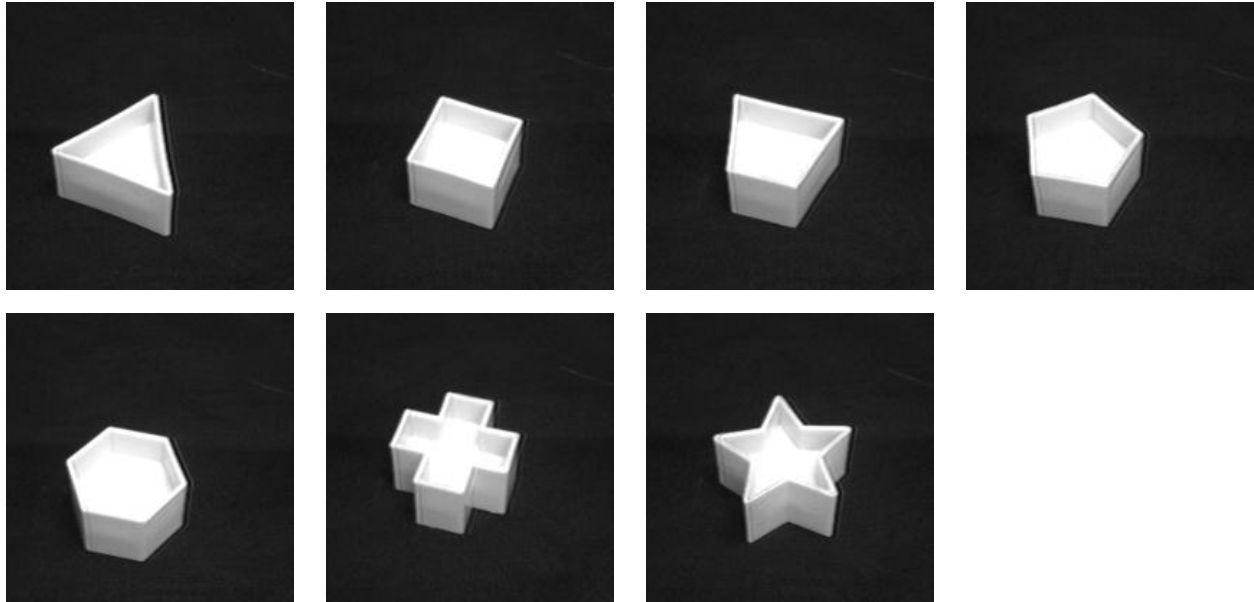


Figure 2: The polyhedral objects used in the test set

5 Experiments

Using the principles described above, we implemented a memory-based recognition system for polyhedral objects using 3-chains as the basic keys. Component segments were extracted using a stick-growing method developed recently at Rochester [29], and organized into chains. For objects entered into the database, the best 10 chains were selected to represent the object. The threshold on the distance metric between chains was adjusted so that it would tolerate approximately 15-20 degrees deviation in the appearance of a frontal plane (less for oblique ones). The practical considerations leading to this selection were to allow the system to discriminate pentagons from hexagons without requiring more than about 50 views for an object.

We performed experiments using a set of 7 polyhedral objects from a child's toy. These are shown in Figure 2, and, from the top appear as a triangle, a square, a trapezoid, a pentagon, a hexagon, a star, and a cross. Note, however that the objects are not simple prisms, but have an H-shaped cross section. This produces interesting edges and shadow effects when the objects are viewed from any angle other than straight down.

We obtained a training database of approximately 150 views of these objects from different directions ranging from 12 for the hexagon, to 60 for the trapezoid, and covering all viewing angles except straight-on from the side, since from that point of view a number of the objects are indistinguishable without measurements accurate to a few percent. The variation in the number of views needed is due to varying degrees of symmetry in the objects. All training images were acquired under normal room illumination, with the objects in isolation against a dark background. The training database was used to compile a segment-based associative memory for recognition of the objects.

We then subjected the recognition system to a series of increasingly stringent tests. Recall that the geometric design of the geometric indexing system ensures invariance to 2-D

translation, rotation, and scale down to the point where there are insufficient pixels to provide a good estimate of segment attributes. Invariance to out-of-plane rotations is provided by the combination of slightly flexible match criteria for the chain features coupled with multiple views. Robustness against clutter and occlusion is provided by the representation in terms of multiple features. The experiments were designed to test various aspects of this design.

A basic assumption made during these tests is that some other process has isolated a region of the image where a recognizable object may occur. We do not assume prior segmentation, but we do assume that only one, or at most a few objects of interest (as opposed to tens or hundreds) will occur in a window handed to the system. The system has a certain capability to state that it recognizes nothing in a window (don't know), and, in fact, tended to do this when given windows in which none of the known objects appeared. However, we have not statistically grounded this ability and hence the results reported here should be considered to be essentially forced choice experiments.

The first test, was simply to identify top down views of the objects, in various positions, scales, and rotations. This was essentially a test of the 2-D invariance built into the geometry. The system was tested first with a reduced database generated from 7 top down views (one for each object), and then with the full 3-D database, to ensure that the additional information stored did not produce enough cross talk to interfere with the recognition. Object presentations were under ordinary room lighting, with the objects isolated against a dark background. The system performed as expected in both cases, with no mistakes.

For the second test, we acquired 14 additional views of the objects, two of each, again isolated against a dark background, and taken from viewpoints intermediate between the ones in the database. The idea here is to test the 3-D rotation invariance. No errors were made in the 14 test cases, even between similar objects such as the square and the trapezoid, or the pentagon and the hexagon, despite the fact that we had anticipated some confusion in these cases. These results alone allow us to say that the system is probably at least 90 percent accurate in situations of this type. Results from other tests lead us to believe that the actual performance is, in fact, somewhat better.

In the third test we took a number of images containing multiple objects viewed from modest angles (45 degrees or less from overhead) under normal lighting against a dark background. An example is shown in Figure 3. We then supplied the system with windows containing one object and parts of others. Since the system performs no explicit segmentation of its own, the intent of this experiment is to test robustness against minor clutter. Examples of the sort of windows passed to the system are shown in Figure 4. In twenty plus tests, we observed no errors due to clutter. We did have one failure, but it was due to an object in the image being too small for the segment finder to find good boundary sets. We also tried examples with two objects in the window. In this case, the system typically identified one of the objects, and when asked what else was there identified the second as well.

The fourth experiment was a more severe clutter test. Here we took pictures of different objects held in a robot hand at various angles. Examples are shown in Figure 5. This was a hard problem for our system, and we obtained recognition rates only on the order of 75 percent - i.e. we saw a significant number of failures. On analysis, we found that the primary reason for failure was not crosstalk in the memory caused by clutter, but poor performance of the low-level feature identification process caused by the added complexity in the image. Thus the memory index has nothing to work on. Potential solutions involve improving the

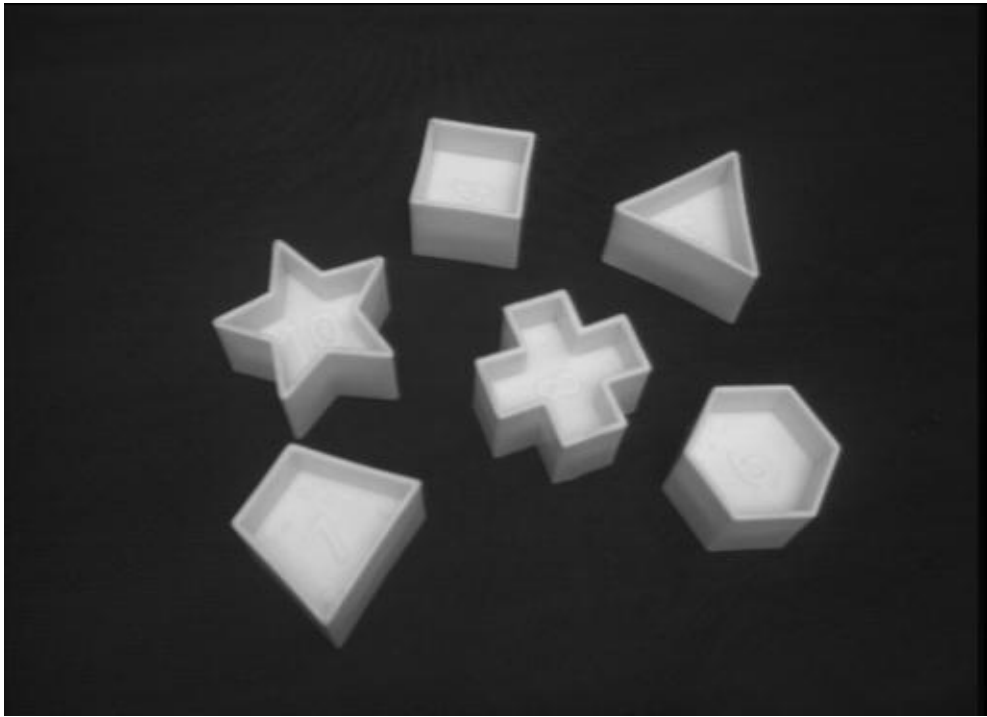


Figure 3: View of a group of objects

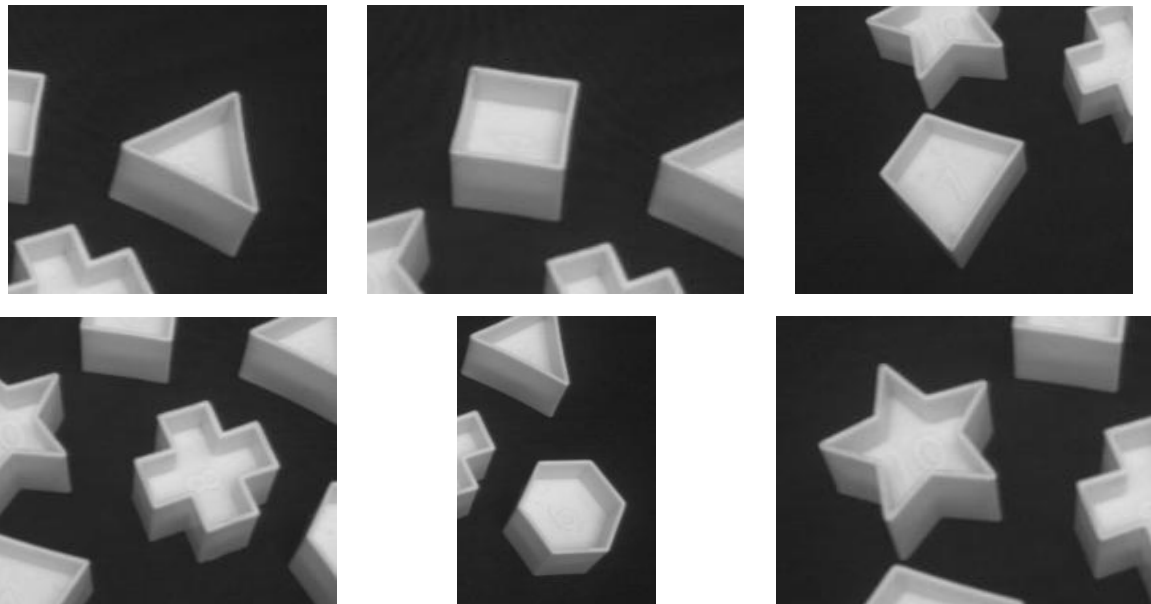


Figure 4: A set of windows containing objects and minor clutter. The central object was correctly identified in all these cases.

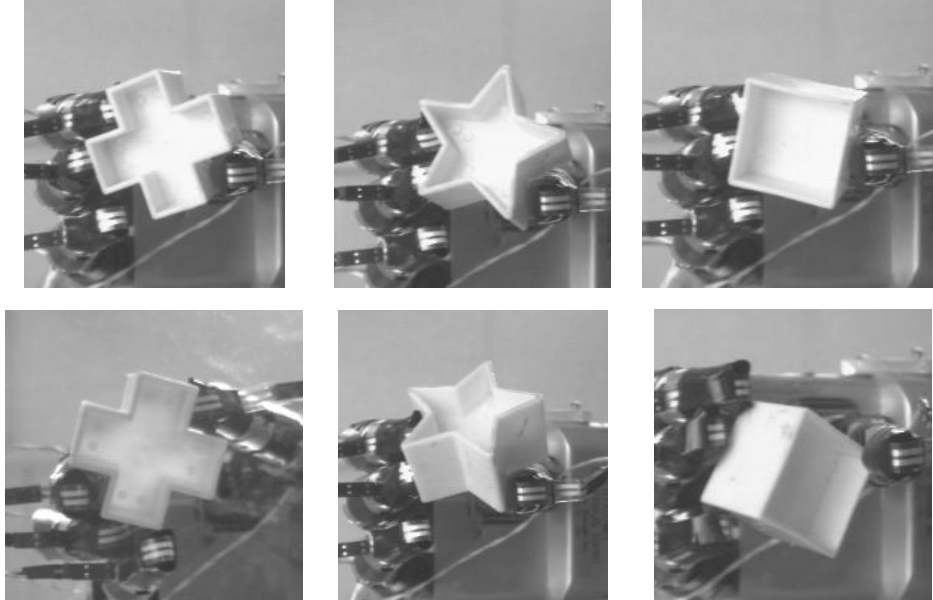


Figure 5: A set of windows containing objects help by a robot hand representing moderate clutter. The system successfully identified the object in all cases except the example in the lower right.

segment finder, which at present is strictly bottom up, and does no local grouping of its own, or using other features. Figure 6 shows a degree of clutter that broke the system completely. Recognition in windows from this image was essentially at the level of chance. Again, the failure is in the low-level processes.

Processing times were dominated by the low-level feature extraction. On a typical window containing one object plus clutter, the indexing process in the full database took a couple of seconds on a SPARC1. The low level processing could take a few tens of seconds, depending on the complexity of the image.

6 Conclusions and Future Work

In this paper we have argued for a memory-access interpretation of recognition, and proposed a general framework for memory-based recognition using a 2-stage association process. We have illustrated the concept by implementing a memory-based recognition system for 3-D polyhedral objects using chains of line segments as memory keys. The system actually performs quite well for a small database of 3-D shapes, and exhibits a certain amount of robustness against clutter and occlusion. When the algorithm fails, it is not due to crosstalk in the memory, but to failure of the low-level processes to extract robust features. We are currently engaged in embedding the system into a robotic manipulation system that we will use for assembly tasks.

The next step we plan to take is to generalize the system to use keys based on boundary curves rather than just straight segments. This work is nearly complete at the time. We also plan to incorporate multi-modal features into the database, including color and texture



Figure 6: An image with severe clutter. Performance of the system on windows drawn from this image was only slightly above chance level (2 out of 6 identified correctly).

as well as shape information. We anticipate that this will give us a capability to recognize less well structured objects such as leaves or clothing in addition to objects having a strictly defined shape.

There are a couple of other areas we eventually propose to address. One, which is not so much a research issue as an implementation issue is putting the various algorithms onto platforms having appropriate parallel hardware for the various processes involved in recognition. The implementation might well involve heterogeneous architectures, as the low-level and high level parts involve rather different computational processes. We are also interested in evaluating the performance of the system when embedded in a larger, real-time system.

There is also the issue of dealing with hierarchical classification of objects. As described above, the system handles object classes in a flat manner. There is no way of specifying that a particular small portion of an object may, in some application, be crucial for making a distinction (e.g. the license plate of a car). Similarly, there is no explicit way of dealing with multiple scales of structure in a single object. Adapting a memory-based recognition system do deal effectively with hierarchical class/subclass distinctions and multi-resolution structure is probably the single most interesting long-term goal.

References

- [1] Y. Aloimonos. Active vision. *International Journal of Computer Vision*, 2:333–356,

- 1988.
- [2] N. Ayache and O. Faugeras. Hyper: a new approach for the recognition and positioning of two-dimensional objects. *IEEE Trans. PAMI*, 8(1):44–54, January 1986.
 - [3] D. H. Ballard. Reference frames for animate vision. In *Proc. IJCAI*, pages 1635–1641, August 1989.
 - [4] D. H. Ballard and C. M. Brown. Principles of animate vision. *CVGIP*, 56(1):3–21, July 1992.
 - [5] P. J. Besl and R. C. Jain. Three dimensional object recognition. *ACM Computing Surveys*, 17(1):75–154, 1985.
 - [6] A. F. Bobick and R. C. Bolles. Representation space: An approach to the integration of visual information. In *Proc. CVPR*, pages 492–499, San Diego CA, June 1989.
 - [7] R. C. Bolles and R. A. Cain. Recognizing and localizing partially visible objects: The local-features-focus method. *International Journal of Robotics Research*, 1(3):57–82, Fall 1982.
 - [8] R. A. Brooks. Achieving artificial intelligence though building robots. Technical Report TR 899, MIT, 1986.
 - [9] R. A. Brooks. A robust layered control system for a mobile robot. *IEEE Journal of Robotics and Automation*, 2:14–23, April 1986.
 - [10] P. J. Burt. Smart sensing within a pyramid vision machine. *IEEE Proceedings*, 76(8):1006–1015, 1988.
 - [11] T. J. O. David J. Coombs and C. M. Brown. Gaze control and segmentation. In *Proc. AAAI Qualitative Vision Workshop*, Boston MA, August 1990.
 - [12] F. Stein and G. Medioni. Efficient 2-dimensional object recognition. In *Proc. ICPR*, pages 13–17, Atlantic City NJ, June 1990.
 - [13] J. J. Gibson. *The Perception of the Visual World*. Houghton Mifflin, Boston, 1950.
 - [14] J. J. Gibson. *The Senses Considered as Perceptual Systems*. Houghton Mifflin, Boston, 1966.
 - [15] J. J. Gibson. *The Ecological Approach to Visual Perception*. Houghton Mifflin, Boston, 1979.
 - [16] W. E. L. Grimson. *Object Recognition by Computer: The role of geometric constraints*. The MIT Press, Cambridge, 1990.
 - [17] W. E. L. Grimson and D. P. Huttenlocher. On the sensitivity of geometric hashing. In *3rd International Conference on Computer Vision*, pages 334–338, 1990.

- [18] W. E. L. Grimson and D. P. Huttenlocher. On the sensitivity of the hough transform for object recognition. *IEEE PAMI*, 12(3):255–274, 1990.
- [19] D. P. Huttenlocher and S. Ullman. Recognizing solid objects by alignment with an image. *International Journal of Computer Vision*, 5(2):195–212, 1990.
- [20] K. N. Kutulakos and C. R. Dyer. Recovering shape by purposive viewpoint adjustment. In *Proc. CVPR*, pages 16–28, Champaign Il, June 1992.
- [21] Y. Lamdan and H. J. Wolfson. Geometric hashing: A general and efficient model-based recognition scheme. In *Proc. International Conference on Computer Vision*, pages 238–249, Tampa FL, December 1988.
- [22] D. G. Lowe. Three-dimensional object recognition from single two-dimensional images. *Artificial Intelligence*, 31:355–395, 1987.
- [23] D. C. Marr. *Vision*. W. H. Freeman and Co., 1982.
- [24] B. Mel. Object classification with high-dimensional vectors. In *Proc. Telluride Workshop on Neuromorphic Engineering*, Telluride CO, July 1994.
- [25] H. Murase and S. K. Nayar. Learning and recognition of 3d objects from appearance. In *Proc. IEEE Workshop on Qualitative Vision*, pages 39–50, 1993.
- [26] R. C. Nelson. Qualitative detection of motion by a moving observer. *International Journal of Computer Vision*, 7(1):33–46, November 1991.
- [27] R. C. Nelson. Vision as intelligent behavior: Research in machine vision at the university of rochester. *International Journal of Computer Vision*, 7(1):5–9, November 1991.
- [28] R. C. Nelson. Visual homing using an associative memory. *Biological Cybernetics*, 65:281–291, 1991.
- [29] R. C. Nelson. Finding line segments by stick growing. *IEEE Trans PAMI*, 16(5):519–523, May 1994.
- [30] R. P. Rao. Top-down gaze targeting for space-variant active vision. In *Proc. ARPA Image Understanding Workshop*, pages 1049–1058, Monterey CA, November 1994.
- [31] R. D. Rimey and C. M. Brown. Where to look next using a bayes net: Incorporating geometric relations. In *Proc ECCV*, pages 542–550, May 1992.
- [32] R. K. Ruud M. Bolle and D. Sabbah. Primitive shape extraction from range data. In *Proc. IEEE Workshop on Computer Vision*, pages 324–326, Miami FL, Nov-Dec 1989.
- [33] M. Seibert and A. M. Waxman. *Learning Aspect Graph Representations from View Sequences*. Morgan Kaufmann, 1991.
- [34] F. Solina and R. Bajcsy. Recovery of parameteric models from range images. *IEEE Trans. PAMI*, 12:131–147, February 1990.

- [35] S. Ullman and R. Basri. Recognition by linear combinations of models. *IEEE Trans. PAMI*, 13(10), 1991.
- [36] D. Wilkes and J. Tsotsos. Active object recognition. In *Proc. CVPR*, pages 136–141, Champaign IL, June 1992.