Large-Scale tests of a Keyed, Appearance-Based 3-D Object Recognition System *

Randal C. Nelson
Andrea Selinger
Department of Computer Science
University of Rochester
Rochester, NY 14627
(nelson, selinger)@cs.rochester.edu

Abstract

We describe and analyze an appearance-based 3-D object recognition system that avoids some of the problems of previous appearance-based schemes. We describe various large-scale performance tests and report good performance for full-sphere/hemisphere recognition of up to 24 complex, curved objects, robustness against clutter and occlusion, and some intriguing generic recognition behavior. We also establish a protocol that permits performance in the presence of quantifiable amounts of clutter and occlusion to be predicted on the basis of simple score statistics derived from clean test images and pure clutter images.

Key Words: Object recognition, Appearance-based representations, Visual learning.

1 Introduction

Object recognition has been an important and much-researched problem in the study of both machine and human vision. Until recently, the most successful computational work on object recognition has used model-based approaches in which the image is matched against explicitly represented 3-D geometric models. Notable recent examples are Lowe (1987), Lamdan & Wolfson (1988), Huttenlocher & Ullman (1990), and Grimson (1990) [21; 19; 17; 10]. The 3-D geometric models on which these systems are based are both their strength and their weakness. On the one hand, explicit models provide a framework that allows powerful geometric constraints to be used to good effect; for example, matching just a few point features (3 to 5, depending on the projection and calibration information) in an image to a 3-D model completely determines the location of all other model features. On the other, model schemas are generally severely limited in the sort of objects that they can represent, and obtaining the models is typically a difficult and time-consuming process. Analyses of

*Support for this work was provided by ONR grant N00014-93-I-0221, and NSF IIP Grant CDA-94-01142

the performance of some of these schemes is given by Grimson & Huttenlocher (1990a,b) [12; 11]. There has been a substantial amount of work on the automatic acquisition of geometric models, mostly with range sensors, (Bolle 1989, Solina & Bajcsy 1990, Bobick & Bolles 1989) [29; 32; 2] but also visually, for various representations (Ullman & Basri 1991, Bolles & Cain 1982, Ayache & Faugeras 1986, Stein & Medioni 1990) [34; 3; 1; 9]. However, these techniques are limited to a particular geometric schema, and even within their domain, especially with visual techniques, their performance is often unsatisfactory.

Appearance-based object recognition methods have been proposed in order to make recognition systems more general, and more easily trainable from visual data. Most of these operate by comparing a two-dimensional, image-like representation of object appearance against many prototype representations stored in a memory, and finding the closest match. They have the advantage of being fairly general, and often easily trainable. In recent work, Poggio & Edelman (1990) [27] have recognized wire objects and Brunelli & Poggio (1993) [4] have recognized faces using appearance models. Rao & Ballard (1994) [28] describe an approach based on memorizing the responses of a set of steerable filters to images of objects. Mel (1994) [22] takes a somewhat similar approach using a database of stored feature vectors representing multiple low-level cues. Murase & Nayar (1993) [23] find the major principal components of an image dataset, and use the projections of unknown images onto these as indices into a recognition memory. Huang & Camps (1997) [15] have recently adapted this approach to segmented image regions, thus obtaining some tolerance to clutter and occlusion. Schmid & Mohr (1996) [30] have recently reported good results for an appearance based system with a local-feature approach similar in spirit to what we use, though with different features and without using feature likelihood measures in the evidence combination scheme. Both Nayar's and Mohr's approaches carry out recognition tests only over a 1-dimensional range of views rather than over the full 2-D viewing sphere as we do in the tests on our model. In a slightly less image-like approach, Chen & Stockman (1996) [6] use contour features to index a 3-D model of local structure. This produces hypotheses that are then subject to global model verification. Since the mean rank of the correct hypothesis is typically around 20 (in the best version), much of the power of the technique derives from the 3-D verification step. Another feature-based example is a recent generalization of the alignment method by Huttonlocher & Lorigo (1996) [16] which finds consistent point matches via linear combination of model feature images.

In general, the appearance-based approach has proven to be a useful technique; however because matches are generally made to representations of complete objects, some such methods have been more sensitive to clutter and occlusion than is desirable, and require that the image be first segmented into regions that represent entire objects. In order to overcome the dependence on good whole-object segmentation, evidence combination schemes such as Hough transform methods (and other voting techniques), have been employed to allow evidence from disconnected parts to be effectively combined. However, the size of the voting space increases exponentially with the number of degrees of visual freedom. Difficulties deriving from the size of this space make it difficult to apply such techniques directly when more than about 3 DOF are involved, thus limiting the use of the technique for 3-D object recognition, which generally involves at least 6 DOF.

We describe a method that, by combining an appearance database of semi-local, intermediate-level key features with a Hough-like evidence combination technique, overcomes the problems

with clutter and occlusion observed in traditional memory-based methods. The method also addresses the problems of space and false-positives seen in voting methods for high DOF problems. The method makes double use of a general purpose associative memory. This stores both semi-invariant, local objects called *keys* associated with object hypotheses, and object configuration hypotheses associated with evidence.

This system demonstrates robust recognition of a variety of 3-D shapes, ranging from sports cars and fighter planes to snakes and lizards over a full spherical or hemispherical range of views and over changes in scale. (More specifically, the system demonstrates recognition with full, 6DOF, orthographic invariance.) This is in contrast to results by Murase & Nayar (1993) [23] where only one of the two out-of-plane rotational degrees of freedom is spanned. We report the results of several large-scale performance tests, involving, over 2000 separate test images. In these experiments we investigate variation in performance with respect to increasing database size, clutter, and occlusion. We develop a statistical model for predicting the performance in a variety of situations from a few basic measurements of score distributions for clean test images and pure clutter. We report results on a generic recognition experiment, where the system is trained on several objects in each of several classes, and asked to classify example objects from the same generic classes, but not in the training set. We also discuss the biological relevance of the model.

2 The Method

2.1 Overview

Our system represents 3-D objects as a modest set of flexible, 2-D views each derived from a training image. For each view, the visual appearance of an object is represented as a loosely structured combination of a number of local context regions keyed by distinctive key features, or fragments. For the moment, a local context region can be thought of as an image patch surrounding the key feature and containing a representation of other features that intersect the patch. The idea is, that under different conditions (e.g. lighting, background, or small changes in orientation) the feature extraction process will find some of these distinctive keys, but in general not all of them. Also, even with local contextual verification, such keys may well be consistent with a number of global object hypotheses. However, we show that the fraction of the keys that can be found by existing feature extraction processes is frequently sufficient to identify objects in the scene, once the global evidence is assembled. This addresses one of the principle problems of object recognition, which is that, in any but rather artificial conditions, it has so far proved impossible to reliably segment the image into regions corresponding to whole objects on a bottom-up basis. In this paper, local features based on automatically extracted boundary fragments are used to represent multiple 2-D views (aspects) of rigid 3-D objects, but the basic idea could be applied to other features and other representations.

In more detail, we make use of distinctive local features we call *keys*, embedded in a local context. A key is any robustly extractable part or feature that has sufficient information content to specify a configuration of an associated object together with enough additional, pose-insensitive (sometimes called semi-invariant) parameters to allow efficient indexing into

the database. The local context amplifies the power of the feature by providing a means of verifying whether the key is likely to be part of a particular object. This local verification step is critical, because the invariant parameters of the key features are relatively weak evidence. If only this weak evidence is used in an evidence combination scheme, a proliferation of high-scoring false object hypotheses results. This is a well known problem with voting schemes, but can be alleviated if the voting features are sufficiently powerful.

The basic recognition strategy is to use a database (here viewed as an associative memory) of key features embedded in local contexts, which is organized so that access via an unknown key feature evokes associated hypotheses for the identity and configuration of all known objects that could have produced such an embedded feature. These hypotheses are fed into a second stage associative memory, keyed by configurations, which lumps the hypotheses into clusters that are mutually consistent within a loose global context, thus providing flexibility in the representation. In the current implementation, this looseness is obtained by tolerating a specified deviation position, size, and orientation of key features relative to a nominal position.

The secondary database maintains a probabilistic estimate of the likelihood of each cluster 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 that arise in an explicit decomposition of the parameter space. In our case, since 3-D objects are represented by a set of views, the configurations represent two dimensional rigid transforms of specific views. As mentioned above, this local verification step gives the voting features sufficient power to substantially ameliorate well known problems with false positives in Hough-like voting schemes.

The approach has several advantages. First, because it is based on a merged percept of local contexts rather than global properties, the method works well in the presence of occlusion and background clutter, and does not require prior segmentation of the image into whole objects. 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 can be an active, automatic procedure. Essentially, the system can explore the object visually from different viewpoints, accumulating 2-D views, until it has seen enough not to confuse it with any other object in the database. This is an advantage over conventional alignment techniques, which typically require a prior 3-D model of the object. 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 output of the system is a list of hypotheses as to the identity and pose of objects in the scene, ranked by the total evidence for each hypothesis. Each hypothesis also retains pointers to the supporting key features. At this point, it would be possible to undertake a top-down verification of the top hypotheses, making a broader search for features that should be present, but did not contribute evidence to the hypothesis (e.g. due to differing bottom-up boundary segmentation). We do not currently perform this step; however, unlike appearance-based systems based on whole-object appearance, the structure of our representation is such that this could be performed to advantage, and such a step has the potential to significantly improve the performance of the system as a whole. The results given should thus be interpreted as representing the power of an initial hypothesis generator or indexing

system.

2.2 Biological Relevance

A natural question is to what degree the approach we have taken is consistent with what is known about object recognition in the human visual system. At the low level, of course, the algorithms are likely to look very different, simply because machines and brains are trying to optimize performance on very different hardware architectures. On the other hand, the higher level question of whether the representation used in the brain for fast recognition is best modeled as view-based or (3-D) object-based, is one that may be important to address. Note that we are careful to qualify the task as that of fast recognition - the common operation that people carry out in 100 milliseconds or so, as opposed to the more deliberative, "deductive" sort of recognition (e.g. "there are a bunch of these objects around a table, about the right height to sit on, maybe they are some kind of chair"), which may very well use different techniques and representations.

At the neurophysiological level, very little concrete is known about object level representations, and certainly not enough to resolve the question of whether the implementation is view-based or object based. Recent results on the existence of object specific cortical cells in monkeys and other animals, specifically on cells that seem to be selective for particular views of faces are intriguing, but still too preliminary to say much about the underlying implementation (Oram & Perret 1994, Perret & Oram 1993, Gross 1992) [25; 26; 13].

On the other hand, there is a body of psychophysical work that is relevant to the question. Some early work addressed the problem of mental rotation of images of 3-D objects, and determined that people were, in general able to do this, and in a way that took increasing amounts of time as the required rotation was increased (Shepard & Cooper 1982, Tarr & Pinker 1989) [31; 33]. This was taken as evidence for the existence of internal 3-D object models. More recent work, however, while confirming that people are indeed able to perform mental operations that seem most consistent with the existence of 3-D, object-centered representations, has raised questions about whether these representations are what is used for fast recognition (Bulthoff et al. 1995, Edelman & Bulthoff 1992) [5; 7]. It can be plausibly argued that the 3-D representations are used for example, for planning manipulations, while fast recognition uses a separate representation.

The work most relevant to our approach is that of Bulthoff, Edelman & Tarr (1995) [5]. In their research they looked at the expected performance of several representations used in 3-D object recognition and compared it with the results obtained by psychophysical experiments. They wanted to see whether response times and/or error rates are equivalent for all changes in viewpoint or are systematically related to the magnitude of changes in viewpoint.

On the basis of these experiments, they argue that the representation most similar to the one used by the human visual system is the viewpoint dependent two-dimensional representation. Methods using this representation try to achieve object constancy by storing multiple 2-D viewpoint-specific representations and using mechanisms for matching input images to stored views or to views derived computationally from stored views. All methods using viewpoint dependent two-dimensional representations may be considered as computa-

tional variants of the empirically-based multiple-views-plus-transformation (MVPT) theory of recognition (Tarr & Pinker 1989) [33]. Since there is evidence indicating that this process can result in the same dependence of the response time on the pose of the stimulus object as obtained in the mental rotation experiments, MVPT can be considered as a psychological model of human performance that predicts recognition behavior under specific conditions.

When presented with a new image, our method looks for the stored view that is most similar to the image. This can be thought of as an interpolation of stored views. A lower error rate is obtained for familiar test views (if we test our system on the training images the error is 0) than for novel test views, depending on the distance from the novel view to the nearest familiar stored view. By storing a large number of views that are placed at modest (20 degree) distances from each other we managed to obtain a very low error rate even in the case of novel views.

The human visual model advocated by Bulthoff et al. is similar to our system in that it represents objects by small sets of canonical views and uses a variant of mental rotation to recognize objects at attitudes other than the canonical ones. Each canonical view is essentially an image-based representation of the object as it is seen from a certain viewpoint and might be augmented by limited depth information. Their experiments showed that even in the case of human observers, generalization to novel views was severely limited, with performance dropping to chance levels at a misorientation of about 40° relative to familiar views (Edelman & Bulthoff 1992) [7]. Also, in this human visual model, as in certain computational models, e.g. Edelman & Weinshall (1991) [8], views that "belong" together are more closely associated with each other. Computationally, this method of recognition is analogous to an attempt to express the input as an interpolation of the stored views. In this case, recognition normally requires neither 3-D reconstruction of the stimulus, nor the maintenance of a library of 3-D models of objects. Instead, information sufficient for recognition can be found directly in the 2-D image locations of object features.

In psychological experiments there are several levels of category organization in recognition performance. The *basic level* is the most salient according to psychological criteria. The *entry level* is the first categorical label generally assigned to a given object. Objects whose recognition implies finer distinctions than those required for entry-level categorization are said to belong to a *subordinate level*. The patterns of response times and error rates in recognition experiments are influenced by the category level at which the distinction between the different stimuli is to be made. If the subject is required to classify the stimulus (i.e. to determine its entry level category) error rates and response times are viewpoint invariant. If the task is to identify an object (i.e. to discriminate one individual from other, visually similar objects sharing parts and spatial relations), error rates and response times are viewpoint-dependent.

In this parlance, our experiments deal with the entry-level classification of objects. When the objects present some similarities, the categorization needs to be done on the subordinate level and the error level gets higher. For example in the generic experiments we sometimes saw confusion between planes and fighter jets.

In summary, our system has characteristics, especially in the high-level, approach that are consistent with recent psychophysical results concerning human recognition. This similarity, of course, applies equally to a number of view-based computational models. Whether the distinctive features of our system - the use of a loose assembly of local contexts keyed by

distinctive features - are more consistent with biology than other view-based approaches, cannot be said with certainty on the basis of existing evidence.

2.3 Key Features

The recognition technique is based on the assumption that robustly extractable, pose-insensitive keys can be efficiently recovered from image data. More specifically, the keys must possess the following characteristics. First, they must be complex enough to specify the configuration of the object, and to have additional parameters left over that can be used for indexing and matching. 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 (insensitivity to pose). 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 pose-insensitive.

A basic conflict that must be resolved is that between feature complexity and robust detectability. In order to reduce multiple matches, 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 spatial adjacency and good continuation. Such *perceptual grouping* processes have been extensively researched in the last few years (Kubovy 1997, Havalder et al. 1996, Lowe 1986) [18; 14; 20]. Our keyed local contexts can be viewed as an example of perceptual grouping.

The use of pose-insensitive, but not truly invariant features 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 curved 3-D objects. We settle for pose insensitivity 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 features). Second, the appearance-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 views. The pose insensitive nature of the features prevents this number from being too large.

We currently make use of a single key feature type consisting of robust boundary fragments (curves). These fragments, which are probabilistically segmentable in similar views of an object, are placed in a local context consisting of a square image region, oriented and normalized for size by the key curve, which is placed at the center. Each local context contains a representation of all other segmented curves, key or not, that intersect it. We call these local contexts *context patches*. In more detail, a curve-finding algorithm is run on an image, producing a set of segmented contour fragments broken at points of high curvature. The longest curves are selected as key curves, and a fixed-size template (21 x 21) is constructed. A base segment determined by the endpoints (or the diameter in the case of closed or nearly closed curves) of the key curve occupies a canonical position in the template. All image curves that

intersect the normalized template are mapped into it with a code specifying their orientation relative to the base segment. Since the templates are of fixed size, regardless of the size of the keying curve, this is, to a certain extent, a multiple resolution representation.

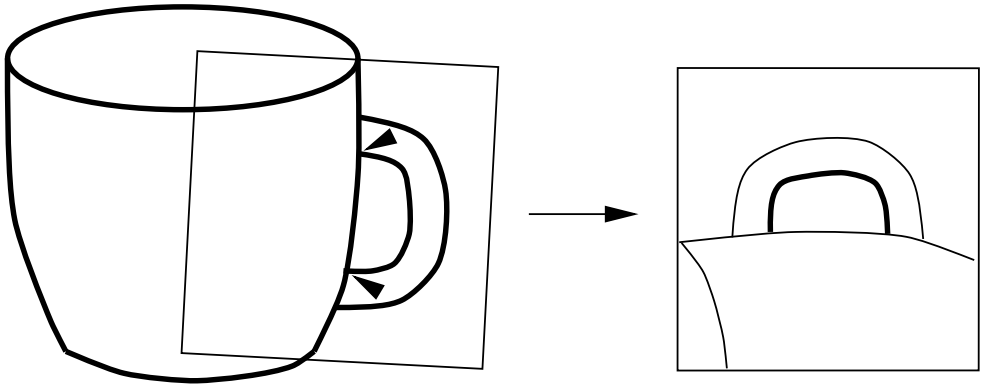

Figure 1: Example of a patch generated by a boundary fragment in a simple cup sketch. In this case the keying fragment is the inner loop of the handle, shown in canonical position in the center of the template square. The template represents not just the keying fragment, but all portions of other curves that intersect the square.

Figure 1 shows how a single patch context is generated by a boundary fragment in a simple sketch of a cup. Figure 2 shows the patches that would be generated by the indicated set of boundary fragments in the sketch. The left-hand side of the figure shows the key curves displaced, while preserving loose global relationships. This illustrates the sort of fragmentation that is implicit in our representation. Note that the representation is redundant, and that local contexts arising from large curves may contain all or most of the curves in an object. This redundancy is important, since the output of the segmentation process may vary over the range of views that need to be covered by a particular 2-D training view, and a substantial fraction of the key fragments may not be matchable in a new view.

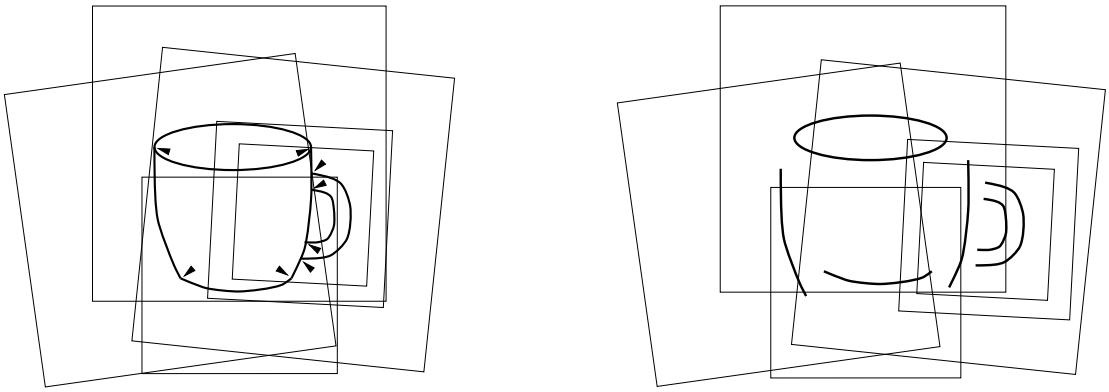

Figure 2: Right, example of patches generated by a set of boundary fragments for the cup sketch; arrows indicate the location of the fragment endpoints or diameters. Left, key fragments displaced, while preserving loose global relationships. Our representation implicitly contains this kind of distortion.

Verifying a local context match between a candidate patch keyed by a curve fragment and a stored model patch involves taking the model patch curve points and verifying that a curve point with similar orientation lies nearby in the candidate template. Essentially this amounts to loose directional correlation. The matching process is modified in that curves that lie parallel to the base segment and within half a diameter of it do not contribute to the match. The reason for this is that close parallel structure is so common in the world, (narrow objects, shadows, highlights, steep gradient effects) that such structures contribute little evidence while adding enormously to the "accidental" match population.

2.4 Overall Recognition Procedure

In order to recognize objects, we must first prepare a database against which the matching takes place. To do this, we first take a number of images of each object, covering the region on the viewing sphere over which the object may be encountered. The exact number of images per object may vary depending on the features used and any symmetries present, but for the patch features we use, obtaining training images about every 20 degrees is sufficient. To cover the entire sphere at this sampling requires about 100 images. For every image so obtained, the boundary extraction procedure is run, and the best 20 or so boundaries are selected as keys, from which patches are generated and stored in the database. Currently, the "best" features are simply the largest; other distinctiveness measures could be used as well. With each patch is associated the identity of the object that produced it, the viewpoint it was taken from, and three geometric parameters specifying the 2-D size, location, and orientation of the image of the object relative to the key curve. This information permits a hypothesis about the identity, viewpoint, size, location and orientation of an object to be made from any match to the patch feature.

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 database memory (via hashing on key feature characteristics and verification via local context), 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 regarding position, orientation, and scale, to produce hypotheses about the identity and configuration of potential objects. These "pose" hypotheses serve as the loose global contexts into which information is integrated. This integration is the fourth step, and it is performed by using the pose hypotheses themselves as keys into a second associative memory, where evidence for the various hypotheses is accumulated. Specifically, all global hypotheses in the secondary memory that are consistent (in our loose sense) with a new hypothesis have the associated evidence updated. After all features have been so processed, the global hypothesis with the highest evidence score is selected. Secondary hypotheses can also be reported.

2.5 Global Context and Evidence Combination

In the final step described above, an important issue is the method of combining evidence within a loose global context. The simplest technique is to use an elementary voting scheme - each feature (local context patch) consistent with a pose contributes equally to the total

evidence for that pose. 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. For example, with 24 3-D objects stored in the database, comprising over 30,000 context patches, we find that some image features match 1000 or more database features, even after local context verification, while others match only one or two. An evidence combination scheme should take this into account. An obvious approach is to use statistics computed over the information contained in the associative memory to evaluate the quality of a piece of information. 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 thus we must use some approximation.

One approach is to use the first order feature frequency distribution over the entire database in a Bayesian framework. This, with minor modifications, is what we do. In the following discussion, the term "feature" should be taken to mean the entire key curve plus local context, since this is what is being matched. Also recall that the pose hypotheses serve as the global contexts within which evidence is accumulated. The resulting algorithm, which we derive below, is to accumulate evidence, for each match supporting a pose, proportional to $F \log(k/m)$ where m is the number of matches to the image feature in the whole database, and k is a proportionality constant that attempts to make m/k represent the actual geometric probability that some image feature matches a particular patch in the pose model by accident. F represents an additional factor proportional to the square root of the size of the feature in the image, and the 4th root of the number of key features in the model. This factor was introduced to improve performance on the basis of empirical analysis. These modifications capture certain aspects that seem important to the recognition process, but are difficult to model using formal probability.

Before proceeding with a more formal derivation, it is worth noting that a simple way to understand the source of the logarithmic term is to interpret the total evidence score as representing the log of the reciprocal of the probability that the particular assemblage of features (local context patches) is due to chance. If the features are independent (which they are not, but we do not have any better information to use) then we just multiply the probabilities. Equivalently, to keep the actual values small, we can add the logarithms. Because the independence assumption is unwarranted in the real world, the evidence values actually obtained are far too low if interpreted as actual probabilities. However, the rank ordering of the values, which is all that is important for classification, is fairly robust to distortion due to this independence assumption.

More formally, we can derive the above formula from a Bayesian evidence combination model using the match frequency as an estimate of the prior probability of the feature type, and assuming independence of observations. To see this, let $A_{(O,\theta,\phi,x,y,s,\rho)}$ be a pose hypothesis corresponding to the sort of object we are accumulating evidence for in the secondary memory; namely, a particular object O, seen from a particular viewpoint parameterized by two angles (θ, ϕ), with center at a particular image location (x, y), having a certain size s, and with planar orientation ρ. (Note that this basically implies an orthographic model of rigid objects.) Implicitly, all the parameters have tolerances associated with them. To simplify notation in the following, we will drop the subscripted index parameters. Now associated with each hypothesis A are a set of model features (patches) $\mathbf{M} = \{M_1, M_2, \ldots\}$. There is also a set of image features $\mathbf{I} = \{I_1, I_2, \ldots\}$ that have been extracted from the image under

consideration. Each image feature I_i may or may not match any particular model feature M_j. We will designate the occurrence of such a match by $X_{(A,i,j)}$. Again, to simplify notation, we will drop the subscripted index parameters in the following, and refer to different matches by a single index where necessary.

Now suppose we have a set of image features (context patches). By matching these against the model features associated with A in the database, we can generate a set $\mathbf{X} = \{X_1, X_2, \ldots\}$ of possible matches. In order to maintain the fiction of independence, we impose the condition that, for a given hypothesis, there can be at most one match involving each image feature, and at most one match involving each model feature. This makes sense intuitively - it is a basic graph-matching constraint used in explicit model-matching approaches. In a Bayesian framework, we are interested in maximizing the probability $P(A|X_1 \wedge X_2 \wedge \cdots)$ over all possible poses A. (Note that the X_i refer to different matches for different A). If the X_i are independent, then Bayes rule gives us that

$$P(A|X_1 \wedge X_2 \wedge \cdots) = P(A)\frac{P(X_1|A)P(X_2|A)\cdots}{P(X_1)P(X_2)\cdots}$$

Note that in the above analysis we do not attempt to include a contribution for image features that do not match a model feature. This is valid under the assumption that non-matching image features are generated by some random clutter process, and thus any feature not in the model is equally likely to occur whether or not the particular hypothesis holds. This may not always be quite true - it is possible to use information of the sort "teapots are almost never associated with triangles", but this is hard to get at and to use, and we do not try in the current system.

Now the quantities $P(A|X_i)$ in the numerator can be interpreted as the probability of a particular model feature finding a match in an image of the object taken within the parameter tolerances. We cannot figure out from first principles what this is, but by looking at the number of key features that are matched in correct classifications of images of objects within the hypothesis tolerances, we observe empirically that these probabilities are somewhere between a quarter and a half, for features that are in the model, and that they do not seem to depend strongly on the particular object or feature. The quantity $P(A)$ is the prior probability of a particular pose, and in the absence of other information, we can assume all poses in the range of consideration (there are cutoffs on the x and y values, and on the size s) to be equally likely.

The quantities in the denominator, $P(X_i)$, represent the prior probabilities of various feature matches. As mentioned above, we have strong evidence that these are not all equal. Some sorts of patch features (for instance those involving parallel or enclosing structures) occur far more frequently than others. However, we have a natural method for estimating these. Since we find all matches for an image feature in the database in any case, we just take the prior probability to be proportional to the number N_m of such matches. The other factors involved are the total number of image features N_i, the total number of features in the database N_d, and a geometric probability factor G, which we assume to be constant for now:

$$P(X_i) = GN_i\frac{N_m}{N_d}$$

If all we are looking for is rank ordering, then we can compare the various hypothesis probabilities by summing the logarithms of the reciprocal prior probabilities. Thus

$$\log(P(A|X_1 \wedge X_2 \wedge \cdots)) = \sum_i \log(\frac{k}{N_m}) + C$$

where k and C are constants. Since the logarithm is monotonic, the rank ordering is preserved. This is just the weighting we used initially, and the constant k, can now be seen as a lumped estimate of the quantities assumed constant above. The constant k was initially determined using a rough calculation of the geometric probability G that randomly occurring features would match in position, orientation, and scale given the tolerances associated with the hypotheses, and increased somewhat to compensate for expected non-uniformity of feature distributions coming from purported objects. We later ran a series of tests where we varied G over nearly three orders of magnitude, and found that the algorithm was quite insensitive to the exact value within about an order of magnitude around the initial educated guess (which was 1/200).

The preceding discussion describes, in theory, how we combine evidence for all feature matches associated with a given pose hypothesis and a set of evidence. We now want to find the maximum of this measure over all possible poses. Clearly, we cannot directly evaluate all possible pose hypotheses: there are too many of them (e.g. 20 objects x 100 viewpoints x 100 image locations x 20 orientations x 10 sizes = 40,000,000 poses to check). This is where the secondary memory comes into play. In our algorithm, the indexing into the secondary associative memory functions as an efficient way of accumulating the evidence for all poses (global contexts) that have any evidence consistent with them at all (most possible poses have none, for a given set of evidence). Specifically, as mentioned above, once a pose hypothesis is formulated, all previously formulated hypotheses that are consistent with it, within our sense of loose global structure, are retrieved and have the associated evidence updated. (If there are no consistent hypotheses, a new one is generated.) For the specific case of rigid objects, consistency is defined as being within certain set bounds on the rigid transformation parameters (currently 20 degrees rotation, 1/10 of the object size in translation, and 20% in scale).

2.6 Implementation

Using the principles described above, we implemented a recognition system for rigid 3-D objects. The system needs a particular shape or pattern to index on, and does not work well for objects whose character is statistical, such as generic trees or pine cones. Component boundaries were extracted by modifying a stick-growing method for finding segments developed recently by Nelson (1994) [24] so that it could follow curved boundaries. Figure 3 shows the performance of the boundary finding algorithm on a good image of some of the objects we later use for testing. Training images generally produce contours of about this quality.

The system is trained using images taken approximately every 20 degrees around the sphere, amounting to about 100 views for a full sphere, and 50 for a hemisphere. The key detection procedure is run on these images, and the resulting (key, association) pairs stored in a database. The number represents a tradeoff between the storage requirements of

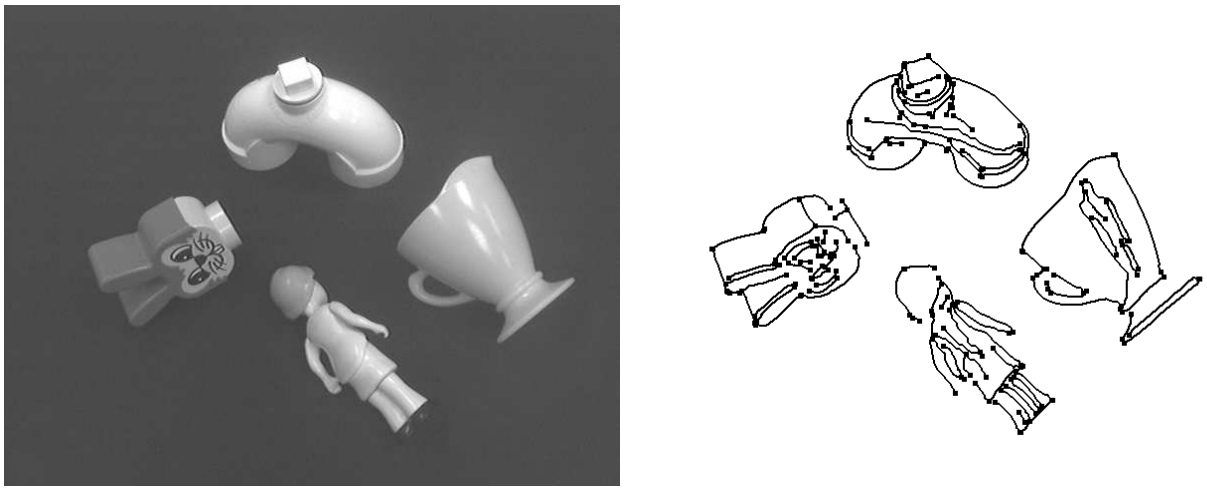

Figure 3: Curves extracted by boundary finding algorithm. Dots mark the ends of curves. These are the sort of features on which the recognition system is based.

increasing the number of views, and the computational requirements of making the templates sufficiently flexible to match between views. For objects entered into the database, the best 20 key features were selected to represent the object in each view. The thresholds on the distance metrics between features were adjusted so that they would tolerate approximately 15-20 degrees deviation in the appearance of a frontal plane (less for oblique ones).

Figure 4 illustrates the operation of the recognition system on an image of a cup from the test set. The boundary extraction system finds 15 curves in the image; of these, 5 key patches contribute to the best hypothesis (which happens to be the "correct" answer in all the experiments where this image was used). This image illustrates several of the problems that make matching key curves a probabilistic process: boundaries that wash out, ambiguous "corners", boundaries due to highlights, and boundaries produced by shading effects.

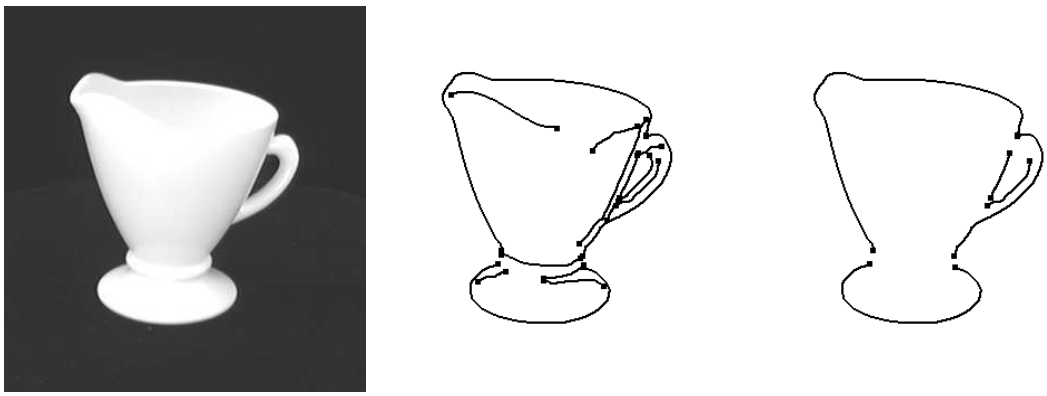

Figure 4: Illustration of the operation of the recognition system. The first panel shows an image of a cup given to the system. The second shows the curves found in the test image by the boundary extraction system. The third panel shows the curves which keyed matching patches that contributed evidence to the best (and correct) hypothesis.

3 Experiments

3.1 Variation in Performance with Size of Database

One measure of the performance of an object recognition system is how the performance changes as the number of classes increases. To test this, we obtained test and training images for a number of objects, and built 3-D recognition databases using different numbers of objects. The objects used were chosen to be "different" in that they were easy for people to distinguish on the basis of shape. Data was acquired for 24 different objects and 34 hemispheres. The objects are shown in Figure 5. The number of hemispheres is not equal to twice the number of objects because a number of the objects were either unrealistic or painted flat black on the bottom which made getting training data against a black background difficult.

Clean image data was obtained automatically using a combination of a robot-mounted camera, and a computer controlled turntable covered in black velvet. Training data consisted of 53 images per hemisphere, spread fairly uniformly, with approximately 20 degrees between neighboring views. The test data consisted of 24 images per hemisphere, positioned in between the training views, and taken under the same good conditions. Since this is a test of scaling of performance under increasing database size, use of such good test images is appropriate. What is actually being tested then, is invariance under out-of-plane rotations which are the most interesting and difficult of the 6 orthographic freedoms. Note that this system does not deal with extreme perspective, though modest perspective distortion presents no problem.

To elaborate, we note that the remaining planar invariances are mathematically guaranteed by the structure of the representation, once past the curve extraction stage. The curve extraction process itself has very minor sensitivity to rotation and translation. Scale sensitivity in the curve finder is somewhat greater, since the decision as to whether there is a corner or a smooth curve can depend on the magnification. In general, the scale sensitivity is not severe. We repeated the 6-object test using new pictures taken from a 50% greater distance and observed no increase in the error rate. In cases where we wanted the system to work over a large range of scales, we simply ran the boundary finder at multiple scales, which generally doubled the number of features, and effectively eliminated spatial scaling problems (except for very small scales, where the curve finder does not operate well at all).

No particular attempt was made either to include or avoid pathological views, that is, ones where identification is difficult for people, and we have found that most of our objects have a few of these. Our data is probably biased against containing these views, since they tend to be perpendicular to natural stable attitudes, and we did not take any pictures from lower than 10 degrees elevation due to physical and optical constraints on viewing an object on a turntable. Analysis of the mistakes made by the system in the scaling test revealed that a substantial proportion of the mistakes arose in cases of such odd views.

We ran tests with databases built for 6, 12, 18 and 24 objects, shown in Figure 5, and obtained overall success rates (correct classification on forced choice) of 99.6%, 98.7% 97.4% and 97.0% respectively. The total number of training aspects for the 24 object database was 1802, which compares favorably to the number of aspects stored in any general recognition database reported in the literature. The results are summarized in the following table.

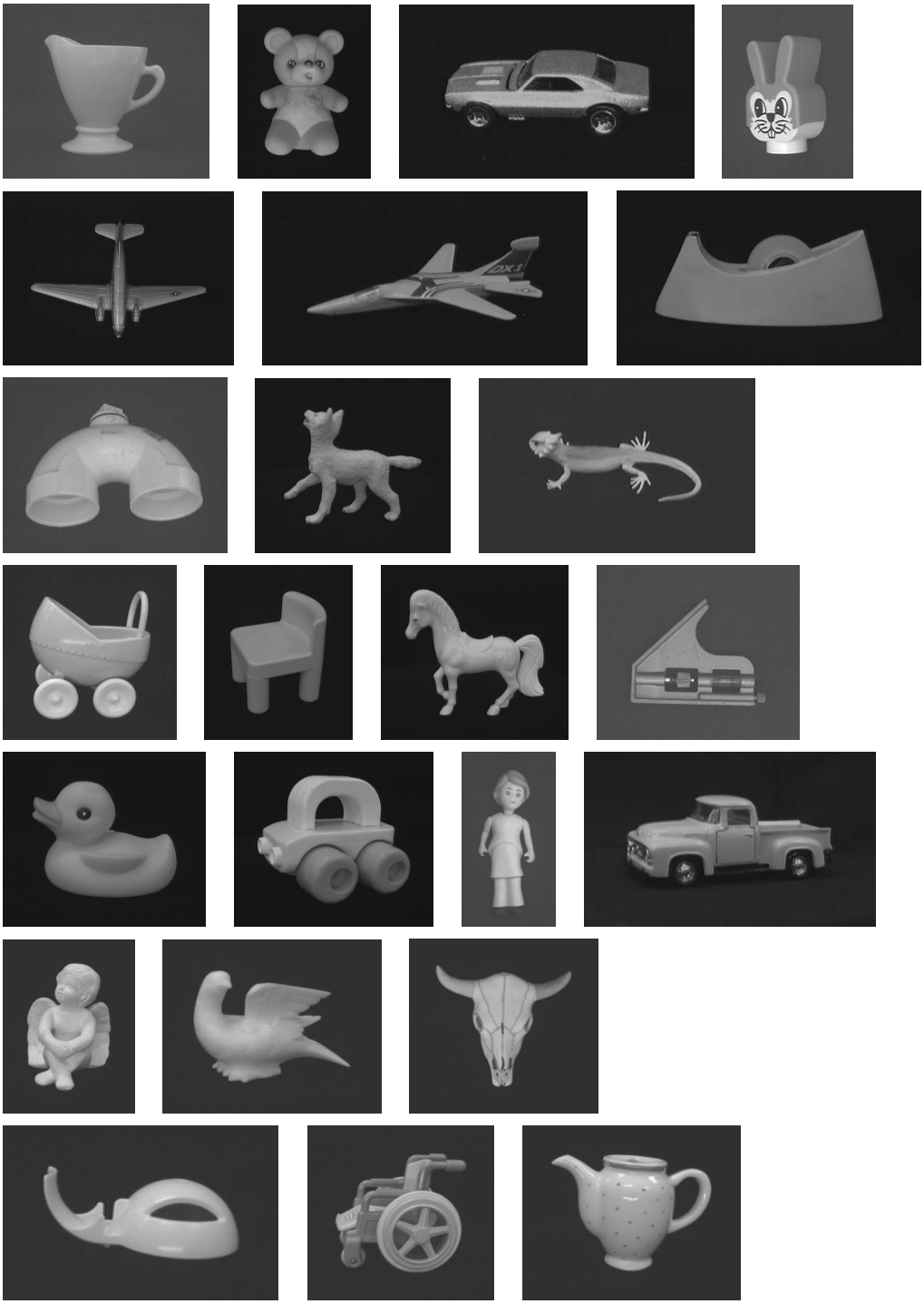

Figure 5: The objects used in testing the system set

number of objects	number of hemi-spheres	number of test images	number of correct	percent correct
6	11	264	263	99.6
12	18	408	403	98.7
18	26	576	561	97.4
24	34	768	745	97.0

Table 1: Performance of forced-choice recognition for databases of different sizes

Overall, the performance is fairly good. A naive estimate of the theoretical error trends in this sort of matching system would lead us to expect a linear increase in the error rates as the size of the database increased (best-case). Our results are consistent with this, but we do not have enough data points to provide convincing support for a linear trend. More important, perhaps, is the fact that the error rates are not uniform. For the 24 object case, 9 out of 23, or over one third of the total errors are due to the wolf and the horse, which are the most complicated objects in the set in terms of both structural and non-structural (i.e. texture and shadow) features.

The above results represent the output of an indexing system using the "best guess" without whole object verification. It is of some interest to know how far down the correct hypothesis is in the cases where the top-ranked hypothesis was not correct. For the 24-object test, there were a total of 23 misses. Of these, the correct hypothesis was in the top 10 in 20 cases. Details are presented in Table 2. This suggests that the error rate could be improved by an order of magnitude by adding a verification step applied to the top hypotheses.

Rank	Correct hypotheses at rank
1	745
2	6
3	4
4	0
5	3
6	3
7	1
8	3
9	0
10	0
>10	3

Table 2: Rank of correct classification hypotheses

3.2 Performance in the presence of clutter and occlusion

The feature-based nature of the algorithm provides some immunity to the presence of clutter and occlusion in the scene; this, in fact, was one of the design goals. This is in contrast to appearance-based schemes that use the structure of the full object, and require good prior segmentation. The algorithm, in fact seems reasonably robust against both modest clutter and occlusion.

In order to evaluate this, we ran a series of experiments involving increasingly difficult examples, starting with isolated clutter on dark and light fields, where we could easily generate exhaustive test sets, simple occluded scenes, and then graduating to examples involving both clutter that is not trivially segmentable and minor occlusion. The problem with these later images is that, unlike examples with added dark-field clutter, it is difficult to generate large numbers of such images of "equivalent" difficulty, and covering all pose variations. Hence these examples unavoidably have a "look ma, no hands" nature.

In order to generate a more principled method of predicting performance in the presence of clutter and occlusion, we generated a number of images containing pure clutter, but no known objects. We then looked at the statistics of expected best scores for the process when run on pure clutter with varying numbers of features. By comparing these statistics to those for the performance on clean examples, we can generate estimates for the probability of various sorts of errors. This is the subject of a later section.

3.2.1 Simple clutter

The first experiment involved modest dark-field clutter in high quality images, that is, extra objects or parts thereof in the same image as the object of interest. Note that in this case individual whole objects could be segmented out relatively easily, and the clutter dealt with that way. The point of the experiment, however, is to test, over the full spherical range, how the system performance is affected by extra features arising from extraneous structure. We will present examples later showing the system working in cases where segmentation is not easy.

We ran a series of tests where we acquired test sets of the six objects used in the previous 6-object case in the presence of non-occluding clutter. In this experiment, clutter typically produced about 50% of the features passed to the recognition system. Examples of the test images are shown in Figure 6. Out of 264 test cases, 252 were classified correctly which gives a recognition rate of about 96%, compared to 99% for uncluttered test images. A confusion matrix is shown in Figure 3.

In a second experiment, to illustrate that the dark background is irrelevant, we took pictures of the objects against a light background. Clutter in these images, again amounting to about 50% of the features, arises from shadows, from wrinkles in the fabric, and from a substantial shading discontinuity between the turntable and the background. The objects could still probably be segmented, but it is not quite so easy in this case. Examples of the test images are shown in Figure 7, and the boundaries found in Figure 8. showing the substantial numbers of clutter curves arising from shadowing and wrinkles, even on this fairly nice background. All the images shown were classified correctly.

Out of 264 test cases, 236 were classified correctly which gives an overall recognition rate

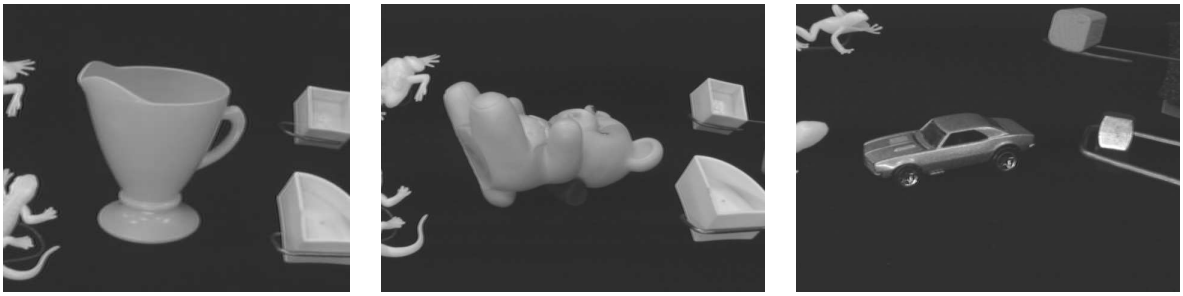

Figure 6: Examples of test images with modest dark-field clutter

class name	index	smpls	0	1	2	3	4	5
cup	0	48	47	0	1	0	0	0
toy bear	1	48	2	46	0	0	0	0
sports car	2	24	0	0	24	0	0	0
toy rabbit	3	48	0	0	1	47	0	0
plane	4	48	0	0	2	1	45	0
fighter	5	48	0	0	1	0	4	43
Total hypotheses for class			49	46	29	48	49	43

Table 3: Error matrix for object classification experiment with clutter. Each row shows how the test images for a particular object were classified.

Figure 7: Examples of test images on light background, with shadows and minor texture

Figure 8: Curves found by boundary extraction algorithm in light background images

of about 90%, which is not as good as the dark-field results. However, almost half the errors were due to instances of the toy bear, where many of the main boundaries are invisible due to gray level similarity of object and background. If this case is excluded, the rate is about 94%, which matches the dark-field results. A confusion matrix is shown in Figure 4.

class name	index	smpls	0	1	2	3	4	5
cup	0	48	44	2	0	1	1	0
toy bear	1	48	3	32	1	5	2	5
sports car	2	24	0	0	24	0	0	0
toy rabbit	3	48	1	0	0	47	0	0
plane	4	48	0	0	0	0	45	3
fighter	5	48	0	0	1	0	3	44
Total hypotheses for class			48	34	26	53	51	52

Table 4: Error matrix for light field classification experiment. Each row shows how the test images for a particular object were classified.

3.2.2 Simple occlusion

The current system is not designed to deal with arbitrary occlusion; specifically occlusion that breaks up all or most of the key features will cause the recognition process to fail. That said, for objects that are complex enough to contain recognizable subparts, the system can deal with significant amounts of occlusion. For our database, many of the objects are sufficiently complex that they can be chopped in half, for instance, and still recognized by the system. Figure 9 shows examples from the six object database of the sort of occluded instances the system can handle.

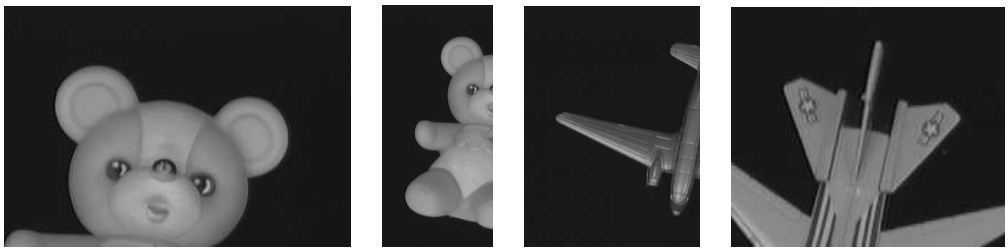

Figure 9: Examples of manageable occluded images

3.2.3 More difficult clutter

To demonstrate that the recognition system can operate in the presence of moderately textured backgrounds, we took pictures of objects from the 6 object database on three different textured backgrounds: A ceiling tile, a floor tile, and a piece of crumpled cloth. These disrupt different aspects of the algorithm. The ceiling tile, with the small dark regions, breaks up the low-level boundary finding process when one of the small regions intersects a boundary on the silhouette. Granted, some modification of the low-level algorithm could

probably fix this particular case, but this was not done. The floor tile just produces lots of extraneous boundary fragments. The crumpled cloth produces a background with large regions of different shadings and strong curvature gradients of the sort that would tend to break any attempt at whole-object segmentation. Figure 10 shows examples from the 6 object database on the different textures. All examples shown were classified correctly by the system.

Figure 10: Examples of manageable images with textured backgrounds

To demonstrate that the clutter resistance is not dependent on whole-object segmentability, we next took a number of individual pictures of known objects with adjacent and partially overlapping distractors. These pictures are not trivially segmentable, but on the other hand it is not easy, as in the previous cases, to automatically generate hundreds of test cases of "comparable" difficulty over the full test sphere. (A distractor that partially occludes or lies behind an object in one view, is likely to totally or severely occlude it in many others). So in one sense, these are "look ma, no hands" examples, but they do serve to make an important point. Figure 11 shows examples from the 6 object database where the system correctly answered the question "what is this?". In these examples, between 50% and 75% of the features arise from distractors. The system also handles pictures containing two or three known objects. It initially finds one, and if asked "what else is there" will identify the other objects.

Figure 11: Examples of manageable images with adjacent slightly occluding clutter

As mentioned previously, it is hard to quantify performance with hand-generated situations such as those in the above two examples, but performance with images of this "difficulty" seems to be somewhere around 90%. The next section addresses this problem of quantifying performance in the presence of clutter and disruption of segmentation in more detail.

3.3 Prediction of Performance

The preceding experiments indicate that the system performs reasonably well on "what is this" tasks for 3-D objects in general position, in the presence of 50% to 75% distractors, and minor (e.g. 25%) occlusion. Up to 50% occlusion seems manageable if there are few distractors. These are reasonable conditions for a "what is this" system, where the assumption is that some process has passed in an interesting region of the image. There are however, a number of tradeoffs operating. Performance decreases with increasing occlusion, increasing numbers of objects in the database, and increasing number of distractors. It would be nice to be able to predict the performance from some more basic measurements. What follows is an attempt to establish a framework for doing this. It is not a complete answer, but it does provide a common framework for interpreting the various performance effects.

We first generated a number (32) of large images consisting of structured clutter, basically jumbles of large numbers of all sorts of objects, but containing none of the objects in any of the databases. Examples of these "pure clutter" images are shown in Figure 12. We then extracted independent subimages of various sizes (ranging from 64x64 to 256x256 pixels in steps of 32, 128 images for each size, 896 total). These images were then ordered by the number of strong contour features found in each (roughly from 1 to 150), and the recognizer run, (using a particular database) on each image, asking "what is this". Of course, no recognizable object is in the images, but certain hypotheses receive some evidence. The scores for the best (false) match in each image were recorded, and statistics gathered for the distribution of scores with respect to the number of clutter features.

Figure 12: Examples of pure clutter images.

We now have (for a particular database) a distribution describing the highest score from scenes containing clutter in which no database object is present. Table 5 shows relevant statistics for the 6 and 24 object databases respectively. Total cases do not quite add to 896 because a few (8) of the smallest images did not generate any hypotheses at all. The values of interest are the mean and standard deviations of the best false match score. As expected, the mean score increases with the complexity of the image, but at a very slow rate, once a certain level is reached; for our data, the mean scores appear to asymptote. This behavior is consistent with a model where the mean scores increase rapidly until the cluttered image is complicated enough to give all the features in an object model a fair chance at matching some image feature. After that, the behavior is dominated by the expected value of the highest outlier in a sample from a bell-shaped, and probably approximately normal distribution,

which grows slowly with sample size. In support of this, we note that the "leveling off" of the curve occurs between 20 and 40 features, and that the number of features in a typical database model is about 20. We also note that the mean increases with the size of the database, though again, quite slowly, governed by the same outlier generation process.

The distribution of the highest scores about the mean appears to be fairly well described by a Gaussian, at least out to a couple of standard deviations. This is expected, since the value of the highest score is due to the sum of a large number of mostly independent random variables whose distributions and relative weights are fairly well behaved. Such a sum (by consequences of the central limit theorem) tends to produce a Gaussian distribution.

Bin range	Number of samples in range	Avg. number of features in bin	Mean score 6 obj.	Standard deviation 6 obj.	Mean score 24 obj.	Standard deviation 24 obj.
0-5	60	3.9	6.9	1.8	7.8	2.1
6-10	96	7.3	8.0	2.2	9.6	2.9
11-20	147	15.3	11.2	2.8	13.5	3.2
21-40	210	29.7	13.1	2.8	15.8	3.5
41-80	270	57.8	14.3	2.7	16.8	3.3
81-160	85	101.2	14.3	2.7	16.3	2.8

Table 5: Score statistics for clutter images of different complexity for 6 and 24 object databases.

From the test images, we can also obtain distributions describing the "correct" match scores when a recognizable object (and no other features) is present. We obtained statistics on these distributions both for the individual object classes (multiple test images for each object) and for the lumped samples in each database. For the same reasons as in the case of the pure clutter images, we expect the distribution for the lumped samples to be approximately Gaussian. Table 6 shows the statistics for the lumped samples for the different databases.

Database	Avg number of features in image	Mean score	Standard deviation
6-object	22.4	31.6	9.2
12-object	21.1	31.2	9.4
18-object	23.9	31.6	9.1
24-object	24.8	31.7	8.7

Table 6: Score statistics for test images for different databases.

A classification error with a database object present occurs when a score due to random clutter exceeds the score of the object. Let $f(s)$ be the probability density function describing the clutter scores and $g(s)$ be the pdf describing the "correct" scores for a given situation. The probability P_e that the clutter score will exceed the "correct" score thus producing a classification error is

$$P_e = \int_{-\infty}^{+\infty} f(s)G(s)ds$$

where $G(s)$ is the cumulative density function corresponding to $g(s)$. Thus from the distributions it is possible to estimate the probability of misclassification due to the presence of a given amount of clutter. We will do this presently.

These distributions can also be used to estimate the probability of misclassification for the test images for the various databases, thus providing a cross-check of the procedure. To do this, we note that as far as the portion of the database that does not represent a particular view of an object is concerned, a test image is just clutter. Thus we can estimate the probability of misclassification by using the statistics for clutter corresponding to the average number of features in a test image.

We did this for the different databases, using Gaussian approximations of the appropriate mean and standard deviation for the clutter distribution and the lumped "correct" score distribution. The statistics for expected clutter (false) scores corresponding to the average number of features for the test images were estimated by interpolating between data points in the clutter experiments (previously presented in Table 5). These results are tabulated in Table 7. Overall, the agreement is reasonably close, though the predicted error rates are a little higher than the actual ones. The worst disagreement is for the 6 object database, where the estimated error rate is 2.2% compared with an observed rate of .4%. This is still reasonably good agreement considering that the .4% measured rate corresponds to a single misclassification in the 200+ test cases and hence a confidence bound of a couple of percent.

Database	Avg. # features in image	Mean correct score	Std. correct score	Mean clutter score	Std. clutter score	pred. error rate	obsrv. error rate
6-object	22.4	31.6	9.2	12.2	2.8	2.2%	.4%
12-object	21.1	31.2	9.4	12.8	2.8	3.0%	1.3%
18-object	23.9	31.6	9.1	14.7	3.3	4.0%	2.6%
24-object	24.8	31.7	8.7	15.0	3.4	4.3%	3.0%

Table 7: "Correct" score statistics, estimated clutter ("false") score statistics, and predicted and observed error rates for test image sets using different databases. The first four columns are the same as Table 6, and duplicated here for reference.

We can also estimate the error for individual classes with the qualification that, since the individual distributions seem to be Gaussian only out to about 2 standard deviations, any misclassification estimates below a couple of percent do not bound the expected misclassification rate below the same couple of percent. If we do this, we find that the objects with the highest estimated misclassification rates are the wolf (16% estimated, 16% observed) and the horse (8.6% estimated, 20% observed), which were the worst objects observationally as well. Note that with only 24 samples, the 90% confidence bound for these observations is on the order of 10 percentage points. Agreement between predicted and observed misclassification rates for the rest of the objects is generally within the 90% confidence bounds of the measurements.

With the general approach validated, at least somewhat, by the above cross checks, we can use the same approach to estimate the overall performance in the presence of varying amounts of clutter. If we initially assume that the clutter only adds features, without

destroying features of the object of interest, then we can estimate expected error rates for total number of features (object plus clutter) equal to the various feature count averages in the pure clutter experiments.

Table 8 shows such predictions based on the statistics obtained from the more complex clutter images. The percent clutter represents the approximate percentage of image features that are not due to an object of interest. The most notable aspect of these results is that, because we are in the asymptotic range noted previously, the predicted error rate does not change significantly with increasing amounts of clutter over the range our experiments cover. The numbers are consistent with the dark-field clutter experiments, where approximately 4% misclassification was observed for the 6 object database with 50% to 75% clutter features. Resistance to pure added clutter thus appears to be very good.

Avg number of features in image	Percent clutter	Predicted error rate 6 obj db	Predicted error rate 24 obj db
(21-24)	0%	2.2%	4.3%
29.7	$\approx$ 25%	2.7%	4.5%
57.8	$\approx$ 50%	3.6%	5.5%
101.2	$\approx$ 75%	3.5%	4.7%

Table 8: Predicted error rates for varying amounts of added clutter for the 6 and 24 object databases. The first line is the predicted error rate (presented previously) for clean images.

A much more serious problem is that adjacent or occluding clutter tends to disrupt the segmentation process that produces the features. This reduces the mean expected score for object-present cases, and increases the error rate. We can model this effect within our framework by computing the error rates for situations where the number of matched features, and hence the mean score, has been reduced by a given factor. Since the score is due to fewer matched features, we must reduce the standard deviation by the square root of the reduction factor.

Table 9 shows the results of these computations for the 6 and 24 object databases using the worst-case false-match statistics. This time, the effect is quite dramatic. 20% disruption brings the expected error rate to 10% to 15%, and 50% disruption brings the rate close to 50%, more or less independent of the actual number of clutter features. The values in parentheses represent error rates higher than what would be obtained from random guessing, and thus are not really meaningful.

Our baseline (0% disruption) figures already incorporate an approximately 40% feature miss rate caused by factors such as viewpoint variation and ordinary lighting effects (highlights and self shadowing). This noted, the table allows us to quantify the effect of disruptive clutter and occlusion. The examples we ran with adjacent and overlapping clutter typically had about 25% of the boundary features disrupted, which would produce an error rate 13% by the above table - in line with our empirical estimate of around 10%. The isolated examples with 50% occlusion worked better than the above table predicts because the number of features is below the asymptotic (high clutter) range for which the above table was derived.

To summarize this section, we have established a statistical protocol that has allowed us to validate the original claims based on wholly empirical experiments. Our protocol predicts

Feature disruption rate	Predicted error rate 6 obj db	Predicted error rate 24 obj db
0%	3.6%	5.5%
5%	4.7%	7.3%
10%	6.1%	9.6%
20%	10%	16%
30%	17%	25%
40%	27%	38%
50%	42%	56%
60%	60%	74%
70%	80%	89%
80%	(95%)	(98%)
90%	(99+%)	(99+%)

Table 9: Predicted classification error rates for varying amounts of feature disruption in the presence of clutter for the 6 and 24 object databases.

performance reasonably well from measurements of system responses to "pure clutter" and clean images, and allows both overall performance, and performance on individual objects to be forecast.

An important result is that the primary sensitivity of the technique is not to extraneous structure, to which it seems almost immune, but to disruption of the segmentation process. Efforts to improve system performance should thus focus on this phenomenon. Improving segmentation is the most direct approach, but possibly not the most effective. Another is to include additional, independent sources of evidence, thus increasing the expected "correct" score. A third approach is to include some higher level verification, that goes back to look for "missed" features.

3.4 Experiments on "Generic" Recognition

This set of experiments was suggested when, on a whim, we tried showing our coffee mugs to an early version of the system that had been trained on the creamer cup in the previous database (among other objects), and noticed that even though the creamer is not a very typical mug, the system was making the "correct" generic call a significant percentage of the time. Moreover, the features that were keying the classification were the "right" ones, i.e., boundaries derived from the handle, and the circular sections, even though there was no explicit part model of a cup in the system.

The notion of generic visual classes is ill defined scientifically. What we have is human subjective impressions that certain objects look alike, and belong in the same group (e.g. airplanes, sports cars, spiders, teapots etc.) Unfortunately, human visual classes tend to be confounded with functional classes, and biased by experience and other factors to an extent that makes it difficult to formalize such classes, even phenomenologically. On the other hand, the subjective intuition is so strong, and the early evidence of correct "generalization" so intriguing, that the matter seemed worth looking into.

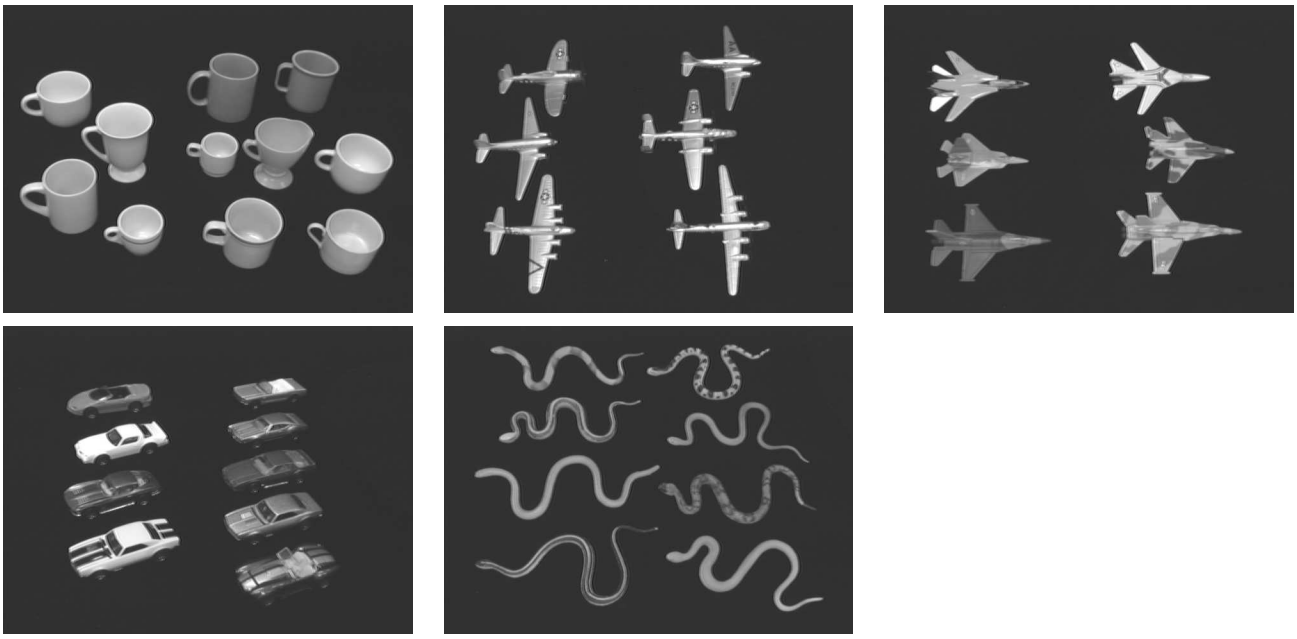

Figure 13: Test sets used in generic recognition experiment. The training objects are on the left side of each image (4 cups, 3 planes, 3 fighters, 4 cars, 4 snakes) and the test objects are on the right.

We gathered multiple examples of objects from several classes, which an (informal) sample of human volunteers agreed looked pretty much alike (our rough criterion was that you could tell at a glance what class an object was in, but had to take a "second look" to determine which member of the class it was). The final database consisted of five classes consisting of 11 cups, 6 "normal" airplanes, 6 fighter jets, 9 sports cars, and 8 snakes.

The recognition system was trained on a subset of each class, and tested on the remaining elements. The training sets consisted of 4 cups, 3 airplanes, 3 jet fighters, 4 sports cars, and 4 snakes. These classes are shown in Figure 13, with the training objects on the left of each picture, and the test objects on the right. The training and test views were taken according to the same protocol as in the previous experiment. The cups, planes, and fighter jets were sampled over the full sphere; the cars and snakes over the top hemisphere (the bottom sides were not realistically sculpted). Overall performance on forced choice classification for 792 test images was 737 correct, or 93.0%. If we average performance for each group so that the fact that the best group, the cups, is not weighted more because it contains more samples, we get 92% (91.96%) performance. The error matrix is shown in Figure 10.

The performance is best for the cups at about 98%, and the planes, sports cars and snakes came in around 92%-94%. The fighter planes were the worst by a significant factor, at about 83%. The reason seems to be that there is quite a bit of difference between the exemplars in some views in terms of armament carried, which tends to break up some of the lines in a way the current boundary finder does not handle. Two of the test cases also have camouflage patterns painted on them. We expect that a few more training cases would improve performance. The performance with snakes was surprisingly high, given the degree of flexibility, and the fact that none of the curves are actually the same (this is supposedly

class name	index	smpls	0	1	2	3	4
cup	0	288	282	0	6	0	0
fighter	1	144	0	120	7	16	1
snake	2	96	5	0	88	1	2
plane	3	144	0	2	7	135	0
sports car	4	120	1	0	6	1	112
Total hypotheses for class			288	122	114	153	115

Table 10: Error matrix for generic classification experiment. Each row shows how the test images for a particular object class were classified.

a rigid object recognition system). The key seems to be the generic "S" shape, which recurs in various ways in all the exemplars, and is quite rare in general scenes.

These results do not say anything conclusive about the nature of "generic" recognition, but they do suggest a route by which generic capability could arise in an appearance based system that was initially targeted at recognizing specific objects, but needed enough flexibility to be able to deal with inter-pose variability and environmental lighting effects. They also suggest that one way of viewing generic classes is that they correspond to clusters in a (relatively) spatially uniform metric space defined by a general, context-free, classification process. This is in contrast to distinctions, such as those needed to tell a cow from a bull, an F16 from an F18, or distinguish faces, that, though they may become fast and automatic in people, involve focusing attention on specific small areas, and assigning disproportionate weight to differences in those regions.

It is our experience that, for appearance-based systems, it is not possible to construct a spatially uniform metric that will match slightly different views of a 3-D object with each other (e.g. 10 degrees out-of-plane rotation), while simultaneously distinguishing objects such as those mentioned above. Some prior information about the identity of the object is necessary in order to know where to look to make fine distinctions, and what distinctions to make. A generic classification based on the fact that certain groups of specific objects are naturally lumped together with a spatially uniform metric, could be used to provide the prior information needed to direct attention to significant details.

4 Comparisons to Other Methods

As far as we have been able to ascertain, the above results represent the most accurate reported in the literature for fully (orthographically) invariant recognition of general 3-D shapes tested on large sets of real images. There is some model-based work that seems accurate for shapes describable with planar regions or line segments; however none of these techniques are applicable to the sort of complex, curved shapes that form the majority of our examples. Furthermore, almost all of the papers illustrate the results on just a few examples, without the sort of full-sphere verification we present here.

Of the appearance-based techniques not using color, the best results on large, real image databases have been reported by Murase & Nayar (1993), and Schmid & Mohr (1996) [23; 30]. Both groups present large scale tests on databases of real images. Nayar presents

results using eigenspace techniques for 3-D recognition in databases containing several tens of objects with accuracy comparable to what we report. Since the system is trained only over a single azimuth on the viewing sphere rather than the full sphere as we do, the results should be scaled accordingly. (We expect there is a factor of 5-10 between the number of images required to cover the full sphere as opposed to a circle for Nayar's approach). The eigenspace techniques also require accurate whole-object segmentation, and would fail with several of the problem classes where we demonstrate success; especially the occluded examples, but also the cases with changed background and clutter. On the other hand, the eigenspace techniques are much faster than ours, operating in a fraction of a second, whereas we take several seconds. Huang & Camps (1997) [15] recently modified the eigenspace approach to use regions extracted using a minimum description length (MDL) segmentation algorithm, and demonstrated some robustness to clutter and occlusion in find-object tasks, again using just a single azimuth circle. A drawback is a requirement that the object be robustly segmentable into uniform-colored regions, which would fail for some of our examples, particularly the animals.

Mohr's methods are based on differential invariants, and exhibit good tolerance for clutter and occlusion. The group shows results for 3-D recognition and obtains good results for a few tens of objects, again training over a circle rather than the full sphere (using Nayar's database in fact). The method is also tested with a database of over 1000 2-D images, which is the same order of magnitude as the number of aspects we use in the 24 object database (1802 aspects). The drawbacks of this method are that it does not handle geometric scaling gracefully, and since the features are differential invariants of the gray-scale image, it is somewhat sensitive to dramatic lighting and contrast changes. Our method is less sensitive in this respect, and handles geometric scaling implicitly. On the other hand, Mohr's techniques probably perform better in the presence of clutter and occlusion since they work with many more, and much smaller features than we do.

A third approach that has considerable similarity to ours is that of Chen and Stockman (1996) [6]. This method uses 2-D invariants of silhouette contour features to index a local (automatically derived) 3-D model. The method is tested on databases containing approximately 600 aspects. Both straight voting and Bayesian evidence combination are considered. However, because only the contour invariants are used for indexing, rather than entire local contexts as we use, the indexing process is comparatively weak, with the expected rank of the correct hypothesis ranging between 16 and 27 depending on the evidence combination scheme, compared with our system, where the expected rank is very close to 1. Consequently, much of the power of their technique derives from the 3-D verification step.

5 Conclusions and Future Work

We have described a framework for keyed appearance-based 3-D recognition, which avoids some of the problems of previous appearance-based schemes. We ran various large-scale performance tests and found good performance for full-sphere/hemisphere recognition of up to 24 complex, curved objects, robustness against clutter, and some intriguing generic recognition behavior. We also established a protocol that permitted performance in the presence of quantifiable amounts of clutter and occlusion to be predicted on the basis of

simple score statistics derived from clean test images and pure clutter images.

In the future, we would like to produce more difficult test databases, both with more objects, and with disruptive clutter. This would permit us to better observe the functional form of the error dependence on number of objects, and provide a basis for testing improvement of the algorithm, e.g. by adding more powerful perceptual grouping processes. It would also be interesting to see how the performance can be improved by adding a final verification stage, since we have observed that even when the system provides the wrong answer, the "right" one is generally in the top few hypotheses. In another direction, we think it would be interesting to look more closely at the nature of "generic" recognition, particularly as regards the idea of common and distinguishing features. At the moment, the system treats all features equally. It might be profitable to explore methods of enhancing the value of matchable features that tend to recur within classes, while dropping, or labeling as distinguishing ones that do not. More generally it would be interesting to adapt the system to allow fine discrimination of similar objects (same generic class) using directed processing driven by the generic classification.

Studies of end-to end systems such as ours are also useful in that they can shed light on efforts to understand biological vision processes. An attempt to actually implement systems that have a relationship to biological models is valuable not only as a direct analog of a theory, but as a reality check on the primitives involved. People are so comfortable manipulating high-level primitives, that it is often not apparent, until an attempt is made to ground a system in hardware, how much of the system complexity lies in the primitives. The history of both the psychological and computational study of vision is littered with models whose primitives turned out to be unexpectedly difficult to define or implement. Well analyzed performance studies of complete, implemented, computational systems thus have a crucial roll to play in attempting to understand the process of vision.

References

[1] Nicholas Ayache and Olivier Faugeras. Hyper: a new approach for the recognition and positioning of two-dimensional objects. *IEEE Trans. PAMI*, 8(1):44–54, January 1986.

[2] Aaron F. Bobick and Robert C. Bolles. Representation space: An approach to the integration of visual information. In *Proc. CVPR*, pages 492–499, San Diego CA, June 1989.

[3] Robert 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.

[4] R. Brunelli and Thomaso Poggio. Face recognition: Features versus templates. *IEEE Trans. PAMI*, 15(10):1042–1062, 1993.

[5] H. H. Bulthoff, S. Y. Edelman, and M. J. Tarr. How are three-dimensional objects represented in the brain? *Cerebral Cortex*, 5(3):247–260, 1995.

[6] Jin-Long Chen and George C. Stockman. Indexing to 3d model aspects using 2d contour features. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR96*, pages 913–920, San Francisco CA, June 1996.

[7] S. Edelman and H. H Bulthoff. Modeling human visual object recognition. In *International Joint Conference on Neural Networks (IJCNN92)*, pages 37–42, Baltimore, MD, June 7-11 1992.

[8] S. Edelman and D. Weinshall. A self-organizing multiple-view representation of 3d objects. *Biological Cybernetics*, 64:209–219, 1991.

[9] F.Stein and Gerard Medioni. Efficient 2-dimensional object recgnition. In *Proc. ICPR*, pages 13–17, Atlantic City NJ, June 1990.

[10] W. E. L Grimson. *Object Recognition by Computer: The role of geometric constraints.* The MIT Press, Cambridge, 1990.

[11] W. E. L. Grimson and Danial P. Huttenlocher. On the sensitivity of the hough transform for object recognition. *IEEE PAMI*, 12(3):255–274, 1990.

[12] W. E. L. Grimson and Daniel P. Huttenlocher. On the sensitivity of geometric hashing. In *3rd International Conference on Computer Vision*, pages 334–338, 1990.

[13] C. G. Gross. Representation of visual stimuli in the inferior temporal cortex. *Philosophical Transaction of the Royal Society of London B*, 335:3–10, 1992.

[14] P. Havalder, G. Medioni, and F. Stein. Percetual grouping for generic recognition. *Internation Journal of Computer Vision*, 20(1-2):59–80, October 1996.

[15] Chien-Yuan Huang and Octavia I. Camps. Object recognition using appearance-based parts and relations. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR97)*, pages 877–883, San Juan, Puerto Rico, June 1997.

[16] Daniel P. Huttenlocher and Liana M. Loriga. Recognizing three-dimensional objects by comparing two-dimensional images. In *IEEE Conferens on Computer Vision and Pattern Recognition (CVPR96)*, pages 878–884, San Francisco, CA, June 1996.

[17] Daniel P. Huttenlocher and Shimon Ullman. Recognizing solid objects by alignment with an image. *International Journal of Computer Vision*, 5(2):195–212, 1990.

[18] M. Kubovy. Gestalt laws of grouping revisited and quantified. In *Proc. SPIE Conference on Human Vision and Electronic Imaging*, pages 402–408, San Jose, CA, February 1997.

[19] 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.

[20] David G. Lowe. *Perceptual Organization and Visual Recognition.* Kluwer Academic Publishers, Boston, MA, 1986.

[21] David G. Lowe. Three-dimensional object recognition from single two-dimensional images. *Artificial Intelligence*, 31:355–395, 1987.

[22] Bartlett Mel. Object classification with high-dimensional vectors. In *Proc. Telluride Workshop on Neuromorphic Engineering*, Telluride CO, July 1994.

[23] Hiroshi Murase and Shree K. Nayar. Learning and recognition of 3d objects from appearance. In *Proc. IEEE Workshop on Qualitative Vision*, pages 39–50, 1993.

[24] Randal. C. Nelson. Finding line segments by stick growing. *IEEE Trans PAMI*, 16(5):519–523, May 1994.

[25] M. W. Oram and D. I. Perret. Modeling visual recognition from neurobiological constraints. *Neural Networks*, 7(6-7):945–972, 1994.

[26] D. I. Perret and M. W. Oram. Neurophysiology of shape processing. *Image and Vision Computing*, 11(6):317–333, July-August 1993.

[27] Thomaso Poggio and Shimon Edelman. A network that learns to recognize three-dimensional objects. *Nature*, 343:263–266, 1990.

[28] Rajesh P.N. Rao. Top-down gaze targeting for space-variant active vision. In *Proc. ARPA Image Understanding Workshop*, pages 1049–1058, Monterey CA, November 1994.

[29] R. Kjeldsen Ruud M. Bolle and Daniel Sabbah. Primitive shape extraction from range data. In *Proc. IEEE Workshop on Computer Vision*, pages 324–326, Miami FL, Nov-Dec 1989.

[30] C. Schmid and R. Mohr. Combining greyvalue invariants with local constraints for object recognition. In *Proc. CVPR96*, pages 872–877, San Francisco CA, June 1996.

[31] R. N. Shepard and L. A. Cooper. *Mental Images and Their Transformations.* MIT Press, Cambridge, MA, 1982.

[32] F. Solina and Ruzena Bajcsy. Recovery of parameteric models from range images. *IEEE Trans. PAMI*, 12:131–147, February 1990.

[33] M. J. Tarr and S. Pinker. Mental rotation and orientation-dependence in shape recognition. *Cognitive Psychology*, 21:233–282, 1989.

[34] Shimon Ullman and R. Basri. Recognition by linear combinations of models. *IEEE Trans. PAMI*, 13(10), 1991.