

# Learning 3D Recognition Models for General Objects from Unlabeled Imagery: An Experiment in Intelligent Brute Force

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

## 1 Introduction

### 1.1 The Promise of Intelligent Brute Force

The artifact that supports human intelligence consists of some  $10^{10}$  to  $10^{11}$  neurons, each with  $10^3$  to  $10^4$  (variable) input connections to other neurons, and each evaluating some function of these inputs on the order of  $10^3$  times per second. Starting with these numbers, estimates can be made of the amount of conventional computational resources required to emulate the functionality of this artifact. Such estimates vary widely, depending among other factors, on the degree of representational and computational redundancy assumed to exist, but range from  $10^{12}$  to  $10^{15}$  bytes for storage, and from  $10^{12}$  to  $10^{18}$  or more operations per second for computation.

By year 2000 computational standards, such requirements are extremely high. A top-of-the-line workstation has about  $10^9$  OPS/Bytes, and is thus 3 to 9 orders of magnitudes removed from emulating brain function. The largest existing computer systems currently possess computational power of a few teraflops, and core memories of a few terabytes (tera is  $10^{12}$ ), still small with respect to most of the above numbers.

Neural tissue is metabolically expensive, and given that efficient energy utilization is strongly selected for, it is plausible to consider the possibility that tera- to peta- ops/byte levels of computational resources are a necessary (though not sufficient) condition for obtaining human-like (or even animal-like) "intelligence". If this is true, past (and current) attempts to emulate significant aspects of human intelligence (including vision) on a workstation seem a bit ludicrous. This might explain why machine vision has seen such limited success compared to human abilities.

Tera-op/second machines, currently represent multi-million dollar investments, but it is not unlikely that comparable resources will be readily available at

the department if not the desktop level in a decade. It is interesting to note, that for the first time in history, the power of existing computers overlaps (though just barely) some estimates of the computational resources employed by mammalian brains. An interesting thought experiment is to suppose that we, as vision or AI researchers, were handed tera-op capacity in an easy-to-use form. What could we do with it?

One interesting question concerns the existence of "threshold" problems in vision/AI. Are there "easy" techniques for solving certain vision/AI problems that require resources at the tera-op/byte level to work at all, but work well if the resources are available (e.g. analogs of Z-buffer algorithm in graphics). Can this be proven without an actual demonstration? Are there algorithms whose scalability cannot be proven theoretically, but would be worth investigating in a full-scale test were tera-op resources available?

A related issue is the existence of what might be termed "intelligent brute-force" algorithms. There are a number of problems in vision (and elsewhere) for which brute-force algorithms of low complexity are trivially known to exist, but with enormous constant factors. For example, the object recognition in clutter problem is simple to solve in theory, just by comparing every possible view of an object, over appropriate discretized pose and illumination spaces, with the image. This requires resources that are linear in the number of objects, but with a constant factor that puts it above even the highest estimates of human brain capacity. However, there are a number of relatively simple techniques that can be used to reduce the constant factor. No methods have been found that achieve human-level performance with 80s, 90s, or year 2000 workstation level resources, but there are some indications that very interesting behavior can be obtained at tera-op level.

We propose that it is worth looking at intelligent

brute force methods for machine vision with an eye towards figuring out what can be done with tera- to peta- ops/byte resources. In this paper, we work with an object recognition system that uses an intelligent, brute-force approach, and achieves results for the problem of general 3D object recognition in clutter that are as good or better than anything demonstrated to date. It does not scale to human-level performance, or use peta-op resources (yet). However, the performance of this system, and of a few other recently reported resource intensive systems, is good enough to allow us to begin investigating other aspects of human visual intelligence. In particular, we look at the problem of acquiring visual recognition models from unlabeled imagery. Some of our experiments, making extensive use of an already resource intensive recognition system, do utilize on the order of  $10^{15}$  operations (a peta-op).

## 1.2 Training 3D Object Recognition Systems from Imagery

A number of current 3D object recognition methods rely on training from imagery. This allows the recognition of objects without the requirement of constructing a 3D geometric model, and the recognition of objects for which current geometric recognition technologies do not apply. In various forms, image-based methods are currently the most successful general approach. Several of these systems can handle 5 or 6 continuous dimensions, usually orthographic freedoms, for individual objects. Some are also able to handle generic, visually similar, classes with moderate variation, for example airplanes, cars, or faces. Typically, a system is trained with a fully labeled (identity and pose) set of clean (segmentable) example views. The view set generally covers variation due to out-of plane rotation, and sometimes geometric distortion among class exemplars.

One of the first appearance-based systems was the one developed by Poggio that recognized wire objects [13]. Rao and Ballard [14] describe an approach based on the memorization of the responses of a set of steerable filters. Mel's SEEMORE system [9] uses a database of stored feature channels representing multiple low-level cues describing contour shape, color and texture. Schiele and Crowley [15] used histograms of the responses of a vector of local linear neighborhood operators. Murase and Nayar [10] find the major principal components of an image dataset, and use the projections of unknown images onto these as indices into a recognition memory. This approach was extended by Huang and Camps [8] to appearance-based parts and relationships among them. Wang and Ben-

Arie [18] were able to do generic object detection using vectorial eigenspaces derived from a small set of model shapes which are affine transformed in a wide parameter range. The approach taken by Schmid and Mohr [16] is based on the combination of differential invariants computed at keypoints with a robust voting algorithm and semilocal constraints. Nelson and Selinger [11], have developed a "Cubist" approach based on groupings of robustly keyed local contexts or fragments. This method will be more fully described later.

Two issues that have been little explored are (1) whether such systems can be trained from imagery that is unlabeled, and (2) whether they can be trained from imagery that is not trivially segmentable. A recognition system that could be trained from either unlabeled or unsegmented imagery would be valuable for reducing the effort required to obtain a training set. Of greater practical impact, a 3D recognition system that could be trained from cluttered imagery would be useful for automatic, object-level labeling of image databases, which is an important outstanding problem.

In this paper we explore the problem of training a general, 3D object recognition system from unlabeled imagery. In particular, we attempt to identify critical issues and stumbling blocks associated minimizing the supervision necessary to train such a system. Because class learning seems to be a relatively slow and resource intensive process even for people, we consider approaches and perform experiments that entail on the order of  $10^{15}$  (one quadrillion) basic operations, even for relatively small databases. This is the current practical limit of the computation we can get our hands on. For experiments, we use a recognition system developed previously [12],

Our initial exploration centers on the use of simple clustering algorithms exploiting the property that our recognition system, if trained on a few views of an object, is able to correctly recognize views that are topologically close to the original ones. By iteratively adding such "close" views to an object representation we attempt to produce clusters of views characterizing each object. We look at the way these clusters develop and at how well they cover the view set. We investigate how the performance of such a learning system can be improved by strategies such as extending the set of initial views, increasing the sampling rate, introducing a small amount of user supervision, or by devising more complicated learning algorithms.

## 2 Previous Work in Unsupervised Training of 3D Recognizers

A system that is trained from unlabeled images has to be able to perform unsupervised clustering of multiple views into multiple object classes. There are several things that make this kind of clustering difficult. The object image can change significantly as the viewpoint changes. In the meantime, some views of different objects are more similar to each other than views of the same object.

In one approach, Ando *et al.* [1] observed that although the input dimension of such an image set is very high, the view data of an object often resides in a low-dimensional subspace. In addition, the view distribution of an object is inherently continuous, i.e. a chain of multiple views constitutes a continuous data manifold. Therefore their strategy is to identify multiple non-linear subspaces each of which contains the views of each object class. This is done by iterating the computation of the distances between each view data and the estimated subspaces, and the re-estimation of the subspaces using these distances. The unsupervised algorithm uses a combination of autoencoders, where each autoencoder network is considered as a module that discovers a non-linear subspace of each object class. The system was tested on synthetic wireframe objects, as well as on a few gray-level images real objects. Good performance requires a very high density of views (one view at every degree).

A similar approach was taken by Basri *et al.* [2]. Their method examines the space of all images and partitions the images into sets that form smooth and parallel surfaces in this space. Nearby images are grouped into surface patches that form the nodes of a graph. Further grouping becomes a standard graph clustering problem. Again, good results are obtained only if a very large number of images, or even sequences of images are considered.

The dependence of the performance on the sampling density of the viewing sphere is not surprising. If the number of images is high, clustering becomes the effect of a phase transition phenomenon: when the parameters of the image set reach a certain value, the topology of the network suddenly changes from small isolated clusters to a giant one containing very many nodes.

This phenomenon was first displayed in the seminal work of Erdős and Rényi [4] on random graphs, who showed that, for many properties  $Q$ , the limit probability that a random graph has property  $Q$  exhibits a sharp increase at some critical value of the edge probability. Hogg and Kephart [7] applied this

theory for the case of information-intensive categorization tasks. They showed that the performance of algorithms used in solving such problems changes abruptly as the number of categorization classes, the variability of the input features, and the algorithm quality are varied. Under the proper conditions, a classification algorithm which performs poorly in small problems can perform extremely well for larger problems.

The unsupervised learning methods discussed above were based on computing distances between images. Basri *et al.* [2] use a similarity measure based on the distortion of salient features between images. Gdalyahu and Weinshall [5] use a curve dissimilarity measure. The disadvantage of such similarity measures is that they generally require full object segmentation and cannot deal with scale changes. A system that combines object recognition with an unsupervised learning method would be robust to clutter and changes of scale.

## 3 A Cubist Object Recognition System

In the following experiments, we use a recognition system utilizing what we have termed a “Cubist” approach because of the similarity of the top-level representation to some cubist artwork [11]. The system is based on a hierarchy of perceptual grouping processes [17]. A 3D object is a fourth-level group consisting of a topologically structured set of flexible 2D views, each derived from a training image. In these views, which represent third-level perceptual groups, the visual appearance of an object is represented as a geometrically consistent cluster of several overlapping local context regions. These local context regions represent high-information second-level perceptual groups, and are essentially windows centered on and normalized by key first-level features that contain a representation of all first-level features that intersect the window. The first level features are the result of first level grouping processes run on the image, typically representing connected contour fragments.

In more detail, distinctive local features called *keys*, selected from the first level groups in our hierarchy, seed and and normalize *keyed context regions*, the second level groups. In the current system, the keys are contours automatically extracted from the image. These context regions, as they contain more information amplify the power of the key features. This is necessary because the invariant parameters of simple key features such as contour fragments 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, especially

when applied to complex, cluttered scenes, or when using features that are subject to partial occlusion (such as edges) [6]. The fact that the high-information units are overlapping fragments of a view gives robustness to clutter and occlusion not present in other image-based methods.

Even these high information local context regions are generally consistent with several object/pose hypotheses; hence we use the third-level grouping process to organize the context patches into globally consistent clusters that represent hypotheses about object identity and pose. This is done through a hypothesis database that maintains a probabilistic estimate of the likelihood of each third level group (cluster) based on statistics about the frequency of matching context patches 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 clusters represent two dimensional rigid transforms of specific views. As mentioned above, the use of keyed contexts rather than first-level groups gives the voting features sufficient power to substantially ameliorate well known problems with false positives in Hough-like voting schemes.

The system was trained on labeled imagery of a variety of 3D shapes, ranging from sports cars and fighter planes to snakes and lizards. These images were taken at about 20 degrees apart over the whole viewing sphere. When trained on 24 objects the system achieved a recognition performance of 97% on images that were taken in between the training views, under the same good conditions [11]. Performance remains relatively good in the case of clutter and partial occlusion [12]. The system also displays an ability to generalize to “similarly” shaped (in human terms), but previously unencountered objects. This has permitted remarkably robust generic recognition of visually similar classes such as cups, cars and airplanes.

The system is resource intensive in the “intelligent brute-force” sense mentioned above. Identifying and locating known objects in a standard 512x512 image with moderate clutter currently involves on the order of  $10^{10}$  (10 billion) operations per object class.

## 4 Training with Limited Supervision

In our baseline experiments in minimally supervised training, we labeled 1 to 4 images per object and used them as seeds for initial classes. The recognition system was then used to extract one or more views from a mixed database of images that could be classified with high confidence using the existing (partial) rep-

resentations. We added these views to the appropriate class representations and iterated the process.

### 4.1 The image corpus

In our initial experiments, we used several image corpora consisting of clean, black background images of 6 objects (the cup, bear, car, rabbit, plane and fighter seen in Figure 1). The smallest corpus represented a uniform sampling of the viewing sphere at about 20 degrees intervals. This produced 106 images per object (53 images per object hemisphere), except for the sports-car where we had only the top hemisphere since the bottom was black and featureless. The total thus consisted of 583 images of the six objects.

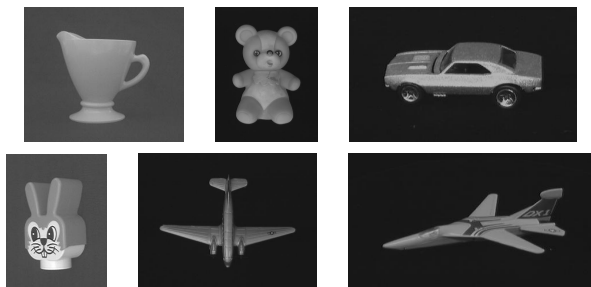


Figure 1: The objects used in testing the system.

In order to investigate the effect of sampling density we also obtained extended image sets, with a denser sampling of the viewing sphere. The first set has a total of 1035 clean, black background images taken approximately 15 degrees apart over the viewing sphere. There are 105 images of the sports-car (one hemisphere) and 186 images of each of the other objects. We also obtained sets with sampling densities down to 5 degrees for the car and fighter objects.

### 4.2 Training on one object

The purpose of this experiment was to establish how well it is possible to do using our recognition algorithm to drive a naive clustering algorithm, and to investigate the effect of differing strategies. Using a corpus of clean images establishes an upper bound on the performance. For this experiment we seeded the recognition system with a single image of one of the object classes (e.g. the sports-car) and then iteratively found the best match to the current representation over the entire corpus and rebuilt the class representation incorporating the new image. We estimated the best possible performance by continuing the process until images not belonging to the initial class were

attracted. The overall procedure is essentially a minimum spanning tree algorithm.

To illustrate, consider the case with the sports car object. Similar behavior was observed for the other objects. Using the database with 20 degree separation, we found that the first 32 images attracted to the growing representation were images of the sports-car. These represented over half of the 52 sports-car images in the entire corpus. Figure 2 shows the tree by which these  $1 + 32 = 33$  views (62.2% of the sports-car views) attracted each other to the growing representation. The 52 sports-car images in the corpus represent one hemisphere, and are represented by circles on the polar coordinate system in the figure. Dark circles represent images attracted to the representation prior to the first false match. The double circle represents the seed. Arrows show the topology of the growth process. The incorporated images represented a coherent patch of the viewing sphere, and spanned over 120 degrees. The attraction process generally operated between close geometric neighbors, with one exception, where an image from the same elevation, but separated by 180 degrees in azimuth was attracted. This is due to the rough front/rear symmetry of the sedan.

Figure 3 shows the seed image and some of the sports-car images attracted to the representation (indicated by a,b,c in Figure 2). It also shows the first non-sports-car image attracted at step 33 that stopped the growth process. The image is actually an odd view of a toy rabbit block, but one that interestingly, looks a bit like the sports-car.

The other objects had somewhat similar results: the cup attracted 50.9%, the bear 61.3%, the rabbit 31.1%, the plane 48.1% and the fighter 61.3% of their total number of views. The rabbit’s low percentage is due to its odd shape, that makes it look rectangular from the side; this ring of pathological views stops propagation from one hemisphere to the other. Since the plane and the fighter are somewhat similar in shape, the first incorrect image attracted by the plane class was a fighter, and the first incorrect image attracted by the fighter class was a plane. In both cases the first image attracted from a different class occurred after 61.1% of the 212 plane and fighter images were attracted.

Next we performed the same experiment on the extended image set with views taken at 15 degrees over the viewing sphere (a total of 1035 images involving 6 different objects). expected, this reduced the number of isolated views, and thus the learning performance increased in general, as can be seen in Table 1. The biggest improvement can be seen in the case of the cup

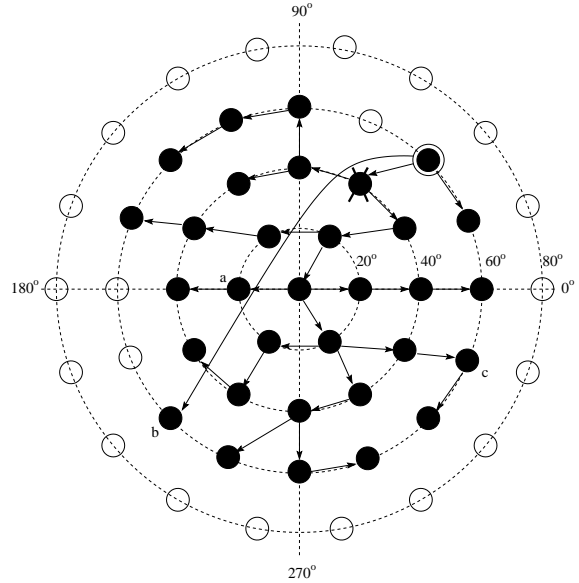


Figure 2: Tree by which training images were attracted to representation during growth process. The seed image is marked with a circle, while the image that attracted the first incorrect image is marked with an X.

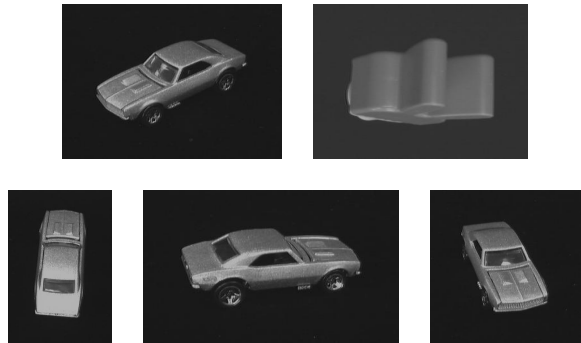


Figure 3: Top: Seed image of sports-car for propagation experiment and terminating non-car image. Bottom: car images, a,b, and c, attracted to the representation during the experiment.

and the bear where increasing the sampling from every 20° to every 15° decreased significantly the number of isolated views. The decrease in the performance for the car is due to the inclusion of a number of views around the “pathological” 90 degree equator that were not present in the 20 degree sampling. As we show below, this aliasing effect vanishes with further increase in the sampling density.

To investigate the limits of the improvement with sampling density, we used a database containing im-

object	performance on initial set	performance on extended set
cup	50.9%	80.1%
bear	61.3%	80.6%
car	62.2%	54.2%
rabbit	31.1%	34.4%
plane	48.11%	54.3%
fighter	61.3%	63.3%

Table 1: Percentage of images attracted by each object before the first incorrect classification

ages of the sports car and fighter that had been sampled every 5 degrees near the equator, where the greatest change between views occurs. When these were mixed with the other images, the nearest neighbor clustering process attracted 370 out of 391, or 94.6% of the car images before attracting a non-car image. For the fighter, 653 out of 710 or 91.9% were attracted. The missed images were all low profile, mostly head-on views of the front and rear of the vehicles, where the shape is indistinct even to human observers for both the car and the fighter. These results suggest that the performance on clean images can be pushed close to 100% by sufficiently increasing the sampling density.

### 4.3 Training on several objects

The next experiment addressed the effect of growing multiple classes simultaneously. The idea was that a view incorrectly attracted to a class in the one-class-at-a-time experiment might be attracted to its correct class first if multiple growth categories existed, thus improving performance. For this experiment we seeded the system with one view of each of 6 objects and iteratively found the best match to some current representation over the entire corpus. The first incorrect classification in the 20 degree sampled database occurred at image 231 out of 583. The performance started degrading after that, reaching an overall performance of 60%, i.e. 60% of the images in the database were correctly classified and thus attracted to the representation of the correct model.

The percentage of each class that was clustered before that cluster attracted an incorrect example ranged from 66% for the fighter plane (better than the one-class result of 61%) to about 19%. One observation we made during these experiments is that objects often produce a few, large connected clusters that do not hook to each other with coarsely sampled databases. This suggests that it might be advantageous to use several different seed images for the same object. To see how the number of seed images affects system performance, we seeded the system with two, four and

eight images per object. We obtained a significant improvement when using two seed images (an overall final performance of 82%). A slight further improvement was obtained using four seed images per object (overall final performance was 86%). Increasing the number of seed images to eight didn't change the overall performance.

Adding additional seeds dramatically improved the percentage of each class that was attracted before the first incorrect match. With two seed images, the performance of the simultaneous clustering algorithm was better than the single-class for all classes except the fighter, which attracted a plane image at 53%. These results are shown in Table 2

Object	1-class	1 seed	2 seeds	4 seeds	8 seeds
cup	50.9%	63.2%	66.0%	66.9%	81.1%
bear	61.3%	19.8%	68.8%	66.9%	70.7%
car	62.2%	60.3%	77.3%	81.1%	77.3%
rabbit	31.1%	18.8%	81.1%	82.0%	82.0%
plane	48.1%	47.1%	49.0%	50.9%	54.7%
fighter	61.3%	66.0%	52.8%	55.6%	61.3%

Table 2: Percentage of images attracted by each object before the first incorrect classification, comparing performance of the one-class-at-a-time approach to multiple-class method using different numbers of seeds.

The jump in performance for two seeds is probably due to the fact that flattened objects have two large view clusters separated by an equator of "pathological" views, and by using two seed images we were able to attract both clusters. This phenomenon suggests the need for variable sampling of the viewing sphere, with more images in the areas with difficult, or rapidly changing views.

Figure 4 shows the change in overall clustering accuracy as images are added to the system for different numbers of seeds. again, the jump in performance at two seeds is evident.

### 4.4 Interactive factors

Better performance can be achieved by an interactive system that asks an expert (e.g. a human user) about the identity of images that could not be classified. This provides a way of connecting disjoint clusters corresponding to the same class. The question is, how much human effort is required to achieve a given performance level. To investigate this, we modified the multiple object clustering experiment. Images that get recognition scores above a fixed threshold are added to the object representation automatically. If none of the images obtained a high enough score dur-

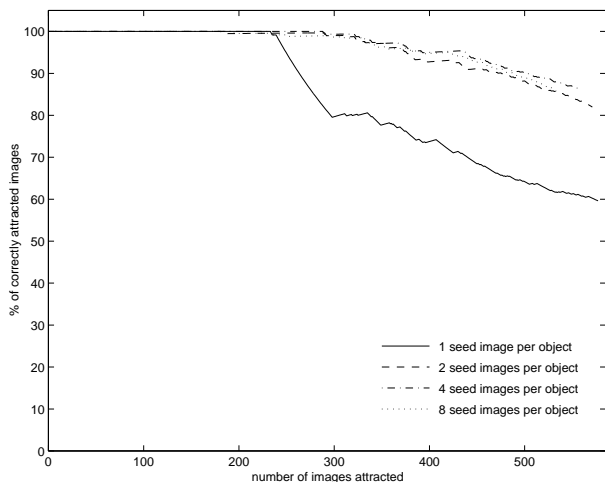


Figure 4: Performance of the learning system with 1, 2, 4 and 8 seed images per object

ing one iteration, the system asks the user about the identity of the object in the image that got the highest score. This is usually the image that has the most information and hence it might be able to attract more images to the representation.

We ran this interactive version on our 20 degree sampled corpus (583 images) using 1 seed image per object, and a variety of thresholds in order to observe how the accuracy of the final classification varied with the number of questions answered by the user. Table 3 shows some data points. To obtain near perfect performance, a relatively large percentage of the database (about 40%) had to be classified by the user. There is also a sharp transition to a dramatically lower mis-clustering rate as the number of user classifications increases. Using the 15 degree sampled corpus, an equivalent low error rate (0.5%) can be obtained with a much lower percentage (20%) and a somewhat lower number (200) of the samples classified by the user. Overall, the behavior is consistent with a model of large easily clustered regions on the viewing sphere surrounded by boundary regions that are not densely sampled enough to be attracted. Again, some process of varying sampling density is indicated.

## 5 Conclusions and Future Work

Our exploratory study suggests that there is considerable potential for training 3-D recognition systems from imagery with minimal supervision. For all the objects we investigated, large, coherent chunks of the viewing sphere - between 50 and 80 percent - could be clustered with one or two seeds, and fairly coarse sampling (e.g. 20 degrees).

questions answered	incorrectly classified
No questions	230 ( 40%)
132 (22.8%)	84 (14.5%)
180 (31.1%)	61 (10.5%)
230 (39.8%)	2 (0.3%)

Table 3: Variation of clustering accuracy with number of questions answered by expert user

The flip side, of course, is that many objects also displayed a set of “pathological” orientations, where appearance, at least as measured by our recognition system, changed rapidly, and clustering was more difficult. Our experiments indicate that, for the most part, this can be overcome by increasing the sampling density, but over some small regions, the required density may be quite high - less than 5 degrees between views. This is higher than is practical to expect from general imagery, on which one might eventually hope to train recognition systems.

One possible solution is to allow interactive access to example objects, which is biologically plausible, but violates the spirit of unsupervised clustering. Another is to ask a human expert for help in classifying difficult views. Our experiments indicate that to completely close the gaps, a fairly high degree of help (compared to a couple of seeds) is required. A third possibility would be to use virtual views, such as those used by Beymer and Poggio for face recognition [3] to generate additional examples from existing images. This requires some information about 3D structure, either explicit or implicit. However, a number of techniques exist that might be able to provide sufficient if approximate, information, including use of shading information, generic shape models, and coarse stereo derived from previous matches.

A somewhat separate issue is the existence of what might be termed “accidentally similar” views - situations where a peculiar view of one object looks like another. This can cause undesired merging of classes. Here a more sophisticated clustering algorithm could be used to advantage; in particular, looking at the breadth of the connectivity between components. Narrow connectivity suggests that two distinct classes are involved.

The complexity of the straight-forward incremental algorithm used in this study is  $O(n^3)$  where  $n$  is the number of images. With a fixed database and appropriate preprocessing, this could probably be brought down to  $O(n^2 \log(n))$ , since it is a spanning tree type problem. For systems where we start discarding images as “containing nothing new”, the effective com-

plexity comes down further, because the size of the overall representation stops growing quickly.

A future goal is to investigate systems that can be trained from cluttered images. The performance of our object recognition system is such that we believe this might ultimately be possible. Probably the most significant additional difficulty is deciding what new structure to introduce into the representation when a match is found in clutter - as any new feature, by definition, is not involved in the match that detected a new view in clutter. One possibility is to use perceptual grouping procedures to associate new features with the features contributing to the match. Such additions might be subject to a "verification" process, where they would need to be detected at least twice in independent situations before being incorporated into the overall representation.

## References

- [1] H. Ando, S. Suzuki, and T. Fujita. Unsupervised visual learning of three-dimensional objects using a modular network architecture. *Neural Networks*, 12:1037–1051, 1999.
- [2] R. Basri, D. Roth, and D. Jacobs. Clustering appearances of 2d objects. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 414–420, Santa Barbara, CA, June 1998.
- [3] D. Beymer and T. Poggio. Face recognition from one example view. In *Proc. International Conference on Computer Vision*, pages 500–507, Cambridge, MA, June 1995.
- [4] P. Erdős and A. Rényi. On the evolution of random graphs. *Publications of the Mathematical Institute of the Hungarian Academy of Science*, 5:17–61, 1960.
- [5] Y. Gdalyahu and D. Weinshall. Automatic hierarchical classification of silhouettes of 3d objects. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 787–793, Santa Barbara, CA, June 1998.
- [6] W. Grimson and D. Huttenlocher. On the sensitivity of the hough transform for object recognition. *IEEE Trans. PAMI*, 12(3):255–274, 1990.
- [7] T. Hogg and J. O. Kephart. Phase transitions in high-dimensional pattern classification. *Computer Systems Science and Engineering*, 5(4):223–232, October 1990.
- [8] C. Huang, O. Camps, and T. Kanungo. Object recognition using appearance-based parts and relations. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 878–884, San Juan, Puerto Rico, June 1997.
- [9] B. Mel. Seemore: Combining color, shape, and texture histogramming in a neurally-inspired approach to visual object recognition. *Neural Computation*, 9:777–804, 1997.
- [10] H. Murase and K. Nayar. Visual learning and recognition of 3-d objects from appearance. *Int. Journal of Computer Vision*, 14(1):5–24, 1995.
- [11] R. Nelson and A. Selinger. A cubist approach to object recognition. In *Proc. International Conference on Computer Vision (ICCV98)*, pages 614–621, Bombay, India, January 1998.
- [12] R. Nelson and A. Selinger. Large-scale tests of a keyed, appearance-based 3-d object recognition system. *Vision Research*, 38(15-16):2469–88, August 1998.
- [13] T. Poggio and S. Edelman. A network that learns to recognize three-dimensional objects. *Nature*, 343(1):263–266, 1990.
- [14] R. Rao and D. Ballard. An active vision architecture based on iconic representations. *Artificial Intelligence*, 78:461–505, 1995.
- [15] B. Schiele and J. Crowley. Object recognition using multidimensional receptive field histograms. In *Proc. Fourth European Conference on Computer Vision*, pages 610–619, 1996.
- [16] C. Schmid and R. Mohr. Combining greyvalue invariants with local constraints for object recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 872–877, San Francisco, CA, June 1996.
- [17] A. Selinger and R. Nelson. A perceptual grouping hierarchy for appearance-based 3d object recognition. *Computer Vision and Image Understanding*, 76(1):83–92, October 1999.
- [18] Z. Wang and J. Ben-Arie. Generic object detection using model based segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 428–433, Fort Collins, CO, June 1999.