MASSACHUSETTS INSTITUTE OF TECHNOLOGY
ARTIFICIAL INTELLIGENCE LABORATORY
and
CENTER FOR BIOLOGICAL INFORMATION PROCESSING
WHITAKER COLLEGE

A.I. Memo No. 1181 April 1990
C.B.I.P Memo No. 52

Bringing the Grandmother back into the picture: a memory-based view of object recognition

Shimon Edelman Tomaso Poggio

Abstract

We describe experiments with a versatile pictorial prototype based learning scheme for 3D object recognition. The GRBF scheme seems to be amenable to realization in biophysical hardware because the only kind of computation it involves can be effectively carried out by combining receptive fields. Furthermore, the scheme is computationally attractive because it brings together the old notion of a "grandmother" cell and the rigorous approximation methods of regularization and splines.

© Massachusetts Institute of Technology (1990)

This report describes research done at the Massachusetts Institute of Technology within the Artificial Intelligence Laboratory and the Center for Biological Information Processing in the Department of Brain and Cognitive Sciences and Whitaker College. The Center's research is sponsored by grant N00014-88-K-0164 from the Office of Naval Research (ONR), Cognitive and Neural Sciences Division; by the Alfred P. Sloan Foundation; and by National Science Foundation grant IRI-8719392. The Artificial Intelligence Laboratory's research is sponsored by the Advanced Research Projects Agency of the Department of Defense under Army contract DACA76-85-C-0010 and in part by ONR contract N00014-85-K-0124. TP is supported by the Uncas and Helen Whitaker Chair at Whitaker College, MIT. SE is supported by a Chaim Weizmann Postdoctoral Fellowship from the Weizmann Institute of Science and by a NSF Presidential Young Investigator Award to Professor Ellen C. Hildreth.

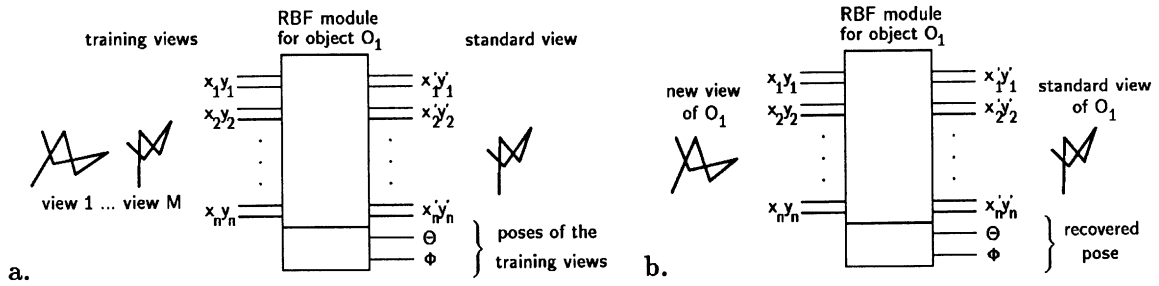


Figure 1: Application of a general module for multivariate function approximation to the problem of recognizing a 3D object from any of its perspective views. In (a), the module is trained to produce the vector representing the standard view of the object, given a set of examples of random perspective views of the same object. The module is also capable of recovering the viewpoint coordinates θ, ϕ (the latitude and the longitude of the observer on an imaginary sphere centered at the object) that correspond to the training views. When given a new random view of the same object (b), the module recognizes it by producing the standard view. Other objects are rejected by thresholding the euclidean distance between the actual output of the model and the standard view.

1 Introduction

An intelligent visual system is expected to be able to retain representations of objects it encounters and to recognize these objects later, under potentially different viewing conditions. This requires the solution of at least three difficult problems. The first problem is the variability of object appearance due to changing illumination, which may be addressed by working with relatively stable features, such as intensity edges [1] (preferably, in conjunction with cues from visual motion and stereo [2]), rather than with raw intensity images. The second problem, the removal of the variability due to unknown pose of the object, may be solved by first hypothesizing the viewpoint (e.g., using information on feature correspondences between the image and a model), then computing the appearance of the model of the object to be recognized from that viewpoint and comparing it with the actual image [3, 4, 5, 6]. Generally, recognition schemes of this type employ 3D models of objects. Automatic learning of 3D models is the third difficult problem faced by state-of-the-art recognition schemes. Few of these schemes learn to recognize objects from examples and most use 3D models acquired through user interaction (see, e.g., [6]) or through active sensing (e.g., range data; [7, 8]).

In this paper, we describe an implemented scheme for recognizing wire-frame objects that addresses two of the three aspects of the recognition problem mentioned above: learning object representations and generalizing recognition to novel viewpoints. We base our approach on a recently proposed network scheme for the approximation of multivariate functions, by coaching the problem in terms of the synthesis of a module that generates a representation of an object (e.g., produces a "standard" view) given any of its perspective views (Figure 1).

2 Theoretical basis

2.1 How much information is necessary for learning 3D structure?

Structure from motion theorems [9, 10], pioneered by Ullman [11], indicate that full information about the 3D structure of an object represented as a set of feature points (at least five to eight) is present in just two of their perspective views, provided that corresponding points are identified in each view. A view is represented as a $2N$ vector $x_1, y_1, x_2, y_2, \ldots, x_N, y_N$ of the coordinates on the image plane of N labeled and visible feature points on the object. Here and in most of the following we assume that all features are visible, as they are in wire-frame objects. The generalization to opaque objects follows by partitioning the viewpoint space for each object into a set of "aspects" [12], corresponding to stable clusters of visible features. In principle, therefore, having enough 2D views of an object is equivalent to having its 3D structure specified.

2.2 Learning as hypersurface interpolation

This line of reasoning, together with properties of perpective projection, suggest (a) that for each object there exists a smooth function mapping any perspective view into a "standard" view of the object and (b) that this multivariate function may be synthesized, or at least approximated, from a small number of views of the object. Such a function would be object specific, with different functions corresponding to different 3D objects. Furthermore, the application of the function that is specific for one object to the views of a different object is expected to result in a "wrong" standard view that can be easily detected as such.

Synthesizing an approximation to a function from a small number of sparse data – the views – can be considered as learning an input-output mapping from a set of examples [13, 14]. A powerful scheme for the approximation of smooth functions has been recently proposed under the name of Generalized Radial Basis Functions (GRBFs) and shown [13, 14] to be equivalent to standard regularization [15, 16] and generalized splines ([13]; see closely related work by Powell [17] and Broomhead and Lowe [18]). The approximation of $f : R^n \rightarrow R$ is given by

$$f(\mathbf{x}) = \sum_{\alpha=1}^{K} c_\alpha G(\|\mathbf{x} - \mathbf{t}_\alpha\|) \tag{1}$$

where the coefficients c_α and the centers $\mathbf{t}_\alpha$ are found during the learning stage and G is an appropriate basis function (see [13, 14]), such as the Gaussian. If the function f is vector-valued, each component f_i is computed using eq. 1 with the appropriate $c_{i\alpha}$, in which case the equation is precisely equivalent to the network of Figure 2. The function f(x) in equation 1 minimizes the error functional

$$H[f] = \sum_{i=1}^{M} (y_i - f(\mathbf{x}_i))^2 + \lambda \|Pf\|^2 \tag{2}$$

on the set of examples. In equation 2, P is usually a differential operator and λ is a positive real number, called the regularization parameter [15]. The radial function G is fully determined by the stabilizer P in eq. 2 and therefore by the prior assumptions on the function to be approximated, such as its degree of smoothness [13]. P also determines whether a polynomial term of the form $\sum_i d_i p_i(\mathbf{x})$ should be added to the right-hand side of eq. 1. In most of the

experiments described in section 3.6 we omitted the polynomial term and used the Gaussian as the radial basis function. The optimal width σ of the Gaussian RBFs can be found, along with c_α and $\mathbf{t}_\alpha$, by minimizing H in equation 2.

In a special simple case, there are as many basis functions (K) as views in the training set (M; in general, $K \leq M$). The centers of the radial functions are then fixed and are identical with the training views. Each basis unit in the "hidden" layer computes the distance of the new view from its center and applies to it the radial function. The resulting value $G(\|\mathbf{x} - \mathbf{t}_\alpha\|)$, can be regarded as the "activity" of the unit. If G is Gaussian, a basis unit will attain maximum activity when the input exactly matches its center. The output of the network is a linear superposition of the activities of all the basis units in the network.

Figure 2*b* illustrates the special case of Gaussian basis functions. A multidimensional Gaussian can be synthesized as the product of two-dimensional Gaussian receptive fields operating on retinotopic maps of features. The solid circles in the image plane represent the 2D Gaussians associated with the first radial basis function, which corresponds to the first view of the object. The dotted circles represent the 2D receptive fields that synthesize the Gaussian radial function associated with another view. The Gaussian receptive fields transduce positions of features, represented implicitly as activity in a retinotopic array, and their product "computes" the radial function without the need of calculating norms and exponentials explicitly.[1]

The weights $\mathbf{C}$ are found during learning by minimizing a measure of the error between the network's prediction and the desired output for each of the examples. Computationally, this amounts to inverting a matrix (when $M \neq K$, the generalized inverse is computed instead) and is equivalent to finding an optimal generalized spline approximation (interpolation when $\lambda = 0$ in equation 2) with fixed knots.

If the centers of the basis functions are allowed to move (which may be desirable, e.g., when the number of basis functions is less than the number of views in the training set), the scheme becomes equivalent to a spline with free knots. The centers may be updated during learning by a gradient descent minimizing the approximation error expressed by equation 2. A further generalization may be achieved by using a weighted norm in equation 1:

$$\|\mathbf{x} - \mathbf{t}_\alpha\|_W^2 = (\mathbf{x} - \mathbf{t}_\alpha)^T W^T W (\mathbf{x} - \mathbf{t}_\alpha) \tag{3}$$

Updating the centers is equivalent to modifying the corresponding "prototypical views" and corresponds to task-dependent clustering. Finding the optimal weights for the norm is equivalent to a transformation of the input coordinate space and corresponds to task-dependent dimensionality reduction. A more detailed description of the GRBF approximation technique, of its theoretical motivation and of its relation to other techniques such as backpropagation [20] can be found in [13, 14].

3 Implementation and performance

We have conducted an empirical investigation of the applicability of GRBFs, under a variety of conditions, to the problem of shape-based object recognition. The results of a series of

[1] Implementing a multidimensional receptive field as a product of 2D receptive fields all of which look at the same retina can result in "cross-talk" between different features if the spatial extent of the receptive fields is not limited. This does not seem to be a problem with Gaussian receptive fields, which respond very weakly to features that are far from the field's center (cf. [19]).

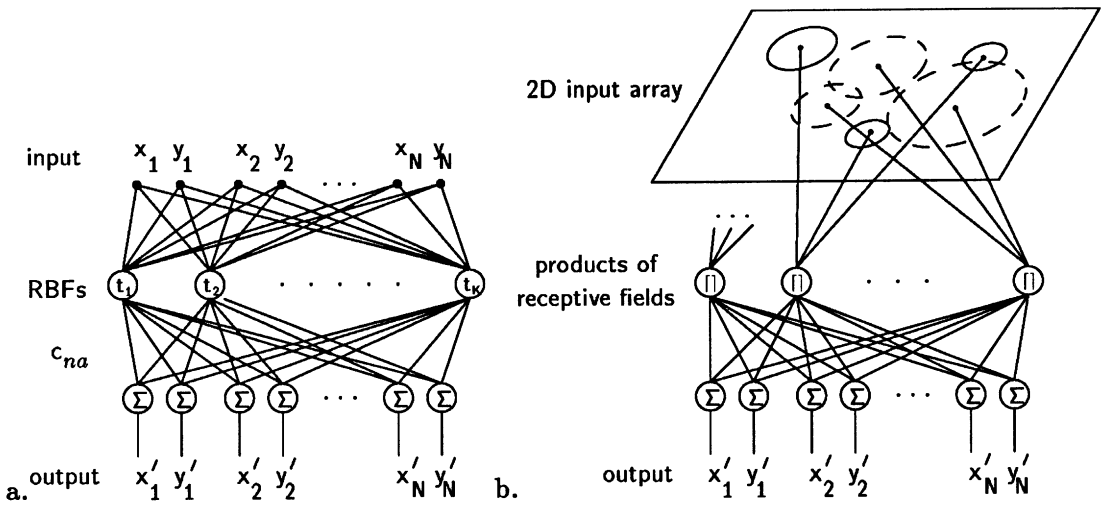


Figure 2: (a) A network representation of approximation by Generalized Radial Basis Functions. (b) shows an equivalent interpretation of (a) for the case of Gaussian radial basis functions. The solid circles in the image plane represent the 2D Gaussians associated with the first radial basis function, which corresponds to the first view of the object. The dotted circles represent the 2D receptive fields that synthesize the Gaussian radial function associated with another view.

experiments that involved simple computer-generated shapes are described below.

3.1 Input objects

Objects for testing the recognition scheme were created using the Symbolics S-Geometry 3D graphics modeling system. The objects were 5-segment random wire frames[2] (Figure 3). All the objects were positioned in such a manner that their centers of mass coincided with the origin of the 3D coordinate system defined by the modeling program. Different views of the objects were obtained by rotating the S-Geometry "camera" around the 3D origin, so that it could assume any position specified by two viewpoint coordinates, θ and ϕ, corresponding to the latitude and the longitude on an imaginary sphere centered at the object. No rotation of the camera around its optical axis was allowed.

3.2 Input representations

We have experimented with several different methods of encoding object shape, all of which employed exclusively the 2D information available in the projection of the objects' vertices onto the imaging plane. The first and most straightforward method was used in most of the experiments described in this section.

[2]In some of the experiments, 7-segment wires or other objects such as wire-frame cubes and octahedra were used.

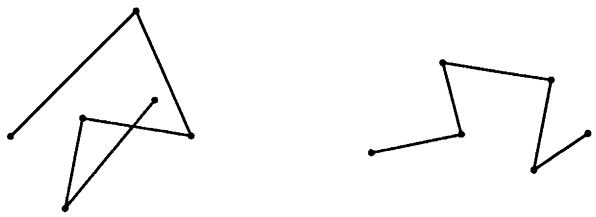


Figure 3: Two examples of wire objects used in the experiments. The wires were created by a random walk in 3D. They were encoded for training and subsequent recognition by projecting the vertices onto an imaging plane (under either orthographic or perspective projection). The resulting vector of x, y-coordinates could be further preprocessed to obtain different encodings (see section 3.2).

1. **XY-coordinates.** A list of the screen coordinates of the wire's vertices, $(x_1, y_1, \ldots, x_n, y_n)$. The origin of the screen coordinate system was at the upper left corner of the screen, and the coordinates varied in the [0..127] range.

2. **Centered XY-coordinates.** Same as previous, but with the origin at the screen projection of the 3D center of rotation common to all the objects.

3. **Segment lengths.** Screen distances between the projections of the successive vertices of the objects.

4. **Normalized segment lengths.** Same as previous, but with the lengths divided by the length of the first segment.

5. **Angles.** Angles formed by the projections of the successive segments.

6. **Angles + lengths.** A mixed encoding, combining the angles and the segment lengths in one heterogeneous vector.

Note that the fifth encoding method (angles) leads to the invariance of recognition performance with respect to translation, scaling and image-plane rotation of the objects. Another point of interest is that nothing in the present approach precludes information other than 2D shape from being incorporated into the input representation. In particular, 3D shape cues (obtained, e.g., through binocular stereo) can be used within the same framework depicted in Figure 1. We shall return to this point in the discussion.

3.3 Output representation

As depicted in Figure 1, the recognition module was trained to produce a standard output for any input that showed a view of the target object. The output representation was identical to the input one (as a matter of fact, the first input view was chosen as the standard one). However, in addition to the standard view of the object whose arbitrary view was presented as

input, the system was also capable of recovering other information about that object, namely, its attitude (as expressed by the viewpoint coordinates θ and ϕ).[3]

3.4 Test paradigm

The primary measure of the system's performance was the standard view recovery error, defined as the euclidean distance between an actual output and the ideal one. Two statistical measures of performance were computed in each of the experiments to be described below. These measures involved training the system on each of 10 different wire objects and comparing the standard view recovery errors for views of the trained object with those of the other nine objects. The errors for the trained object should be small, compared to the errors for the other objects (Figure 4). Ideally, the smallest error on a non-target object (call it $MIN_{nontarget}$) should be larger than the largest error on the target (MAX_{target}): a MIN/MAX ratio greater than 1 is required for a perfect separation between the target and other objects using a simple threshold decision. A less conservative measure is the ratio of the averages of the two error classes, AVG/AVG.[4]

3.5 Example of operation

Two examples of the module's operation, one in which the input is the training object, and another in which it is a different but similar object, appear in Figure 5. The top row shows the standard view of a wire frame object, superimposed on its estimate by the GRBF network (large black dots), when its input is a random view of the same object (second from top row). The fit is much closer than in the bottom two rows, where the input view belongs to a different object.

From Figure 5 it appears that arbitrary views of the target object cause the GRBF module to output a vector that is close to the ideal (trained) one. It also appears that views of non-target objects are transformed into scaled versions of the ideal vector, so that $Y'_{out} = kY_{out}(ideal)$, where $k < 1$. To understand why that happens, it is convenient to consider first a linear associative memory that is realized by a matrix operator C trained to recognize views of a target object by transforming them into a preset standard vector Y. Since C maps distinct vectors V_i to the same vector Y, it must be singular (it can be shown that the rows of C are all collinear). If the number of (randomly chosen) training views V_i is sufficiently large, there is a good chance that they span a 6-dimensional manifold that, to a first approximation, is a hyperplane in R^{2N} (see the appendix). Any new view V will lie within this hyperplane and will be mapped to a scaled kY. Views of non-target objects will tend to be orthogonal to the space spanned by the training views, resulting in $k \approx 0$. An analogous argument can be made for the RBF scheme, in which the linear mapping C is preceded by the application of the radial basis

[3] We have also experimented with a scalar output representation; see section 3.6.8.

[4] Standard statistical methods of parameter estimation and hypothesis testing may be used to translate the means and the standard deviations of $MIN_{nontarget}$, MAX_{target}, $AVG_{nontarget}$ and $AVG_{nontarget}$ into probabilities of Type I and Type II recognition errors (see e.g. [21]). Since these methods involve table lookup of probability distributions, we did not use them on-line. Characteristically for our experiments, a ratio of $AVG_{nontarget}$ to $AVG_{nontarget}$ of 5.0 sufficed to impose a 0.001 upper bound on the probabilities of both Type I and Type II errors.

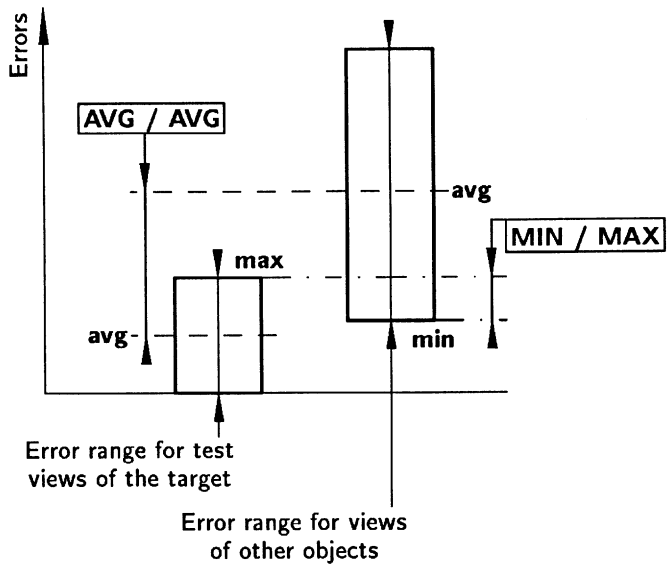


Figure 4: Definitions of the AVG/AVG and the MIN/MAX performance criteria used throughout the paper. The error here is defined as the euclidean distance between the standard view of the target and the actual output of the system (the smaller the error, the greater the likelihood that the input view belongs to the target). In this illustration, the average error for non-targets is considerably greater than that of target views. Consequently, there is a good chance of correct recognition of the target (and correct rejection of non-targets). An ideal performance requires that there be no overlap between the error value ranges corresponding to target and non-target views, in which case $MIN/MAX > 1$ and the two classes of views are separable by thresholding.

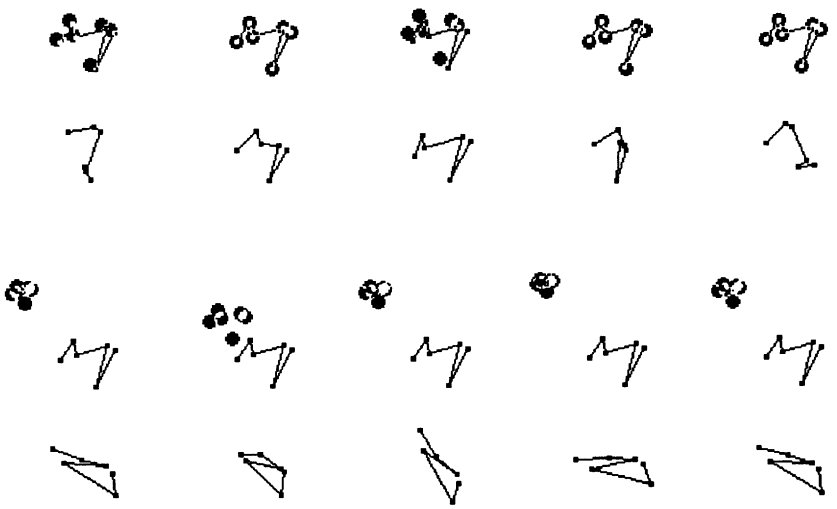
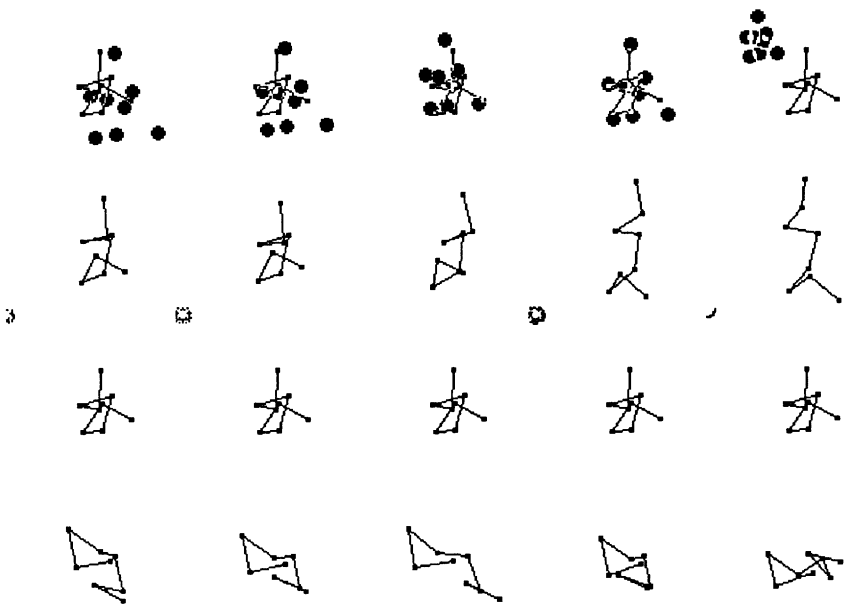


Figure 5: Examples of the module's operation. Above: standard view of a wire frame object (top row), superimposed on its estimate by the GRBF network (large dots), when its input is a random view of the same object (second from top row). The fit is much closer than in the bottom two rows, where the input view belongs to a different object. The number of training views $M = 40$, the number of RBFs $K = 20$ and the range of attitudes θ, ϕ is 0^o to 90^o. A naive fixed-step gradient descent (with a small number of steps) was used to obtain the optimal positions of the GRBF centers. Below: within a smaller range of $\theta, \phi \in [0^o, 45^o]$, the performance was acceptable with only two radial basis units: $M = 40$, $K = 2$ (Note that in the "different object" row the dots signifying the predicted vertex locations are in most cases off the scale).



functions G_α. The analogy is then between the original training vectors V_i and their images under G_α.

3.6 Performance

3.6.1 Effects of receptive field size and of number of centers

In the first experiment, the number K of RBF centers is made equal to the number M of training views by letting the centers coincide with the views themselves. Consider Figure 6, which shows the dependency of the error (distance between actual and ideal outputs) for random views of the trained object (left column) and the error for views of other objects (right column), as a function of K and of the size σ of the (Gaussian) basis functions. Figure 6 conveys information as to the relative significance of the average and worst-case performance of the recognition module over the depicted range of K and σ. The worst-case performance (assessed by comparing the upper curve in the left column with the lower curve in the right column) lags far behind the average performance (assessed by comparing the middle curves in the two columns). It should be noted that the role of the outliers that contribute to the worst-case measure is statistically insignificant, as long as the average performance does not drop below a certain threshold (corresponding to an AVG/AVG ratio of about 5).

The next plot provides a direct answer to the question of the optimal combination of K and σ. Under the AVG/AVG measure (Figure 7, middle column), it is $\sigma = 25$, for $K = 100 = M$ (clearly, increasing the number of training views and RBF centers improves the performance, but the price in terms of computational resources makes it probably not worthwhile to increase K and M beyond about $80 - 100$). Under the MIN/MAX measure, the best performance is achieved for $\sigma = 30$ (Figure 7, right column). The left column of Figure 7 gives a different perspective on the module's performance, by plotting the proportions of Type I and Type II recognition errors vs. σ. Note that having too much interpolation (in this case, $\sigma > 25$) sharply increases the probability of a Type II (false alarm or overgeneralization) error, as expected.

3.6.2 Effect of perspective projection

The result of Ullman and Basri [22] on representing objects by linear combinations of views suggests that recognition posed as a problem in function approximation is better behaved under orthographic than under perspective projection. We have tested the GRBF module with two different settings of the distance of the simulated camera from the objects: "near", in which there was an appreciable perspective distortion, and "far", in which the distortion was almost unnoticeable (this served as an approximation of orthographic projection condition). From Figure 8 it can be seen that doubling the distance from "near" to "far" made no significant difference in the performance.

A separate look at the false alarm and the miss rates (Figure 9) shows that if camera distance had any effect, it was on the miss rate. The most prominent effect was the decrease in the miss rate under orthographic approximation for $M = K = 20$. This finding is consistent with the Ullman-Basri theoretical argument for the relative ease of recognition under the assumption of orthographic projection.

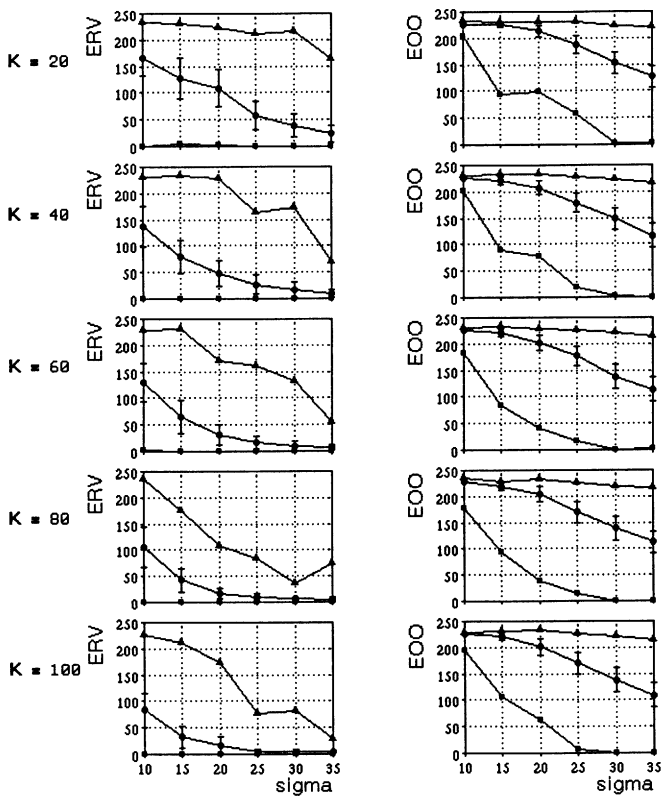


Figure 6: Error (distance between actual and ideal output) vs. the size σ of the basis functions, for modules with different number of centers K (the number of training views M is equal here to K). Data are shown for two input sets: random views of the trained object (ERV, left column) and views of other nine objects (EOO, right column). Three measures of the error, MIN (lower curves), AVG (middle curves) and MAX (upper curves) are shown separately. Bars indicate standard deviation, computed over ten different training objects.

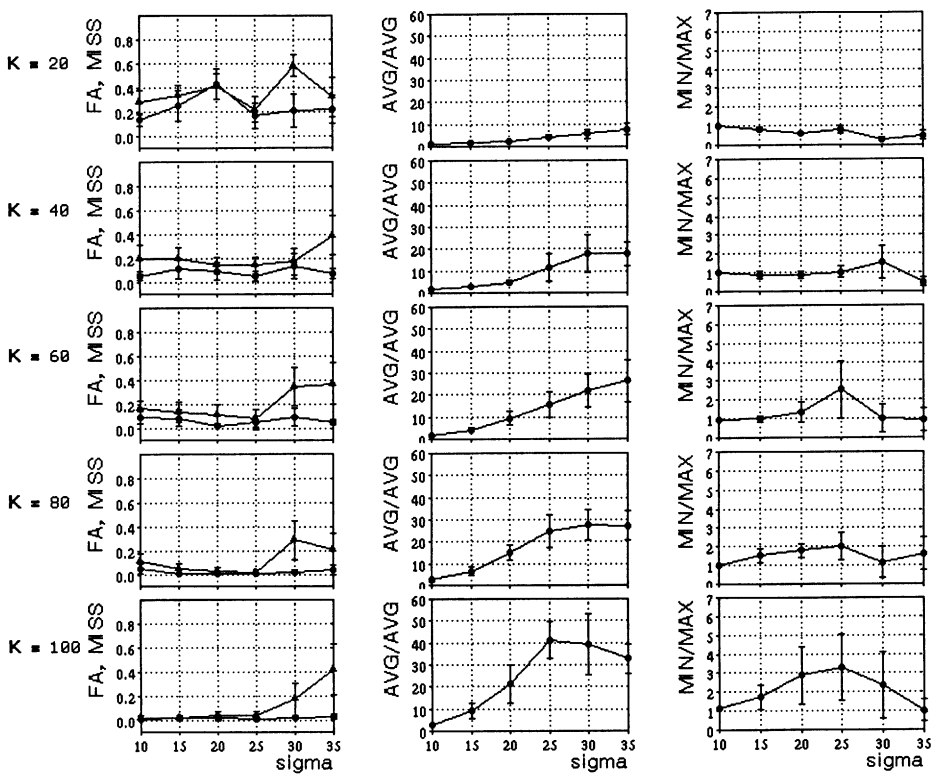


Figure 7: Left column: Type I or miss (MISS; lower curve) and Type II or false alarm (FA; upper curve) recognition error rates, vs. σ, by the number of centers K. Middle column: AVG/AVG performance index. Right column: MIN/MAX performance index (see section 3.4 and Figure 4) vs. σ, by K.

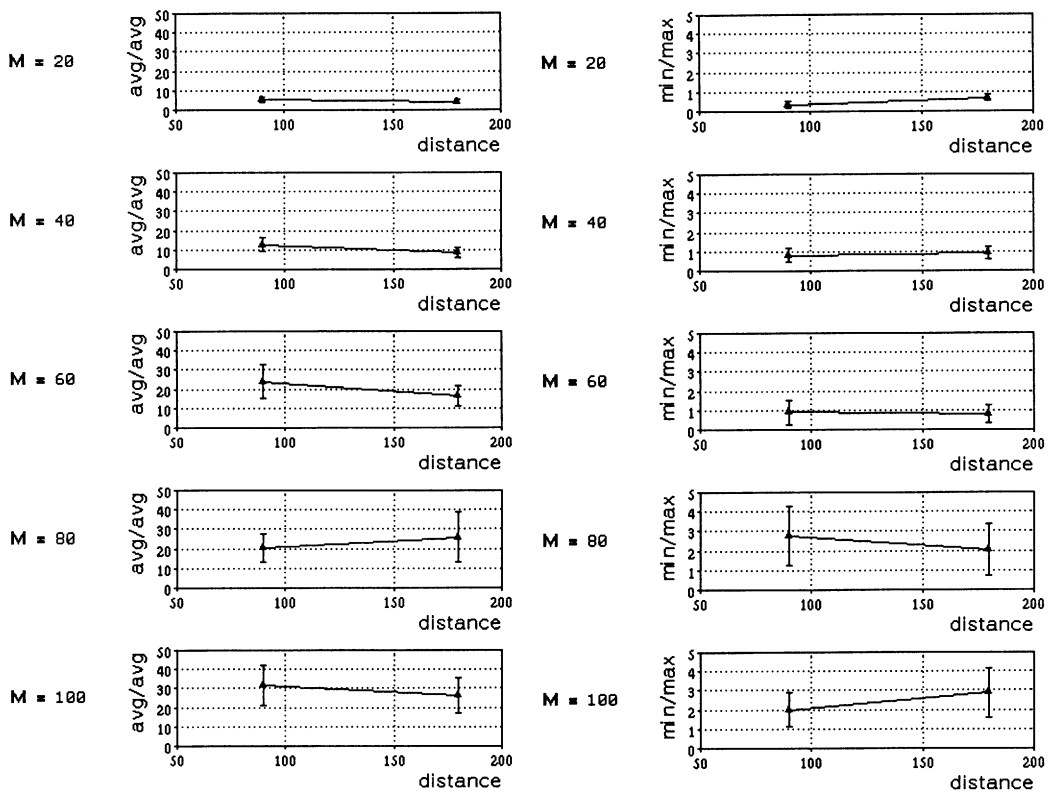


Figure 8: Left column: AVG/AVG performance index vs. the distance of the objects from the simulated camera, by the number of training views M (here $K = M$, $\sigma = 30.0$). The "near" distance is about seven times the apparent width of the wire objects used for this experiment. Right column: MIN/MAX performance index vs. the distance of the objects from the simulated camera, by the number of training views M (here $K = M$, $\sigma = 30.0$).

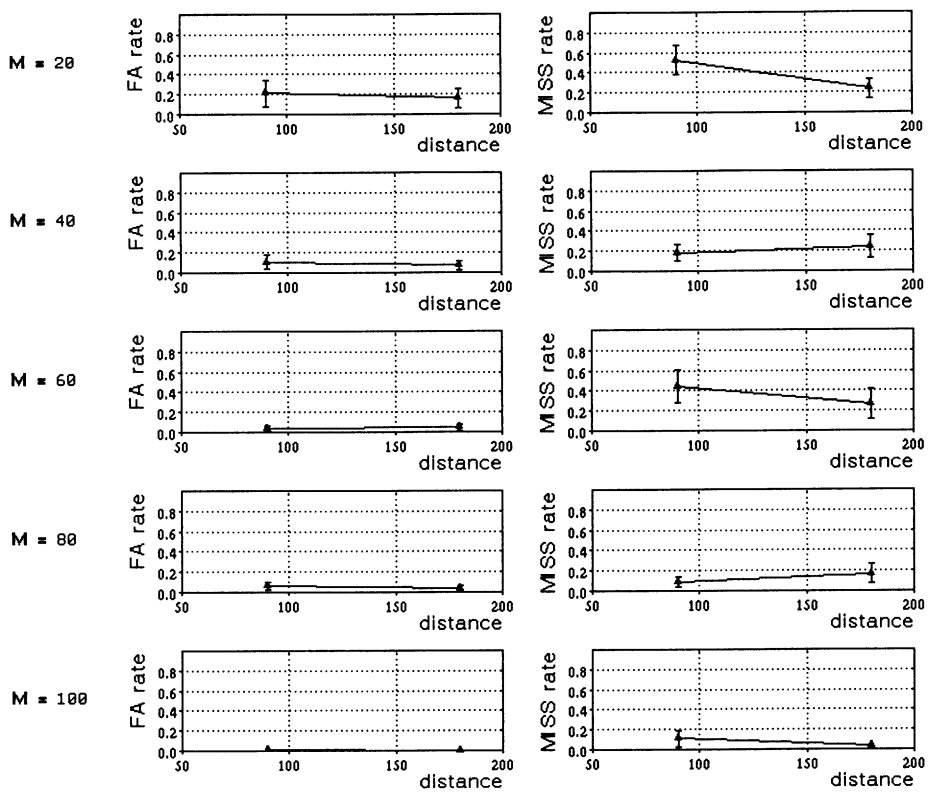


Figure 9: False alarm (FA, left column) and miss (MISS, right column) rates vs. the distance of the objects from the simulated camera, by the number of training views M (here $K = M$, $\sigma = 30.0$).

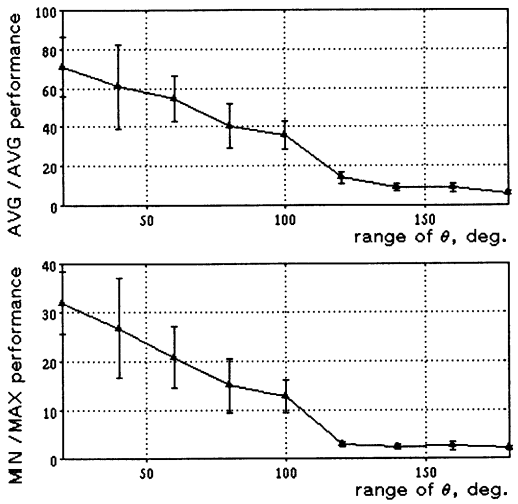


Figure 10: AVG/AVG and MIN/MAX performance vs. the range of the viewpoint coordinates θ, ϕ (the objects are a cube and an octahedron, $M = K = 40$ $\sigma = 30.0$, and the error bars are standard deviations over 10 sets of random training and testing views). Here and in the next figure, $\phi_{max} = 2\theta_{max}$, so that $\theta_{max} = 180^o$ corresponds to the full viewing sphere.

3.6.3 Effect of range of attitudes

If the number of training views is held constant, the performance of the GRBF module is expected to deteriorate with the increase in the range of the viewpoint coordinates into which the training views fall. Figure 10 shows that this indeed happens: for $M = K = 40$ and $\sigma = 30$, both the AVG/AVG and the MIN/MAX measures take a sharp dip when θ, ϕ reach $(120^o, 240^o)$.

3.6.4 Recovery of attitude

The range of the allowed orientations has a similar influence on the precision of the recovery of the orientation parameters θ, ϕ (Figure 11). For $M = K = 40$ and $\sigma = 30$, the mean square error of the recovered orientation stays below 10^o for $\theta < 120^o$, $\phi < 240^o$, rising to about 60^o for the full range of orientations. Doubling M and K extends effective recovery of θ, ϕ to the full range of orientations.

3.6.5 Effect of number of vertices

The power of the GRBF module to discriminate between trained object and other, similar objects increases with the increase in the number of vertices used in the encoding (Figure 12). The discrimination power is nil ($MIN/MAX = 0$) for two-vertex objects, rises steadily with the number of vertices, then starts to drop. This may be due to an interplay of two factors: the amount of information and cross-talk among GRBF centers. At least four points on each object are necessary for discrimination (see the appendix). The more vertices are used, the more

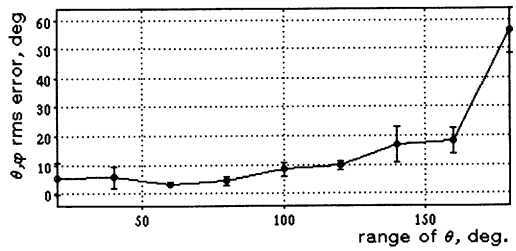


Figure 11: Errors in the viewpoint coordinates θ, ϕ recovered by the module vs. the range of the viewpoint coordinates ($M = K = 40$, $\sigma = 30$).

information there is for the recognizer to go by, until cross-talk sets in (which will happen if the size of the basis functions σ is not allowed to decrease in proportion with the increased density of object vertices in the image plane). In human recognition, a similar effect is intuitively expected (30-vertex wire objects seem to be too complicated to be distinguished by vertex positions alone).

3.6.6 Different input/output representations

The versatility of the present approach to recognition is illustrated in Figure 13, which shows superimposed a plot of the MIN/MAX performance vs. the number K of RBF centers for the regular encoding used throughout the paper (x, y-coordinate vectors) and a shift, scale and image-plane rotation invariant encoding (angles between successive segments of the wire objects). For a six-vertex object, the x, y-coordinate vector has length 12, while the angles vector has length 4. The relatively smaller amount of information in the angle encoding puts it at a disadvantage for smaller K's. For a large enough K the angle encoding yields higher MIN/MAX ratio, in addition to possessing desirable invariance with respect to shift, scale and rotation of the input.

3.6.7 Sensitivity to occlusion

To find out the sensitivity of the GRBF scheme to occlusion, we have repeatedly trained it on views each of whose constituent features had a fixed probability of being "occluded" (in which case the corresponding component of the representation vector was set to 0). Note that more than one feature could be occluded at a time.

The performance of the GRBF module in subsequent testing, plotted vs. the probability of individual vertex occlusion, is shown in Figure 14.[5] It appears from the figure that decent performance can be expected even when the probability of having any particular feature occluded is 0.2, in which case about three quarters of the training views had at least one of the features

[5] Although no occlusion was assumed in testing, one can get an idea of the scheme's sensitivity to this factor by considering Figure 12, which shows the effect of the number of features on the discrimination power.

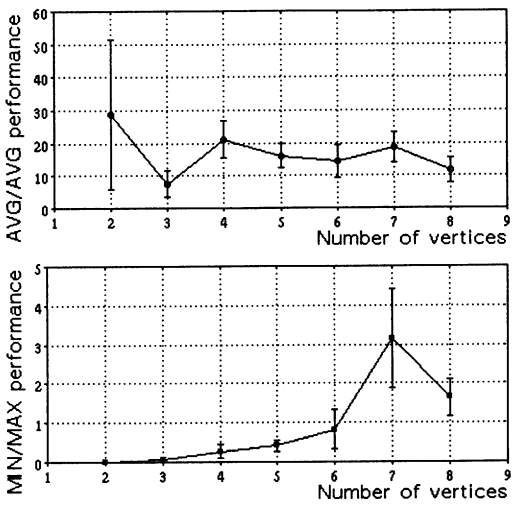


Figure 12: AVG/AVG and MIN/MAX indices vs. the number of vertices used in training (the data for number of vertices from 2 to 6 are for six-vertex random wire objects; the data for number of vertices 7 and 8 are for eight-vertex wires; $M = K = 60$, $\sigma = 30.0$).

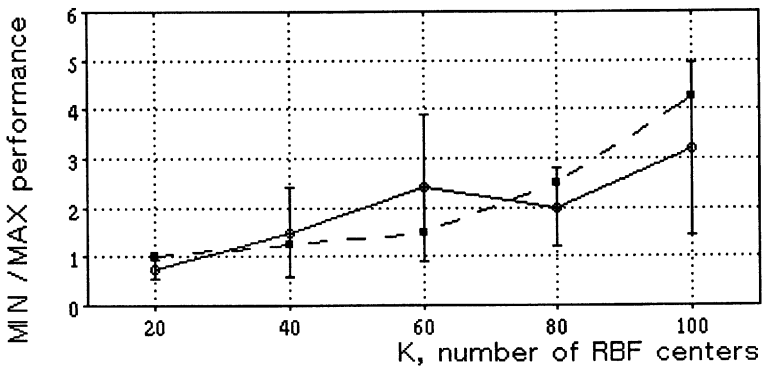


Figure 13: MIN/MAX performance for two types of input encoding: vertex coordinates (solid line) and angles formed by successive pairs of segments (dashed line; data for six-vertex random wire objects, σ chosen optimal for each encoding, $M = K$).

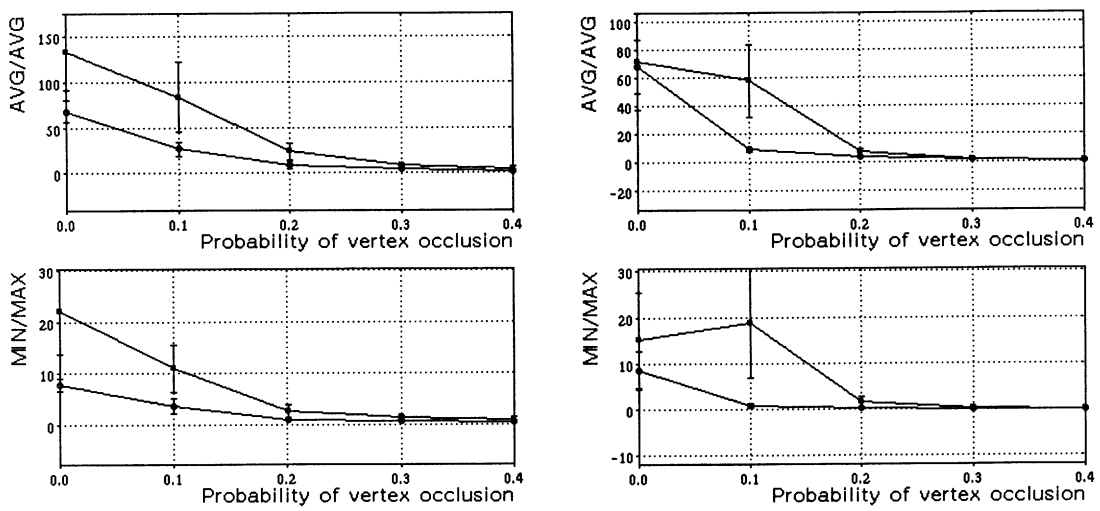


Figure 14: AVG/AVG and MIN/MAX indices vs. the probability of any given vertex being occluded (left: six-vertex random wire objects; right: eight-vertex objects). θ, ϕ were confined to one half of the viewing sphere; $K = M = 50$ (lower curves), $K = M = 100$ (upper curves); $\sigma = 30.0$.

occluded. Occlusion has had a somewhat stronger effect on the learning of eight-vertex wires (Figure 14, right column).

Note that in the present experiment the basic GRBF scheme was not augmented by any mechanism specifically designed to deal with occlusion. A better insensitivity to the deletion (occlusion) of features can be achieved by providing a basis function (center) for each possible subset of features. We conjecture that in practice the maximum size of necessary feature subsets is rather small. This size could be found during learning, by analyzing the weight matrix $\mathbf{W}$.

3.6.8 Scalar vs. vector output

If a compact output representation is required, it is possible to train the recognition module to produce a scalar output, as opposed to a vector that represents a standard view. Figure 15) shows that the single-output network performs on the average almost as well as the network of Figure 2 (which outputs a standard view vector). The advantage of the vector-output module may be explained by the larger number of its free parameters (elements of the $\mathbf{C}$ matrix).

3.6.9 GRBF, using gradient descent

In most of the experiments described in this report, the GRBF module was trained without searching for optimal center locations $\mathbf{t}_\alpha$, coefficients $\mathbf{C}$ or weights $\mathbf{W}$ (equation 3). In these cases, the centers were set at some of the training views, the matrix $\mathbf{C}$ was found by a generalized inverse method (see section 2), and an identity matrix was used as $\mathbf{W}$.

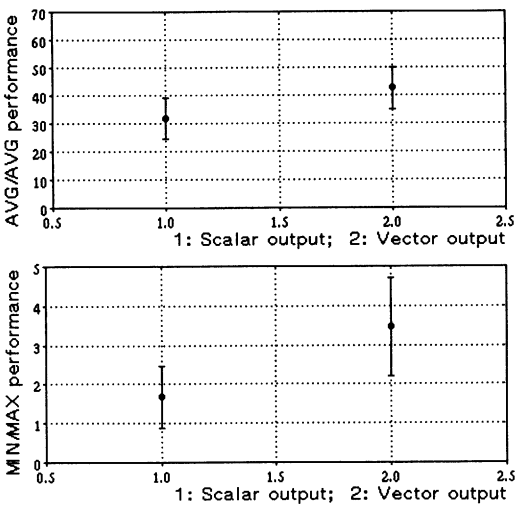


Figure 15: AVG/AVG and MIN/MAX performance for scalar and vector output (data for six-vertex random wire objects, $\sigma = 30$, $M = K = 80$; error bars show standard deviation computed over 10 objects). In the first case, the network is trained to output 1 when show views of the target. In the second case, the output is a standard view of the target.

The parameters $\mathbf{t}_\alpha$ and $\mathbf{C}$ obtained in this manner serve as a convenient starting point for improvement using gradient descent search in the parameter space. The gradient descent was performed according to the expressions given in [14].[6] We have compared the performance improvement for this encoding under three conditions: changing centers $\mathbf{t}_\alpha$, or weights $\mathbf{W}$, or both (the coefficient matrix $\mathbf{C}$ was always allowed to change), for two sets of parameter values. Only trials for which the gradient descent procedure actually converged were included in the comparison. The results for $M = 40$, $K = 10$ and a full range of viewpoints appear in Figure 16. Note that the best effects were achieved by a combined adjustment of $\mathbf{C}$, $\mathbf{t}_\alpha$ and $\mathbf{W}$ together. A visual example of the performance of the GRBF module with $K = 10$ centers and $M = 40$ training views after the adjustment of the centers' locations through gradient descent appears in Figure 5.

3.7 Comparison with related schemes

At this point it is natural to ask whether other, simpler network schemes can perform in the recognition task defined in this report as well as does the GRBF module. To address this question, we investigated the performance of three related schemes: linear associative memory and two versions of the nearest neighbor classifier (with and without feature correspondence).

[6] Since these expressions pertain to the case of a single-output network, we used such a network in this experiment.

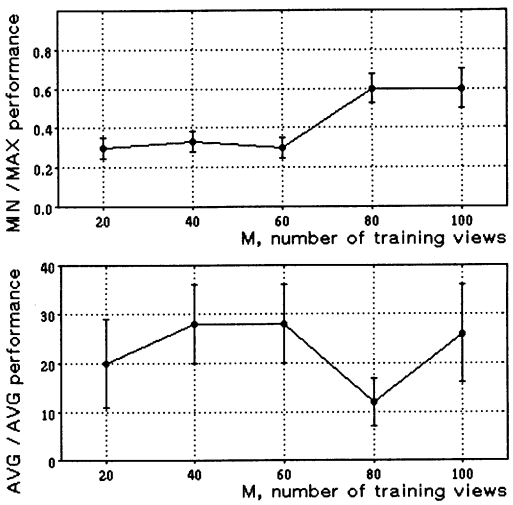


Figure 17: AVG/AVG and MIN/MAX indices vs. the number of training views for the linear associative scheme (six-vertex random wire objects).

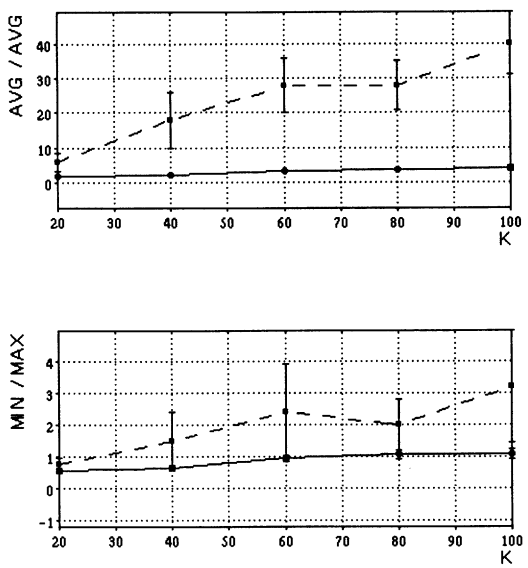


Figure 18: AVG/AVG and MIN/MAX indices vs. the number of remembered views for the nearest-neighbor method that uses correspondence information (six-vertex random wire objects). For comparison purposes, the performance of the RBF scheme with $M = K$ is also shown (dashed curve).

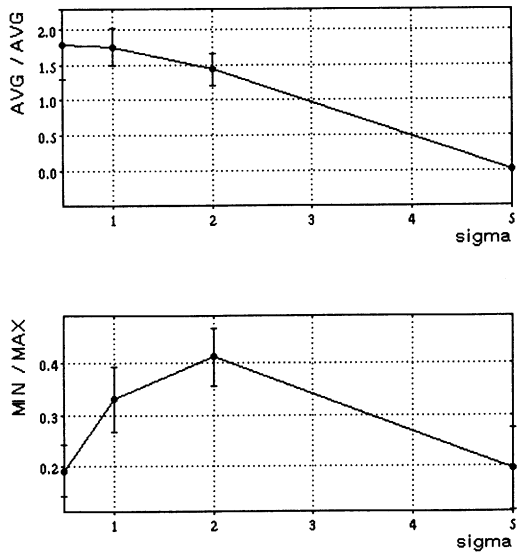


Figure 19: AVG/AVG and MIN/MAX indices vs. the width σ of the Gaussian blurring mask (see section 3.7.3) for the nearest-neighbor method that uses 2D correlation and 2D array representation of views, instead of correspondence information and 1D vector representation of views (six-vertex random wire objects; the number of remembered views is 80).

This requirement can be dispensed with, at the cost of reduced performance, as follows. Define recognition error for a given object as the inverse of the sum of 2D correlations between each of the stored training views (represented in this case as 2D arrays rather than as 1D vectors of vertex coordinates) and the input view. Low error would then be obtained for an input that is "close" to at least one of the stored views. To improve the generalization ability of the NN classifier that relies on 2D correlation, the input view is blurred (convolved with a Gaussian mask) before the correlations are computed. The dependence of the performance on the size of the blurring mask is shown in Figure 19.

4 Discussion

The reconstructionist dogma of computational vision appears recently to have fallen upon hard times. A standard version of this dogma holds it that the ultimate goal of a visual recognition system is the formation of object representations that make explicit the relevant 3D structure, just as a toy airplane makes explicit the relative size and position of the wings and the fuselage in the real airplane [24]. This view of recognition considers the 2D image bottleneck that necessarily intervenes between the distal object and its percept a nuisance, to be overcome, e.g., by invoking relevant physical and computational constraints [1]. Due to persistent difficulties at the higher levels of the reconstructionist program (see [25, 26, 27] for reviews), inverse optics all

the way to the top [1, 28] no longer seems to be the most promising approach to recognition.[8]

The performance of the GRBF module described in the foregoing sections suggests that object recognition can be done without first reconstructing the third dimension of the visual input, and without relying on three-dimensional object models (see also [29, 22, 19]). Furthermore, adopting the present approach to recognition does not mean giving up the use of information beyond 2D shape (color, texture and depth). Computationally, therefore, there seems to be no reason to reject the memory-based function approximation approach to recognition out of hand.

In the study of biological vision, the notion that in the primate visual system objects are represented by single units each of which responds selectively to a specific object, dubbed the grandmother cell dogma, used to draw criticism, for a number of reasons. The arguments given against it included the limited memory capacity of the brain and the lack of neurobiological and psychological support. The results reported in the previous sections indicate that doing function approximation rather than straightforward template matching may solve the memory capacity problem. Furthermore, the function approximation approach is also compatible with prominent biological and psychophysical findings on recognition, outlined below.

4.1 Biological aspects

4.1.1 Receptive fields

One feature of the GRBF scheme that may guide its biological interpretation is the expressibility of its function in terms of combinations of receptive fields. It is possible to decompose a multidimensional Gaussian radial basis function into a product of Gaussians of lower dimensions (Figure 2b). In our case, the center of a basis unit plays a role similar to a prototype and the unit's response profile is synthesized as the product of feature detectors with two-dimensional Gaussian receptive fields (i.e., the activity of a detector depends on the distance r between the stimulus and the center of the receptive field as e^{-r^2/σ^2}). The network's output (see equation 1) is the sum of these products and therefore represents the logical disjunction of conjunctions "$\bigvee_\alpha \bigwedge_i (feature\ F_i\ at\ (x_i, y_i))$", where the disjunction ranges over all the prototypes of the given object.

4.1.2 View-specific units

Cells that respond preferentially not only to a specific object, but to a limited range of that object's views, have been found in the inferotemporal cortex of monkeys by a number of researchers (see [30] for a review). The existence of these "grandmother cells" is compatible with the notion of a hierarchical structure of object representations. The lower level of this structure may be composed of receptive fields that transduce position of individual features into activity of units that encode their presence. The next level would correspond to "grandmother" units that encode specific views. In the GRBF terminology, these are the basis units, each centered around the view it is tuned to. At a still higher level, a "disembodied" representation of an

[8] Inverse methods appear to be useful in low-level visual tasks such as stereo and motion computation which contribute to the representation that Marr called $2\frac{1}{2}$D-sketch [16]. At the higher levels, the lack of well-defined constraints on the solution that are general enough to be relevant in real-life situations hinders the application of inverse methods.

object could be formed by combining several view-specific units, arriving at the disjunction of conjunctions representation that stands for the object, irrespective of viewpoint (or position, or size).[9]

4.1.3 Separating "what" from "where"

Rather than discarding the viewpoint information in the process of arriving at the viewpoint-invariant representation, the GRBF scheme can retrieve and output it separately (see Figure 1 and section 3.6.4). As a final parallel between GRBFs and visual neuroscience, we note that this separation of form and space resembles the separation between the ventral and the dorsal visual pathways, the first of which carries predominantly shape and the second – predominantly spatial information from the striate cortex towards temporal and parietal regions, respectively (see e.g. [31]).

4.2 Psychophysical aspects

4.2.1 Human object recognition

Another aspect on the biological plausibility of our approach to recognition is provided by psychological studies. Different features of human performance in object recognition can be interpreted in terms of characteristics of the underlying information-processing mechanism. We first mention briefly several of the most prominent relevant findings and phenomena.

Object constancy

Perhaps the most familiar of these is the phenomenon of object constancy: our ability to recognize things under widely changing conditions from a variety of viewpoints, and the lack of change in the apparent shape of objects under these different conditions. This phenomenon is usually illustrated with a simple object such as a coin, which can be easily recognized when seen at an oblique angle, and whose outline then appears to us as a circle tilted in depth rather than an image-plane ellipse. Importantly, object constancy works for considerably more complex things such as faces and letters. This has prompted some researchers [32] to postulate a shape normalization mechanism in object perception, whose function is to bring the viewed shape to a standard appearance before recognition is attempted.

Canonical views

The existence of standard or canonical views of objects predicted by the shape normalization theory is supported by a wide range of experimental data [33]. Canonical views of commonplace objects can be reliably characterized using several criteria. For example, when asked to form a mental image of an object, people usually imagine it as seen from a canonical perspective. In recognition, canonical views are identified more quickly than others, with response times decreasing monotonically with increasing subjective goodness.

[9] Marr ([1], p.15) argued that little understanding of how vision is done is gained by invoking the grandmother cell hypothesis if it is based only on neurophysiological data. Our approach complements the neurophysiological hypothesis by providing one possible computational account of the hierarchical structure of object representations, from feature detectors, through view-specific encoding, to grandmother cells.

Mental rotation

The monotonic increase in the recognition latency with misorientation of the object relative to a canonical view (as defined independently, e.g. through subjective judgement) prompts the interpretation of the recognition process in terms of a mechanism related to mental rotation [34]. Specifically, it seems that the recognition process may be decomposed into two stages, normalization and comparison [5]. In the first stage, the system carries out the transformation necessary to normalize the appearance of the input object. In the second stage, a comparison is made between the normalized input and a model stored in memory. Close agreement between the two then leads to the recognition of the input as an instance of the model (the two stages are presumably executed in parallel for a number of candidate object models). Practice with specific objects appears to cause the two-stage strategy to be abandoned in favor of a more memory-intensive, less time-consuming direct comparison strategy. Under direct comparison, many views of the objects are stored and recognition proceeds in essentially constant time, provided that the presented views are sufficiently close to one of the stored views [34, 35].

Invariant features and integration

Another possible way around the need for time-consuming mental rotation in recognition is through the use of viewpoint-invariant features. When the object shapes include potentially informative viewpoint-invariant features, and when the experimental setup encourages the use of such features, they apparently lead to the disappearance of sequential effects in recognition, even for objects that normally do exhibit such effects [36]. The human visual system also appears to be able to put to use in recognition cues other than shape, such as color and texture, when these are available.[10]

4.2.2 GRBF and the psychology of recognition

Although the GRBF-based recognition system can hardly be considered a complete model of human object recognition, some of its functional characteristics agree with the features of human performance outlined above. In particular, recognition by the recovery of a fixed standard view of the input object may be considered analogous to the phenomenon of object constancy. Furthermore, as an interpolation scheme, a GRBF module necessarily performs better on some of the views of the object it has been trained upon (specifically, on the views corresponding to the centers of the basis functions) than on other, random views. This characteristic resembles the phenomenon of canonical views. Finally, as we have already mentioned above, a GRBF-based recognizer can accept inputs from diverse sources of visual information (as well as from non-visual sensors).

At least two of the features inherent in the present formulation of the interpolation-based approach to recognition mar its plausibility as a functional model of human object recognition. The first of these, the reliance on a supervised learning procedure, can in principle be dispensed with by modifying the scheme to incorporate adaptive data-driven clustering, and to associate a constant output vector with all the inputs that fall within the same cluster (rather than relying on an externally supplied input-output pairs as it is done at present). The second shortcoming of the present formulation lies in its disregard of the dynamics of object recognition. In particular,

[10] In addition, cross-modality cues (auditory and haptic) are readily incorporated.

the GRBF approach ignores the time course of the recognition process and its modification with practice (as manifested in the shift from time-intensive to memory-intensive strategy apparent in human performance).

4.2.3 Comparison with CLF

A model of object recognition that attempts to address these issues and appears to be related to GRBF is the CLF (conjunction of localized features) of [37, 19]. In this model, formulated as a two-layer network, units in the second (representation) layer come to represent patterns in the first (input) layer through an unsupervised Hebbian reinforcement mechanism. Sequences of second-layer units correspond to multiple-view representations of input objects, with the association between successive views predicated on the existence of apparent motion between the views during training (that is, the two successive views must resemble each other and must be sufficiently close in time to be able to elicit the perception of apparent motion in a human observer). Thus, on one hand, the CLF model represents an object by a disjunction of conjunctions of the presence of features in specific (fuzzy or blurred) locations in the image, just as the GRBF module does. On the other hand, the CLF model is able to replicate the dynamic behavior of the human object recognition system, mentioned above, through non-uniform activation of sequences of representation units and their modification with practice.

4.2.4 Two predictions

Assuming that a scheme resembling GRBF or other kind of prototype interpolation is the basis of the human ability to recognize objects allows one to formulate strong predictions regarding human performance in specific experiments. The most important of these predictions states that the ability of the visual system to generalize recognition to a novel view of an object should drop off significantly with the misorientation of the novel view relative to the familiar views of that object. Furthermore, the drop-off rate should be independent of the relative configuration of the familiar views in the viewpoint coordinate space.

Other contemporary models of object recognition generate different predictions when brought to bear on these points. Ullman's alignment model [5] (as well as the related models of Lowe [6] and Thompson & Mundy [4], Biederman's RBC theory [38] and most computer vision works on recognition) predicts no first-order dependency of recognition rate on misorientation. The linear scheme of Ullman and Basri [22] also predicts no such dependency, except when its three training views are coplanar in the viewpoint coordinate space. The predictions of the CLF model of Edelman [37, 19], on the other hand, agree with these of the GRBF scheme (which is not surprising, since the two appear to be related).

Experimental findings seem to support the limited generalization view of recognition shared by the GRBF and the CLF models. Descriptions of some relevant experiments can be found in [39, 40, 41, 34, 42, 35, 43]. Work that should further elucidate this point is currently under way in our laboratory.

5 Summary

We have described experiments with a versatile pictorial prototype-based learning scheme for 3D object recognition. The GRBF scheme seems to be amenable to realization in biophysical

hardware because the only kind of computation it involves can be effectively carried out by combining receptive fields. Furthermore, the scheme is computationally imposing because it brings together the old notion of a "grandmother" cell and the rigorous approximation methods of regularization and splines.

Acknowledgement

We thank Federico Girosi and Shimon Ullman for useful comments, and Daphna Weinshall for many discussions and for helping us with understanding [29].

6 Appendix: The result of Ullman and Basri for orthographic projection

Ullman and Basri [29, 22] have recently discovered the striking fact that under orthographic projection a view of a 3D object is the linear combination of a small number of views of the same object. In this appendix, we reformulate their results in the more abstract setting of linear algebra. This framework makes the result very transparent: the constraint of linear transformation (the same linear transformation for each vertex) implies immediately that the set of views of an object spans a 9-dimensional space, independently of the number of vertices; orthographic projection preserves linearity while reducing the number of dimensions to 6. Simple considerations show that the linear spaces of the x and y coordinates are nonintersecting and that each has dimension 3. This appendix describes the previous statements in more details.

6.1 Any view of a 3D object is a linear combination of a small, fixed numer of views

This section provides the main result (in the second subsection).

6.1.1 Any 3D-view of an object is a linear combination of 9 views

Let us define a *3D-view* of a 3D object as:

$$\mathbf{X}^{obj} = \begin{pmatrix} x_1 \\ y_1 \\ z_1 \\ x_2 \\ y_2 \\ z_2 \\ \cdot \\ \cdot \\ \cdot \\ x_n \\ y_n \\ z_n \end{pmatrix}$$

with $X \in \mathcal{R}^{3n}$, which is a vector space in the usual way.

Consider the set of *uniform* (our definition) linear operators on X^{obj}, defined by the $3n \times 3n$ matrices $\mathbf{L}^{3n}$:

$$\mathbf{L}^{3n} = \begin{pmatrix} L & 0 & . & 0 \\ 0 & L & . & 0 \\ . & . & . & . \\ 0 & 0 & . & L \end{pmatrix}$$

where

$$L = \begin{pmatrix} l_{11} & l_{12} & l_{13} \\ l_{21} & l_{22} & l_{23} \\ l_{31} & l_{32} & l_{33} \end{pmatrix}$$

is an affine transformation on $\mathcal{R}^3$. Translation in $3D$ space is taken care of separately (see later).

The space of the $\mathbf{L}^{3n}$ operators is a vector space which is *isomorphic* to the vector space of the L matrices. It therefore has a basis of 9 elements independently of n. We can express

$$\mathbf{L}^{3n} = \sum_{i=1}^{9} a_i \mathbf{L}_i^{3n}$$

where a_i are the $l_{i,j}$ and L_i^{3n} is the usual basis for L^{3n}, and thus

$$\mathbf{X}^{obj} = \mathbf{L}^{3n}\mathbf{X}_0^{obj} = \sum_{i=1}^{9} a_i \mathbf{L}_i^{3n}\mathbf{X}_0^{obj} = \sum_{i=1}^{9} a_i \mathbf{X}_i^{obj}$$

where $\mathbf{X}_i^{obj}$ are 9 independent 3D views of the specific object, needed to span the 9 elements of L, 3 for each coordinate. Thus:

Theorem 6.1 *The vector space V_{0b}^{3D} generated by uniform linear transformations on a 3D view of a specific object is a 9-dimensional subspace of $\mathcal{R}^{3n}$ (3 dimensions each for x, y and z).*

Thus any object ob_i generates a corresponding low dimensional subspace $V_{ob_i}^{3D}$ of all possible views of all objects ($\mathcal{R}^{3n}$). Of course, $V_{ob_i}^{3D} \not\equiv \mathcal{R}^{3n}$, *iff* $n > 3$. In other words, to have object specificity, i.e., for this result to be *nontrivial*, it is necessary that $n > 3$. Notice that in a sense $\mathcal{R}^{3n} = V_{ob_i} + V_{ob_2} + \ldots$.

6.1.2 Any 2D-view of a 3D object is a linear combination of 6 2D-views

Now consider the orthographic projection $P : \mathcal{R}^{3n} \rightarrow \mathcal{R}^{2n}$, defined by $P\mathbf{X} = \mathbf{x}$, that is

$$P\begin{pmatrix} x_1 \\ y_1 \\ z_1 \\ x_2 \\ y_2 \\ z_2 \\ . \\ . \\ . \\ x_n \\ y_n \\ z_n \end{pmatrix} = \begin{pmatrix} x_1 \\ y_1 \\ x_2 \\ y_2 \\ x_3 \\ y_3 \\ . \\ . \\ . \\ x_n \\ y_n \end{pmatrix}$$

with P being a linear operator with the matrix representation

$$P = \begin{pmatrix} 1 & 0 & . & . & . & . & . & . & 0 \\ 0 & 1 & 0 & . & . & . & . & . & 0 \\ 0 & 0 & 0 & 1 & 0 & . & . & . & 0 \\ . & . & . & . & . & . & . & . & . \\ . & . & . & 0 & 0 & . & 1 & 0 & 0 \\ 0 & 0 & . & . & . & . & 0 & 1 & 0 \end{pmatrix}$$

We define $\mathbf{x}$ as the *2D-view* of a 3D object. The result below follows immediately (6 views span the elements of L in the first 2 rows) and is the main result of Ullman and Basri (in a different formulation):

Theorem 6.2 *The vector space V_{ob_i} given by $V_{ob_i} = PV_{ob_i}^{3D}$ is a 6-dimensional subspace of $\mathcal{R}^{2n}$ (the space of all 2D orthographic views of all 3D objects), i.e. $\mathbf{x_{ob}} = \sum_{i=1}^{6} a_i \mathbf{x}_{ob}^i$.*

The inclusion of rigid translations is equivalent to the addition of a two-dimensional linear subspace (the same for all objects), spanned by the vectors

$$\mathbf{t_x}^{obj} = \begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \\ . \\ . \\ . \end{pmatrix}$$

and

$$\mathbf{t_y}^{obj} = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 1 \\ . \\ . \\ . \end{pmatrix}$$

6.2 The x and the y coordinates of a view are each a separate linear combination of 3 views

In the previous section we have seen that any *2D-view* of a 3D object under orthographic projection is the linear combination of 6 *2D-views*. This section reformulates another observation of Ullman and Basri: the x coordinates of a *2D-view* are a linear combination of the x coordinates of 3 *2D-views* and the y coordinates are an independent linear combination of the y coordinates of 3 *2D-views*.

Let us consider a similarity transformation of x:

$$T\mathbf{X} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ . \\ . \\ . \\ x_n \\ y_1 \\ y_2 \\ . \\ . \\ . \\ y_n \\ z_1 \\ . \\ . \\ . \\ z_n \end{pmatrix}$$

Under this similarity transformation, $\mathbf{L}^{3n}$ becomes a 3×3 matrix of 9 (that is 3×3) blocks. Each block is a multiple of $I \in \mathcal{R}^{n,n}$ (notice the "isomorphism" to L).

$$T^T L T = \begin{pmatrix} I_{11} & I_{12} & I_{13} \\ I_{21} & I_{22} & I_{23} \\ I_{31} & I_{32} & I_{33} \end{pmatrix}$$

where

$$I_{11} = \begin{pmatrix} l_{11} & 0 & 0 & . \\ 0 & l_{11} & 0 & . \\ 0 & 0 & l_{11} & . \\ . & . & . & . \end{pmatrix}$$

and so on for the other blocks.

The same argument of the previous section makes it clear that if we define

$$\xi = \begin{pmatrix} x_1 \\ . \\ . \\ . \\ x_n \end{pmatrix}$$

$$\eta = \begin{pmatrix} y_1 \\ . \\ . \\ . \\ y_n \end{pmatrix}$$

then the following holds:

$$\xi = \sum_{i=1}^{3} l_{1i}\xi_i$$

$$\eta = \sum_{i=1}^{3} l_{2i}\eta_i.$$

Thus we have proved:

Theorem 6.3 *The subspace spanned by the vectors ξ - the x components of $\mathbf{x}^{obj}$ - which is an n-dimensional subspace of V_{ob}^{2D} (which is 2n-dimensional), is spanned by three views of the x coordinates of the object undergoing uniform transformations, i.e., each ξ can be represented as the linear combination of 3 independent ξ_i. The same is true for the η: each η is an independent linear combination of 3 η_i. Again, $n > 3$ in order for this to be non-trivial (since $\xi \equiv \mathcal{R}^n$ for $n \leq 3$).*

Remark: The bases of ξ and the basis of η depend on the specific object.

6.3 V_x and V_y have the same basis, i.e., 1.5 snapshots suffice

We know from the previous sections that $V_{ob_i}^{2N} = V_x^N \oplus V_y^N$, where $dim V_x = dim V_y = 3$. A stronger property holds

Theorem 6.4 $V_x = V_y$

Proof: Assume that V_x and V_y are not identical: then there is a vector $\mathbf{y}$ which is in V_y and not in V_x (or vice versa). Then one can take the 3D view that originated $\mathbf{y}$ (through orthogonal projection) and apply to it a legal transformation consisting of a rigid rotation of 90 degrees in the image plane (such a transformation is in L and therefore is legal). The x view of that 3D vector is the $\mathbf{y}$, contradicting the assumption. It follows that $V_x = V_y$.

Remarks:

1. The same argument shows that $V_x = V_y = V_z$.
2. The same basis of three vectors spans V_x and V_y (separately).

3. The property that the x views and the y views of the same 3D object from the same snapshot are independent is generic, since if they were dependent, a very slightly different object, differing only in the y coordinate of one vertex would have independent views (observation due to Bruno Caprile).

4. In general, 1.5 snapshots are sufficient.

6.4 The case of rigid transformations, i.e., rotations in 3D

The previous two section have considered the case of *uniform linear transformations* in 3D of a 3D object. The space of such transformations is a vector space that contains as a nonlinear subspace the space of the rigid rotations in 3D (which is easily seen not to be a vector space). Can we characterize what the restriction to rigid rotations means? This section addresses this question.

Consider the restriction $L = R$ with $R^T R = I$. Then:

$$\begin{cases} l_{11}^2 + l_{12}^2 + l_{13}^2 = 1 \\ l_{11}l_{21} + l_{12}l_{22} + l_{13}l_{23} = 0 \\ l_{21}^2 + l_{22}^2 + l_{23}^2 = 1 \end{cases}$$

The equations define a nonlinear subspace of the space $\xi = \{l_{11}, l_{12}, l_{13}\}$ isomorphic to $\mathcal{R}^3$, and of $\eta = \{l_{21}, l_{22}, l_{23}\}$, also isomorphic to $\mathcal{R}^3$. Of course, ξ is a linear subspace of $\mathcal{R}^n$, the *space of all views of the x coordinates of all objects.* Rotations are the intersection of ξ with the conics defined by the previous equations.

The 2D views of one object defined by uniform affine transformations span $\{l_{11}, l_{12}, l_{13}\} = \mathcal{R}^3$. The 2D views of one object defined by rigid transformations, i.e., rotations, span a nonlinear subspace of $\mathcal{R}^3$, namely, the surface of the unit sphere in $\mathcal{R}^3$. All points on the *unit* sphere are allowed for $\{l_{11}, l_{12}, l_{13}\}$ (thus we "use up" two parameters). The triplet $(l_{21}l_{22}l_{23})$ is determined as one parameter family. Geometrically, once the vector l_{11}, l_{12}, l_{13} is fixed on the unit sphere, an orthogonal circle is determined on which the vector (l_{21}, l_{22}, l_{23}) must lie.

6.5 Summary of the appendix

The main point of this appendix can be summarized as a characterization of the algebraic structure (as a linear vector space) of the views of one object under orthographic projection.

Consider the space $\mathcal{R}^{3N}$ of 3D views of all objects. Consider the subspace $V_{ob_i}^{3N}$ generated by one view of a specific object and by the action on it of the group of *uniform* transformations $\mathcal{L}$, that transform in the same way each vertex. $\mathcal{L}$ is an algebra of order 9, and therefore a linear vector space isomorphic to $\mathcal{R}^3 \times \mathcal{R}^3$. Thus, $V_{ob_i}^{3N}$ is a linear vector space isomorphic to $\mathcal{R}^9$. The projection operator (orthographic projection) that deletes the z components from the 3D views, maps $V_{ob_i}^{3N}$ into a linear vector subspace $V_{ob_i}^{2N}$, isomorphic to $\mathcal{R}^6$. $V_{ob_i}^{2N}$ consists of vector with x and y components and can be written as the direct sum $V_{ob_i}^{2N} = V_x^N \oplus V_y^N$, where V_x^N and V_y^N are non-intersecting linear subspaces, each isomorphic to $\mathcal{R}^3$. In addition, we have proved that $V_x^N = V_y^N$, which implies that 1.5 snapshots are sufficient for "learning" an object (in general). If 3D translations are included, a linear subspace, isomorphic to $\mathcal{R}^2$, must be added to the linear space spanned by the 2D views of one object. The above is a short

alternative proof of the main results of Ullman and Basri [29] (with the exception of the 1.5 views result, see [22]).

References

[1] D. Marr. *Vision.* W. H. Freeman, San Francisco, CA, 1982.

[2] T. Poggio, E. B. Gamble, and J. J. Little. Parallel integration of vision modules. *Science*, 242:436–440, 1988.

[3] M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24:381–395, 1981.

[4] D. W. Thompson and J. L. Mundy. Three-dimensional model matching from an unconstrained viewpoint. In *Proceedings of IEEE Conference on Robotics and Automation*, pages 208–220, Raleigh, NC, 1987.

[5] S. Ullman. Aligning pictorial descriptions: an approach to object recognition. *Cognition*, 32:193–254, 1989.

[6] D. G. Lowe. *Perceptual organization and visual recognition.* Kluwer Academic Publishers, Boston, MA, 1986.

[7] W. E. L. Grimson and T. Lozano-Pérez. Localizing overlapping parts by searching the interpretation tree. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 9:469–482, 1987.

[8] T. J. Fan, G. Medioni, and R. Nevatia. Recognizing 3D objects using surface descriptions. In *Proceedings of the 2nd International Conference on Computer Vision*, pages 474–481, Tarpon Springs, FL, 1988. IEEE, Washington, DC.

[9] R.Y. Tsai and T.S. Huang. Uniqueness and estimation of three dimensional motion parameters of rigid objects with curved surfaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6:13–27, 1984.

[10] H. C. Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. *Nature*, 293:133–135, 1981.

[11] S. Ullman. *The interpretation of visual motion.* MIT Press, Cambridge, MA, 1979.

[12] J. J. Koenderink and A. J. van Doorn. The internal representation of solid shape with respect to vision. *Biological Cybernetics*, 32:211–217, 1979.

[13] T. Poggio and F. Girosi. A theory of networks for approximation and learning. A.I. Memo No. 1140, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, 1989.

[14] T. Poggio and F. Girosi. Regularization algorithms for learning that are equivalent to multilayer networks. *Science*, 247:978–982, 1990.

[15] A. N. Tikhonov and V. Y. Arsenin. *Solutions of ill-posed problems.* W. H. Winston, Washington, D.C., 1977.

[16] T. Poggio, V. Torre, and C. Koch. Computational vision and regularization theory. *Nature*, 317:314–319, 1985.

[17] M. J. D. Powell. Radial basis functions for multivariable interpolation: a review. In J. C. Mason and M. G. Cox, editors, *Algorithms for approximation.* Clarendon Press, Oxford, 1987.

[18] D. S. Broomhead and D. Lowe. Multivariable functional interpolation and adaptive networks. *Complex Systems*, 2:321–355, 1988.

[19] S. Edelman and D. Weinshall. A self-organizing multiple-view representation of 3D objects. A.I. Memo No. 1146, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, August 1989.

[20] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning representations by back-propagating errors. *Nature*, 323:533–536, 1986.

[21] W. Mendenhall and T. Sincich. *Statistics for the engineering and computer sciences.* Macmillan, London, 1988.

[22] S. Ullman and R. Basri. Recognition by linear combinations of models. A.I. Memo No. 1152, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, 1990.

[23] A. Hurlbert and T. Poggio. Synthesizing a color algorithm from examples. *Science*, 239:482–485, 1988.

[24] J. H. Connell. Learning shape descriptions: generating and generalizing models of visual objects. A.I. TR No. 853, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, 1985.

[25] A. Sloman. What are the purposes of vision? CSRP 066, University of Sussex, 1987.

[26] P. Quinlan. Visual object recognition reconsidered, 1989. submitted for publication.

[27] S. Edelman and D. Weinshall. Computational vision: a critical review. A.I. Memo No. 1158, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, October 1989.

[28] J. Y. Aloimonos and D. Shulman. *Integration of visual modules: an extension of the Marr paradigm.* Academic Press, Boston, 1989.

[29] R. Basri and S. Ullman. Recognition by linear combinations of models. Technical report, The Weizmann Institute of Science, 1989.

[30] D. I. Perrett, A. J. Mistlin, and A. J. Chitty. Visual neurones responsive to faces. *Trends in Neurosciences*, 10:358–364, 1989.

[31] S. Zeki and S. Shipp. The functional logic of cortical connections. *Nature*, 335:311–317, 1989.

[32] S. E. Palmer. The psychology of perceptual organization: a transformational approach. In J. Beck, B. Hope, and A. Rosenfeld, editors, *Human and machine vision*, pages 269–340. Academic Press, New York, 1983.

[33] S. Palmer, E. Rosch, and P. Chase. Canonical perspective and the perception of objects. In J. Long and A. Baddeley, editors, *Attention and Performance IX*, pages 135–151. Erlbaum, Hillsdale, NJ, 1981.

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

[35] S. Edelman and H. H. Bülthoff. Recognition and representation of novel 3D objects, 1990. submitted.

[36] P. Jolicoeur and B. Milliken. Identification of disoriented objects: effects of context of prior presentation, 1989. in press.

[37] S. Edelman and T. Poggio. Integrating visual cues for object segmentation and recognition. *Optic News*, 15:8–15, May 1989.

[38] I. Biederman. Human image understanding: Recent research and a theory. *Computer Vision, Graphics, and Image Processing*, 32:29–73, 1985.

[39] P. Jolicoeur. The time to name disoriented objects. *Memory and Cognition*, 13:289–303, 1985.

[40] I. Rock and J. DiVita. A case of viewer-centered object perception. *Cognitive Psychology*, 19:280–293, 1987.

[41] I. Rock, D. Wheeler, and L. Tudor. Can we imagine how objects look from other viewpoints? *Cognitive Psychology*, 21:185–210, 1989.

[42] S. Edelman, H. Bülthoff, and D. Weinshall. Stimulus familiarity determines recognition strategy for novel 3D objects. A.I. Memo No. 1138, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, July 1989.

[43] H. H. Bülthoff and S. Edelman. Psychophysical support for a 2D interpolation theory of object recognition, 1990. submitted.

CS-TR Scanning Project
Document Control Form

Date : 11 / 30 / 94

Report # AIM-1181

Each of the following should be identified by a checkmark:
Originating Department:

☒ Artificial Intellegence Laboratory (AI)
☐ Laboratory for Computer Science (LCS)

Document Type:

☐ Technical Report (TR) ☒ Technical Memo (TM)
☐ Other:_____

Document Information

Number of pages: 35 (40-IMAGES)
Not to include DOD forms, printer intstructions, etc... original pages only.

Originals are:
☒ Single-sided or
☐ Double-sided

Intended to be printed as :
☐ Single-sided or
☒ Double-sided

Print type:
☐ Typewriter ☐ Offset Press ☒ Laser Print
☐ InkJet Printer ☐ Unknown ☐ Other:_____

Check each if included with document:

☒ DOD Form ☐ Funding Agent Form ☐ Cover Page
☐ Spine ☐ Printers Notes ☐ Photo negatives
☐ Other: _____

Page Data:

Blank Pages(by page number):_____

Photographs/Tonal Material (by page number):_____

Other (note description/page number):

Description :	Page Number:
IMAGE-MAP - (1) TITLE PAGE UNNUMBERED	
(2-35) PAGES #'ED 1-34	
(36) SCANNING TARGET	
(37-39) TARGET	
(40) DOD	

Scanning Agent Signoff:
Date Received: 11/30/94 Date Scanned: 12/4/95 Date Returned: 12/5/95

Scanning Agent Signature: Michael W. Cooper

Scanning Agent Identification Target

Scanning of this document was supported in part by the **Corporation for National Research Initiatives,** using funds from the **Advanced Research Projects Agency** of the **United states Government** under Grant: **MDA972-92-J1029.**

The scanning agent for this project was the **Document Services** department of the **M.I.T Libraries.** Technical support for this project was also provided by the **M.I.T. Laboratory for Computer Sciences.**

Scanned

Date: 1/4/1995

M.I.T. Libraries
Document Services

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: AIM 1181	2. GOVT ACCESSION NO: AD-A225514
3. RECIPIENT'S CATALOG NUMBER:	
4. TITLE (and Subtitle): Bringing the Grandmother back into the Picture: A Memory-based View of Object Recognition	5. TYPE OF REPORT & PERIOD COVERED: memorandum
	6. PERFORMING ORG. REPORT NUMBER:
7. AUTHOR(s): Shimon Edelman, and Tomaso Poggio	8. CONTRACT OR GRANT NUMBER(s): N00014-88-K-0164, IRI-8719392, DACA76-85-C-0010, N00014-85-K-0124
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Artificial Intelligence Laboratory, 545 Technology Square, Cambridge, MA 02139	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS:
11. CONTROLLING OFFICE NAME AND ADDRESS: Advanced Research Projects Agency, 1400 Wilson Blvd., Arlington, VA 22209	12. REPORT DATE: April 1990 13. NUMBER OF PAGES: 35
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office): Office of Naval Research, Information Systems, Arlington, VA 22217	15. SECURITY CLASS. (of this report): UNCLASSIFIED 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE:

16. DISTRIBUTION STATEMENT (of this Report)

Distribution is unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

None

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

object recognition
representation
nonlinear interpolation
Grandmother cells

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

We describe experiments with a versatile pictorial prototype based learning scheme for 3D object recognition. The GRBF scheme seems to be amenable to realization in biophysical hardware because the only kind of computation it involves can be effectively carried out by combining receptive fields. Furthermore, the scheme is computationally attractive because it brings together the old notion of a "grandmother" cell and the rigorous approximation methods of regularization and splines.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)