BRINGING THE GRANDMOTHER BACK INTO THE PICTURE: A MEMORY-BASED VIEW OF OBJECT RECOGNITION

SHIMON EDELMAN* and TOMASO POGGIO
Center for Biological Information Processing
Department of Brain and Cognitive Sciences, MIT
Cambridge, MA 02139, USA

We describe experiments with a versatile pictorial prototype-based learning scheme for 3-D object recognition. The Generalized Radial Basis Function (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.

Keywords: Artificial neural network, object recognition, vision, learning, radial basis functions, approximation.

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 conjuction 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.[3–6] Generally, recognition schemes of this type employ 3-D models of objects. Automatic learning of 3-D 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 3-D models acquired through user interaction (see, e.g. Ref. 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 view-

* Present address: Department of Applied Mathematics and Computer Science, The Weizmann Institute of Science, Rehovot 76100, Israel.

International Journal of Pattern Recognition and Artificial Intelligence Vol. 6 No. 1 (1992) 37–61

points. We base our approach on a recently proposed network scheme for the approximation of multivariate functions, by couching 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 (Fig. 1).

2. THEORETICAL BASIS

2.1. How Much Information is Necessary for Learning 3-D Structure?

Structure from motion theorems,[9,10] pioneered by Ullman,[11] indicate that full information about the 3-D 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 2-D views of an object is equivalent to having its 3-D structure specified.

2.2. Learning as Hypersurface Interpolation

This line of reasoning, together with properties of perspective 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 3-D objects. Furthermore, the application of the function that is specific for

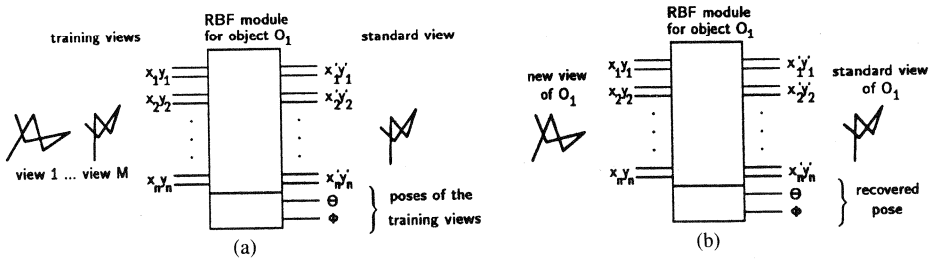


Fig. 1. Application of a general module for multivariate function approximation to the problem of recognizing a 3-D object from any of its perspective views. In (a), the module is trained to produce a vector representing the standard view of the object, given a set of examples of random perspective views 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 module and the standard view.

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.[13,17,18] The approximation of $f : R^n \rightarrow R$ is given by

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

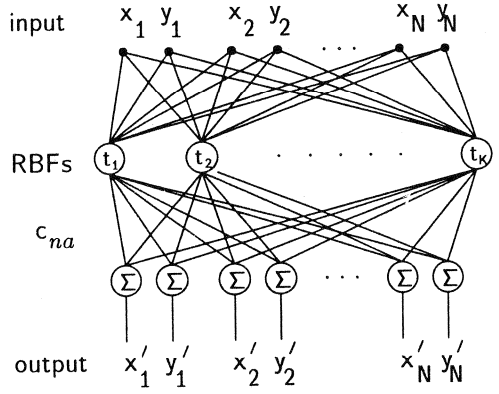
where the coefficients c_α and the centers $\mathbf{t}_\alpha$ are found during the learning stage and G is an appropriate basis function (see Refs. 13, 14), such as the Gaussian. If the function f is vector-valued, each component f_i is computed using Eq. 1 with the appropriate $c_{i\alpha}$, in which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

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

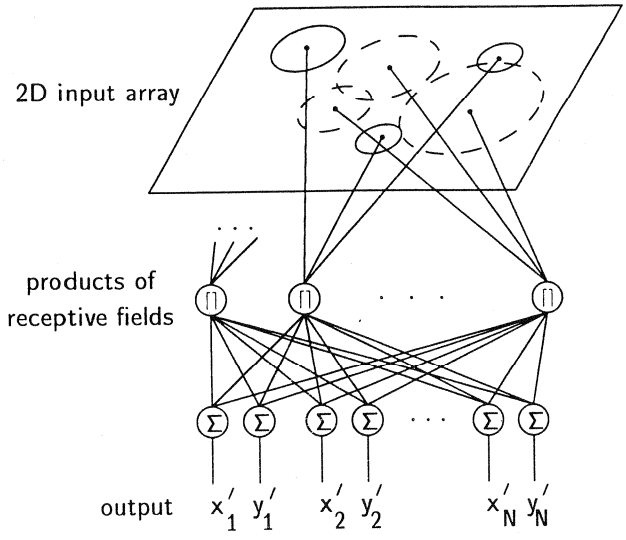
on the set of examples. In Eq. 2, P is usually a differential operator and λ is a positive real number, called the regularization parameter.[17] The radial function G is fully determined by the stabilizer P in Eq. 2 and therefore by the prior assumptions on the function to be approximated, such as its degree of smoothness.[13] P also determines whether a polynomial term of the form $\Sigma_i d_i p_i(\mathbf{x})$ should be added to the right-hand side of Eq. 1. In most of the experiments described in Sec. 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 Eq. 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 multi-dimensional 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 2-D Gaussians associated with the first radial basis function, which corresponds to the first view of the object. The dotted circles represent the 2-D



(a)



(b)

Fig. 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 2-D Gaussians associated with the first radial basis function, which corresponds to the first view of the object. The dashed circles represent the 2-D receptive fields that synthesize the Gaussian radial function associated with another view.

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.[a]

The weights **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 Eq. 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 Eq. 2. A further generalization may be achieved by using a weighted norm in Eq. 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, can be found in Refs. 13 and 14 (see also Ref. 19).

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 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 3-D graphics modeling system. The objects were 5-segment random wire frames[b] (Fig. 3). All the objects were positioned in such a manner that their centers of

[a] Implementing a multidimensional receptive field as a product of 2-D 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.

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

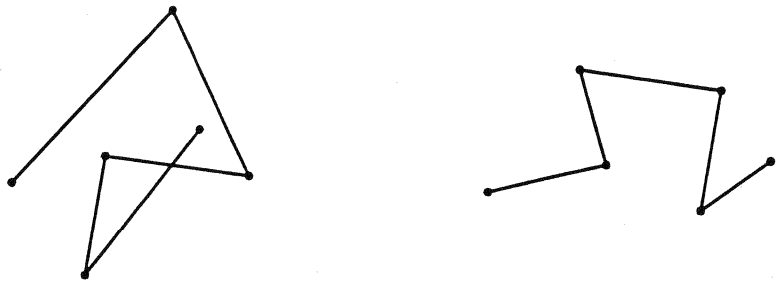


Fig. 3. Two examples of wire objects used in the experiments. The wires were created by a random walk in 3-D. They were encoded for training and subsequent recognition by projecting the vertices onto an imaging plane (under either orthographic or perspective projection). The resulting vector of x, y-coordinates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

mass coincided with the origin of the 3-D coordinate system defined by the modeling program. Different views of the objects were obtained by rotating the S-Geometry "camera" around the 3-D origin, so that it could assume any position specified by two viewpoint coordinates, θ and ϕ, corresponding to the latitude and the longtitude 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 2-D 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.

(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 3-D 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 2-D shape from being incorporated into the input representa-

tion. In particular, 3-D shape cues (obtained, e.g. through binocular stereo) can be used within the same framework depicted in Fig. 2.

3.3. Output Representation

As depicted in Fig. 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).[c]

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 ten 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 (Fig. 4). Ideally, the smallest error on a non-target object (call it $min_{\text{nontarget}}$) 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.[d]

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 Fig. 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 Fig. 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 $Y'_{\text{out}} = kY_{\text{out}}(ideal)$, where $k < 1$. To understand why that happens, it is

[c] We have also experimented with a scalar output representation; see Sec. 3.6.8.

[d] Standard statistical methods of parameter estimation and hypothesis testing may be used to translate the means and the standard deviations of $min_{\text{nontarget}}$, max_{target}, avg_{target} and $avg_{\text{nontarget}}$ into probabilities of Type I and Type II recognition errors (see e.g. Ref. 20). Since these methods involve table lookup of probability distributions, we did not use them on-line. Characteristically for our experiments, a ratio of $avg_{\text{nontarget}}$ to avg_{target} of 5.0 sufficed to impose a 0.001 upper bound on the probabilities of both Type I and Type II errors.

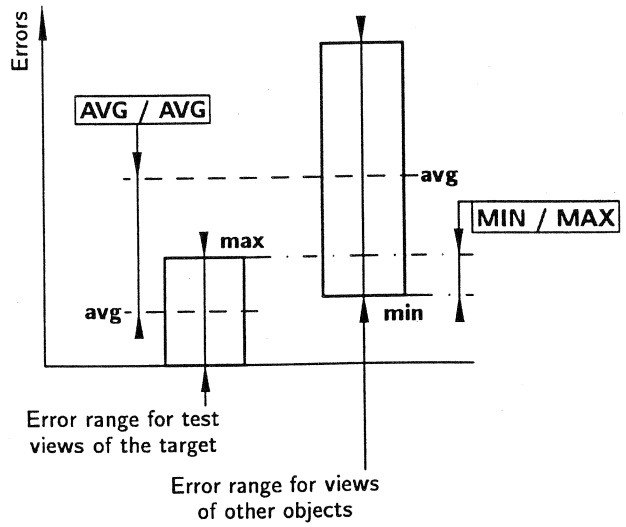
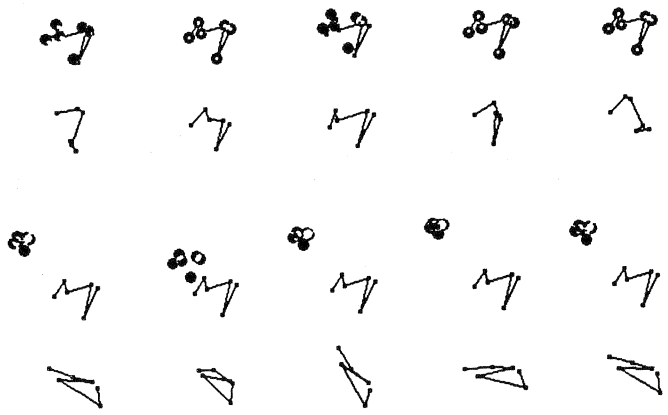
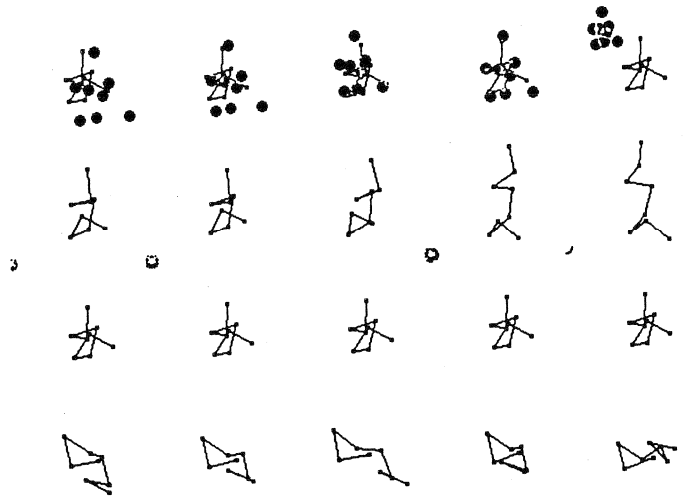


Fig. 4. Definitions of the AVG/AVG and the MIN/MAX performance criteria used throughout the paper. The error here is defined as the Euclidean distance between the standard view of the target and the actual output of the system (the smaller the error, the greater the likelihood that the input view belongs 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.



(a)

Fig. 5. Examples of the module's operation. (a) 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° to 90°. A naive fixed-step gradient descent (with a small number of steps) was used to obtain the optimal positions of the GRBF centers.



(b)

Fig. 5 Cont'd. (b) Within a smaller range of θ, $\phi \in [0°, 45°]$, 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.)

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 six-dimensional manifold that, to a first approximation, is a hyperplane in R^{2N} (see the appendix in Ref. 21). 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 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 was made equal to the number M of training views by letting the centers coincide with the views themselves. Consider Fig. 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

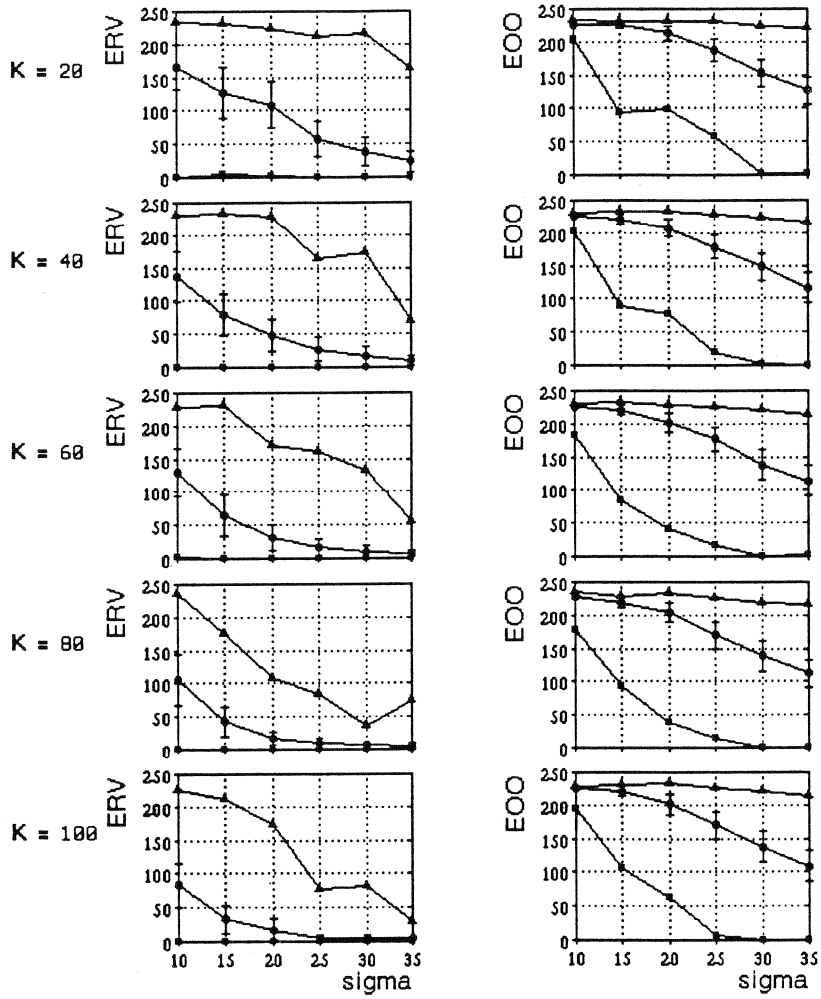


Fig. 6. Mean error ±SEM vs. size σ of the basis functions, for different number of centers K (number of training views $M = K$). Three measures of the error, MIN (lower curves), AVG (middle curves) and MAX (upper curves) are shown for two input sets: random views of the trained object (left) and views of nine different objects (right).

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 (Fig. 7 middle column), it is $\sigma = 25$, for $K = 100 = M$ (clearly, increasing the number of training views and RBF centers

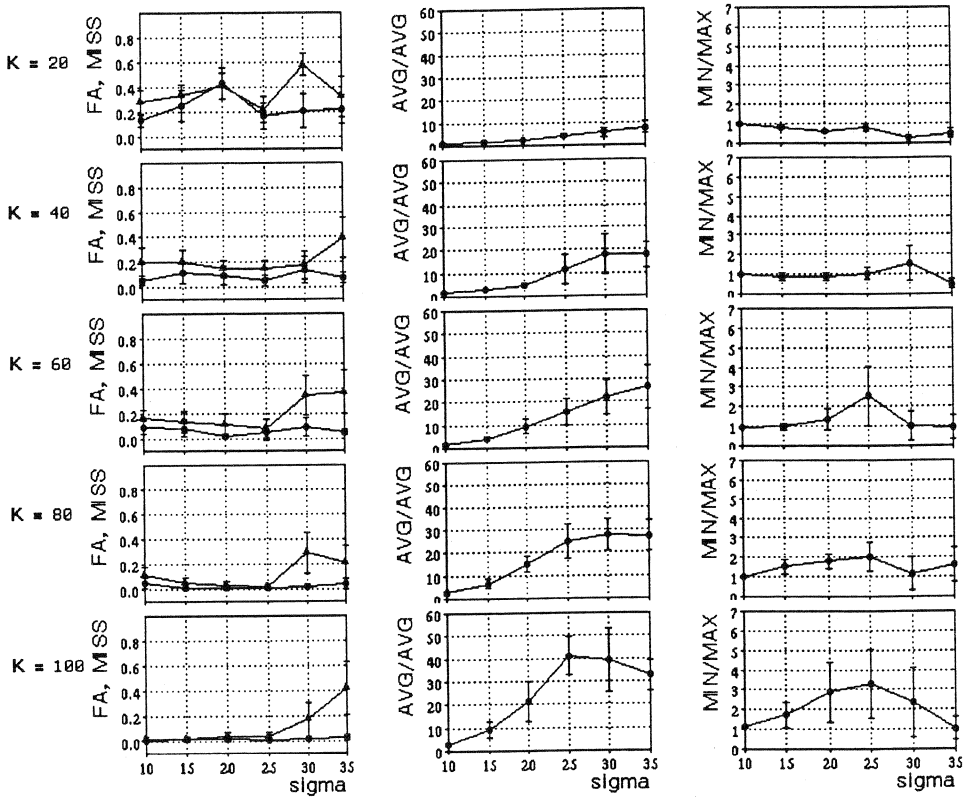


Fig. 7. Left: Type I or miss (lower curve) and Type II or false alarm (upper curve) error rates, vs. σ, by K. Middle: AVG/AVG performance index. Right: MIN/MAX performance index (see Fig. 4), vs. σ, by K.

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 $\sigma = 30$ (Fig. 7, right column). The left column of Fig. 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 Fig. 8 it can be seen that doubling the

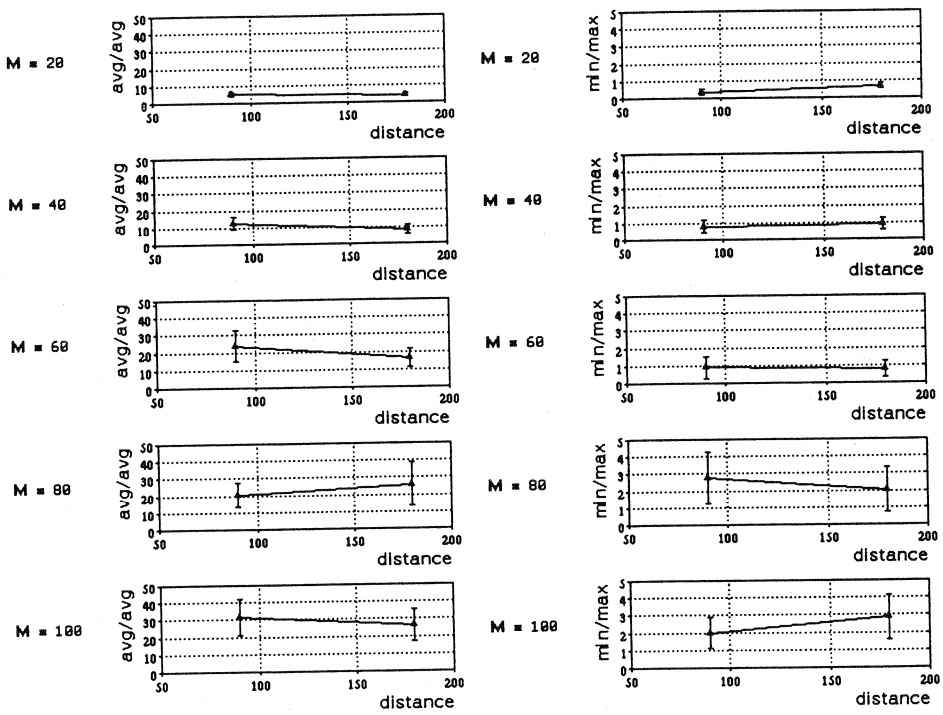


Fig. 8. Left: AVG/AVG performance vs. distance from the simulated camera, by number of training views M ($K = M$, $\sigma = 30.0$). The "near" distance was about seven times the apparent size of the wire objects. Right: MIN/MAX performance vs. distance from the simulated camera, by number of training views M ($K = M$, $\sigma = 30.0$).

distance from "near" to "far" made no significant difference in the performance.

A separate look at the false alarm and the miss rates (Fig. 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.

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°, 240°).

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 (Fig. 11). For $M = K = 40$ and $\sigma = 30$,

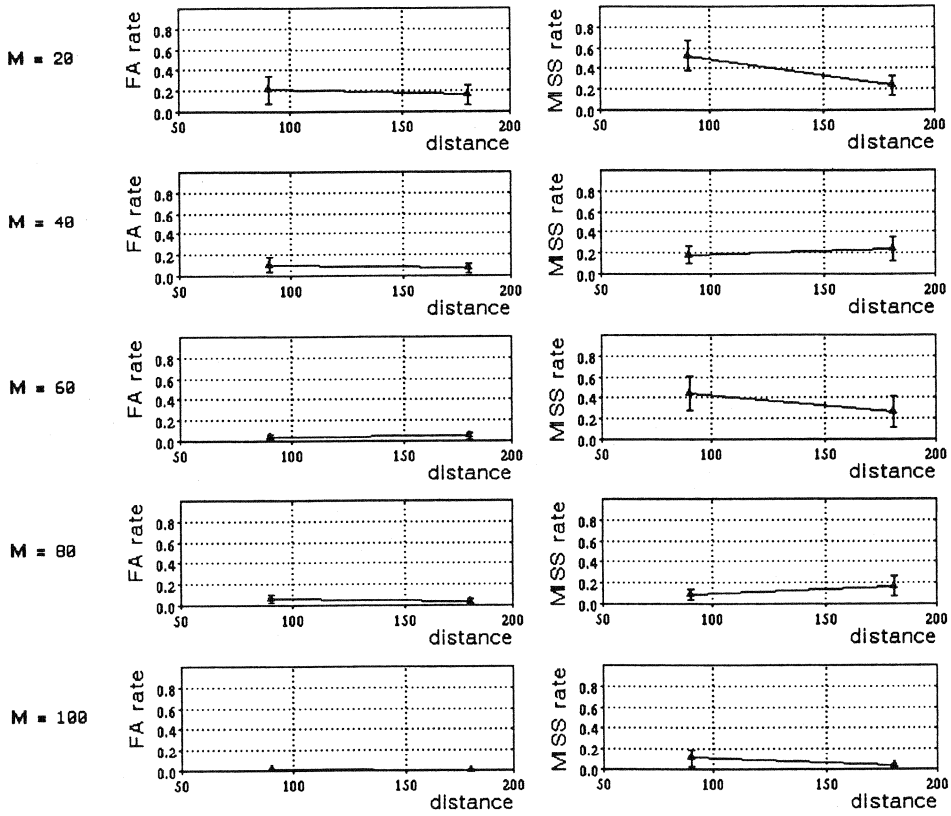


Fig. 9. False alarm (left) and miss (right) rates vs. distance from the simulated camera, by number of training views M ($K = M$, $\sigma = 30.0$).

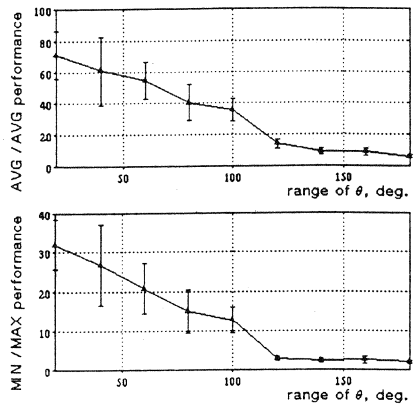


Fig. 10. AVG/AVG and MIN/MAX performance vs. the range of the viewpoint coordinates θ, ϕ (the objects are a cube and an octahedron, $M = K = 40$, $\sigma = 30.0$, and the error bars are standard deviations over ten sets of random training and testing views). Here and in the next figure, $\phi_{\max}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

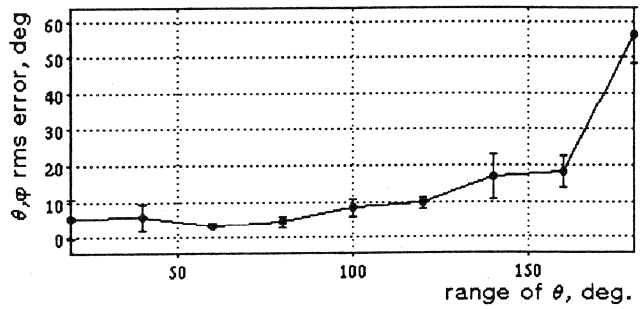


Fig. 11. Errors in the viewpoint coordinates θ, ϕ recovered by the module vs. the range of the viewpoint coordinates ($M = K = 40$, $\sigma = 30.0$).

the mean square error of the recovered orientation stays below 10° for $\theta < 120°$, $\phi < 240°$, rising to about 60° 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 (Fig. 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. The more vertices are used, the more information there is for the recognizer to go by, until cross-talk sets in (which will happen if the size of the basis function 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 Fig. 13, which shows a superimposed 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 repeatedly trained it on views each of whose constituent features had a fixed probability of being

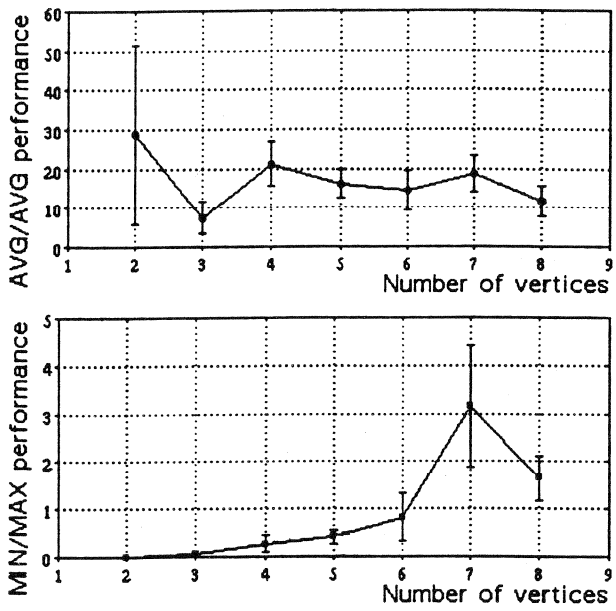


Fig. 12. AVG/AVG and MIN/MAX indices vs. the number of vertices used in training (the data for number of vertices from 2 to 6 are for six-vertex random wire objects; the data for number of vertices 7 and 8 are for eight-vertex wires; $M = K = 60$, $\sigma = 30.0$).

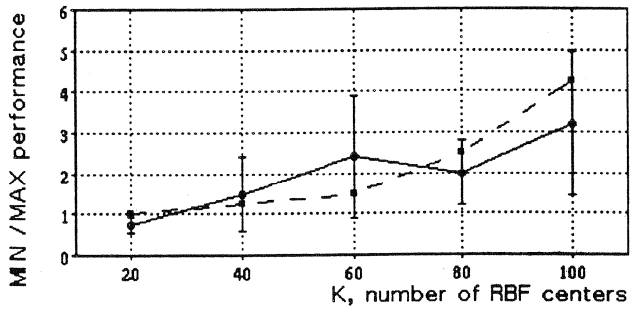


Fig. 13. MIN/MAX performance for two types of input encoding: vertex coodinates (solid line) and angles formed by successive pairs of segments (dashed line; data for six-vertex random wire objects, σ chosen optimal for each encoding, $M = K$).

"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 Fig. 14.[e] It appears from the figure that satisfactory 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 occluded. Occlusion has had a somewhat stronger effect on the learning of eight-vertex wires (Fig. 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 **W**.

3.6.8. Scalar vs. vector output

If a compact output representation is required, it is possible to train the recogniiton module to produce a scalar ouptut, 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 Fig. 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 **C** matrix).

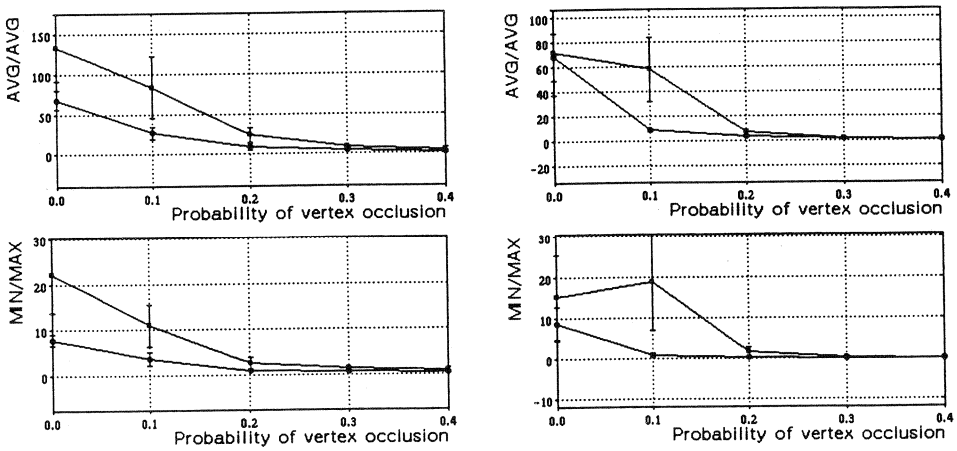


Fig. 14. AVG/AVG and MIN/MAX indices vs. the probability of any given vertex being occluded (left: six-vertex random wire objects; right: eight-vertex objects). θ, ϕ were confined to one half of the viewing sphere; $K = M = 50$ (lower curves), $K = M = 100$ (upper curves); $\sigma = 30.0$.

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

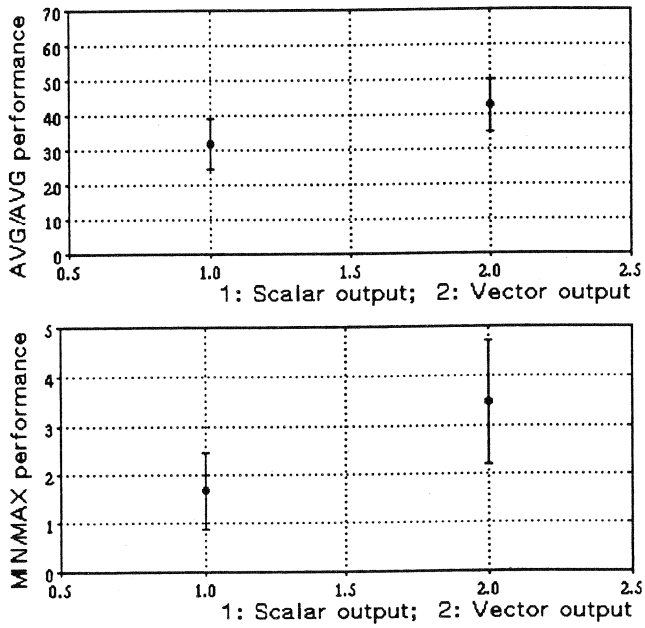


Fig. 15. AVG/AVG and MIN/MAX performance for scalar and vector output (data for six-vertex random wire objects, $\sigma = 30$, $M = K = 80$; error bars show standard deviation computed over ten objects). In the first case, the network is trained to output 1 when it is shown views of the target. In the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

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 $\mathbf{W}$ (Eq. 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 Sec. 2), and an identity matrix was used as $\mathbf{W}$.

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 Ref. 14.[f] 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 Table 1. Note that the best effects were achieved by a combined adjustment of $\mathbf{C}$, $\mathbf{t}_\alpha$ and $\mathbf{W}$

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

Table 1. Improvement ratios in the AVG/AVG and the MIN/MAX performance measures caused by 100 steps of gradient descent, with the step size $\omega = 10^{-3}$ (six-vertex wires, $M = 40$, $K = 10$, $\sigma = 30.0$, *angles* encoding, perspective projection, full range of viewpoint coordinates). The numbers are exponentials of the averages of logarithms of the appropriate measures over ten trials. Training was carried out on one object and testing on five other objects.

Allowed to change	AVG/AVG improvement	MIN/MAX improvement
$\mathbf{C}$, $\mathbf{t}_\alpha$, $\mathbf{W}$	1.59	1.24
$\mathbf{C}$, $\mathbf{t}_\alpha$	1.45	0.58
$\mathbf{C}$, $\mathbf{W}$	2.30	0.74

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 Fig. 5.

3.7. Comparison with Related Schemes

At this point it is natural to ask whether other, simpler network schemes can perform the recognition task defined in this report as well as 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).

3.7.1. Linear associative memory

The GRBF network of Fig. 2 can be converted into a linear associator by omitting the middle layer (the basis units; the full GRBF scheme has a linear part connected in parallel with the network of Fig. 2 at all times[13]). The association function in this case is realized by the matrix $\mathbf{C}$. Let $\mathbf{V}$ be the matrix whose rows are the training views and $\mathbf{Y}$ the matrix whose rows are the vectors to be associated with the rows of $\mathbf{V}$ (in our case, all of these are the same vector, e.g. the first training view). $\mathbf{C}$ is then found by solving the equation $\mathbf{Y} = \mathbf{CV}$, that is, $\mathbf{C} = \mathbf{YV}^+$ (pseudoinverse is needed, since generally $\mathbf{V}$ is not square).

The performance of the linear associative memory (Fig. 16) was considerably worse than that of the GRBF module. The main difference is in the MIN/MAX measure, which failed to exceed 1.0 even with 100 training views. A closer look revealed that this was due to the tendency of the linear associator to overgeneralize.

3.7.2. Nearest neighbor scheme

Another recognition scheme that we tested, the nearest neighbor (NN) classifier, operated as follows. In training, it stored all the views of the target object presented to it. To decide whether a new view belonged to the target object, the NN classifier found among the stored views the one with the shortest Euclidean distance from the input view. This distance, which could be interpreted as the inverse of a classification confidence measure, was then returned as the classification error. This simple recognition scheme performed surprisingly well, with the MIN/MAX measure exceed-

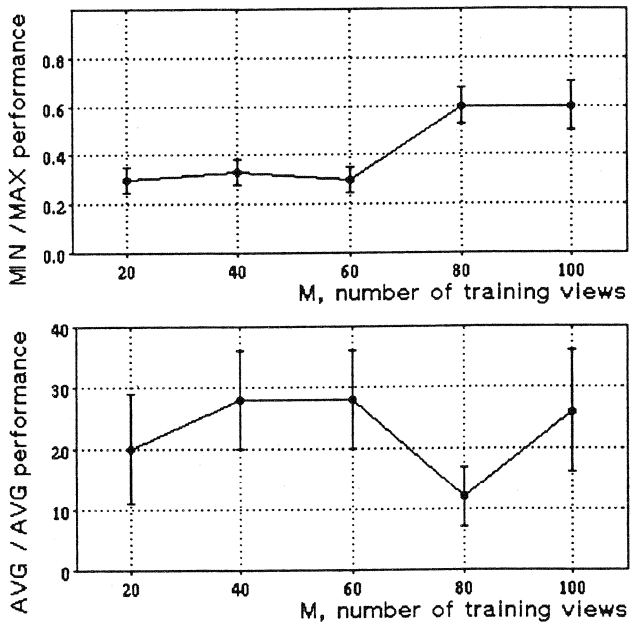


Fig. 16. AVG/AVG and MIN/MAX indices vs. the number of training views for the linear associative scheme (six-vertex random wire objects).

ing 1.0 with just 100 views (Fig. 17). As the number of stored views grows, the performance of the NN classifier is expected to improve, asymptotically matching that of the RBF scheme. Any comparison between the two schemes should include, therefore, the amount of memory they use.

3.7.3. Nearest neighbor without correspondence

The computation of the Euclidean distance between the input and each of the stored views in the NN scheme requires that the correspondence between the features of the objects be known. 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 2-D correlations between each of the stored training views (represented in this case as 2-D arrays rather than as 1-D 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 2-D 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 Fig. 18.

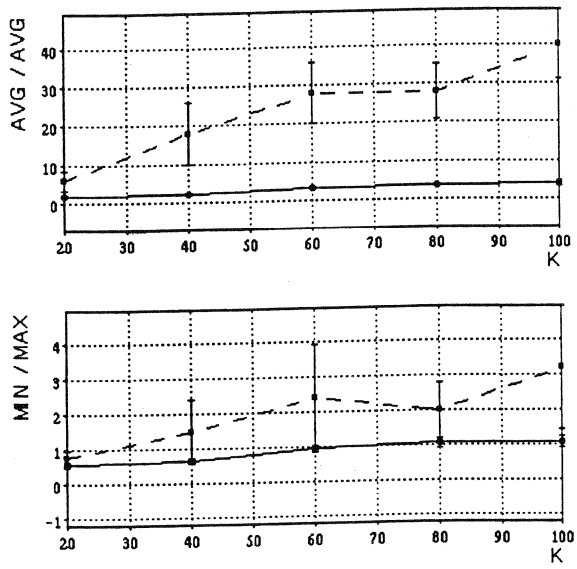


Fig. 17. AVG/AVG and MIN/MAX indices vs. the number of remembered wires for the nearest-neighbor method that uses correspondence information (six-vertex random wire objects). For comparison purposes, the performance of the RBF scheme with $M = K$ is also shown (dashed curve).

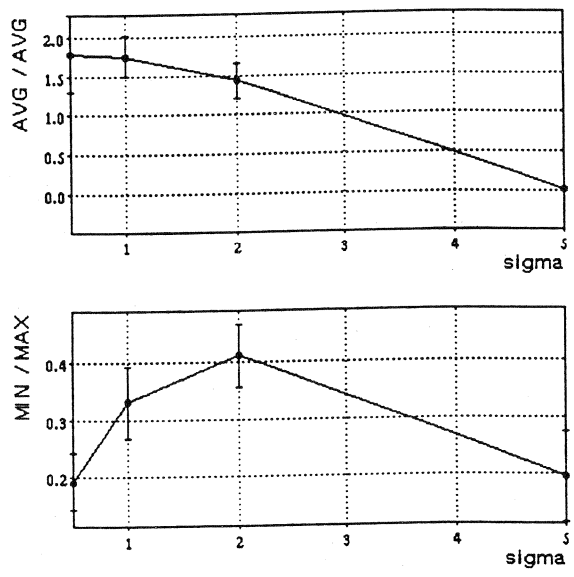


Fig. 18. AVG/AVG and MIN/MAX indices vs. the width σ of the Gaussian blurring mask (see Sec. 3.7.3) for the nearest-neighbor method that uses 2-D correlation and 2-D array representation of views, instead of correspondence information and 1-D vector representation of views (six-vertex random wire objects; the number of remembered views is 80).

4. DISCUSSION

The reconstructionist dogma of computational vision appears recently to have fallen upon hard times. A standard version of this dogma holds that the ultimate goal of a visual recognition system is the formation of object representations that make explicit the relevant 3-D structure, just as a toy airplane makes explicit the relative size and position of the wings and the fuselage in the real airplane. This view of recognition considers the 2-D image bottleneck that necessarily intervenes between the distal object and its percept as 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 Ref. 23 for a review), inverse optics all the way to the top no longer seems to be the most promising approach to recognition.[g]

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 Refs. 22, 24). Moreover, adopting the present approach to recognition does not mean giving up the use of information beyond 2-D shape, such as color, texture and depth, which can be incorporated naturally into the GRBF module (see Sec. 3.2). 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 (Fig. 2(b)). 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

[g] 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.

the receptive field as e^{-r^2/σ^2}). The network's output (see Eq. 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 Ref. 25 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 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).[h]

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 Fig. 1 and Sec. 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. Ref. 26).

4.2. Psychophysical Aspects

Another aspect of the biological plausibility of our approach to recognition is provided by psychological studies. Although the GRBF-based recognition system can hardly be considered a complete model of human object recognition, some of its functional characteristics parallel those of human performance. 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.[27] 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.[28] Finally, a model mathematically related to GRBFs

[h] Marr (Ref. 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.

has been shown to replicate central features of the time course of object recognition, including effects of unsupervised learning (practice).[24]

Assuming that a scheme resembling GRBF or some 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 when viewpoint-sensitive features (such as wire-frame vertex locations) are predominant in the input, 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. Thus, experimental evidence of such dependency[29–33] supports the prototype interpolation approach that is central to the GRBF model.

5. SUMMARY

We have described experiments with a versatile pictorial prototype-based learning scheme for 3-D 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.

ACKNOWLEDGEMENTS

We thank Federico Girosi and Shimon Ullman for useful comments. This research was supported in part by ONR (Cognitive and Neural Sciences Division), by the Sloan Foundation, and by DARPA. T. Poggio is supported by the Uncas and Helen Whitaker Chair at Whitaker College, MIT. S. Edelman was supported by a Chaim Weizmann Postdoctoral Fellowship from the Weizmann Institute of Science.

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** (1988) 436–440.
3. M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography", *Commun. ACM* **24** (1981) 381–395.
4. D. W. Thompson and J. L. Mundy, "Three-dimensional model matching from an unconstrained viewpoint", *Proc. IEEE Conf. on Robotics and Automation*, Raleigh, NC, 1987, pp. 208–220.
5. S. Ullman, "Aligning pictorial descriptions: an approach to object recognition", *Cognition* **32** (1989) 193–254.
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 Trans. Pattern Anal. Mach. Intell.* **9** (1987) 469–482.

8. T. J. Fan, G. Medioni and R. Nevatia, "Recognizing 3-D objects using surface descriptions", *Proc. 2nd Int. Conf. on Computer Vision*, Tarpon Springs, FL, 1988, IEEE, Washington, D.C., pp. 474–481.
9. R. Tsai and T. Huang, "Uniqueness and estimation of three dimensional motion parameters of rigid objects with curved surfaces", *IEEE Trans. Pattern Anal. Mach. Intell.* **6** (1984) 13–27.
10. H. C. Longuet-Higgins, "A computer algorithm for reconstructing a scene from two projections", *Nature* **293** (1981) 133–135.
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", *Biol. Cybern.* **32** (1979) 211–217.
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** (1990) 978–982.
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** (1985) 314–319.
17. M. J. D. Powell, "Radial basis functions for multivariable interpolation: a review", *Algorithms for Approximation*, eds. J. C. Mason and M. G. Cox, Clarendon Press, Oxford, 1987.
18. D. S. Broomhead and D. Lowe, "Multivariable functional interpolation and adaptive networks", *Complex Systems* **2** (1988) 321–355.
19. J. Moody and C. Darken, "Fast learning in networks of locally tuned processing units", *Neural Comput.* **1** (1989) 281–289.
20. W. Mendenhall and T. Sincich, *Statistics for the Engineering and Computer Sciences*, Macmillan, London, 1988.
21. S. Edelman and T. Poggio, "Bringing the Grandmother back into the picture: a memory-based view of object recognition", A. I. Memo No. 1181, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, 1990.
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. S. Edelman and D. Weinshall, "Computational vision: a critical review", A. I. Memo No. 1158, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Oct. 1989.
24. S. Edelman and D. Weinshall, "A self-organizing multiple-view representation of 3-D objects", *Biol. Cybern.* **64** (1991) 209–219.
25. D. I. Perrett, A. J. Mistlin and A. J. Chitty, "Visual neurones responsive to faces", *Trends in Neurosciences* **10** (1989) 358–364.
26. S. Zeki and S. Shipp, "The functional logic of cortical connections", *Nature* **335** (1988) 311–317.
27. G. W. Humphreys and P. Quinlan, "Normal and pathological processes in visual object constancy", *Visual Object Processing: A Cognitive Neuropsychological Approach*, eds. G. W. Humphreys and M. J. Riddoch, Erlbaum, Hillsdale, NJ, 1987, pp. 43–106.
28. S. E. Palmer, E. Rosch and P. Chase, "Canonical perspective and the perception of objects", *Attention and Performance IX*, eds. J. Long and A. Baddeley, Erlbaum, Hillsdale, NJ, 1981, pp. 135–151.
29. I. Rock and J. DiVita, "A case of viewer-centered object perception", *Cogn. Psychol.* **19** (1987) 280–293.