Natural Basis Functions and Topographic Memory for Face Recognition*

Rajesh P.N. Rao and Dana H. Ballard
Department of Computer Science
University of Rochester
Rochester, NY 14627, USA
{rao,dana}@cs.rochester.edu

Abstract

Recent work regarding the statistics of natural images has revealed that the dominant eigenvectors of arbitrary natural images closely approximate various oriented derivative-of-Gaussian functions; these functions have also been shown to provide the best fit to the receptive field profiles of cells in the primate striate cortex. We propose a scheme for expression-invariant face recognition that employs a fixed set of these "natural" basis functions to generate multiscale iconic representations of human faces. Using a fixed set of basis functions obviates the need for recomputing eigenvectors (a step that was necessary in some previous approaches employing principal component analysis (PCA) for recognition) while at the same time retaining the redundancy-reducing properties of PCA. A face is represented by a set of iconic representations automatically extracted from an input image. The description thus obtained is stored in a topographically-organized sparse distributed memory that is based on a model of human long-term memory first proposed by Kanerva. We describe experimental results for an implementation of the method on a pipeline image processor that is capable of achieving near real-time recognition by exploiting the processor's frame-rate convolution capability for indexing purposes.

1 Introduction

The problem of object recognition has been a central subject in the field of computer vision. An especially interesting albeit difficult subproblem is that of recognizing human faces. In addition to the difficulties posed by changing viewing conditions, computational methods for face recognition have had to confront the fact that faces are complex non-rigid stimuli that defy easy geometric characterizations and form a dense cluster in the multidimensional space of input images. One of the most important issues in face recognition has therefore been the representation of faces. Early schemes for face recognition utilized geometrical representations; prominent features such as eyes, nose, mouth, and chin were detected and geometrical models of faces given by feature vectors whose dimensions, for instance, denoted the relative positions of the facial features were used for the purposes of recognition [Bledsoe, 1966; Kanade, 1973]. Recently, researchers have reported successful results using *photometric representations* i.e. representations that are computed directly from the intensity values of the input image. Some prominent examples include face representations based on biologically-motivated Gabor filter "jets" [Buhmann *et* al., 1990], randomly placed zeroth-order Gaussian kernels [Edelman *et* a/., 1992], isodensity maps [Nakamura *et* a/., 1991], and principal component analysis (PCA) [Turk and Pentland, 1991; Pentland *et al.,* 1994].

This paper explores the use of an iconic representation of human faces that exploits the dimensionality-reducing properties of PCA. However, unlike previous approaches employing PCA for recognition [Turk and Pentland, 1991; Murase and Nayar, 1995], our approach uses a *fixed* set of basis functions that are *learned* during an initial "development" phase; the costly and time consuming step of having to recompute basis functions when new faces (or other objects) are encountered is thereby avoided. In addition, the basis functions used to generate face representations are based on PCA of *localized natural image patches* at *multiple scales* rather than PCA of entire face images at a single scale; the localized nature of the representation helps to make it tolerant to minor changes in facial expressions and partial occlusions while the multiscale structure allows strategies for scale invariance.

The iconic face representations are formed from *n*-dimensional photometric feature vectors comprised of the responses of m derivative of Gaussian basis functions at a range of orientations, each at *k* scales (n = *mk);* for the experiments in the paper, nine basis filters at five scales were used generating a forty-five element response vector characterizing the local image region at a point of interest. We have previously shown these iconic feature vectors to be useful for active vision [Rao and Ballard, 1995a], visuomotor learning [Rao and Ballard, 1995b], and general object indexing [Rao and Ballard,

*This work is supported by NSF research grant no. CDA-8822724, NIH/PHS research grant no. 1 R24 RR06853, and a grant from the Human Science Frontiers Program.

1995c]. Here, we show that such a representation may be used for the difficult problem of expression-invariant face recognition as well.

A face is represented by a collection of iconic feature vectors automatically extracted from specific locations in the input image (Section 3). A topographically-organized sparse distributed memory is used to learn the association between the appearance of a face as given by its feature vectors and the identity of the face (Section 4). Implementation of the recognition scheme is achieved using a Datacube MV200 pipeline image processor for both real-time visual preprocessing as well as indexing into the face database, utilizing frame-rate convolutions for distance computations (Section 5). We present preliminary results on the performance of the method on a face database of 140 images from 20 different persons exhibiting a range of facial expressions; a recognition rate of 93.3% was achieved by the method when a set of 33 points were used for characterizing a face.

2 Natural Basis Functions

Images of natural scenes, unlike random collections of pixels, are characterized by a high degree of statistical regularity. For instance, pixel values in a given neighborhood tend to be highly correlated owing to the morphological consistency of objects. Thus, a pixel-wise representation of objects obtained from a camera is highly redundant and some form of redundancy reduction is desirable.

2.1 Redundancy Reduction via Principal Component Analysis

The optimal linear method (in the least mean squared error sense) for reducing redundancy is the Karhunen-Loéve transform or eigenvector expansion via Principal Component Analysis (PCA). PCA generates a set of orthogonal axes of projections known as eigenvectors or *principal components* of the input data distribution in the order of decreasing variance. The eigenvectors form a set of orthogonal basis functions for representing the input. In particular, consider a set of input images of size $N \times N$ represented as N^2-dimensional vectors $\vec{I}_1, \ldots, \vec{I}_n$ and the corresponding mean-centered set of vectors $\vec{I}_1, \ldots, \vec{I}_n$ obtained by subtracting each pixel value from the mean value for that pixel over all input images:

$$\vec{I}_i = \vec{I}_i - \frac{1}{n}\sum_{j=1}^{n} \vec{I}_j, \quad i = 1, \ldots, n. \tag{1}$$

PCA achieves decorrelation by extracting n N^2-dimensional eigenvectors $\vec{e}_i$ such that the variance of projections along the direction $\vec{e}_i$

$$\lambda_i = \frac{1}{n}\sum_{j=1}^{n} (\vec{e}_i^T \vec{I}_j)^2 \tag{2}$$

is maximized subject to the orthonormality condition

$$\vec{e}_j^T \vec{e}_k = \begin{cases} 1 & \text{if } j = k \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

In principle, all $n < N^2$ eigenvectors are needed in order to completely represent the input image set but due to the statistics of natural images, it is usually the case that only a small number m of eigenvectors *(m << n)* account for almost all of the variance in the input data. Thus, by using only the first *m* dominant eigenvectors as basis functions (or orthogonal axes) for projecting new inputs, considerable computational savings can be achieved.

Turk and Pentland [I99I] used PCA to synthesize the eigenvectors ("eigenfaces") of a training set of face images; they achieve recognition by using a template-matching strategy with the vectors obtained by projecting new face images along a small number of eigenfaces. Murase and Nayar [1995] applied PCA to the problem of object recognition and pose estimation; they represent objects as manifolds in the low-dimensional subspace ("eigenspace") formed by the dominant eigenvectors of a set of training images and achieve recognition by finding the manifold that is closest to the projection of an input image in the eigenspace formed by all objects.

2.2 Unsupervised Learning of Basis Functions

The methods of [Turk and Pentland, 1991] and [Murase and Nayar, 1995] both require recomputation of the eigenvectors when new faces/objects are encountered. It is therefore natural to ask what the results of PCA would be *if one were to take the above process to its limit* i.e. to perform PCA on a set $J\backslash, \ldots, J_n$ of arbitrary natural images containing a wide variety of natural and man-made stimuli. Recently, Hancock et al. [1992] used a neural network introduced by Sanger [1989] to extract the first few eigenvectors of an ensemble of natural images. They discovered that the eigenvectors were very close approximations of different oriented derivative-of-Gaussian operators.

We employed Sanger's PCA network to ascertain whether the results of Hancock et al. remained true for collections of images containing equal proportions of natural and man-made stimuli. In particular, we used 32 x 32 Gaussian-windowed image patches obtained by scanning across a number of arbitrary images of natural scenes (Figure 1 (a)). Suppose / represents an input mean-centered image patch and W_j represents the weight vector from the input layer (which in our case consists of 1024 units) to the output unit *j*. Sanger's PCA network uses linear output units i.e.

$$y_j = \vec{W}_j^T \vec{I} \tag{4}$$

When presented with input $\vec{I}$, the network adapts its weight vectors according to a *generalized Hebbian Learning rule*:

$$\vec{W}_j' = \vec{W}_j + \gamma y_j (\vec{I} - \sum_{k=1}^{j} y_k \vec{W}_k) \tag{5}$$

where $\gamma > 0$ is a gain term whose value is decreased gradually as training progresses. Figure 1 (b) shows the eigenvectors to which the weight vectors $\vec{W}_j$ converged

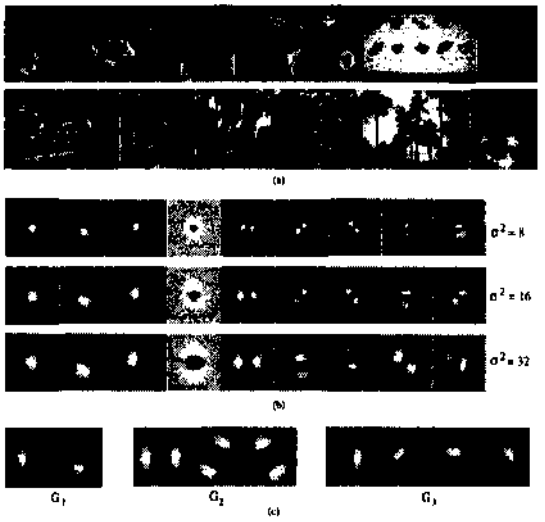

Figure 1: (a) Twelve of the 20 images that we used for training Sanger's PCA network. (b) First nine dominant eigenvectors that the weights of the network converged to, shown here for different scales (σ) of the Gaussian window (intensity is proportional to magnitude). (c) The Gaussian derivative basis functions of up to the third-order used in our iconic representations. The first few dominant eigenvectors of natural images shown in (b) closely resemble these analytically derived function profiles.

after 12000 presentations. These nine eigenvectors alone account for as much as 83% of the input variance. It is clear that regardless of the scale of analysis, the eigenvectors closely approximate different oriented derivative-of-Gaussian operators.

In summary, the derivative-of-Gaussian filters are well-suited for use as *natural basis functions* for general-purpose visual recognition because: (a) they are obtained as result of applying PCA to arbitrary collections of images containing diverse elementary features from natural as well as man-made structures rather than just the images of particular objects or faces. In addition, correlation filters generated via PCA have been shown to maximize signal-to-noise ratio and yield much sharper correlation peaks than traditional raw image cross-correlation techniques [Kumar *et al*., 1982]; (b) they form the class of *real-valued* functions that simultaneously minimize the product of the standard deviation of the spatial position sensitivity and spatial frequency sensitivity ([Gabor, 1946] p. 441)[1]; and (c) they are endorsed by neurobiological studies [Young, 1985] which show that the different order derivative-of-Gaussian functions provide the best fit to primate cortical receptive field profiles among the different functions suggested in the literature.

1The class of *complex-valued* functions that minimize this conjoint localization metric are the well-known Gabor elementary functions [Gabor, 1946].

3 Iconic Representations of Faces

The iconic representations used in our recognition scheme are based on the natural basis functions mentioned in the previous section. The exact number and type of Gaussian derivative basis functions used is motivated by the need to make the representations invariant to rotations in the image plane. This invariance can be achieved by exploiting the property of *steerability* [Freeman and Adelson, 1991] and using a minimal basis set of two first-order directional derivatives at 0° and 90°, three second-order derivatives at 0°, 60° and 120°, and four third-order derivatives oriented at 0°, 45°, 90°, and 135° (Figure 1 (c)). We omit the zeroth order to reduce illumination dependence and do not use higher orders since variance of the higher-order filters can be expected to approach that of image noise as suggested by the results of PCA. The use of the non-orthogonal oriented filters also obviates the use of mixed derivatives (some of which were obtained in Figure 1 (b)) since the other oriented filters yield a complete basis.

3.1 Representing Image Regions

The current implementation uses nine Gaussian directional derivatives denoted by

$$G_n^{\theta_n}, n = 1, 2, 3, \theta_n = 0, \ldots, k\pi/(n+1), k = 1, \ldots, n \quad (6)$$

where n denotes the order of the filter and θ_n refers to the preferred orientation of the filter. The response of an image patch I centered at (x_o, y_o) to a particular basis filter G_i^j can be obtained by convolving the image patch with the filter :

$$r_{i,j}(x_0, y_0) = \iint G_i^{\theta_j}(x_0 - x, y_0 - y) I(x, y) dx\, dy \quad (7)$$

The iconic representation for the local image patch centered at (x_0, y_0) is formed by combining into a single high-dimensional vector the responses from the nine basis filters, each (in the current implementation) at five different scales:

$$\vec{r}(x_0, y_0) = \left[r_{i,j,s}(x_0, y_0) \right] \quad (8)$$

where i = 1,2,3 denotes the order of the filter, j = 1, . . . ,i -I- 1 denotes the different filters per order, and $s = s_{min}, \ldots, s_{max}$ denotes the different scales (as given by the levels of a low-pass filtered image pyramid). The use of multiple scales increases the perspicuity of the representation and allows interpolation strategies for scale invariance (see [Rao and Ballard, 1995a] for more details). In addition, the high-dimensionality of the vectors makes them remarkably robust to noise due to the *orthogonality* inherent in high-dimensional spaces: given any vector, most of the other vectors in the space tend to be relatively uncorrelated with the given vector [Rao and Ballard, 1995c].

The iconic representations can be made invariant to rotations in the image plane (for a fixed scale) by exploiting the property of steerability [Freeman and Adelson, 1991]. The current orientation is computed using the first-order responses as:

$$\theta = atan2(r_{1,1,s_{max}}, r_{1,2,s_{max}}) \quad (9)$$

(a) (b)

Figure 2: **Extracting Iconic Face Representations.** (a) A face is represented by response vectors from the small number of points lying on the intersections of the radial lines with the concentric circles of exponentially increasing radii centered on the approximate centroid of the face. (b) shows the receptive fields of the filter kernels at the centroid across the five different scales used in the current implementation.

and the entire set of filter responses is rotated to a canonical orientation (in this case, horizontal) by using linear combinations of the original responses:

$$r'_{i,j,s} = \sum_{j'=1}^{i+1} r_{i,j',s} k_{j'i}(\theta) \qquad (10)$$

where $i = 1, 2, 3; j = 1, \ldots, i+1; s = s_{min}, \ldots, s_{max}$ and $k_{j'i}$ are interpolation functions given by :

$$k_{j'1}(\theta) = \frac{1}{2}\Big[2\cos(\theta - (j'-1)\pi/2)\Big], j' = 1, 2 \qquad (11)$$

$$k_{j'2}(\theta) = \frac{1}{3}\Big[1 + 2\cos(2(\theta - (j'-1)\pi/3))\Big], j' = 1, 2, 3 \qquad (12)$$

and

$$k_{j'3}(\theta) = \frac{1}{4}\Big[2\cos(\theta-(j'-1)\pi/4)+2\cos(3(\theta-(j'-1)\pi/4))\Big] \qquad (13)$$

with $j' = 1, \ldots, 4$. Note that this normalization procedure does not apply to rotations in *3D*. Modest rotations in depth can be handled as noise by the representation but larger 3D rotations require the use of responses from multiple views as we have shown in [Rao and Ballard, 1995c]. This view-based approach is similar in spirit to the one used by Beymer [1993] (see also [Pentland *et al*., 1994]).

3.2 Representing Faces

The response vectors described in the previous section serve as iconic descriptions of individual image regions. In order to represent a given model face with a set of such vectors, the problem of selecting suitable points within a face from which model response vectors can be extracted must be addressed. In our current implementation, we apply the following simple strategy after the approximate boundary of the face is determined (by using, for instance, stereo and a technique such as *zero disparity filtering* [Coombs, 1992]):

> A face is represented by response vectors from the centroid of the face and each of the points lying on the intersections of radial lines with concentric circles of exponentially increasing radii centered on the centroid.

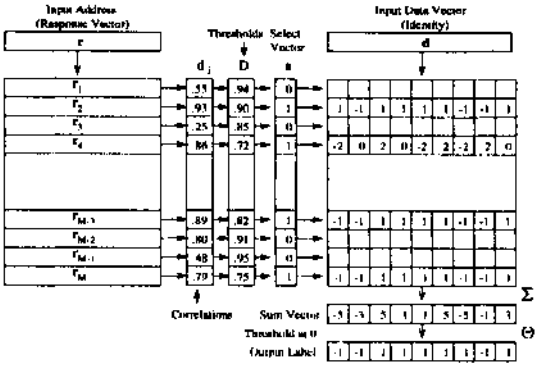

Figure 3: **Sparse Distributed Memory.** The diagram illustrates the operation of the modified sparse distributed memory model for learning associations between the visual appearance of a face and the identity of the face.

This strategy is illustrated in Figure 2. Only points lying within the approximate face boundary are used. This method ensures a dense representation of the "foveal" region near the centroid while at the same time including information from the "parafoveal" regions as well. Note that this strategy is a form of representation by parts. Also, due to the tendency of nearby points to be highly correlated, the above method *does not* require the centroid and the other points of an input face to be precisely registered with the corresponding points from a model face; an approximate registration suffices in most cases.

4 Topographic Sparse Distributed Memory

For the task of face recognition using the iconic representations discussed above, two requirements need to be met: (a) a mode of long-term storage for the representations of model faces, and (b) a method for learning the association between the representation of a face and the identity of the face. Both these requirements are met by using a modified form of sparse distributed memory (SDM), first proposed by Kanerva [1988; 1993] as a model of human long-term memory.

4.1 Operation of the SDM

The organization of sparse distributed memory (Figure 3) is similar to that of conventional random access memory. There exists an array of data storage locations, each identified by a number (the address of the location). However, the address vectors are usually high-dimensional and hence only a *sparse* subset of the address space is used for identifying data locations. Suppose the SDM contains *M* storage locations where the address vectors are p-ary and n-dimensional, and the data vectors are derived from the set $\{-1, 1\}^k$. For our current purposes, the feature vectors describing faces correspond to the addresses of the SDM while the data being associated with the feature vectors represent the identities of the faces, each person being defined by a small interval of possible values. Specific values of the

data vector may be interpreted as corresponding to a particular facial expression of a person.

Organization of the Address Space

Kanerva suggests randomly picking *M* unique vectors A_i from the p^n possible address vectors for addressing each of the *M* data storage locations of the memory. However, in our case, the set of response vectors will be clustered in many correlated groups distributed over a large portion of the response vector space. Therefore, if addresses are picked randomly, a large number of locations will never be activated while a number of locations will be selected so often that their contents will resemble noise. The solution is to pick addresses according to the distribution of the data [Keeler, 1988]. In our case, we simply use an initial subset of the training response vectors as the addresses. When all address locations have subsequently been filled, the address space can be allowed to *self-organize* using the well-known *competitive Hebbian learning rule:* given a new input vector f, the closest addresses *Ak* are adapted according to

$$\vec{A}'_k = \vec{A}_k + \eta g(\vec{r}, \vec{A}_k)(\vec{r} - \vec{A}_k) \qquad (14)$$

where n > 0 is a gain term and *g* is a *radial basis function* [Poggio and Girosi, 1990] that weights the second summand according to the distance between *r* and A_k. However, for the experiments in this paper, we did not employ the above self-organization rule.

Activation of Data Locations

The distance between response vectors r_1 and r_2 is defined to be their normalized dot product (or correlation):

$$d(\vec{r}_1, \vec{r}_2) = \frac{\vec{r}_1 \cdot \vec{r}_2}{||\vec{r}_1|| \; ||\vec{r}_2||} \qquad (15$$

Given a response vector r for indexing into the memory, all storage locations whose addresses lie within a distance of *D* from r are selected (the selected locations are indicated by ones in the vector s*in Figure 3). Note that in general, an arbitrary radial basis function such as a Gaussian may be used to obtain the components of *s* instead of a strict binary threshold function. This allows for smoother interpolation between stored data vectors especially when they are used to indicate facial expression in addition to identity.

Hebbian Learning of Identity

During the training phase, the input response vector *r* is used to find *s* and the associated identity vector *d* is *added* to the previous contents of each of the selected storage locations. This corresponds to a form of generalized *Hebbian learning* as pointed out in [Keeler, 1988]. Note that this is different from a conventional memory where addresses are required to exactly match for selection and previous contents are overwritten with new data.

Retrieval of Identity

After training, the memory can be used to yield hypotheses for the identity of the object given a response vector r. First, the locations selected for *r* are found as above and the values of these selected locations are added in parallel (vector addition) to yield a sum vector 5 containing the *k* sums. These *k* sums are thresholded at 0 to obtain the data vector d i.e. d_i = 1 if $_{Si}$ > 0 and d'_t = -1 otherwise.

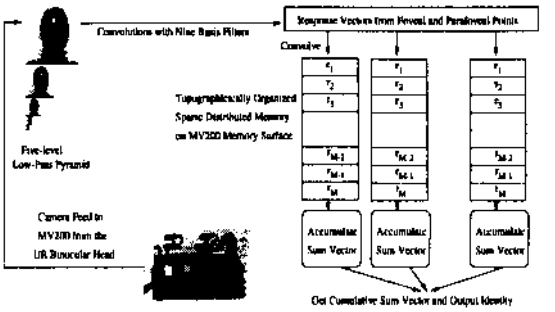

Figure 4: Implementation Diagram for the Face Recognition Scheme.

The statistically reconstructed data vector should closely resemble the original data vector (or some linear combination of the stored vectors in the case of interpolation between stored facial expressions) provided the capacity of the SDM [Kanerva, 1993] has not been exceeded. The intuitive reason for this is as follows: When storing *d* using r, each of the selected locations receives one copy of the data. During retrieval with an address close to *r,* say r, *most* of the locations that were selected with r are also selected with r. Thus, the sum vector contains most of the copies of *d,* plus copies of other different words; however, due to the orthogonality of the address space for large n, these extraneous copies are much fewer than the number of copies of *d.* This biases the sum vector in the direction of *d* and hence, *d* is output with a high probability (see [Kanerva, 1993] for a more rigorous argument based on a signal-to-noise ratio analysis).

4.2 Topographical Organization of Memory

Topology of the model points, as given by the concentric circular template (Figure 2), is preserved by using *separate SDMs* for storing vectors from each of the sparse number of locations on a face. The final output of the memory is obtained from the cumulative sum vector over the SDMs for the different facial locations. This arrangement offers at least two advantages over using a single SDM for storing vectors as proposed earlier in [Rao and Ballard, 1995c]: (a) the crosstalk between response vectors from different locations on a face is eliminated, and (b) a given response vector from a facial location needs to compared to only the model vectors in the SDM for that location, thereby speeding up the recognition pro-

5 Implementation

The face recognition scheme described above has been implemented using an active vision system comprised of

the University of Rochester (UR) binocular head which (Figure 4) provides input to a Datacube *MaxVideo*™ MV200 pipeline image-processing system capable of performing convolutions at frame-rate *(30/sec)*. The pitch of the two-eye platform is controlled by a single servo motor while separate motors control each camera's pan angle, thereby providing independent vergence control. This allows strategies for figure-ground segmentation of faces using, for instance, *zero disparity filtering* [Coombs, 1992]; once a face has been approximately segmented from the background, the concentric circular template of points can be centered on the centroid of the face. Given an input face image, the MV200 executes nine convolutions with the different 8 x 8 Gaussian derivative kernels on a low-pass filtered five-level pyramid of the input image; filter responses are then extracted from each of the sparse number of points whose coordinates are given by the concentric circular template.

Traditionally, the most time-consuming step during object indexing has been linearly accessing the large number of object representations in memory. However, our implementation greatly optimizes this step by *implementing memory directly within the pipeline image processing system itself and using convolutions for distance computations.* During indexing, an input response vector is loaded into the 8 x 8 convolution kernel and convolved with a "memory surface" containing the stored model vectors; the closest vectors can be selected by simply thresholding the results of the convolution.

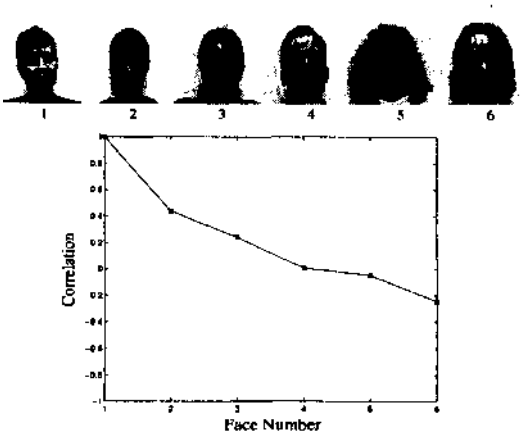

Figure 5: Discrimination Ability of the Feature Vet tors. The graph shows a plot of correlation between th vector from the approximate centroid (marked by '+') of th first face and the corresponding vectors for the other face: In this case, a threshold *D* (for the SDM) of upto 0.45 retair discriminability between the first face and the others; the us of multiple vectors at other facial locations resolves furthe ambiguities.

6 Experimental Results

The first experiment (Figure 5) illustrates the discrimination ability of the feature vectors. The vector extracted from a point near the approximate centroid of a given face was compared with those for five other faces. It is clear that the five vectors are relatively uncorrected with the vector for the original face, the closest vector having a correlation of 0.43. For the SDMs, thresholds *D* in the range 0.80-0.95 were found to yield satisfactory results. Further discriminability is obtained by using more than one vector per face from different facial locations as previously discussed.

The next experiment examines the effect of varying facial expressions on the iconic feature vectors. Figure 6 shows a set of face images of a person exhibiting a range of facial expressions. The correlation between the vectors for two different points on the neutral face image and the corresponding vectors for the other images is plotted below. The graph indicates that as expected, vectors for some facial points change much more than the others though the correlation still remains relatively high (above 0.45). This motivates the need for using a small number of images of a person under varying facial expressions for training the memory in order to achieve expression-invariant recognition. Due to the interpolation inherent in the output of the SDM, the output of the memory can then be interpreted as an indication of facial expression in addition to the identity of the person.

The third experiment tests vulnerability to occlusions. Figure 7 shows a sequence of face images with increasing facial occlusions; a plot below shows the correlations between the iconic vectors for two different points on the first face image and the corresponding vectors for the other images. The results seem to indicate that modest occlusions (as in images 2, 3, and 4) can be handled but larger occlusions (such as in 5) may require other strategies such as the one suggested in [Ballard and Rao, 1994]. The results also motivate the need for using more than one point per face in order to be able to compensate for partial occlusions near specific facial locations.

Finally, we tested the recognition performance of the method by training the memory on a face database consisting of images of 20 persons (Figure 8 (a)) exhibiting 6 different facial expressions (as in Figure 6 (a)). All images were of size 128 x 128, greyscale, 8 bit quantized, and taken under normal (overhead) illumination conditions with the face only approximately centered in the image frame. For testing the method, we used face images of the persons exhibiting facial expressions which were not used in the training set. Figure 8 (b) and (c) give examples of success and failure of the method respectively. Figure 8 (d) shows the recognition rate (the fraction of 60 test faces correctly recognized) plotted as a function of the number of points used per training face. A peak performance of 93.3% was achieved when 33 facial points were used; only 4 of the 60 test faces were incorrectly classified, with the correct identity finishing second in 3 of these 4 cases.

7 Conclusions and Future Work

A new approach to the problem of face recognition was proposed which uses iconic representations of faces as input to a topographically-organized sparse distributed memory. The iconic feature vectors are attractive as representations of faces because:

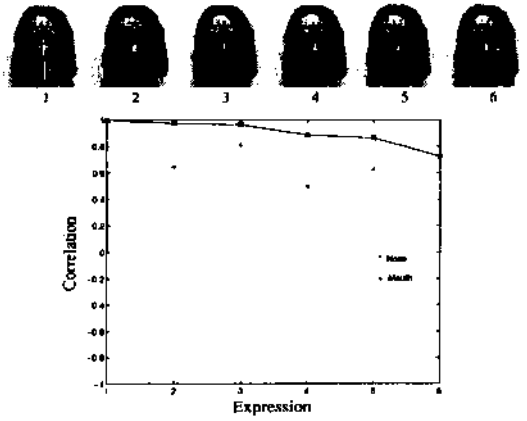

Figure 6: **Varying Facial Expressions**. The graph shows the effect of change in facial expression on the feature vectors for for two different points (marked by '+') on the face.

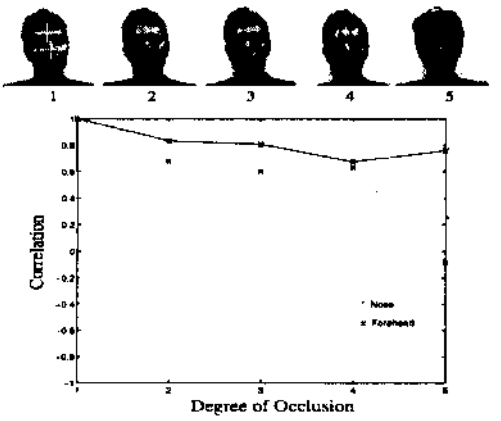

Figure 7: **Occlusion Tolerance**. The effect of increasing facial occlusions on the feature vectors for two different facial points (marked by '+') are shown.

- They simultaneously achieve the dual goals of dimensionality-reduction and orthogonality.
- They are tolerant to modest changes in facial features or expressions due to the large number of measurements incorporated in the representations.
- They allow simple strategies for rotation normalization and scale invariance[2].
- They can be computed efficiently on pipeline image processors such as the Datacube MV200.
- They facilitate real-time indexing of large face databases by allowing strategies such as using convolutions for distance computations.

The sparse distributed memory model of Kanerva was used for learning the association between feature vectors of a face and its identity. This model enjoys several favorable properties such as the ability to interpolate between stored facial expressions/views of a person, theoretically constant indexing time (due to the fixed number *M* of storage locations), possibly greater storage capacity over conventional linear memory, and anthropomorphic learning behavior in addition to the favorable properties (such as fault tolerance) that are known to accrue to distributed representations. In addition, recognition memory was topographically organized, thereby reducing crosstalk and speeding up the indexing process.

The method is clearly computation intensive; however, the recent availability of pipeline image processors significantly ameliorates this drawback since the ability of these processors to perform convolutions at frame rate (30/sec) can be effectively exploited. For example, the feature vectors for a face can be extracted after only 13 convolutions (four for generating the low-pass filtered five-level pyramid and nine for the basis filter kernels) i.e. in approximately half a second. Indexing into the SDMs using convolutions for distance computations further optimizes the recognition process. Storing upto 33 vectors for a face may seem extravagant but note that this choice still results in considerable savings over the alternative of pixel-wise storage of images (33 x 45 versus 128 x 128). An interesting question is whether the method will fail when extremely large model bases of faces are used. However, the use of more than one vector per face potentially allows an extremely large number of persons to be handled. Kanerva [1993] estimates the capacity of the SDM to be about 5% of the number of storage locations; thus, even with only 1000 storage locations for each SDM, the number of *potentially distinguishable* items is still 50^{33} which is an extremely large number even after ruling out a significant proportion of the possible combinations as being unlikely to be encountered in practice. The accuracy of the above naive estimate clearly depends on the extent to which the filter response vectors are shared between the different stored model faces; while there exists some sharing in general due to the similarity of certain facial features across persons, we believe that the possible use of self-organization within the address space of the SDMs will greatly help in further extending the capacity of the memory.

As described in Section 6, preliminary results using the proposed method have been encouraging. Future work will involve augmentation of the filter responses with color information (using, for instance, a variety of color-opponent Gaussian center-surround mechanisms derived from unsupervised learning along the RGB planes), motion-based segmentation and recognition of persons, and further testing of the method on large face databases.

[2]See [Rao and Ballard, 1995a] for more details.

References

[Ballard and Rao, 1994] Dana H. Ballard and Rajesh P.N. Rao. Seeing behind occlusions. In *Proc. of ECCV,* pages 274-285, 1994.

[Beymer, 1993] David J. Beymer. Face recognition under varying pose. Technical Report (A.I. Memo) 1461, M.I.T., December 1993.

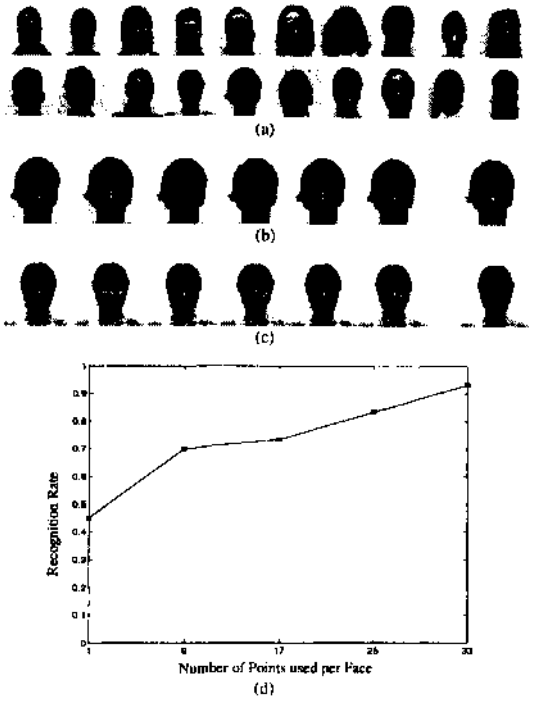

Figure 8: Recognition Performance, (a) Images of the subjects in the test database shown here for an arbitrary expression, (b) An example of a correctly recognized face (on the right) and the six images of the person used in the training set (on the left), (c) One of the 4 cases (out of 60 test cases) where the method failed, (d) Recognition rate plotted as a function of the number of points used per face.

[Bledsoe, 1966] W. W. Bledsoe. Man-machine facial recognition. Technical Report PRL22, Panoramic Research Inc., Palo Alto, CA, 1966.

[Buhmann *et* a/., 1990] Joachim M. Buhmann, Martin Lades, and Christoph v.d. Malsburg. Size and distortion invariant object recognition by hierarchical graph matching. In *Proc. IEEE IJCNN, San Diego (Vol II),* pages 411-416, 1990.

[Coombs, 1992] David J. Coombs. *Real-Time Gaze Holding in Binocular Robot Vision.* PhD thesis, University of Rochester Computer Science Dept., 1992. Available as Technical Report 415.

[Edelman *et* a/., 1992] S. Edelman, D. Reisfeld, and Y. Yeshurun. Learning to recognize faces from examples. In *Proc. of ECCV,* pages 787-791, 1992.

[Freeman and Adelson, 1991] William T. Freeman and Edward H. Adelson. The design and use of steer able filters. *IEEE PAMI* 13(9):891-906, September 1991.

[Gabor, 1946] D. Gabor. Theory of communication. .7 *IEE,* 93:429-459, 1946.

[Hancock *et* a/., 1992] Peter J.B. Hancock, Roland J. Baddeley, and Leslie S. Smith. The principal components of natural images. *Network,* 3:61-70, 1992.

[Kanade, 1973] Takeo Kanade. Picture processing by computer complex and recognition of human faces. Technical report, Technical Report, Department of Information Science, Kyoto University, 1973.

[Kanerva, 1988] Pentti Kanerva. *Sparse Distributed Memory.* Cambridge, MA: Bradford Books, 1988.

[Kanerva, 1993] Pentti Kanerva. Sparse distributed memory and related models. In Mohamad H. Hassoun, editor, *Associative Neural Memories,* pages 50-76. New York: Oxford University Press, 1993.

[Keeler, 1988] James D. Keeler. Comparison between Kanerva's SDM and Hopfield-type neural networks. *Cognitive Science,* 12:299-329, 1988.

[Kumar *et al.,* 1982] V.K. Kumar, D. Casasent, and H. Murakami. Principal-component imagery for statistical pattern recognition correlators. *Optical Engineering,* 21(1):43 47, 1982.

[Murase and Nayar, 1995] Hiroshi Murase and Shree K. Nayar. Visual learning and recognition of 3D objects from appearance. *IJCV,* 14:5-24, 1995.

[Nakamura *et al,* 1991] 0. Nakamura, S. Mathur, and T. Minami. Identification of human faces based on isodensity maps. *Pattern Recognition,* 24(3) :263-272, 1991.

[Pentland *et al,* 1994] Alex Pentland, Baback Moghaddam, and Thad Starner. View-based and modular eigenspaces for face recognition. In *Conference on Computer Vision and Pattern Recognition,* 1994.

[Poggio and Girosi, 1990] T. Poggio and F. Girosi. Networks for approximation and learning. *Proc. IEEE,* 78:1481-1497, 1990.

[Rao and Ballard, 1995a] Rajesh P.N. Rao and Dana H. Ballard. An active vision architecture based on iconic representations. Technical Report 548, Department of Computer Science, University of Rochester, 1995.

[Rao and Ballard, 1995b] Rajesh P.N. Rao and Dana H. Ballard. Learning saccadic eye movements using multiscale spatial filters. In G. Tesauro, D.S. Touretzky, and T.K. Leen, editors, *Advances in Neural Information Processing Systems* 7. Cambridge, MA: MIT Press, 1995. (To appear).

[Rao and Ballard, 1995c] Rajesh P.N. Rao and Dana H. Ballard. Object indexing using an iconic sparse distributed memory. In *Proceedings of the International Conference on Computer Vision (ICCV),* 1995. (To appear).

[Sanger, 1989] Terence David Sanger. Optimal unsupervised learning in a single-layer linear feedforward neural network. *Neural Networks,* 2:459-473, 1989.

[Turk and Pentland, 1991] Matthew Turk and Alex Pentland. Eigcnfaces for recognition. *Journal of Cognitive Neuroscience,* 3(I):71-86, 1991.

[Young, 1985] R.A. Young. The Gaussian derivative theory of spatial vision: Analysis of cortical cell receptive field line-weighting profiles. *General Motors Research Publication GMR-4920,* 1985.