Face Recognition: Eigenface, Elastic Matching, and Neural Nets

JUN ZHANG, MEMBER, IEEE, YONG YAN, MEMBER, IEEE, AND MARTIN LADES, MEMBER, IEEE

Invited Paper

This paper is a comparative study of three recently proposed algorithms for face recognition: eigenface, autoassociation and classification neural nets, and elastic matching. After these algorithms were analyzed under a common statistical decision framework, they were evaluated experimentally on four individual data bases, each with a moderate subject size, and a combined data base with more than a hundred different subjects. Analysis and experimental results indicate that the eigenface algorithm, which is essentially a minimum distance classifier, works well when lighting variation is small. Its performance deteriorates significantly as lighting variation increases. The elastic matching algorithm, on the other hand, is insensitive to lighting, face position, and expression variations and therefore is more versatile. The performance of the autoassociation and classification nets is upper bounded by that of the eigenface but is more difficult to implement in practice.

Keywords*—*Eigenface, elastic matching, face recognition, neural networks, pattern recognition.

I. INTRODUCTION

Interest and research activities in automatic face recognition have increased significantly over the past few years. While this growth largely is driven by growing application demands, such as *identification* for law enforcement and *authentication* for banking and security system access, advances in signal analysis techniques, such as wavelets and neural nets, are also important catalysts. As the number of proposed techniques increases, survey and evaluation becomes important. Samal and Iyengar [1] reviewed feature-based techniques, where the features are related to face geometry. Valentin *et al.* [2] is a survey on connectionist/neural-net techniques, where associative memory and classification neural nets are used for face recall, recognition, and psychology-related studies. The most recent and comprehensive survey perhaps is that of Chellappa *et al.* [3], which contains many of the latest techniques, such as eigenface and elastic matching.

In this paper, we selected three techniques surveyed in [2] and [3] for comparative study and evaluation, using a common face data base that contains more than 100 persons. The three techniques are eigenface [6], elastic matching [9], and autoassociation and back-propagation neural nets [10]. These techniques are recent, have apparently promising performances, and are representative of new trends in face recognition. For example, both eigenface and elastic matching are presented in [3] as major recent techniques as opposed to "earlier approaches" [3, Section IV-C2, pp. 718–725]. All three techniques were reported to have recognition rates of more than 80–90% on data bases of moderate sizes (e.g., 16–50 persons).

We believe this work would be a useful complement to [1]–[3], where the surveyed techniques were not evaluated on a common data base of relatively large size. Indeed, through a more focused and detailed comparative study of three important techniques, our goal is to gain more insights into their underlying principles, interrelations, advantages, limitations, and design tradeoffs and, more generally, into what the critical issues really are for an effective recognition algorithm.

The rest of this paper is organized as follows. Section II provides a brief review of the three techniques and some theoretical analysis of their strengths and limitations. Section III presents and discusses experimental results, followed by a summary in Section IV.

While revising this paper, we came across three interesting recent techniques. These techniques, due to Atick *et al.* [23], [24], Moghaddam *et al.* [25], and Kung *et al.* [26], are closely related to eigenface, elastic matching, and neural nets, respectively and will be discussed at the end of Section II.

Manuscript received February 1, 1997; revised May 1, 1997. This work was supported in part by the National Science Foundation under Grant IRI-9315193. This work was performed by the Lawrence Livermore National Laboratory under U.S. Department of Energy Contract W-7405-ENG-48.

J. Zhang is with the Department of Electrical Engineering and Computer Science, University of Wisconsin–Milwaukee, WI 53201 USA.

Y. Yan is with Intelligent Medical Imaging Inc., Palm Beach Gardens, FL 33410 USA.

M. Lades is with the Institute for Scientific Computing Research, Lawrence Livermore National Laboratory, Livermore, CA 94550 USA.

Publisher Item Identifier S 0018-9219(97)06637-1.

II. Eigenface, Elastic Matching, and Neural Nets

For almost all previously proposed techniques, the success of face recognition depends on the solution of two problems: *representation* and *matching*. We now discuss these in some detail, thereby providing a unified statistical decision (pattern recognition) framework [19] for the three techniques studied in this paper.

A. Representation, Matching, and Statistical Decision

At an elementary level, the image of a face is a two-dimensional (2-D) array of numbers, i.e., pixel gray levels. This can be written as

$$\mathbf{x} = \{x_i, i \in \mathbf{S}\} \tag{1}$$

where $\mathbf{S}$ is a square lattice. Sometimes, however, it is more convenient to express $\mathbf{x}$ as a one-dimensional (1-D) column vector of concatenated rows of pixels, i.e.,

$$\mathbf{x} = [x_1, x_2, \cdots, x_n]^T \tag{2}$$

where $n = \|\mathbf{S}\|$ is the total number of pixels in the image. Unless stated otherwise, we will use this 1-D notation and let $\mathbf{x} \in \mathbf{R}^n$, the n-dimensional Euclidean space. For a given representation, two things are important: *discriminating power* and *efficiency*, that is, how far apart are the faces under the representation and how compact is the representation.

While some previous techniques represent faces in their most elementary forms of (1) or (2), many others use a *feature vector*

$$\mathbf{f}(\mathbf{x}) = [f_1(\mathbf{x}), f_2(\mathbf{x}), \cdots, f_m(\mathbf{x})]^T \tag{3}$$

where $f_1(\cdot), f_2(\cdot), \cdots, f_m(\cdot)$ are linear or nonlinear functionals. Since generally $m \ll n$, feature-based representations usually are more efficient.

A simple way to achieve good efficiency is to use an alternative orthonormal basis of $\mathbf{R}^n$. Specifically, suppose $\mathbf{e}_1, \mathbf{e}_2, \cdots, \mathbf{e}_n$ are an orthonormal basis. Then $\mathbf{x}$ can be expressed as

$$\mathbf{x} = \sum_{i=1}^{n} \hat{x}_i \mathbf{e}_i \tag{4}$$

where $\hat{x}_i = \langle \mathbf{x}, \mathbf{e}_i \rangle$ (inner product) and $\mathbf{x}$ can be equivalently represented by $\hat{\mathbf{x}} = [\hat{x}_1, \hat{x}_2, \cdots, \hat{x}_n]^T$. Two examples of orthonormal basis are the natural basis used in (2) with $\mathbf{e}_i = [0, 0, \cdots, 1, 0, \cdots, 0]^T$, where one is at the ith position, and the Fourier basis $\mathbf{e}_i = (1/\sqrt{n})[1, e^{j2\pi(i/n)}, e^{j2\pi(2i/n)}, \cdots, e^{j2\pi(n-1)(i/n)}]^T$. If for a given orthonormal basis $\hat{x}_i$ are small when $i \geq m$, then the face vector $\hat{\mathbf{x}}$ can be compressed into an m-dimensional vector

$$\hat{\mathbf{x}} \simeq [\hat{x}_1, \hat{x}_2, \cdots, \hat{x}_m]^T. \tag{5}$$

It is important to notice that an efficient representation does not necessarily have good discriminating power [19].

In the matching problem, an incoming face is recognized by identifying it with a prestored known face. For example, suppose the input face is $\mathbf{x}$ and there are K prestored faces $\mathbf{c}_k$, $k = 1, 2, \cdots, K$. Then, one possibility is to assign $\mathbf{x}$ to $\mathbf{c}_{k_0}$ if

$$k_0 = \arg \min_{1 \leq k \leq K} \|\mathbf{x} - \mathbf{c}_k\| \tag{6}$$

where $\|\cdot\|$ represents the Euclidean distance in $\mathbf{R}^n$. If $\|\mathbf{c}_k\|$ is normalized so that $\|\mathbf{c}_k\| = c$ for all k, the *minimum distance* matching in (6) simplifies to *correlation matching*

$$k_0 = \arg \max_{1 \leq k \leq K} \langle \mathbf{x}, \mathbf{c}_k \rangle. \tag{7}$$

Since distance and inner product are *invariant* to change of orthonormal basis, minimum distance and correlation matching can be performed using any orthonormal basis and the recognition performance will be the same. To do this, simply replace $\mathbf{x}$ and $\mathbf{c}_k$ in (6) or (7) by $\hat{\mathbf{x}}$ and $\hat{\mathbf{c}}_k$. Similarly, (6) and (7) also could be used with feature vectors.

Due to such factors as viewing angle, illumination, facial expression, distortion, and noise, the face images for a given person can have random variations and therefore are better modeled as a random vector. In this case, *maximum-likelihood* (ML) matching is often used, i.e.,

$$k_0 = \arg \min_{1 \leq k \leq K} \log p(\mathbf{x} \mid \mathbf{c}_k) \tag{8}$$

where $p(\mathbf{x} \mid \mathbf{c}_k)$ is the density of $\mathbf{x}$ conditioning on its being the kth person. The ML criterion minimizes the probability of recognition error when *a priori*, the incoming face is equally likely to be that of any of the K persons. Furthermore, if we assume that variations in face vectors are caused by additive white Gaussian noise (AWGN), i.e.,

$$\mathbf{x}_k = \mathbf{c}_k + \mathbf{w}_k \tag{9}$$

where $\mathbf{w}_k$ is a zero-mean AWGN with power σ^2, then the ML matching becomes the minimum distance matching of (6).

B. Eigenface

As mentioned, alternative orthonormal bases often are used to compress face vectors. One such basis is the Karhonen–Loéve (KL) [17]. Suppose that the face vectors are modeled by a random vector $\mathbf{x}$ with a covariance matrix

$$\mathbf{C} = \mathrm{E}[\mathbf{x}\mathbf{x}^T]. \tag{10}$$

The KL bases are formed by the eigenvectors of $\mathbf{C}$. Since the eigenvectors associated with the first few largest eigenvalues have face-like images (see also Fig. 1), they also are referred to as eigenfaces [6]. Specifically, suppose the eigenvectors of $\mathbf{C}$ are $\mathbf{u}_1, \mathbf{u}_2, \cdots, \mathbf{u}_n$ and are associated with eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$. Then

$$\mathbf{x} = \sum_{i=1}^{n} \hat{x}_i \mathbf{u}_i \tag{11}$$

Fig. 1. Examples of eigenfaces.

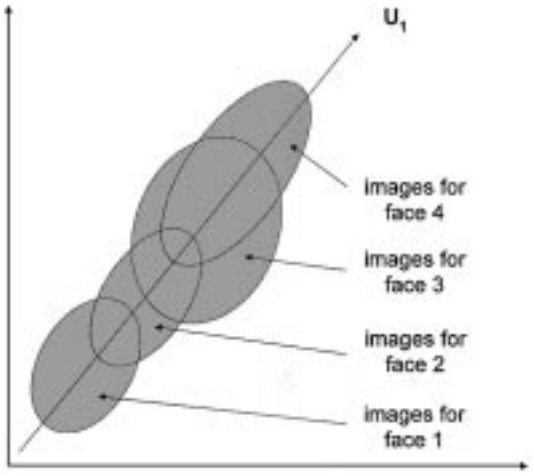


Fig. 2. Efficiency does not necessarily imply discriminating power.

and compression can be achieved by letting

$$\mathbf{x} \simeq \sum_{i=1}^{m} \hat{x}_i \mathbf{u}_i, \quad \hat{\mathbf{x}} \simeq [\hat{x}_1, \hat{x}_2 \cdots, \hat{x}_m]^T \tag{12}$$

where m usually is selected such that λ_i is small for $i > m$.

The KL (eigenface) representation of (12) is well known in statistics literature as the principle component analysis [13]. It is optimal in the sense of efficiency—for any given $m < n$, (12) has the minimum mean square error among all possible approximations of $\mathbf{x}$ that use m orthonormal vectors. Here is another way to look at it: since λ_i is the spread (variance) of the face population along direction $\mathbf{u}_i$ and since the amount of information (entropy) in a population increases with the spread [20], eigenfaces of (12) retain, for any m, the most amount of information in the face population. All these, however, do not mean that the KL representation is optimal in the sense of discrimination power, which relies more on the *separation* between different faces (different persons) than the spread of all faces [19] (see also Fig. 2).

To use eigenface in face recognition, we can use the minimum distance matching of (6) under the KL basis

$$k_0 = \arg \min_{1 \le k \le K} \|\hat{\mathbf{x}} - \hat{\mathbf{c}}_k\|. \tag{13}$$

To implement this, we need to find the covariance matrix $\mathbf{C}$ and its eigenvectors from training data. Specifically, suppose that $\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N$ are N training face vectors. Then, by definition, $\mathbf{C}$ can be estimated by [11]

$$\mathbf{C} = \mathrm{E}[\mathbf{x}\mathbf{x}^T] \simeq \frac{1}{N} \sum_{k=1}^{N} \mathbf{x}_k \mathbf{x}_k^T. \tag{14}$$

If we pack the training vectors into a matrix

$$\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N] \tag{15}$$

then the estimate of $\mathbf{C}$ can be written as

$$\mathbf{C} \simeq \frac{1}{N} \mathbf{X}\mathbf{X}^T. \tag{16}$$

To find the eigenvectors of $\mathbf{C}$, we just need to find the eigenvectors of $\mathbf{X}\mathbf{X}^T$ (the factor $1/N$ has no effect on the eigenvectors). Even for images of moderate size, however, this is difficult. For example, if the face vectors $\mathbf{x}_k$ are from 128×128 images [see (1) and (2)], the dimension of $\mathbf{X}\mathbf{X}^T$ will be $128^2 \times 128^2 \simeq 2.7 \times 10^8$.

The key to a solution is to realize that [16], through a singular value decomposition (SVD) of $\mathbf{X}$, the eigenvectors of $\mathbf{X}\mathbf{X}^T$ (an n by n matrix) can be found from eigenvectors of $\mathbf{X}^T\mathbf{X}$ (an N by N matrix), which are much easier to obtain since usually $N \ll n$ (for example, $N = 100$ gives reasonably good results). Specifically, suppose the rank of $\mathbf{X}$ is r, $r \le N$. According to linear algebra theory [5], $\mathbf{X}$ has an SVD

$$\mathbf{X} = \sum_{k=1}^{r} \sqrt{\lambda_k} \mathbf{u}_k \mathbf{v}_k^T. \tag{17}$$

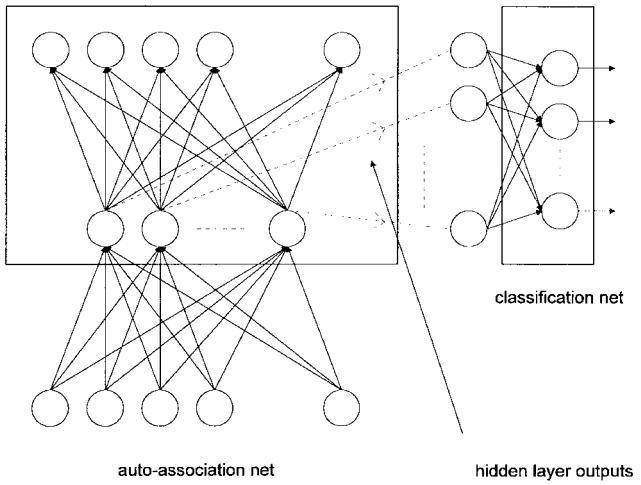


Fig. 3. Autoassociation and classification networks.

Here, $\sqrt{\lambda_k}$, $\mathbf{u}_k$, and $\mathbf{v}_k$ are, respectively, singular values and left and right singular vectors of $\mathbf{X}$. Furthermore [5], λ_k are nonzero eigenvalues of $\mathbf{X}\mathbf{X}^T$ and $\mathbf{X}^T\mathbf{X}$ and $\mathbf{u}_p$ (n by one vector) and $\mathbf{v}_p$ (N by one vector) are, respectively, the eigenvectors of $\mathbf{X}\mathbf{X}^T$ and $\mathbf{X}^T\mathbf{X}$. This means (by multiplying both sides of (17) with $\mathbf{v}_k$)

$$\mathbf{u}_k = \frac{1}{\sqrt{\lambda_k}}\mathbf{X}\mathbf{v}_k. \tag{18}$$

Hence, we can find eigenface $\mathbf{u}_k$ easily after finding $\mathbf{v}_k$, which is relatively easy.

C. Neural Nets

In principle, the popular back-propagation (BP) neural net [15] may be trained to recognize face images directly. For even a moderate image size, however, the network can be quite complex and, therefore, difficult to train. For example, if the images are 128 × 128, the number of inputs of the network would be 16 384. To reduce complexity, Cottrell and Fleming [10] used two BP nets, as illustrated in Fig. 3. The first net operates in the *autoassociation* mode [14] and extracts features for the second net, which operates in the more common *classification* mode.

The autoassociation net has n inputs, n outputs, and p hidden-layer nodes. Usually, $p \ll n$. The network takes a face vector $\mathbf{x}$ as an input and is trained to produce an output $\mathbf{y}$ that is a "best approximation" of $\mathbf{x}$. In this way, the hidden-layer output $\mathbf{h}$ constitutes a compressed version of $\mathbf{x}$, or a feature vector, and can be used as the input to the classification net (see Fig. 3).

What is this feature vector? How good is it? Bourlard and Kamp [14] showed that "under the best circumstances," i.e., when the sigmoidal functions at the network nodes are replaced by linear functions (i.e., when the network is *linear*), the feature vector is the same as that produced by the KL basis, or the eigenfaces, of Section II-B. When the network is nonlinear, the feature vector could deviate from the best. While the details of the mathematical development can be found in [14], the key here turns out to be an application of the SVD.

Specifically, suppose that for each training face vector $\mathbf{x}_k$ (n-dimensional), $k = 1, 2, \cdots, N$, the outputs of the hidden layer and output layer for the autoassociation net are, respectively, $\mathbf{h}_k$ (p-dimensional, usually $p \ll n$ and $p < N$) and $\mathbf{y}_k$ (n-dimensional), with

$$\mathbf{h}_k = F(\mathbf{W}_1\mathbf{x}_k), \quad \mathbf{y}_k = \mathbf{W}_2\mathbf{h}_k. \tag{19}$$

Here, $\mathbf{W}_1$ (p by n) and $\mathbf{W}_2$ (n by p) are corresponding weight matrixes and $F(\cdot)$ is either a linear or a nonlinear function, applied "component-by-component." If we pack $\mathbf{x}_k$, $\mathbf{y}_k$, and $\mathbf{h}_k$ into matrixes as in the eigenface case [see (15)], then the above relations can be rewritten as

$$\mathbf{H} = F(\mathbf{W}_1\mathbf{X}), \quad \mathbf{Y} = \mathbf{W}_2\mathbf{H}. \tag{20}$$

Minimizing the training error for the autoassociation net amounts to minimizing the Frobenius matrix norm

$$\|\mathbf{X} - \mathbf{Y}\|^2 = \sum_{k=1}^{n} \|\mathbf{x}_k - \mathbf{y}_k\|^2. \tag{21}$$

Since $\mathbf{Y} = \mathbf{W}_2\mathbf{H}$, its rank is no more than p. Hence, to minimize the training error, $\mathbf{Y} = \mathbf{W}_2\mathbf{H}$ should be the best rank-p approximation to $\mathbf{X}$, which means [14]

$$\mathbf{W}_2\mathbf{H} = \mathbf{U}_p\mathbf{\Lambda}\mathbf{V}_p^T \tag{22}$$

where

$$\mathbf{U}_p = [\mathbf{u}_1, \mathbf{u}_2, \cdots \mathbf{u}_p]^T, \quad \mathbf{V}_p = [\mathbf{v}_1, \mathbf{v}_2, \cdots \mathbf{v}_p]^T$$

are, respectively, the first p left and right singular vectors in the SVD of $\mathbf{X}$ [see also (17)], which also are the first p eigenvectors of $\mathbf{X}\mathbf{X}^T$ and $\mathbf{X}^T\mathbf{X}$.

One way to achieve this optimum is to have a linear $F(\cdot)$ (e.g., $F(u) = u$) and to set the weights to

$$\mathbf{W}_1^T = \mathbf{W}_2 = \mathbf{U}_p. \tag{23}$$

Since $\mathbf{U}_p$ contains the first p eigenvectors of $\mathbf{X}\mathbf{X}^T$, we have for any input $\mathbf{x}$

$$\mathbf{h} = \mathbf{W}_1\mathbf{x} = \mathbf{U}_p\mathbf{x} = [\langle \mathbf{u}_1, \mathbf{x}\rangle, \langle \mathbf{u}_2, \mathbf{x}\rangle, \cdots, \langle \mathbf{u}_p, \mathbf{x}\rangle]^T = \hat{\mathbf{x}}$$

which is the same as the feature vector in the eigenface approach [see (12)].

We need to remark, however, that the autoassociation net, when it is trained by the BP algorithm with a nonlinear $F(\cdot)$, generally cannot achieve this optimal performance [13].

D. Elastic Matching

Since most face-recognition algorithms are in some sense minimum distance classifiers, it is important to consider

more carefully how distance should be defined. In the previous two examples (eigenface and autoassociation and classification nets), the distance between the observed face $\mathbf{x}$ and a template $\mathbf{c}$ (known face) is the common Euclidean distance[1]

$$d(\mathbf{x}, \mathbf{c}) = ||\mathbf{x} - \mathbf{c}||.$$

While easy to compute, it also has some shortcomings. For example, when $\mathbf{x}$ is exactly the same as $\mathbf{c}_k$ except for a shift and dilation (i.e., affine transformation), $d(\mathbf{x}, \mathbf{c})$ will not be zero and can even be quite large. As another example, suppose that $\mathbf{x}$ is a "smiling" version of $\mathbf{c}$ (local transformations/deformations); then $d(\mathbf{x}, \mathbf{c})$ will also not be zero. Last, it is often desirable for $\mathbf{c}$ to have a much smaller dimension than that of $\mathbf{x}$. For example, one may want the template to contain only information about key points in the face (e.g., the edges that define the eyes and mouth). In such cases, the Euclidean distance is not well defined.

From these observations, it is desirable to have a distance measure that is "invariant" to common face transformations and does not require $\mathbf{c}$ and $\mathbf{x}$ to have the same dimension. One such approach is the elastic matching algorithm proposed by Lades *et al.* [9], which has roots in aspect-graph matching [22].

To describe this algorithm, we start by defining a new type of representation for the face template, as illustrated in Fig. 4. Let $\mathbf{S}$ be the original two-dimensional image lattice [see (1)]. Then, the face template is a *vector field*

$$\mathbf{c} = \{\mathbf{c}_i, i \in \mathbf{S}_1\} \tag{24}$$

where $\mathbf{S}_1$ is a lattice embedded in $\mathbf{S}$ and $\mathbf{c}_i$ is a feature vector at position i. Normally, $\mathbf{S}_1$ is much coarser and smaller than $\mathbf{S}$ [see Fig. 4(a)]. The idea is that $\mathbf{c}$ should contain only the most critical information about the face.

Many choices exist for the feature vectors $\mathbf{c}_i$. In this work, the Gabor features of Lades *et al.* [9] were used. Let the image from which the template $\mathbf{c}$ is constructed be denoted as α (defined over $\mathbf{S}$) and let

$$\mathbf{G} = [g^1, g^2, \cdots, g^m]^T \tag{25}$$

be a vector whose components are 2-D Gabor filters with various center frequencies, bandwidths, and orientations [9]. Then, $\mathbf{c}_i$ consists of the magnitude of Gabor filter outputs sampled at position $i \in \mathbf{S}_1$, i.e., [see Fig. 4(b)]

$$\mathbf{c}_i = |(\mathbf{G} * \alpha)_i| \tag{26}$$

where $*$ represents convolution and the absolute value is taken component by component. The Gabor features $\mathbf{c}_i$ provide multiscale edge strengths at position i. A perceptually salient part of an image tends to have a large $||\mathbf{c}_i||$.

[1] This distance is sometimes computed using an alternative orthonormal basis as $||\hat{\mathbf{x}} - \hat{\mathbf{c}}_k||$.

Similarly, we can define the observed face as a vector field on the original image lattice $\mathbf{S}$

$$\mathbf{x} = \{\mathbf{x}_j, j \in \mathbf{S}\} \tag{27}$$

where $\mathbf{x}_j$ is the same type of feature vector as $\mathbf{c}_i$ but defined on the fine-grid lattice $\mathbf{S}$. To differentiate this $\mathbf{x}$ from the face image of (1), we call it the *face vector field*.

In the elastic matching approach, the distance $d(\mathbf{c}, \mathbf{x})$ is defined through a "best match" between $\mathbf{c}$ and $\mathbf{x}$. Specifically, a match between $\mathbf{c}$ and $\mathbf{x}$ can be described uniquely through a *mapping* between $\mathbf{S}_1$ and $\mathbf{S}$, denoted by

$$\mathbf{M} : \mathbf{S}_1 \to \mathbf{S}. \tag{28}$$

Fig. 4(c) shows two such mappings (it also shows a graphical way to visualize a mapping). Without restriction, the total number of such mappings is $||\mathbf{S}||^{||\mathbf{S}_1||}$. For example, when $\mathbf{S}_1$ is five by five and $\mathbf{S}$ is ten by ten (very moderate sizes), the total number of mappings is $(10{\times}10)^{5\times5} = 10^{50}$.

As a distance measure for face recognition, the best match we want is one that preserves features and local geometry. Suppose that $i \in \mathbf{S}_1$ and $j = \mathbf{M}(i) \in \mathbf{S}$. Feature preserving means that $\mathbf{x}_j$ is not too different from $\mathbf{c}_i$. An example of local geometry to be preserved is the local (spatial) distance. That is, if i_1 and i_2 are close in $\mathbf{S}_1$, so should $j_1 = \mathbf{M}(i_1)$ and $j_2 = \mathbf{M}(i_2)$ in $\mathbf{S}$. Last, the match is called elastic since the preservation can be *approximate* rather than *exact* (rigid). Using the graphical scheme in Fig. 4(c), this means that lattice $\mathbf{S}_1$ can be "stretched" unevenly.

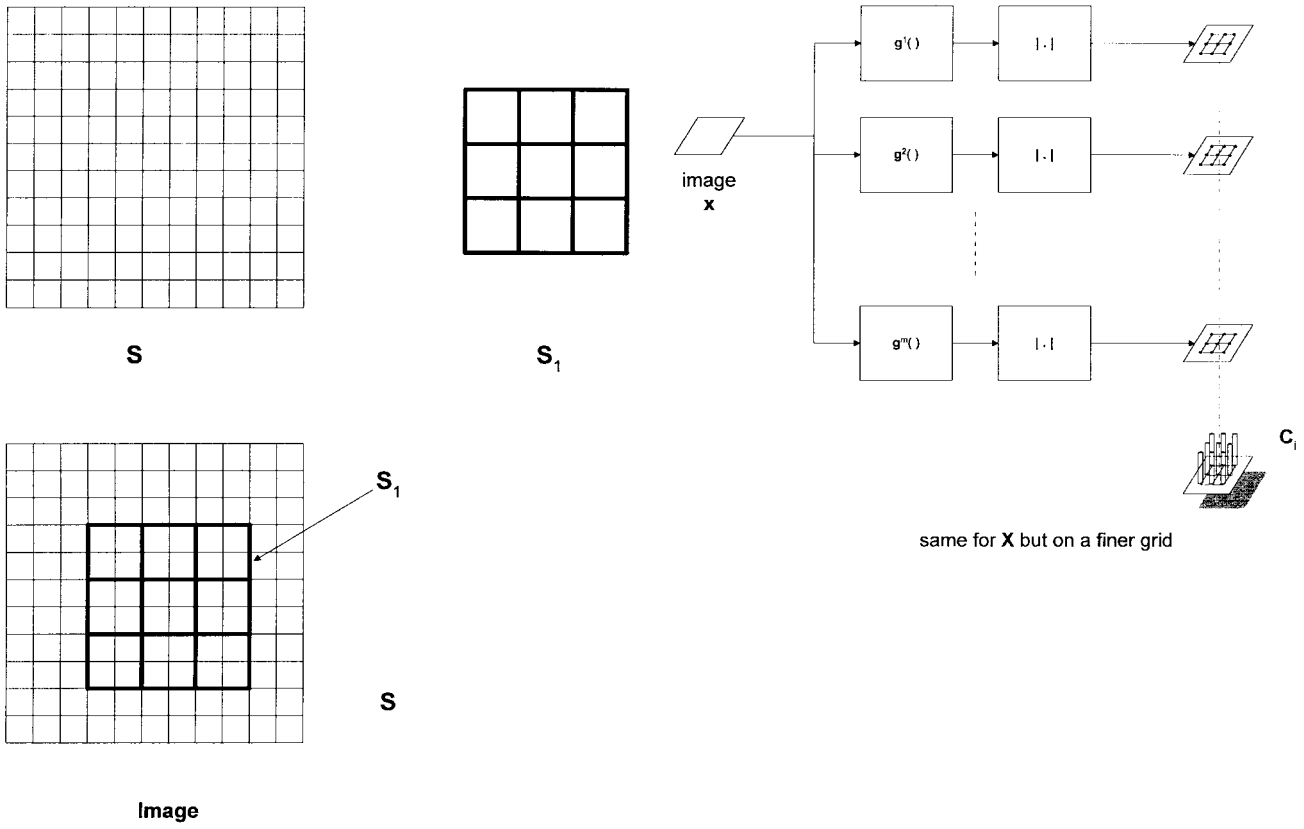
The quality of different matches can be evaluated using an energy function. The one used by Lades *et al.* [9], for example, is

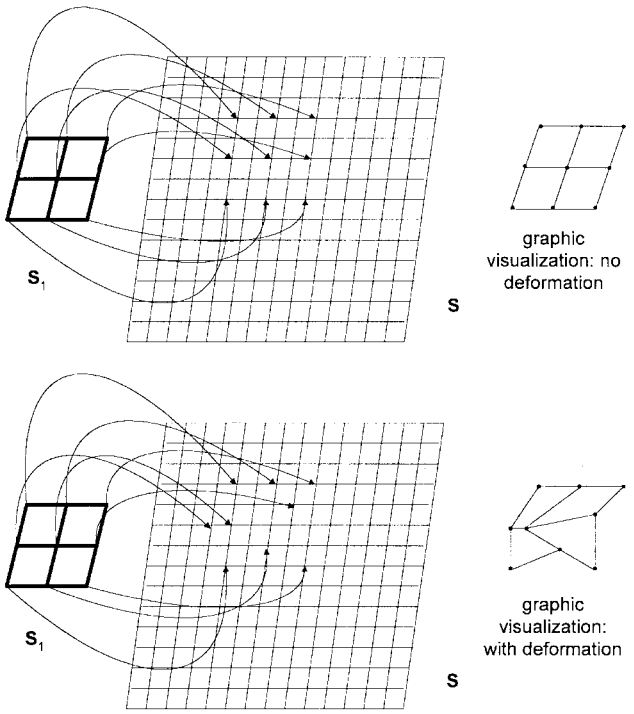
$$E(\mathbf{M}) = \sum_i \left[-\frac{\langle \mathbf{c}_i, \mathbf{x}_j \rangle}{||\mathbf{c}_i|| ||\mathbf{x}_j||} \right] + \lambda \sum_{(i_1, i_2)} [(i_1 - i_2) - (j_1 - j_2)]^2. \tag{29}$$

In this equation, the first sum runs over $\mathbf{S}_1$, the second sum runs over all i_1 and i_2 that are neighbors in $\mathbf{S}_1$, and j, j_1, and j_2 are positions in $\mathbf{S}$ that are mapped from i, i_1, and i_2, respectively. The term in the first sum is small when $\mathbf{x}_j \simeq \mathbf{c}_i$ (feature preserving) and the term in the second sum is small when $j_1 - j_2 \simeq i_1 - i_2$ (preserving local distance and ordering).[2]

For a given energy function, the best match is one with the minimum energy. Due to the large number of possible matches, however, this might be difficult to obtain in finite time. Hence, an approximate solution has to be found. In [9], this is achieved in two stages: *rigid matching* and *deformable matching*. In rigid matching, $\mathbf{c}$ is moved around in $\mathbf{x}$ like conventional template matching and, at each position, $||\mathbf{c} - \mathbf{x}'||$ is calculated. Here, $\mathbf{x}'$ is the part of $\mathbf{x}$ that, after being matched with $\mathbf{c}$, does not lead to any

[2] For example, "one position to the left" is not the same as "one position to the right," even though they correspond to the same distances.



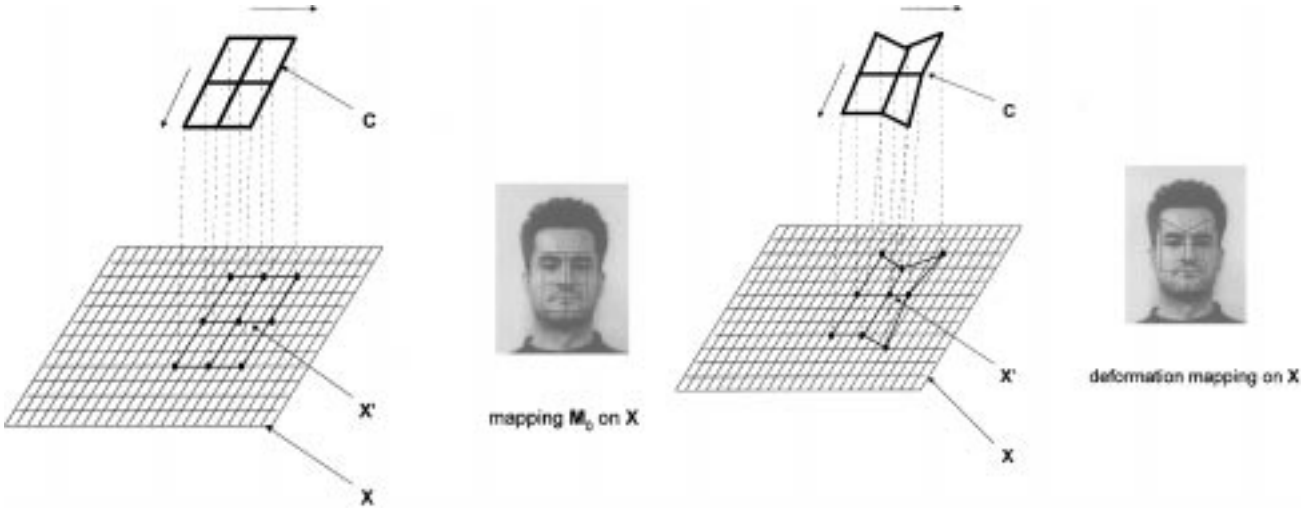


(c)

Fig. 4. Elastic matching. (a) Illustration of $\mathbf{S}$ and $\mathbf{S}_1$. (b) The template and face vector fields. (c) Examples of mappings.

deformation of $\mathbf{S}_1$. This is illustrated in Fig. 4(d). Once a minimum distance position is found, an initial match $\mathbf{M}_0$ is defined and has energy $E(\mathbf{M}_0)$ [see Fig. 4(d)].

In deformable matching, lattice $\mathbf{S}_1$ is stretched through random local perturbations to $\mathbf{M}_0$ to reduce further the energy function. This is illustrated in Fig. 4(e). Here, a



(d) (e)

(f)

Fig. 4. *(Continued.)* Elastic matching. (d) Stage 1 of elastic matching: $\mathbf{C}$ and $\mathbf{X}$. (e) Stage 2 of elastic matching: local deformation. (f) Place a coarse-grid $\mathbf{S}_1$ "by hand" on $\mathbf{X}$.

random local perturbation to $\mathbf{M}_0$ amounts to the following.

- Pick a point in $\mathbf{S}_1$ at random, denoted as i.
- Suppose that $M_0(i) = j$. Select at random a point $j' \in \mathbf{S}$ that is (spatially) close to j. Let $M_0'(i) = j'$, where M_0' is the match after perturbation. Notice that for any $i' \neq i, i' \in \mathbf{S}_1$, $M_0'(i') = M(i')$.

After the deformable matching converges to a mapping $\mathbf{M}^*$, the distance between $\mathbf{x}$ and $\mathbf{c}$ is taken as

$$d(\mathbf{c}, \mathbf{x}) = E(\mathbf{M}^*)$$

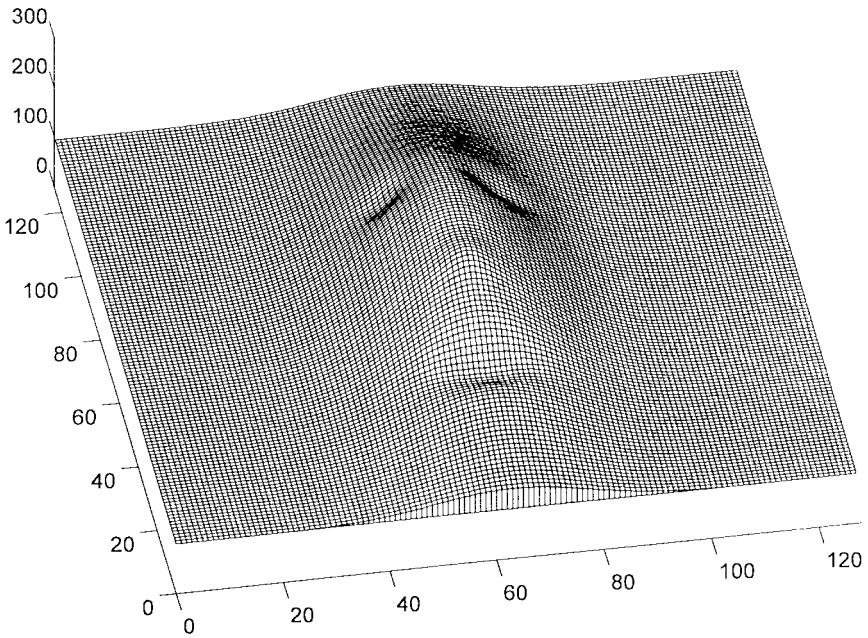
and used for face recognition through a minimum distance classifier like (6).

Last, there is the question of face template/model construction. An automated scheme has been shown to work quite well [18]. Although the idea of this scheme is simple and interesting, its description is tedious and, therefore, can be found in the Appendix.

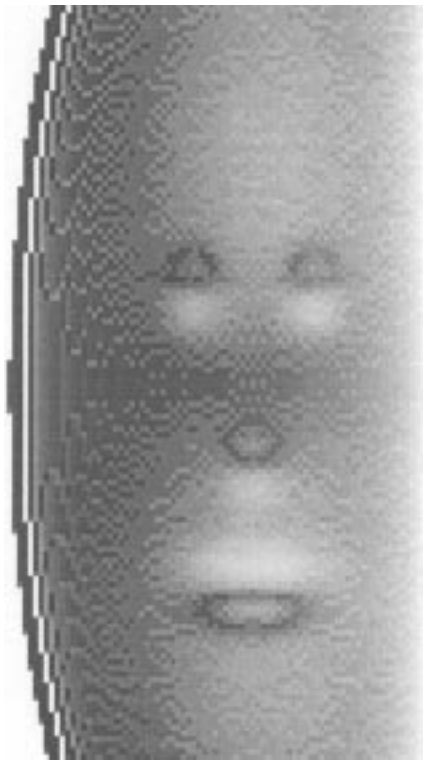
E. Three Recent Techniques

Atick *et al.*'s recent work [23], [24] is a significant extension to the eigenface. Its basic idea, as shown in Fig. 5, is to model the frontal (physical) human face as a surface $z = f(x, y)$ in a 3-D space and recover it from its 2-D image $I(x, y)$. This problem, known as "shape from shading," is generally an ill-posed inverse problem since the image-formation process (from the physical $f(x, y)$ to $I(x, y)$) is many-to-one. Traditionally, this problem is solved by imposing smoothness constraints on the estimate of $f(x, y)$ [27]. Such constraints are too weak to produce good results for real-world objects, however, such as faces, which have more specific structures than just being smooth. Atick *et al.*'s rather ingenious idea is to expand $f(x, y)$ with respect to a set of eigensurfaces $\{\phi_k(x, y)\}$ called *eigenheads*, i.e.,

$$f(x, y) = m(x, y) + \sum_k f_k \phi_k(x, y) \tag{30}$$



(a)



(b)

Fig. 5. A synthetic physical face surface $f(x, y)$ and its face image $I(x, y)$. (a) Face surface. (b) The image of the face surface, illuminated from the side.

where $m(x, y)$ is a mean image. The mean image and the eigenheads are obtained from training samples (laser scans of human faces) and usually, only a small number of them are needed (compression property).

Using the representation of (30), the shape-from-shading problem becomes that of finding coefficients f_k from $I(x, y)$. Since the number of f_k's is very small, the problem becomes well posed, and smoothness constraints are not needed. Indeed, this leads to better results, since now the estimate of $f(x, y)$ is constrained to be in the subspace spanned by the eigenheads, which contains surfaces that not only are smooth but also look like real face surfaces.

The eigenhead has a significant advantage over the eigenface (which amounts to the eigenrepresentation for $I(x, y)$). That is, $f(x, y)$, if it can be recovered, is the *real face* and is independent of the image-formation process (illu-

Table 1 The Original Data Bases and Their Variations

Database	Subject	Variation	Total
MIT	16	27	432
Olivetti	40	10	400
Weizmann	28	30	840
Bern	30	10	300

mination, etc.). Hence, eigenhead-based face recognition could be significantly more robust. The eigenhead approach, however, also has a problem (as does the eigenface): it cannot deal effectively with rotation. One possible solution is to introduce a 3-D rotation parameter in the Atick *et al.* formulation.

In their recent work, Moghaddam *et al.* [25] proposed an interesting elastic matching technique that can be viewed as an alternative to that described in Section II-D. The idea here is to consider two images as 2-D surfaces in a 3-D space. For example, image $I(x,y)$ can be considered as a surface $z = I(x,y)$ or $(x,y,I(x,y))$. A match is obtained by deforming one surface so that its shape becomes close to that of the other surface. The deforming dynamics are controlled by a set physically motivated differential equations. Moghaddam *et al.* [25] reported improved recognition performance over simple L^2 distance classifiers (such as the eigenface). Since the face image is used in the matching, however, the result still may be sensitive to lighting variation (just as the eigenface is).

Last, Kung *et al.* [26] proposed a neural-net technique and tested it on a subset of the Army Research Laboratory FERET data base (using 200 persons, two images/person). The neural net, called the decision-based neural net (DDBN), classifies its input, which is, basically, a preprocessed and subsampled (8×8) face image. From a mathematical point of view, the DDBN computes (Gaussian-type) likelihood functions and implements the ML classification rule of (8). Very high recognition rates were reported (96–99%). It would be of interest to understand how much the good performance is due to preprocessing and how much of it is due to the neural net.

III. Experimental Results

The eigenface, autoassociation and classification nets, and elastic matching algorithms were tested in face-recognition experiments. Here, the results are presented and discussed, starting with the data bases used.

A. Data Bases

Since most of the data bases available to us were of moderate subject size, we used four data bases to achieve a reasonably large size. The data bases were those of the Massachusetts Institute of Technology (MIT), the Olivetti Research Lab,[3] the Weizmann Institute of Science, and Bern University.[4] As detailed in Table 1, the number of subjects in each data base ranges from 16 to 40, resulting in a total of 114. For each subject, there are ten to 30 pictures and, as shown in Fig. 6, they often contain significant variations in scale and viewing angle. Since the eigenface and neural-net techniques generally require the images to be of the same scale and viewing angle, the data bases were "trimmed" such that inside each trimmed data base, all images are frontal view and have roughly the same scale. As detailed in Table 2, the four trimmed data bases still have a total of 114 subjects but for each subject, there are now two to three pictures with variations in lighting, background, and expression. The four trimmed data bases[5] were also combined to generate a single data base. Since the four data bases have different scales (even though *within* each data base the pictures have roughly the same scale), scale normalization was used such that in the combined data base, all images have roughly the same scale.

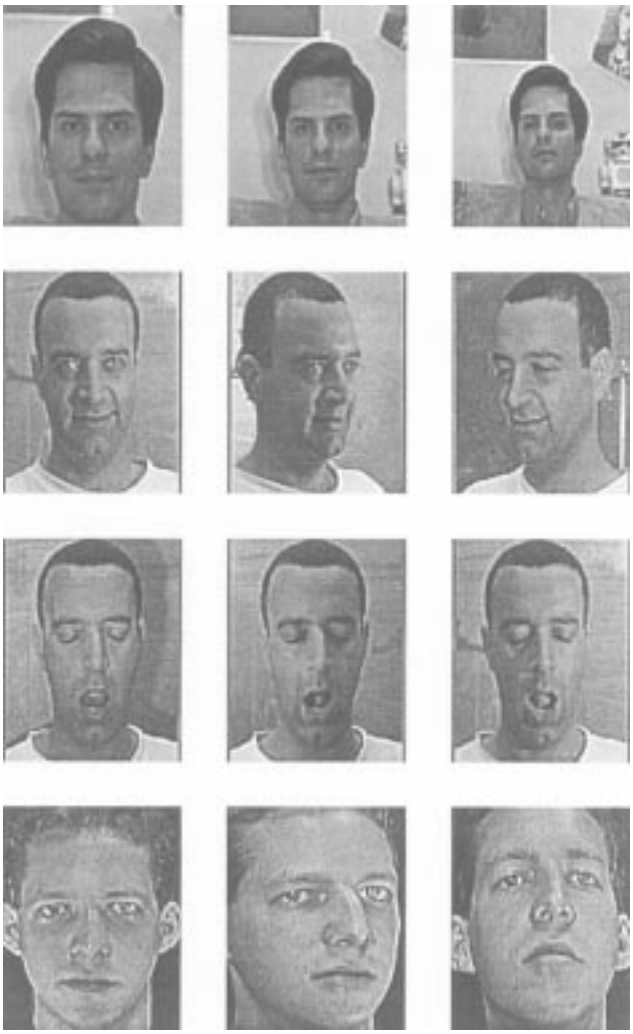


Fig. 6. Examples of faces in the data bases.

[3] Copyright © 1996, ORL, All Rights Reserved.

[4] Copyright © 1995, University of Bern, All Rights Reserved.

[5] Hereafter, "four data bases."

Table 2 The "Trimmed" Data Bases and Their Variations

Database	Subject	Variation	Total
MIT	16	3	48
Olivetti	40	2	80
Weizmann	28	3	84
Bern	30	2	60
Combined	114	2, 3	272

Fig. 7. Examples of faces in the Weizmann data base.

B. Experimental Results

The eigenface, autoassociation and classification nets, and elastic matching algorithms were run on each of the four individual data bases as well as on the combined data base. The former was intended to test these algorithms' robustness over different data bases, while the latter was intended to test these algorithms' efficacy on a relatively large data base (in terms of the number of subjects).

In the experiments, the images were divided into training and test sets. To describe how the training samples were picked, we first need to describe how the data bases are organized. The four individual data bases (MIT, Olivetti, Weizmann, and Bern) have the same organization, and the combined data base, being the combination of the four, also has that organization. Specifically, in each data base, every subject is photographed under a number of *settings*, say, settings $1, 2, \cdots, K$. Here, a setting amounts to a particular combination of lighting, background, etc. Setting 1 is usually taken as the "standard." Fig. 7 shows the pictures of two subjects under three different settings from the Weizmann data base. The training set in our studies was formed by picking, for each subject, the picture taken under setting 1. Implementation details of the three algorithms are summarized in Table 3.

Table 3 The Implementation Details. (a) Eigenface. (b) Elastic Matching. (c) The Autoassociation and Classification Networks

Database	Number of Eigenfaces Used
MIT	15
Olivetti	39
Weizmann	27
Bern	29
Combined	113

(a)

Size of Template	Node Spacing in Pixels	Lambda	Node Deformation Range in Pixels
7x10	10	0.0003	11x11

* For all databases

(b)

Database	Number of Hidden Layer Nodes in Auto-Association Net	Number of Input/Output Nodes in Classification Net
MIT	16	16
Olivetti	40	40
Weizmann	28	28
Bern	30	30

* No hidden layers in classification nets
The number of input/output nodes in the auto-association net is 23x28 for all databases.
For both of the auto-association and classification nets, the learning rate is 0.4, and the minimum error is 0.0002, and the maximum number of training cycle is 100,000.

(c)

Table 4 Classification Results for Four Individual Data Bases

Database	Eigenface	Elastic Matching	Auto-Association and Classification Networks
MIT	97%	97%	72%
Olivetti	80%	80%	20%
Weizmann	84%	100%	41%
Bern	87%	93%	43%

The results of the face-recognition experiments are summarized in Tables 4 and 5, from which the following can be observed.

1) On the Four Individual Data Bases:

- The eigenface did very well on the MIT data base (97% recognition) and the Bern data base (87%). Its performance on the two other individual data bases was somewhat less satisfactory. To understand why,

Table 5 Classification Results for the Combined Data Bases

Eigenface	Elastic Matching
66%	93%

we first recall that the eigenface is essentially a technique that implements the minimum distance classifier of (6), which is optimal if the lighting variation between the training and testing samples can be modeled as zero-mean AWGN [see (8) and (9)]. It might still work if the mean is nonzero but small. When the lighting variation is not small and is spatiall varying, however, as it is for the Weizmann data base (Fig. 7), the zero-mean AWGN model of (9) is invalid and the minimum distance classifier can deviate significantly from the optimal ML classifier of (8), thereby resulting in significant performance deterioration. A more intuitive explanation: when the lighting variation is not small, it could introduce a large bias in the distance calculation. In such cases, the distance between two face images is dominated by the difference in their lighting conditions rather than the differences between the two faces, thereby rendering the distance an unreliable measure for face recognition. Brunelli and Poggio [21] proposed to take the derivative of the images to reduce biases caused by lighting change. When the lighting change is spatially varying, however—for example, half bright and half dark, as in the Weizmann data base (Fig. 7)—the derivative will introduce its own biases at the boundary of the lighting change.

- The elastic matching did very well on all data bases (93–100% recognition) except on the Olivetti, where its performance was acceptable (80% recognition). The Olivetti data base contains some scale and rotation variations (see the last row of Fig. 6), while our current elastic matching software could deal only with position, lighting, and expression variations. While competitive to the eigenface on all data bases, it did particularly well on the Weizmann data base (100% versus the eigenface's 84%). Two factors contributed to this good performance. First, the Gabor features, being the output of bandpass filters, are closely related to derivatives and are therefore less sensitive to lighting change. Second, the elastic matching uses features only at key points of an image rather than the entire image. Hence, biases at other points in the image do not contribute to the distance calculation.
- The performance of the autoassociation and classification nets was not satisfactory. There are several contributing factors. First, to have good performance, this technique requires that the lighting variations be small. All but the MIT data base, however, contain significant lighting variations. Second, this technique also requires a lot of preprocessing; for example, that the images be cut so that they only contain faces but not background. This was not done since the other two techniques do not have such requirements. Indeed, the only preprocessing done was a subsampling of the images to reduce the number of input nodes in the autoassociation net (the original image size is too large for the net to be trained within a reasonable amount of computer time). Third, the autoassociation and classification nets were nonlinear and contain many nodes and weights. It is possible that after a very long training period, the weights still did not converge to the optimal eigenface solution described in Section II-C. Last, from the analysis of Section II-C, the performance of the autoassociation and classification nets is upper bounded by that of the eigenface.

2) On the Combined Data Base:

- The performance of the eigenface deteriorated considerably, to 66% recognition. This is because there are significant lighting variations *between* the four data bases that make up the combined data base in addition to the lighting variations *within* each individual data base. As a result, the lighting-change-related distance biases are much larger for the combined data base than those for the four individual data bases, This makes image distance a highly unreliable measure of face differences (see also the discussion of the eigenface's performance on individual data bases).
- The elastic matching performed well, with 93% recognition. The main reason for this robust performance is that the elastic matching is relatively insensitive to the various lighting changes described above.
- The autoassociation and classification nets were not tested on the combined data base since they were unlikely to achieve dramatically better performance than that on the individual data bases but doing so would have required significantly more computation effort.

C. Advantages and Disadvantages

From the analysis of Section II and the experimental results presented above, what are the relative advantages/disadvantages of the three techniques studied in this paper?

From Section II, the recognition performance of the eigenface should be the same as the minimum distance classifier of (6), since the Euclidean distance is invariant to changes of orthonormal basis. As the minimum distance classifier, it works well when the face images have relatively small lighting and moderate expression variations. The weakness of this technique is that its performance deteriorates when face-position and lighting variations in the images cannot be characterized as "very small." Indeed,

if the eigenface is used in a practical system, a front end that provides scale, position, and lighting compensations is needed. Does the eigenface offer any computational advantages (efficiency)? The answer is: it depends. If all the face images in the data base are already stored as eigenfeature vectors and if the dimension of these vectors is not large, the answer is yes. On the other hand, if the images are stored as images, using (6) directly requires much less computation time.

The most important advantage of elastic matching is that it is relatively insensitive to variations in lighting, face position, and expression. While such robustness comes from the use of the Gabor features, the rigid matching stage, and the deformable matching stage, it also comes from the fact that only information at key positions of the image, rather than the entire image, is used in the face template. Another advantage of the elastic matching is easy data base expansion: when a new subject (new template) is added, there is no need to modify templates already in the data base. This may not be the case for the eigenface: when new subjects are added to a face data base, the eigenfaces may need to be recalculated, unless one assumes that the existing eigenfaces are "universal," an assumption that may or may not be valid. A disadvantage of the elastic matching is that it may require more computational effort than the eigenface. Its superior performance, however, seems to justify the additional computational effort. Furthermore, techniques such as multiresolution processing can be investigated to make it more efficient.

Last, the performance of the autoassociation and classification nets is upper bounded by that of the eigenface approach (see Section II-C). Due to practical implementation difficulties, however, the eigenface algorithm is the more preferable of the two.

IV. Summary

In this paper, a comparative study has been performed for three of the latest face-recognition techniques, namely, eigenface, autoassociation and classification networks, and elastic matching. First, these techniques were analyzed under a statistical decision framework. Then they were evaluated experimentally on four different data bases of moderate subject size and a combined data base of more than 100 subjects. Our results indicate that the eigenface algorithm, which is essentially a minimum distance classifier, works well when lighting variation is small. Its performance deteriorates significantly as lighting variation increases. The reason for this deterioration is that lighting variation introduces biases in distance calculations. When such biases are large, the image distance is no longer a reliable measure of face difference. The elastic matching algorithm, on the other hand, is insensitive to lighting, face-position, and expression variations and therefore is more versatile. This owes to the Gabor features, which are insensitive to lighting variation, rigid, and deformable matching, which allows for position and expression variation, and the fact that only features at key points in the image, rather than the entire image, are used. The performance of the autoassociation and classification nets is upper bounded by that of the eigenface but is more difficult to implement in practice.

For future work, it would be of interest to investigate ways to make the elastic matching more computationally efficient.

Appendix

Since in any realistic applications a face data base usually contains a large number of individuals (e.g., more than 100 in our work), it is desirable to automate the process of constructing face templates. According to [18], this can be done by using the following simple procedure.

A. Automated Template Extraction

1) Pick the face image of an individual and generate a face vector field using Gabor filters, as in (26). Denote this vector field as $\mathbf{x}$.
2) Place a coarse-grid lattice $\mathbf{S}_1$ "by hand" on $\mathbf{x}$ such that the vertices are close to important facial features such as the eyes [see Fig. 4(f)].
3) Collect the feature vectors at vertices of $\mathbf{S}_1$ to form the template $\mathbf{c}$.
4) Pick another face image (of a different person) and generate a face vector field, denoted as $\mathbf{y}$.
5) Perform template matching between the $\mathbf{c}$ generated in steps 1)–3) with $\mathbf{y}$ using the rigid matching of Section II-D.
6) When a best match is found, collect the feature vectors at vertices of $\mathbf{S}_1$ to form the template of $\mathbf{y}$.
7) Repeat steps 4)–6).

A face template generated according to the above procedure can be further improved by noticing that if the grid of $\mathbf{S}_1$ is allowed to deform, the feature vectors in the template will correspond more precisely with key facial features. Specifically, let $\mathbf{c}$ be the template generated from a face vector field $\mathbf{x}$ using the above procedure. An improved $\mathbf{c}$ can be found by finding the best match between $\mathbf{c}$ and $\mathbf{x}$ using the deformable matching operation of Section II-D. Notice, however, that the energy function needs to be different. Specifically, let $\mathbf{M}$ be a mapping between $\mathbf{S}_1$ and $\mathbf{S}$. The energy function should be

$$E(\mathbf{M}) = -\sum_i ||\mathbf{x}_j||^2 \tag{31}$$

where the sum is over all $i \in \mathbf{S}_1$, $j = M(i)$, and $\mathbf{c}_i = \mathbf{x}_j$. This energy function has an intuitive appeal: $\mathbf{S}_1$ should be deformed in such a way that the vertices correspond to points in the image where the Gabor features (i.e., edge features) are more prominent (i.e., large $||\mathbf{c}_i||$).

References

[1] A. Samal and P. A. Iyengar, "Automatic recognition and analysis of human faces and facial expressions: A survey," *Pattern Recognit.*, vol. 25, pp. 65–77, 1992.

[2] D. Valentin, H. Abdi, A. J. O'Toole, and G. W. Cottrell, "Connectionist models of face processing: A survey," *Pattern Recognit.*, vol. 27, pp. 1209–1230, 1994.
[3] R. Chellappa, C. L. Wilson, and S. Sirohey, "Human and machine recognition of faces: A survey," *Proc. IEEE*, vol. 83, pp. 705–741, May 1995.
[4] L. Sirovich and M. Kirby, "Low-dimensional procedure for the characterization of human face image," *J. Opt. Cos. Amer.*, vol. 4, pp. 519–524, 1987.
[5] G. Strang, *Linear Algebra and Its Applications*, 3rd ed. Orlando, FL: Harcourt Brace Jovanovich, 1986.
[6] M. A. Turk and A. P. Pentland, "Face recognition using eigenfaces," in *Proc. IEEE Computer Society Conf. Computer Vision and Pattern Recognition*, Maui, Hawaii, 1991, pp. 586–591.
[7] A. P. Pentland, B. Moghaddam, T. Starner, and M. A. Turk, "View-based and modular eigenspaces for face recognition," in *Proc. IEEE Computer Society Conf. Computer Vision and Pattern Recognition*, Seattle, WA, 1994, pp. 84–91.
[8] C. von der Malburg, "The correlation theory in brain function," reprinted in *Models of Neural Networks II*, E. Domany, J. L. Von Hemmen, and K. Schulten, Eds. Berlin: Springer-Verlag, 1994, ch. 2, pp. 95–119.
[9] M. Lades, J. Vorbruggen, J. Buhmann, J. Lange, C. V. D. Malburg, and R. Wurtz, "Distortion invariant object recognition in the dynamic link architecture," *IEEE Trans. Comput.*, vol. 42, no. 3, pp. 300–311, 1993.
[10] G. W. Cottrell and M. Fleming, "Face recognition using unsupervised feature extraction," in *Proc. Int. Neural Network Conf.*, vol. 1, Paris, France, July 9–13, 1990, pp. 322–325.
[11] J. Tou and R. Gonzalez, *Pattern Recognition Principles*. Reading, MA: Addison-Wesley, 1979.
[12] J. Gunning and N. Murphy, "Neural network based classification using automatically selected features sets," *IEE Conf. Series 359*, pp. 29–33, 1992.
[13] K. I. Diamantaras and S. Y. Kung, *Principal Component Neural Networks, Theory and Applications*. New York: Wiley-InterScience, 1996.
[14] H. Bourlard and Y. Kamp, "Auto-association by multilayer perceptrons and singular value decomposition," *Biological Cybern.*, vol. 59, pp. 291–294, 1988.
[15] D. E. Rumelhart and J. L. McClellan, *Parallel and Distributed Processing*, vol. 1. Cambridge, MA: MIT Press, 1986.
[16] L. Sirovich and M. Kirby, "Low-dimensional procedure for the characterization of human face," *J. Opt. Soc. Amer.*, vol. 4, pp. 519–524, 1987.
[17] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*, 3rd ed. New York: McGraw-Hill, 1991.
[18] M. Lades, "Invariant object recognition with dynamical links, robust to variations in illumination," *Dissertation zur Erlangung des Doktorgrades in den Naturwissenschaften*, vorgelegt von Martin Lades aus Nurnberg, Bochum, 1994.
[19] K. Fukunaga, *Introduction to Statistical Pattern Recognition*, 2nd ed. New York: Academic, 1990.
[20] R. G. Gallager, *Information Theory and Reliable Communication*. New York: Wiley, 1968.
[21] R. Brunelli and T. Poggio, "Face recognition: Features vs. templates," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 15, pp. 1042–1052, Oct. 1993.
[22] D. H. Ballard and C. M. Brown, *Computer Vision*. Englewood Cliffs, NJ: Prentice-Hall, 1982.
[23] J. J. Atick, P. A. Griffin, and N. A. Redlich, "Face recognition from live video," *Advance Imaging*, vol. 10, no. 5, pp. 58–62, 1995.
[24] J. J. Atick, P. A. Griffin, and A. N. Redlich, "Statistical approach to shape from shading: Reconstruction of 3D face surfaces from single 2D images," *Neural Computation*, to be published.
[25] B. Moghaddam, C. Nastar, and A. Pentland, "A Bayesian similarity measure for direct image matching," Massachusetts Institute of Technology, Cambridge, MIT Media Lab PCS TR393, Aug. 1996.
[26] S. Y. Kung, M. Fang, S. P. Liou, M. Y. Chiu, and J. S. Taur, "Decision-based neural network for face recognition system," in *Proc. ICIP'96*, vol. I, Washington, D.C., Oct. 1995, pp. 430–437.
[27] B. K. P. Horn, *Robot Vision*. Cambridge, MA: MIT Press, 1986.

Jun Zhang (Member, IEEE) received the B.S. degree in electrical and computer engineering from Harbin Shipbuilding Engineering Institute, Harbin, Heilongjiang, China in 1982. He was a graduate student in the Radio Electronic Department, Tsinghua University, China, until receiving a scholarship from the Li Foundation, Glen Cove, NY. He received the M.S. and Ph.D. degrees in electrical engineering from Rensselaer Polytechnic Institute, Troy, NY, in 1985 and 1988, respectively.

He currently is an Associate Professor with the Department of Electrical Engineering and Computer Science, University of Wisconsin–Milwaukee. His research interests include image processing and computer vision, signal processing, and digital communications.

Yong Yan (Member, IEEE) received the B.S. degree in wireless communication from Northeastern University, Shenyang, China, in 1982 and the M.S. degree in biomedical/electrical engineering from Xi'an Jiaotong University, Xi'an, China, in 1986. He currently is pursuing the Ph.D. degree at the University of Wisconsin–Milwaukee.

From 1986 to 1990, he conducted research and development on medical image processing and recognition in the image processing center of Xi'an Jiaotong University. From 1991 to 1994, he was a Research and Teaching Assistant in the Electrical Engineering and Computer Science Department, University of Wisconsin–Milwaukee. In 1994, he joined Intelligent Medical Imaging, Inc. as a Senior Software Engineer, where he is responsible for product development in medical imaging. His interests are computer vision, digital signal processing, and computer graphics.

Martin Lades (Member, IEEE) received the Diplom-Physiker degree from the University of Erlangen, Germany, in 1988 with a topic related to optical computing. He received the Ph.D. degree in physik from the University of Bochum, Germany, in 1995, for which he conducted research in the area of neural networks at the University of Duesseldorf, Germany (robotics), the University Southern California, Los Angeles, and Bochum, Germany (object recognition).

In 1994, he joined the Institute for Scientific Computing Research (ISCR), Lawrence Livermore National Laboratory, Livermore, CA, as a Postdoctoral Research Fellow. He currently is the Principal Investigator of ISCR's computer vision research. His interests include computer vision and object/face recognition, artificial neural networks and neuromorphic engineering, and specifically artificial sensory systems, such as silicon retinas, and their applications.