1

Probabilistic Visual Learning for Object Representation

Baback Moghaddam and *Alex Pentland*
Massachusettes Institute of Technology

Abstract

We present an unsupervised technique for visual learning which is based on density estimation in high-dimensional spaces using an eigenspace decomposition. Two types of density estimates are derived for modeling the training data: a multivariate Gaussian (for unimodal distributions) and a Mixture-of-Gaussians model (for multimodal distributions). These probability densities are then used to formulate a maximum-likelihood *estimation framework for visual search and target detection for automatic object recognition and coding. Our learning technique is applied to the probabilistic visual modeling, detection, recognition, and coding of human faces and non-rigid objects such as hands.*

1. Introduction

Visual attention is the process of restricting higher-level processing to a subset of the visual field, referred to as the *focus-of-attention* (FOA). The critical component of visual attention is the *selection* of the FOA. In humans this process is not based purely on bottom-up processing and is in fact goal-driven. The measure of interest or *saliency* is modulated by the behavioral state and the demands of the particular visual task that is currently active.

Palmer [24] has suggested that visual attention is the process of locating the object of interest and placing it in a *canonical* (or object-centered) reference frame suitable for recognition (or template matching). We have developed a computational technique for automatic object recognition, which is in accordance with Palmer's model of visual attention (see section 4.1.). The system uses a probabilistic formulation for the estimation of the position and scale of the object in the visual field and remaps the FOA to an object-centered reference frame, which is subsequently used for recognition and verification.

At a simple level the underlying mechanism of attention during a visual search task can be based on a spatiotopic *saliency* map $S(i, j)$ which is a function of the

(a) (b) (c) (d)

Figure 1: *(a) input image, (b) face detection, (c) input image, (d) hand detection*

image information in a local region R

$$S(i,j) = f\left[\{I(i+r, j+c) : (r,c) \in R\}\right] \tag{1}$$

For example saliency maps have been constructed which employ spatio-temporal changes as cues for foveation [1] or other low-level image features such as local symmetry for detection of interest points [30]. However bottom-up techniques based on low-level features lack *context* with respect to high-level visual tasks such as object recognition. In a recognition task, the selection of the FOA is driven by higher-level goals and therefore requires internal representations of an object's appearance and a means of comparing candidate objects in the FOA to the stored object models.

In view-based recognition (as opposed to 3D geometric or invariant-based recognition), the saliency can be formulated in terms of visual similarity using a variety of metrics ranging from simple template matching scores to more sophisticated measures using, for example, robust statistics for image correlation [5]. In this paper, however, we are primarily interested in saliency maps which have a *probabilistic* interpretation as object-class membership functions or *likelihoods*. These likelihood functions are learned by applying density estimation techniques in complementary subspaces obtained by an eigenvector decomposition. Our approach to this learning problem is *view-based* — *i.e.,* the learning and modeling of the visual appearance of the object from a (suitably normalized and preprocessed) set of training imagery. Figure 1 shows examples of the automatic selection of FOA for detection of faces and hands. In each case, the target object's probability distribution was *learned* from training views and then subsequently used in computing likelihoods for detection. The face representation is based on appearance (normalized grayscale image) whereas the hand's representation is based on the shape of its contour. The maximum likelihood (ML) estimates of position and scale are shown in the figure by the cross-hairs and bounding box, respectively.

1.1. Object Detection

The standard detection paradigm in image processing is that of normalized correlation or template matching. However this approach is only optimal in the simplistic case of a *deterministic* signal embedded in additive white Gaussian noise. When we begin to consider a target *class* detection problem — *e.g,* finding a generic human face or a human hand in a scene — we must incorporate the underlying probability distribution of the object. Subspace methods and eigenspace decompositions are particularly well-suited to such a task since they provide a compact and *parametric* description of the object's appearance and also automatically identify the *degrees-of-freedom* of the underlying statistical variability.

In particular, the eigenspace formulation leads to a powerful alternative to standard detection techniques such as template matching or normalized correlation. The reconstruction error (or residual) of the eigenspace decomposition (referred to as the "distance-from-face-space" in the context of the work with "eigenfaces" [34]) is an effective indicator of similarity. The residual error is easily computed using the projection coefficients and the original signal energy. This detection strategy is equivalent to matching with a linear combination of *eigentemplates* and allows for a greater range of distortions in the input signal (including lighting, and moderate rotation and scale). In a statistical signal detection framework, the use of eigentemplates has been shown to yield superior performance in comparison with standard matched filtering [18][26].

In [26] we used this formulation for a modular eigenspace representation of facial features where the corresponding residual — referred to as "distance-from-*feature*-space" or DFFS — was used for localization and detection. Given an input image, a saliency map was constructed by computing the DFFS at each pixel. When using M eigenvectors, this requires M convolutions (which can be efficiently computed using an FFT) plus an additional local energy computation. The global minimum of this distance map was then selected as the best estimate of the location of the target.

In this paper we will show that the DFFS can be interpreted as an estimate of a marginal component of the probability density of the object and that a complete estimate must also incorporate a second marginal density based on a complementary "distance-*in*-feature-space" (DIFS). Using our estimates of the object densities, we formulate the problem of target detection from the point of view of a maximum likelihood (ML) estimation problem. Specifically, given the visual field, we estimate the position (and scale) of the image region which is most representative of the target of interest. Computationally this is achieved by sliding an m-by-n observation window throughout the image and at each location computing the *likelihood* that the local subimage $\mathbf{x}$ is an instance of the target class Ω — *i.e.,* $P(\mathbf{x}|\Omega)$. After this probability map is computed, we select the location corresponding to the highest likelihood as our ML estimate of the target location. Note that the likelihood map can be evaluated over the entire parameter space affecting the object's appearance which can include transformations such as scale and rotation.

1.2. Relationship to Previous Research

In recent years, computer vision research has witnessed a growing interest in eigenvector analysis and subspace decomposition methods. In particular, eigenvector decomposition has been shown to be an effective tool for solving problems which use high-dimensional representations of phenomena which are intrinsically low-dimensional. This general analysis framework lends itself to several closely related formulations in object modeling and recognition which employ the *principal modes* or characteristic *degrees-of-freedom* for description. The identification and parametric representation of a system in terms of these principal modes is at the core of recent advances in physically-based modeling [25], correspondence and matching [32], and parametric descriptions of shape [7].

Eigenvector-based methods also form the basis for data analysis techniques in pattern recognition and statistics where they are used to extract low-dimensional subspaces comprised of statistically uncorrelated variables which tend to simplify tasks such as classification. The Karhunen-Loeve Transform (KLT) [19] and Principal Components Analysis (PCA) [14] are examples of eigenvector-based techniques which are commonly used for dimensionality reduction and feature extraction in pattern recognition.

In computer vision, eigenvector analysis of *imagery* has been used for characterization of human faces [17] and automatic face recognition using "eigenfaces" [34][26]. More recently, principal component analysis of imagery has also been applied for robust target detection [26][6], nonlinear image interpolation [3], visual learning for object recognition [22][36], as well as visual servoing for robotics [23].

Specifically, Murase & Nayar [22] used a low-dimensional *parametric* eigenspace for recovering object identity and pose by matching views to a spline-based hypersurface. Nayar *et al.* [23] have extended this technique to visual feedback control and servoing for a robotic arm in "peg-in-the-hole" insertion tasks. Pentland *et al.* [26] proposed a view-based multiple-eigenspace technique for face recognition under varying pose as well as for the detection and description of facial features. Similarly, Burl *et al.* [6] used Bayesian classification for object detection using a feature vector derived from principal component images. Weng [36] has proposed a visual learning framework based on the KLT in conjunction with an optimal linear discriminant transform for learning and recognition of objects from 2D views.

However, these authors (with the exception of [26]) have used eigenvector analysis primarily as a dimensionality reduction technique for subsequent modeling, interpolation, or classification. In contrast, our method uses an eigenspace decomposition as an integral part of an efficient technique for probability density estimation of high-dimensional data.

2. Density Estimation in Eigenspace

In this section we present our recent work using eigenspace decompositions for object representation and modeling. Our learning method estimates the *complete* probability distribution of the object's appearance using an eigenvector decomposition of the image space. The desired target density is decomposed into two components: the density in the principal subspace (containing the traditionally-defined principal components) and its orthogonal complement (which is usually discarded in standard PCA). We derive the form for an optimal density estimate for the case of Gaussian data and a near-optimal estimator for arbitrarily complex distributions in terms of a Mixture-of-Gaussians density model.

We note that this learning method differs from *supervised* visual learning with function approximation networks [28] in which a hypersurface representation of an input/output map is automatically learned from a set of training examples. Instead, we use a probabilistic formulation which combines the two standard paradigms of *unsupervised* learning — PCA and density estimation — to arrive at a computationally feasible estimate of the class conditional density function.

Specifically, given a set of training images $\{\mathbf{x}^t\}_{t=1}^{N_T}$, from an object class Ω, we wish to estimate the class membership or *likelihood* function for this data — *i.e.,* $P(\mathbf{x}|\Omega)$. In this section, we examine two density estimation techniques for visual learning of high-dimensional data. The first method is based on the assumption of a Gaussian distribution while the second method generalizes to arbitrarily complex distributions using a Mixture-of-Gaussians density model. Before introducing these estimators we briefly review eigenvector decomposition as commonly used in PCA.

2.1. Principal Component Imagery

Given a training set of m-by-n images $\{I^t\}_{t=1}^{N_T}$, we can form a training set of vectors $\{\mathbf{x}^t\}$, where $\mathbf{x} \in \mathcal{R}^{N=mn}$, by lexicographic ordering of the pixel elements of each image I^t. The basis functions for the KLT [19] are obtained by solving the eigenvalue problem

$$\Lambda \; = \; \Phi^T \Sigma \Phi \tag{2}$$

where Σ is the covariance matrix, Φ is the eigenvector matrix of Σ and Λ is the corresponding diagonal matrix of eigenvalues. The unitary matrix Φ defines a coordinate transform (rotation) which *decorrelates* the data and makes explicit the *invariant subspaces* of the matrix operator Σ. In PCA, a partial KLT is performed to identify the largest-eigenvalue eigenvectors and obtain a principal component feature vector $\mathbf{y} = \Phi_{\mathrm{M}}^T \, \tilde{\mathbf{x}}$, where $\tilde{\mathbf{x}} = \mathbf{x} - \bar{\mathbf{x}}$ is the mean-normalized image vector and Φ_M is a submatrix of Φ containing the principal eigenvectors. PCA can be seen as a linear transformation $\mathbf{y} = \mathcal{T}(\mathbf{x}) : \mathcal{R}^N \rightarrow \mathcal{R}^M$ which extracts a lower-dimensional subspace of the KL basis corresponding to the maximal eigenvalues. These principal components preserve the major linear correlations in the data and discard the minor ones.[1]

[1] In practice the number of training images N_T is far less than the dimensionality of the imagery N, consequently the covariance matrix Σ is singular. However, the first $M < N_T$ eigenvectors can always be computed (estimated) from N_t samples using, for example, a Singular Value Decomposition [12].

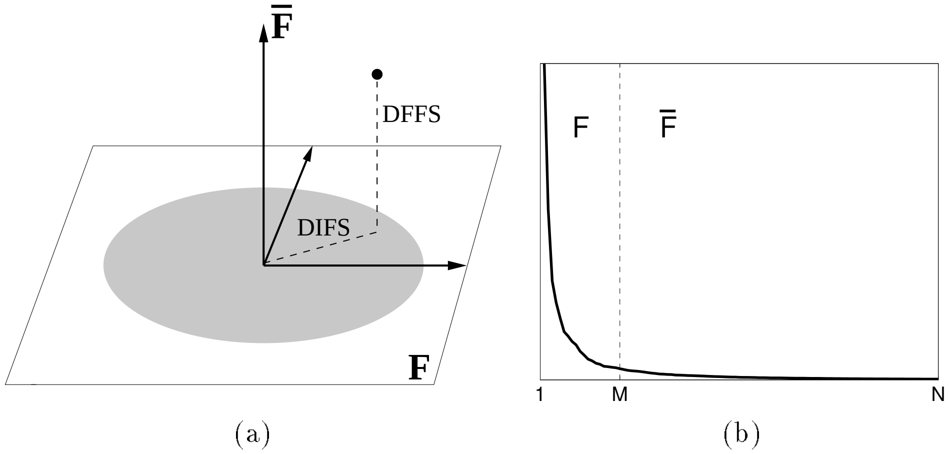


Figure 2: *(a) Decomposition into the principal subspace F and its orthogonal complement $\bar{F}$ for a Gaussian density, (b) a typical eigenvalue spectrum and its division into the two orthogonal subspaces.*

By ranking the eigenvectors of the KL expansion with respect to their eigenvalues and selecting the first M principal components we form an orthogonal decomposition of the vector space $\mathcal{R}^N$ into two mutually exclusive and complementary subspaces: the principal subspace (or feature space) $F = \{\Phi_i\}_{i=1}^{M}$ containing the principal components and its orthogonal complement $\bar{F} = \{\Phi_i\}_{i=M+1}^{N}$. This orthogonal decomposition is illustrated in Figure 2(a) where we have a prototypical example of a distribution which is embedded entirely in F. In practice there is always a signal component in $\bar{F}$ due to the minor statistical variabilities in the data or simply due to the observation noise which affects every element of $\mathbf{x}$.

In a partial KL expansion, the residual reconstruction error is defined as

$$\epsilon^2(\mathbf{x}) = \sum_{i=M+1}^{N} y_i^2 = ||\tilde{\mathbf{x}}||^2 - \sum_{i=1}^{M} y_i^2 \tag{3}$$

and can be easily computed from the first M principal components and the L_2 norm of the mean-normalized image $\tilde{\mathbf{x}}$. Consequently the L_2 norm of every element $\mathbf{x} \in \mathcal{R}^N$ can be decomposed in terms of its projections in these two subspaces. We refer to the component in the orthogonal subspace $\bar{F}$ as the "distance-from-feature-space" (DFFS) which is a simple Euclidean distance and is equivalent to the residual error $\epsilon^2(\mathbf{x})$ in Eq.(3). The component of $\mathbf{x}$ which lies *in* the feature space F is referred to as the "distance-in-feature-space" (DIFS) but is generally not a distance-based norm, but can be interpreted in terms of the probability distribution of y in F.

2.2. Gaussian Densities

We begin by considering an optimal approach for estimating high-dimensional Gaussian densities. We assume that we have (robustly) estimated the mean $\bar{\mathbf{x}}$ and covariance Σ of the distribution from the given training set $\{\mathbf{x}^t\}$.[2] Under this assumption, the likelihood of an input pattern $\mathbf{x}$ is given by

$$P(\mathbf{x}|\Omega) = \frac{\exp\left[-\frac{1}{2}(\mathbf{x}-\bar{\mathbf{x}})^T\Sigma^{-1}(\mathbf{x}-\bar{\mathbf{x}})\right]}{(2\pi)^{N/2}\,|\Sigma|^{1/2}} \tag{4}$$

The sufficient statistic for characterizing this likelihood is the *Mahalanobis* distance

$$d(\mathbf{x}) \;=\; \tilde{\mathbf{x}}^T\Sigma^{-1}\tilde{\mathbf{x}} \tag{5}$$

where $\tilde{\mathbf{x}} = \mathbf{x}-\bar{\mathbf{x}}$. However, instead of evaluating this quadratic product explicitly, a much more efficient and robust computation can be performed, especially with regard to the matrix inverse Σ^{-1}. Using the eigenvectors and eigenvalues of Σ we can rewrite Σ^{-1} in the diagonalized form

$$\begin{aligned} d(\mathbf{x}) &= \tilde{\mathbf{x}}^T\Sigma^{-1}\tilde{\mathbf{x}} \\ &= \tilde{\mathbf{x}}^T\left[\mathbf{\Phi}\Lambda^{-1}\mathbf{\Phi}^T\right]\tilde{\mathbf{x}} \\ &= \mathbf{y}^T\Lambda^{-1}\mathbf{y} \end{aligned} \tag{6}$$

where $\mathbf{y} = \mathbf{\Phi}^T\tilde{\mathbf{x}}$ are the new variables obtained by the change of coordinates in a KLT. Because of the diagonalized form, the *Mahalanobis* distance can also be expressed in terms of the sum

$$d(\mathbf{x}) \;=\; \sum_{i=1}^{N}\frac{y_i^2}{\lambda_i} \tag{7}$$

In the KLT basis, the *Mahalanobis* distance in Eq.(5) is conveniently *decoupled* into a weighted sum of uncorrelated component energies. Furthermore, the likelihood becomes a *product* of independent separable Gaussian densities. Despite its simpler form, evaluation of Eq.(7) is still computationally infeasible due to the high-dimensionality. We therefore seek to *estimate* $d(\mathbf{x})$ using only M projections. Intuitively, an obvious choice for a lower-dimensional representation is the principal subspace indicated by PCA which captures the major degrees of statistical variability in the data.[3] Therefore, we divide the summation into two

[2] In practice, a full rank N-dimensional covariance Σ can not be estimated from N_T independent observations when $N_T < N$. But as we shall see our estimator does not require the full covariance, but only its first M principal eigenvectors where $M < N_T$.

[3] We will see shortly that given the typical eigenvalue spectra observed in practice (*e.g.*, Figure 2(b)), this choice is optimal for a different reason: it minimizes the information-theoretic *divergence* between the true density and our estimate of it.

independent parts corresponding to the principal subspace $F = \{\Phi_i\}_{i=1}^{M}$ and its orthogonal complement $\bar{F} = \{\Phi_i\}_{i=M+1}^{N}$

$$d(\mathbf{x}) \;=\; \sum_{i=1}^{M} \frac{y_i^2}{\lambda_i} \;+\; \sum_{i=M+1}^{N} \frac{y_i^2}{\lambda_i} \tag{8}$$

We note that the terms in the first summation can be computed by projecting $\mathbf{x}$ onto the M-dimensional principal subspace F. The remaining terms in the second sum $\{y_i\}_{i=M+1}^{N}$, however, can not be computed explicitly in practice because of the high-dimensionality. However, the *sum* of these terms is available and is in fact the DFFS quantity $\epsilon^2(\mathbf{x})$ which can be computed from Eq.(3). Therefore, based on the available terms, we can formulate an estimator for $d(\mathbf{x})$ as follows

$$\begin{aligned} \hat{d}(\mathbf{x}) \;&=\; \sum_{i=1}^{M} \frac{y_i^2}{\lambda_i} \;+\; \frac{1}{\rho}\left[\sum_{i=M+1}^{N} y_i^2\right] \\ &=\; \sum_{i=1}^{M} \frac{y_i^2}{\lambda_i} \;+\; \frac{\epsilon^2(\mathbf{x})}{\rho} \end{aligned} \tag{9}$$

where the term in the brackets is $\epsilon^2(\mathbf{x})$, which as we have seen can be computed using the first M principal components. We can therefore write the form of the likelihood estimate based on $\hat{d}(\mathbf{x})$ as the product of two marginal and independent Gaussian densities

$$\begin{aligned} \hat{P}(\mathbf{x}|\Omega) \;&=\; \left[\frac{\exp\left(-\frac{1}{2}\sum_{i=1}^{M}\frac{y_i^2}{\lambda_i}\right)}{(2\pi)^{M/2}\prod_{i=1}^{M}\lambda_i^{1/2}}\right] \cdot \left[\frac{\exp\left(-\frac{\epsilon^2(\mathbf{x})}{2\rho}\right)}{(2\pi\rho)^{(N-M)/2}}\right] \\ &=\; P_F(\mathbf{x}|\Omega)\; \hat{P}_{\bar{F}}(\mathbf{x}|\Omega) \end{aligned} \tag{10}$$

where $P_F(\mathbf{x}|\Omega)$ is the true marginal density in F-space and $\hat{P}_{\bar{F}}(\mathbf{x}|\Omega)$ is the estimated marginal density in the orthogonal complement $\bar{F}$-space. The optimal value of ρ can now be determined by minimizing a suitable cost function $J(\rho)$. From an information-theoretic point of view, this cost function should be the Kullback-Leibler divergence or *relative entropy* [9] between the true density $P(\mathbf{x}|\Omega)$ and its estimate $\hat{P}(\mathbf{x}|\Omega)$

$$J(\rho) \;=\; \int P(\mathbf{x}|\Omega)\, \log \frac{P(\mathbf{x}|\Omega)}{\hat{P}(\mathbf{x}|\Omega)}\, d\mathbf{x} \;=\; \mathrm{E}\left[\log \frac{P(\mathbf{x}|\Omega)}{\hat{P}(\mathbf{x}|\Omega)}\right] \tag{11}$$

Using the diagonalized forms of the *Mahalanobis* distance $d(\mathbf{x})$ and its estimate $\hat{d}(\mathbf{x})$ and the fact that $\mathrm{E}[y_i^2] = \lambda_i$, it can be easily shown that

$$J(\rho) \;=\; \frac{1}{2} \sum_{i=M+1}^{N} \left[\frac{\lambda_i}{\rho} - 1 + \log \frac{\rho}{\lambda_i}\right] \tag{12}$$

The optimal weight ρ^* can be then found by minimizing this cost function with respect to ρ. Solving the equation $\frac{\partial J}{\partial \rho} = 0$ yields

$$\rho^* \;=\; \frac{1}{N-M} \sum_{i=M+1}^{N} \lambda_i \tag{13}$$

which is simply the arithmetic average of the eigenvalues in the orthogonal subspace $\bar{F}$. [4] In addition to its optimality, ρ^* also results in an *unbiased* estimate of the *Mahalanobis* distance — *i.e.,* $\mathrm{E}[\hat{d}(\mathbf{x};\rho^*)] = \mathrm{E}[d(\mathbf{x})]$. This derivation shows that once we select the M-dimensional principal subspace F (as indicated, for example, by PCA), the optimal estimate of the sufficient statistic $\hat{d}(\mathbf{x})$ will have the form of Eq.(9) with ρ given by Eq.(13).

It is interesting to consider the minimal cost $J(\rho^*)$

$$J(\rho^*) \;=\; \frac{1}{2} \sum_{i=M+1}^{N} \log \frac{\rho^*}{\lambda_i} \tag{14}$$

from the point of view of the $\bar{F}$-space eigenvalues $\{\lambda_i : i = M+1, \cdots, N\}$. It is easy to show that $J(\rho^*)$ is minimized when the the $\bar{F}$-space eigenvalues have the *least* spread about their mean ρ^*. This suggests a strategy for selecting the principal subspace: choose F such that the eigenvalues associated with its orthogonal complement $\bar{F}$ have the least absolute deviation about their mean. In practice, the higher-order eigenvalues typically decay and stabilize near the observation noise variance. Therefore this strategy is usually consistent with the standard PCA practice of discarding the higher-order components since these tend to correspond to the "flattest" portion of the eigenvalue spectrum (see Figure 2(b)). In the limit, as the $\bar{F}$-space eigenvalues become exactly equal, the divergence $J(\rho^*)$ will be zero and our density estimate $\hat{P}(\mathbf{x}|\Omega)$ approaches the true density $P(\mathbf{x}|\Omega)$.

We note that in most applications it is customary to simply discard the $\bar{F}$-space component and simply work with $P_F(\mathbf{x}|\Omega)$. However, the use of the DFFS metric or equivalently the marginal density $P_{\bar{F}}(\mathbf{x}|\Omega)$ is critically important in formulating the likelihood of an observation $\mathbf{x}$ — especially in an object detection task — since there are an infinity of vectors which are *not* members of Ω which can have likely F-space projections. Without $P_{\bar{F}}(\mathbf{x}|\Omega)$ a detection system can result in a significant number of false alarms.

2.3. Multimodal Densities

In the previous section we assumed that probability density of the training images was Gaussian. This lead to a likelihood estimate in the form of a product of two independent multivariate Gaussian distributions (or equivalently the sum of two *Mahalanobis* distances: DIFS + DFFS). In our experience, the distribution

[4] Cootes *et al.* [8] have used a similar decomposition of the Mahalanobis distance but instead use an ad-hoc parameter value of $\rho = \frac{1}{2}\lambda_{M+1}$ as an approximation.

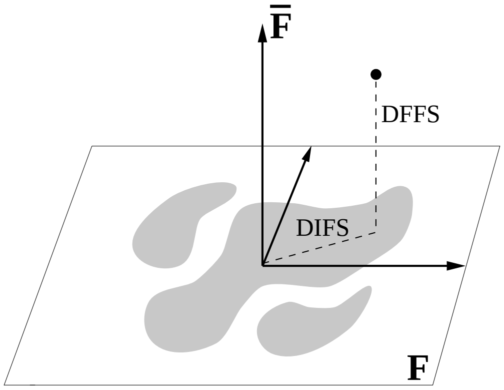


Figure 3: *Decomposition into the principal subspace F and its orthogonal complement $\bar{F}$ for an arbitrary density.*

of samples in the feature space is often accurately modeled by a single Gaussian distribution. This is especially true in cases where the training images are accurately aligned views of similar objects seen from a standard view (*e.g.,* aligned frontal views of human faces at the same scale and orientation). However, when the training set represents multiple views or multiple objects under varying illumination conditions, the distribution of training views in F-space is no longer unimodal. In fact the training data tends to lie on complex and non-separable low-dimensional manifolds in image space. One way to tackle this multimodality is to build a view-based (or object-based) formulation where separate eigenspaces are used for each view [26]. Another approach is to capture the complexity of these manifolds in a universal or *parametric* eigenspace using splines [22], or local basis functions [3].

If we assume that the $\bar{F}$-space components are Gaussian and independent of the principal features in F (this would be true in the case of pure observation noise in $\bar{F}$) we can still use the separable form of the density estimate $\hat{P}(\mathbf{x}|\Omega)$ in Eq.(10) where $P_F(\mathbf{x}|\Omega)$ is now an *arbitrary* density $P(\mathbf{y})$ in the principal component vector $\mathbf{y}$. Figure 3 illustrates the decomposition, where the DFFS is the residual $\epsilon^2(\mathbf{x})$ as before. The DIFS, however, is no longer a simple *Mahalanobis* distance but can nevertheless be interpreted as a "distance" by relating it to $P(\mathbf{y})$ — *e.g.,* as DIFS $= -\log P(\mathbf{y})$.

The density $P(\mathbf{y})$ can be estimated using a parametric mixture model. Specifically, we can model arbitrarily complex densities using a Mixture-of-Gaussians

$$P(\mathbf{y}|\Theta) \;=\; \sum_{i=1}^{N_c} \pi_i \; g(\mathbf{y}; \mu_i, \Sigma_i) \tag{15}$$

where $g(\mathbf{y}; \mu, \Sigma)$ is an M-dimensional Gaussian density with mean vector μ and covariance Σ, and the π_i are the mixing parameters of the components, satisfying $\sum \pi_i = 1$. The mixture is completely specified by the parameter $\Theta = \{\pi_i, \mu_i, \Sigma_i\}_{i=1}^{N_c}$. Given a training set $\{\mathbf{y}^t\}_{t=1}^{N_T}$ the mixture parameters can be estimated using the ML principle

$$\Theta^* = \operatorname{argmax} \left[\prod_{t=1}^{N_T} P(\mathbf{y}^t|\Theta) \right] \tag{16}$$

This estimation problem is best solved using the Expectation-Maximization (EM) algorithm [11] which consists of the following two-step iterative procedure:

- E-step:

$$h_i^k(t) = \frac{\pi_i^k g(\mathbf{y}^t; \mu_i^k, \Sigma_i^k)}{\sum_{j=1}^{N_c} \pi_j^k g(\mathbf{y}^t; \mu_j^k, \Sigma_i^k)} \tag{17}$$

- M-step:

$$\pi_i^{k+1} = \frac{\sum_{t=1}^{N_T} h_i^k(t)}{\sum_{i=1}^{N_c} \sum_{t=1}^{N_T} h_i^k(t)} \tag{18}$$

$$\mu_i^{k+1} = \frac{\sum_{t=1}^{N_T} h_i^k(t)\mathbf{y}^t}{\sum_{t=1}^{N_T} h_i^k(t)} \tag{19}$$

$$\Sigma_i^{k+1} = \frac{\sum_{t=1}^{N_T} h_i^k(t)(\mathbf{y}^t - \mu_i^{k+1})(\mathbf{y}^t - \mu_i^{k+1})^T}{\sum_{t=1}^{N_T} h_i^k(t)} \tag{20}$$

The E-step computes the *a posteriori* probabilities $h_i(t)$ which are the *expectations* of "missing" component labels $z_i(t) = \{0, 1\}$ which denote the membership of $\mathbf{y}^t$ in the i-th component. Once these expectations have been computed, the M-step maximizes the joint likelihood of the data *and* the "missing" variables $z_i(t)$. The EM algorithm is monotonically convergent in *likelihood* and is thus guaranteed to find a local maximum in the total likelihood of the training set. Further details of the EM algorithm for estimation of mixture densities can be found in [29].

Given our operating assumptions — that the training data is M-dimensional (at most) and resides solely in the principal subspace F with the exception of perturbations due to white Gaussian measurement noise, or equivalently that the $\bar{F}$-space component of the data is itself a separable Gaussian density — the estimate of the complete likelihood function $P(\mathbf{x}|\Omega)$ is given by

$$\hat{P}(\mathbf{x}|\Omega) \;=\; P(\mathbf{y}|\Theta^*)\; \hat{P}_{\bar{F}}(\mathbf{x}|\Omega) \tag{21}$$

where $\hat{P}_{\bar{F}}(\mathbf{x}|\Omega)$ is a Gaussian component density based on the DFFS, as before.

3. Maximum Likelihood Detection

The density estimate $\hat{P}(\mathbf{x}|\Omega)$ can be used to compute a local measure of target saliency at each spatial position (i,j) in an input image based on the vector $\mathbf{x}$ obtained by the lexicographic ordering of the pixel values in a local neighborhood R

$$S(i,j;\Omega) = \hat{P}(\mathbf{x}|\Omega)\;, \quad \mathbf{x} \;=\; \downarrow [\{I(i+r, j+c) : (r,c) \in R\}] \tag{22}$$

where $\downarrow [\bullet]$ is the operator which converts a subimage into a vector. The ML estimate of position of the target Ω is then given by

$$(i,j)^{\mathrm{ML}} \;=\; \mathrm{argmax}\; S(i,j;\Omega) \tag{23}$$

This ML formulation can be extended to estimate object scale with *multiscale* saliency maps. The likelihood computation is performed (in parallel) on linearly scaled versions of the input image $I^{(\sigma)}$ corresponding to a pre-determined set of scales $\{\sigma_1, \sigma_2, \cdots \sigma_n\}$

$$S(i,j,k;\;\Omega) = \hat{P}\left(\downarrow \{I^{(\sigma_k)}(\sigma_k i + r, \sigma_k j + c) : (r,c) \in R\} \mid \Omega\right) \tag{24}$$

where the ML estimate of the spatial and scale indices is defined by

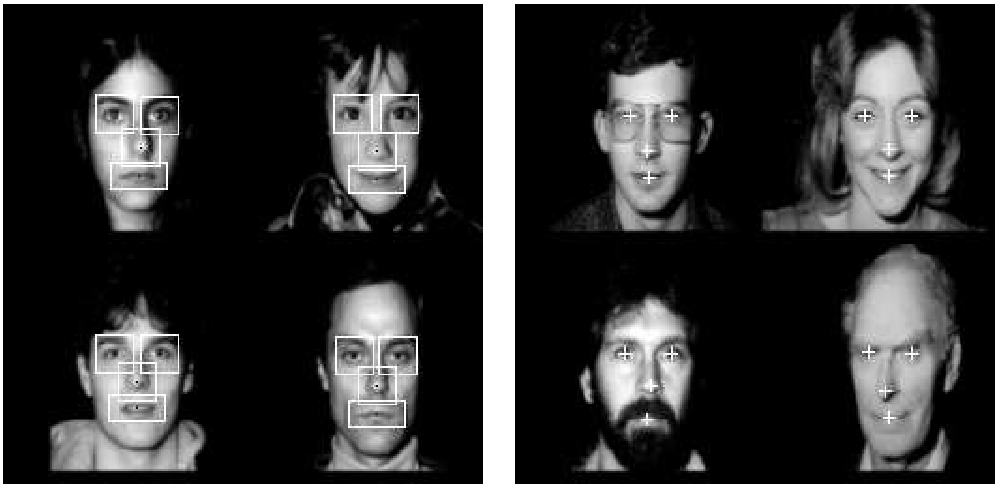
$$(i,j,k)^{\mathrm{ML}} \;=\; \mathrm{argmax}\; S(i,j,k\;;\;\Omega) \tag{25}$$

4. Applications

The above ML detection technique has been tested in the detection of complex natural objects including human faces, facial features (*e.g.,* eyes), and non-rigid articulated objects such as human hands. In this section we will present several examples from these application domains.

4.1. Faces

Over the years, various strategies for facial feature detection have been proposed, ranging from edge map projections [15], to more recent techniques using generalized symmetry operators [30] and multilayer perceptrons [35]. In any robust face processing system this task is critically important since a face must be first geometrically normalized by aligning its features with those of a stored model before recognition can be attempted.



(a) (b)

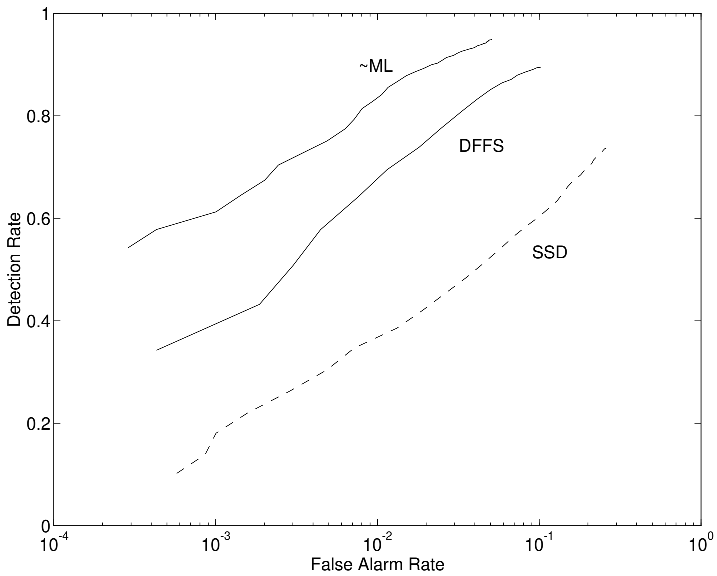
Figure 4: *(a) Examples of facial feature training templates and (b) the resulting typical detections.*

The eigentemplate approach to the detection of facial features in "mugshots" was proposed in [26], where the DFFS metric was shown to be superior to standard template matching for target detection. The detection task was the estimation of the position of facial features (the left and right eyes, the tip of the nose and the center of the mouth) in frontal view photographs of faces at fixed scale. Figure 4 shows examples of facial feature training templates and the resulting detections on the MIT Media Laboratory's database of 7,562 "mugshots".

We have compared the detection performance of three different detectors on approximately 7,000 test images from this database: a sum-of-square-differences (SSD) detector based on the average facial feature (in this case the left eye), an eigentemplate or DFFS detector and a ML detector based on $S(i,j;\Omega)$ as defined in section 2.2.. Figure 5(a) shows the *receiver operating characteristic* (ROC) curves for these detectors, obtained by varying the detection threshold independently for each detector. The DFFS and ML detectors were computed based on a 5-dimensional principal subspace. Since the projection coefficients were unimodal a Gaussian distribution was used in modeling the true distribution for the ML detector as in section 2.2.. Note that the ML detector exhibits the best detection vs. false-alarm tradeoff and yields the highest detection rate (95%). Indeed, at the *same* detection rate the ML detector has a false-alarm rate which is nearly 2 orders of magnitude lower than the SSD.

Figure 5(b) provides the geometric intuition regarding the operation of these detectors. The SSD detector's threshold is based on the *radial* distance between the average template (the origin of this space) and the input pattern. This leads to hyperspherical detection regions about the origin. In contrast, the DFFS detector

(a)



(b)

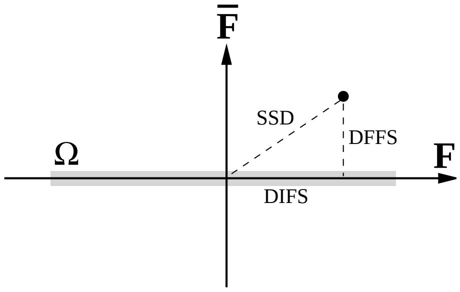


Figure 5: *(a) Detection performance of an SSD, DFFS and a ML detector, (b) geometric interpretation of the detectors.*

measures the orthogonal distance to F, thus forming planar acceptance regions about F. Consequently to accept valid object patterns in Ω which are very different from the mean, the SSD detector must operate with high thresholds which result in many false alarms. However, the DFFS detector can not discriminate between the object class Ω and non-Ω patterns in F. The solution is provided by the ML detector which incorporates both the $\bar{F}$-space component (DFFS) and the F-space likelihood (DIFS). The probabilistic interpretation of Figure 5(b) is as follows: SSD assumes a *single* prototype (the mean) in additive white Gaussian noise whereas the DFFS assumes a *uniform* density in F. The ML detector, on the other hand, uses the complete probability density for detection.

We have incorporated and tested the multiscale version of the ML detection technique in a face detection task. This multiscale head finder was tested on the ARPA FERET database where 97% of 2,000 faces were correctly detected.

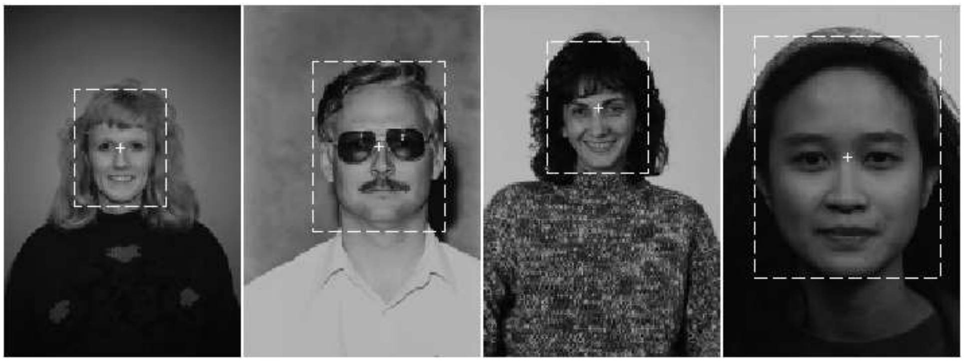


Figure 6: *Examples of multiscale face detection.*

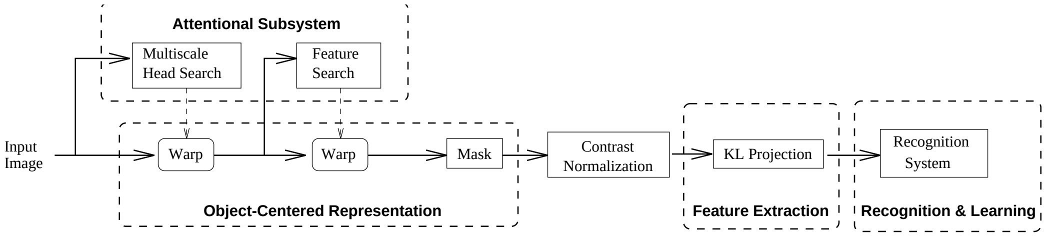


Figure 7: *The face processing system.*

Figure 6 shows examples of the ML estimate of the position and scale on these images. The multiscale saliency maps $S(i,j,k;\Omega)$ were computed based on the likelihood estimate $\hat{P}(\mathbf{x}|\Omega)$ in a 10-dimensional principal subspace using a Gaussian model (section 2.2.). Note that this detector is able to localize the position and scale of the head despite variations in hair style and hair color, as well as presence of sunglasses. Illumination invariance was obtained by normalizing the input subimage $\mathbf{x}$ to a zero-mean unit-norm vector.

4.1.1. Using ML detection for coding

We have also used the multiscale version of the ML detector as the *attentional* component of an automatic system for recognition and model-based coding of faces. The block diagram of this system is shown in Figure 7 which consists of a two-stage object detection and alignment stage, a contrast normalization stage, and a feature extraction stage whose output is used for both recognition and coding. Figure 8 illustrates the operation of the detection and alignment stage on a natural test image containing a human face. The function of the face finder is to locate regions in the image which have a high likelihood of containing a face.

The first step in this process is illustrated in Figure 8(b) where the ML estimate of the position and scale of the face are indicated by the cross-hairs and bounding box. Once these regions have been identified, the estimated scale and position

(a) (b) (c) (d)

Figure 8: *(a) original image, (b) position and scale estimate, (c) normalized head image, (d) position of facial features.*

(a) (b) (c)

Figure 9: *(a) aligned face, (b) eigenspace reconstruction (85 bytes) (c) JPEG reconstruction (530 bytes).*

are used to normalize for translation and scale, yielding a standard "head-in-the-box" format image (Figure 8(c)). A second feature detection stage operates at this fixed scale to estimate the position of 4 facial features: the left and right eyes, the tip of the nose and the center of the mouth (Figure 8(d)). Once the facial features have been detected, the face image is warped to align the geometry and shape of the face with that of a canonical model. Then the facial region is extracted (by applying a fixed mask) and subsequently normalized for contrast. The geometrically aligned and normalized image (shown in Figure 9(a)) is then projected onto a custom set of eigenfaces to obtain a feature vector which is then used for recognition purposes as well as facial image coding.

Figure 9 shows the normalized facial image extracted from Figure 8(d), its reconstruction using a 100-dimensional eigenspace representation (requiring only 85 bytes to encode) and a comparable non-parametric reconstruction obtained using a standard transform-coding approach for image compression (requiring 530 bytes to encode). This example illustrates that the eigenface representation used for recognition is also an effective *model-based* representation for data compression. The first 8 eigenfaces used for this representation are shown in Figure 10.

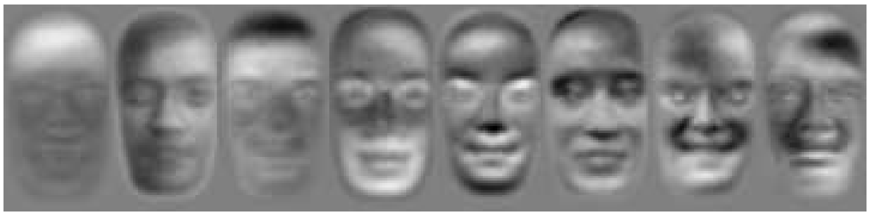


Figure 10: *The first 8 eigenfaces.*

4.1.2. Using ML detection for Recognition

Figure 11 shows the results of a similarity search in an image database tool called Photobook [27]. Each face in the database was automatically detected and aligned by the face processing system in Figure 7. The normalized faces were then projected onto a 100-dimensional eigenspace. The image in the upper left is the one searched on and the remainder are the ranked nearest neighbors in the FERET database. The top three matches in this case are images of the same person taken a month apart and at different scales. The recognition accuracy (defined as the percent correct rank-one matches) on a database of 155 individuals is 99% [21].

4.1.3. Recognition on Large Databases

In order to have an estimate of the recognition performance on much larger databases, we have conducted tests on a database of $7,562$ images of approximately $3,000$ people. The images were collected in a small booth at a Boston photography show, and include men, women, and children of all ages and races. Head position was controlled by asking people to take their own picture when they were lined up with the camera. Two LEDs placed at the bottom of holes adjacent to the camera allowed them to judge their alignment; when they could see both LEDs then they were correctly aligned.

The eigenfaces for this database were approximated using a principal components analysis on a representative sample of 128 faces. Recognition and matching was subsequently performed using the first 20 eigenvectors.

To assess the average recognition rate, 200 faces were selected at random, and a nearest-neighbor rule was used to find the most-similar face from the entire database. If the most-similar face was of the same person then a correct recognition was scored. In this experiment the eigenvector-based recognition system produced a recognition accuracy of 95%. This performance is somewhat surprising because the database contains wide variations in expression, and has relatively weak control of head position and illumination. In a *verification* task, our system yielded a false rejection rate of 1.5% at a false acceptance rate of 0.01%.

Figure 11: *Photobook: FERET face database.*

4.1.4. View-Based Recognition

The problem of face recognition under general viewing conditions (change in pose) can also be approached using an eigenspace formulation. There are essentially two ways of approaching this problem using an eigenspace framework. Given N individuals under M different views, one can do recognition and pose estimation in a universal eigenspace computed from the combination of NM images. In this way a single "parametric eigenspace" will encode both identity as well as pose. Such an approach, for example, has recently been used by Murase and Nayar [22] for general 3D object recognition.

Alternatively, given N individuals under M different views, we can build a "view-based" set of M distinct eigenspaces, each capturing the variation of the N individuals in a common view. The view-based eigenspace is essentially an extension of the eigenface technique to multiple sets of eigenvectors, one for each combination of scale and orientation. One can view this architecture as a set of parallel "observers" each trying to explain the image data with their set of eigenvectors (see also Darrell and Pentland [10]). In this view-based, multiple-observer approach, the first step is to determine the location and orientation of the target object by selecting the eigenspace which best describes the input image. This can be accomplished by calculating the likelihood estimate using each viewspace's eigenvectors and then selecting the maximum.

Figure 12: *Some of the images used to test accuracy at face recognition despite wide variations in head orientation. Average recognition accuracy was 92%, the orientation error had a standard deviation of* $15°$.

The key difference between the view-based and parametric representations can be understood by considering the geometry of facespace. In the high-dimensional vector space of an input image, multiple-orientation training images are represented by a set of M distinct regions, each defined by the scatter of N individuals. Multiple views of a face form non-convex (yet connected) regions in image space [2]. Therefore the resulting ensemble is a highly complex and non-separable manifold.

The parametric eigenspace attempts to describe this ensemble with a projection onto a single low-dimensional linear subspace (corresponding to the first n eigenvectors of the NM training images). In contrast, the view-based approach corresponds to M independent subspaces, each describing a particular region of the facespace (corresponding to a particular view of a face). The relevant analogy here is that of modeling a complex distribution by a single cluster model or by the union of several component clusters. Naturally, the latter (view-based) representation can yield a more accurate representation of the underlying geometry.

This difference in representation becomes evident when considering the quality of reconstructed images using the two different methods. Figure 13 compares reconstructions obtained with the two methods when trained on images of faces at multiple orientations. In Figure 13(a) we see first an image in the training set, followed by reconstructions of this image using first the parametric eigenspace and then the view-based eigenspace. Note that in the parametric reconstruction neither the pose nor the identity of the individual is adequately captured. The view-based reconstruction, on the other hand, provides a much better characterization of the object. Similarly, in Figure 13(b) we see a novel view ($+68°$) with respect to the training set ($-90°$ to $+45°$). Here, both reconstructions correspond to the nearest view in the training set ($+45°$) but the view-based reconstruction is seen to be more representative of the individual's identity. Although the qual-

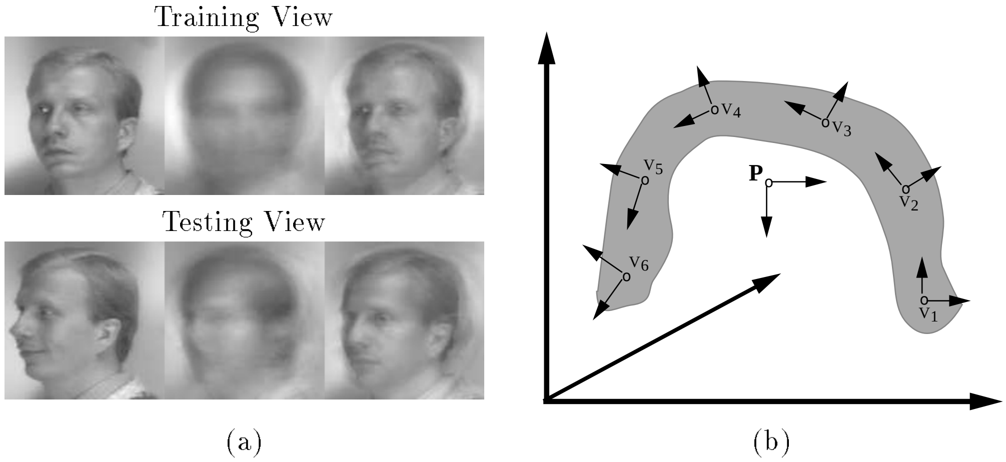


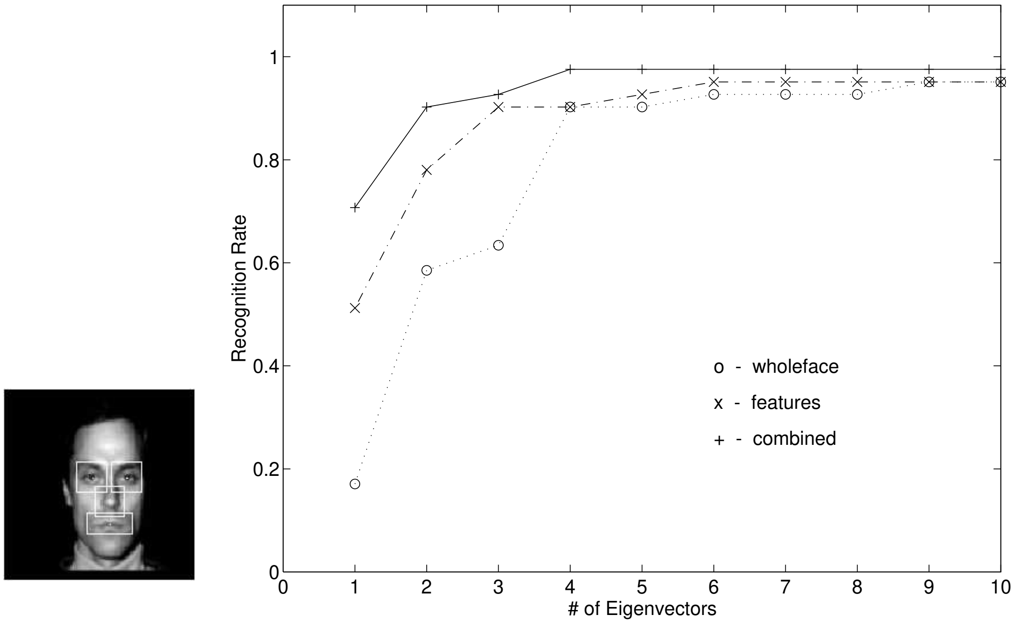
Figure 13: *(a) parametric vs. view-based eigenspace reconstructions for a training view and a novel testing view. The input image is shown in the left column. The middle and right columns correspond to the parametric and view-based reconstructions, respectively. All reconstructions were computed using the first 10 eigenvectors. (b) a schematic representation of the two approaches.*

ity of the reconstruction is not a direct indicator of the recognition power, from an information-theoretic point-of-view the multiple eigenspace representation is a more accurate representation of the signal content.

We have evaluated the view-based approach with data similar to that shown in Figure 12. This data consists of 189 images consisting of nine views of 21 people. The nine views of each person were evenly spaced from $-90°$ to $+90°$ along the horizontal plane. In the first series of experiments the *interpolation* performance was tested by training on a subset of the available views $\{\pm 90°, \pm 45°, 0°\}$ and testing on the intermediate views $\{\pm 68°, \pm 23°\}$. A 90% average recognition rate was obtained. A second series of experiments tested the *extrapolation* performance by training on a range of views (*e.g.,* $-90°$ to $+45°$) and testing on novel views outside the training range (*e.g.,* $+68°$ and $+90°$). For testing views separated by $\pm 23°$ from the training range, the average recognition rates were 83%. For $\pm 45°$ testing views, the average recognition rates were 50% (see [26] for further details).

4.1.5. Modular Recognition

The eigenface recognition method is easily extended to facial features as shown in Figure 14(a). This leads to an improvement in recognition performance by incorporating an additional layer of description in terms of facial features. This can be viewed as either a modular or layered representation of a face, where a coarse (low-resolution) description of the whole head is augmented by additional (higher-resolution) details in terms of salient facial features.



(a) (b)

Figure 14: *(a) facial eigenfeature regions, (b) recognition rates for eigenfaces, eigenfeatures and the combined modular representation.*

The utility of this layered representation (eigenface plus eigenfeatures) was tested on a small subset of our large face database. We selected a representative sample of 45 individuals with two views per person, corresponding to different facial expressions (neutral vs. smiling). These set of images was partitioned into a training set (neutral) and a testing set (smiling). Since the difference between these particular facial expressions is primarily articulated in the mouth, this feature was discarded for recognition purposes.

Figure 14(b) shows the recognition rates as a function of the number of eigenvectors for eigenface-only, eigenfeature-only and the combined representation. What is surprising is that (for this small dataset at least) the eigenfeatures alone were sufficient in achieving an (asymptotic) recognition rate of 95% (equal to that of the eigenfaces). More surprising, perhaps, is the observation that in the lower dimensions of eigenspace, eigenfeatures outperformed the eigenface recognition. Finally, by using the combined representation, we gain a slight improvement in the asymptotic recognition rate (98%). A similar effect was reported by Brunelli and Poggio [4] where the cumulative normalized correlation scores of templates for the face, eyes, nose and mouth showed improved performance over the face-only templates.

A potential advantage of the eigenfeature layer is the ability to overcome the

(a)

(b)

(c)

Figure 15: *(a) Test views, (b) Eigenface matches, (c) Eigenfeature matches.*

shortcomings of the standard eigenface method. A pure eigenface recognition system can be fooled by gross variations in the input image (hats, beards, etc.). Figure 15(a) shows additional testing views of 3 individuals in the above dataset of 45. These test images are indicative of the type of variations which can lead to false matches: a hand near the face, a painted face, and a beard. Figure 15(b) shows the nearest matches found based on standard eigenface matching. Neither of the 3 matches correspond to the correct individual. On the other hand, Figure 15(c) shows the nearest matches based on the eyes and nose, and results in correct identification in each case. This simple example illustrates the potential advantage of a modular representation in disambiguating low-confidence eigenface matches.

4.1.6. Recognition using Edge-based Features

We have also extended the normalized eigenface representation into an edge-based domain for facial description. We simply run the normalized facial image through a Canny edge detector to yield an edge map as shown in Figure 16(a). Such an edge map is simply an alternative representation which imparts mostly *shape* (as opposed to texture) information and has the advantage of being less susceptible to illumination changes. The recognition rate of a pure edge-based normalized eigenface representation (on a FERET database of 155 individuals) was found to be 95% which is surprising considering that it utilizes what appears to be (to humans at least) a rather impoverished representation. The slight drop in recognition rate is most likely due to the increased dimensionality of this representation

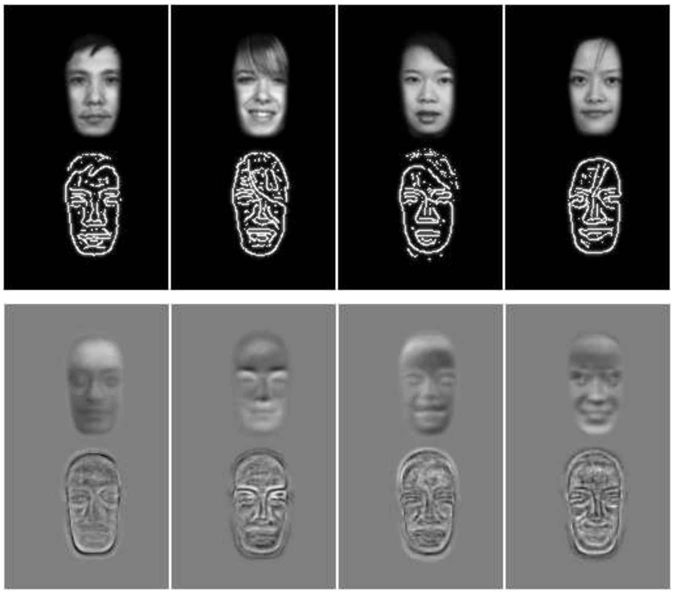


Figure 16: *(a) Examples of combined texture/edge-based face representations and (b) few of the resulting eigenvectors.*

space and its greater sensitivity to expression changes, *etc.*

Interestingly, we can combine both texture and edge-based representations of the object by simply performing a KL expansion on the augmented images shown in Figure 16. The resulting eigenvectors conveniently decorrelate the joint representation and provide a basis set which optimally spans both domains simultaneously. With this bimodal representation, the recognition rate was found to be 97%. Though still less than a normalized grayscale representation, we believe a bimodal representation can have distinct advantages for tasks other than recognition, such as detection and image interpolation.

4.2. Hands

We have also applied our eigenspace density estimation technique to articulated and non-rigid objects such as hands. In this particular domain, however, the original intensity image is an unsuitable representation since, unlike faces, hands are essentially textureless objects. Their identity is characterized by the variety of *shapes* they can assume. For this reason we have chosen an edge-based representation of hand shapes which is invariant with respect to illumination, contrast and scene background. A training set of hand gestures was obtained against a black background. The 2D contour of the hand was then extracted using a Canny edge-operator. These binary edge maps, however, are highly uncorrelated with each other due to their sparse nature. This leads to a very high-dimensional principal

(a)

(b)

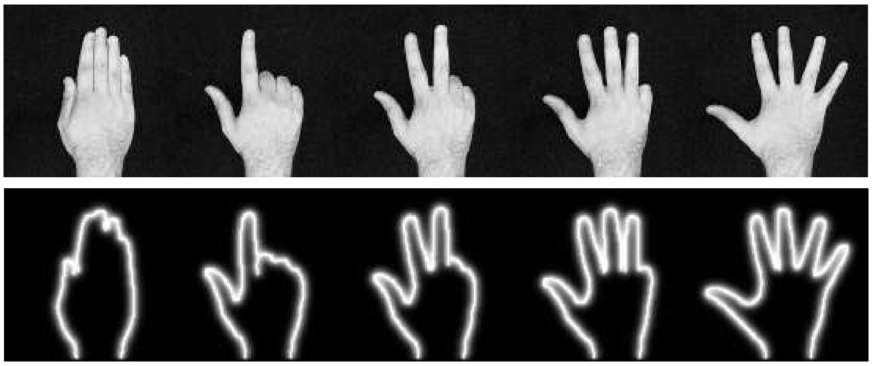
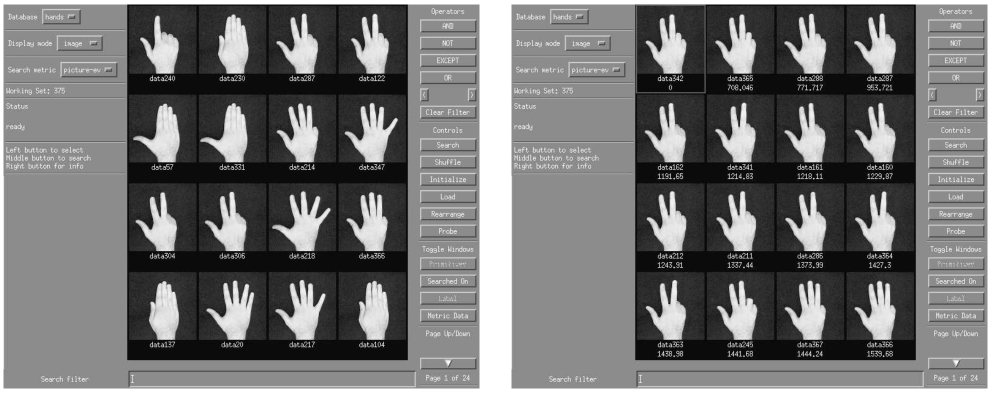


Figure 17: *(a) Examples of hand gestures and (b) their diffused edge-based representation.*

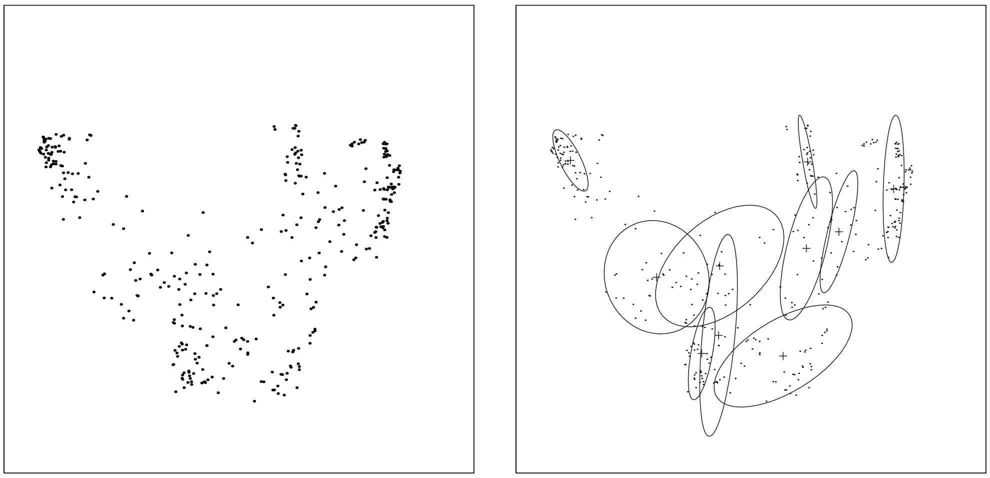


(a) (b)

Figure 18: *(a) a random collection of hand gestures (b) images ordered by similarity (left-to-right, top-to-bottom) to the image at the upper left.*

subspace. Therefore to reduce the intrinsic dimensionality, we *induced* spatial correlation via a *diffusion* process on the binary edge map, which effectively broadens and "smears" the edges, yielding a continuous-valued contour image which represents the object shape in terms of the spatial distribution of edges. Figure 17 shows examples of training images and their diffused edge map representations. We note that this *spatiotopic* representation of shape is biologically motivated and therefore differs from methods based purely on computational considerations (*e.g.,* moments [13], Fourier descriptors [20], "snakes" [16], Point Distribution Models [7], and modal descriptions [32]).

It is important to verify whether such a representation is adequate for discriminating between different hand shapes. Therefore we tested the diffused contour image representation in a recognition experiment which yielded a 100% rank-one accuracy on 375 frames from an image sequence containing 7 hand gestures. The



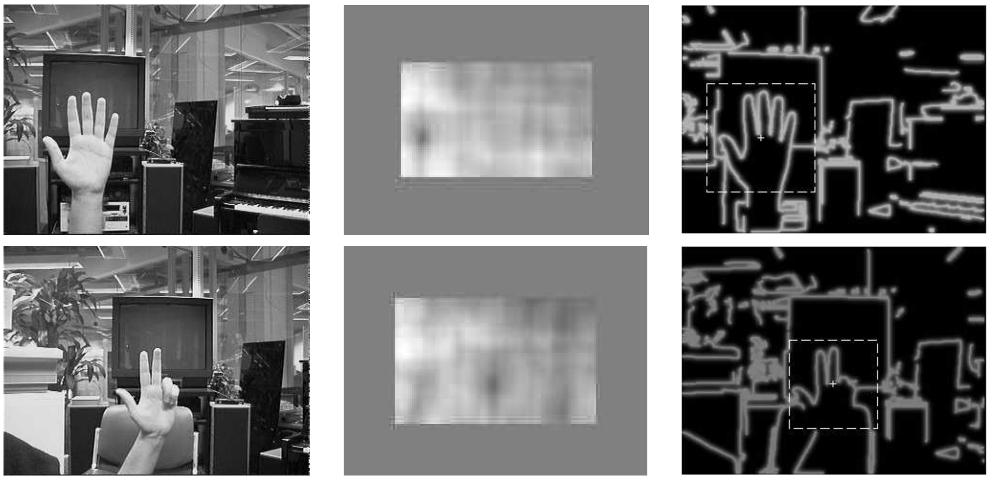
(a) (b)

Figure 19: *(a) Distribution of training hand shapes (shown in the first two dimensions of the principal subspace) (b) Mixture-of-Gaussians fit using 10 components.*

matching technique was a nearest-neighbor classification rule in a 16-dimensional principal subspace. Figure 18(a) shows some examples of the various hand gestures used in this experiment. Figure 18(b) shows the 15 images that are most similar to the "two" gesture appearing in the top left. Note that the hand gestures judged most similar are all objectively the same shape.

Naturally, the success of such a recognition system is critically dependent on the ability to find the hand (in any of its articulated states) in a cluttered scene, to account for its scale and to align it with respect to an object-centered reference frame prior to recognition. This localization was achieved with the same multiscale ML detection paradigm used with faces, with the exception that the underlying image representation of the hands was the diffused edge map rather the grayscale image.

The probability distribution of hand shapes in this representation was automatically learned using our eigenspace density estimation technique. In this case, however, the distribution of training data is *multimodal* due to the different hand shapes. Therefore the multimodal density estimation technique in section 2.3. was used. Figure 19(a) shows a projection of the training data on the first two dimensions of the principal subspace F (defined in this case by $M = 16$) which exhibit the underlying multimodality of the data. Figure 19(b) shows a 10-component Mixture-of-Gaussians density estimate for the training data. The parameters of this estimate were obtained with 20 iterations of the EM algorithm. The orthogonal $\bar{F}$-space component of the density was modeled with a Gaussian distribution as in section 2.3..



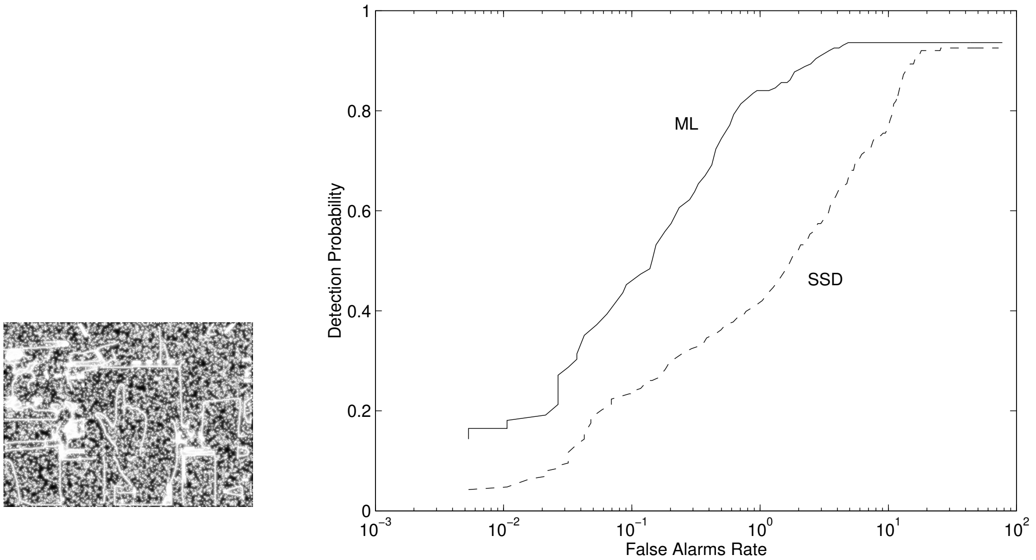
(a) (b) (c)

Figure 20: *(a) Original grayscale image, (b) negative log-likelihood map (at most likely scale) and (c) ML estimate of position and scale superimposed on edge map.*

The resulting complete density estimate $\hat{P}(\mathbf{x}|\Omega)$ was then used in a detection experiment on test imagery of hand gestures against a cluttered background scene. In accordance with our representation, the input imagery was first pre-processed to generate a diffused edge map and then scaled accordingly for a multiscale saliency computation. Figure 20 shows two examples from the test sequence, where we have shown the original image, the negative log-likelihood saliency map, and the ML estimates of position and scale (superimposed on the diffused edge map). Note that these examples represent two different hand gestures at slightly different scales.

To better quantify the performance of the ML detector on hands we carried out the following experiment. The original 375-frame video sequence of training hand gestures was divided into 2 parts. The first (training) half of this sequence was used for learning, including computation of the KL basis and the subsequent EM clustering. For this experiment we used a 5-component mixture in a 10-dimensional principal subspace. The second (testing) half of the sequence was then embedded in the background scene, which contains a variety of shapes. In addition, severe noise conditions were simulated as shown in Figure 21(a).

We then compared the detection performance of an SSD detector (based on the mean edge-based hand representation) and a probabilistic detector based on the complete estimated density. The resulting negative-log-likelihood detection maps were passed through a valley-detector to isolate local minimum candidates which were then subjected to a ROC analysis. A correct detection was defined as a below-threshold local minimum within a 5-pixel radius of the ground truth target location. Figure 21(b) shows the performance curves obtained for the two



(a) (b)

Figure 21: *(a) Example of test frame containing a hand gesture amidst severe background clutter and (b) ROC curve performance contrasting SSD and ML detectors.*

detectors. We note, for example, that at an 85% detection probability the ML detector yields (on the average) 1 false alarm per frame, where as the SSD detector yields an order of magnitude more false alarms.

5. Discussion

In this paper we have described an eigenspace density estimation technique for unsupervised visual learning which exploits the *intrinsic* low-dimensionality of the training imagery to form a computationally simple estimator for the complete likelihood function of the object. Our estimator is based on a subspace decomposition and can be evaluated using only the M-dimensional principal component vector. We derived the form for an optimal estimator and its associated expected cost for the case of a Gaussian density. In contrast to previous work on learning and characterization — which uses PCA primarily for dimensionality reduction and/or feature extraction — our method uses the eigenspace decomposition as an integral part of estimating *complete* density functions in high-dimensional image spaces. These density estimates were then used in a maximum likelihood formulation for target detection. The multiscale version of this detection strategy was demonstrated in applications in which it functioned as an attentional subsystem for object recognition. The performance was found to be superior to existing detection techniques in experiments with large numbers of test data.

We note that from a probabilistic perspective, the class conditional density $P(\mathbf{x}|\Omega)$ is the most important object representation to be learned. This density is the critical component in detection, recognition, prediction, interpolation and general inference. For example, having learned these densities for several object classes $\{\Omega_1, \Omega_2, \cdots, \Omega_n\}$, one can invoke a Bayesian framework for classification and recognition:

$$P(\Omega_i|\mathbf{x}) \;=\; \frac{P(\mathbf{x}|\Omega_i)P(\Omega_i)}{\sum\limits_{j=1}^{n} P(\mathbf{x}|\Omega_j)P(\Omega_j)} \tag{26}$$

where now a maximum *a posteriori* (MAP) classification rule can be used for object/pose identification.

Such a framework is also important in detection. In fact, the ML detection framework can be extended using the notion of a "not-class" $\bar{\Omega}$, resulting in *a posteriori* saliency maps of the form

$$S(i,j,k;\Omega) \;=\; P(\Omega|\mathbf{x}) \;=\; \frac{P(\mathbf{x}|\Omega)P(\Omega)}{P(\mathbf{x}|\bar{\Omega})P(\bar{\Omega}) \;+\; P(\mathbf{x}|\Omega)P(\Omega)} \tag{27}$$

where now a maximum *a posteriori* (MAP) rule can be used to estimate the position and scale of the object. One difficulty with such a formulation is that the "not-class" $\bar{\Omega}$ is, in practice, too broad a category and is therefore multimodal and very high-dimensional. One possible approach to this problem is to use ML detection to identify the particular subclass of $\bar{\Omega}$ which has high likelihoods (*e.g.,* false alarms) and then to estimate this distribution and use it in the MAP framework. This can be viewed as a probabilistic approach to learning using positive as well as *negative* examples.

In fact, such a MAP framework can be viewed as a Bayesian formulation of some neural network approaches to target detection. Perhaps the most closely related is the neural network face detector of Sung & Poggio [33] which is essentially a trainable nonlinear binary pattern classifier. They too learn the distribution of the object class with a Mixture-of-Gaussians model (using an elliptical k-means algorithm instead of EM). Instead of likelihoods, however, input patterns are represented by a set of distances to each mixture component (similar to a combination of the DIFS and DFFS), thus forming a feature vector indicative of the overall class membership. In addition, Sung & Poggio explicitly model the "not-class" by learning the distribution of nearby *non-face* patterns. The set of distances to both classes are then used to train a neural network to discriminate between face and non-face patterns (similar to computing a likelihood ratio in MAP). Another recent example of a neural network technique for object detection which also utilizes negative examples is the face-finder system of Rowley *et al.* [31]. The experimental results obtained with these methods clearly demonstrate the need for incorporating negative examples in building robust detection systems.

References

[1] Anderson, C.H., Burt, P.J., and Van der Wall, G.S., "Change Detection and Tracking Using Pyramid Transform Techniques," *Proc. of SPIE Conf. on Intelligence, Robots and Computer Vision*, vol. 579, pp. 72-78, 1985.

[2] Bichsel, M., and Pentland, A., "Human Face Recognition and the Face Image Set's Topology," *CVGIP: Image Understanding*, vol. 59, no. 2, pp. 254-261, 1994.

[3] Bregler, C., and Omohundro, S.M., "Surface Learning with Applications to Lip Reading," in *Advances in Neural Information Processing Systems 6*, eds. J.D. Cowan, G. Tesauro and J. Alspector, Morgan Kaufman Publishers, San Fransisco, pp. 43-50, 1994.

[4] Brunelli, R., and Poggio, T., "Face Recognition: Features vs. Templates,", *IEEE Trans. on Pattern Analysis and Machine Intelligence,*, vol. 15, no. 10, Oct. 1993.

[5] Brunelli, R., and Messelodi, S., "Robust Estimation of Correlation: An Application to Computer Vision," *IRST Tech. Report no. 9310-015*, October 1993.

[6] Burl, M.C., *et al.*, "Automating the Hunt for Volcanos on Venus," *Proc. IEEE Conf. on Computer Vision & Pattern Recognition*, Seattle, WA, June 21-23, 1994.

[7] Cootes, T.F. and Taylor, C.J., "Active Shape Models: Smart Snakes," in *Proc. British Machine Vision Conf.*, Springer-Verlag, pp. 9-18, 1992.

[8] Cootes, T.F., Hill, A., Taylor, C.J. and Haslam, J., "Use of Active Shape Models for Locacting Structures in Medical Images," *Image and Vision Computing*, vol. 12, no. 6, pp. 355-365, July/August 1994.

[9] Cover, M. and Thomas, J.A., *Elements of Information Theory*, John Wiley & Sons, New York, 1994.

[10] Darrell, T., and Pentland, A., "Space-Time Gestures," *Proc. IEEE Conf. on Computer Vision & Pattern Recognition*, New York, NY, June 1993.

[11] Dempster, A.P., Laird, N.M., Rubin, D.B., "Maximum Likelihood from Incomplete Data via the EM Algorithm," *Journal of the Royal Statistical Society B*, vol. 39, 1977.

[12] Golub, G.H. and Van Loan, C.F., *Matrix Computations*, Johns Hopkins Press, 1989.

[13] Hu, M.K., "Visual Pattern Recognition by Moment Invariants," *IEEE Trans. on Information Theory*, vol. 8, pp. 179-187, 1962.

[14] Jolliffe, I.T., *Principal Component Analysis*, Springer-Verlag, New York, 1986.

[15] Kanade, T., "Picture Processing by Computer Complex and Recognition of Human Faces," Tech. Report, Kyoto University, Dept. of Information Science, 1973.

[16] Kass, M., Witkin, A., and Terzopoulos, D., "Snakes: Active Contour Models," *Int'l Journal of Computer Vision*, vol. 1, no. 4, pp. 321-331, 1987.

[17] Kirby, M., and Sirovich, L., "Application of the Karhunen-Loeve Procedure for the Characterization of Human Faces," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 12, no. 1, Jan. 1990.

[18] Kumar, B., Casasent, D., and Murakami, H., "Principal Component Imagery for Statistical Pattern Recognition Correlators," *Optical Engineering*, vol. 21, no. 1, Jan/Feb 1982.

[19] Loeve, M.M., *Probability Theory*, Van Nostrand, Princeton, 1955.

[20] McElroy, T., Wilson, E., and Anspach, G. "Fourier Descriptors and Neural Networks for Shape Classification," in *Proc. of Int'l Conf. on Acoustics, Speech and Signal Processing*, Detroit, MI, May 1995.

[21] Moghaddam, B. and Pentland, A., "Face Recognition Using View-Based and Modular Eigenspaces," in *Automatic Systems for the Identification and Inspection of Humans*, SPIE vol. 2277. 1994.

[22] Murase, H., and Nayar, S.K., "Visual Learning and Recognition of 3D Objects from Appearance," *Int'l Journal of Computer Vision*, vol. 14, no. 1, 1995.

[23] Nayar, S.K., Murase, H., and Nene, S.A., "General Learning Algorithm for Robot Vision," in *Neural & Stochastic Methods in Image & Signal Processing*, SPIE vol. 2304, July 1994.

[24] Palmer, S.E., "The Psychology of Perceptual Organization: A Transformational Approach," in *Human and Machine Vision*, J. Beck, B. Hope and A. Rosenfeld (eds.), Academic Press, 1983.

[25] Pentland, A. and Sclaroff, S., "Closed-Form Solutions for Physically Based Shape Modeling and Recovery," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 13, no. 7, pp. 715–729, July 1991.

[26] Pentland, A., Moghaddam, B. and Starner, T., "View-based and Modular Eigenspaces for Face Recognition," *Proc. of IEEE Conf. on Computer Vision & Pattern Recognition*, Seattle, WA, June 1994.

[27] Pentland, A., Picard, R., and Sclaroff, S., "Photobook: Tools for Content-Based Manipulation of Image Databases," in *Storage and Retrieval of Image and Video Databases II*, SPIE vol. 2185, San Jose, Feb 6-10, 1994.

[28] Poggio, T. and Girosi, F., "Networks for Approximation and Learning," *Proceedings of the IEEE*, vol. 78, no. 9, pp. 1481-1497, 1990.

[29] Redner, R.A., and Walker, H.F., "Mixture Densities, Maximum Likelihood and the EM Algorithm," *SIAM Review*, vol. 26, no. 2, pp. 195-239, 1984.

[30] Reisfeld, D., Wolfson, H., and Yeshurun, Y., "Detection of Interest Points Using Symmetry,", *Proc. of Int'l Conf. on Computer Vision*, Osaka, Japan, Dec. 1990.

[31] Rowley, H., Baluja, S. and Kanade, T., "Human Face Detection in Visual Scenes," Technical Report CMU-CS-95-158, Carnegie Mellon University, July 1995.

[32] Sclaroff, S. and Pentland, A., "Modal Matching for Correspondence and Recognition," *IEEE Trans. on Pattern Analysis & Machine Intelligence,* vol. 17, no. 6, pp. 545-561, 1995.

[33] Sung, K., and Poggio, T., "Example-based Learning for View-based Human Face Detection," in *Proc. of Image Understanding Workshop*, Monterey, CA, November 1994.

[34] Turk, M., and Pentland, A., "Eigenfaces for Recognition," *Journal of Cognitive Neuroscience,* vol. 3, no. 1, 1991.

[35] Vincent, J. M., Waite, J. B., and Myers, D. J, "Automatic Location of Visual Features by a System of Multilayered Perceptrons," *IEE Proceedings,* vol. 139, no. 6, Dec. 1992.

[36] Weng, J.J., "On Comprehensive Visual Learning," in *Proc. NSF/ARPA Workshop on Performance vs. Methodology in Computer Vision,* Seattle, WA, June 24-25, 1994.

Acknowledgements

The FERET database was provided by the US Army Research Laboratory. The multiple-view face database was provided by Westinghouse Electric Systems.