MVA'90 IAPR Workshop on Machine Vision Applications Nov. 28-30,1990, Tokyo

FACE RECOGNITION WITHOUT FEATURES

Matthew Turk and Alex Pentland
The Media Laboratory
Massachusetts Institute of Technology
20 Ames St., Cambridge, MA 02139 USA

ABSTRACT

We have developed a near-real-time computer system which can locate and track a subject's head, and then recognize the person by comparing characteristics of the face to those of known individuals. Our approach treats the face recognition problem as an intrinsically two-dimensional recognition problem, taking advantage of the fact that faces are are normally upright and thus may be described by a small set of 2-D characteristic views. The system functions by projecting face images onto a feature space that spans the significant variations among known face images. The significant features are known as "eigenfaces", because they are the eigenvectors (principal components) of the set of faces; they do not necessarily correspond to features such as eyes, ears, and noses.

INTRODUCTION

Computers that recognize and identify people may be useful in a number of applications: criminal identification, security systems, human-computer interaction, and photographic processing, to name a few. The face is a natural, reasonably reliable, and nonintrusive means of identification, and in a machine vision system faces can be used both to detect the presence of people (*recognition*) and to identify the individuals (*identification*).

The appropriateness of the face model or representation is critical to meeting such general and robust criteria for the tasks of face recognition and identification. Typical modeling of object shape for computer vision and graphics includes surface, sweep, and volumetric representations. Polygonal approximations to analytic surfaces, B-spline surfaces, and elastically deformable models have been used for generating and animating faces in computer graphics. Such representations are not particularly appropriate for the inverse problem of recovering information from images and matching to stored models.

Most approaches to automated face recognition and identification, since the late 1960's, have focused on locating and modeling individual features (such as eyes, the nose, the mouth) and their relationships (e.g. [2, 3, 4]). These systems have met with limited success, largely because the face models are not robust to small changes such as different expressions, and performance degrades quickly as the input face image differs from the expected configuration. As a number of researchers have shown, individual features and their immediate relationships comprise an insufficient representation to account for the performance of adult human face identification [5]. Feature locations are not a sufficient basis for face recognition.

Taking the position of the opposite extreme, we set out to identify and recognize faces without explicitly locating and modeling features. Instead we characterize the variation among a set of faces and calculate a small number of characteristic face images which serve as a basis set. These "eigenfaces" span the "face space", in that linear combinations of these images can approximate an appropriately large number of face images. An image (or subimage) is recognized as a face when it lies within or near the low-dimensional "face space". The location in face space determines the identity among the known faces.

The calculation of the "eigenfaces" is performed offline. The projection into face space is a computationally simple process involving image multiplies and summations, and the identification is a simple low-dimensional pattern recognition process. The system therefore performs face identification quickly with available hardware. In the following sections, we will briefly discuss the computation of the eigenfaces, recognition based on the face space, learning new faces, and application to other tasks.

DEFINING THE FACE SPACE

In the language of information theory, we want to extract the relevant information in a face image, encode it as efficiently as possible, and compare one face encoding with a database of models encoded similarly. A simple approach to extracting the information contained in an image of a face is to somehow capture the variation in a collection of face images, independent of any judgement of features, and use this information to encode and compare individual face images.

In mathematical terms, we wish to find the principal components of the distribution of faces, or the eigenvec-

Figure 1: Face images used as the training set.

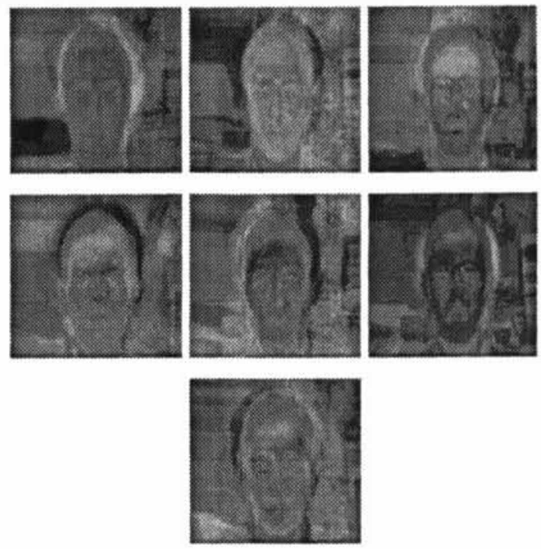

Figure 2: Seven of the eigenfaces calculated from the training set of Figure 1.

tors of the covariance matrix of the set of face images, treating an image as a point (or vector) in a very high dimensional space. The eigenvectors are ordered, each one accounting for a different amount of the variation among the face images.

These eigenvectors can be thought of as a set of features which together characterize the variation between face images. Each image location contributes more or less to each eigenvector, so that we can display the eigenvector as a sort of ghostly face which we call an *eigenface*. Some of these faces are shown in Figure 2. Each eigenface deviates from uniform grey where some facial feature differs among the set of training faces; they are a sort of map of the variations between faces.

Each individual face can be represented exactly in terms of a linear combination of the eigenfaces. Each face can also be approximated using only the "best" eigenfaces — those that have the largest eigenvalues, and which therefore account for the most variance within the set of face images. The best M eigenfaces span an M-dimensional subspace — "face space" — of all possible images.

Within this framework, a face image should be "close" to the face space — i.e. the distance between the image and its projection onto face space should be small — and a non-face image should be "far" from the face space. This distance from face space ε is used as a measure of "faceness".

The idea of using eigenfaces was motivated by a technique developed by Sirovich and Kirby [6, 7] for efficiently representing pictures of faces using principal component analysis. They argued that, at least in principle, any collection of face images can be approximately reconstructed by storing a small collection of weights for each face and a small set of standard pictures (the eigenpictures).

While the use of eigenspace for image coding seems inefficient in general, it occurred to us that perhaps a useful way to learn and recognize faces would be to build up the characteristic features by experience over time and recognize particular faces by comparing the feature weights needed to (approximately) reconstruct them with the weights associated with known individuals. Each individual, therefore, would be characterized by the small set of feature or eigenpicture weights needed to describe and approximately reconstruct them — an extremely compact representation when compared with the images themselves.

CONSTRUCTING FACE SPACE

Let a face image $I(x,y)$ be a two-dimensional N by N array of (8-bit) intensity values. An image may also be considered as a vector of dimension N^2, so that a typical image of size 256 by 256 becomes a vector of dimension $65,536$, or, equivalently, a point in $65,536$-dimensional space. An ensemble of images, then, maps to a collection of points in this huge space.

Images of faces, being similar in overall configuration, will not be randomly distributed in this huge image space and thus can be described by a relatively low dimensional

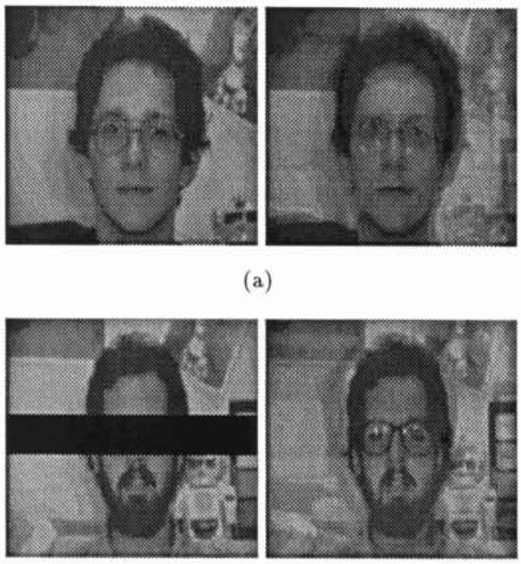

Figure 3: (a) An original face image and its projection onto the face space defined by the eigenfaces of Figure 2. (b) The projection of a degraded image into face space.

subspace. The main idea of the principal component analysis (or Karhunen-Loeve expansion) is to find the vectors which best account for the distribution of face images within the entire image space. These vectors define the subspace of face images, which we call "face space". Each vector is of length N^2, describes an N by N image, and is a linear combination of the original face images. Because these vectors are the eigenvectors of the covariance matrix corresponding to the original face images, and because they are face-like in appearance, we refer to them as "eigenfaces." Some examples of eigenfaces are shown in Figure 2.

The eigenface images span a basis set with which to describe face images. The eigenfaces span an M'-dimensional subspace of the original N^2 image space. In many of our test cases, based on $M = 16$ face images, $M' = 7$ eigenfaces were used. A new face image (Γ) is transformed into its eigenface components (projected into "face space") by a simple operation, $\omega_k = \mathbf{u}_k^T(\Gamma - \Psi)$, for $k = 1, \ldots, M'$. This describes a set of point-by-point image multiplications and summations, operations performed at approximately frame rate on current image processing hardware. Figure 3 shows an image and its projection into the seven-dimensional face space.

The weights form a vector $\Omega^T = [\omega_1\ \omega_2\ \ldots\ \omega_{M'}]$ that describes the contribution of each eigenface in representing the input face image, treating the eigenfaces as a basis set for face images. The vector is then used in a pattern recognition algorithm to find which of a number of predefined face classes, if any, best describes the face.

There are four possibilities for an input image and its pattern vector: (1) near face space and near a face class; (2) near face space but not near a known face class; (3) distant from face space and near a face class; and (4) distant from face space and not near a known face class.

In the first case, an individual is recognized and identified. In the second case, an unknown individual is present. The last two cases indicate that the image is not a face image. Case three typically shows up as a false positive in most recognition systems; in our framework, however, the false recognition may be detected because of the significant distance between the image and the subspace of expected face images.

In summary, this approach to face recognition involves the following initialization operations:

1. Acquire an initial set of face images (the training set).

2. Calculate the eigenfaces from the training set, keeping only the M images which correspond to the highest eigenvalues. These M images define the *face space*. As new faces are experienced, the eigenfaces can be updated or recalculated.

3. Calculate the corresponding distribution in M-dimensional weight space for each known individual, by projecting their face images onto the "face space".

These operations can also be performed from time to time whenever there is free excess computational capacity.

Having initialized the system, the following steps are then used to recognize new face images:

1. Calculate a set of weights based on the input image and the M eigenfaces by projecting the input image onto each of the eigenfaces.

2. Determine if the image is a face at all (whether known or unknown) by checking to see if the image is sufficiently close to "face space."

3. If it is a face, classify the weight pattern as either a known person or as unknown.

4. (Optional) Update the eigenfaces and/or weight patterns.

5. (Optional) If the same unknown face is seen several times, calculate its characteristic weight pattern and incorporate into the known faces.

In our current system calculation of the eigenfaces is done offline as part of the training. The recognition currently takes about 400 msec running rather inefficiently in Lisp on a Sun4, using face images of size 128x128. With some special-purpose hardware, the current version could run at close to frame rate (33 msec).

LEARNING NEW FACES

The concept of "face space" allows for the ability to learn and subsequently recognize new faces in an unsupervised manner. When an image is sufficiently close to face space but is not classified as one of the familiar faces, it is initially labeled as "unknown". The computer stores the pattern vector and the corresponding unknown image. If a collection of "unknown" pattern vectors cluster in the pattern space, the presence of a new but unidentified face is postulated.

The images corresponding to the pattern vectors in the cluster are then checked for similarity by requiring that the distance from each image to the mean of the images is less than a predefined threshold. If the images pass the similarity test, the average of the feature vectors is added to the database of known faces. Occasionally, the eigenfaces may be recalculated using these stored images as part of the new training set.

DISCUSSION

The eigenface approach to face recognition was motivated by information theory, leading to the idea of basing face recognition on a small set of image features that best approximate the set of known face images, without requiring that they correspond to our intuitive notions of facial parts and features. This approach to recognition is applicable to any domain in which the objects to be recognized are seen in typical views and share a common configuration, e.g. written characters or automobiles.

The eigenfaces may be viewed as a set of templates used in a "generalized correlation" scheme. Rather than matching to a single correlation template, e.g. of an individual's face, we can account for a number of faces with the set of eigenface templates or images.

We have used the techniques described above to build a system which locates and recognizes faces in near-real-time in a reasonably unstructured environment. A fixed camera, monitoring part of a room, is connected to a Datacube image processing system, which resides on the bus of a Sun 3/160. The Datacube digitizes the video image and performs spatiotemporal filtering, thresholding, and subsampling at frame rate (30 frames/sec).

Recognition occurs in this system at rates of up to two or three times per second. Until motion is detected, or as long as the image is not perceived to be a face, there is no output. When a face is recognized, the image of the identified individual is displayed on the Sun monitor.

It is important to note that many applications of face recognition do not require perfect identification, although most require a low false positive rate. In searching a large database of faces, for example, it may be preferable to find a small set of likely matches to present to the user. For applications such as security systems or human-computer interaction, the system will normally be able to "view" the subject for a few seconds or minutes, and thus will have a number of chances to recognize the person. Our experiments show that the eigenface technique can be made to perform at very high accuracy, although with a substantial "unknown" rejection rate, and thus is potentially well suited to these applications.

References

[1] A. Pentland, "Perceptual organization and the representation of natural form," *Artificial Intelligence* **28** (1986) 293-331.

[2] T. Kanade, "Picture processing system by computer complex and recognition of human faces," Dept. of Information Science, Kyoto University, Nov. 1973.

[3] Craw, Ellis, and Lishman, "Automatic extraction of face features," *Pattern Recognition Letters* 5 (1987) 183-187.

[4] K. Wong, H. Law, and P. Tsang, "A system for recognising human faces," *Proc. ICASSP*, May 1989, pp. 1638-1642.

[5] S. Carey and R. Diamond, "From piecemeal to configurational representation of faces," *Science*, Vol. 195, Jan. 21, 1977, 312-13.

[6] L. Sirovich and M. Kirby, "Low-dimensional procedure for the characterization of human faces," *J. Opt. Soc. Am. A*, Vol. 4, No. 3, March 1987, 519-524.

[7] M. Kirby and L. Sirovich, "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.