

A Survey of Face Recognition Algorithms and Testing Results

William A. Barrett
National Biometrics Test Center
San Jose State University
One Washington Square
San Jose, CA 95192-0180

Abstract

Automated face recognition (AFR) has received increased attention in recent years. We describe two general approaches to the problem and discuss their effectiveness and robustness with respect to several possible applications. We also discuss some issues of run-time performance.

1. Introduction

A formal method of classifying faces was first proposed by Francis Galton in 1888 [GALT88]. He proposed collecting facial profiles as curves, finding their norm, and then classifying other profiles by their deviations from the norm. The classification was to be *multi-modal*, i.e. resulting in a vector of (hopefully) independent measures that could be compared with other vectors in a database.

Automated face recognition (AFR) has been of interest to a growing number of research groups since 1990. Driving the recent development have been improvements in the technology of neural networks, wavelet analysis, computer graphics and machine vision.

As in most other biometric measurement systems, a general goal of AFR is to achieve a high level of performance in *matching a given face against a database of faces*. The performance of an AFR system will be judged by some combination of precision of matching (low level of false negatives and false positives), robustness against adverse factors, high speed, and low cost of the equipment. Adverse factors in AFR include lighting conditions, noise in the image, facial expression variations, glasses, hirsute changes, and posture.

The matching performance in current AFR systems is relatively poor compared to that achieved in fingerprint and iris matching, yet it may be the only available measuring tool for an application. Error rates of 2-25% are typical. It is also effective if combined with other biometric measurements.

A survey of commercial AFR systems is given in [BIOM97].

2. AFR Technology Categories

The AFR technology falls into three main subgroups, which represent more-or-less independent approaches to the problem: *neural network solutions*, *eigenface solutions*, and *wavelet/elastic matching solutions*. Each of these first requires that a facial image be identified in a scene, a process called *segmentation*. The image should be normalized to some extent. Normalization is usually a combination of linear translation, rotation and scaling, although the elastic matching method includes spatial transformations. If the eyes and the mouth can be located, these reference points can be used to drive normalization to yield a standardized facial image. This of course, supposes that a nearly frontal view is provided. Few AFR systems work effectively with profile views, if the database consists of frontal views.

3. Segmentation

Segmentation (locating a face in a busy scene) is often considered a pre-processing step. However, it isn't necessarily simple. The images of many common objects resemble faces, and they may have to be rejected later.

An common segmentation approach uses video motion sequences. A video camera in a fixed location simply watches for moving targets against a stationary background. Then finding the head and something resembling a face is relatively simple. However, this can also be fooled by viewing a television set or some other moving object.

Elastic matching [LADES93] provides some built-in segmentation, and some work by Pentland [PENT94] suggests that eigenfaces can be effective in segmentation.

4. Applications

A short list of applications is given below. This does not include face recognition problems commonly

performed by humans, for example, the use of Identikits, witness testimony, etc.

Table 1		
<i>Application</i>	<i>Prospects of AFR</i>	<i>AFR problems</i>
Credit card, driver's license, passport, personal ID: verification	Very good For accurate verification, should be augmented with other measures	Expanding card code for image Image coding standards Potentially large database
Mug shot matching - yield a smaller list of suspects: identification	Good. Controlled segmentation	Digital conversion of mug shot library Candidate photo required
Bank/store security - identifying a suspect	Good Motion video segmentation	Image may be poor quality - few pixels, varying lighting, expressions
Crowd surveillance - searching for wanted persons	Fair to good	Poor image quality Segmentation difficult Real time performance
Smart room - identifying and tracking people in a meeting room	Good Motion video segmentation	Uncontrolled position and expression

5. Static matching

In *static matching*, we have a single facial photograph, and are required to find any or all matching faces in a database. The database will typically contain *mugshots* taken under controlled lighting conditions with deadpan expressions. A typical database will already be segmented, whether by manual or automatic methods, with the eye and mouth locations identified.

A candidate photo is often taken with uncontrolled lighting conditions, pose and expression. The subject may attempt a disguise.

An AFR should supply a measure of "closeness" between the candidate and each of the database members. Most AFR systems produce a many-dimensional vector that characterizes a face. Two such vectors can then be compared by reducing their difference to a single linear measure. For example, the Cartesian distance between two such vectors yields an easily computed linear difference measure $d'(i,j)$ between a candidate i and a database member j .

Ideally, d' will be zero for a match and large otherwise. In fact, for a large set of candidates, there will be a double distribution of d' , one for the expected matches and another for the expected non-matches. By setting a threshold criterion for d' sufficiently small, we can minimize the rate of accepting impostors, but at the expense of also rejecting authentic. By setting the threshold larger, we can minimize the rate of rejecting authentic, but at the expense of accepting impostors.

The relationship between false matches and false acceptances is commonly expressed as a *Receiver Operating Characteristic*, or *ROC* curve.

6. The FERET tests

Another way to describe the quality of an AFR system is by *rank-ordering*. For each candidate face, the system is asked to rank-order the faces in the database by the quality of the match. If the system develops a linear measure d' in this process, d' is merely used to produce a simple rank position. A large number of candidates, some in the database and others not, are classified this way.

This works reasonably well in comparing AFR systems provided that the candidate set is large and diverse. However, the performance of a particular AFR on a particular task is not predictable from rank-ordering.

Comparative tests were performed by Phillips, [PHIL96, PHIL97] on systems provided by research teams at MIT (eigenfaces), USC (elastic matching), Rutgers, the Rockefeller Institute, and others. The reported results are mostly rank-ordered, but ROC tests are provided in recent reports.

Variations included pose (full frontal vs. quarter profile, half profile and full profile), glasses/no glasses, image brightness, image scale, and different capture times.

The database includes some lighting variations and changes in facial expression. (Candidates were asked to "make a face" for certain shots). Many of the candidates were photographed over a period of two years, in order to studying aging effects.

The results show that image size was easily overcome by all the methods. Illumination level was a problem for the USC system, but not MIT. Rotation negatively impacted all the methods, but to a somewhat different degree. The USC system was more robust to head rotation than the MIT system. Two years of aging significantly reduced recognition.

The most recent results show the highest scores for Joseph Attick's FACEIT system [FACE97]. Faceit is a commercial product. Details on its algorithm are lacking.

7. Preprocessing

Segmentation is most easily achieved in a typical surveillance situation through motion video. We merely look for changes from one frame to the next, which usually indicates the motion of a person. Pulling a facial image can be done in a number of ways.

Given a rough outline of a face, the eyes can usually be found by examining the horizontal intensity signature, and correlating it to that from a typical face. The eyes and the mouth will usually be darker than the other areas. Some rough pattern matching of dark circles to find the eye position can then be followed by a triangulation, using a typical mouth position, possibly refining this against the horizontal signature.

Scaling and rotation of the face can next be done by classical methods of pixel averaging. The eye-mouth triangle form the basis of a transformation by which all the pixels can be mapped into a standard orientation.

8. Neural Networks

A back-propagation neural network can be trained to recognize face images. This is in principle an associative memory problem, for which neural networks offer efficient solutions. However, a simple network can be very complex and difficult to train.

A typical image recognition network requires $N = m \times n$ input neurons, one for each of the pixels in an $m \times n$ image. For example, a low resolution image of 128 pixels square requires $N = 16,385$ input neurons. These are typically mapped to a number of hidden-layer neurons, p in number. These in turn map to n output neurons, at least one of which is expected to fire on matching a particular face in the database. It happens that p can be much less than N . The hidden layer is considered to be a *feature vector*. Roughly speaking, it expresses the facial features in a condensed way.

Such a network is difficult to train. To reduce the complexity, Cottrell and Fleming [COTT90] introduced a second back-propagation net as a *classification net*. The autoassociation net is used to train the network, and the classification net yields the matching information.

Although neural networks are used for many image recognition problems, Cottrell and Fleming show in their paper that, "under the best circumstances", a neural network of this design is no better than an eigenface feature network.

9. Eigenfaces

Eigenface recognition was first proposed by Sirovich and Kirby [SIRO87] as an application of principal-component analysis (PCA) of an n -dimensional matrix. They also present some simple experiments that illustrate the power of their method.

Start with a preprocessed image $I(x, y)$, which is a two-dimensional N by N array of intensity values (usually 8 bit gray scale). This may be considered a vector of dimension N^2 , so that an 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. The central idea is to find a small set of faces (the *eigenfaces*) that can approximately represent any point in the face space as a linear combination. Each of the eigenfaces is of dimension $N \times N$, and can be interpreted as an image.

We expect that some linear combination of a small number of eigenfaces will yield a good approximation to any face in a database, and (of course) also to a candidate for matching. An image can therefore be reduced to an *eigenvector* $\vec{B} = b_i$ which is the set of best-fit coefficients of an eigenface expansion. Now we can compare a candidate's eigenvector against each of those in a database through a distance matching, for example, a Cartesian measure. The distances found against the database yield both a rank-ordering and a linear closeness measure.

Sirovich and Kirby used an ensemble of 115 images of Caucasian males, digitized and preprocessed in a controlled manner, and found that about 40 eigenfaces were sufficient for a very good description of their set of face images. The root-mean-square pixel-by-pixel errors in representing cropped images (background clutter and hair removed) were about 2%.

Turk and Pentland [PENT91] refined their method, by adding preprocessing and expanding the database statistics. They, too, found that a relatively small number of eigenfaces drawn from a diverse population of frontal images is sufficient to describe an arbitrary face to good precision.

The runtime performance of an eigenface system is very good. The construction of a set of eigenfaces is computationally intense, but need only be done infrequently. A set could in theory be developed once and for all time that adequately describes all of man and womankind, including persons yet unborn.

Given a candidate image, the task is finding its characteristic eigenvector, which is computationally equivalent to solving a least mean-squares minimization

problem, albeit with N^2 datapoints and p unknowns. This is a matter of a few seconds work on a modern machine.

The final task, of matching an image against a database, is a matter of computing distances between the candidate's eigenvector and those of the database. Using a Cartesian distance, the unit computation is one of adding the squares of p variables, each of which is a difference between two eigenvectors. Even without special hardware, this can be reduced to a few dozen microseconds per comparison, making possible the search of a database of 100,000 images in a few seconds. Pentland claims that a match against a more modest database (a few hundred images) can be achieved on standard hardware (Sun Sparc stations) at frame-rate of the capturing video camera.

The robustness of eigenfaces to facial distortions, pose and lighting conditions is fair. Although Sirovich and Kirby were pleased to discover that their system found matches between images with different poses, the quality of matching clearly degrades sharply with pose, and probably also with expression, as Phillips discovered.

10. Wavelets and Elastic Matching

Wavelets were first proposed by Dennis Gabor as a tool for signal detection in noise.

A complex Gabor wavelet is described by the equation

$$\psi_{\vec{k},\sigma}(\vec{x}) = \exp\left(-\frac{k^2|\vec{x}|^2}{2\sigma^2}\right) \exp(i\vec{k}\vec{x})$$

k determines the oscillating frequency of the wavelet, and the direction of the oscillation. σ describes the rate at which the wavelet collapses to zero as one moves from its center outward. One can view a wavelet as a continuous wave (the second $\exp(i)$ function) propagating in the k direction, modulated by a Gaussian envelope (the first $\exp()$ function).

The general idea is to describe an arbitrary two-dimensional image function $I(x, y)$ as a linear combination of a set of wavelets. In image applications, the x, y plane is first subdivided into a grid of non-overlapping regions, which may or may not be rectangular. At each grid point, the local image is decomposed into a set of wavelets chosen to represent a range of frequencies, directions and extents that "best" characterize that region. Each grid point will then be characterized by a set of wavelets varying in k , but with a constant σ . By limiting k to a few values, the resulting coefficients become largely invariant to translation, scale and angle, though not completely. Of course, the initial choice of the subdivision grid implies an arbitrary translation.

The finite wavelet set at a particular grid point forms a feature vector called a *jet*. The set of jets will now

characterize the image. These comprise a relatively small set of numbers by which two images may be compared.

10.1. Application to face recognition

In the work of Lades, von Malsburg and others [LADES93], each jet consists of 5 logarithmically spaced frequency levels and eight orientations. Their initial grid had 7×10 points spaced by 11 pixels each. Thus an image covered by the grid will be characterized by a total feature vector of 110 values. These are not necessarily statistically independent, owing to some overlap between the grid regions, and other correlations found in face images.

Unfortunately, Gabor wavelets are not orthogonal and complete. The lack of orthogonality implies a computational overhead in finding an optimal decomposition. However, as in the PVD method, this need only be done once for each face in a database. Since the operation is local, only a small number of pixels are involved in each convolution.

10.2. Elastic grid matching

Matching a jet feature set with a fixed grid will only be effective if the face image is carefully preprocessed, and the face is reasonably expressionless. Gabor filtering relieves some, but not all, of the burden of preprocessing. In order to accommodate different scales, translations, and even facial expressions and pose variations, Lades and von Malsburg [LADES93] discovered that the grid could be elastically distorted (within constraints), in order to find a best match between two images.

The graph matching is first performed by large translations in order to center the grid on the face. It is followed by small local distortions, chosen in such a way to maintain a planar grid. At each stage, certain of the jets must be re-computed, and their feature vectors combined to obtain a quality measure. The matching is improved by changing the local coordinate system of the jets to correspond to the graph distortion, i.e. when the new grid points are closer together, the coordinate axes are similarly compressed.

It is perhaps not surprising that elastic matching is quite effective in dealing with changes in posture and expression. Small rotations of the face image around any axis result in what might be considered to be the same face, except for local scale and rotation transformations. The jets therefore should be nearly alike. Changes in expression will affect the jets somewhat, but the grid distortion will to some extent track these changes, and after all, the face is essentially an elastic membrane pushed about by a complex of distributed muscles. No one can remove a freckle or wrinkle through a change in facial

expression, though its relative position changes to some extent, and the elastic matching approach seems consistent with this observation.

10.3. Performance

The time performance of a *rigid* grid Gabor filtering system is comparable to that of the eigenfaces. Given that each image is characterized by a small set of jet vectors, the matching problem is the same, i.e. one of comparing Euclidean distances between the vectors of a candidate and each of the database members. If the grid can be positioned, scaled and rotated into a canonical position (for example, by first locating the eyes and mouth) by a preprocessor, then a high matching performance on conventional processors can be expected.

However, if elastic matching is employed, the time performance will be relatively poor. Elastic grid matching must be performed on the candidate against *each* database element, a task which requires high speed, and preferably parallel, processors. Lades [LADES93] used a system consisting of 23 transputers operating in parallel. Each transputer is a microprocessor with integrated support for message-passing and distributed memory. The convolution of a 128×128 pixel image with 40 wavelet filters requires less than 7 seconds. Comparison of an image to a stored object takes between 2 and 5 seconds on one transputer. A recognition run, comparing one image against a gallery of 87 stored objects (which is amenable to parallel computation) then takes about 25 seconds, a matching rate which is an order of magnitude smaller than with eigenfaces. But note that elastic matching is more robust with respect to pose and expression.

11. Bibliography

An extensive bibliography, including many online papers and tutorial materials, can be found online, thanks to the work of Peter Kruizinga:

<http://www.cs.rug.nl/~peterkr/FACE/frhp.html>

Specific references cited in this paper are as follows:

BIOM97: "Survey: Face Recognition Systems", in *Biometric Technology Today*, July/August 1997. Contains a list of commercially available facial systems and estimates of their effectiveness.

COTT90: 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.

FACEIT97: See the Web page for papers by J. Attkick at the Rockefeller Institute. Details on Faceit are lacking.

GALT88: Francis Galton, "Personal identification and description", in *Nature*, June 21, 1888, p 173-177.

LADES93: M. Lades, J. Vorbruggen, J. Buhmann, J. Lange, C. V. D. Malsburg, and R. Wurtz, "Distortion invariant object recognition in the dynamic link architecture", *IEEE Trans. Comput.*, vol. 42, no. 3, pp. 300-311, 1993. Describes Gabor wavelet filtering and elastic matching.

LADES97: Jun Zhang, Yong Yan, and Martin Lades, "Face Recognition: Eigenface, Elastic matching, and neural nets", in *Proc. IEEE*, vol. 85, No. 9, Sept. 1997, p 1423-1435. Follow-on to LADES93.

PENT91: Matthew Turk and Alex Pentland, "Eigenfaces for recognition", in *Journal of Cognitive Neuroscience*, vol. 3, No. 1, 1991, pp 71-86. A fundamental paper on the eigenface approach.

PENT94: M. Bichsel and A.P.Pentland, "Human face recognition and the face image set's topology", in *CVGIP: Image Understanding*, vol 59, No. 2, March, pp 254-261, 1994, Academic Press.

PHIL96: P. Johnathon Phillips, Patrick J. Rauss, and Sandor Z. Der, "FERET (Face Recognition Technology) Recognition Algorithm Development and Test Results", *Army Research Laboratory, ARL-TR-995*, October 1996. Contains a large number of comparative results, mostly using rank-ordering.

PHIL97: P. Johnathon Phillips, Hyeonjoon Moon, Patrick J. Rauss, and Syed A. Rizvi, "The FERET Evaluation Methodology for Face-Recognition Algorithms", to appear in *Proc. IEEE Conf. On Computer Vision & Pattern Recognition 97*, June 17-19, 1997.

SIRO87: L. Sirovich and M. Kirby, "Low-dimensional procedure for the characterization of human faces", in *J. Opt. Soc. Am. A*, Vol. 4, No. 3, March 1987, pp 519-524. A key paper on eigenfaces.