FACE RECOGNITION

In the context of computer engineering, *face recognition,* broadly defined, is the study of algorithms that automatically process images of the face. Problems include recognizing faces from still and moving (video) images, analyzing and synthesizing faces, recognizing facial gestures and emotions, modeling human performance, and encoding faces. The study of face recognition is multidisciplinary and draws on the fields of computer vision, pattern recognition, neuroscience, and psychophysics, with many approaches spanning more than one of these fields.

The face is a curved three-dimensional surface, whose image varies with changes in illumination, pose, hairstyle, facial hair, makeup, and age. All faces have basically the same shape, yet the face of each person is different. From a computer vision perspective, the goal of face recognition is to find a representation that can distinguish among faces of different people, yet at the same time be invariant to changes in the image of each person.

This goal has led to numerous approaches to face recognition, with roots in computer vision, pattern recognition, and neuroscience. Historically, the initial emphasis was on recognizing faces in still images in which the sources of variation were highly controlled. As the field progressed, emphasis moved to detecting faces, processing video sequences, and recognizing faces under less controlled settings. Important to the advancement of face recognition was the establishment of large databases of faces and standard protocols for testing algorithms.

This article is written from a computer vision prospective, but the reader should be aware of work in related areas. For a more comprehensive introduction to computer vision approaches, see Chellappa et al. (1); for a more general overview, see Wechsler et al. (2); and for papers on face and gesture recognition see Refs. 3–5.

In human beings, face recognition is a basic cognitive function that is a critical tool in human interaction. Faces allow us to identify with whom we are communicating, and they assist in interpreting a person's emotions and reactions. Knowing with whom we are communicating is an integral part of determining what we say and how we behave and speak to a person. A person's facial expression provides us feedback on how we are being received, how effectively we are communicating, or whether we are stepping outside social bounds.

The importance of this aspect of cognition motivates psychophysicists and neuroscientists to study and model human face recognition performance. Psychophysicists who study face recognition are primarily interested in the properties of the human system for visually processing faces. Their studies determine these properties by comparing human performance on different face-recognition tasks. Among the objects of study are the effects of lighting or of pose changes on face recognition, the effects of facial changes over time on the ability to recognize faces, and the perception of a person's gender and race. Another goal is to develop biologically plausible models of how the human visual system processes faces. A model is biologically plausible if it models observed performance. For an overview of biological models, see Valentine (6).

Neuroscientists are primarily interested in the physiological and biological bases of face recognition. While neuroscience includes the study of the physiology of the human visual system, the aspect of neuroscience most closely related to algorithmic face recognition is the physiologically based mathematical modeling of the human visual system, which has served as the foundation of a number of face recognition algorithms (see the section on wavelet-based algorithms).

APPLICATIONS

One of the great motivations for developing face recognition algorithms is that there are potentially many applications; including law enforcement and security; human/computer interfaces; image compression and coding of facial images and the related areas of facial gesture recognition; and analysis and synthesis of faces. There are three basic scenarios that face-recognition systems might address: (1) identifying an unknown person, (2) verifying a claimed identity of a person, and (3) analyzing a face in an image (such as for emotional content or for efficient compression of a facial image).

In law enforcement and security, the primary interest is in identification and verification. A major identification task in law enforcement is searching a database of known individuals for the identity of an unknown person. A potential application is thus the electronic mugbook, where mugshots would be digitized and stored electronically. The input to the electronic mugbook could be a new mugshot, witness sketch, or surveillance photo to be compared to the images in the electronic mugbook; the output would be the top N matches, ordered by their similarity to the input image. This would allow a person to examine the most likely mugshots, as opposed to searching a mugbook randomly. For this class of applications, systems are not designed to replace humans, but rather to provide assistance and improve human performance in executing potentially sensitive tasks.

A similar application is maintaining the integrity of an identity database, which could be compromised by (1) a person having two identities, or (2) two people having the same identity. Both types of errors can result in degradation of recognition performance or in false accusations being made. The first type of error can result from a person making false claims: For example, getting a driver's license under a second name. Robust computer face recognition could detect potential compromises by comparing all faces in the database and reporting suspect matches.

The main application for security is verification. The input to a verification system is a facial image and a claimed identity of the face; the output is either acceptance or rejection of the claim. Depending on the application, the output is a measure of belief in the claim. Potential applications include controlling access to buildings or computer terminals, confirming identities at automated teller machines (ATMs), and verifying the identity of passport holders at immigration ports of entry.

Video-based systems could be used to monitor hallways, entrances, and rooms. The system would detect, track, and count faces, and recognize the detected faces. A system that monitored a hall or doorway could act as a sophisticated identification/verification system. On the other hand, for room monitoring, the system might report results at various levels: identifying a person if possible, tracking where a person went, reporting how many people were in the room at a particular moment, and reporting where they were.

Currently, interactions with a computer require the use of a keyboard or mouse, with a person assuming an active role and the computer a passive one. A goal for future human/computer interfaces is for the computer to assume an active role. A critical component of such a system would be a face monitoring and recognition system. The recognition system would confirm the identity of a user, and the monitoring portion would let the computer read a person's expressions. From the expressions, the computer would know how to react to a person. For example, if a user looked confused, the computer could provide additional help. Thus, identity and facial expression would facilitate interaction between humans and computers, much as they assist communications among humans.

Another application for face recognition would be intelligent, automated kiosks, with an enhanced computer interface, that would provide information or services to people. For example, an intelligent kiosk could provide directions or information to people, and based on their facial expressions, could provide greater detail.

With the advent of inexpensive video cameras it is possible to consider large-scale production of video telephones. One obstacle is the difficulty of transmitting high-quality facial imagery over existing low-bandwidth channels. A proposed method of transmitting high-fidelity images is to compress facial video sequences through encoding methods specialized for faces, where the encoding method is derived from the class of representations used for face recognition. These methods work by encoding a face as the parameters of the representation, transmitting the parameters, and using the parameters to reconstruct the face at the receiving end.

FACE RECOGNITION VERSUS PATTERN RECOGNITION

The face-recognition problem is substantially different from classical pattern-recognition problems such as character recognition. In classical pattern recognition, there are relatively few classes, many examples per class, and substantial differences between classes. With many examples per class, algorithms can classify examples not previously seen by *interpolating* among the training samples. On the other hand, in face recognition, there are many different individuals (classes), only a few images (examples) per person, and all faces have the same basic shape and spatial organization. Because the number of training samples is small, algorithms must recognize faces by *extrapolating* from the training samples.

For face-recognition algorithms to be successful, they must either implicitly or explicitly extrapolate from the images in the database how a person will look under varying viewing conditions. The variations in faces are results of combinations

Figure 1. Three pairs of faces demonstrating variability in facial images. (Facial images courtesy of the FERET database.)

of changes in appearance, illumination, pose, and the method of acquisition of the image. To be able to identify faces, one must separate variations that are due to external factors from those that represent actual differences between faces. This is illustrated in Fig. 1, which shows three pairs of images. For each pair of images, the task is to decide whether the faces are from the same person. [Pairs (a) and (b) in Fig. 1 are from the same person and pair (c) is from different people.] This figure also suggests a potential limitation of face recognition. If an algorithm can distinguish between the faces in Fig. 1(c), can the same algorithm reliably report that the pair in Fig. 1(a) are the same face?

GENERAL OUTLINE OF ALGORITHMS

The starting point for the development of computer face recognition was the recognition of faces from still images. The limited computational power available in the 1970s precluded the possibility of working on anything other than still images. The cost of computational power is still one of the main limiting factors in face recognition. As the price of computers has decreased, there has been a concomitant increase in the complexity and sophistication of the algorithms and the problems addressed.

The initial approach for the development of face-recognition algorithms was geometric feature matching (7,8). In this approach, a set of fiducial features is located (these are point features: such as the tip of the nose or center of the left eye). A face is then represented as a set of statistics computed from the distance between features (the ratio of the distance between the eyes and the length of the nose, for example). The main drawback to this approach was the difficulty of developing algorithms that could reliably, accurately, and automatically locate the fiducial features. One solution was for a human to locate the fiducial features, which is time consuming.

The next set of approaches to face recognition, and to date the most popular and successful, are viewed-based. View-based algorithms represent a face as a set of images (each image is a different view of the face). View-based algorithms are popular because they avoid the difficult problems associated with extracting three-dimensional or other models from an individual image or set of images (the algorithms discussed in this article are view-based).

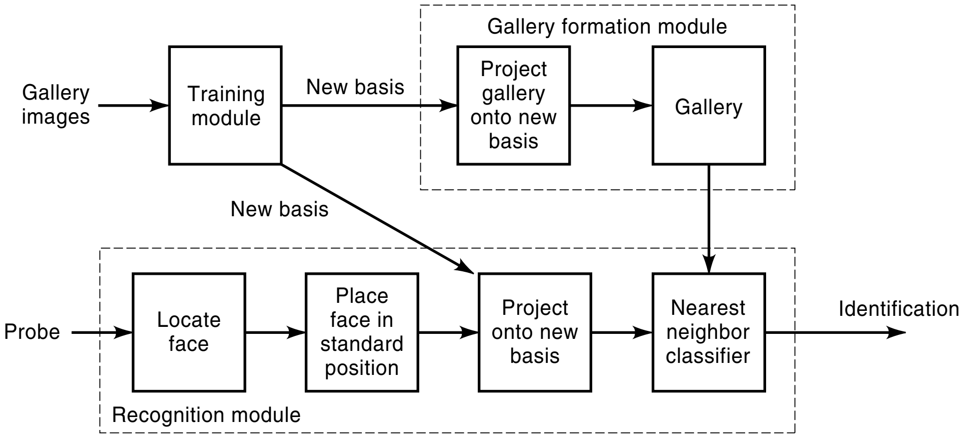

Figure 2. The generic organization of a face-recognition algorithm.

A drawback to view-based algorithms is that their performance declines with changes in pose and illumination. These variations can be compensated for if the face is explicitly modeled as a three-dimensional (3-D) object. If a 3-D model is used, an algorithm can compare faces of different poses by rotating the 3-D model of the first face to the pose of the second face. A 3-D surface is independent of illumination: from a 3-D model, one can compute the appearance of a face under different illuminations. Thus, when two faces with different illumination are compared, the illumination of the first face is converted to that of the second.

The technical challenges of applying 3-D methods to facial images are in extracting the 3-D model of the face. Two basic methods of computing a 3-D representation are (1) using video (this method infers shape from motion) and (2) using still images (this method infers shape from shading). An alternative is to collect images that are explicitly 3-D, such as laser scans or structured light (9,10). The main drawbacks to these systems are the cost and practicality of fielding them.

View-based algorithms differ primarily in how faces are represented. Most view-based algorithms have the same general structure, which consists of training, gallery (database) formation, and recognition modules. Figure 2 shows a schematic diagram of a generic face-recognition algorithm.

In the training module, the representation is determined. The representation can either be set by the algorithm designer or learned from a set of training images. Whereas an image of a face is stored as a set of pixel values, the algorithm's representation of the face is stored as a set of coefficients, which are parameters of the representation.

The gallery formation module processes a set of images $\{\mathscr{I}_k\}_{k=1}^M$ of known individuals into the representation used by the face recognition algorithm. The set of known individuals is referred to as the *gallery* (an image of an unknown face presented to the algorithm is a *probe,* and the collection of probes is called the *probe set*). To simplify the explanation of face-recognition algorithms, we assume that the gallery images are still, the face occupies most of the image, and there is one image per person in the gallery (extension to different scenarios is conceptually straightforward, but can be technically demanding). The images $\{\mathscr{I}_k\}_{k=1}^M$ are stored in the gallery by being converted to the new representation. The images in the new representation are denoted by $\{\mathbf{g}_k\}_{k=1}^M$.

The design of the recognition module depends on the application that the algorithm addresses. For an identification applications, the module identifies the face in a probe $\mathbf{p}$; for a verification applications, the module verifies the claimed identity of a probe $\mathbf{p}$. The first three steps are the same for both applications: First, the face is located in the image. (Locating faces is different from detecting them: For locating, we know that there is one face in the image, whereas, in detection, one does not know how many faces, if any, there are in the image.) Second, the face is placed in a standard position, which usually places the eyes in a fixed location, masks out the background and hair, and normalizes the dynamic range of the facial pixels. Third, the face in the probe is converted to the desired representation (projected onto the basis learned in the training module). For identification applications, a probe is identified by being compared with each person in the gallery. This is done with a similarity measure, which is a measure of the likelihood that two facial images are of the same person. The probe is identified as the person in the gallery with which it has the highest similarity score (nearest neighbor classifier). Usually, the output is a gallery sorted by the similarity measure. For verification applications, the identity is verified if the similarity measure between the probe and corresponding gallery image is above a given threshold. The setting of the threshold is determined by the desired performance of the face recognition system (see the section on databases and evaluation).

PROJECTION-BASED ALGORITHMS

An image is inherently a two-dimensional object, which can be stored as an $M \times N$ two-dimensional array of pixel values. This is not the only possible representation, however. The same image can as easily be represented as a vector in $\Re^V$, where $V = M \times N$, and each pixel corresponds to a dimension. For example, the 2×2 image

$$\begin{bmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \end{bmatrix} \tag{1}$$

becomes the vector

$$X = (x_{11}, x_{12}, x_{21}, x_{22})^T \tag{2}$$

an element of $\Re^4$.

Even for relatively small images, V is too large for statistical and pattern recognition applications. For a 100×100 image, V is equal to 10,000. To efficiently identify faces, one needs to find a V_p dimensional subspace of $\Re^V$, $V_p \ll V$. The subspace is chosen to allow for efficient and accurate recognition of faces.

One of the first view-based approaches to face recognition was the *eigenface* algorithm of Turk and Pentland (11), which is based on principal component analysis (PCA). PCA generates an orthonormal basis $\{e_1, \ldots, e_{V_p}\}$ that efficiently (in a linear least squares sense) captures the statistical variance of the faces in a training set (12). The premise for this approach is that there is a correspondence between the variance of a set of faces and the ability to recognize faces.

This orthonormal basis is generated from a training set $\{X_1, \ldots, X_N\}$ of N facial images. The training set of images is normalized by removal of the ensemble average $\overline{X} = \sum_i X_i$ to form the normalized set $Y_i = X_i - \overline{X}$. From the normalized set the matrix A is formed by concatenation of the vectors Y_i as rows, that is,

$$A = \left[Y_1 | \cdots | Y_N\right] \tag{3}$$

From A, the covariance matrix $C = AA^T$ is computed. Since the size of A is $V \times N$, C is a $V \times V$ matrix. The basis elements are the eigenvectors of C. A vector $\boldsymbol{e}$ is an eigenvector of C, and λ is an eigenvalue if

$$C\boldsymbol{e} = \lambda\boldsymbol{e} \tag{4}$$

Because C is a symmetric matrix, its eigenvectors $\boldsymbol{e}_i$ are orthonormal and its eigenvalues λ_i are nonnegative (13). The eigenvectors $\boldsymbol{e}_i$ are the same size as the image's X_i and can be interpreted as images, which is the origin of the name eigenface (Fig. 3). The eigenvectors and eigenvalues are ordered so that $\lambda_i \geq \lambda_j$ when $i < j$. The variance of the training set projected onto the basis vector $\boldsymbol{e}_i$ is equal to λ_i; or mathematically, λ_i; = Var($\langle X, \boldsymbol{e}_i\rangle$), where *Var* is the variance of a set, and $\langle \mathbf{a}, \mathbf{b}\rangle = \sum_i a_i b_i$ is the inner product of the vectors $\mathbf{a} = (a_1, \ldots, a_n)$ and $\mathbf{b} = (b_1, \ldots, b_n)$.

The rationale for selecting the PCA basis is that it captures the variance in the training set. This is because for $m \leq N - 1$, the subspace spanned by $\{\boldsymbol{e}_1, \ldots, \boldsymbol{e}_m\}$ maximizes the variance of the training set over all subspaces of dimension m; and the variance is equal to $\sum_{i=1}^{m} \lambda_i$. Since there are only $N - 1$ nonzero eigenvectors, the subspaces of dimension greater than $N - 1$ are not of interest. In fact, the subspace generated by the first $N - 1$ eigenvectors is the same as the subspace spanned by the normalized training set Y_i.

Even for small images, it is computationally difficult to find the eigenvectors of C: For example, if the image size is 100×100, then C is a $10{,}000 \times 10{,}000$ element matrix. Instead, one computes the eigenvectors of A^TA, and obtains the eigenvectors of C by noting that $A^TAA^Tu = \lambda A^Tu$.

The face in image Y is represented as the vector $(\langle Y_i, \boldsymbol{e}_1\rangle, \ldots, \langle Y_i, \boldsymbol{e}_{N-1}\rangle)$, the projection onto the subspace spanned by the eigenvectors. A face that was originally represented by V numbers is now represented by $N - 1$ coefficients in eigenspace.

The off-line portion of a principal-components face-recognition algorithm generates a set of eigenvectors $\{\boldsymbol{e}_i\}$, which are basis functions for determining the representation of a face in the on-line portion of an identification algorithm. For an identification application, the gallery of M known individuals is $\{\mathscr{I}_k\}_{k=1}^{M}$, and each person in the gallery is represented by

$$g_k = \left(\left\langle(\mathscr{I}_k - \overline{X}), \boldsymbol{e}_1\right\rangle, \ldots, \left\langle(\mathscr{I}_k - \overline{X}), \boldsymbol{e}_{N-1}\right\rangle\right) \tag{5}$$

where $\overline{X}$ is the mean image.

The representation of a probe in eigenspace is denoted by $\mathbf{p}$ and is computed in the same manner as the gallery images [Eq. (5)]. The similarity measure between the probe $\mathbf{p}$ and the gallery image $\mathbf{g}_k$ is the Euclidean distance between them and is denoted by $d_k = \|\mathbf{p} - \mathbf{g}_k\|$. The probe is identified as the person $\hat{k}$ with which it has the smallest similarity measure: that is, $d_{\hat{k}} = \min_k d_k$.

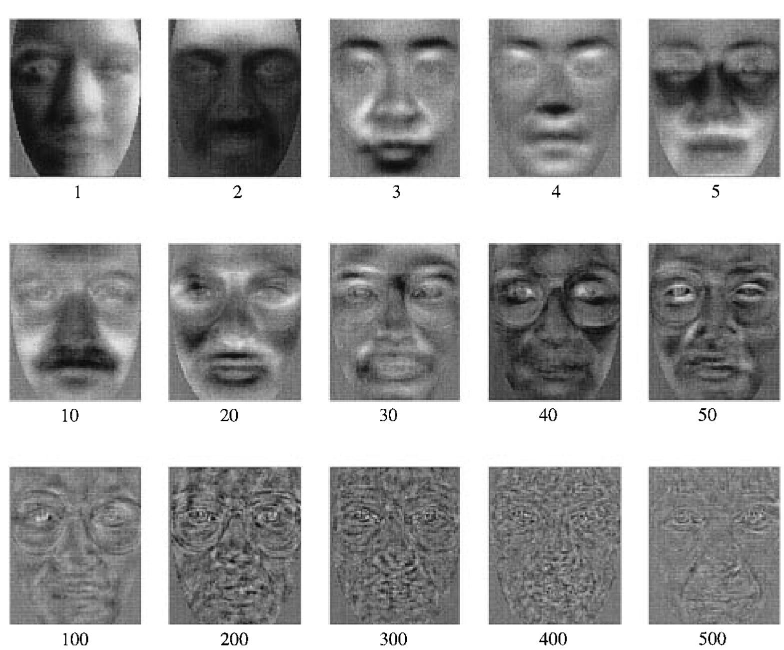

Figure 3. Select eigenfaces computed from a set of 501 images (the number under the eigenface is its number). Eigenfaces 1 and 2 encode illumination directions; eigenfaces 3 to 100, the overall shape of faces; eigenfaces 200 to 500, the minor variations among faces (sometimes interrupted as high frequency noise). (Facial images courtesy of the FERET database.)

The identification and verification implementations of this algorithm differ. For the verification implementation, the person in probe $\mathbf{p}$ claims the identity of person ℓ in the gallery. The claim is accepted if $\|\mathbf{p} - \mathbf{g}_\ell\| \leq \Delta$, where Δ is a threshold that determines system performance (see the section on databases and evaluation). A person's identity being verified does not necessary imply that an identification implementation of the same algorithm would report the identity of probe $\mathbf{p}$ as person ℓ: or, $\|\mathbf{p} - \mathbf{g}_{\hat{k}}\| < \|\mathbf{p} - \mathbf{g}_\ell\|$. Also, a probe being identified as person $\hat{k}$ does imply verification as person $\hat{k}$: that is, $\|\mathbf{p} - \mathbf{g}_{\hat{k}}\| > \Delta$.

Moghaddam and Pentland (14) adapted PCA to face detection and location. PCA selects the basis element without explicitly considering differences between people (in statistical language, classes). In contrast, Fisher discriminant analysis searches for a basis that separates classes (people) according to a separation criterion (12), where the criterion is a function of the within-class and between-class scatter matrices. This technique has been applied to face recognition by Etemad and Chellappa (15) and Swets and Weng (16).

PCA and Fisher discriminant analysis are not explicitly two-dimensional and the representations are global. Penev and Atick (17) introduce a technique called local feature analysis, which generates a set of filters that are explicitly two-dimensional, local, and tuned to a set of training images.

WAVELET-BASED ALGORITHMS

A wavelet encoding of facial images produces a multiresolution representation that is local and explicitly two-dimensional. This encoding is the basis of the dynamic link architecture (DLA) of Wiskott et al. (18), which represents facial features by Gabor wavelets and the global organization of the face by a graph.

A two-dimensional Gabor wavelet is the Gaussian probability-density function modulated (multiplied) by either a planar cosine or sine wave (19). The Gabor wavelet modulated by a cosine function is also called the even-phase component (in the image processing literature, this component corresponds to a ridge detector) and the sine part is the odd-phase component (the edge detector in image processing). The Gabor wavelet is characterized by three parameters: the scale (standard deviation) of the Gaussian probability density σ, the frequency of the cosine and sine wave ω, and the direction of the wave (filter) in the plane θ. The even-phase function is

$$C_e \exp\left(\frac{x^2 + y^2}{\sigma^2}\right) \cos\left(\pi[u_0 x + v_0 y]\right) \tag{6}$$

and the odd-phase function is

$$C_o \exp\left(\frac{x^2 + y^2}{\sigma^2}\right) \sin\left(\pi[u_0 x + v_0 y]\right) \tag{7}$$

The normalizing constants C_e and C_o scale the wavelets so that the energy (L_2 metric) of the wavelets is 1. The direction of the wavelet is $\theta = \arctan(u_0/v_0)$.

In DLA, a set of Gabor wavelets is grouped together to form a Gabor jet. The parameters σ and ω are selected so that: (1) $\sigma/\omega = C_1$, where C_1 is a constant, and (2) $\sigma = C_2 2^\ell$ for a small set of integer values for ℓ, and where C_2 is a constant. Each octave contains both phases uniformly sampled in the direction parameter θ. The response to a Gabor jet at location v in an image is the vector J_v. The vector J_v encodes a facial feature at different scales and directions.

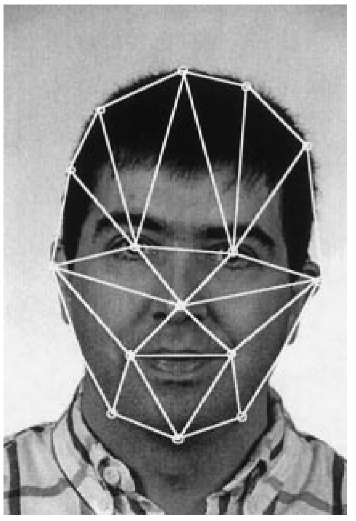

Figure 4. A graph G that models the global shape of the face in DLA. (Facial images courtesy of the FERET database.)

In DLA the geometric structure of the face is modeled by a graph G, where the vertices of the graph correspond to fiducial facial features (Fig. 4). (The same graph is used for each face.) A Gabor jet is placed at each vertex of the graph, where the jet encodes the structure of the face at that feature. The response to the jet placed at vertex v in a gallery image $\mathbf{g}_k$ is denoted by the vector J_v^k, and the corresponding set of coefficients in probe $\mathbf{p}$ is $J_v^{\mathbf{p}}$. The similarity between jets J_v^k and $J_v^{\mathbf{p}}$ is the angle between two vectors: mathematically,

$$d_v(J_v^k, J_v^{\mathbf{p}}) = \frac{\langle J_v^k, J_v^{\mathbf{p}} \rangle}{\|J_v^k\| \, \|J_v^{\mathbf{p}}\|} \tag{8}$$

The similarity between the probe $\mathbf{p}$ and gallery image $\mathbf{g}_k$ is

$$d_k = 1 - \frac{1}{N} \sum_{v \in V} d_v(J_v^k, J_v^{\mathbf{p}}) \tag{9}$$

where V is all the vertices in graph G. The distance d_k is the average distance between jets scaled so that $d_k = 0$ when the jet values are identical. The method for identifying and verifying probes using d_k is the same as in PCA.

The vertices of G are placed on the same fiducial feature in each face. However, the metric arrangement of features in each face is slightly different. One can capture this difference by comparing the difference in the edge lengths of the graphs placed on two faces, and modifying d_k to include this comparison. Let $\Delta(e)$ be the length of an edge and E the edges of G; the modified similarity between a probe and gallery image is then

$$d_k = d_k + \frac{\lambda}{E} \sum_{e \in E} (\Delta(e^k) - \Delta(e^{\mathbf{p}}))^2 \tag{10}$$

where λ determines the relative importance of the jets and graph deformations.

DATABASES AND EVALUATION

Face recognition is a fast growing area of research, in which numerous claims are made about progress and algorithm performance. To assess the conflicting claims and place face recognition on a solid experiment footing, we must have standard evaluation protocols and associated databases of facial imagery. The FERET evaluation, the de facto standard in face recognition, is one such method (Refs. 20–21).

The design of a face-recognition algorithm requires compromise among competing parameters (for example, speed versus accuracy). Such compromises mean that each algorithm has strengths and weaknesses: One algorithm may be better suited for a particular application than for others. A well-constructed evaluation procedure allows us to determine such suitability differences. An evaluation procedure can also help in the assessment of the state of the art of face recognition and point out future avenues of research. Knowing the state of the art is critical for deciding what applications can be met by existing systems. One does not want to field a system under conditions for which it was not designed or to make unrealistic claims for system performance.

In face recognition (and computer vision in general), the database and testing procedure are tightly coupled: They define the problem(s) against which an algorithm is evaluated. For example, if the faces are in predetermined position in the images, the problem is different from that for images in which the faces are randomly located. Other factors influencing the problem include image quality and variability in lighting, pose, and background within the test set of images.

Algorithms are designed from a development set of images. For some algorithms, the representation of the set of faces is learned from this set. Even if the representation is not explicitly learned, an algorithm's performance is dependent on this set because of the design decisions and tuning of numerous parameters in an algorithm.

One of the goals of testing is to determine the performance of an algorithm on images not in the development set. For a meaningful evaluation, the test image set needs to include a large number of images not in the development set.

The basic methods for evaluating algorithm performance are the closed and open universes. The closed universe models the identification scenario, where all probes are in the gallery. The open universe models verification, where there are probes outside the gallery.

The main performance measure for the identification scenario is the ability of an algorithm to identify the person in the probe. For identification, the question is not always "is the top match correct?", but "is the correct answer in the top n matches?" For example, in an electronic mugbook, performance can be measured by how many images the average person examines. The performance statistics are reported as a cumulative match score. (For example, for the curve in Fig. 5, the correct answer was Rank 1 for 80% of the probes scored, and the correct answer was Rank 10 or less for 87 % of the probes scored. In other words, the correct answer was in the top ten 87% of the time.)

The performance of a verification system is characterized by (1) the probability of correctly verifying an identity and (2) the false-alarm rate (or probability of a false alarm). A false alarm occurs when a system incorrectly verifies an identity. There is a trade-off between the probability of correct verification and the probability of false alarms, with the trade-off being reported on a *receiver operating characteristic* (ROC) (Fig. 6).

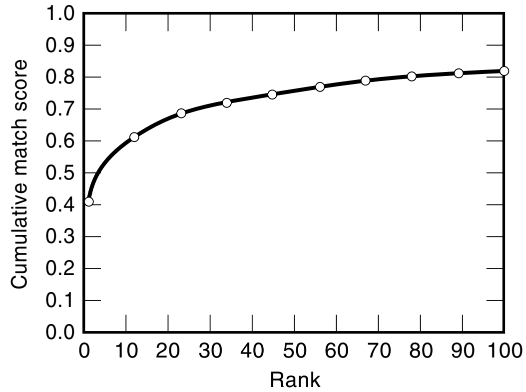

Figure 5. A cumulative match score graph for reporting identification results. The horizontal axis gives the rank, and the vertical axis is the percentage correct.

The false-alarm rate P_F is equal to $\hat{F}/F^*$, where F^* is the number of attempts by imposters to gain access, and $\hat{F}$ is the number of these attempts that are successful. The probability of correct verification P_I is equal to $\hat{V}/V^*$, where V^* is the number of legitimate access attempts, and $\hat{V}$ is the number of those attempts that were successful (in signal detection theory, P_V is known as the *detection probability*).

The P_F and P_V trade-off are illustrated by two extreme cases. At one extreme, if access is denied to everyone, then $P_F = 0$ and $P_V = 0$. At the other, if everyone is given access, then $P_F = 1$ and $P_V = 1$. For an algorithm, performance is not characterized by a single pair of statistics (P_V, P_F), but rather by all pairs (P_V, P_F), and this set of values is an ROC (Fig. 6).

Although, it is possible to compare algorithms from a ROC, it is not possible to compare two algorithms from a single performance point from each. Say we are given algorithm **A** and algorithm **B**, along with a performance point (P_V^A, P_F^A) and (P_V^B, P_F^B) (verification and false-alarm probabilities) from each. Algorithms **A** and **B** cannot be compared from (P_V^A, P_F^A) and

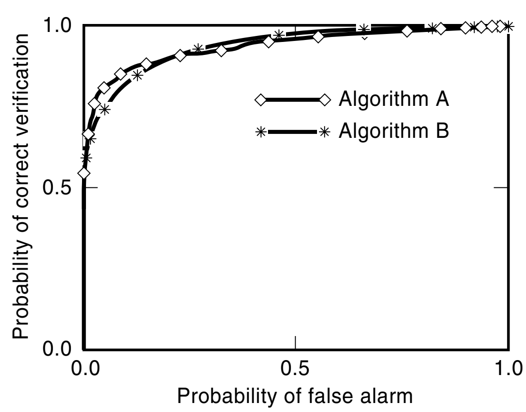

Figure 6. ROC for reporting the trade-off between probability of correct verification and probability of false alarm. The horizontal axis is P_F and the vertical axis is P_V. Results for two algorithms are reported. In one region, algorithm **A** has superior performance; in another region algorithm **B** is superior.

(P_V^B, P_F^B), for two primary reasons: The two systems may be operating at different points on the same ROC, or, for different values of P_F or P_V, one algorithm could have better performance (Fig. 6).

The setting of the performance point for an algorithm depends on the application. For an automated teller machine (ATM), where the overriding concern may be not to irritate legitimate customers, the P_V will be set high at the cost of the false-alarm rate. On the other hand, for access to a secure area, the false-alarm rate may be the overriding concern.

FUTURE DIRECTIONS AND CONCLUSIONS

Face recognition has emerged as an important area of research in computer vision, psychophysics, and neuroscience. The interest is driven by potential applications, scientific interest, and the availability of inexpensive computational power. Prototype face-recognition systems are emerging from the laboratory. The transition is being driven by the availability of inexpensive computer systems (at the time this article was written, Intel Pentium processors running on Windows NT).

One potential area of applications is in biometrics, which is the study of methods to identify humans for different signatures. These signatures include fingerprints, voice, irises, and hand geometry. The availability of large databases and evaluation procedures has helped face-recognition algorithms to mature and provided a quantitative measure of progress (20,21). Advances in face recognition will require the continued collection of databases and improved methods of evaluating algorithms. The improved methods need to be designed to provide a detailed analysis of the current strengths and weaknesses of algorithms and a characterization of the categories of images where future research is warranted. The collected database should reflect these findings. (The size of the database may not be a critical factor for supporting future research.) Evaluation methods have shown that an important area for future research is developing algorithms that are invariant to changes of illumination and to time lapses of more than one year between the acquisitions of images of a person (20). Other areas of active research are in compensating for changes in rotation and in 3-D methods (extracting 3-D information from still or video images).

Computational psychophysics is the study of modeling the human visual system. For face recognition it is the merging of face-recognition algorithms, psychophysics, and neuroscience to investigate how humans process images of human faces. Face recognition is of particular interest, because compared to other visual tasks, it is relatively constrained, but at the same time provides valuable insights into the workings of the human visual system.

A future direction of research is to combine face recognition with other biometrics. Law enforcement agencies are interested in fingerprints and faces because they systematically collect fingerprints and mugshots. Combining face and voice recognition is of interest for purposes of access control and in human/computer interfaces. The goal of multimodal imaging is to improve algorithm/system performance. For example, an objective of human/computer interfaces is to enable a human to speak to a computer. Use of lip movement and speech is a natural method to improve performance (in fact, humans use both these modes). For papers on multimodal recognition see Bigün et al. (22).

The successful development of algorithms that can process human facial images has the potential to affect our daily lives, from how we interact with computers, enter our homes or workplaces, and communicate over the phone or internet; and to contribute to our understanding of the basic cognitive functions of the human brain.

BIBLIOGRAPHY

1. R. Chellappa, C. L. Wilson, and S. Sirohey, Human and machine recognition of faces, a survey, *Proc. IEEE,* **84**: 705–740, 1995.
2. H. Wechsler, P. J. Phillips, V. Bruce, F. Fogelman Soulie, and T. Huang (eds.), *Face Recognition: from Theory to Applications,* Berlin: Springer, 1998.
3. M. Bichsel (ed.), *Int. Workshop on Automatic Face and Gesture Recognition,* Zürich: University of Zürich, 1995.
4. *Proc. 2nd Int. Conf. Autom. Face Gesture Recognition,* Los Alamitos, CA: IEEE Computer Society Press, 1996.
5. *Proc. 3rd Int. Conf. Autom. Face Gesture Recognition,* Los Alamitos, CA: IEEE Computer Society Press, 1998.
6. T. Valentine (ed.), *Cognitive and Computational Aspect of Face Recognition,* London and New York: Routledge, 1995.
7. T. Kanade, *Computer Recognition of Human Faces,* Basel and Stuttgart: Birkhauser, 1977.
8. A. Samal and P. A. Iyengar, Automatic recognition and analysis of human faces and facial expressions: a survey, *Pattern Recognition,* **25**: 65–77, 1992.
9. G. Gordon, Face recognition based on depth maps and surface curvature, *SPIE Proc. Geometric Methods Comput. Vision,* **1570**: 234–247, 1991.
10. A. J. O'Toole et al., Three-dimensional caricatures of human heads: distinctiveness and the perception of facial age, *Perception,* **26**: 719–732, 1997.
11. M. Turk and A. Pentland, Eigenfaces for recognition, *J. Cognitive Neurosci.,* **3**: 72–86, 1991.
12. K. Fukunga, *Statistical Pattern Recognition,* New York: Academic Press, 1989.
13. G. Strang, *Linear Algebra and its Applications,* 3rd ed., Fort Worth, TX: Harcourt Brace Jovanovich, 1988.
14. B. Moghaddam and A. Pentland, Probabilistic visual learning for object recognition, *IEEE Trans. Pattern Anal. Mach. Intell.,* **19**: 696–710, 1997.
15. K. Etemad and R. Chellappa, Discriminant analysis for recognition of human faces, *J. Opt. Soc. Amer. A,* **14**: 1724–1733, 1997.
16. D. L. Swets and J. Weng, Using discriminant eigenfeatures for image retrieval, *IEEE Trans. Pattern Anal. Mach. Intell.,* **18**: 831–836, 1996.
17. P. Penev and J. Atick, Local feature analysis: a general statistical theory for object representation, *Network: Computation in Neural Systems,* **7**: 477–500, 1996.
18. L. Wiskott et al., Face recognition by elastic bunch graph matching, *IEEE Trans. Pattern Anal. Mach. Intell.,* **19**: 775–779, 1997.
19. J. G. Daugman, Uncertainty relation for resolution in space, spatial frequency, and orientation by two-dimensional visual cortical filters, *J. Opt. Soc. Amer. A,* **2**: 1160–1169, 1985.
20. P. J. Phillips et al., The FERET evaluation methodology for face-recognition algorithms, *Proc. IEEE Conf. Compu. Vision Pattern Recognition,* 137–143, 1997.

21. P. J. Phillips et al., The FERET database and evaluation procedure for face recognition algorithms, *Image and Vision Computing J.,* to appear, 1998.

22. J. Bigün, G. Chollet, and G. Borgefors (eds.), *Audio- and Video-based Biometric Person Authentication,* Berlin: Springer, 1997.

P. Jonathon Phillips
National Institute of Standards
and Technology

FACIAL FEATURE EXTRACTION. See Maximum likelihood detection.

FACIAL IMAGES. See Face recognition.

FACSIMILE. See Facsimile equipment.