PERGAMON

Pattern Recognition 35 (2002) 2727–2738

PATTERN RECOGNITION
THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY

www.elsevier.com/locate/patcog

# Biometric perils and patches

Ruud M. Bolle*, Jonathan H. Connell, Nalini K. Ratha

*Exploratory Computer Vision Group, IBM Thomas J. Watson Research Center, Yorktown Heights, NY 10598, USA*

Received 31 October 2001; accepted 31 October 2001

**Abstract**

Biometrics authentication offers many advantages over conventional authentication systems that rely on possessions or special knowledge. With conventional technology, often the mere possession of an employee ID card is proof of ID, while a password potentially can be used by large groups of colleagues for long times without change. The fact that biometrics authentication is non-repudiable (hard to refute) and, yet, convenient, is among its most important advantages. Biometrics systems, however, suffer from some inherent biometrics-specific security threats. These threats are mainly related to the use of digital signals and the need for additional input devices, though we also discuss brute-force attacks of biometrics systems. There are also problems common to any pattern recognition system. These include "wolves" and "lambs", and a new group we call "chameleons". An additional issue with the use of biometrics is the invasion of privacy because the user has to enroll with an image of a body part. We discuss these issues and suggest some methods for mitigating their impact. © 2002 Pattern Recognition Society. Published by Elsevier Science Ltd. All rights reserved.

*Keywords:* Secure authentication; Threat model; Biometrics; Fingerprint; WSQ compression; Data hiding; Cancellable biometrics

## 1. Introduction

Today's prevailing techniques for *user authentication* involve mainly passwords and user IDs or magstripe magnetic cards and PINs. These methods suffer from several limitations. One of the main problems is that such systems can be fooled relatively easily. First of all, passwords, PINs, and magstripe cards can be easily shared among users of a system or resource. Moreover, passwords and PINs can be illicitly acquired (say) by direct covert observation. Once an intruder has the password, the person has total access to the associated resource. Hence, a major problem with current authentication technology is that there is no way to positively link the usage of a system to the actual user, i.e., the issue of "repudiation". Similarly, while critical credit card transaction information is sent over the web using secure encryption methods, the present practice is not capable of assuring that the rightful credit card owner pays for the transaction. In summary, in a networked environment where the access points to systems and resources are widely distributed geographically, remote authentication policies based on a simple combination of user ID and password, or, worse, simply based on possession, have become inadequate.

The consequences of incorrect and insecure authentication methods in commercial environments can be catastrophic. The value of a reliable user authentication is not just limited to computer access. Many other applications in everyday life could benefit from more reliable user authentication, e.g., banking, immigration and physical access control such as an airport. Thus biometrics technology is attractive because it provides true user authentication. Biometrics is a rapidly advancing area concerned with identifying a person based on their physiological or behavioral characteristics. Rather than checking the knowledge or possessions of the user, physiological or behavioral traits that are more or less unique to an individual are checked to authenticate the user. Examples of physiological biometrics include fingerprint, face, and iris; behavioral biometrics include speech pattern and signature.

While automated biometrics helps to alleviate many of the problems associated with the existing authentication methods, there are still weak points where these systems can be

* Corresponding author. Fax: +1-914-784-7455.
*E-mail addresses:* bolle@us.ibm.com (R.M. Bolle), jconnell@us.ibm.com (J.H. Connell), ratha@us.ibm.com (N.K. Ratha).

PII: S0031-3203(01)00247-3

attacked. Password-based systems are prone to brute-force dictionary attacks. Biometrics systems, on the other hand, require substantially larger efforts to attack. Although standard encryption techniques are useful in many ways to prevent security breaches, there are several new types of attacks possible in the biometrics domain. If biometrics is used in a supervised fashion, this may not be a concern. In remote unattended applications such as web-based e-business applications, however, there is ample time and opportunity available to carry out sophisticated security attacks.

Another problem with biometric authentication is the re-issuance of identity tokens. For authentication based on physical possessions, e.g., keys and badges, a token can be easily cancelled and the user can be reassigned a new one. Similarly, logical entities, such as user ID and passwords, can be changed as often as required. Yet, a user has only a limited number of biometrics such as one face, ten fingers, and two eyes. Furthermore, a biometrics authentication system uses private details of users and there is an immediate privacy concern about misuse of this information.

In this paper, we will discuss in detail a few of the problems unique to biometric authentication. We focus on fingerprint recognition; however, our discussion is very general and can be extended to other biometrics. First, in Section 2 we discuss authentication systems in terms of a pattern recognition system. We use this to identify an attack technique not present in password-based systems, i.e., replay attack. In Section 3 we explore other differences between conventional password- and biometrics-based systems. We briefly describe false accepts and rejects, issues that do not have their counterparts in password-based systems. Further, we study the bit strengths of fingerprints and compare this with passwords. As part of this analysis, we introduce a new animal, the chameleon, to Doddington's zoo. In Section 4 we briefly describe some techniques for validating the integrity of the biometrics signal and we present the concept of "cancellable biometrics", which is our approach to ensuring the privacy of enrolled users. Some conclusions are given in Section 5.

## 2. Biometrics systems

As shown in Fig. 1, a biometrics authentication system is a pattern recognition system. Excellent introductions to such biometrics systems can be found in Refs. [1,2]. We examine the architecture of a biometrics system and, based on this architecture, analyze the security aspects.

### 2.1. Pattern recognition system

A general biometrics-based authentication system, as shown in Fig. 1, consists of several stages, or modules. These modules are indicated by the numbers in the figure and perform the following functions:

1. An input device that captures and digitizes a sample of the biometrics signal, i.e., the signal acquisition module. This could be the microphone in a telephone, which is connected to a server through an A/D converter. It could be a stand-alone fingerprint scanner connected to a client through an input port, or it could be a fingerprint sensor that is more tightly integrated with the client. Either way, some sample live phenomena are sensed and digitized.
2. A feature extraction module that computes significant "landmarks" in this sample. A fingerprint image, for example, consists of a flow pattern of ridges and valleys where the ridges have endings and bifurcations. While human experts use up to 18 types of fingerprint features for matching, most fingerprint matching algorithms use only ridge ending and bifurcations. The ridge pattern of the fingerprint image is extracted by judicious image processing algorithms that localize ridges. This processing results in the ridge endings and bifurcations, collectively called minutiae.
3. A module for constructing an invariant representation (often called template), which uses these features to build a compact representation of the biometrics sample. Typically, the biometrics signal acquired from a user varies significantly from presentation to presentation. For this reason, this stage attempts to eliminate the known sorts of variation from the basic signal (e.g., in the form of fingerprint minutiae). During enrollment, such an invariant representation is stored in a database to represent the biometrics of a particular individual. Such a database can be centralized or distributed (for example, smart cards).
4. To authenticate a user against a claimed ID, the corresponding stored template is retrieved and matched against a new template derived from the current input signal. The output of this module is *YES* when the two templates match and *NO* when they are substantially different. The matching decision is often based on some score or degree of match, either a similarity measure (see Section 3.1) or a dissimilarity (distance) measure. The matcher arrives at a decision while taking into account geometric, lighting and other signal acquisition variables. Designing such measures is hard, as is matching representations using such measures. For fingerprints, the use of the Hough transform (as in Ref. [3]) or the exploitation of edit distances (as in Ref. [4]) are examples of methods for matching minutiae.

Note that password-based authentication is very similar to the biometrics system shown in Fig. 1. The keyboard where the password is entered is the input device (stage 1). The feature extraction process (stage 2) is the process of intercepting the typed password characters. The template construction process (stage 3) is similar to password encryption. The template database represents the database of

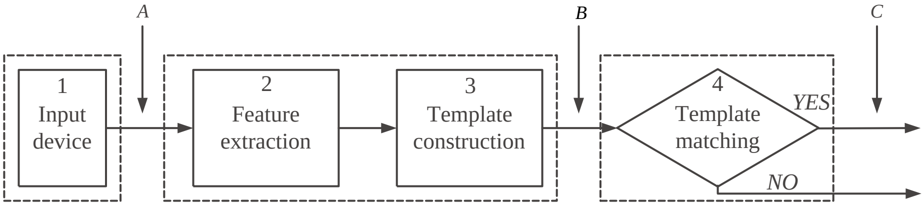

Fig. 1. The different stages of an authentication viewed as a pattern recognition system.

encrypted passwords. The matcher (stage 4) resembles the process of comparing encrypted passwords (passwords are often matched in their encrypted form for security reasons).

The system of Fig. 1 depicts a distributed pattern recognition system in that processing may be divided between the sensor, the client, and the server. This is reflected in the grouping implied by the three dashed boxes of Fig. 1. In such a system there are three communication channels, indicated by the letters in the figure:

A. Transmission channel $A$ between the sensing device and the feature extraction/template construction module. Such a channel transmits the input biometrics, such as an image of the user's finger.

B. The channel between the feature extraction/template construction module and the matching module. This communication channel will transmit the biometrics template. Alternatively, if the sensor has its own processing capability, the sensor may compute the biometrics template itself. The client then just passes this through to the server.

C. The channel between the matcher and the application. We do not consider the security aspects of this channel in this paper, since these are the same as in a password-based system.

Observe that there are many points of attack in a biometrics authentication system in addition to those shown in Fig. 1. Many security technologies and policies that are employed in today's password-based system are directly applicable to biometrics-based systems. For example, encrypted communication channels [5] can eliminate remote attacks. Additional attack points can be eliminated if the matcher and enrolled fingerprint templates reside in a secure location.

### 2.2. Biometrics system threats

There are various sources of attack, both on traditional and biometrics authentication systems. Schneier [6] describes some of the abuses that can occur with biometrics authentication systems. Observing the client–server model of Fig. 2, we focus on two types of attack. One of these, replay, is specific to biometrics systems.

- Brute-force attack at the sensor (client) or at the server—just like a brute-force attack on traditional authentication systems, which involves enumerating all possible passwords, such an attack on a biometrics system involves enumerating all possible biometrics signals or templates. We examine this stratagem as applied to fingerprints in Section 3.2.
- Resubmission of a previously acquired signal at the client—a recorded signal is replayed to the system, bypassing the sensor. Examples here include the presentation of a copy of a fingerprint image, or a recorded audio signal from a speaker. Section 4.1 discusses some techniques to detect such replays.

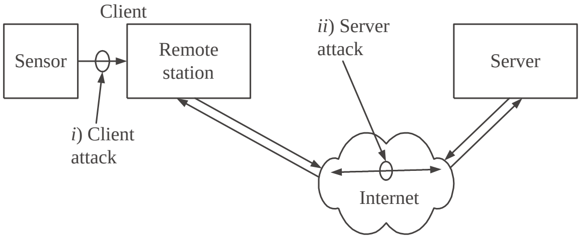

Fig. 2. The biometrics specific attack points in a biometrics authentication system are indicated.

As shown in Fig. 2, it is possible to attack both the client (i) and the server (ii) in these ways.

Another type of attack is the presentation of a fake biometrics at the sensor. In this mode of attack, a replica of a biometrics is presented to the system. Examples include a fake finger, a forged copy of a signature, or a face mask. Fake finger detection may be achieved at the sensor by, for instance, sensing finger conductivity or pulse. Solutions are still needed to detect other types of fake biometrics. When processing power increases, software algorithms will be able to detect such attacks by processing video rather than single still images [7].

## 3. Biometrics and conventional passwords

One difference between password- and biometrics-based systems is that there is no "fake password" detector equivalent to fake biometrics detection. It is unclear, even, what a fake password would be (Perhaps a word in the dictionary?). Also, in a password- or token-based authentication system no precautions need to be taken against replay attacks since there is no variation of the "signal" from one presentation to another.

Further, a password-based system always provides only one of two results, the password either matches or it does not. In a biometrics-based system, however, the situation is quite different. A decision must be made based upon a "degree of match". The system can therefore make errors and the tradeoffs between various error rates must be considered.

### 3.1. Error rates

The error rate of a pattern recognition system in general, and an automated biometrics system in particular, is dependent on several factors. Typically, the system performance reflects the quality of the input and enrolled biometrics signals, along with the basic characteristics of the underlying algorithms.

While biometrics systems most often store a compact representation of the sample, it is also possible, of course, to store the original signal itself. Either way, both the biometric signal samples and their representations/templates are patterns. That is, the pattern $P$ is a sample $S(\mathscr{B})$ of biometric $\mathscr{B}$, or it is a template that represents $S(\mathscr{B})$. Here, $\mathscr{B}$ can be viewed as uniquely associated with an individual. Therefore, $\mathscr{B} \equiv \mathrm{ID}(\mathit{individual})$, the identity of an individual. Authenticating a person can then be formulated in terms of hypothesis testing. Let the stored biometric sample or template be pattern $P' = S(\mathscr{B}')$ and the acquired one be pattern $P = S(\mathscr{B})$. In terms of hypothesis testing, we have

$$\begin{aligned} &\mathrm{H}_0 : \mathscr{B} = \mathscr{B}', \quad \text{the claimed identity is correct,} \\ &\mathrm{H}_1 : \mathscr{B} \neq \mathscr{B}', \quad \text{the claimed identity is } \textit{not} \text{ correct.} \end{aligned} \qquad (1)$$

Often, some similarity measure, $s = Sim(P, P') = Sim(S(\mathscr{B}), S(\mathscr{B}'))$, determines how similar patterns $P$ and $P'$ are. Decisions are then made based on a decision threshold $T$: $\mathrm{H}_0$ is decided if $s \geqslant T$ and $\mathrm{H}_1$ is decided if $s < T$.

For expression (1), deciding $\mathrm{H}_1$ when $\mathrm{H}_0$ is true, incorrectly rejects an individual. Such a false reject is also called a *false negative* or *Type I error*. Deciding $\mathrm{H}_0$ when $\mathrm{H}_1$ is true, on the other hand, results in the false acceptance of an individual, also known as *false positive* or *Type I error*. The False Accept Rate (FAR) and False Reject Rate (FRR) together characterize the accuracy (error rate) of a recognition system. The FAR and FRR are closely interrelated variables and depend strongly on the decision threshold $T$ (see Fig. 3). The distribution on the left is of scores from intruders, while the distribution on the right is of scores from genuine users. The decision threshold $T$ determines the tradeoff between FAR and FRR.

The error rates are a function of the match/non-match decision threshold as shown in Fig. 3. Often the interplay of the two errors is presented by plotting FAR against FRR with the decision threshold $T$ as the free variable. This plot is called the receiver operator characteristics (ROCs) curve.

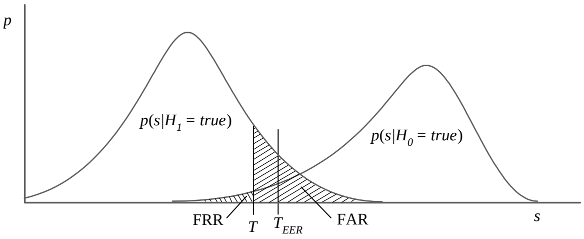

Fig. 3. There are two types of error rates in a biometrics authentication system: FRR and FAR.

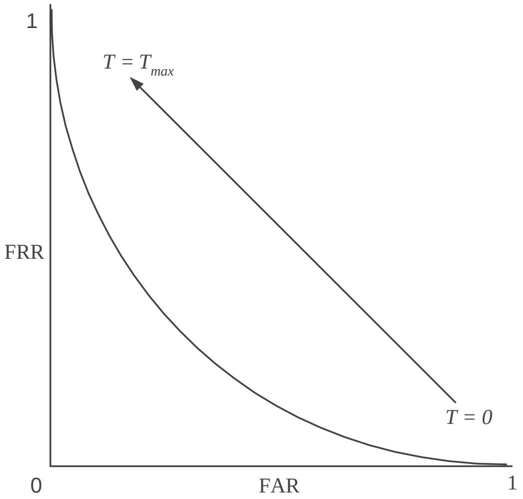

Fig. 4. An ROC curve is the relation between the FRR and FAR as a function of decision threshold $T$.

An example of an ROC curve is shown in Fig. 4. One can improve one of the error rates only at the expense of the other, i.e., any effort to lower one of the errors automatically increases the other error rate. Depending on the application, the system's operating point can be shifted toward a low FAR or a low FRR; the equal error point $T_{EER}$ is seldom used. Typical error rates for a fingerprint system are in the range of $10^{-6}$ for *false accept* and $10^{-4}$ for *false reject* [4].

There is, however, yet another system performance issue known as the "fail to enroll" rate (see Ref. [8]). This is the percentage of subjects that simply cannot be enrolled because of poor biometrics signals, or signals that are too hard (noisy) to match. Obviously, if such individuals can be detected and excluded from using the system by some sort of exception handling, both FRR and FAR can be much improved.

### 3.2. Brute-force attacks

Both biometrics- and password-based systems can be attacked by brute-force. The difficulty by which passwords can be attacked is relatively easy to analyze. Here we analyze

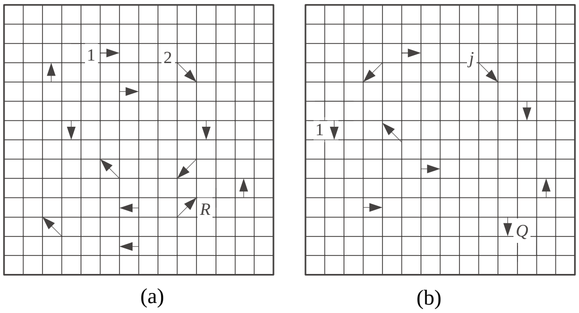

Fig. 5. (a) Enrolled template with $R$ minutiae. (b) Generated query template with $Q$ minutiae. The enrolled minutiae constellation on the left is attacked by randomly generating minutiae constellations on the right.

the vulnerability of fingerprint-based systems to brute-force attacks. This is achieved by estimating the probability of success of a brute-force attack on a fingerprint template.

The idea is to estimate the likelihood of randomly generating a minutiae constellation that matches the constellation found in the fingerprint of an enrolled person. The enrolled, or reference, fingerprint has $R$ minutiae. The print with $R$ reference minutiae is shown in Fig. 5a. Each minutiae has $d$ possible directions and one of $K$ possible locations. We randomly generate a set of $Q$ query minutiae, as in Fig. 5b, and estimate the probability of a match.

The probability that a first randomly generated minutiae will match one of the minutiae in the reference print in both location and direction equals

$$p_1 = \frac{R}{Kd}. \tag{2}$$

In a comprehensive model one would really need to model the probability distribution of minutiae locations relative to the center of the print. In addition, the directional proclivities based on position (they tend to swirl around the core) would also need to be modelled. In this model, however, we will ignore such statistical correlation between minutiae and use the simpler formulation of a uniform distribution of minutiae over the $K$ sites and the $d$ directions.

While $p_1$ of Eq. (2) is valid for the first generated minutiae, when creating the full synthetic set it is undesirable to generate two minutiae with the same location. So after $j-1$ minutiae have been generated, the probability $p_j$ that the $j$th minutiae will match is smaller than the upper bound:

$$p_j \leqslant \frac{R}{(K-j+1)d}. \tag{3}$$

This assumes that the previous $j-1$ all fail. If some of the previous $j-1$ minutiae have matched, the bound in Eq. (3) will be smaller. So, to be conservative, while generating $Q$ random minutiae we can assume each of the minutiae has matching probability

$$p_Q = \frac{R}{(K-Q+1)d}, \tag{4}$$

which is obtained by substituting $Q$ for $j$ in expression (3).

Typical parameters values are $K=400$, $R=Q=50$ and $d=4$. Note that brute-force attacks with $Q$ excessively large (say, as large as $K$) would be easy to detect and reject out of hand. For this reason there is an upper bound on the $Q$ that can be used and still give the facsimile of a real finger. Using the values above we find $p_1 = 0.03125$ while $p_Q = 0.03561$ (14% higher). This is a relatively small effect in itself, but important in the overall calculation.

Therefore, using $p_Q$ from Eq. (4), the probability that exactly $t$ of the $Q$ generated minutiae match is about

$$p_Q^t(1-p_Q)^{Q-t}. \tag{5}$$

This breaks down for small $K$ because the minutiae matching probability changes depending on how many other minutiae have already been generated, as well as on how many of those minutiae have matched. However, for the large $K$'s typically encountered (e.g., 400) it is reasonably close.

Now there are typically a large number of ways of selecting which $t$ out of the $R$ minutiae in the reference print are the ones that match. Taking this and Eq. (5) into account, the total match probability $P$ is

$$P = \binom{R}{t} p_Q^t(1-p_Q)^{Q-t}. \tag{6}$$

Matches of *m or more* minutiae typically count as authentication. If, for convenience, we assume $R=Q=N$ (and, therefore, $p=p_Q$), we get

$$\begin{aligned} P_{match} &= \sum_{t=m}^{N} \binom{R}{t} p_Q^t(1-p_Q)^{Q-t} \\ &= \sum_{t=m}^{N} \binom{N}{t} p^t(1-p)^{N-t}. \end{aligned} \tag{7}$$

Since $p$ is fairly small in our case, we can use the Poisson approximation to the binomial probability density function in Eq. (7). We also note that the summation for the total probability is usually dominated by its first term (where $t=m$). For typical parameter values, the second term is 10 –20 times smaller than the first. While neglecting all but the first term may make the overall estimate approximately 20% lower, for order of magnitude calculations this is fine. Using this, we can then rewrite the expression as simply

$$P_{match} \simeq \sum_{t=m}^{N} \frac{(Np)^t \mathrm{e}^{-Np}}{t!} \simeq \frac{(Np)^m \mathrm{e}^{-Np}}{m!}. \tag{8}$$

Finally, because $m$ is moderately large, we can use Stirling's approximation for the factorial and further rewrite the equation as

$$P_{match} = \frac{(Np)^m \mathrm{e}^{-Np}}{\sqrt{(2\pi m)}\mathrm{e}^{-m}m^m} = \frac{\mathrm{e}^{-Np}}{\sqrt{2\pi m}} \left(\frac{\mathrm{e}Np}{m}\right)^m, \tag{9}$$

where the second expression has been regrouped to emphasize the exponential dependency.

Let $N = 40$, $d = 4$, and $K = 400$. For a value of $m = 10$ (the number of minutiae required to match), we have about 22 bits of information. For the legal threshold of $m = 15$, we have around 40 bits of information (about 140 times the population of the Earth). For a more typical value of $m=25$, we have roughly 82 bits of information content in this representation. This is equivalent to a nonsense password which is 16 characters long (like "m4yus78xpmks3bc9") assuming 5 bits per character.

Studies have been reported in the literature to evaluate the individuality of a fingerprint [9,10] based on the minutiae information. These analyses are based on minutiae frequency data as characterized by a human expert on a small set of fingers. They also used all ten types of Galton characteristics, whereas our study is based on just one type of feature—a minutiae (with no differentiation between ridge endings and bifurcations). In general, the point of these studies was to quantify the information content of a fingerprint rather than set thresholds for matching in the face of brute-force attacks.

Examining the final equation we make two other important observations. First, it can be seen that adding extra feature information at every minutiae (i.e., raising $d$) increases the strength of the system significantly. Similarly, if the spatial domain extent is increased, or the location tolerance decreased (i.e., raising $K$), the strength also increases. Both these factors directly affect $p$ from Eq. (4), the single minutiae matching probability, which shows up inside the exponential term of $P_{match}$ of Eq. (9).

Second, there is a strong dependence on $N$, the overall number of minutiae in a fingerprint. For the best security, this number needs to be kept as low as possible. This is one reason why the probability of break-ins is much smaller when good-quality fingers are enrolled as opposed to poor-quality images which often have many spurious minutiae (yielding a higher overall $N$). Practical systems usually reject a bad-quality fingerprint image for this reason, instead of taking a hit on the accuracy and security of the system.

We should explain that the brute-force break-in is not dependent in any way on the FAR. That is, if the FAR is $10^{-6}$ this does not mean that, on average, the system is broken into after 500,000 trials. The FAR is based on actual human fingers and is typically attributable to errors in feature extraction (extra or missing features). To a lesser extent, it is also due to changes in geometry such as finger rolling or skin deformations due to twisting. The statistics governing the occurrence of these types of errors are different from those describing a brute-force attack.

### 3.3. Matching score computation

The above brute-force analysis accepts two minutiae sets as matching if $t$ individual minutiae pairs can be matched simultaneously, i.e., a match is declared if $m \geqslant t$ (see Eq. (5)). This is based on the fact that two prints are considered the same with $m$ between 12 and 15 in courts of law in many countries (with the correspondence performed by human experts).

Fingerprint match scores are often normalized with respect to the number of minutiae in the reference template $R$ and/or the number of minutiae in the query template $Q$. Including the straight minutiae count $m$ as match criterion, fingerprint match scores $s$ can be chosen to be proportional to (say)

$$m, \tag{10a}$$

$$\frac{m}{Q}, \tag{10b}$$

$$\frac{m}{R} \tag{10c}$$

or

$$\frac{m}{(R+Q)}. \tag{10d}$$

Let us denote the match score function between two templates $T_a$ and $T_b$ by $s(T_a, T_b)$, and denote the reference and query template by $T_R$ and $T_Q$, respectively. It is seen then that $s(T_R, T_Q) = s(T_Q, T_R)$ for Eq. (10d), while for all other cases in (Eqs. 10a–10c) $(s(T_R, T_Q) \neq s(T_Q, T_R)$.

This brings us to the classification of enrollees in terms of zoo animals, as introduced by Doddington et al. in Ref. [11] (the "Doddington's zoo"). This classification was developed in the field of voice recognition for speaker identification, but is applicable to other biometrics as well. The categories are:

*Sheep*: This is the group of subjects that dominate the population and for which authentication systems perform reasonably well.

*Goats*: The group of subjects that are particularly difficult to authenticate. This portion of the population generates the majority of false rejects.

*Lambs*: These are subjects who are easy to imitate. A randomly selected individual from the general population is highly likely to be authenticated (erroneously) as one of the lambs. Thus lambs contribute primarily to false accepts.

*Wolves*: Subjects who are particulary good at imitating other subjects. That is, their biometrics are likely to be accepted as those of another subject. Such wolves make successful intruders and also contribute to false accepts.

A possible way to distinguish the two types of false accepts is to think of lambs causing "passive false accepts" and wolves causing "active false accepts". That is, for a closed world, i.e., all subjects are enrolled, both types of false accepts are fixed numbers. However, actively attacking such a system with unenrolled wolves will result in a significantly higher FAR because of wolves.

The match score computation method directly determines the Doddington populations. Let us first look at the straight minutiae count $m$ as match criterion Eq. (10a). This criterion may be fine when a human is the ultimate decision maker, but it is surely not for automated decision making. This is because an enrolled print with $R$ (the number of minutiae in the stored reference print) large has much greater chance of matching. Hence, enrolled prints with a large number of minutiae $R$ will be lambs. Enrolled prints with few minutiae $R$ will be goats, though, in general it will be true that such prints are goats for any matching criterion.

For the case of Eq. (10b), the match score is high when the number of minutiae in the query print is low. Hence, query prints with few minutiae are wolves. On the other hand, for the case of Eq. (10c), we have that the match score is high when the number of minutiae in the reference print is low. Hence, enrolled prints with *few minutiae* are lambs, in contrast to Eq. (10a).

The case of Eq. (10d) results in a symmetric matching function, $s(T_R, T_Q) = s(T_Q, T_R)$. Curiously, in this case there is no difference between lambs and wolves, i.e., *lambs* $\equiv$ *wolves*. Therefore, we propose to add another animal, the *chameleon*, to the class of animals. Chameleons only exist if the matcher is symmetric.

*Chameleons*: These are the subjects who are both easy to imitate and are good at imitating others. They are a source of passive false accepts when enrolled, and a source of active false accepts when being authenticated.

Typically an authentication system has more confidence in the $R$ enrolled minutiae in template $T_R$. Thus it makes sense to design the matcher to be asymmetric. Increasing the reliability of the reference template $T_R$ is a sound thing to do because (i) the reference template is used many more times than the query template and (ii) at enrollment time, one can afford to spend more computation to ensure reliable minutiae. From the Fingerprint Verification Contest 2000 [8], it can be observed that some commercial matchers implement this policy since the authors note that fingerprint image processing for enrollment takes more computation time. When more computational effort is expended on determining $T_R$, a match score can be selected to be proportional to

$$\frac{m}{\alpha R + \beta Q}, \tag{11}$$

with $\alpha > \beta$. A score computation as Eq. (11) is a generalization of Eqs. (10b)–(10d).

The above observations indicate that the design of a match function has direct repercussions on the four Doddington groups of subjects. Designing a match score function is non-trivial and it may be beneficial to account explicitly for the four Doddington populations during design.

## 4. Security and privacy enhancements

One of the drawbacks of biometric authentication systems is their vulnerability to replay attacks. That is, the authentication system may be fraudulently supplied with prerecorded biometrics signals or templates in an attempt to gain access. Another drawback is that biometric systems invade the user's privacy to some extent, in that biometrics signals that are more or less unique to a user are required upon enrollment. Once acquired, it is possible that the biometrics signals might be used for other purposes, unknown to the user. This section proposes some solutions to these two problems.

### 4.1. Biometrics signal validation

Besides the simple interception of network traffic, more insidious attacks might be perpetrated against an automated biometrics authentication system. One of these is a replay attack directed at the input of the remote station, the "client attack", (i) in Fig. 2. Another, more serious threat is that an imposter poses as a real user to the server in a "server attack", (ii) in Fig. 2. In this type of attack, it is assumed that the perpetrator has somehow obtained a valid biometrics signal or template. This is a possibility since either the biometrics signal or template must be transmitted from the client to the server (in compressed and encrypted form, of course).

We propose a new method to thwart such attempts based on a modified challenge/response system. Conventional challenge/response systems are based on challenges $C_u$ to the user, such as requesting their mother's maiden name, or challenges to a physical device $C_d$, like a special-purpose calculator that computes a numerical response. That is,

$$\text{the challenge is } C; \quad \text{the response is } f(C), \tag{12}$$

where the function $f$ is only known to the user or the physical device. Note, however, that for a biometrics authentication system a challenge as in Eq. (12) only proves the responder (the user or physical device) knows the function $f$. It does not ensure that the biometrics signal itself is fresh.

Our approach is based on challenges $C_s$ to the sensor instead. As shown in Fig. 6, the sensor itself is assumed to have enough intelligence to respond to the challenges. For example, many silicon fingerprint scanners [12] can exploit the proposed method since a processor can be integrated without much effort (as indicated by the integrated sensor and processor in the figure).

Note that standard cryptographic techniques are not a suitable substitute. While they are mathematically strong, they are also computationally intensive and require maintaining secret keys for a large number of sensors. Moreover, encryption techniques cannot check for liveliness of a signal. An old stored signal can be given to the encrypter and it will simply encrypt it. Similarly, a digital checksum of a signal only ensures its integrity, not its liveliness.

Our system differs in that it computes a response string which depends not only on the challenge string, but also on the content of the returned template or signal. The changing challenges ensure that the signal was acquired, or the template was derived, after the challenge was issued. The dependence on signal or template values meanwhile guards

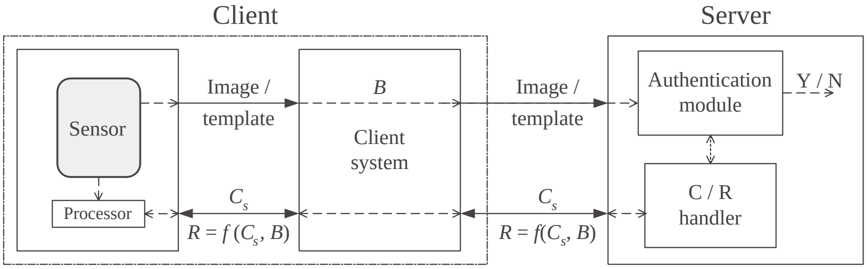

Fig. 6. The client–server model with challenge/response biometrics signal validation.

against substitution of data after the response has been generated.

Our proposed solution works as shown in Fig. 6. When a transaction is initiated at the user terminal or system, the server generates a pseudo-random challenge $C_s$ for the transaction. Note that we assume that the transaction server itself is secure. The client system then receives this challenge and passes it on to the intelligent sensor. Now, the sensor first acquires a new signal, and then computes a response to the challenge based in part on the newly acquired signal or derived template $B$. That is, unlike the conventional challenge/response system of Eq. (12), we have

$$\text{the challenge to the sensor is } C_s; \quad \text{the response is } f(C_s, B). \tag{13}$$

Because the response processor is tightly integrated with the sensor (preferably on the same chip), the signal channel into the response processor *is assumed ironclad and inviolable.* That is, it is just about impossible to inject a fake signal or template under such circumstances.

As an example of an image-based response, consider the function "$x1+$" which operates by appending pixel values of the fingerprint image (in scan order) to the end of the challenge string. A typical challenge might be "3, 10, 50". In response to this, the integrated processor then selects the 3rd, 10th, and 50th pixel value to generate an output response composed of gray levels such as "133, 92, 176". The complete image as well as the response is then transmitted to the server where the response can be verified and checked against the image.

Other examples of responder functions include computing a checksum of a segment of the signal, a set of pseudo-random samples, a hash of signal values, a block of contiguous samples starting at a specified location with a given size, or a specified known function applied to selected samples of the signal. Of course, a combination of these functions can be used to achieve arbitrarily complex responder functions. The important point is that the response depends on the challenge *and* the image itself.

Another approach based on data hiding is described in Ref. [13]. Here data are hidden in the WSQ-compressed [14] fingerprint images, that is, the least significant bits of quantized WSQ coefficients are altered by the bits of the embedded message. This is one way to transmit the response to a challenge so that an eavesdropper might not even detect that a challenge/response protocol was being used.

### 4.2. Cancellable biometrics

Deploying biometrics in a mass market, like credit card authorization or bank ATM access, raises additional concerns beyond the security of the transactions. One is the public's perception of invasion of privacy. In addition to private information such as name and date of birth, the user is asked to give images of their body parts, such as fingers, face, and iris. These images, or other biometrics signals, will be stored in digital form in a database somewhere. A concern is the possible sharing of such a database of biometrics signals with law enforcement agencies, or sharing of these databases among commercial organizations. These privacy concerns can be summarized as follows:

1. Much data about customers and customer behavior are already being stored. The public is increasingly concerned about every bit of additional information that is known about them.
2. The public is, in general, suspicious of central storage of information that is associated with individuals. This type of data ranges from medical records to biometrics. These databases can be used and misused for all sorts of purposes, and the databases can be shared among organizations.
3. The public is, rightfully or wrongfully so, worried about giving out biometrics because these could be used for matching against databases used by law enforcement agencies. Their biometrics might, for example, be matched against the FBI or INS fingerprint databases to obtain criminal records.

Hence, coupling biometrics with other personal parametric and demographic data is a concern, as is the potential use of stored biometrics for searching other databases not directly relevant to the transaction at hand.

These concerns are aggravated by the fact that a biometrics cannot be changed. One of the properties that make biometrics so attractive for authentication purposes, their invariance over time, is also one of their largest liabilities. When a credit card number is somehow compromised, the issuing bank can just assign the customer a new credit card number. When a biometrics is compromised, however, a new one cannot be issued—a person only has so many fingers, and only one face.

As an answer to these issues, we propose a novel concept called "cancellable biometrics". This is an intentional, repeatable distortion of a biometrics signal based on a chosen transform. The biometrics signal is distorted in the same fashion at each presentation, that is, during enrollment and for every subsequent authentication. With this approach, every instance of enrollment can use a different transform thus rendering cross-matching impossible. Furthermore, if one variant of the biometric is compromised, then the transformation can simply be changed to create a new variant (transformed representation) for re-enrollment as, essentially, a new person. In general, the distortion transforms are selected to be non-invertible. So even if the exact transform is known and the resulting transformed biometrics is known, the original (undistorted) biometrics cannot be recovered.

Cancellable transforms can be applied in the biometrics signal domain or in the domain of features that are used to represent the biometrics. That is, the biometrics signal can be transformed directly after acquisition, or the signal can be processed as usual and the extracted features can be transformed. We describe two particular transforms in the following sections, though many other types of transforms are of course possible.

### 4.2.1. Signal domain distortions

This category refers to the distortion (preferably non-invertible) of the raw biometric signal as it is acquired by the sensor, i.e., the original voice print or fingerprint impression.

For face or fingerprint images, a morphed version of the image can be enrolled. Such morphing can be achieved in various ways. For example, a regular point pattern can be overlaid on the image. The morphed image is then obtained by randomly perturbing this point pattern in a structured fashion. Note that a subject can be enrolled in a legacy fingerprint or face authentication systems with such a morphed image. The legacy authentication system does not have to be aware of the fact that the image is morphed. Moreover, matching of this morphed image to any other existing fingerprint or face database will not identify the identity of the owner of the fingerprint or face. An example of transformed facial images is shown in Fig. 7. Other examples of

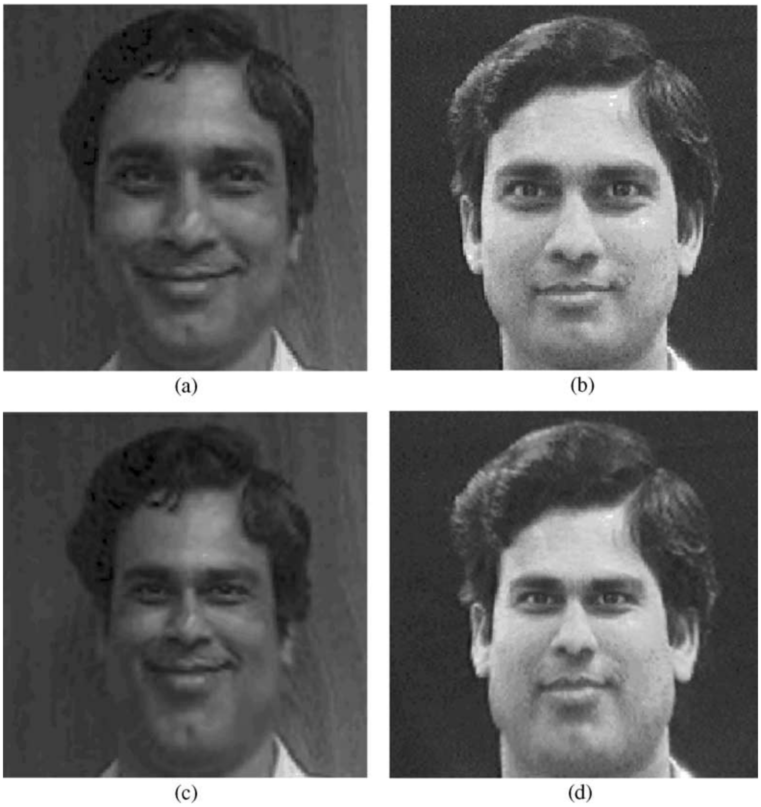

Fig. 7. Two images of an individual (a,b). The same distortion applied to the original images gives (c,d). Note that (c) and (d) appear similar, yet do not match (a) and (b).

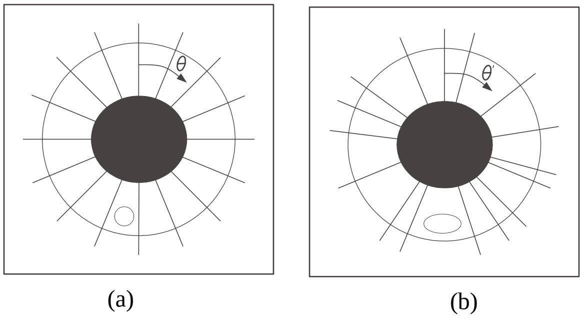

Fig. 8. (a) The original iris image. (b) The scrambled iris image. The original iris image can be scrambled through a variety of methods.

transformed images can be found in Refs. [13,15]. Note that in order to apply the same image morphing for each authentication, the fingerprint or face image needs to be transformed into a canonical position before the distortion. This can be done by aligning intrinsic points in the image, such as the intra-eye segment in a face or the core and delta in a fingerprint.

The iris, the colored area around the pupil of the eye, is another biometrics that is believed to be unique to a person. Daugman [16] popularized the use of iris codes for user authentication and identification. This biometrics is derived from an image of the user's iris as depicted in Fig. 8a. User authentication and identification through the iris image is achieved by developing a binary code $c=$ "0100101110....011" from a processed image of the iris. Identification can then be achieved very quickly even on large databases of enrolled users, since comparing these codes is very simple (Hamming distance).

Again there is the problem that if a person's iris code is compromised, it is compromised forever. Thus, it is desirable to have a cancellable version of iris ID. Fig. 8 shows an example where the iris image on the right is intentionally distorted to yield the version on the left. In this case, $\theta' = f(\theta)$, where $f(\theta)$ is a piecewise linear function of $\theta$. That is, $f(\theta) = \alpha_1\theta$ for $0 \leqslant \theta \leqslant \theta_1$, $f(\theta) = A_2 + \alpha_2\theta$ for $\theta_1 \leqslant \theta \leqslant \theta_2$, up to $f(\theta) = A_n + \alpha_n\theta$ for $\theta_{n-1} \leqslant \theta \leqslant \theta_n = 2\pi$. An annular segment is compressed if $\alpha < 1$, or correspondingly expanded if $\alpha > 1$. Many other variations are possible, for instance, $f(\theta)$ could be a monotonically increasing function of $\theta$ with varying first derivative.

A similar morphing technique can be applied to signals not usually considered as images. Fig. 9 shows a two-dimensional original voice biometrics signal $D(f,t)$. At each given time $t_o$, $D(f,t_o)$ gives the frequency content of the voice signal at that time point, like a spectrogram. The voice print in Fig. 9a is divided into time segments $A$, $B$, $C$, and $D$ arranged in a time sequence $(A,B,C,D)$. In this figure, the time segments are of equal length but that is not a requirement. A scrambled voice signal is constructed as in Fig. 9b, as the sequence $(\underline{A},C,\underline{D},B)$. Here the underscore notation $\underline{A}$ means that the time segment $A$ is played in reverse. In general, any time segment $X$ can be played as a monotonically increasing function of time starting at the begin point of $X$; any time segment $\underline{Y}$ can be played as a monotonically decreasing function of time starting at the end point of $Y$.

Note that for voice scrambling as in Fig. 9, only minimal registration of the query voice print with the enrolled voice print is needed, such as aligning the onset time.

#### 4.2.2. Feature domain distortions

Processed biometrics signals can also be intentionally distorted. Here, we present an example of a non-invertible distortion of a point pattern. Such a point pattern could, for example, be a set of fingerprint minutiae

$$S = \{(x_i, y_i, \theta_i),\ i = 1,\ldots,M\}.$$

However, point sets could also represent other biometrics, say, for example, the quantized frequencies and amplitudes of a speech pattern. A non-invertible transform maps such a set $S$ into a new set $S'$ so that the original set $S$ cannot be recovered from $S'$, i.e.,

$$\begin{aligned} S &= \{(x_i, y_i, \theta_i),\ i = 1,\ldots,M\} \\ &\mapsto S' = \{(X_i, Y_i, \Theta_i),\ i = 1,\ldots,M\}. \end{aligned} \tag{14}$$

Fig. 10 shows how the $x$ coordinates of the point set $S$ can be transformed through a mapping $x \mapsto X$, or $X = F(x)$. An example of a suitable function of $x$ might be a high-order polynomial

$$X = F(x) = \sum_{n=0}^{N} a_n x^n = \prod_{n=0}^{N} (x - b_n). \tag{15}$$

Assume for now that the polynomial of Eq. (15) is third order. A requirement then is that at least the second root $b_1$ (with the roots ordered according to $x$ value) lies in the range of the $x$ values of the original point set. A second requirement is that the smallest and largest root do not lie too far from $b_1$. If these conditions are met, as can be seen from Fig. 10, the reverse mapping $X \mapsto x$ is one-to-$K$ (where $K$ in our example is 2 for most $X$ if $b_0$ and $b_2$ are well chosen).

More generally, a one-to-one mapping $X = F(x)$, with $X \mapsto x$ a one-to-many mapping can be constructed as follows:

- Let the range for $x$ be $[x_{min}, x_{max}]$ and choose a range for $X$ as $[X_{min}, X_{max}]$.
- Set $X_{min} = F(x_{min})$ and $X_{max} = F(x_{max})$.
- Select an odd number $k$ of (roughly equally spaced) roots $b_i \in [x_{min}, x_{max}]$.
- Set the first derivatives of $F(x)$ at $x_{min}$ and $F(x)$ at $x_{max}$ to different values but close to $(X_{max} - X_{min})/k$.

Similar polynomial non-invertible transforms

$$Y = G(y) \quad \text{and} \quad \Theta = H(\theta)$$

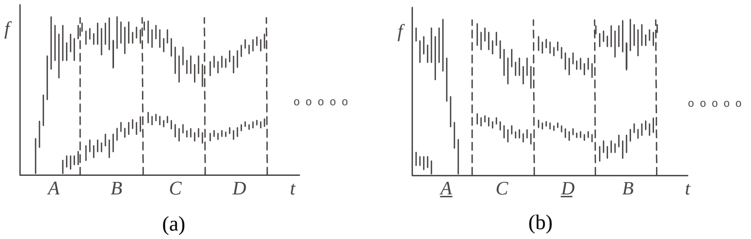

Fig. 9. (a) The original voice print. (b) The scrambled voice print. The original voice print is scrambled by dividing the signal up into segments. The scrambled voice print is constructed by randomly reordering and reversing the segments.

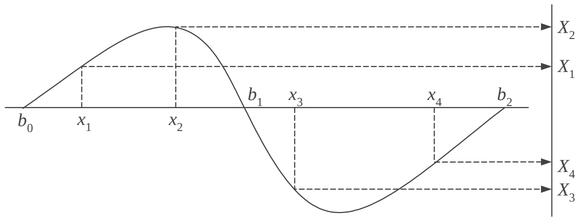

Fig. 10. The mapping of one of the coordinates of a higher-dimensional point set into a new set of coordinates. Here a third-order polynomial is used that has its roots in the domain of the $x$ values.

can be used for the remaining coordinates of the point set of Eq. (14). For other types of feature distortions, see Ref. [13].

### 4.2.3. Compression and encryption

Cancellable biometrics differ significantly from compression of the signal using standard image processing techniques. In signal compression, the signal temporarily loses its spatial domain characteristics such as geometric proximity. That is, two points in the original uncompressed signal are unlikely to remain at a comparable distance in the compressed domain. However, after decoding the original signal is either perfectly restored, or approximately restored if the compression is lossy. In cancellable biometrics, by contrast, much of the local geometry is retained.

Cancellable biometrics are also quite different from encryption technologies. In encryption, the desire is to recover the original signal at the other end of the secure transmission. Whereas for the non-invertible distortions we use, the original signal is not restored and, in fact, it should be (close to) impossible to do this.

Furthermore, existing biometrics systems cannot directly authenticate compressed or encrypted signals, whereas our transformed signals are intended to be processed by existing legacy software as if they were normal signals.

## 5. Conclusions

Currently, the weakest link in secure authentication system design is user authentication. As biometrics-based authentication becomes an integral part of overall security, biometrics systems have to be designed to be more robust to attacks from intruders. We have highlighted some of the weak points in a generic biometrics authentication system, analyzed the probability of a break-in using brute-force, and suggested some techniques to thwart replay attacks. We have also discussed design considerations that should be kept in mind when defining matching functions that determine how “close” two biometrics signals are.

In addition, we have addressed some of the privacy protection issues of a biometrics signal by introducing the new technique of intentional, non-invertible distortions that we call *cancellable biometrics*. This approach stymies cross-matching of biometric signals that were acquired from the same individual but for access to different services/applications. It further allows for cancellation of the original biometrics signal with which the user is enrolled and the assignment of a “new” biometrics signal, much as a new credit card number can be issued if the old one is somehow compromised.

## References

[1] A.K. Jain, L. Hong, S. Pankanti, Biometrics identification, Commun. ACM 43 (2) (2000) 91–98.

[2] B. Miller, Vital signs of identity, IEEE Spectrum 31 (2) (1994) 22–30.

[3] N.K. Ratha, K. Karu, S. Chen, A.K. Jain, A real-time matching system for large fingerprint database, IEEE Trans. Pattern Anal. Mach. Intell. 18 (8) (1996) 799–813.

[4] A.K. Jain, L. Hong, S. Pankanti, R.M. Bolle, An identity authentication system using fingerprints, Proc. IEEE 85 (9) (1997) 1365–1388.

[5] B. Schneier, Security pitfalls in cryptography, Proceedings of CardTech/SecurTech, Washington, DC, Vol. 1, 1998, pp. 621–626.

[6] B. Schneier, The uses and abuses of biometrics, Commun. ACM 42 (8) (1999) 136.

[7] C. Dorai, N. Ratha, R.M. Bolle, Detecting dynamic behavior in compressed fingerprint videos: distortion, Proceedings of IEEE Computer Vision and Pattern Recognition, Hiltonhead, FL, June 2000, pp. 320–326.

[8] D. Maio, D. Maltoni, R. Capelli, J.L. Wayman, A.K. Jain, FVC2000: fingerprint verification competition, Technical Report, Univ. of Bologna, September 2000.

[9] J.W. Osterberg, T. Parthasarathy, T.E.S. Raghavan, S.L. Sclove, Development of a mathematical formula for the calculation of fingerprint probabilities based on individual characteristics, J. Am. Stat. Assoc. 72 (1977) 772–778.

[10] S.L. Sclove, The occurrence of fingerprint characteristics as a two dimensional process, J. Am. Stat. Assoc. 74 (1997) 588–595.

[11] G. Doddington, W. Liggett, A. Martin, M. Przybocki, D. Reynolds, Sheep, goats, lambs and wolves: a statistical analysis of speaker performance, Proceedings of IC-SLD'98, NIST 1998 Speaker Recognition Evaluation, Sydney, Australia, November 1998, pp. 1351–1354.

[12] T. Rowley, Silicon fingerprint readers: a solid state approach to biometrics, Proceedings of the CardTech/SecurTech, Orlando, FL, Vol. 1, May 1997, pp. 152–159.

[13] N.K. Ratha, J.H. Connell, R.M. Bolle, Enhancing security and privacy in biometrics-based authentication systems, IBM Syst. J. 40 (3) (2001) 614–634.

[14] WSQ gray-scale fingerprint image compression specification, Technical Report, U.S. Department of Justice, Federal Bureau of Investigation (FBI), Washington, DC, 1993.

[15] R.M. Bolle, N. Ratha, J.H. Connell, Biometric authentication: security and privacy, Proceedings of the First International Workshop on Pattern Recognition in Information Systems, PRIS 2001, Sebutal, Portugal, ICEIS Press, July 2001, pp. 2–11.

[16] J.G. Daugman, High confidence visual recognition of persons by a test of statistical independence, IEEE Trans. Pattern Anal. Mach. Intell. 15 (11) (1993) 1148–1161.

**About the Author**—RUUD M. BOLLE is the founding manager of the Exploratory Computer Vision Group, which is now part of the Human Languages Technologies department. He received his Ph.D. in Electrical Engineering from Brown University, Providence, RI in 1984. He is a Fellow of the IEEE and the IAPR and is a Member of the IBM Academy of Technology. His research interests are focussed on video database indexing, video processing, visual human–computer interaction, and biometric.

**About the Author**—JONATHAN CONNELL is a Research Staff Member in the Exploratory Computer Vision Group. He received his Ph.D. degree in 1989 at the MIT Artificial Intelligence Laboratory, working with Rod Brooks on behavior-based mobile robot control. His research interests include robotics, vision, natural language, and complete AI systems.

**About the Author**—NALINI K. RATHA is a Research Staff Member in the Exploratory Computer Vision Group. He received his Ph.D. Degree in Computer Science from Michigan State University in 1996, working in the Pattern Recognition and Image Processing Laboratory. His research interests include automated biometrics, computer vision, image processing, reconfigurable computing architectures, and performance evaluation.