

Review Article

Biometric Template Security

Anil K. Jain, Karthik Nandakumar, and Abhishek Nagar

*Department of Computer Science and Engineering, Michigan State University, 3115 Engineering Building,
East Lansing, MI 48824, USA*

Correspondence should be addressed to Karthik Nandakumar, nandakum@cse.msu.edu

Received 2 July 2007; Revised 28 September 2007; Accepted 4 December 2007

Recommended by Arun Ross

Biometric recognition offers a reliable solution to the problem of user authentication in identity management systems. With the widespread deployment of biometric systems in various applications, there are increasing concerns about the security and privacy of biometric technology. Public acceptance of biometrics technology will depend on the ability of system designers to demonstrate that these systems are robust, have low error rates, and are tamper proof. We present a high-level categorization of the various vulnerabilities of a biometric system and discuss countermeasures that have been proposed to address these vulnerabilities. In particular, we focus on biometric template security which is an important issue because, unlike passwords and tokens, compromised biometric templates cannot be revoked and reissued. Protecting the template is a challenging task due to intrauser variability in the acquired biometric traits. We present an overview of various biometric template protection schemes and discuss their advantages and limitations in terms of security, revocability, and impact on matching accuracy. A template protection scheme with provable security and acceptable recognition performance has thus far remained elusive. Development of such a scheme is crucial as biometric systems are beginning to proliferate into the core physical and information infrastructure of our society.

Copyright © 2008 Anil K. Jain et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

1. INTRODUCTION

A reliable identity management system is urgently needed in order to combat the epidemic growth in identity theft and to meet the increased security requirements in a variety of applications ranging from international border crossings to securing information in databases. Establishing the identity of a person is a critical task in any identity management system. Surrogate representations of identity such as passwords and ID cards are not sufficient for reliable identity determination because they can be easily misplaced, shared, or stolen. Biometric recognition is the science of establishing the identity of a person using his/her anatomical and behavioral traits. Commonly used biometric traits include fingerprint, face, iris, hand geometry, voice, palmprint, handwritten signatures, and gait (see Figure 1). Biometric traits have a number of desirable properties with respect to their use as an authentication token, namely, reliability, convenience, universality, and so forth. These characteristics have led to the widespread deployment of biometric authentication systems. But there are still some issues concerning the security of biometric recognition systems that need to be addressed in order to ensure the integrity and public acceptance of these systems.

There are five major components in a generic biometric authentication system, namely, sensor, feature extractor, template database, matcher, and decision module (see Figure 2). Sensor is the interface between the user and the authentication system and its function is to scan the biometric trait of the user. Feature extraction module processes the scanned biometric data to extract the salient information (feature set) that is useful in distinguishing between different users. In some cases, the feature extractor is preceded by a quality assessment module which determines whether the scanned biometric trait is of sufficient quality for further processing. During enrollment, the extracted feature set is stored in a database as a template (X_T) indexed by the user's identity information. Since the template database could be geographically distributed and contain millions of records (e.g., in a national identification system), maintaining its security is not a trivial task. The matcher module is usually an executable program, which accepts two biometric feature sets X_T and X_Q (from template and query, resp.) as inputs, and outputs a match score (S) indicating the similarity between the two sets. Finally, the decision module makes the identity decision and initiates a response to the query.

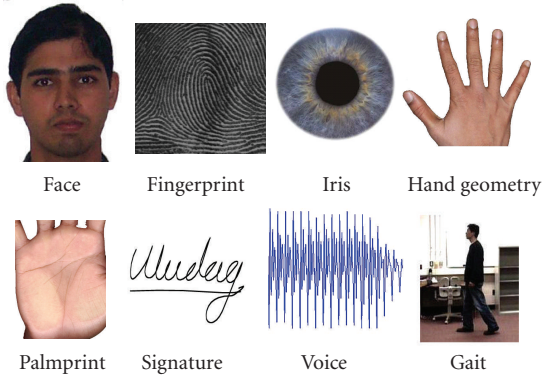


FIGURE 1: Examples of body traits that can be used for biometric recognition. Anatomical traits include face, fingerprint, iris, palmprint, hand geometry, and ear shape; while gait, signature, and keystroke dynamics are some of the behavioral characteristics. Voice can be considered either as an anatomical or as a behavioral characteristic.

Due to the rapid growth in sensing and computing technologies, biometric systems have become affordable and are easily embedded in a variety of consumer devices (e.g., mobile phones, key fobs, etc.), making this technology vulnerable to the malicious designs of terrorists and criminals. To avert any potential security crisis, vulnerabilities of the biometric system must be identified and addressed systematically. A number of studies have analyzed potential security breaches in a biometric system and proposed methods to counter those breaches [1–5]. Formal methods of vulnerability analysis such as attack trees [6] have also been used to study how biometric system security can be compromised.

In this paper, we first summarize the various aspects of biometric system security in a holistic and systematic manner using the fish-bone model [7]. Our goal here is to broadly categorize the various factors that cause biometric system failure and identify the effects of such failures. This paper is not necessarily complete in terms of all the security threats that have been identified, but it provides a high-level classification of the possible security threats. We believe that template security is one of the most crucial issues in designing a secure biometric system and it demands timely and rigorous attention. Towards this end, we present a detailed overview of different template protection approaches that have been proposed in the literature and provide example implementations of specific schemes on a public domain fingerprint database to illustrate the issues involved in securing biometric templates.

2. BIOMETRIC SYSTEM VULNERABILITY

A fish-bone model (see Figure 3) can be used to summarize the various causes of biometric system vulnerability [1]. At the highest level, the failure modes of a biometric system can be categorized into two classes: *intrinsic failure* and *failure due to an adversary attack*. Intrinsic failures occur due to inherent limitations in the sensing, feature extraction, or matching technologies as well as the limited discriminabil-

ity of the specific biometric trait. In adversary attacks, a resourceful hacker (or possibly an organized group) attempts to circumvent the biometric system for personal gains. We further classify the adversary attacks into three types based on factors that enable an adversary to compromise the system security. These factors include system administration, nonsecure infrastructure, and biometric overtress.

2.1. Intrinsic failure

Intrinsic failure is the security lapse due to an incorrect decision made by the biometric system. A biometric verification system can make two types of errors in decision making, namely, *false accept* and *false reject*. A genuine (legitimate) user may be falsely rejected by the biometric system due to the large differences in the user's stored template and query biometric feature sets (see Figure 4). These intrauser variations may be due to incorrect interaction by the user with the biometric system (e.g., changes in pose and expression in a face image) or due to the noise introduced at the sensor (e.g., residual prints left on a fingerprint sensor). False accepts are usually caused by lack of individuality or uniqueness in the biometric trait which can lead to large similarity between feature sets of different users (e.g., similarity in the face images of twins or siblings). Both intrauser variations and interuser similarity may also be caused by the use of nonsalient features and nonrobust matchers. Sometimes, a sensor may fail to acquire the biometric trait of a user due to limits of the sensing technology or adverse environmental conditions. For example, a fingerprint sensor may not be able to capture a good quality fingerprint of dry/wet fingers. This leads to failure-to-enroll (FTE) or failure-to-acquire (FTA) errors.

Intrinsic failures can occur even when there is no explicit effort by an adversary to circumvent the system. So this type of failure is also known as *zero-effort attack*. It poses a serious threat if the false accept and false reject probabilities are high (see Table 1). Ongoing research is directed at reducing the probability of intrinsic failure, mainly through the design of new sensors that can acquire the biometric traits of an individual in a more reliable, convenient, and secure manner, the development of invariant representation schemes and robust and efficient matching algorithms, and use of multibiometric systems [8].

2.2. Adversary attacks

Here, an adversary intentionally stages an attack on the biometric system whose success depends on the loopholes in the system design and the availability of adequate computational and other resources to the adversary. We categorize the adversary attacks into three main classes: administration attack, nonsecure infrastructure, and biometric overtress.

(i) Administration attack

This attack, also known as the insider attack, refers to all vulnerabilities introduced due to improper administration of the biometric system. These include the integrity of the

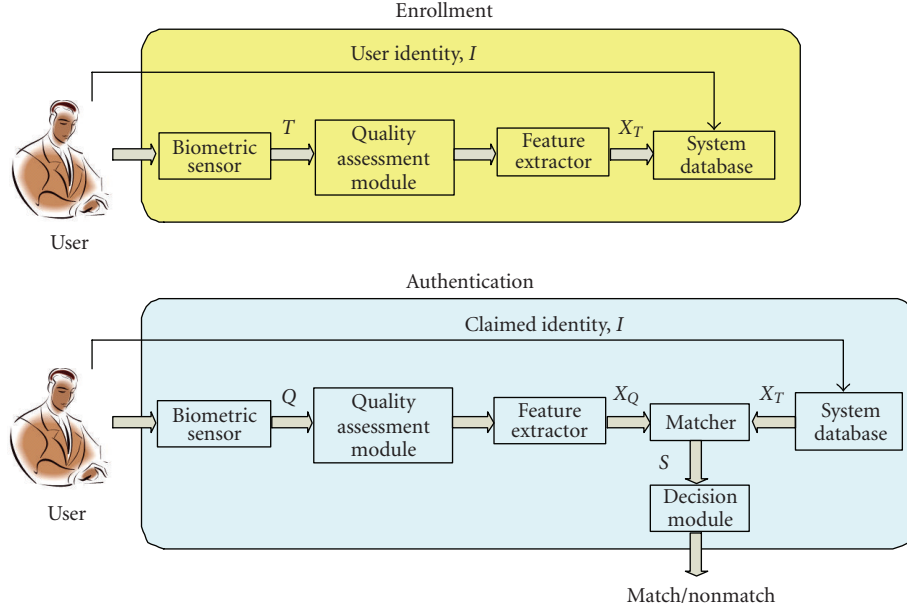


FIGURE 2: Enrollment and recognition stages in a biometric system. Here, T represents the biometric sample obtained during enrollment, Q is the query biometric sample obtained during recognition, X_T and X_Q are the template and query feature sets, respectively, and S represents the match score.

TABLE 1: False reject and false accept rates associated with state-of-the-art fingerprint, face, voice, and iris verification systems. Note that the accuracy estimates of biometric systems are dependent on a number of test conditions and target population.

Biometric trait	Test	Test conditions	False reject rate	False accept rate
Fingerprint	FVC 2006 [9]	Heterogeneous population including manual workers and elderly people	2.2%	2.2%
	FpVTE 2003 [10]	US government operational data	0.1%	1%
Face	FRVT 2006 [11]	Controlled illumination, high resolution	0.8%–1.6%	0.1%
Voice	NIST 2004 [12]	Text independent, multi-lingual	5–10%	2–5%
Iris	ICE 2006 [11]	Controlled illumination, broad-quality range	1.1%–1.4%	0.1%

enrollment process (e.g., validity of credentials presented during enrollment), collusion (or coercion) between the adversary and the system administrator or a legitimate user, and abuse of exception processing procedures.

(ii) Nonsecure infrastructure

The infrastructure of a biometric system consists of hardware, software, and the communication channels between the various modules. There are a number of ways in which an adversary can manipulate the biometric infrastructure that can lead to security breaches. A detailed discussion on these types of attacks is presented in Section 2.4.

(iii) Biometric overtress

It is possible for an adversary to covertly acquire the biometric characteristics of a genuine user (e.g., fingerprint impressions lifted from a surface) and use them to create physical artifacts (gummy fingers) of the biometric trait. Hence, if the biometric system is not capable of distinguishing between a

live biometric presentation and an artificial spoof, an adversary can circumvent the system by presenting spoofed traits.

2.3. Effects of biometric system failure

When a biometric system is compromised, it can lead to two main effects: (i) denial-of-service and (ii) intrusion.

Denial-of-service refers to the scenario where a legitimate user is prevented from obtaining the service that he is entitled to. An adversary can sabotage the infrastructure (e.g., physically damage a fingerprint sensor) thereby preventing users from accessing the system. Intrinsic failures like false reject, failure-to-capture, and failure-to-acquire also lead to denial-of-service. Administrative abuse such as modification of templates or the operating parameters (e.g., matching threshold) of the biometric system may also result in denial-of-service.

Intrusion refers to an impostor gaining illegitimate access to the system, resulting in loss of privacy (e.g., unauthorized access to personal information) and security threats (e.g., terrorists crossing borders). All the four factors that cause biometric system vulnerability, namely, intrinsic failure,

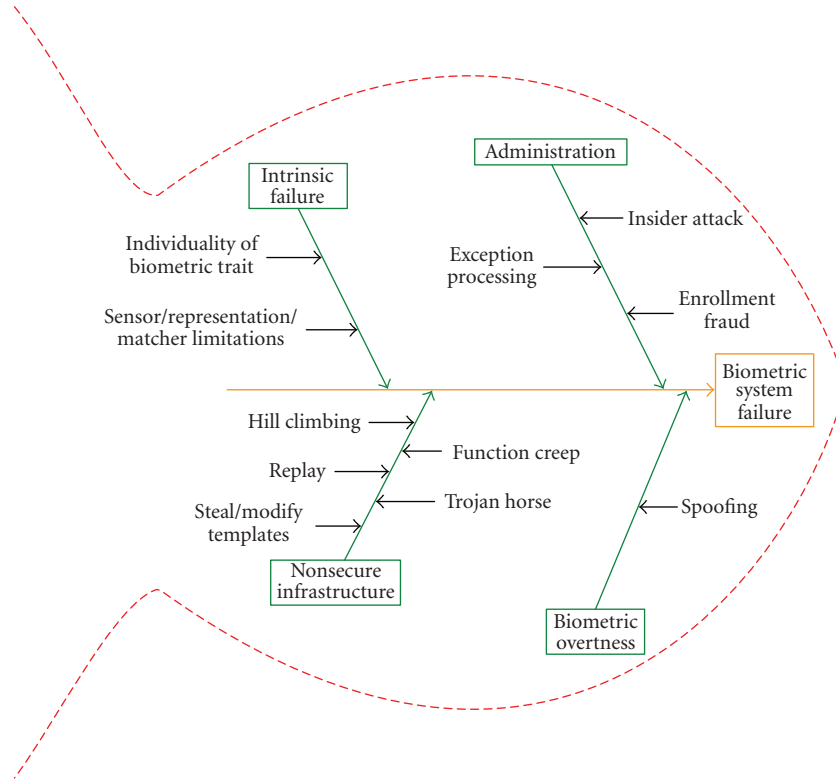


FIGURE 3: Fish-bone model for categorizing biometric system vulnerabilities (adapted from [1]).



FIGURE 4: Illustration of biometric intraclass variability. Two different impressions of the same finger obtained on different days are shown with minutiae points marked on them. Due to translation, rotation, and distortion, the number and location of minutiae in the two images are different. The number of minutiae in the left and right images is 33 and 26, respectively. The number of common minutiae in the two images is 16 and few of these correspondences have been indicated in the figure.

administrative abuse, nonsecure infrastructure, and biometric overtress, can result in intrusion.

2.4. Countering adversary attacks

Adversary attacks generally exploit the system vulnerabilities at one or more modules or interfaces. Ratha et al. [13] identified eight points of attack in a biometric system (see Figure 5). We group these attacks into four categories,

namely, (i) attacks at the user interface (input level), (ii) attacks at the interfaces between modules, (iii) attacks on the modules, and (iv) attacks on the template database.

2.5. Attacks at the user interface

Attack at user interface is mostly due to the presentation of a spoof biometric trait [14–17]. If the sensor is unable to distinguish between fake and genuine biometric traits, the adversary easily intrudes the system under a false identity. A number of efforts have been made in developing hardware as well as software solutions that are capable of performing liveness detection [18–26].

2.6. Attacks at the interface between modules

An adversary can either sabotage or intrude on the communication interfaces between different modules. For instance, he can place an interfering source near the communication channel (e.g., a jammer to obstruct a wireless interface). If the channel is not secured physically or cryptographically, an adversary may also intercept and/or modify the data being transferred. For example, Juels et al. [27] outlined the security and privacy issues introduced by insecure communication channels in an e-passport application that uses biometric authentication. Insecure communication channels also allow an adversary to launch replay [28] or hill-climbing attacks [29].

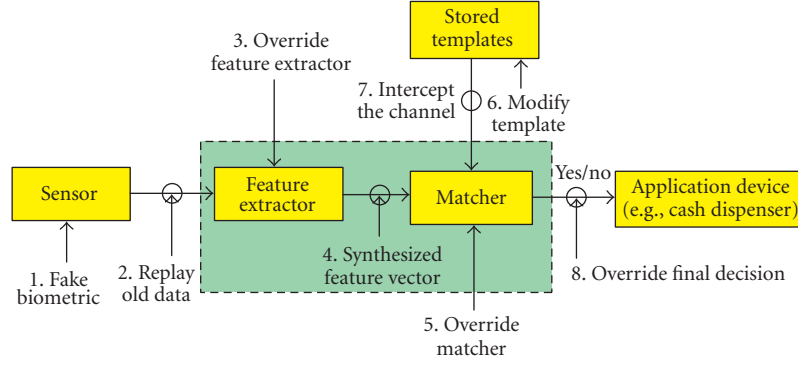


FIGURE 5: Points of attack in a generic biometric system (adapted from [13]).

A common way to secure a channel is by cryptographically encoding all the data sent through the interface, say using public key infrastructure. But even then, an adversary can stage a replay attack by first intercepting the encrypted data passing through the interface when a genuine user is interacting with the system and then sending this captured data to the desired module whenever he wants to break into the system. A countermeasure for this attack is to use time-stamps [30, 31] or a challenge/response mechanism [32].

2.7. Attacks on the software modules

The executable program at a module can be modified such that it always outputs the values desired by the adversary. Such attacks are known as Trojan-horse attacks. Secure code execution practices [33] or specialized hardware which can enforce secure execution of software should be used. Another component of software integrity relates to algorithmic integrity. Algorithmic integrity implies that the software should be able to handle any input in a desirable manner. As an example of algorithmic loophole, consider a matching module in which a specific input value, say X_0 , is not handled properly and whenever X_0 is input to the matcher, it always outputs a match (accept) decision. This vulnerability might not affect the normal functioning of the system because the probability of X_0 being generated from a real-biometric data may be negligible. However, an adversary can exploit this loophole to easily breach the security without being noticed.

2.8. Attacks on the template database

One of the most potentially damaging attack on a biometric system is against the biometric templates stored in the system database. Attacks on the template can lead to the following three vulnerabilities. (i) A template can be replaced by an impostor's template to gain unauthorized access. (ii) A physical spoof can be created from the template (see [34–36]) to gain unauthorized access to the system (as well as other systems which use the same biometric trait). (iii) The stolen template can be replayed to the matcher to gain unauthorized access. A potential abuse of biometric identifiers is cross-matching or *function creep* [37] where the biometric identifiers are used

for purposes other than the intended purpose. As an example, a fingerprint template stolen from a bank's database may be used to search a criminal fingerprint database or cross-link to person's health records.

The most straightforward way to secure the biometric system, including the template, is to put all the system modules and the interfaces between them on a smart card (or more generally a secure processor). In such systems, known as match-on-card or system-on-card technology, sensor, feature extractor, matcher, and template reside on the card [38]. The advantage of this technology is that the biometric information never leaves the card. However, system-on-card solutions are not appropriate for most large-scale applications; they are expensive and users must carry the card with them all the time. Further, it is possible that the template can be gleaned from a stolen card. So it is important to protect the template even in match-on-card applications. Passwords and PIN have the property that if they are compromised, the system administrator can issue a new one to the user. It is desirable to have the same property of revocability or cancelability with biometric templates. The following section provides a detailed description of the approaches that have been proposed for securing biometric templates.

3. TEMPLATE PROTECTION SCHEMES

An ideal biometric template protection scheme should possess the following four properties [39].

- (1) Diversity: the secure template must not allow cross-matching across databases, thereby ensuring the user's privacy.
- (2) Revocability: it should be straightforward to revoke a compromised template and reissue a new one based on the same biometric data.
- (3) Security: it must be computationally hard to obtain the original biometric template from the secure template. This property prevents an adversary from creating a physical spoof of the biometric trait from a stolen template.
- (4) Performance: the biometric template protection scheme should not degrade the recognition performance (FAR and FRR) of the biometric system.

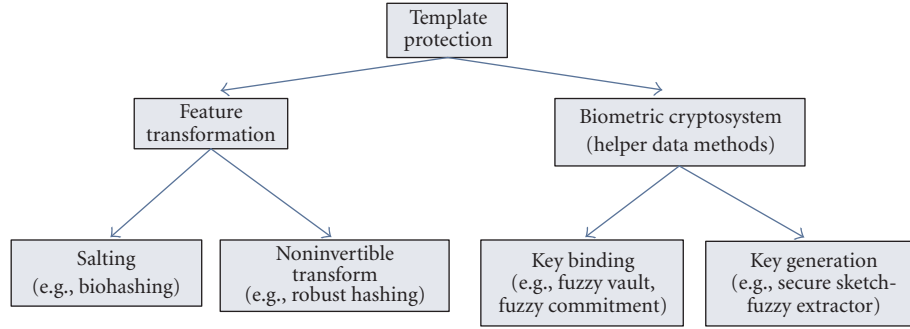


FIGURE 6: Categorization of template protection schemes.

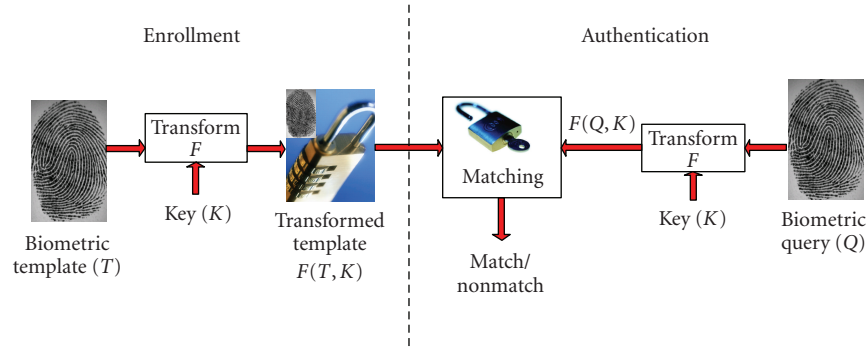


FIGURE 7: Authentication mechanism when the biometric template is protected using a feature transformation approach.

The major challenge in designing a biometric template protection scheme that satisfies all the above requirements is the need to handle intrauser variability in the acquired biometric identifiers. Recall that multiple acquisitions of the same biometric trait do not result in the same feature set (see Figure 4). Due to this reason, one cannot store a biometric template in an encrypted form (using standard encryption techniques like RSA, AES, etc.) and then perform matching in the encrypted domain. Note that encryption is not a smooth function and a small difference in the values of the feature sets extracted from the raw biometric data would lead to very large difference in the resulting encrypted features. While it is possible to decrypt the template and perform matching between the query and decrypted template, such an approach is not secure because it leaves the template exposed during every authentication attempt. Hence, standard encryption techniques are not useful for securing biometric templates.

The template protection schemes proposed in the literature can be broadly classified into two categories (see Figure 6), namely, *feature transformation* approach and *biometric cryptosystem*. In the feature transform approach, a transformation function (\mathcal{F}) is applied to the biometric template (T) and only the transformed template ($\mathcal{F}(T; K)$) is stored in the database (see Figure 7). The parameters of the transformation function are typically derived from a random key (K) or password. The same transformation function is applied to query features (Q) and the transformed query ($\mathcal{F}(Q; K)$) is directly matched against the transformed template ($\mathcal{F}(T; K)$). Depending on the characteristics of the

transformation function \mathcal{F} , the feature transform schemes can be further categorized as *salting* and *noninvertible transforms*. In salting, \mathcal{F} is invertible, that is, if an adversary gains access to the key and the transformed template, she can recover the original biometric template (or a close approximation of it). Hence, the security of the salting scheme is based on the secrecy of the key or password. On the other hand, noninvertible transformation schemes typically apply a one-way function on the template and it is computationally hard to invert a transformed template even if the key is known.

Biometric cryptosystems [40, 41] were originally developed for the purpose of either securing a cryptographic key using biometric features or directly generating a cryptographic key from biometric features. However, they can also be used as a template protection mechanism. In a biometric cryptosystem, some public information about the biometric template is stored. This public information is usually referred to as *helper data*, and hence biometric cryptosystems are also known as helper data-based methods [42]. While the helper data does not (is not supposed to) reveal any significant information about the original biometric template, it is needed during matching to extract a cryptographic key from the query biometric features. Matching is performed indirectly by verifying the validity of the extracted key (see Figure 8). Error correction coding techniques are typically used to handle intrauser variations.

Biometric cryptosystems can be further classified as *key-binding* and *key generation* systems depending on how the helper data is obtained. When the helper data is obtained by binding a key (that is independent of the biometric features)

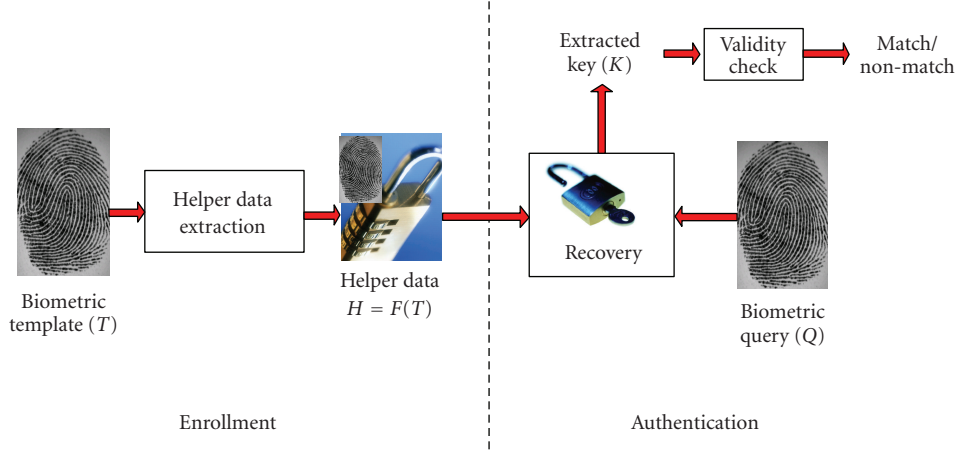


FIGURE 8: Authentication mechanism when the biometric template is secured using a key generation biometric cryptosystem. Authentication in a key-binding biometric cryptosystem is similar except that the helper data is a function of both the template and the key K , that is, $H = \mathcal{F}(T; K)$.

TABLE 2: Summary of different template protection schemes. Here, T represents the biometric template, Q represents the query, and K is the key used to protect the template. In salting and noninvertible feature transform, \mathcal{F} represents the transformation function, and \mathcal{M} represents the matcher that operates in the transformed domain. In biometric cryptosystems, \mathcal{F} is the helper data extraction scheme and \mathcal{M} is the error correction scheme that allows reconstruction of the key K .

Approach	What imparts security to the template?	What entities are stored?	How are intrauser variations handled?
Salting	Secrecy of key K	Public domain: transformed template $\mathcal{F}(T; K)$ Secret: Key K	Quantization and matching in transformed domain $\mathcal{M}(\mathcal{F}(T; K), \mathcal{F}(Q; K))$
Noninvertible transform	Noninvertibility of the transformation function \mathcal{F}	Public domain: transformed template $\mathcal{F}(T; K)$, key K	Matching in transformed domain $\mathcal{M}(\mathcal{F}(T; K), \mathcal{F}(Q; K))$
Key-binding biometric cryptosystem	Level of security depends on the amount of information revealed by the helper data H	Public domain: helper data $H = \mathcal{F}(T; K)$	Error correction and user specific quantization $K = \mathcal{M}(\mathcal{F}(T; K), Q)$
Key-generating biometric cryptosystem	Level of security depends on the amount of information revealed by the helper data H	Public domain: helper data $H = \mathcal{F}(T)$	Error correction and user specific quantization $K = \mathcal{M}(\mathcal{F}(T), Q)$

with the biometric template, we refer to it as a *key-binding biometric cryptosystem*. Note that given only the helper data, it is computationally hard to recover either the key or the original template. Matching in a key-binding system involves recovery of the key from the helper data using the query biometric features. If the helper data is derived only from the biometric template and the cryptographic key is directly generated from the helper data and the query biometric features, it leads to a *key generation biometric cryptosystem*.

Some template protection techniques make use of more than one basic approach (e.g., salting followed by key-binding). We refer to such techniques as *hybrid* schemes. Template protection schemes proposed in [43–46] are examples of the hybrid approach. A brief summary of the various template protection approaches is presented in Table 2. Apart from salting, none of the other template protection schemes requires any secret information (such as a key) that must be

securely stored or presented during matching. We will now discuss these four approaches in detail with one illustrative method for each approach.

3.1. Salting

Salting or *Biohashing* is a template protection approach in which the biometric features are transformed using a function defined by a user-specific key or password. Since the transformation is invertible to a large extent, the key needs to be securely stored or remembered by the user and presented during authentication. This need for additional information in the form of a key increases the entropy of the biometric template and hence makes it difficult for the adversary to guess the template. (Entropy of a biometric template can be understood as a measure of the number of different identities that are distinguishable by a biometric system.)

Advantages

- (1) Introduction of key results in low false accept rates.
- (2) Since the key is user-specific, multiple templates for the same user biometric can be generated by using different keys (allowing diversity). Also in case a template is compromised, it is easy to revoke the compromised template and replace it with a new one generated by using a different user-specific key (allowing revocability).

Limitations

- (1) If the user-specific key is compromised, the template is no longer secure, because the transformation is usually invertible, that is, if an adversary gains access to the key and the transformed template, she can recover the original biometric template.
- (2) Since matching takes place in the transformed domain, the salting mechanism needs to be designed in such a way that the recognition performance does not degrade, especially in the presence of large intrauser variations.

An example of salting approach is the random multi-space quantization technique proposed by Teoh et al. [47]. In this technique, the authors first extract the most discriminative projections of the face template using Fisher discriminant analysis [48] and then project the obtained vectors on a randomly selected set of orthogonal directions. This random projection defines the salting mechanism for the scheme. To account for intrauser variations, the feature vector obtained after random projection is binarized. The threshold for binarization is selected based on the criteria that the expected number of zeros in the template is equal to the expected number of ones so as to maximize the entropy of the template. Note that the security in this scheme is provided by the user-specific random projection matrix. If an adversary gains access to this matrix, she can obtain a coarse (some information is lost due to binarization) estimate of the biometric template. Similar bihashing schemes have been proposed for iris [49] and palmprint [50] modalities. Another example of salting is the cancelable face-filter approach proposed in [51] where user-specific random kernels are convolved with the face images during enrollment and authentication.

3.2. Noninvertible transform

In this approach, the biometric template is secured by applying a noninvertible transformation function to it. Noninvertible transform refers to a one-way function, \mathcal{F} , that is “easy to compute” (in polynomial time) but “hard to invert” (given $\mathcal{F}(x)$, the probability of finding x in polynomial time is small). The parameters of the transformation function are defined by a key which must be available at the time of authentication to transform the query feature set. The main characteristic of this approach is that even if the key and/or the transformed template are known, it is computationally hard (in terms of brute force complexity) for an adversary to recover the original biometric template.

Advantages

- (1) Since it is hard to recover the original biometric template even when the key is compromised, this scheme provides better security than the salting approach.
- (2) Diversity and revocability can be achieved by using application-specific and user-specific transformation functions, respectively.

Limitations

- (1) The main drawback of this approach is the tradeoff between discriminability and noninvertibility of the transformation function. The transformation function should preserve the discriminability (similarity structure) of the feature set, that is, just like in the original feature space, features from the same user should have high similarity in the transformed space, and features from different users should be quite dissimilar after transformation. On the other hand, the transformation should also be noninvertible, that is, given a transformed feature set, it should be hard for an adversary to obtain the original feature set (or a close approximation of it). It is difficult to design transformation functions that satisfy both the discriminability and noninvertibility conditions simultaneously. Moreover, the transformation function also depends on the biometric features to be used in a specific application.

Intrauser variations can be handled either by using transformation functions that are tolerant to input variations (e.g., robust hashing [53]) or by using noninvertible transformation functions that leave the biometric template in the original (feature) space even after the transformation (e.g., fingerprint minutiae can be transformed into another set of minutiae in a noninvertible manner). In the latter scenario, intrauser variations can be handled by applying the same biometric matcher on the transformed features as on the original feature set. Templates that lie in the same space after the application of a noninvertible transform have been referred to as *cancelable templates* in [32]. Noninvertible transformation functions have been proposed for fingerprint [52] and face [54] modalities in the literature.

Ratha et al. [52] proposed and analyzed three noninvertible transforms for generating cancelable fingerprint templates. The three transformation functions are cartesian, polar, and functional. These functions were used to transform fingerprint minutiae data such that a minutiae matcher can still be applied to the transformed minutiae. In cartesian transformation, the minutiae space (fingerprint image) is tessellated into a rectangular grid and each cell (possibly containing some minutiae) is shifted to a new position in the grid corresponding to the translations set by the key. The polar transformation is similar to cartesian transformation with the difference that the image is now tessellated into a number of shells and each shell is divided into sectors. Since the size of sectors can be different (sectors near the center are smaller than the ones far from the center), restrictions are placed on the translation vector generated from the key so that the radial distance of the transformed sector is not very different

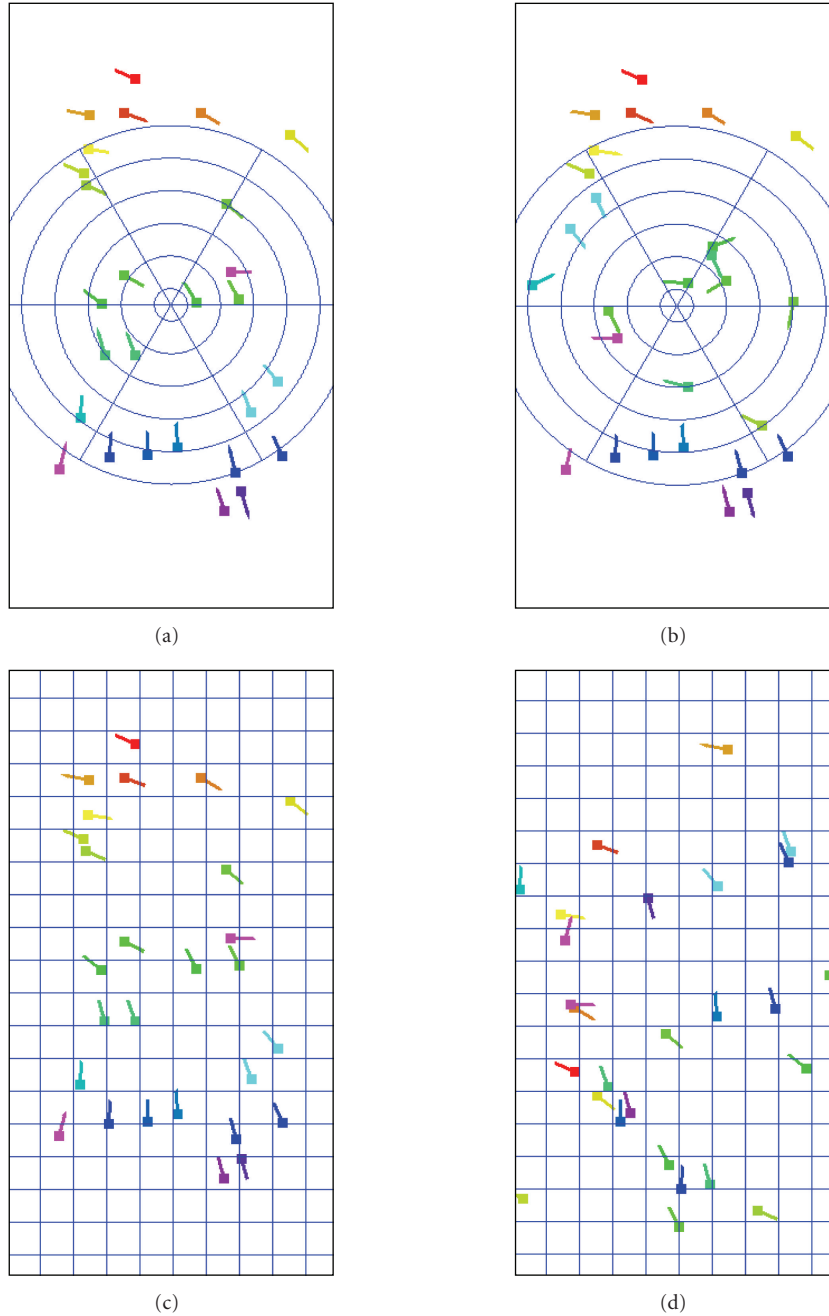


FIGURE 9: Illustration of Cartesian and polar transformation functions used in [52] for generating cancelable biometrics. (a) Original minutiae on radial grid; (b) transformed minutiae after polar transformation; (c) original minutiae on rectangular grid; and (d) transformed minutiae after Cartesian transformation.

than the radial distance of the original position. Examples of minutiae prior to and after polar and cartesian transformations are shown in Figure 9.

For the functional transformation, Ratha et al. [52] used a mixture of 2D Gaussians and electric potential field in a 2D random charge distribution as a means to translate the minutiae points. The magnitude of these functions at the point corresponding to a minutia is used as a measure of the magnitude of the translation and the gradient of a function is used to estimate the direction of translation of the minutiae.

In all the three transforms, two or more minutiae can possibly map to the same point in the transformed domain. For example, in the cartesian transformation, two or more cells can be mapped onto a single cell so that even if an adversary knows the key and hence the transformation between cells, he cannot determine the original cell to which a minutia belongs because each minutiae can independently belong to one of the possible cells. This provides a limited amount of noninvertibility to the transform. Also since the transformations used are locally smooth, the error rates are not affected

significantly and the discriminability of minutiae is preserved to a large extent.

3.3. Key-binding biometric cryptosystem

In a key-binding cryptosystem, the biometric template is secured by monolithically binding it with a key within a cryptographic framework. A single entity that embeds both the key and the template is stored in the database as helper data. This helper data does not reveal much information about the key or the biometric template, that is, it is computationally hard to decode the key or the template without any knowledge of the user's biometric data. Usually the helper data is an association of an error correcting code (selected using the key) and the biometric template. When a biometric query differs from the template within certain error tolerance, the associated codeword with similar amount of error can be recovered, which can be decoded to obtain the exact codeword, and hence recover the embedded key. Recovery of the correct key implies a successful match.

Advantages

- (1) This approach is tolerant to intrauser variations in biometric data and this tolerance is determined by the error correcting capability of the associated codeword.

Limitations

- (1) Matching has to be done using error correction schemes and this precludes the use of sophisticated matchers developed specifically for matching the original biometric template. This can possibly lead to a reduction in the matching accuracy.
- (2) In general, biometric cryptosystems are not designed to provide diversity and revocability. However, attempts are being made to introduce these two properties into biometric cryptosystems mainly by using them in conjunction with other approaches such as salting [43, 45, 55].
- (3) The helper data needs to be carefully designed; it is based on the specific biometric features to be used and the nature of associated intrauser variations.

Fuzzy commitment scheme [56] proposed by Juels and Wattenberg is a well-known example of the key binding approach. During enrollment, we commit (bind) a codeword w of an error-correcting code C using a fixed-length biometric feature vector x as the witness. Given a biometric template x , the fuzzy commitment (or the helper data) consists of $h(w)$ and $x - w$, where h is a hash function [57]. During verification, the user presents a biometric vector x' . The system subtracts $x - w$ stored in the database from x' to obtain $w' = w + \delta$, where $\delta = x' - x$. If x' is close to x , w' is close to w since $x' - x = w' - w$. Therefore, w' can now be decoded to obtain the nearest codeword which would be w provided that the distance between w and w' is less than the error correcting capacity of the code C . Reconstruction of w indicates a successful match.

A number of other template protection techniques like fuzzy vault [58], shielding functions [59], and distributed source coding [60] can be considered as key binding biometric cryptosystems. Other schemes for securing biometric templates such as the ones proposed in [61–65] also fall under this category. The fuzzy vault scheme proposed by Juels and Sudan [58] has become one of the most popular approaches for biometric template protection and its implementations for fingerprint [66–70], face [71], iris [72], and signature [73] modalities have been proposed.

3.4. Key generating biometric cryptosystem

Direct cryptographic key generation from biometrics is an attractive proposition but it is a difficult problem because of the intrauser variability. Early biometric key generation schemes such as those by Chang et al. [74] and Veilhauer et al. [75] employed user-specific quantization schemes. Information on quantization boundaries is stored as helper data which is used during authentication to account for intrauser variations. Dodis et al. [76, 77] introduced the concepts of *secure sketch* and *fuzzy extractor* in the context of key generation from biometrics. The secure sketch can be considered as helper data that leaks only limited information about the template (measured in terms of entropy loss), but facilitates exact reconstruction of the template when presented with a query that is close to the template. The fuzzy extractor is a cryptographic primitive that generates a cryptographic key from the biometric features.

Dodis et al. [76, 77] proposed secure sketches for three different distance metrics, namely, Hamming distance, set difference, and edit distance. Li and Chang [78] introduced a two-level quantization-based approach for obtaining secure sketches. Sutcu et al. [79] discussed the practical issues in secure sketch construction and proposed a secure sketch based on quantization for face biometric. The problem of generating fuzzy extractors from continuous distributions was addressed by Buhan et al. [80]. Secure sketch construction for other modalities such as fingerprints [81, 82], 3D face [83], and multimodal systems (face and fingerprint) [84] has also been proposed. Protocols for secure authentication in remote applications [85, 86] have also been proposed based on the fuzzy extractor scheme.

Key generating biometric cryptosystems usually suffer from low discriminability which can be assessed in terms of *key stability* and *key entropy*. Key stability refers to the extent to which the key generated from the biometric data is repeatable. Key entropy relates to the number of possible keys that can be generated. Note that if a scheme generates the same key irrespective of the input template, it has high key stability but zero entropy leading to high false accept rate. On the other hand, if the scheme generates different keys for different templates of the same user, the scheme has high entropy but no stability and this leads to high false reject rate. While it is possible to derive a key directly from biometric features, it is difficult to simultaneously achieve high key entropy and high key stability.

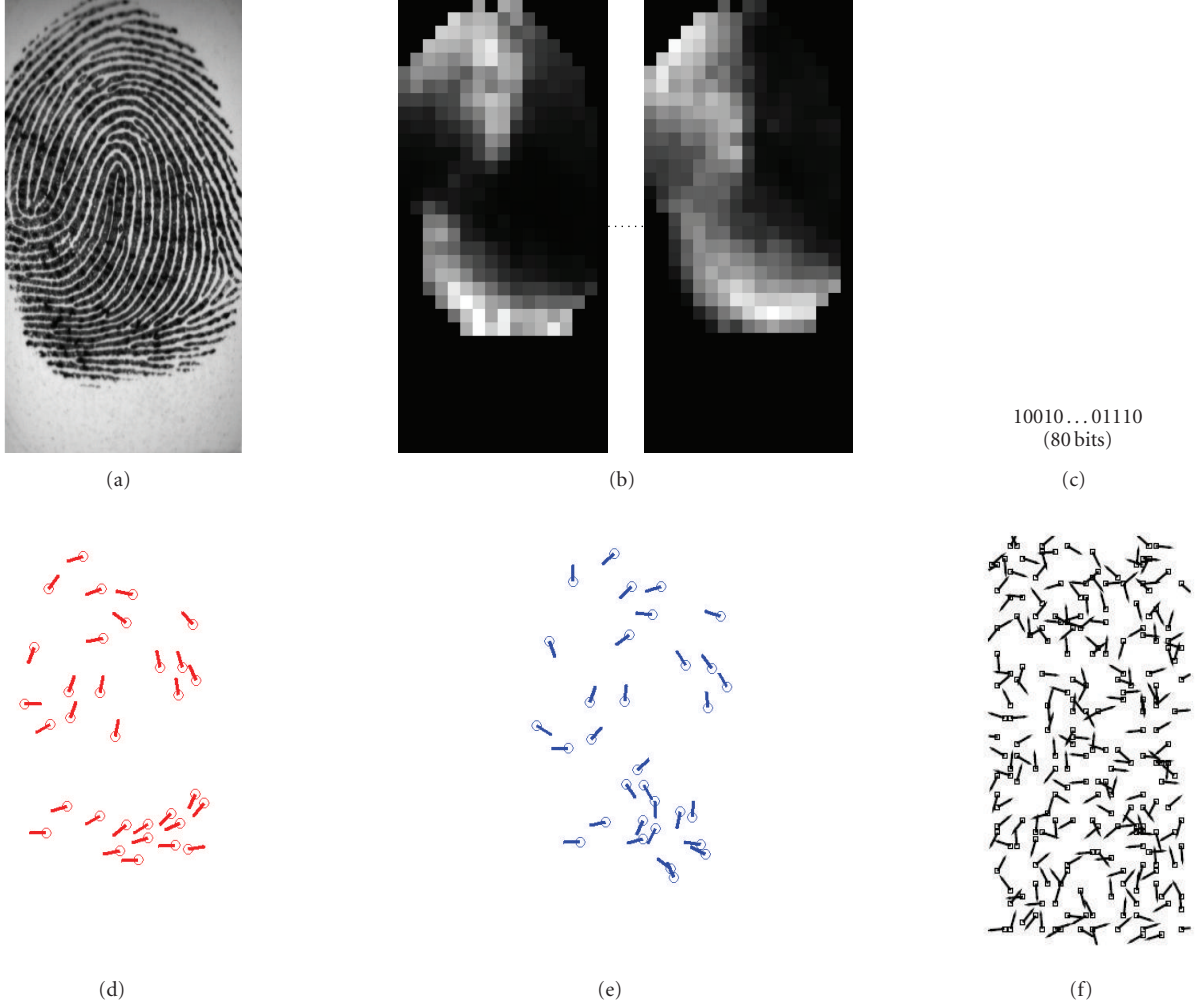


FIGURE 10: Template protection schemes implemented on a fingerprint template. (a) Fingerprint image, (b) fingercode template (texture features) corresponding to the image in (a), (c) fingercode template after salting, (d) minutiae template corresponding to the image in (a), (e) minutiae template after noninvertible functional transformation, and (f) minutiae template hidden among chaff points in a fuzzy vault.

Advantages

- (1) Direct key generation from biometrics is an appealing template protection approach which can also be very useful in cryptographic applications.

Limitations

- (1) It is difficult to generate key with high stability and entropy.

4. IMPLEMENTATION OF TEMPLATE SECURITY APPROACHES

While good implementations of salting [47], noninvertible transform [52], and key binding biometric cryptosystem [45] are available in the literature, key generation biometric cryptosystems with high key entropy and stability have been more difficult to implement in practice [79, 81, 82]. For illustration purposes, we provide implementations of the first three template protection schemes for fingerprint templates

(see Figure 10). Biometric vendors typically have their own template formats that may contain some proprietary features in order to improve the matching accuracy. For example, a fingerprint minutiae template can consist of attributes such as ridges counts, minutia type, quality of the minutia in addition to standard attributes, namely, x coordinate, y coordinate, and minutiae angle. In our implementation, we consider only the commonly used fingerprint features such as texture features and x , y and angle attributes of the minutiae.

To evaluate the performance of the three implementations, we used a public-domain fingerprint database, namely, the FVC2002-DB2. This database [87] consists of 800 images of 100 fingers with 8 impressions per finger obtained using an optical sensor. The size of the images in this database is 560×296 , the resolution of the sensor is 569 dpi and the images are generally of good quality. Our goal here is not to determine the superiority of one template protection method over the other but to simply highlight the various issues that need to be considered in implementing a template protection scheme. Of course, performance varies depending on

the choice of the biometric modality, database, and the values of the parameters used in each scheme.

4.1. Salting

We chose the random multispace quantization (RMQ) technique proposed by Teoh et al. [47] to secure the texture (fingercodes) features described in [88]. The fingercodes features were selected for this implementation, because the RMQ technique works only on fixed-length feature vectors. In this implementation, we considered the first four impressions from each finger in the FVC2002-DB2. Since the algorithm requires alignment of fingerprint images prior to the application of Fisher discriminant analysis, we align the different impressions of a finger with respect to the first impression using minutiae and find the common (overlapping) fingerprint region in all the four impressions. Texture features were extracted only for the common region, and the remaining image region was masked out.

Since our implementation inherently uses information from all the impressions of a finger (by extracting a common region from all the impressions and by doing FDA based on all the finger impressions) and then using the same images for testing (resubstitution method of error estimation), it has excellent performance (0% EER). In our implementation, we used 80 bits to represent the final feature vector. The corresponding ROC curves are shown in Figure 11. It can be inferred from the results that in case the key is secure, the impostor and genuine distributions have little overlap leading to near 0% EER. In cases where the impostor does know (or guesses) the true key, the performance of the system is close to the case when no RMQ technique is applied. Further, if the adversary knows the key, original biometric template can be recovered by him.

4.2. Noninvertible transform

We implemented two noninvertible transforms, namely, polar and functional (with a mixture of Gaussian as the transformation function) defined in [52]. For the polar transform, the central region of the image was tessellated into $n = 6$ sectors of equal angular width and 30-pixel-wide concentric shells. The transformation here is constrained such that it only shifts the sector number of the minutiae without changing the shell. There are $n!$ ways in which the n sectors in each shell can be reassigned. Given k shells in the image (constrained by the width of the image and ignoring the central region of radius 15 pixels), the number of different ways a transformation can be constructed is $(n!)^k$ which is equivalent to $\log_2(n!)^k$ bits of security.

For the functional transformation, we used a mixture of 24 Gaussians with the same isotropic standard deviation of 30 pixels (where the peaks can correspond to +1 or -1 as used in [52]) for calculating the displacement and used the direction of gradient of the mixture of Gaussian function as the direction of minutiae displacement. Since the mean vector of the Gaussians can fall anywhere in the image, there are 296×560 possible different values of means of each Gaussian component. As there are 24 Gaussian components and each

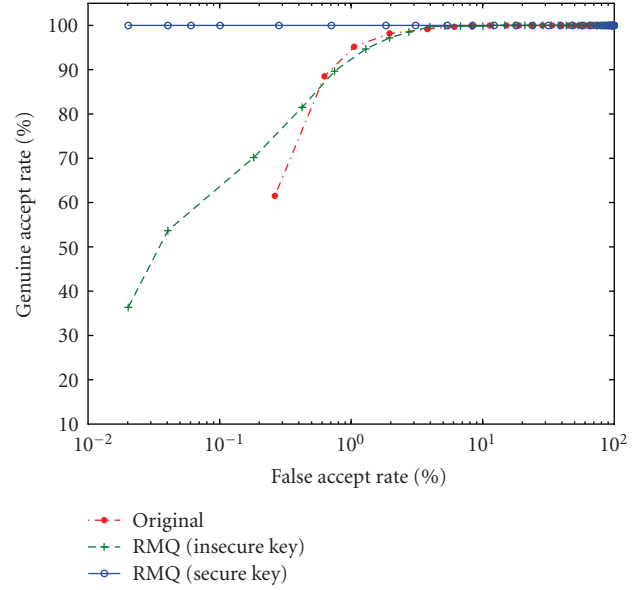


FIGURE 11: ROC curves of random multispace quantization (RMQ) [47] using the texture features proposed in [88]. The “Original” curve corresponds to the matcher using the Euclidean distance between texture features of query and template; “Insecure Key” curve corresponds to the case when the impostor knows the key (used to generate the random feature space); “Secure Key” curve corresponds to the case when the impostor does not know the key.

one can peak at +1 or -1, there are $(296 \times 560 \times 2)^{24}$ possible transformations. However, two transformations with slightly shifted component means will produce two similar templates such that one template can be used to verify the other.

To analyze the security of the functional transformation, Ratha et al. [52] assumed that for each minutiae in the fingerprint, its transformed counterpart could be present in a shell of width d pixels at a distance of K pixels from the minutiae. Further, assuming that the matcher cannot distinguish minutiae that are within δr pixels and their orientations are within $\delta \theta$ degrees, each transformed minutiae encodes $I_m = \log_2(\pi((K+d)^2 - K^2)/(\delta r)^2 * \pi/\delta \theta)$ bits of information. Assuming that there are N minutiae in template fingerprint and one needs to match at least m minutiae to get accepted, the adversary needs to make $2^{I_m * m - \log_2(\binom{N}{m})}$ attempts. Note that this analysis is based on the simplifying assumption that each minutia is transformed independently. This overestimates the number of attempts needed by an adversary to guess the biometric template.

Among the eight impressions available for each of the 100 fingers in FVC2002-DB2, we use only the first two impressions in this experiment because they have the best image quality. The results, based on the minutiae matcher in [89], are shown in Figure 12 which indicates a decrease in GAR for a fixed FAR. In terms of security, noninvertible transformation is one of the better approaches since it is computationally hard (in terms of brute force complexity) to invert the stored template and obtain the true template. The true template is never revealed especially in case when the

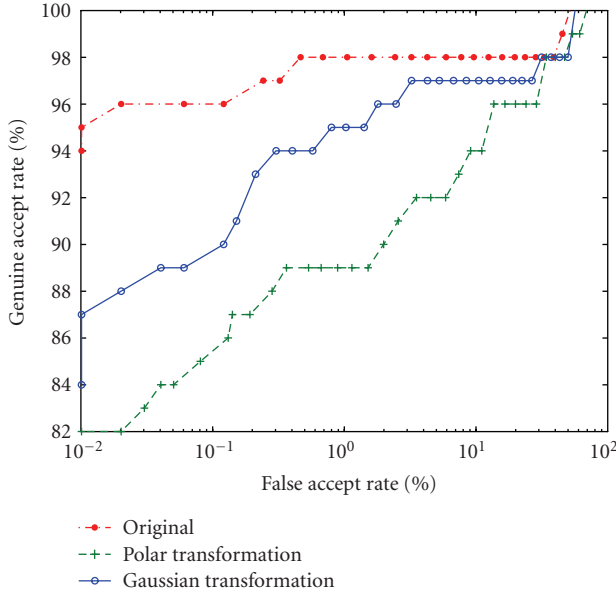


FIGURE 12: ROC curves corresponding to two noninvertible transforms (Gaussian and polar) on FVC2002-DB2. The “Original” curve represents the case where no transformation is applied to the template, “Gaussian” curve corresponds to the functional transformation of the template, and “Polar” corresponds to the polar transformation of the template.

TABLE 3: Performance summary of the fuzzy vault implementation for FVC2002-DB2 database. Here, n denotes the degree of the encoding polynomial used in vault construction. The maximum key size that can be bound to the minutiae template is $16n$ bits.

FTCR	$n = 7$		$n = 8$		$n = 10$	
	GAR	FAR	GAR	FAR	GAR	FAR
2%	91%	0.13%	91%	0.01%	86	0%

transformation of the biometric template is done on a separate module (possibly a handheld device [38]) which does not save the original template in memory and is not accessible to an adversary.

4.3. Key-binding biometric cryptosystem

A fuzzy vault was chosen for implementation because concrete implementations on real fingerprint data sets are not yet available for many of the other key-binding biometric cryptosystems. We implemented the fuzzy vault as proposed in [90] using the first two impressions of each of the 100 fingers in the FVC2002-DB2. Table 3 shows the error rates corresponding to different key sizes used in binding. Compared to the “original” ROC curve in Figure 12, we observe that the fuzzy vault scheme has a lower genuine accept rate by about 4%. Further, this scheme also has failure to capture errors if the number of minutiae in the fingerprint image is not sufficient for vault construction (minimum number of minutiae required in our implementation is 18).

Dodis et al. [76, 77] defined the security of biometric cryptosystems in terms of the min-entropy of the helper data. In particular, they provided the bounds on min-entropy for the fuzzy vault construction in [58]. The security of the fuzzy vault scheme has also been studied by Chang et al. [91]. An advantage of the fuzzy vault (key binding) scheme is that instead of providing a “Match/Non-match” decision, the vault decoding outputs a key that is embedded in the vault. This key can be used in a variety of ways to authenticate a person (e.g., digital signature, document encryption/decryption, etc.).

There are some specific attacks that can be staged against a fuzzy vault, that is, *attacks via record multiplicity*, *stolen key inversion attack*, and *blended substitution attack* [92]. If an adversary has access to two different vaults (say from two different applications) obtained from the same biometric data, he can easily identify the genuine points in the two vaults and decode the vault. Thus, the fuzzy vault scheme does not provide diversity and revocability. In a stolen key inversion attack, if an adversary somehow recovers the key embedded in the vault, he can decode the vault to obtain the biometric template. Since the vault contains a large number of chaff points, it is possible for an adversary to substitute a few points in the vault using his own biometric features. This allows both the genuine user and the adversary to be successfully authenticated using the same identity and such an attack is known as blended substitution. To counter these attacks, Nandakumar et al. [45] proposed a hybrid approach where (i) biometric features are first “salted” based on a user password, (ii) vault is constructed using the salted template, and (iii) the vault is encrypted using a key derived from the password. While salting prevents attacks via record multiplicity and provides diversity and revocability, encryption provides resistance against blended substitution and stolen key inversion attacks.

4.4. Discussion

We believe that as yet there is no “best” approach for template protection. The application scenario and requirements play a major role in the selection of a template protection scheme. For instance, in a biometric verification application such as a bank ATM, a simple salting scheme based on the user’s PIN may be sufficient to secure the biometric template if we assume that both the transformed template and the user’s PIN will not be compromised simultaneously. On the other hand, in an airport watch-list application, noninvertible transform is a more suitable approach because it provides both template security and revocability without relying on any other input from the user. Biometric cryptosystems are more appropriate in match-on-card applications because such systems typically release a key to the associated application in order to indicate a successful match.

The other major factors that influence the choice of a template protection scheme are the selected biometric trait, its feature representation, and the extent of intrauser variations. Design of a template protection scheme depends on the specific type of biometric features used. While good noninvertible transforms have been proposed for fingerprint

minutiae features [52], it may be difficult to design a suitable noninvertible transform for IrisCode representation [25]. In contrast, it may be easier to design a biometric cryptosystem for IrisCode because it is represented as a fixed-length binary string where standard error-correction coding techniques can be readily applied. Moreover, if the intrauser variations are quite large, it may not be possible to apply a noninvertible transform or create a biometric cryptosystem. Therefore, even in a specific application scenario and for a fixed biometric feature representation, more than one template protection scheme may be admissible, and the choice of the suitable approach may be based on a number of factors such as recognition performance, computational complexity, memory requirements, and user acceptance and co-operation.

5. SUMMARY AND RESEARCH DIRECTIONS

Given the dramatic increase in incidents involving identity thefts and various security threats, it is imperative to have reliable identity management systems. Biometric systems are being widely used to achieve reliable user authentication, a critical component in identity management. But, biometric systems themselves are vulnerable to a number of attacks. In this paper, we have summarized various aspects of biometric system security and discussed techniques to counter these threats. Among these vulnerabilities, an attack against stored biometric templates is a major concern due to the strong linkage between a user's template and his identity and the irrevocable nature of biometric templates. We have described various template protection mechanisms proposed in the literature and highlighted their strengths and limitations. Finally, specific implementations of these approaches on a common fingerprint database were presented to illustrate the issues involved in implementing template security.

The available template protection schemes are not yet sufficiently mature for large scale deployment; they do not meet the requirements of diversity, revocability, security, and high-recognition performance. Further, the security analysis of existing schemes is mostly based on the complexity of brute force attacks which assumes that the distribution of biometric features is uniform. In practice, an adversary may be able to exploit the nonuniform nature of biometric features to launch an attack that may require significantly fewer attempts to compromise the system security. While we have pointed out some of the vulnerabilities in specific schemes such as fuzzy vault, a rigorous analysis of the cryptographic strength of the template security schemes similar to those available in the cryptanalysis literature has not been carried out till date. Such an analysis must be performed before the template security schemes are deployed in critical real-world applications.

A single template protection approach may not be sufficient to meet all the application requirements. Hence, hybrid schemes that make use of the advantages of the different template protection approaches must be developed. For instance, a scheme that secures a "salted" template using a biometric cryptosystem (e.g., [44–46]) may have the advantages of both salting (which provides high diversity and revocability) and

biometric cryptosystem (which provides high security) approaches. Finally, with the growing interest in multibiometric and multifactor authentication systems, schemes that simultaneously secure multibiometric templates and multiple authentication factors (biometrics, passwords, etc.) need to be developed.

ACKNOWLEDGMENTS

This research was supported by Army Research Office contract W911NF-06-1-0418 and the Center for Identification Technology Research (CITEr), and NSF/IUCRC program. The authors would like to thank Salil Prabhakar, Sharath Pankanti, Andrew Teoh, and the anonymous reviewers for their feedback and useful suggestions.

REFERENCES

- [1] A. K. Jain, A. Ross, and S. Pankanti, "Biometrics: a tool for information security," *IEEE Transactions on Information Forensics and Security*, vol. 1, no. 2, pp. 125–143, 2006.
- [2] C. Roberts, "Biometric attack vectors and defences," *Computers and Security*, vol. 26, no. 1, pp. 14–25, 2007.
- [3] M1.4 Ad Hoc Group on Biometric in E-Authentication, "Study report on biometrics in E-authentication," Tech. Rep. INCITS M1/07-0185rev, International Committee for Information Technology Standards (INCITS), Washington, DC, USA, August 2007.
- [4] I. Buhan and P. Hartel, "The state of the art in abuse of biometrics," Tech. Rep. TR-CTIT-05-41, Centre for Telematics and Information Technology, University of Twente, Twente, The Netherlands, December 2005.
- [5] A. K. Jain, A. Ross, and U. Uludag, "Biometric template security: challenges and solutions," in *Proceedings of the European Signal Processing Conference (EUSIPCO '05)*, Antalya, Turkey, September 2005.
- [6] B. Cukic and N. Bartlow, "Biometric system threats and countermeasures: a risk based approach," in *Proceedings of the Biometric Consortium Conference (BCC '05)*, Crystal City, Va, USA, September 2005.
- [7] K. Ishikawa, *Guide to Quality Control*, Nordica International, Tokyo, Japan.
- [8] A. Ross, K. Nandakumar, and A. K. Jain, *Handbook of Multi-biometrics*, Springer, Berlin, Germany, 2006.
- [9] Biometric System Laboratory - University of Bologna, "FVC2006: the 4th international fingerprint verification competition," 2006, <http://bias.csr.unibo.it/fvc2006/default.asp>.
- [10] C. Wilson, A. R. Hicklin, M. Bone, et al., "Fingerprint vendor technology evaluation 2003: summary of results and analysis report," Tech. Rep. NISTIR 7123, National Institute of Standards and Technology, Gaithersburg, Md, USA, June 2004.
- [11] P. J. Phillips, W. T. Scruggs, A. J. Ó Toole, et al., "FRVT 2006 and ICE 2006 large-scale results," Tech. Rep. NISTIR 7408, National Institute of Standards and Technology, 2007.
- [12] M. Przybicki and A. Martin, "NIST speaker recognition evaluation chronicles," in *Proceedings of Odyssey: The Speaker and Language Recognition Workshop*, pp. 12–22, Toledo, Spain, May 2004.
- [13] N. K. Ratha, J. H. Connell, and R. M. Bolle, "An analysis of minutiae matching strength," in *Proceedings of the 3rd Interna-*

- tional Conference on Audio- and Video-Based Biometric Person Authentication (AVBPA '01), pp. 223–228, Halmstad, Sweden, June 2001.
- [14] T. Matsumoto, H. Matsumoto, K. Yamada, and S. Hoshino, "Impact of artificial "gummy" fingers on fingerprint systems," in *Optical Security and Counterfeit Deterrence Techniques IV*, vol. 4677 of *Proceedings of SPIE*, pp. 275–289, San Jose, Calif, USA, January 2002.
 - [15] T. Matsumoto, M. Hirabayashi, and K. Sato, "A vulnerability evaluation of iris matching (part 3)," in *Proceedings of the Symposium on Cryptography and Information Security (SCIS '04)*, pp. 701–706, Iwate, Japan, January 2004.
 - [16] W. R. Harrison, *Suspect Documents, Their Scientific Examination*, Nelson-Hall, Chicago, Ill, USA, 1981.
 - [17] A. Eriksson and P. Wretling, "How flexible is the human voice? A case study of mimicry," in *Proceedings of the European Conference on Speech Technology (Eurospeech '97)*, pp. 1043–1046, Rhodes, Greece, September 1997.
 - [18] S. T. V. Parthasaradhi, R. Derakhshani, L. A. Hornak, and S. A. C. Schuckers, "Time-series detection of perspiration as a liveness test in fingerprint devices," *IEEE Transactions on Systems, Man and Cybernetics Part C*, vol. 35, no. 3, pp. 335–343, 2005.
 - [19] A. Antonelli, R. Cappelli, D. Maio, and D. Maltoni, "Fake finger detection by skin distortion analysis," *IEEE Transactions on Information Forensics and Security*, vol. 1, no. 3, pp. 360–373, 2006.
 - [20] D. R. Setlak, "Fingerprint sensor having spoof reduction features and related methods," US patent no. 5953441, 1999.
 - [21] K. A. Nixon and R. K. Rowe, "Multispectral fingerprint imaging for spoof detection," in *Biometric Technology for Human Identification II*, vol. 5779 of *Proceedings of SPIE*, pp. 214–225, Orlando, Fla, USA, March 2005.
 - [22] J. Li, Y. Wang, T. Tan, and A. K. Jain, "Live face detection based on the analysis of fourier spectra," in *Biometric Technology for Human Identification*, vol. 5404 of *Proceedings of SPIE*, pp. 296–303, Orlando, Fla, USA, April 2004.
 - [23] K. Kollreider, H. Fronthaler, and J. Bigun, "Evaluating liveness by face images and the structure tensor," in *Proceedings of the 4th IEEE Workshop on Automatic Identification Advanced Technologies (AUTO ID '05)*, pp. 75–80, Buffalo, NY, USA, October 2005.
 - [24] H.-K. Jee, S.-U. Jung, and J.-H. Yoo, "Liveness detection for embedded face recognition system," *International Journal of Biomedical Sciences*, vol. 1, no. 4, pp. 235–238, 2006.
 - [25] J. Daugman, "Recognizing persons by their iris patterns," in *Biometrics: Personal Identification in Networked Society*, A. K. Jain, R. Bolle, and S. Pankanti, Eds., pp. 103–122, Kluwer Academic Publishers, London, UK, 1999.
 - [26] E. C. Lee, K. R. Park, and J. Kim, "Fake iris detection by using purkinje image," in *Proceedings of International Conference on Advances in Biometrics (ICB '06)*, vol. 3832 of *Lecture Notes in Computer Science*, pp. 397–403, Hong Kong, 2006.
 - [27] A. Juels, D. Molnar, and D. Wagner, "Security and privacy issues in E-passports," in *Proceedings of the 1st International Conference on Security and Privacy for Emerging Areas in Communications Networks, (SecureComm '05)*, pp. 74–88, Athens, Greece, September 2005.
 - [28] P. Syverson, "A taxonomy of replay attacks," in *Proceedings of the Computer Security Foundations Workshop (CSFW '97)*, pp. 187–191, Franconia, NH, USA, June 1994.
 - [29] A. Adler, "Vulnerabilities in biometric encryption systems," in *Proceedings of the 5th International Conference on Audio- and Video-Based Biometric Person Authentication (AVBPA '05)*, vol. 3546 of *Lecture Notes in Computer Science*, pp. 1100–1109, Hilton Rye Town, NY, USA, July 2005.
 - [30] K. Lam and D. Gollmann, "Freshness assurance of authentication protocols," in *Proceedings of the European Symposium on Research in Computer Security (ESORICS '92)*, pp. 261–272, Toulouse, France, 1992.
 - [31] K. Lam and T. Beth, "Timely authentication in distributed systems," in *Proceedings of the European Symposium on Research in Computer Security (ESORICS '92)*, vol. 648, pp. 293–303, Toulouse, France, 1992.
 - [32] R. M. Bolle, J. H. Connell, and N. K. Ratha, "Biometric perils and patches," *Pattern Recognition*, vol. 35, no. 12, pp. 2727–2738, 2002.
 - [33] R. Seacord, *Secure Coding in C and C++*, Addison-Wesley, Reading, Mass, USA, 2005.
 - [34] A. Ross, J. Shah, and A. K. Jain, "From template to image: reconstructing fingerprints from minutiae points," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 4, pp. 544–560, 2007.
 - [35] R. Cappelli, A. Lumini, D. Maio, and D. Maltoni, "Fingerprint image reconstruction from standard templates," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 9, pp. 1489–1503, 2007.
 - [36] A. Adler, "Images can be regenerated from quantized biometric match score data," in *Proceedings of the Canadian Conference on Electrical and Computer Engineering*, vol. 1, pp. 469–472, Niagara Falls, Ontario, Canada, May 2004.
 - [37] A. K. Jain, R. Bolle, and S. Pankanti, Eds., *Biometrics: Personal Identification in Networked Society*, Kluwer Academic Publishers, Dordrecht, The Netherlands, 1999.
 - [38] A. K. Jain and S. Pankanti, "A touch of money," *IEEE Spectrum*, vol. 43, no. 7, pp. 22–27, 2006.
 - [39] D. Maltoni, D. Maio, A. K. Jain, and S. Prabhakar, *Handbook of Fingerprint Recognition*, Springer, Berlin, Germany, 2003.
 - [40] U. Uludag, S. Pankanti, S. Prabhakar, and A. K. Jain, "Biometric cryptosystems: issues and challenges," *Proceedings of the IEEE*, vol. 92, no. 6, pp. 948–960, 2004.
 - [41] A. Cavoukian and A. Stoianov, "Biometric encryption: a positive-sum technology that achieves strong authentication, security and privacy," Tech. Rep., Office of the Information and Privacy Commissioner of Ontario, Toronto, Ontario, Canada, March 2007.
 - [42] A. Vetro and N. Memon, "Biometric system security," in *Proceedings of the 2nd International Conference on Biometrics*, Seoul, South Korea, August 2007.
 - [43] T. E. Boulton, W. J. Scheirer, and R. Woodwork, "Revocable fingerprint biotokens: accuracy and security analysis," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '97)*, pp. 1–8, Minneapolis, Minn, USA, June 2007.
 - [44] C. Soutar, D. Roberge, A. Stoianov, R. Gilroy, and B. V. K. V. Kumar, "Biometric Encryption," in *ICSA Guide to Cryptography*, R. K. Nichols, Ed., McGraw Hill, New York, NY, USA, 1999.
 - [45] K. Nandakumar, A. Nagar, and A. K. Jain, "Hardening fingerprint fuzzy vault using password," in *Proceedings of 2nd International Conference on Biometrics*, pp. 927–937, Seoul, South Korea, August 2007.
 - [46] O. T. Song, A. B. J. Teoh, and D. C. L. Ngo, "Application-specific key release scheme from biometrics," *International Journal of Network Security*, vol. 6, no. 2, pp. 127–133, 2008.
 - [47] A. B. J. Teoh, A. Goh, and D. C. L. Ngo, "Random multi-space quantization as an analytic mechanism for BioHashing of biometric and random identity inputs," *IEEE Transactions*

- on *Pattern Analysis and Machine Intelligence*, vol. 28, no. 12, pp. 1892–1901, 2006.
- [48] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, “Eigenfaces versus fisherfaces: recognition using class specific linear projection,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 9, no. 7, pp. 711–720, 1997.
 - [49] C. S. Chin, A. T. B. Jin, and D. N. C. Ling, “High security iris verification system based on random secret integration,” *Computer Vision and Image Understanding*, vol. 102, no. 2, pp. 169–177, 2006.
 - [50] T. Connie, A. B. J. Teoh, M. Goh, and D. C. L. Ngo, “PalmHashing: a novel approach for cancelable biometrics,” *Information Processing Letters*, vol. 93, no. 1, pp. 1–5, 2005.
 - [51] M. Savvides and B. V. K. Vijaya Kumar, “Cancellable biometric filters for face recognition,” in *Proceedings of the IEEE International Conference Pattern Recognition (ICPR '94)*, vol. 3, pp. 922–925, Cambridge, UK, August 2004.
 - [52] N. K. Ratha, S. Chikkerur, J. H. Connell, and R. M. Bolle, “Generating cancelable fingerprint templates,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 4, pp. 561–572, 2007.
 - [53] Y. Sutcu, H. T. Sencar, and N. Memon, “A secure biometric authentication scheme based on robust hashing,” in *Proceedings of the 7th Multimedia and Security Workshop (MM and Sec '05)*, pp. 111–116, New York, NY, USA, August 2006.
 - [54] A. B. J. Teoh, K.-A. Toh, and W. K. Yip, “ 2^N discretisation of BioPhasor in cancellable biometrics,” in *Proceedings of 2nd International Conference on Biometrics*, pp. 435–444, Seoul, South Korea, August 2007.
 - [55] X. Boyen, “Reusable cryptographic fuzzy extractors,” in *Proceedings of the ACM Conference on Computer and Communications Security (ACM CCS '04)*, pp. 82–91, Washington, DC, USA, October 2004.
 - [56] A. Juels and M. Wattenberg, “A fuzzy commitment scheme,” in *Proceedings of 6th ACM Conference on Computer and Communications Security (ACM CCS '99)*, pp. 28–36, Singapore, November 1999.
 - [57] J. L. Carter and M. N. Wegman, “Universal classes of hash functions,” *Journal of Computer and System Sciences*, vol. 18, no. 2, pp. 143–154, 1979.
 - [58] A. Juels and M. Sudan, “A fuzzy vault scheme,” in *Proceedings of the IEEE International Symposium on Information Theory*, p. 408, Piscataway, NJ, USA, June–July 2002.
 - [59] P. Tuyls, A. H. M. Akkermans, T. A. M. Kevenaar, G.-J. Schrijen, A. M. Bazen, and R. N. J. Veldhuis, “Practical biometric authentication with template protection,” in *Proceedings of the 5th International Conference on Audio- and Video-Based Biometric Person Authentication (AVBPA '05)*, vol. 3546 of *Lecture Notes in Computer Science*, pp. 436–446, Hilton Rye Town, NY, USA, July 2005.
 - [60] S. C. Draper, A. Khisti, E. Martinian, A. Vetro, and J. S. Yedidia, “Using distributed source coding to secure fingerprint biometrics,” in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '07)*, vol. 2, pp. 129–132, Honolulu, Hawaii, USA, April 2007.
 - [61] G. I. Davida, Y. Frankel, and B. J. Matt, “On enabling secure applications through off-line biometric identification,” in *Proceedings of the IEEE Computer Society Symposium on Research in Security and Privacy*, pp. 148–157, Oakland, Calif, USA, May 1998.
 - [62] F. Monrose, M. K. Reiter, and S. Wetzel, “Password hardening based on keystroke dynamics,” in *Proceedings of the 6th ACM conference on Computer and Communications Security (ACM CCS '99)*, pp. 73–82, Singapore, November 1999.
 - [63] F. Monrose, M. K. Reiter, Q. Li, and S. Wetzel, “Cryptographic key generation from voice,” in *Proceedings of IEEE Computer Society Symposium on Research in Security and Privacy*, pp. 202–213, Oakland, Calif, USA, May 2001.
 - [64] F. Hao, R. Anderson, and J. Daugman, “Combining crypto with biometrics effectively,” *IEEE Transactions on Computers*, vol. 55, no. 9, pp. 1081–1088, 2006.
 - [65] E. J. C. Kelkboom, B. Gkberk, T. A. M. Kevenaar, A. H. M. Akkermans, and M. van der Veen, “3D face: biometric template protection for 3D face recognition,” in *Proceedings of 2nd International Conference on Biometrics*, pp. 566–573, Seoul, South Korea, August 2007.
 - [66] T. Clancy, D. Lin, and N. Kiyavash, “Secure smartcard-based fingerprint authentication,” in *Proceedings of the ACM SIGMM Workshop on Biometric Methods and Applications*, pp. 45–52, Berkley, Mich, USA, November 2003.
 - [67] S. Yang and I. Verbauwhede, “Automatic secure fingerprint verification system based on fuzzy vault scheme,” in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '05)*, vol. 5, pp. 609–612, Philadelphia, Pa, USA, March 2005.
 - [68] Y. Chung, D. Moon, S. Lee, S. Jung, T. Kim, and D. Ahn, “Automatic alignment of fingerprint features for fuzzy fingerprint vault,” in *Proceedings of the 1st Conference on Information Security and Cryptology (CISC '05)*, vol. 3822 of *Lecture Notes in Computer Science*, pp. 358–369, Beijing, China, December 2005.
 - [69] U. Uludag and A. K. Jain, “Securing fingerprint template: fuzzy vault with helper data,” in *Proceedings of the Conference on Computer Vision and Pattern Recognition Workshops (CVPRW '06)*, p. 163, New York, NY, USA, June 2006.
 - [70] A. Nagar and S. Chaudhury, “Biometrics based asymmetric cryptosystem design using modified fuzzy vault scheme,” in *Proceedings of the 18th International Conference on Pattern Recognition (ICPR '06)*, vol. 4, pp. 537–540, Hong Kong, August 2006.
 - [71] Y. C. Feng and P. C. Yuen, “Protecting face biometric data on smartcard with reed-solomon code,” in *Proceedings of the Conference on Computer Vision and Pattern Recognition Workshops (CVPRW '06)*, p. 29, New York, NY, USA, June 2006.
 - [72] Y. J. Lee, K. Bae, S. J. Lee, K. R. Park, and J. Kim, “Biometric key binding: fuzzy vault based on iris images,” in *Proceedings of 2nd International Conference on Biometrics*, pp. 800–808, Seoul, South Korea, August 2007.
 - [73] M. Freire-Santos, J. Fierrez-Aguilar, and J. Ortega-Garcia, “Cryptographic key generation using handwritten signature,” in *Biometric Technology for Human Identification III*, vol. 6202 of *Proceedings of SPIE*, pp. 225–231, Orlando, Fla, USA, April 2006.
 - [74] Y.-J. Chang, W. Zhang, and T. Chen, “Biometrics-based cryptographic key generation,” in *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME '04)*, vol. 3, pp. 2203–2206, Taipei, Taiwan, June 2004.
 - [75] C. Viehauer, R. Steinmetz, and A. Mayerhöfer, “Biometric hash based on statistical features of online signatures,” in *Proceedings of the International Conference on Pattern Recognition*, vol. 1, pp. 123–126, Quebec, QC, Canada, August 2002.
 - [76] Y. Dodis, R. Ostrovsky, L. Reyzin, and A. Smith, “Fuzzy extractors: how to generate strong keys from biometrics and other noisy data,” Tech. Rep. 235, Cryptology ePrint Archive, February 2006.
 - [77] Y. Dodis, L. Reyzin, and A. Smith, “Fuzzy extractors: how to generate strong keys from biometrics and other noisy data,” in *Proceedings of International Conference on the Theory and*

- Applications of Cryptographic Techniques: Advances in Cryptology (EUROCRYPT '04)*, vol. 3027 of *Lecture Notes in Computer Science*, pp. 523–540, Interlaken, Switzerland, May 2004.
- [78] Q. Li and E.-C. Chang, “Robust, short and sensitive authentication tags using secure sketch,” in *Proceedings of the 8th Multimedia and Security Workshop (MM and Sec '06)*, pp. 56–61, Geneva, Switzerland, September 2006.
 - [79] Y. Sutcu, Q. Li, and N. Memon, “Protecting biometric templates with sketch: theory and practice,” *IEEE Transactions on Information Forensics and Security*, vol. 2, no. 3, pp. 503–512, 2007.
 - [80] I. Buhan, J. Doumen, P. Hartel, and R. Veldhuis, “Fuzzy extractors for continuous distributions,” in *Proceedings of the 2nd ACM Symposium on Information, Computer and Communications Security (ASIACCS '07)*, pp. 353–355, Singapore, March 2007.
 - [81] A. Arakala, J. Jeffers, and K. J. Horadam, “Fuzzy extractors for minutiae-based fingerprint authentication,” in *Proceedings of the 2nd International Conference on Biometrics*, pp. 760–769, Seoul, South Korea, August 2007.
 - [82] E. C. Chang and S. Roy, “Robust extraction of secret bits from minutiae,” in *Proceedings of 2nd International Conference on Biometrics*, pp. 750–759, Seoul, South Korea, August 2007.
 - [83] X. Zhou, “Template protection and its implementation in 3D face recognition systems,” in *Biometric Technology for Human Identification IV*, vol. 6539 of *Proceedings of SPIE*, pp. 214–225, Orlando, Fla, USA, April 2007.
 - [84] Y. Sutcu, Q. Li, and N. Memon, “Secure biometric templates from fingerprint-face features,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '07)*, Minneapolis, Minn, USA, June 2007.
 - [85] X. Boyen, Y. Dodis, J. Katz, R. Ostrovsky, and A. Smith, “Secure remote authentication using biometric data,” in *Proceedings of the 24th Annual International Conference on Advances in Cryptology (EUROCRYPT '06)*, vol. 3494 of *Lecture Notes in Computer Science*, pp. 147–163, Aarhus, Denmark, May 2005.
 - [86] I. R. Buhan, J. M. Doumen, P. H. Hartel, and R. N. J. Veldhuis, “Secure ad-hoc pairing with biometrics: SAfE,” in *Proceedings of 1st International Workshop on Security for Spontaneous Interaction (IWSSI '07)*, pp. 450–456, Innsbruck, Austria, September 2007.
 - [87] D. Maio, D. Maltoni, R. Cappelli, J. L. Wayman, and A. K. Jain, “FVC2002: second fingerprint verification competition,” in *Proceedings of the International Conference on Pattern Recognition*, vol. 3, pp. 811–814, Quebec, QC, Canada, August 2002.
 - [88] A. Ross, A. K. Jain, and J. Reisman, “A hybrid fingerprint matcher,” *Pattern Recognition*, vol. 36, no. 7, pp. 1661–1673, 2003.
 - [89] A. K. Jain, L. Hong, and R. Bolle, “On-line fingerprint verification,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 4, pp. 302–314, 1997.
 - [90] K. Nandakumar, A. K. Jain, and S. Pankanti, “Fingerprint-based fuzzy vault: implementation and performance,” *IEEE Transactions on Information Forensics and Security*, vol. 2, no. 4, pp. 744–757, 2007.
 - [91] E.-C. Chang, R. Shen, and F. W. Teo, “Finding the original point set hidden among chaff,” in *Proceedings of the ACM Symposium on Information, Computer and Communications Security (ASIACCS '06)*, pp. 182–188, Taipei, Taiwan, March 2006.
 - [92] W. J. Scheirer and T. E. Boult, “Cracking fuzzy vaults and biometric encryption,” in *Proceedings of the Biometrics Symposium*, Baltimore, Md, USA, September 2007.