Protecting Biometric Templates with Sketch: Theory and Practice

Yagiz Sutcu*, Qiming Li, Nasir Memon

***Abstract*— Secure storage of biometric templates has become an increasingly important issue in biometric authentication systems. We study how *secure sketch*, a recently proposed error-tolerant cryptographic primitive, can be applied to protect the templates. We identify several practical issues that are not addressed in the existing theoretical framework, and show the subtleties in evaluating the security of practical systems. We propose a general framework to design and analyze secure sketch for biometric templates, and give a concrete construction for face biometrics as an example. We show that theoretical bounds have their limitations in practical schemes, and the exact security of the system often needs more careful investigations. We further discuss how to use secure sketch in the design of multi-factor authentication systems that allow easy revocation of user credentials.**

***Index Terms*— Biometric template security, secure sketch, entropy loss**

I. Introduction

In many biometric authentication systems, the biometric templates of users are sampled during an *enrollment phase*, and are stored in the system, either in a central database, or in smartcards. Later, when the user wants to authenticate himself/herself to the system, a fresh measurement of the same biometrics is taken and is matched against the corresponding template. If they are sufficiently similar according to some similarity measure, the user is considered as authentic. These biometric templates are often stored in the form of raw samples of the user biometrics (e.g. scanned fingerprints, or photographs of faces). If these templates are compromised by attackers, they can be used to impersonate legitimate users. In some cases, features extracted from raw samples are stored instead (e.g., minutiae of fingerprints, or SVD of face images). When a fresh measurement of the same biometrics is made, the same feature extraction algorithm is applied, and the extracted features are compared against the template. However, in this case, it is often not clear how difficult it is to forge a biometric sample that would generate the same features using the same feature extraction algorithm, especially when the feature extraction algorithm is compromised together with the template.

Secure storage of user credentials is not a new problem. In many UNIX-like systems, user credentials are stored in a shadow password file, where the passwords are hashed and only the hash values are stored. When a user enters a password, it is hashed and matched against the stored hash value, and the user is considered as authentic if the hash values are exactly the same. In this way, if the hashed passwords are compromised, it would still be difficult for any attacker to guess the passwords, even if the hashing function is publicly known. Legitimate users, after detecting the compromise, can change their passwords, which makes old passwords useless to attackers.

Unfortunately, such techniques cannot be easily adapted to protect biometric templates. The main difficulty is that biometric samples cannot be exactly reproduced, and traditional cryptographic primitives do not allow even a single bit of error. To make things worse, biometric templates, once comprised, are difficult (if possible at all) to revoke or replace.

There has been much work to solve this problem with various approaches. These methods can be roughly categorized into two types: (1) Robust hash functions, where small changes in a biometric sample would yield the same hash value (e.g., [1], [2], [3], [4], [5]); (2) Similarity-preserving hard-to-invert transformations, where similarity of biometric samples would be preserved through the transformation, yet it is difficult to find the original template from a transformed one (e.g., [6], [7], [8], [9]). We note, however, that there lacks rigorous security analysis for these techniques. In particular, it is not clear exactly how difficult it is to break these schemes once the hash values (or the transformed templates) are compromised, especially when the hash function, transformation algorithm and related keys and parameters are also compromised.

Yet another approach, which allows more rigorous security analysis, is to employ recently proposed cryptographic primitives, where some public information P can be used to recover the original biometric data X given a fresh sample Y that is sufficiently similar to X, and P itself does not reveal too much information about X. Such schemes include fuzzy commitment [10], fuzzy vault [11], helper data [12], and secure sketch [13]. Here we follow Dodis et al. and call such public information a *sketch* [13].

A sketch scheme (Fig. 1) consists of two algorithms: A sketch generation algorithm Gen, and a reconstruction Rec. Given some data X, the output $P_X = \mathsf{Gen}(X)$ is called a *sketch* of X. Given a sketch P_X and another Y that is sufficiently similar to X according to some measure, $\mathsf{Rec}(P_X, Y)$ would reconstruct the original X. When applying such a sketch in biometric authentication systems, a *strong extractor* (such as pair-wise independent hash functions) can be further applied on the original X to obtain a key K that is robust, in the sense that it can be consistently reproduced given any Y

Yagiz Sutcu is with Electrical & Computer Engineering Department, Polytechnic University. Email: yagiz@isis.poly.edu.

Qiming Li is with Computer & Information Science Department, Polytechnic University. Email: qiming.li@ieee.org.

Nasir Memon is with Computer & Information Science Department, Polytechnic University. Email: memon@poly.edu.

that is similar to X. This key can then be used in the same way as passwords. For instance, in the context of authentication, a one-way hash function h can be applied on K, and only the hash value $h(K)$ and the sketch P_X are stored in the system.

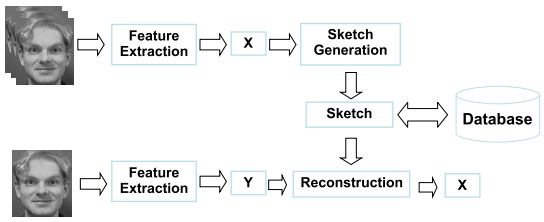

Fig. 1. Sketch Generation and Template Reconstruction

The secure sketch approach is similar in certain ways to both robust hashing and similarity-preserving transformations. On one hand, like a robust hash, a sketch allows exact recovery of the original X, hence the exact key or hash value for authentication. On the other hand, like a similarity-preserving transformation, we would need some extra data associated with each user to guide the authentication process. One may compare a sketch P of X with a syndrome of X w.r.t. some error-correcting code, such that, X can be computed from P and some Y that is close to X [14]. In general, however, constructing a sketch using an error-correcting code in such a straightforward manner may not be feasible or sufficiently secure for real biometric data.

We emphasize that the reconstruction of the original biometrics should be done only locally by the user, so that the reconstructed X is never transmitted and is stored only temporarily. Also, the strong extractors and the hash functions can be randomly chosen for each user at each enrollment, such that even the same biometric data would generate different keys and hash values during multiple enrollments, which further protect the privacy of the users against certain data mining techniques such as database cross-matching.

An important security requirement for sketches is that they should not reveal too much information about the original biometric template X. In the formal framework due to Dodis et al. [13], *min-entropy* is used as the measure of the strength of the key, and *entropy loss* is used as the measure of the advantage a sketch P_X gives to the attacker in guessing X. In this setting, the entropy loss can be conveniently bounded by the size of the sketch. It is worth to note that the entropy loss is a worst case bound for *all* distributions of X.

There are a few difficulties in applying their techniques to biometric templates in the real world. Most importantly, many biometric templates are not discrete, but are instead points in continuous domains (e.g., real numbers resulted from some signal processing techniques). In this case, it would be hard to define what the min-entropy of the original biometric template should be. Furthermore, to extract a discrete key from such a template, some kind of quantization would be necessary. However, since the formulation of secure sketch requires that the original X can be reconstructed exactly, the entropy loss could be arbitrarily high, which can be misleading. For example, consider the quantization of a random variable X uniformly distributed in $[0, 1)$, where any $Y \in (X - 0.01, X + 0.01)$ is considered as "similar" to X. Suppose we apply the sketch scheme in [15] with different quantization steps. If the quantization step is 0.01, the entropy loss after the quantization would be $(\log 3)$, and if we use a quantization step of 0.001, the entropy loss after the quantization would be $(\log 21)$. However, it is not difficult to show that a quantization step of 0.001 leads to a stronger key given that X is uniformly distributed.

Furthermore, even if the biometric templates are represented in discrete forms, existing theoretical results can be either impractical or not applicable. For example, an iris pattern can be represented by a 2048 bit string called *iris code*, and up to 20% of the bits could be flipped during measurements [16]. The fuzzy commitment scheme [10] seems to be applicable at first, but it would be impractical to apply a binary error-correcting code for such long strings with such a high error rate. A two-level error-correcting technique is proposed in [16], which essentially changes the similarity measure such that the space is no longer a metric space.

Minutiae-based fingerprint authentication is another example where the similarity measure for the templates does not define a metric space. In particular, the minutiae of a fingerprint is a set of points in 2-D space, and two sets of minutiae are considered as similar if more than a certain number of minutiae in one set are near distinct minutiae in the other. In this case, the similarity measure has to consider both Euclidean distance and set difference at the same time.

The construction of a secure sketch for point sets [17] is perhaps the first rigorous approach to similarity measures that do not define a metric space. While the schemes proposed in [17] are potentially applicable to minutiae-based fingerprint authentication, other types of biometrics are different both in representations and similarity measures, thus require different considerations.

In a recent work, we further consider the problem of designing and analyzing secure sketch for biometric templates in continuous domains [15]. In [15], we mainly study how to design and analyze different quantization algorithms. Since it is very difficult to have a general algorithm to find the "optimal" quantizer, we instead examine the *relative entropy loss* for any given class of quantizers, which, for any given quantizer in that class, measures the number of additional bits we could have extracted if the optimal quantizer was used in the first place. If we use the quantization example earlier, we would be able to claim that although using a quantization step of 0.01 may not yield the strongest key, but the strength is at most $\log 3$ bits less than the strongest (for all distributions of X). We use the notion of relative entropy loss together with entropy loss to measure the security of the scheme.

In this paper, we identify several important practical issues involved in the design and analysis of secure sketch for biometric templates. Besides the subtleties in the entropy loss due to quantization, a very important aspect of any biometric authentication system is its false accept rate (FAR) and false reject rate (FRR), which are often overlooked in previous theoretical work on secure sketch.

In fact, the use of FAR (with a fixed FRR) as the measure

of security in a biometric authentication system is not new (e.g., [18]). This is the correct measure when the storage of template is secure and the attacker only uses the biometric data of a random user. However, min-entropy would be a better measure when smart attackers are considered. For example, let us consider an extreme case where there are only two users in the system, one of them has an X_1 that is always 0.1, and the other has an X_2 that is always 0.7 (i.e., no error in the measurements). In this case, the min-entropy of the biometric data is 1 bit, which correctly reflects the fact that a smart attacker who knows exactly the distribution of the biometrics can succeed with probability at least 0.5. At the same time, the FAR of the system is 0, which does not tell us anything about how difficult it is to attack the system.

Although secure sketches may have some nice properties that would allow us to handle all attackers and all biometric distributions, using min-entropy and entropy loss alone may not be sufficient to measure the security. In many cases, although the entropy loss can be proved, the min-entropy of the original data X cannot be easily determined, hence making it difficult to conclude the key strength of the resulting system. Even the min-entropy of X can be fixed in some way, the entropy loss may be too large to be useful and it can be misleading. Therefore, cautions have to be taken when analyzing the security of biometric authentication schemes that employs secure sketches.

In this paper we follow the same setting in [15] and consider biometric templates that can be represented as sequences of points in continuous domains, and two sequences are considered as close if sufficiently many points in on sequence are close to the corresponding point in the other sequence. In particular, we examine face biometrics represented by singular values with randomization. Similar to [15], we consider the general 2-step approach where we quantize the data into discrete domains first, and then apply a known secure sketch scheme in discrete domains.

We present a general framework to design and analyze biometric protection schemes using secure sketch, focusing on the trade-off among various parameters. We observe that certain randomization techniques can be applied to achieve better performance in terms of FAR and FRR. However, at the same time, these techniques would make it harder to bound the entropy loss of the sketch. We further estimate the min-entropy of the data in the quantized domain and analyze the key strength in the resulting system. We observe that in some cases theoretical upper bounds on information leakage (i.e., entropy loss) can be too large to be useful, and the exact security of the system needs to be further investigated.

It is worth to note that we are not trying to develop a facial recognition or authentication technique that gives the best FAR and FRR possible. Instead, we study a rather simple scheme with reasonable performance in a controlled environment, and focus on the analysis of the effect of applying the secure sketch scheme on top of the signal processing techniques. Furthermore, we assume that the input is a vector of a fixed length, where all components are independent. For different techniques with other types of features and/or similarity measures, the construction of the sketches as well as the actual security analysis would need to be adapted accordingly.

In many practical systems, a single-factor authentication system (i.e., one that uses only biometrics) may not be sufficient. We discuss the design of multi-factor authentication systems. In such systems, a user would be required not only to produce a correct sample of certain biometrics, but also a correct password and/or a smartcard is required. With the help of secure sketch, it is possible to have a system that is simple, secure, and the user credentials can be easily revoked or replaced.

We will give a review of related work in Section II, followed by some preliminary formal definitions in Section III. We give a concrete secure sketch scheme for face biometrics in Section IV. We further analyze the security and performance of the scheme using real face image data in Section V. We discuss multi-factor authentication schemes in Section VI.

II. Related Work

The construction of secure sketches largely depends on the representation of the biometric templates and the underlying similarity measure. Most of the known techniques assume that the noisy data under consideration are represented as points in some metric space. The fuzzy commitment scheme [10], which is based on binary error-correcting codes, considers binary strings where the similarity is measured by Hamming distance. The fuzzy vault scheme [11] considers sets of elements in a finite field with set difference as the distance function, and corrects errors by polynomial interpolation. Dodis et al. [13] further gives the notion of *fuzzy extractors*, where a "strong extractor" (such as pair-wise independent hash functions) is applied after the original X is reconstructed to obtain an almost uniform key. Constructions and rigorous analysis of secure sketch are given in [13] for three metrics: Hamming distance, set difference and edit distance. Secure sketch schemes for point sets in [17] are motivated by the typical similarity measure used for fingerprints, where each template consists of a set of points in 2-D space, and the similarity measure does not define a metric space. The problem of designing secure sketch for continuous data is first studied in [15], and a notion of relative entropy loss is proposed to measure the quality of a given quantization strategy.

On the other hand, there have been a number of papers on how to extract consistent keys from real biometric templates, some of which may have quite different representations and similarity measures from the above theoretical work. Such biometric templates include handwritten online signatures [19], fingerprints [20], iris patterns [16], voice features [21], and face biometrics [2]. These methods, however, are not accompanied with sufficiently rigorous treatment of the security, compared to well-established cryptographic techniques. Some of the works give analysis on the entropy of the biometrics, and approximated amount of efforts required by a brute-force attacker.

Boyen [22] shows that a sketch scheme that is provably secure may be insecure when multiple sketches of the same biometric data are obtained. Boyen et al. further study the security of secure sketch schemes under more general attacker

models in [23], and techniques to achieve mutual authentication are proposed.

Linnartz and Tuyls [24] consider a similar problem for biometric authentication applications. They consider zero mean i.i.d. jointly Gaussian random vectors as biometric templates, and use mutual information as the measure of security against dishonest verifiers. Tuyls and Goseling [12] consider a similar notion of security, and develop some general results when the distribution of the original is known and the verifier can be trusted. Some practical results along this line also appear in [25].

The concept of cancelable biometrics was first introduced by Ratha et al. [6] (also see [8], [9]). The underlying idea is to apply a user-specific similarity-preserving transformation to biometric templates before they are stored in the database. New biometric samples are transformed in the same way before they are matched with the templates. Hence, the templates can be easily revoked by applying other (random) transformations. The security of these schemes, given that some templates are compromised, relies on the difficulty to *invert* the transformation to obtain the original biometric data. Although it is believed that such transformations are difficult to invert, it seems difficult to rigorously prove the actual one-wayness.

In addition to the above, there are many other approaches which address similar problems. Threshold-based biometric hashing methods for faces, fingerprints and palmprints, are proposed in [3], [4], [26]. The idea of *BioHashing* is further developed in [27], [28], which is mainly for multi-factor authentications. In [5], a non-invertible quantization and ECC based method for creating renewable binary face templates is proposed. As noted by the authors, this technique may not be feasible in practice due to large error correcting capability requirements.

Tulyakov et al. [29] proposed a set of symmetric hash functions and Ang et al. [7] proposed a key-based geometric transformation for minutiae based fingerprint templates. Vielhauer et al. [1] proposed a simple method to calculate biometric hash values using statistical features of online signatures. A key binding algorithm is proposed by Soutar et al. [30] and a face recognition scheme based on minimum average correlation energy filters is proposed by Savvides et al. [31].

III. Preliminaries

A. Entropy and Entropy Loss in Discrete Domain

In the case where X is discrete, we follow the definitions by Dodis et al. [13]. They consider a variant of the *average min-entropy* of X given P, which is essentially the minimum strength of the key that can be consistently extracted from X when P is made public.

In particular, the min-entropy $\mathbf{H}_\infty(A)$ of a discrete random variable A is defined as $\mathbf{H}_\infty(A) = -\log(\max_a \Pr[A = a])$. For two discrete random variables A and B, the average min-entropy of A given B is defined as $\widetilde{\mathbf{H}}_\infty(A \mid B) = -\log(\mathbb{E}_{b \leftarrow B}[2^{-\mathbf{H}_\infty(A|B=b)}])$.

For discrete X, the entropy loss of the sketch P is defined as $\mathcal{L} = \mathbf{H}_\infty(X) - \widetilde{\mathbf{H}}_\infty(X|P)$. This definition is useful in the analysis, since for any ℓ-bit string B, we have $\widetilde{\mathbf{H}}_\infty(A \mid B) \geq \mathbf{H}_\infty(A) - \ell$. For any secure sketch scheme for discrete X, let R be the randomness invested in constructing the sketch, it is not difficult to show that when R can be computed from X and P, we have $\mathcal{L} = \mathbf{H}_\infty(X) - \widetilde{\mathbf{H}}_\infty(X \mid P) \leq |P| - \mathbf{H}_\infty(R)$.

In other words, the entropy loss can be bounded from above by the difference between the size of P and the amount of randomness we invested in computing P. This allows us to conveniently find an upper bound of $\mathcal{L}$ for any distribution of X, since it is independent of X.

B. Secure Sketch in Discrete Domain

Our definitions of secure sketch and entropy loss in the discrete domain follow that in [13]. Let $\mathcal{M}$ be a finite set of points with a *similarity* relation $\mathsf{S} \subseteq \mathcal{M} \times \mathcal{M}$. When $(X, Y) \in \mathsf{S}$, we say the Y is similar to X, or the pair (X, Y) is similar.

DEFINITION 1 *A sketch scheme in discrete domain is a tuple* $(\mathcal{M}, \mathsf{S}, \mathsf{Enc}, \mathsf{Dec})$, *where* $\mathsf{Enc} : \mathcal{M} \to \{0,1\}^*$ *is an encoder and* $\mathsf{Dec} : \mathcal{M} \times \{0,1\}^* \to \mathcal{M}$ *is a decoder such that for all* $X, Y \in \mathcal{M}$, $\mathsf{Dec}(Y, \mathsf{Enc}(X)) = X$ *if* $(X, Y) \in \mathsf{S}$. *The string* $P = \mathsf{Enc}(X)$ *is the sketch, and is to be made public. We say that the scheme is* $\mathcal{L}$*-secure if for all random variables* X *over* $\mathcal{M}$, *the entropy loss of the sketch* P *is at most* $\mathcal{L}$. *That is,* $\mathbf{H}_\infty(X) - \widetilde{\mathbf{H}}_\infty(X \mid \mathsf{Enc}(X)) \leq \mathcal{L}$.

We call $\widetilde{\mathbf{H}}_\infty(X \mid P)$ the *left-over entropy*, which in essence measures the "strength" of the key that can be extracted from X given that P is made public. Note that in most cases, the ultimate goal is to maximize the left-over entropy for some particular distribution of X. However, in the discrete case, the min-entropy of X is fixed but can be difficult to analyze. Hence, entropy loss becomes an equivalent measure which is easier to quantify.

C. Secure Sketch in Continuous Domain

To handle points in some continuous domain $\mathcal{U}$, we follow [15] and use a two-step approach. In particular, we quantize (discretize) the points such that they become points in a discrete domain $\mathcal{M}$. After that we apply known sketch scheme in discrete domain $\mathcal{M}$ to construct the sketch. When a fresh measurement of the same biometrics is given, it is quantized using the same quantizer and the corresponding reconstruction algorithm in the discrete domain is used to recover the quantized version of the original data points.

More formally, let $\mathcal{U}$ be a set that may be uncountable, and let S be a similarity relation on $\mathcal{U}$, i.e., $\mathsf{S} \subseteq \mathcal{U} \times \mathcal{U}$. Let $\mathcal{M}$ be a set of finite points, and let $\mathcal{Q} : \mathcal{U} \to \mathcal{M}$ be a function that maps points in $\mathcal{U}$ to points in $\mathcal{M}$. We will refer to such a function $\mathcal{Q}$ as a *quantizer*.

DEFINITION 2 *A quantization-based sketch scheme is a tuple* $(\mathcal{U}, \mathsf{S}, \mathcal{Q}, \mathcal{M}, \mathsf{Enc}, \mathsf{Dec})$, *where* $\mathsf{Enc} : \mathcal{M} \to \{0,1\}^*$ *is an encoder and* $\mathsf{Dec} : \mathcal{M} \times \{0,1\}^* \to \mathcal{M}$ *is an decoder such that for all* $X, Y \in \mathcal{U}$, $\mathsf{Dec}(\mathcal{Q}(Y), \mathsf{Enc}(\mathcal{Q}(X))) = \mathcal{Q}(X)$ *if* $(X, Y) \in \mathsf{S}$. *The string* $P = \mathsf{Enc}(\mathcal{Q}(X))$ *is the sketch. We say that the scheme is* $\mathcal{L}$*-secure in the quantized domain if for all*

random variable X over $\mathcal{U}$, the entropy loss of P is at most $\mathcal{L}$, i.e., $\mathbf{H}_\infty(\mathcal{Q}(X)) - \widetilde{\mathbf{H}}_\infty(\mathcal{Q}(X) \mid \mathsf{Enc}(\mathcal{Q}(X))) \leq \mathcal{L}$

It is worth to note that according to this definition, we only require the quantized original to be reconstructed. This, in some sense, avoids the problem of possible high entropy loss due to quantization. It is shown in [15] that when the quantization step (assuming scalar quantization) is close to the error that we want to tolerate, the resulting scheme would not be too much different in terms of left-over entropy from using the "optimal" quantization step, which may be difficult to find. Therefore, in this paper we will follow this principle, with some necessary deviation due to be nature of the biometrics in the real world.

D. A General Scheme

Here we give a construction of sketch for biometric data that can be represented as real vectors of a fixed length n, based on the general scheme in [15]. Our scheme differs from that in [15] in that, the scheme in [15] assumes the same error-tolerance for all components and for all users, whereas in our scheme, we allow the error-tolerance to be different for each component in a vector of the same user, and for the same i-th component for different users. This setting is more general, and in fact it is necessary for the data set we use in the experiments.

We assume that a template can be written as $X = [x_1 \ \cdots \ x_n]^T$. For each component x_i, there is a parameter δ_i, and another vector $Y = [y_1 \ \cdots \ y_n]^T$ is considered as similar to X if $|y_i - x_i| \leq \delta_i$ for all i.

The sketch scheme consists of several building blocks: A quantizer, a codebook, an encoder and a decoder. A class of schemes can be defined as below with parameters $\lambda_1, \lambda_2, \cdots, \lambda_n$.

a) Quantizer $\mathcal{Q}_{\lambda_i}$: We define $\mathcal{Q}_{\lambda_i}$ (for each i) as a scalar quantizer with step size $\lambda_i \in \mathbb{R}$. For each $x \in \mathcal{U}$, $\mathcal{Q}_{\lambda_i}(x) = \widehat{x}$ if and only if $\lambda_i \widehat{x} \leq x < \lambda_i(\widehat{x}+1)$, and the quantization of X is defined as $\widehat{X} = \mathcal{Q}(X) \triangleq [\mathcal{Q}_{\lambda_1}(x_1) \ \cdots \ \mathcal{Q}_{\lambda_n}(x_n)]^T$. The corresponding quantized domain is thus $\mathcal{M}_{\lambda_i} = [0, \lceil \frac{1}{\lambda_i} \rceil]^n$. The encoders and the decoders work only on the quantized domain. Let $\widehat{\delta}_{\lambda_i} = \lceil \delta_i / \lambda_i \rceil$. Under noise, a point $\widehat{x}$ in the quantized domain can be shifted by a distance of at most $\widehat{\delta}_{\lambda_i}$. Let us denote $\Delta_{\lambda_i} \triangleq 2\widehat{\delta}_{\lambda_i} + 1$.

b) Codebook $\mathcal{C}_{\lambda_i}$: For each quantized domain $\mathcal{M}_{\lambda_i}$, we consider a *codebook* $\mathcal{C}_{\lambda_i}$, where every codeword $c \in \mathcal{C}_{\lambda_i}$ has the form $c = \beta \Delta_{\lambda_i}$ for some non-negative integer β. We use $\mathcal{C}_{\lambda_i}(\cdot)$ to denote the function where given a quantized point $\widehat{x}$, it returns a value $c = \mathcal{C}_{\lambda_i}(\widehat{x})$ such that $|\widehat{x} - c| \leq \widehat{\delta}_{\lambda_i}$. That is, the functions finds the unique codeword c that is nearest to $\widehat{x}$ in the codebook.

c) Encoder Enc*:* Given a quantized $\widehat{X}$, the encoder Enc does the following.

1) For each $\widehat{x}_i \in \widehat{X}$, compute $c_i = \mathcal{C}_{\lambda_i}(\widehat{x}_i)$;
2) Output $P = \mathsf{Enc}(\widehat{X}) = [d_1 \ \cdots \ d_n]^T$, where $d_i = \widehat{x}_i - c_i$ for $1 \leq i \leq n$.

In other words, for every $\widehat{x}_i$, the encoder outputs the distance of $\widehat{x}_i$ from its nearest codeword in the codebook $\mathcal{C}_{\lambda_i}$.

d) Decoder Dec*:* For a fresh measurement Y, it is first quantized by $\widehat{Y} = \mathcal{Q}(Y)$. Given $P = [d_1 \ \cdots \ d_n]^T$ and $\widehat{Y} = [\widehat{y}_1 \ \cdots \ \widehat{y}_n]^T$, and the decoder Dec does the following.

1) For each $\widehat{y}_i \in \widehat{Y}$, compute $c_i = \mathcal{C}_{\lambda_i}(\widehat{y}_i - d_i)$;
2) Output $\widehat{X} = \mathsf{Dec}(\widehat{Y}) = [c_1 + d_1 \ \cdots \ c_n + d_n]^T$.

In other words, the decoder shifts every $\widehat{y}_i$ by d_i, maps it to the nearest codeword in $\mathcal{C}_{\lambda_i}$, and shifts it back by the same distance.

e) Entropy loss: Using an analysis that is very similar to that in [15], it is not difficult to show that the entropy loss for the above scheme in the quantized domain is the total size of the sketch, which is $\sum_{i=1}^{n} \log \Delta_{\lambda_i}$. The relative entropy loss of the scheme is also $\sum_{i=1}^{n} \log \Delta_{\lambda_i}$. When $\lambda_i = \delta_i$ for all i, both the entropy loss and the relative entropy loss would be $n \log 3$, which agree with the results in [15].

IV. Sketch of Face Biometrics

In this section, we describe our scheme to compute sketches from face images that allow us to extract consistent keys. Our main idea is as the following. For a given image, we first extract a feature vector V of size n (Section IV-A). Next, we apply a randomization on the n component to obtain a randomized feature vector W of size k (Section IV-B). After that, we discretize (quantize) the new feature vector (Section IV-C). Finally, we apply a known sketch scheme to generate a sketch, or to reconstruct the quantized feature vector (Section IV-D), and analyze its security (Section IV-E).

A. Feature Vector Extraction

We assume that from each biometric sample we can extract a feature vector of size n. Let $V_i = [v_{i1} \ v_{i2} \ ... \ v_{in}]^T$ represent the n-dimensional feature vector of i-th user of the system where each component $v_{ij} \in \mathbb{R}$ is a real number. These components can be extracted from certain transformations on the raw measurement. For example, we can apply singular value decomposition and take the n most significant components (Section V).

During different measurements of the same legitimate user, the value of each component v_{ij} can vary within a certain *range*, which is going to be determined through experiments on the data set. In other words, we consider the j-th component for the i-th user to be always associated with a range, which is defined by a *midpoint* $\overline{v}_{ij}$ and a *size* ρ_{ij}.

In the simplest case, for the i-th user in the system, we can consider a sample $V_i = [v_{i1} \ v_{i2} \ ... \ v_{in}]^T$ as authentic if $\overline{v}_{ij} - \rho_{ij} \leq v_j \leq \overline{v}_{ij} + \rho_{ij}$ for all $j = 1, ..., n$.

In this case, the template for the i-th user can be described by two vectors. The first is the list of n midpoints $\overline{v}_{i1}, \ldots, \overline{v}_{in}$, and the other is the list of range sizes for each of the components $\rho_{i1}, \ldots, \rho_{in}$.

Through experiments, we found that the performance of the system in terms of FAR and FRR can be improved if we perform some *randomization* on the feature vector before creating the templates. In this case, a template would consist of the description of the randomization process, and the midpoints and the range sizes of the components after randomization. This will become clearer in Section IV-B.

B. Randomization

Before generating a sketch from the components extracted from raw samples of biometric data, we can further apply user-specific random mapping on these feature vectors. In particular, we generate k-by-n matrices whose elements are uniformly distributed random numbers between $-\theta$ and θ, where θ is a parameter. We call such matrices *randomization matrices*. Through experiments, we found that the overall performance is not sensitive to the value of θ, so we fix the value of θ to be 1.

Let R_i be the randomization matrix for user i and by multiplying the feature vector with this random matrix, an n dimensional feature vector can be mapped into another k dimensional feature vector. That is, for user i and a raw sample $V_i = [v_{i1} \ \dots \ v_{in}]^T$, we compute $W_i = R_i V_i = [w_{i1} \ w_{i2} \ ... \ w_{ik}]^T$.

Similar to the simple case in Section IV-A, we find midpoints $\overline{w}_{ij}$'s and range sizes δ_{ij}'s and for any $W_i = R_i V_i = [w_{i1} \ w_{i2} \ ... \ w_{ik}]^T$, we consider it as authentic if $\overline{w}_{ij} - \delta_{ij} \leq w_{ij} \leq \overline{w}_{ij} + \delta_{ij}$ for all $j = 1, ..., k$.

The midpoint $\overline{w}_{ij}$ for the j-th component of the i-th user is determined as $\overline{w}_{ij} = (\mathsf{mx}_{ij} + \mathsf{mn}_{ij})/2$, where mn_{ij} (resp. mx_{ij}) is the minimum (resp. the maximum) value of the j^{th} component of the feature vector observed in the training data set of user i. Similarly, the range size δ_{ij} for the i-th component of the i-th user is determined as $\delta_{ij} = (\mathsf{mx}_{ij} - \mathsf{mn}_{ij})/2$.

The main reason of using such a random mapping is better noise tolerance. In particular, the noise on the original components seems to be smoothed out by the random mapping, which makes the scheme more robust for the same FAR.

C. Quantization and Codebook

To discretize the data, we employ a straightforward method, which uses a scalar quantizer for each of the components to map them to a discrete vector.

First, we find the global ranges of each component. Let $\mathsf{MN}_j = \min_i(\mathsf{mn}_{ij})$ and $\mathsf{MX}_j = \max_i(\mathsf{mx}_{ij})$. We also determine the quantization step as $\delta_j = \alpha \min_i(\delta_{ij})$, where $\alpha \in (0, 1]$ is some parameter.

Next, the discrete domain $\mathcal{M}_j$ for the j-th component is computed by quantizing the overall user range by the quantization step δ_j. That is, $\mathcal{M}_j = \{\mathsf{MN}_j - r_j, \mathsf{MN}_j - r_j + \delta_j, ..., \mathsf{MN}_j - r_j + L_j\delta_j\}$ where L_j is appropriately chosen integer which satisfies $\mathsf{MN}_j - r_j + L_j\delta_j \geq \mathsf{MX}_j$ and r_j is a positive random number.

In this way, for the j-th component of the i-th user, a range of midpoint $\overline{w}_{ij}$ and size δ_{ij} can be translated to a discrete range where the discrete midpoint is quantization of $\overline{w}_{ij}$ in $\mathcal{M}_j$, and the discrete range size d_{ij} is given by $d_{ij} = \lceil \frac{\delta_{ij}}{\delta_j} \rceil$.

Finally, the codebook C_{ij} can be constructed using the general algorithm in Section III-D, with parameters $D_{ij} = 2d_{ij} + 1$ and $\beta = D_{ij}$.

D. Sketch Generation and Data Reconstruction

1) Sketch Generation: During enrollment, the biometric data of each user are acquired and feature vectors are extracted (Section IV-A), randomized (Section IV-B), and quantized (Section IV-C).

Following the scheme in Section III-D, the sketch P_i for user i is a vector $P_i = [p_{i1} \ p_{i2} \ ... \ p_{ik}]^T$. For each p_{ij} we have $p_{ij} = Q_{ij}(\overline{w}_{ij}) - \overline{w}_{ij}$, where $Q_{ij}(\overline{w}_{ij})$ is the codeword in C_{ij} that is closest to $\overline{w}_{ij}$.

2) Data Reconstruction: During authentication, biometric data of the i-th user is taken and corresponding feature vector is computed. Let us denote this noisy feature vector as $\widetilde{V}_i = [\widetilde{v}_{i1} \ \widetilde{v}_{i2} \ ... \ \widetilde{v}_{in}]^T$.

After applying the random mapping associated with the given identity, we have $\widetilde{W}_i = R_i \widetilde{V}_i = [\widetilde{w}_{i1} \ \widetilde{w}_{i2} \ ... \ \widetilde{w}_{ik}]^T$. Next, the decoder takes $\widetilde{W}_i$ and P_i and calculates $Q_{ij}(\widetilde{w}_{ij}) - p_{ij}$ for $j = 1, ..., k$. Reconstruction of the original biometric will be successful if $-d_{ij} \leq Q_{ij}(\widetilde{w}_{ij}) - Q_{ij}(\overline{w}_{ij}) < d_{ij}$, where d_{ij} is the user specific error tolerance bound for the j-th component.

It is not difficult to see that, $Q_{ij}(\widetilde{w}_{ij}) - p_{ij} = Q_{ij}(\widetilde{w}_{ij}) - Q_{ij}(\overline{w}_{ij}) + \overline{w}_{ij}$ and the errors up to the some preset threshold value will be corrected successfully.

E. Security

As mentioned earlier, $\widetilde{\mathbf{H}}_\infty(X \mid P)$ is called the *left-over entropy*, which measures the "strength" of the key that can be extracted from X given that P is made public and in most cases, the ultimate goal is to maximize the left-over entropy for some particular distribution of the biometric data considered. However, in the discrete case, the min-entropy is fixed but can be difficult to analyze and entropy loss becomes an equivalent measure which is easier to quantify.

For this construction, in order to estimate the left-over entropy, firstly, we tried to estimate the min-entropy of V ($\mathbf{H}_\infty(V)$). Here we assume that the components of the feature vector (before randomization) are independent, we estimated the min-entropy of each component independently and the total min-entropy of the feature vector V is calculated as the summation of the individual min-entropies of the components. That is, $\mathbf{H}_\infty(V) = \Sigma_{i=1}^{n} \mathbf{H}_\infty(v_i)$.

To estimate $\mathbf{H}_\infty(v_i)$, we considered the distribution of the feature vector component v_i over all user space. In particular, we analyzed the histogram of that distribution while setting the bin size to the quantization step size δ_i of that component and determined the number of elements in the most likely bin. This gives a rough estimate of the min-entropy of the feature vector component v_i.

The (component-wise) entropy loss in the quantized domain can be bounded by $\mathcal{L}(P) \leq \Sigma_{i=1}^{k} \mathcal{L}(p_i)$, where $\mathcal{L}(p_i)$ is the entropy loss of the sketch for the component v_i of the feature vector representation of the biometric data after randomization. This can be conveniently bounded by the size of the sketch. That is, $\mathcal{L}(p_i) \leq |p_i| = log(2\lceil \frac{\delta_{ij}}{\delta_j} \rceil + 1)$.

Note that this estimation of left-over entropy may not be accurate in reality. First of all, a more accurate estimation would require the estimation of min-entropy after randomization. However, feature vectors become dependent after the randomization process and it is not easy to estimate the min-entropy in that case. Second, depending on the size of the randomization matrix employed, dimension of the transformed

feature vector becomes larger and as a result, size of the sketch becomes larger as well. Therefore, entropy loss, which are bounded simply by the size of the sketch, becomes too high and it may not be meaningful.

We note that the "entropy loss" is an upper bound of the information leakage. Whether the attacker can really gain that much information needs to be further studied for particular distributions. In other words, entropy loss is a worst case bound, which states that there exists an input distribution that may give such amount of information leakage, but not necessarily the distribution for the particular biometric data. We consider it an open question to bound the "exact information leakage" of the sketch.

V. Experiments and Analysis

A. Singular Value Decomposition

Due to the ease of capturing and availability of many powerful digital signal processing tools to analyze digital images, face images are one of the widely used biometrics for authentication purposes. As it is the case for many face image based biometric recognition systems proposed in recent years, singular values are used as features [32], [33], [34] and due to established properties of singular values, we also used them for testing our scheme. However, it should be noted that the essence of the technique is not specific to face image data and can be applied to any type of ordered biometric features.

Since SVD is one of the well-known topics of linear algebra, we omitted to give detailed analysis of this subject and the following definitions will be helpful to understand the singular value decomposition and robustness properties of singular values.

Singular Value Decomposition: If the matrix $A \in \mathbb{R}^{m\times n}$ then there exist orthogonal matrices $U \in \mathbb{R}^{m\times m}$ and $V \in \mathbb{R}^{n\times n}$ such that $A = U \times \Sigma \times V^T$ where $\Sigma = diag\{\lambda_1, \lambda_2, ..., \lambda_p\}$ with $\lambda_1 \geq \lambda_2 \geq ... \geq \lambda_p$ and $p = \min(m, n)$.

Perturbation: Let $\widetilde{A} = A + E \in \mathbb{R}^{m\times n}$ be a perturbation of A and let $\widetilde{A} = \widetilde{U} \times \widetilde{\Sigma} \times \widetilde{V}^T$ be singular value decomposition of $\widetilde{A}$, then $|\lambda_i - \widetilde{\lambda}_i| \leq \|E\|_2$ for $i = 1, ..., p$ where $\|E\|_2$ is induced-2 norm of E.

It is also worth mentioning that, in many applications, it is often sufficient (as well as faster, and more economical for storage) to consider the first n singular values. Therefore, in our experiments, we choose $n = 20$. To further justify the choice of the number n, we also examine the variances of the singular values (as shown in Fig. 2), since a random variable with smaller variance would usually have less "distinguishing power". It is easy to observe from the figure that, the variances of the singular values decrease dramatically as the magnitude of the singular values decrease. In fact, many small singular values are exactly the same for many users, which would not be useful for authentication. For our data set, the first 20 coefficients consist of more than 99% of the total mass.

B. Experiment Data Set

In our experiments, we use the Essex Faces 94 face database (E94 database) [35], which is essentially created for face recognition related research studies. Sample images from the E94 database are given in Fig. 3. The database contains still images of 152 distinct subjects, with 20 different images for each subject. The size of each JPEG image is 180x200. We first transformed these JPEG images to 8-bit gray level images and then use only the gray level images in our experiments. For each subject, we randomly divide the 20 samples for the subject into two parts, namely, training and test sets. The training set is assigned 12 of the images, and test set has the remaining 8 sample face images. Therefore, 8 test data for every user is used to generate 152x8=1216 genuine authentication attempts and 151x152x8=183616 impostor authentication attempts (8 attempts by 151 remaining users for every user in the system). As noted earlier, in our simulations, only first 20 singular values of the images are considered.

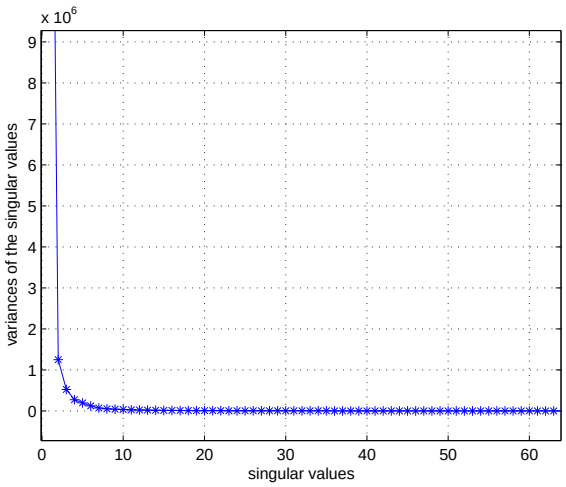

Fig. 2. Variances of the singular values

Fig. 3. Some examples from E94 database

C. FAR and FRR

False accept rate (FAR) and false reject rate (FRR) are important performance parameters of biometric authentication systems. Typically, FAR is defined as the probability that a

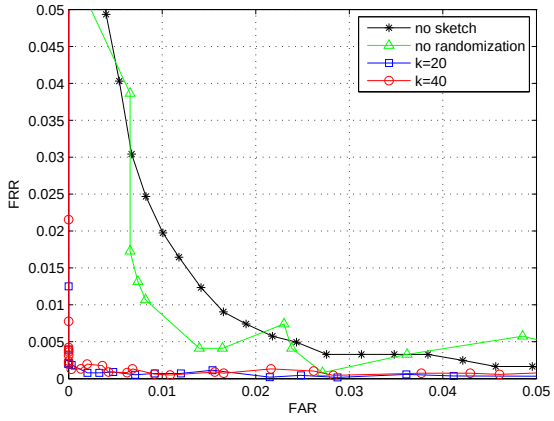

Fig. 4. Comparison of the performance for different values of k

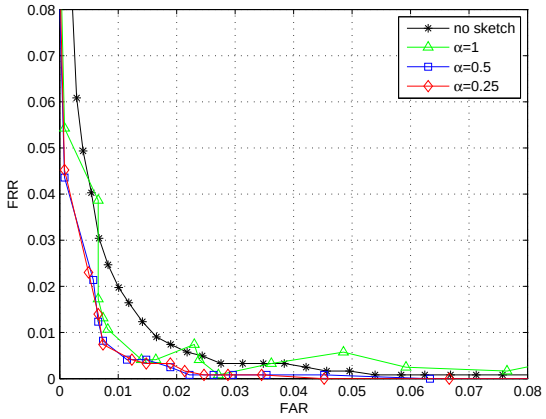

Fig. 5. Comparison of the performance for different values of α

random user in the system is authenticated as another user, and FRR is the probability that a legitimate user is deemed as non-authentic. Usually, FAR and FRR can be traded-off against each other by changing some tunable threshold parameter. Such trade-off can be described using ROC curve of the system.

We conduct experiments using the training and testing data as described earlier. The FAR for a given user is defined as the ratio of the number of testing images of other users that are considered as authentic as that user over the total number of testing images of other users. The overall FAR of the system is then taken as the average of the FAR for every user. The FRR is determined in a similar way. We find the ratio of the number of test images of a legitimate user that are wrongly declared as non-authentic over the total number of test images of the user, and then the average of all users is computed.

This process gives a single point on the ROC curve. To obtain the complete ROC curve, we calculated FAR and FRR values by varying the quantization step size, δ_i. However, it should be noted that, once d_{ij} values are calculated, they are fixed and did not changed during the determination of the ROC curve. In addition, since the random mapping, R_is are different for each realization, it is needed to calculate an average value of the performance metrics over the random choices of R_i's. We evaluated performance metrics over 20 realizations in our experiments to calculate the average.

D. Choosing Parameter k

In the proposed construction, one of the parameters that has an effect on the performance is k, the dimension of the transformed (through random mapping) feature vector. Fig. 4 shows the performance of our scheme for E94 database for different values of k. ROC curve obtained by using only first 20 singular values (without secure sketch) is also provided for illustrating the effect of the quantization-based sketch scheme on the performance. As shown in Fig. 4, the application of secure sketch on top of the singular values does not make significant difference in performance. Furthermore, we can see that random linear transformations improves the performance of the scheme.

However, it can also be seen that increasing the parameter k beyond n has no significant effect on the performance. Although it is very difficult to analyze the effect of increasing k on the the performance of the scheme analytically, this may be explained by the following. By taking the random linear combination of the feature vector components, we inject some amount of redundancy into the randomized feature vectors. When k is large, the redundancy becomes so high that it does not give any additional "distinguishing power". Furthermore, as we will see in Section V-F, it is possible that a choice of $k > n$ can make the entropy loss too large to be meaningful.

On the other hand, choosing k small than n is not advisable. During the experiments, it is found that the number of bits that can be extracted from one coefficient after randomization remains roughly the same regardless the value of k. Hence, the number of bits that can be extracted from the transformed feature vector would be less for smaller k. Intuitively, by choosing a $k < n$, some information about the original feature vector is lost. Such information loss would result in weaker keys. Therefore, $k = n$ seems a reasonable choice for practical systems.

E. Effects of Quantization Steps

Another parameter needs to be considered is the quantization step size δ_j. As already mentioned earlier, the quantization step δ_j can be determined in many different ways depending on operational constraints (such as the noise level which needs to be tolerated) and also depending on the data set considered. Here, we considered a straightforward approach and set the quantization step to be a fraction of the minimum range observed over the whole data set (i.e., $\delta_j = \alpha \min_i(\delta_{ij})$).

Fig. 5 shows the effect of α on the performance of the scheme for 3 different values of α. It should be mentioned that to observe the effects of the parameter α on the performance of the scheme separately, no randomization is involved here. As can be seen from Fig. 5, small values of α seem to improve the performance of the scheme. However, it is easy to observe that decreasing α to below 0.5 has no significant effect on the performance. In addition, it is worth noting that the overall effect of α is not as significant as k.

F. Min-Entropy and Entropy Loss

As mentioned in Section IV, we estimate the min-entropy of the original biometrics by computing the min-entropy of the quantized feature vectors, assuming that all components are independent (i.e., no randomization), and using quantization steps that are equal the minimum errors to be tolerated for each component (i.e., $\alpha = 1$). Under this setting, the min-entropy of the feature vectors is estimated to be about 85 bits.

On the other hand, the entropy loss of the scheme can be calculated by the size of the sketch. Since the errors to be tolerated are quite different for different users even for the same component, the resulting entropy loss is much larger than the theoretically achievable $n \log 3$, when the error-tolerance is the same for all components. From the experiments, the average size of the sketch when $k = 20$ is about 65 bits, which gives a guarantee of 20 bits in the left-over entropy. When k increases, the size of sketch (and hence the entropy loss) increases proportionally. When $k = 30$, the size of the sketch would be larger than 85 bits, and the entropy loss calculated from the size of sketch becomes meaningless.

Since, as we have observed earlier, having a k larger than $n = 20$ would not improve the performance significantly, we would recommend choosing $k = n$ for a practical system with certain guarantee of security from theoretical bounds. It is true, however, that 20 bits of security still looks weak. Nevertheless, as we noted earlier, this is just a lower bound for the left-over entropy, and the exact security requires further investigation.

Another possible way to obtain better left-over entropy is to reduce the value of α. As we observed earlier, this would not gain much advantage in performance. Furthermore, having a smaller α might actually decrease the left-over entropy. This can happen for certain distribution of X where decreasing α does not increase the min-entropy of quantized X, but increases the information leakage (because of the now larger sketch size). Therefore, we would recommend using $\alpha = 1$.

VI. Multi-Factor Authentication

As we mentioned earlier, with the help of a sketch, a user can reconstruct the original biometric data during registration, and hence can use the data as a password or key in other cryptographic schemes that do not tolerate errors.

However, two problems still remain. First, the entropy (or length) of such a key extracted from biometric data may not be large enough compared to the standard practice in widely used cryptographic schemes. Secondly, if such a key is revealed accidentally, it would reveal important information about the original biometric data. This could happen when the key is used in some cryptographic schemes that can be attacked offline. For example, the key may be used to encrypt a file, and when an attacker gets a copy of the encrypted file, he/she can launch an offline attack to decrypt the file. Although it may take a long time to decrypt it and the content of the file may become useless, the discovery of the key would give the attacker information about the biometric data, which could be in turn used to attack newly encrypted files using keys derived from the same or similar biometric data.

These two problems can be both tackled using a multi-factor authentication scheme. In such a scheme, a user is required not only to produce a "correct" sample of the biometrics, but also a smartcard and/or password that match the identity claimed by the user. Clearly, there are a few different types of secrets that the user may be required to produce, and each of them have their own strengths and weaknesses that have to be taken into consideration. For example, a key stored in a smartcard can be almost arbitrarily long, and can be easily made uniformly distributed. However, such keys has to be stored somewhere, which makes it easier to be compromised. Passwords have lower entropy but can be completely stored in our brains. These two types of secrets have the advantage that they can be easily replaced and/or revoked. Biometrics, on the other hand, have entropy higher than passwords but cannot be made arbitrarily long, and they are difficult to revoke or replace.

Here we describe a simple multi-factor scheme using biometrics and smartcards, and other factors can easily fit in using the same principle. Suppose a user has biometric data X and a smart card with a key K of length n. We further assume that there is a cryptographic pseudo-random number generator G that takes a short seed S and outputs pseudo-random bits that cannot be efficiently distinguished from random bits. During registration, the user computes the hash of X and uses it as the seed S (i.e., $S = h(X)$ for some cryptographic hash function h), then applies $G(S)$ to generate n pseudo-random bits. Let $K_p = G(S)$ be the output. Next, the user computes a sketch P_X from X, and chooses a random string Q, where $|Q| = |P_X|$. The string Q is stored in the authentication server, and the result of $Q \oplus P_X$ is stored in the smartcard, where $\oplus$ denotes bit-wise XOR operation. Also, the result of $K \oplus K_p$ is also stored in the authentication database. The use of pseudo-random number generator allows the string K_p to be of any polynomial length, so that it can be easily xor'ed with K.

During authentication, the server retrieves $Q \oplus P_X$ from the smartcard, and uses it to recover P_X, which is then returned to the user. Next the user reconstructs X using P_X and a fresh scan of the biometrics, and applies the same function $G(h(X))$ to recover K_p. After that the user would be able to generate the key $K \oplus K_p$ for authentication.

In this way, if the authentication database is compromised, only Q and $K \oplus K_p$ is revealed. Since K are completely random, so is $K \oplus K_p$. Hence the data stored at the server does not reveal any information about X. On the other hand, if the smartcard is stolen or lost, what an attacker would be able to find out is $Q \oplus P_X$ and K, which are just random strings. Since K and Q are independent from the user password and biometrics, they can be easily revoked and replaced.

In the worst case, the attacker is able to steal the smartcard and compromise the server at the same time. In that case, P_X and K_p would be revealed. However, P_X reveals only limited information about X, and it can be computationally infeasible to compute X from K_p, if the min-entropy of X is high enough. Other secrets (e.g., passwords) can be used in combination with X to make it harder to compute X from K_p. Therefore, we can achieve unconditional security when one of the storage (database and smartcard) is compromised, and some extent of computational security when both storage devices are compromised.

VII. Conclusions

In this paper we study the problem of secure storage of biometric templates in biometric-based authentication systems. We examine a recently proposed cryptographic primitive called secure sketch and identify several practical issues when we apply known theoretical results to real biometrics.

In particular, we note that the security measure in terms of entropy loss may not be sufficient since FAR and FRR should also be taken into consideration of a practical system. Furthermore, entropy loss alone could be just too large to be meaningful, or sometimes becomes misleading, especially when the original biometrics are represented in continuous domains.

We give a concrete construction of secure sketch for face biometrics, and we illustrate the subtleties and difficulties in applying theoretical bounds. We show various trade-offs among different parameters of the system. It seems that, at least in some cases, the exact security of the system needs to be further investigated, and known theoretical results become not very useful.

We also consider the multi-factor setting where multiple secrets (including biometrics) are used together for authentication. We give a simple multi-factor scheme using a sketch and a smartcard.

We consider it as a challenging open problem to find a general and accurate way to compute the min-entropy (or any quantitative means that measures the success probability of smart attackers) of biometric data, and to determine the exact information leakage of the sketches.

Acknowledgment

The authors would like to thank anonymous reviewers for their valuable comments and suggestions.

References

[1] C. Vielhauer, R. Steinmetz, and A. Mayerhoefer, "Biometric hash based on statistical features of online signatures," *IEEE International Conference on Pattern Recognition (ICPR)*, 2002.

[2] Y. Sutcu, T. Sencar, and N. Memon, "A secure biometric authentication scheme based on robust hashing," in *ACM MM-SEC Workshop*, 2005.

[3] T. Connie, A. Teoh, M. Goh, and D. Ngo, "Palmhashing: a novel approach for cancelable biometrics," *Information Processing Letters*, vol. 93, pp. 614–634, 2005.

[4] A. Teoh, D. Ngo, and A. Goh, "Personalised cryptographic key generation based on facehashing," *Computers and Security*, vol. 23, pp. 606–614, 2004.

[5] T. Kevenaar, G. Schrijen, M. V. der Veen, A. Akkermans, and F. Zuo, "Face recognition with renewable and privacy preserving binary templates," *Fourth IEEE Workshop on Automatic Identification Advanced Technologies*, pp. 21–26, 2005.

[6] N. Ratha, J. Connell, and R. Bolle, "Enhancing security and privacy in biometrics-based authentication systems," *IBM Systems Journal*, vol. 40, no. 3, pp. 614–634, 2001.

[7] R. Ang, R. Safavi-Naini, and L. McAven, "Cancelable key-based fingerprint templates," in *ACISP*, ser. LNCS, vol. 3574, 2005, pp. 242–252.

[8] N. Ratha, J. Connell, R. Bolle, and S. Chikkerur, "Cancelable biometrics: A case study in fingerprints," in *Intl. Conf. on Pattern Recognition*, 2006, pp. 370–373.

[9] 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.

[10] A. Juels and M. Wattenberg, "A fuzzy commitment scheme," in *Proc. ACM Conf. on Computer and Communications Security*, 1999, pp. 28–36.

[11] A. Juels and M. Sudan, "A fuzzy vault scheme," in *IEEE Intl. Symp. on Information Theory*, 2002.

[12] P. Tuyls and J. Goseling, "Capacity and examples of template-protecting biometric authentication systems," in *ECCV Workshop BioAW*, 2004, pp. 158–170.

[13] Y. Dodis, L. Reyzin, and A. Smith, "Fuzzy extractors: How to generate strong keys from biometrics and other noisy data," in *Eurocrypt*, ser. LNCS, vol. 3027. Springer-Verlag, 2004, pp. 523–540.

[14] T. Boult, "Robust distance measures for face-recognition supporting revocable biometric tokens," in *IEEE, 7th Intl. Conf. on Automatic Face and Gesture Recognition*, 2006, pp. 560–566.

[15] Q. Li, Y. Sutcu, and N. Memon, "Secure sketch for biometric templates," in *Asiacrypt*, Shanghai, China, December 2006.

[16] F. Hao, R. Anderson, and J. Daugman, "Combining cryptography with biometrics effectively," University of Cambridge, Tech. Rep. UCAM-CL-TR-640, 2005.

[17] E.-C. Chang and Q. Li, "Hiding secret points amidst chaff," in *Eurocrypt*, 2006.

[18] L. O'Gorman, "Comparing passwords, tokens, and biometrics for user authentication," *Proceedings of the IEEE*, vol. 91, no. 12, pp. 2021–2040, December 2003.

[19] F. Hao and C. Chan, "Private key generation from on-line handwritten signatures," *Information Management and Computer Security*, vol. 10, no. 2, 2002.

[20] S. Yang and I. Verbauwhede, "Automatic secure fingerprint verification system based on fuzzy vault scheme," in *IEEE Intl. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, 2005, pp. 609–612.

[21] F. Monrose, M. Reiter, Q. Li, and S. Wetzel, "Cryptographic key generation from voice," in *IEEE Symp. on Security and Privacy*, 2001.

[22] X. Boyen, "Reusable cryptographic fuzzy extractors," in *Proceedings of the 11th ACM conference on Computer and Communications Security*. ACM Press, 2004, pp. 82–91.

[23] X. Boyen, Y. Dodis, J. Katz, R. Ostrovsky, and A. Smith, "Secure remote authentication using biometric data," in *Eurocrypt*, 2005.

[24] J.-P. M. G. Linnartz and P. Tuyls, "New shielding functions to enhance privacy and prevent misuse of biometric templates," in *AVBPA 2003*, 2003, pp. 393–402.

[25] P. Tuyls, A. Akkermans, T. Kevenaar, G. Schrijen, A. Bazen, and R. Veldhuis, "Practical biometric authentication with template protection," in *AVBPA*, 2005, pp. 436–446.

[26] A. B. Teoh and D. C. Ngo, "Cancellable biometerics featuring with tokenized random number," *Pattern Recognition Letters*, vol. 26, pp. 1454–1460, 2005.

[27] A. Teoh, T. Connie, and D. Ngo, "Remarks on BioHash and its mathematical foundation," *Information Processing Letters*, vol. 100, pp. 145–150, 2006.

[28] A. Teoh, A. Gho, and D. Ngo, "Random multispace 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.

[29] S. Tulyakov, F. Farooq, and V. Govindaraju, "Symmetric hash functions for fingerprint minutiae," in *Lecture Notes in Computer Science*, ser. LNCS, 2005.

[30] C. Soutar, D. Roberge, S. Stojanov, R. Gilroy, and B. V. Kumar, "Biometric encryption using image processing," in *SPIE, Optical Security and Counterfeit Deterrence Techniques II*, vol. 3314, 1998.

[31] M. Savvides, B. V. Kumar, and P. Khosla, "Cancelable biometric filters for face recognition," *Proceedings of the 17th International Conference on Pattern Recognition, ICPR 2004*, vol. 3, pp. 922–925, 2004.

[32] Z. Hong, "Algebraic feature extraction of image for recognition," *Pattern Recognition*, vol. 24, pp. 211–219, 1991.

[33] Y.-Q. Cheng, "Human face recognition method based on the statistical model of small sample size," *SPIE Proceedings of the Intell. Robots and Comput. Vision*, vol. 1607, pp. 85–95, 1991.

[34] W. Hong, T. Niu, and Z. Yong, "Face identification based on singular value decomposition and data fusion," *Chinese J. Comput. (in Chinese)*, vol. 23, 2000.

[35] L. Spacek, "The Essex Faces94 database," http://cswww.essex.ac.uk/mv/allfaces/index.html.