# Technical Report: Multibiometric Cryptosystems

Abhishek Nagar, *Student Member, IEEE,* Karthik Nandakumar, *Member, IEEE,* and Anil K. Jain, *Fellow, IEEE*

***Abstract*—Multibiometric systems are being increasingly deployed in many large scale biometric applications (e.g., FBI-IAFIS, UIDAI system in India) because they have several advantages such as lower error rates and larger population coverage compared to unibiometric systems. However, multibiometric systems require storage of multiple biometric templates (e.g., fingerprint, iris, and face) for each user, which results in increased risk to user privacy and system security. One method to protect individual templates is to store only the *secure sketch* generated from the corresponding template using a biometric cryptosystem. This requires storage of multiple sketches. In this paper, we propose a feature level fusion framework to simultaneously protect multiple templates of a user as a single secure sketch. To make this framework practical, we propose algorithms for (i) embedding different biometric feature representations (e.g. set of points, binary strings, or real-valued vectors) into a common representation, (ii) encoding and decoding multibiometric secure sketches using two well-known biometric cryptosystems, namely, *fuzzy vault* and *fuzzy commitment*, and (iii) introducing constraints, such as minimum matching performance requirement for a specific biometric trait. We also analyze the trade-off between matching accuracy and security of the proposed multibiometric cryptosystems through the GAR-Security (G-S) curves, which plot the genuine accept rate of the system against the minimum computational complexity involved in decoding a secure sketch without the genuine user's biometric data. The proposed framework has been evaluated on two different databases, one *real* and one *virtual* multimodal database, each containing the three most popular biometric modalities, namely, fingerprint, iris, and face. Experimental results show that both the multibiometric cryptosystems proposed here have higher security and matching performance compared to their unibiometric counterparts.**

***Index Terms*—Multibiometrics, template security, biometric cryptosystem, fuzzy vault, fuzzy commitment, fusion**

## I. Introduction

Multibiometric systems accumulate evidence from more than one biometric trait (e.g., face, fingerprint, and iris) in order to recognize a person [1]. Compared to unibiometric systems that rely on a single biometric trait, multibiometric systems can provide higher recognition accuracy and larger population coverage. Consequently, multibiometric systems are being widely adopted in many large-scale identification systems, including FBI's IAFIS, Department of Homeland Security's US-VISIT, and Government of India's UID. A number of software and hardware multibiometric products have also been introduced by biometric vendors [2], [3].

While multibiometric systems have improved the accuracy and reliability of biometric systems, sufficient attention has not been paid to security of multibiometric templates. Security of multibiometric templates is especially crucial as they contain information regarding multiple traits of the same user. The leakage of this template information to unauthorized individuals constitutes a serious security and privacy threat due to the following two reasons:

1) **Intrusion attack**: If an attacker can hack into a biometric database, he can easily obtain the stored biometric information of a user. This information can be used to gain unauthorized access to the system by either reverse engineering the template to create a physical spoof or replaying the stolen template.
2) **Function creep**: An adversary can exploit the biometric template information for unintended purposes (e.g., covertly track a user across different applications by cross-matching the templates from the associated databases) leading to violation of user privacy.

The fundamental challenge in designing a biometric template protection scheme is to overcome the large intra-user variability among multiple acquisitions of the same biometric trait. A number of techniques have been proposed to secure biometric templates (see [4] for a detailed review). These techniques can be categorized into two main classes:

- **Biometric cryptosystems**: In a biometric cryptosystem, secure sketch ($\mathbf{y}_c$) is derived from the enrolled biometric template[1] ($\mathbf{x}^E$) and stored in the system database instead of the original template. In the absence of the genuine user's biometric data, it must be computationally hard to reconstruct the template from the sketch. On the other hand, given an authentication query ($\mathbf{x}^A$) that is *sufficiently close* to the enrolled template ($\mathbf{x}^E$), it should be easy to decode the sketch and recover the template. Typically, the sketch is obtained by binding the template with a codeword from an error correcting code, where the codeword itself is defined by a key ($\kappa_c$). Therefore, the sketch ($\mathbf{y}_c$) can be written as $\mathbf{f}_c(\mathbf{x}^E, \kappa_c)$, where $\mathbf{f}_c$ is the sketch generation function. The error correction mechanism facilitates the recovery of the original template and hence, the associated key. Examples of biometric cryptosystems include fuzzy vault [5], fuzzy commitment

A. Nagar and A. K. Jain are with the Dept. of Computer Science and Engineering, Michigan State University, East Lansing, MI. A. K. Jain is also with the Dept. of Brain & Cognitive Engineering, Korea University, Seoul.

K. Nandakumar is with the Institute for Infocomm Research, A*STAR, Fusionopolis, Singapore.

[1] In this paper, we use the notation $\mathbf{x}$ to denote a generic biometric feature vector and $\boldsymbol{X}$ to denote a collection of biometric templates corresponding to the same user. The notations $\mathbf{b}$ and $\mathbf{s}$ denote features that are represented as a binary string and point-set, respectively. Superscripts $E$ and $A$ are used to distinguish between the features extracted during enrollment and authentication, respectively.

[6], PinSketch [7], and secret-sharing approaches [8].

- **Template transformation**: Template transformation techniques modify the biometric template ($\mathbf{x}^E$) with a user specific key ($\kappa_t$) such that it is difficult to recover the original template from the transformed template ($\mathbf{y}_t$). During authentication, the same transformation is applied to the biometric query ($\mathbf{x}^A$) and the matching is performed in the transformed domain to avoid exposure of the original biometric template. Since the key $\kappa_t$ needs to be stored in the system along with $\mathbf{y}_t$, the template security is guaranteed only if the transformation function is non-invertible even when $\kappa_t$ is known to the attacker. Some well-known examples of template transformation include Bio-Hashing [9] and cancelable biometrics [10].

Different combinations of the above two basic approaches, called hybrid biometric cryptosystems, have also been proposed [11], [12]. In this paper, we focus on the biometric cryptosystem approach for multibiometric template protection due to two reasons: (i) well-known biometric cryptosystems such as fuzzy vault and fuzzy commitment are available for securing different types of biometric features and (ii) it is relatively easy to analyze the security of a secure sketch by leveraging on the characteristics of error correcting codes.

Biometric cryptosystems have been designed only for specific biometric feature representations. For example, the fuzzy commitment scheme assumes a binary string representation, where the dissimilarity between template and query is measured in terms of the Hamming distance. The fuzzy vault and PinSketch techniques assume point-set based representations and use set difference as the dissimilarity metric. However, multiple templates of a user may not follow the same feature representation. Point-set based features are used when the image has a set of salient points (e.g., fingerprint minutiae). If different samples of a biometric trait exhibit limited relative geometric transformation and limited occlusion, real-valued feature vectors obtained through PCA [13] and LDA [14] can be used. Binary strings are typically obtained through quantization of a real-valued feature vector, which reduces the storage space and matching complexity. For example, the bits in an iriscode [15] are obtained through quantization of the phase response of a Gabor filter applied to the iris image.

This diversity of biometric representations naturally requires a separate template protection scheme for each trait, and a fusion of the decisions made by each trait [16]. This is analogous to a security system that requires multiple low strength (fewer bits) passwords, which is less secure than a system that uses a single password with a larger number of bits. This motivates the proposed approach to protect the multiple biometric templates using a single secure sketch.

While the concept of securing multiple templates simultaneously as a single entity using a biometric cryptosystem has been reported in the literature, published approaches usually assume that different templates follow the same representation scheme. This enables simple concatenation of the individual templates to obtain the fused template [17]. The objective of this work is to examine the feasibility of creating a single multibiometric secure sketch when the traits that are being fused have different feature representations. This paper makes the following contributions:

- A feature level fusion framework to simultaneously secure multiple templates of a user using biometric cryptosystems. To implement this framework, we propose algorithms for the following three tasks:
  1) Converting different biometric representations into a common representation space using various embedding algorithms: (a) binary strings to point-sets, (b) point-sets to binary strings, and (c) fixed-length real-valued vectors to binary strings.
  2) Fusing different features into a single multibiometric template that can be secured using an appropriate biometric cryptosystem such as fuzzy vault and fuzzy commitment; efficient decoding strategies for these biometric cryptosystems are also proposed.
  3) Incorporating a minimum matching constraint for each trait, in order to counter the possibility of an attacker gaining illegitimate access to the secure system by simply guessing/knowing only a subset of the biometric traits.
- A practical implementation and evaluation of the proposed multibiometric cryptosystems using two different databases each containing three biometric modalities, namely, fingerprint, iris, and face.
- An analysis of the GAR-security trade-off in the proposed multibiometric cryptosystems.

The rest of the paper is organized as follows. Section II provides a background on fuzzy vault and fuzzy commitment techniques and compares the various multibiometric template security schemes proposed in the literature. The feature level fusion framework for multibiometric cryptosystems and the associated algorithms are introduced in Section III. Section IV presents the security analysis methodology. Implementation details and performance evaluation of the proposed multibiometric cryptosystems are discussed in Section V. Our conclusions are summarized in section VI.

## II. Related Work

### A. Fuzzy commitment and Fuzzy Vault

Fuzzy commitment [6] is a biometric cryptosystem that can be used to secure biometric traits represented in the form of binary vectors (e.g. iriscodes). Suppose that the enrolled biometric template $\mathbf{b}^E$ is an $N$-bit binary string. In fuzzy commitment, a uniformly random key $\kappa_c$ of length $L$ ($L \leq N$) bits is generated and used to uniquely index a $N$-bit codeword $\mathbf{c}$ of an appropriate error correcting code. The sketch is then extracted from the template as $\mathbf{y}_c = \mathbf{c} \oplus \mathbf{b}^E$, where $\oplus$ indicates the modulo-2 addition. The sketch $\mathbf{y}_c$ is stored in the database along with $\mathbf{h}(\kappa_c)$, where $\mathbf{h}(.)$ is a cryptographic hash function. During authentication, the codeword is obtained from the query biometric $\mathbf{b}^A$ and the sketch $\mathbf{y}_c$ as follows: $\mathbf{c}^* = \mathbf{y}_c \oplus \mathbf{b}^A = \mathbf{c} \oplus (\mathbf{b}^E \oplus \mathbf{b}^A)$. This codeword $\mathbf{c}^*$, which is generally a corrupted version of the original codeword $\mathbf{c}$, can be decoded to get the key $\kappa^*$. The authentication is deemed successful if $\mathbf{h}(\kappa^*)$ is the same as $\mathbf{h}(\kappa_c)$. If the Hamming distance between $\mathbf{b}^E$ and $\mathbf{b}^A$ is not greater than the error

TABLE I
COMPARISON OF FUZZY COMMITMENT AND FUZZY VAULT.

| | **Fuzzy Vault** | **Fuzzy Commitment** |
|---|---|---|
| Representation | Point-set | Binary string |
| Main advantage | Ability to secure fingerprint minutiae | Compact size of the sketch |
| Main limitation | Difficult to generate chaff that are indistinguishable from genuine points | Lack of perfect codes for desired code lengths |
| Parameters | Polynomial degree ($k$), size of the template set ($r$), and number of chaff points ($q$) | Key length $L$, length of codeword $N$, and error correcting capacity of the code |
| GAR-Security tradeoff | Higher values of $(k/r)$ and $q$ lead to lower GAR, but higher security and vice versa | Higher values of $(L/N)$ lead to lower GAR, but higher security and vice versa |
| Implementations | Fingerprint ([19], [20]), face ([21]), iris ([22]), signature ([23]) | Fingerprint ([24]), face ([24], [25]), iris ([26]), signature ([27]) |

correcting capacity of the code, $\kappa^*$ would be the same as $\kappa$ and the matching will be successful.

Fuzzy vault [5] is useful for securing point-set based biometric features such as fingerprint minutiae. Let $\mathbf{s}^E = \{x_1, x_2, ..., x_r\}$ denote a biometric template consisting of a set of $r$ points from a finite field $\mathcal{F}$. In order to secure $\mathbf{s}^E$, a uniformly random cryptographic key $\kappa_c$ of length $L$ bits is generated and this key is transformed into a polynomial $P$ of degree $k$ ($k < r$) over $\mathcal{F}$. All the elements in $\mathbf{s}^E$ are then evaluated on this polynomial to obtain the set $\{P(x_i)\}_{i=1}^r$. The set of points $\{(x_i, P(x_i))\}_{i=1}^r$ is then secured by hiding them among a large set of $q$ randomly generated chaff points $\{(a_j, b_j)\}_{j=1}^q$ that do not lie on the polynomial $P$ (i.e., $b_j \neq P(a_j)$ and $a_j \notin \mathbf{s}^E, \; \forall \; j = 1, 2, \cdots, q$). The set of genuine and chaff points along with their polynomial evaluations constitute the sketch or vault $\mathbf{y}_c$. During authentication, if the query biometric set $\mathbf{s}^A$ is sufficiently close to $\mathbf{s}^E$, the polynomial $P$ can be successfully reconstructed by identifying the genuine points in $\mathbf{y}_c$ that are associated with $\mathbf{s}^E$. Note that for successful reconstruction of $P$ of degree $k$, a minimum of $(k + 1)$ genuine points need to be identified from $\mathbf{y}_c$.

The effectiveness of a biometric cryptosystem depends on the matching performance and the template security. Matching performance is usually quantified by the False Accept Rate (FAR) and the Genuine Accept Rate (GAR) of the biometric system. Security is measured in terms of the information leakage rate[2] or the computational complexity involved in recovering the original template from the secure sketch [18], [8]. Due to intra-user variability in biometric traits, there is usually a trade-off between the GAR and the security. Schemes with higher security tend to have lower GAR and vice versa. Table I summarizes the comparative characteristics of fuzzy vault and fuzzy commitment.

### B. Multibiometric Cryptosystems

A number attempts have been made to extend the secure biometric recognition framework to incorporate multiple biometric traits [28], [29], [17], [16]. Sutcu et al. [28] combined face and fingerprint templates that are both transformed into binary strings. These binary strings are concatenated and used as the input to a fuzzy commitment scheme.

Nandakumar and Jain [29] proposed a multibiometric cryptosystem in which biometric templates based on binary strings and point-sets are combined. The binary string is divided into a number of segments and each segment is separately secured using a fuzzy commitment scheme. The keys associated with these segment-wise fuzzy commitment schemes are then used as additional points in the fuzzy vault constructed using the point-set based features.

Kelkboom et al. [17] provided results for feature level, score level and decision level fusion of templates represented as fixed-length real-valued vectors. Since the match scores are not explicitly available in a biometric cryptosystem, Kelkboom et al. used the number of errors corrected by an error correcting code in a biometric cryptosystem as a measure of the score. Such scores are, however, meaningful only if the cryptobiometric match is successful and the key $\kappa_c$ can be successfully recovered. Moreover, multiple scores can be obtained only if the different templates are secured individually, which leads to suboptimal security. This is also true for decision level fusion. The feature level fusion scheme in [17] involves simple concatenation of two real-valued vectors and binarization of the combined vector using quantization thresholds.

Fu et al. [16] theoretically analyzed the template security and recognition accuracy imparted by a multibiometric cryptosystem, which can be operated in four different ways: nosplit, MN-split, package, and biometric model. The first three models correspond to decision level fusion, where the biometric templates are secured individually. The biometric model is based on feature level fusion of homogeneous templates. However, no system implementation was reported.

Cimato et al. [30] follow a modular approach to design multibiometric cryptosystems. Suppose that $\mathbf{b}_1^E$ and $\mathbf{b}_2^E$ are two biometric templates. A secure sketch $\mathbf{y}_1$ is extracted from $\mathbf{b}_1^E$ along with a hash of the $\mathbf{b}_1^E$, which is further used as a key to secure the second template. This approach is similar to the package model proposed in [16], which in turn is based on the AND decision fusion rule. One advantage of this modular approach is that additional biometric traits can be easily introduced in the multibiometric cryptosystem. Another benefit is that it allows the use of heterogeneous templates. For example, in [30], a secure sketch is used to protect the iriscode template, and the hash value of the iriscode based on the secret key is used to encrypt a fingerprint minutiae template. A limitation of this approach is that its overall security is bounded by the security of the sketch in the outermost layer.

In this paper, we propose a generic framework for the design of a multibiometric cryptosystem with heterogeneous templates and consider practical implementation issues in the case of both binary string and point-set based representations.

[2] Given the secure sketch, leakage rate relates to the uncertainty about the original biometric template (known as privacy leakage) or the cryptographic key associated to it (secret key leakage). In both fuzzy vault and fuzzy commitment, the privacy leakage rate is equal to the secret-key leakage rate because it is trivial to recover (i) the biometric template given the key and the secure sketch and (ii) the key given the template and the secure sketch.

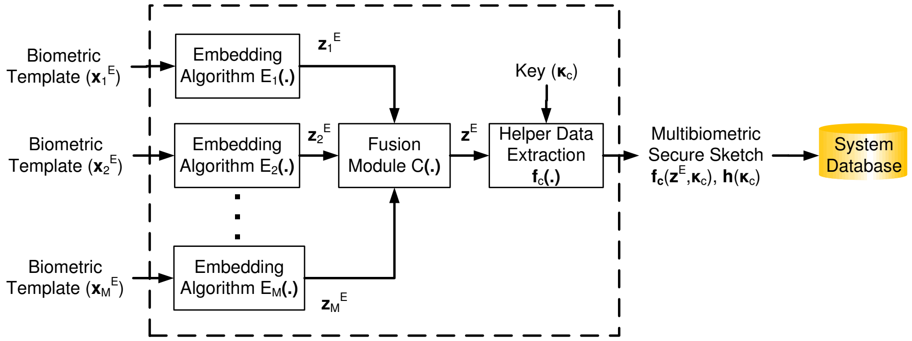

Biometric Template ($\mathbf{x}_1^E$)
Biometric Template ($\mathbf{x}_2^E$)
Biometric Template ($\mathbf{x}_M^E$)
Embedding Algorithm E1(.)
Embedding Algorithm E2(.)
Embedding Algorithm EM(.)
$\mathbf{z}_1^E$
$\mathbf{z}_2^E$
$\mathbf{z}_M^E$
Fusion Module C(.)
$\mathbf{z}^E$
Key ($\kappa_c$)
Helper Data Extraction $\mathbf{f}_c$(.)
Multibiometric Secure Sketch $\mathbf{f}_c(\mathbf{z}^E,\kappa_c)$, $\mathbf{h}(\kappa_c)$
System Database

Fig. 1. Schematic diagram of a multibiometric cryptosystem based on the proposed feature level fusion framework during the enrollment phase.

## III. Proposed Framework for Multibiometric Cryptosystems

We propose a feature level fusion framework for multibiometric cryptosystems that consists of three basic modules: (i) embedding algorithm ($\mathcal{E}$), (ii) fusion module ($\mathcal{C}$), and (iii) biometric cryptosystem ($\mathbf{f}_c$). The generic framework of the proposed multibiometric cryptosystem is shown in Figure 1. Suppose that we have a set of biometric feature representations $\boldsymbol{X} = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_M\}$, where $\mathbf{x}_m$ represents the features corresponding to the $m^{th}$ biometric modality of a user, and $M$ represents the number of modalities, $m = 1, 2, \cdots, M$. The functionalities of the three modules are as follows:

- **Embedding algorithm** ($\mathcal{E}$): The embedding algorithm transforms a biometric feature representation $\mathbf{x}_m$ into a new feature representation $\mathbf{z}_m$, where $\mathbf{z}_m = \mathcal{E}_m(\mathbf{x}_m)$, for all $m = 1, 2, \cdots, M$. The input representation $\mathbf{x}$ can be a real-valued feature vector, a binary string, or a point-set. The output representation $\mathbf{z}$ could be a binary string or a point-set that could be secured using fuzzy commitment or fuzzy vault, respectively.
- **Fusion module** ($\mathcal{C}$): The fusion module combines a set of homogeneous biometric features $\boldsymbol{Z} = \{\mathbf{z}_1, \mathbf{z}_2, \cdots, \mathbf{z}_M\}$ to generate a fused multibiometric feature representation $\mathbf{z}$. For point-set based representations, one can use $\mathbf{z} = \mathcal{C}_s(\mathbf{Z}) = \cup_{m=1}^{M} \mathbf{z}_m$. In the case of binary strings, the fused feature vector can be obtained by simply concatenating the individual strings, i.e., $\mathbf{z} = \mathcal{C}_b(\mathbf{Z}) = [\mathbf{z}_1 \; \mathbf{z}_2 \; \cdots \; \mathbf{z}_M]$. Note that it is also possible to define more complex fusion schemes, where features could be selected based on criteria such as reliability and discriminability.
- **Biometric cryptosystem** ($\mathbf{f}_c$): During enrollment, the biometric cryptosystem generates a secure sketch $\mathbf{y}_c$ using the fused feature vector $\mathbf{z}^E$ (obtained from the set of biometric templates $\boldsymbol{X}^E = \{\mathbf{x}_1^E, \mathbf{x}_2^E, \cdots, \mathbf{x}_M^E\}$) and a key $\kappa_c$, i.e., $\mathbf{y}_c = \mathbf{f}_c(\mathbf{z}^E, \kappa_c)$. During authentication, the biometric cryptosystem recovers $\kappa_c$ from $\mathbf{y}_c$ and $\mathbf{z}^A$ (obtained from the set of biometric queries $\boldsymbol{X}^A = \{\mathbf{x}_1^A, \mathbf{x}_2^A, \cdots, \mathbf{x}_M^A\}$). Fuzzy commitment is used if $\mathbf{z}$ is a binary string, whereas a fuzzy vault is used if $\mathbf{z}$ is a point-set.

### A. *Embedding Algorithms*

We shall now discuss three types of embedding algorithms that can perform the following feature transformations: (i) real-valued vector into a binary string, (ii) point-set into a binary string, and (iii) binary string into a point-set (see Table II).

*1) Real-valued vector to binary string:* A number of schemes have been proposed in literature for binarization of real-valued biometric features. Examples include Binary Multidimensional Scaling techniques [31], Locality Sensitive Hashing [32], Detection Rate Optimized Bit Allocation [33], and quantization of element pairs in the polar domain [34].

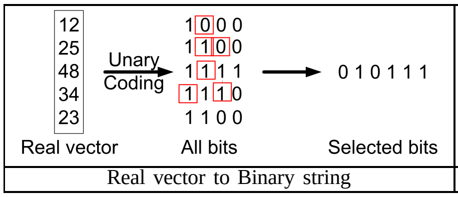
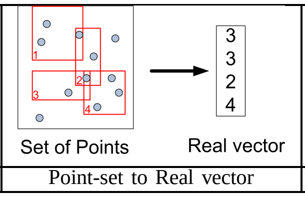
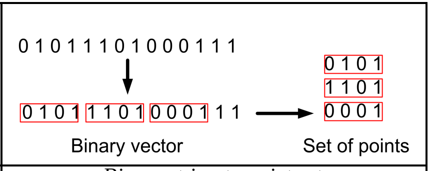
Since no single feature binarization technique is provably better than all others, we propose the following simple algorithm for transforming a real-valued vector into a binary string. First, we quantize each element of the real-valued vector into $(\tau + 1)$ fixed size quanta. The quantized values are then represented using $\tau$-bit unary[3] representation in order to obtain a binary string of length $\tau\ell$, where $\ell$ is the dimensionality of the original vector. In the second stage, we select a desired number of most discriminable bits ($N$). The discriminability of each bit is computed as $((1 - p_g^e)p_i^e)$, where $p_g^e$ and $p_i^e$ are the genuine and impostor bit-error probabilities, respectively.

*2) Point-sets to binary string:* A number of techniques have been proposed for converting point-sets into binary feature vectors. These techniques include local point aggregates [35], spectral minutiae [36], geometric transformation [28], triplet histogram [37], and the bag-of-words approach [38]. In this paper, we implement the simple local aggregates based technique, which works as follows. Let us assume that each point can be represented as an $\nu$-tuple. The available point-set is aligned such that the bounding box of the points is centered at the origin. Then, a set of axis-aligned hyper-rectangles with randomly selected position and size are generated. Among these hyper-rectangles, a fraction of hyper-rectangles with large overlap with other hyper-rectangles is discarded.

Statistics for each hyper-rectangle based on the points falling inside it are computed. These statistics include the

[3] A unary encoding works as follows. Suppose that a real-value $a$ needs to be encoded using $\tau$ bits. The range of $a$, say $[a_{min}, a_{max}]$, is quantized into $(\tau + 1)$ bins. If $a$ falls into the $i^{th}$ bin, it is represented as $(\tau - i + 1)$ ones followed by $(i - 1)$ zeros, where $i = 1, 2, \cdots, (\tau + 1)$.

TABLE II
A SIMPLIFIED ILLUSTRATION OF THE PROPOSED EMBEDDING ALGORITHMS.

| | | |
|---|---|---|
|  |  |  |
| Real vector to Binary string | Point-set to Real vector | Binary string to point-set |

number of points in the hyper-rectangle, and the mean and variance of the points along each of the $\nu$ dimensions. The statistics from different hyper-rectangles are concatenated to generate a feature vector. A Linear Discriminant Analysis (LDA) is applied to the resultant feature vector to reduce the dimensionality. Finally, the real-valued LDA features are binarized using the algorithm presented in section III-A1.

*3) Binary string to point-set:* Conversion of binary string to point-set is required when the final biometric cryptosystem is based on point-set features. In order to obtain a point-set from a binary string, we simply divide the binary string into the desired number of segments. Each segment can be considered as a point in the point-set representation. The only parameter in this technique is the number of segments. A similar technique was also used in [29], where instead of directly using the segments of the binary strings as points, a key is associated with each segment through fuzzy commitment and the keys are used as additional points in the vault.

## *B. Fuzzy Vault Implementation*

We briefly describe how a biometric template (either unibiometric or multibiometric) can be encoded as a vault and how an authentication query can be used to decode the vault.

*1) Vault encoding:* Let $\mathbf{s}^E = \{u_i\}_{i=1}^{r}$ be the biometric template represented as a set of $r$ points, which is to be secured using a vault. Let $\mathbf{U}$ be the universe of all possible biometric points. In practice, the points in $\mathbf{U}$ may not necessarily be elements of the field $\mathcal{F}$. To construct a vault, each point in $\mathbf{U}$ is assigned[4] to a point from $\mathcal{F}$. Let $x_i$ be the element in $\mathcal{F}$ associated with the point $u_i$ in $\mathbf{s}^E$, $\forall\ i = 1, 2, \cdots, r$ and let $\mathbf{s}_g^E = \{x_i\}_{i=1}^{r}$. A set of $q$ chaff points are randomly selected from $(\mathbf{U} \backslash \mathbf{s}^E)$ ('$\backslash$' denotes the set difference operator). Let $\mathbf{s}^C = \{u_j^*\}_{j=1}^{q}$ be the set of chaff points and let $\mathbf{s}_g^C = \{x_j^*\}_{j=1}^{q}$ be the corresponding set of points obtained by mapping elements in $\mathbf{s}^C$ to elements in $\mathcal{F}$. Given a key $\kappa_c$ of length $L$ bits, we encode it as a polynomial $P$ of degree $k$. Finally, the vault is obtained as a set of 3-tuples as follows: $\mathbf{y}_c = \{(\alpha_i, \beta_i, \gamma_i)\}_{i=1}^{t}$, where $t = (r + q)$, $\alpha_i \in (\mathbf{s}^E \cup \mathbf{s}^C)$, $\beta_i$ is the corresponding element in $(\mathbf{s}_g^E \cup \mathbf{s}_g^C)$, and $\gamma_i$ is given by

$$\gamma_i = \begin{cases} P(\beta_i), & \text{if } \alpha_i \in \mathbf{s}^E, \\ \\ b_i, \text{where } b_i \in \mathcal{F} \setminus \{P(\beta_i)\}, & \text{if } \alpha_i \in \mathbf{s}^C. \end{cases}$$

*2) Vault decoding:* Let $\mathbf{s}^A = \{u'_j\}_{j=1}^{r'}$ be the set of $r'$ points in the authentication query. For each point $\alpha_i$ $(i = 1, 2, \cdots, t)$ in the vault, its distance to the closest query point is computed and the list of vault points is sorted based on this distance. The ordered set of points in the vault is given by $\mathbf{y}_c^o = [(\alpha(1), \beta(1), \gamma(1)), \cdots, (\alpha(t), \beta(t), \gamma(t))]$, where $\min_w d(\alpha(i), u'_w) < \min_w d(\alpha(j), u'_w)$ if $i < j$, and $w \in \{1, \cdots, r'\}$. Finally, the Berlekamp-Massey (B-M) algorithm [39] is applied on subsets of different lengths derived from $\mathbf{y}_c^o$ to decode the vault and thereby recover the associated polynomial and the key $\kappa_c$ (see algorithm 1).

**Algorithm 1** Fuzzy vault decoding based on Berlekamp Massey algorithm [39].

```
Input: $\mathbf{y}_c^o = [(\alpha(1), \beta(1), \gamma(1)), \cdots, (\alpha(t), \beta(t), \gamma(t))]$
(Ordered vault points); $k$ (Degree of polynomial)
forall [5] $n = (k+1)$ to $t$ do
  $\mathbf{s}_n \leftarrow \{(\alpha(i), \beta(i), \gamma(i))\}_{i=1}^{n}$
  for $m = 0$ to $n - (k+1)$ do
    forall $\mathbf{s}_* \subset \mathbf{s}_n, |\mathbf{s}_*| = m$ do
      $\mathbf{s}_n^- \leftarrow \mathbf{s}_n \setminus \mathbf{s}_*$
      $P \leftarrow DecodeBM(\mathbf{s}_n^-, k)$
      if $P$ is the required polynomial then
        Return $P$
      end if
    end forall
  end for
end forall
Return $\phi$
{$DecodeBM(\mathbf{s}, k)$ performs a Berlekamp-Massey decoding
of the set of points $\mathbf{s}$ for a polynomial of degree $k$}
```

Algorithm 1 is based on the following principle. Given a set of $n$ points from the vault, the Berlekamp-Massey decoding allows recovery of the polynomial if there are at least $(n + k + 1)/2$ genuine points in the given set. Since the points in the vault are ordered according to their likelihood of being genuine, we consider subsets of $n$ $((k+1) \leq n \leq t)$ most likely points in parallel. If a selected subset of length $n$ cannot decode the vault, some points in the subset are randomly removed to obtain smaller subsets of minimum size $(k + 1)$. Since all points in the vault are used in decoding, the vault will always be eventually decoded, but the decoding complexity will be different for each query.

The proposed vault decoding algorithm differs from the decoding procedure followed in [20] on two main accounts.

[4] This mapping can be stored as a lookup table or defined by a hash function.

[5] **forall** is the parallel for-loop which runs all the instances of the loop simultaneously

Firstly, the use of Berlekamp-Massey algorithm (in place of Lagrange interpolation used in [20]) is expected to make the decoding more efficient. This is because of several possible subsets of size $(k+1)$ must be tried in Lagrange interpolation. Secondly, since the points in the vault are ordered based on their distance to the points in the query biometric set, one would expect the decoding complexity for a genuine user to be significantly less than the decoding complexity for an impostor. While this property provides the required security to the biometric template, it also enables us to easily measure the minimum decoding complexity for an impostor attack.

### *C. Fuzzy Commitment Implementation*

In the fuzzy commitment technique, the biometric template $\mathbf{b}^E$ of length $N$ is bound to a codeword **c** of the same length to generate the secure sketch $\mathbf{y}_c$ as follows: $\mathbf{y}_c = \mathbf{b}^E \oplus \mathbf{c}$. During authentication, the query biometric data ($\mathbf{b}^A$) is used along with the secure sketch to obtain a corrupted codeword $\mathbf{c}^*$, which can be corrected to recover the key $\kappa_c$ that is associated with the codeword **c**. While we follow the encoding procedure that is generally used in the literature, we introduce some modifications to the decoding procedure.

We note that error correcting codes can typically handle more erasures than errors. For example, a linear error correcting code can correct any combination of $g$ erasures and $e$ errors as long as $(g + 2e + 1) \leq D_{min}$, where $D_{min}$ is the minimum distance between the codewords of the code [40]. Hence, if the error (crossover) probabilities of each bit in the biometric feature vector is known, it is possible to consider some of the least reliable bits as erasures during decoding. Algorithm 2 provides a fuzzy commitment decoding procedure that exploits the above characteristic of linear error correcting codes. As in the case of fuzzy vault, we consider the $n$ most reliable bits in parallel ($(N - D_{min} + 1) \leq n \leq N$) and treat the remaining bits as erasures. If the decoding is still not successful, a subset of reliable bits of size $m$ are flipped. If the number of errors among the bits selected for flipping is more than $(m/2)$, then the number of errors will be less after flipping, thereby increasing the possibility of successful decoding. Note that if the selected error correcting code is maximum distance separable (i.e., it satisfies the Singleton bound), then $(D_{min} - 1) = (N - L)$. In this case, the secure sketch can be decoded as long as $L$ bits in the biometric feature vector can be correctly guessed.

**Algorithm 2** A fuzzy commitment decoding algorithm that allows for erasures in the codeword based on the crossover probabilities.

```
Input: $\mathbf{c}^*$ (corrupted codeword); $\mathbf{p} = [p_1, \cdots, p_N]$ (bit reliability vector where $p_i$ indicates the reliability (1-crossover probability) of $\mathbf{c}^*(i)$, $i = 1, 2, \cdots, N$); $D_{min}$.
forall $n = (N - D_{min} + 1)$ to $N$ do
    $\mathbf{s}_n \leftarrow RBS(\mathbf{p}, n, N)$
    for $m = 0$ to $D_{min} + 1$ do
        forall $\mathbf{s}_* \subset \mathbf{s}_n, |\mathbf{s}_*| = m$ do
            $\mathbf{c}' \leftarrow Flip(\mathbf{c}^*, \mathbf{s}_*)$
            $\kappa_c \leftarrow DecodeFC(\mathbf{c}', \mathbf{s}_n, L)$
            if $\kappa_c$ is the required key then
                Return $\kappa_c$
            end if
        end forall
    end for
end forall
Return $\phi$
{$DecodeFC(\mathbf{c}', \mathbf{s}_n, L)$ is an error correction decoder that corrects the errors in the corrupted codeword $\mathbf{c}'$ to obtain a key of length $L$, while considering all bits whose indices are not indicated in $\mathbf{s}_n$ as erasures. The function $RBS(\mathbf{p}, n, N)$ returns the indices of the $n$ most reliable bits. $Flip(\mathbf{c}^*, \mathbf{s}_*)$ returns the codeword $\mathbf{c}'$, in which the bits in $\mathbf{c}^*$ corresponding to points in $\mathbf{s}_*$ are flipped.}
```

### *D. Constrained Multibiometric Cryptosystem*

One of the limitations of multibiometric systems is that they can be circumvented if an adversary can successfully spoof a subset of the involved biometric traits [41]. This issue is also a concern for a multibiometric cryptosystem. Furthermore, the complete multibiometric template can be recovered as a result of successful authentication. Consequently, if an impostor is able to decode the system with a subset of spoofed traits, he can recover the templates corresponding to the other traits as well. It is thus important to design multibiometric cryptosystems that require a minimum amount of discriminatory information from a subset or all the biometric traits, especially those that are difficult to spoof.

We propose a constrained multibiometric system similar in concept to the modular multibiometric cryptosystem proposed in [30]. In our approach, we first identify the biometric traits that are required to be constrained. Our approach requires two different representations of the constrained biometric trait with the following property: it should be hard to obtain one of the representations (called the *free representation*) from the other (called the *primary representation*). One example of a *primary representation* is the minutiae aggregates [35] and the corresponding *free representation* is the set of minutiae. The *primary representation* is secured using the multibiometric cryptosystem as before, whereas the *free representations* of each constrained trait is secured using a unibiometric cryptosystem (see Figure 2). These unibiometric cryptosystems will use different keys than the key used in the multibiometric cryptosystem. Finally, the unibiometric or component secure sketches are encrypted with a symmetric cryptographic algorithm such as AES, where the encryption key is the same as the key associated with the multibiometric cryptosystem.

The authentication involves two stages. In the first stage, the key associated with the multibiometric cryptosystem is recovered. This key is used to decrypt the component secure sketches. In the second stage, the component secure sketches are decoded. All the keys associated with the unibiometric sketches must be correctly recovered for successful authentication. Note that successful authentication of all the unibiometric systems ensures that the user has a minimum amount of

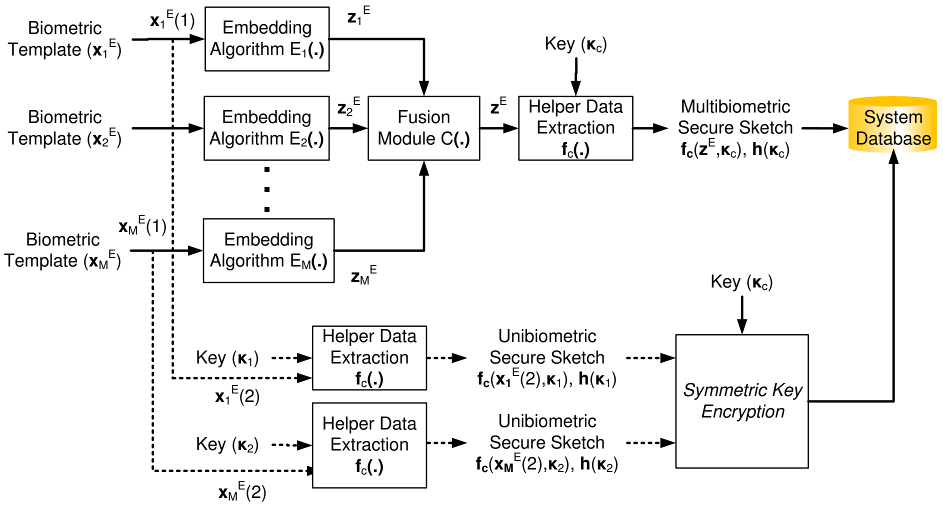

Biometric Template (x1E)
x1E(1)
Embedding Algorithm E1(.)
z1E
Biometric Template (x2E)
Embedding Algorithm E2(.)
z2E
Fusion Module C(.)
zE
Key (κc)
Helper Data Extraction fc(.)
Multibiometric Secure Sketch fc(zE,κc), h(κc)
System Database
Biometric Template (xME)
xME(1)
Embedding Algorithm EM(.)
zME
Key (κc)
Key (κ1)
Helper Data Extraction fc(.)
x1E(2)
Unibiometric Secure Sketch fc(x1E(2),κ1), h(κ1)
Symmetric Key Encryption
Key (κ2)
Helper Data Extraction fc(.)
xME(2)
Unibiometric Secure Sketch fc(xME(2),κ2), h(κ2)

Fig. 2. Enrollment phase of a constrained multibiometric cryptosystem. The templates corresponding to each constrained trait (traits 1 and $M$ in this example) have two representations (the primary representation ($\mathbf{x}_i^E(1)$) and the free representation ($\mathbf{x}_i^E(2)$) for modality $i$). The primary representation is secured using a multibiometric secure sketch, while the free representation is secured using a unibiometric sketch that is further encrypted using the key associated with the multibiometric cryptosystem.

information about each of the constrained biometric traits. The proposed implementation reliably addresses the requirement of a constrained multibiometric cryptosystem, provided there exists a *free* and a *primary* representation of each of the constrained biometric modalities.

## IV. Methodology for Security Analysis

Our security analysis is based on the assumption that the attacker has access to a large biometric database (analogous to a dictionary attack in password-based systems). We then empirically estimate the minimum computational effort required from the attacker to decode a given secure sketch. By following this approach, we combine the two attack strategies traditionally used to estimate system security, i.e., finding sufficiently close biometric features from an available database (false accept attack) and brute-force attack (that indirectly utilizes the distribution of biometric features).

It is possible to decode a secure sketch by directly guessing the associated key ($\kappa_c$). The average complexity of this attack is $(L-1)$ bits, where $L$ is the length of $\kappa_c$. Such an attack can be easily thwarted by choosing a sufficiently large value for $L$. Therefore, we only consider the more practical approach of decoding the sketch by guessing the biometric features. While estimating the computational complexity, we assume that the complexity of the error correction decoder (e.g., B-M algorithm) is unity, and consider only the number of times this decoder needs to be invoked.

### A. Fuzzy Vault Security

To decode a fuzzy vault, one needs to guess the genuine points in the vault. Typically, it is assumed that genuine points in the vault are indistinguishable from the chaff points [20], which leads to optimistic estimates of security. However, the decoding complexity decreases if the attacker has knowledge about the distribution of biometric features [42], [43]. To account for this factor, we assume that the attacker has a large database of non-mate (impostor) biometric samples and he tries to decode the vault using each of those impostor samples. Therefore, we consider the minimum decoding complexity among all impostor matches as a measure of security.

Suppose that the attacker has access to $\mathcal{N}_I$ impostor samples to decode a vault ($\mathbf{y}_c$). Let $\mathbf{s}_n^I$ denote a set containing the first $n$ points from the ordered set of vault points ($\mathbf{y}_c^o$). Here, the ordering is based on the distance of the vault points to the points in the query biometric set from impostor $I$. Let $r_n^I$ be the number of genuine points in $\mathbf{s}_n^I$, i.e., $r_n^I = |\mathbf{s}_n^I \cap \mathbf{s}^E|$, where $\mathbf{s}^E$ is the enrolled template secured using $\mathbf{y}_c$. For $(k+1) \leq n \leq t$, where $t$ is the total number of points in the vault, three different scenarios are possible.

1) If $r_n^I \geq (n+k+1)/2$, the B-M algorithm will return the correct polynomial in a single attempt.
2) If $(k+1) \leq r_n^I < (n+k+1)/2$, one needs to find the minimum value of $m_n^I$ such that when $m_n^I$ chaff points are removed from $\mathbf{s}_n^I$, $r_n^I$ becomes greater than $((n - m_n^I) + k + 1)/2$. Hence, $m_n^I = \max(0, (n - 2r_n^I + k + 1))$ and the corresponding complexity is approximately $\frac{\binom{n}{m_n^I}}{\binom{n-r_n^I}{m_n^I}}$.
3) If $r_n^I < (k+1)$, the vault cannot be decoded using $\mathbf{s}_n^I$. In this case, the corresponding value of complexity is considered to be $\infty$.

Based on the above analysis, the security of the vault can be expressed as

$$\begin{aligned}\mathcal{S}_{FV} &= \min_{n,I}\left(\log_2 \sum_{i=0}^{m_n^I} \frac{\binom{n}{i}}{\binom{n-r_n^I}{i}}\right) + \Omega \\ &\approx \min_{n,I}\left(\log_2 \frac{\binom{n}{m_n^I}}{\binom{n-r_n^I}{m_n^I}}\right) + \Omega, \qquad (1)\end{aligned}$$

where $\Omega = \log_2\left(\mathcal{N}_I(t-k)\right)$. Since the first term in eqn. (1) is minimized over all impostor samples, adding more impostors will lower this term. However, adding more impostors will also increase the number of computations needed, which is reflected by the $\Omega$ term. An increase in the polynomial degree $k$ will increase $n$ and consequently result in higher security.

In the case of multibiometric fuzzy vault, it is possible that a poor quality sample from one of the modalities can lead to a higher decoding complexity if the relative quality of the samples is not taken into account when generating the multibiometric template. In order to address this issue, we also check if any subset of biometric modalities can decode the vault. The final value of security is taken as the minimum value of security computed based on the multibiometric query as well as that corresponding to different subsets of the query biometric traits.

Since the decoding algorithm is common to both the genuine user and the impostor, we can also estimate the decoding complexity for a genuine match. Let $\mathbf{s}_n$ denote a set containing the first $n$ points from the ordered set of vault points $(\mathbf{y}_c^o)$, where the ordering is based on the distance of the vault points to the points in the query from the genuine user. Let $r_n$ be the number of genuine points in $\mathbf{s}_n$, i.e., $r_n = |\mathbf{s}_n \cap \mathbf{s}^E|$. The decoding complexity for the genuine user can be expressed as

$$\mathcal{S}_{FV}^{gen} \approx \min_{n}\left(\log_2 \frac{\binom{n}{m_n}}{\binom{n-r_n}{m_n}}\right) + \log_2(t-k), \qquad (2)$$

where $m_n = \max(0, (n - 2r_n + k + 1))$.

### B. Fuzzy Commitment Security

To decode a fuzzy commitment sketch, one needs to guess the bits in the binary template $\mathbf{b}^E$. Though the length of the template $\mathbf{b}^E$ is $N$ bits, the entropy[6] of the template $(N_*)$ is typically much less than $N$ bits. This is because some bits may not be uniformly distributed (0 and 1 values are not equally likely), while there may also be correlation between the bits.

Suppose that the attacker has access to $\mathcal{N}_I$ impostor samples and a sketch $\mathbf{y}_c$. For each impostor $I$, a corrupted codeword $\mathbf{c}^I$ is obtained as $(\mathbf{y}_c \oplus \mathbf{b}^I)$, where $\mathbf{b}^I$ is the binary feature vector from impostor $I$. Let $\mathbf{s}_n$ denote a set containing the indices of the $n$ most reliable bits in the biometric template[7]. Let $\mathbf{b}_n^E$, $\mathbf{b}_n^I$, and $\mathbf{c}_n^I$ be substrings of $\mathbf{b}^E$, $\mathbf{b}^I$, and $\mathbf{c}^I$, respectively, containing only those bits whose indices are in $\mathbf{s}_n$. The Hamming distance between $\mathbf{b}_n^E$ and $\mathbf{b}_n^I$ is denoted as $\rho_n^I$.

Let $DecodeFC(\mathbf{c}^I, \mathbf{s}_n, L)$ be the error correction decoder that corrects the errors in the corrupted codeword $\mathbf{c}^I$ to obtain a key of length $L$ while considering all bits whose indices are not in $\mathbf{s}_n$ as erasures. When the attacker invokes the above error correction decoder for values of $n$ in the range $[N - D_{min} + 1, N]$, where $D_{min}$ is the minimum distance of the code, three different scenarios are possible.

1) The values of $n$ and $\rho_n^I$ are such that $((N-n)+2\rho_n^I) \leq (D_{min} - 1)$, where $(N-n)$ is the number of erasures and $\rho_n^I$ is the number of errors. In this case, the decoder will return the correct key in a single attempt.
2) If $((N-n)+2\rho_n^I) > (D_{min} - 1)$, the attacker can try to find $m_n^I$ $(0 \leq m_n^I \leq ((D_{min}-1)-(N-n))/2 = (n-L)/2)$ such that, when $m_n^I$ errors are corrected from $\mathbf{c}_n^I$, $((N-n)+2(\rho_n^I - m_n^I))$ becomes less than or equal to $(D_{min}-1)$. If such an $m_n^I$ exists, then its minimum value is given by $m_n^I = \max(0, (((N-n)-(D_{min}-1))/2+\rho_n^I))$ and the corresponding complexity is approximately $\frac{\binom{n}{m_n^I}}{\binom{\rho_n^I}{m_n^I}}$.
3) If no such $m_n^I$ can be found, the secure sketch cannot be decoded by considering the least reliable $(N-n)$ bits as erasures. Hence, the corresponding value of complexity is considered to be $\infty$.

Based on the above analysis, the security of the fuzzy commitment scheme can be expressed as

$$\begin{aligned}\mathcal{S}_{FC} &= \min_{n,I}\left(\log_2 \sum_{i=0}^{m_n^I} \frac{\binom{n}{i}}{\binom{\rho_n^I}{i}}\right) + \Omega \\ &\approx \min_{n,I}\left(\log_2 \frac{\binom{n}{m_n^I}}{\binom{\rho_n^I}{m_n^I}}\right) + \Omega, \qquad (3)\end{aligned}$$

where $\Omega = \log_2\left(\mathcal{N}_I D_{min}\right)$. The above expression, however, assumes that the bits in $\mathbf{b}_n^E$ are independent and uniformly random. Suppose that the entropy of $\mathbf{b}_n^E$ is only $n_*$ bits. In this case, the effective Hamming distance between $\mathbf{b}_n^E$ and $\mathbf{b}_n^I$ is $\rho_{n_*}^I = (n_* \rho_n^I)/n$ and the corresponding value of $m_n^I$ is $m_{n_*}^I = \max(0, (((N-n)-(D_{min}-1))/2+\rho_n^I)n_*/n)$. Thus, the security is given by

$$\mathcal{S}_{FC} \approx \min_{n,I}\left(\log_2 \frac{\binom{n_*}{m_{n_*}^I}}{\binom{\rho_{n_*}^I}{m_{n_*}^I}}\right) + \Omega. \qquad (4)$$

Suppose $\mathbf{b}^A$ is a genuine authentication query and $\rho_{n_*}$ is the effective Hamming distance between $\mathbf{b}_n^E$ and $\mathbf{b}_n^A$, where $\mathbf{b}_n^E$ and $\mathbf{b}_n^A$ are the substrings of $\mathbf{b}^E$ and $\mathbf{b}^A$, respectively, containing only the $n$ most reliable bits. The decoding complexity for a genuine match can be expressed as

$$\mathcal{S}_{FC}^{gen} \approx \min_{n}\left(\log_2 \frac{\binom{n_*}{m_{n_*}}}{\binom{\rho_{n_*}}{m_{n_*}}}\right) + \log_2(D_{min}), \qquad (5)$$

where $m_{n_*} = \max(0, (((N-n)-(D_{min}-1))/2+\rho_n)n_*/n)$.

[6] We use a procedure similar to the one used in [44] to estimate the entropy. See Appendix A for details.

[7] We assume that the attacker can somehow estimate the bit reliability vector (i.e., the crossover probability for each bit in the biometric template).

## V. Experimental results

### A. Databases

We have evaluated the recognition performance and security of the proposed multibiometric cryptosystems on two different multimodal databases, each containing face, fingerprint, and iris modalities. The first database is a *virtual* multimodal database obtained by randomly linking subjects from FVC2002-DB-2 (fingerprint), CASIA Iris database Ver-1, and XM2VTS (face) databases. The virtual multimodal database consists of the full fingerprint database (100 subjects), first 100 subjects from the face database, and first 100 subjects from the iris database. We also use the WVU multimodal database, which is a real multimodal database containing fingerprint, iris, and face images from $138$ different users. Figure 3 show sample images from the different biometric databases used.

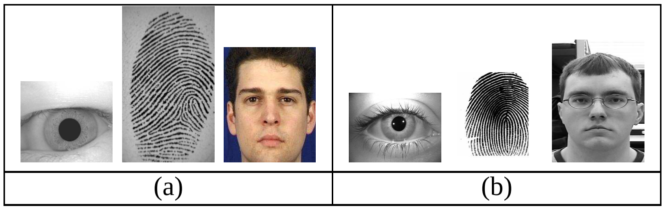


Fig. 3. Sample iris, fingerprint, and face images from (a) CASIA Ver-1, FVC2002 DB-2, and XM2VTS databases, respectively, and (b) WVU multimodal database. Note that the quality of iris images from WVU database is much lower than that from the CASIA database.

*1) Fingerprint Features:* Fingerprint minutiae are extracted using the procedure detailed in [45]. To obtain the binary string representation from the minutiae set, we follow the approach outlined in section III-A2 with $500$ hyper-rectangles (cuboids in 3D space) aligned along the horizontal location, vertical location, and orientation axis associated with minutiae. Different features such as sum of distances of minutiae from the six walls of the cuboids and mean and standard deviations of minutiae along each of the three axes, are extracted from each cuboid in order to obtain a vector of length $3,500$. Linear Discriminant Analysis (LDA) is used to reduce the dimensionality of this vector to $80$. Each LDA coefficient is converted into a $40$-bit unary representation and they are concatenated to obtain a $3200(40 \times 80)$-bit binary string. We select a subset of the most discriminable bits ($N_p$) using the procedure described in section III-A1. First impression of the finger is used for enrollment, the second one is used as authentication sample and the remaining impressions are used as training set in order to compute the LDA features. Since no training is required for extracting minutiae, only the first two impressions are used in constructing the fuzzy vault.

*2) Iris Features:* The binary IrisCode features are extracted based on the algorithm described in [46]. In case of CASIA Ver-1 database, $48$ different radii and $360$ different angles are used whereas in case of WVU Iris database $20$ different radii and $240$ different angles are used. The complete IrisCode are thus $34,560$ and $9,600$-bits long for the CASIA Ver-1 and WVU Iris databases respectively.

In order to reduce the dimensionality of the iriscode and remove the redundancy present in the code, LDA is applied to the iriscode features. Only the top $80$ LDA coefficients are retained ($\ell = 80$) and these real-valued features are then binarized using the technique proposed in Section III-A1 with $\tau = 40$. In order to obtain the point-set representation, $800$ bits selected from the binarized LDA features are divided into $20$ segments of $40$-bits each. As in the case of fingerprints, one iris sample is used for enrollment, one sample is used for authentication, and the remaining samples are used as training set in order to compute the LDA features.

*3) Face Features:* Alignment of face images is essential prior to feature extraction. For the WVU database, eye locations were automatically extracted using Identix FaceIT software, a region of size $120 \times 100$ was cropped such that inter-pupil distance is 60 pixels. In case of XM2VTS database, we use FaceVACS software from Cognitec in order to extract the eye coordinates to align all the face images. The inter-pupil distance is set to $37.5$ pixels. We then crop the aligned face image to a region of size $120\times100$ pixels. Histogram equalization is used to reduce the effect of illumination variations. Finally, we extract $80$ LDA coefficients ($\ell = 80$) that constitute the real-valued feature vector representing a face image. The same procedure applied to the iris LDA coefficients is also applied to the face LDA coefficients to generate a binary string and point-set representations for the face modality. Again, one face image each is used for enrollment and authentication, while the remaining samples are used as the training set in order to compute the LDA features.

### B. Parameter Selection

*1) Unibiometric fuzzy vaults:* We consider the Galois field $GF(2^{16})$ as the finite field $\mathcal{F}$ in all our experiments. In the case of fingerprint fuzzy vault, a set of at most $24$ good quality and well separated minutiae is selected from the given fingerprint image as the biometric points. The chaff points are randomly generated as in [20] to obtain a vault with $224$ points ($r = 24, q = 200$, and $t = 224$). In addition to genuine minutiae and chaff points, points on the fingerprint corresponding to high ridge curvature are also stored in the system. These points are not expected to reveal significant information about the minutiae but can be effectively used to align the query fingerprint [20]. During authentication, the query minutiae set is first aligned with the vault using the high curvature points. A bounding box is then used to filter out points in the vault that are not in close proximity [20] of the query minutiae. The query is then further aligned with the remaining vault points using a minutiae matcher. These aligned points are then used to compute the closest distances of the vault points to the query point based on which the vault points are ordered prior to decoding.

The point-set representations for iris and face modalities can be directly used to construct the iris and face vaults, respectively. To generate chaff points in the iris (face) vault, we pool the iris (face) points extracted from all the iris (face) images in the database (excluding the images of the user under consideration) and select the desired number (200) of chaff points from this pool. During authentication, Hamming distance is used to obtain the closest point in the query for each vault point.

2) *Multi-biometric fuzzy vault:* Multiple unibiometric vaults can be easily converted into a single multibiometric vault by associating the same key $\kappa_c$ with them. Note that the key length ($L$) and hence, the polynomial degree $k$ of such a multibiometric vault is typically higher than the unibiometric case. During decoding, multiple query biometrics are matched with the corresponding unibiometric vaults and an ordered sequence of points from each vault is obtained. These individual sequences of points are then merged such that the first $l$ elements of the merged sequence contain approximately top $\eta_i l$ points from the vault corresponding to the $i^{th}$ biometric. In the current implementation, we choose $\eta_i$ to be the same for all the biometric traits. However, specific strategies can be designed to select proper values of $\eta_i$ based on the quality of the individual biometric traits and the number of genuine points from each trait.

3) *Fuzzy commitment:* We select $1,023$ most discriminable bits from each of the three biometrics for the unibiometric fuzzy commitments ($N = 1,023$). In order to create a multibiometric cryptosystem with $M$ different biometric traits, we extract $N = 1,023 \times M$ most discriminative bits from the pool bits available from all the constituent biometric traits. In our experiments, we assume different values of $D_{min}$ (the minimum distance of the error correcting code) in the range $0.02$ to $0.6$ times the total number of bits $N$.

### C. Performance Evaluation

We evaluate the trade-off between recognition accuracy and security of the proposed multibiometric cryptosystems using the GAR-Security (G-S) curves. The genuine accept rate is measured as the fraction of genuine authentication attempts, where the decoding complexity ($\mathcal{S}_{FV}^{gen}$ and $\mathcal{S}_{FC}^{gen}$ for fuzzy vault and fuzzy commitment, respectively) is less than $15$ bits. The security is measured as the minimum computational complexity faced by the attacker for a successful decoding among the various impostor match attempts. The G-S curve is obtained by varying the length ($L$) of the key ($\kappa_c$) used in the biometric cryptosystem. Note that based on our formulation, the minimum value of security corresponds to the value of $\Omega$ as defined in eq. (1) and eq. (3) for fuzzy vault and fuzzy commitment respectively.

Figures 4, 5 and 6, 7 show the performance of the multibiometric fuzzy vault for the virtual and real multimodal databases, respectively. In general, it can be observed that incorporating additional biometric features does increase the performance of the system. In case of the virtual multimodal database, the security of the iris fuzzy vault at a GAR of $90\%$ is $45$ bits; however, when fingerprint and face are also incorporated in the fuzzy vault, the security increases to around $90$ bits at the same GAR. When the templates are secured individually and the AND fusion rule is applied, i.e., the authentication is deemed successful only when all the unibiometric cryptosystems are decoded, the security at $90\%$ GAR is around $40$ bits. However, in case of the WVU database, there is only a marginal increase in performance compared to the best modality (face). This can be attributed to the lower quality of the iris and fingerprint images in the WVU database compared to the CASIA and FVC2002-DB2 databases, respectively. See Figure 12. In fact, the GAR of the iris fuzzy vault for the WVU database at zero-FAR is $0\%$, which is the reason why the G-S curve corresponding to iris is not shown in Figure 6.

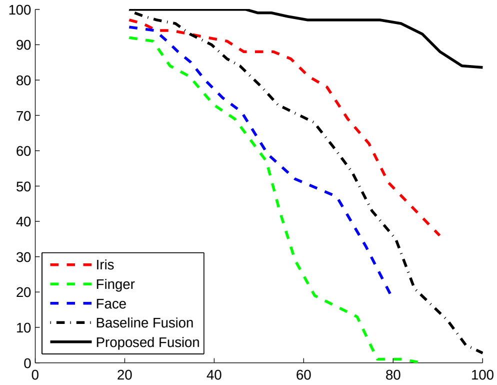


Fig. 4. The G-S curves for fuzzy vault for iris, fingerprint, and face images from CASIA Ver-1, FVC 2002 DB-2, and XM2VTS databases, respectively, the baseline multibiometric cryptosystem based on AND-fusion rule and the proposed multibiometric crytposystem using all three modalities.

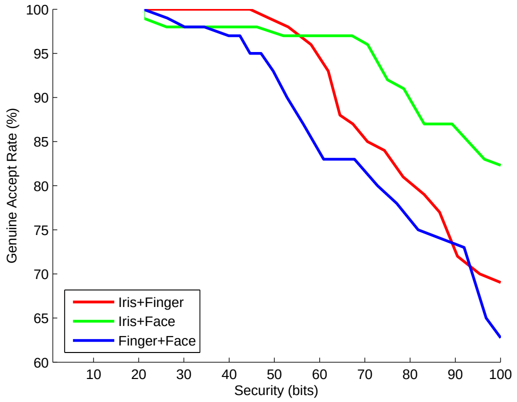


Fig. 5. The G-S curves for fuzzy vault for pairs of biometric traits from CASIA Ver-1, FVC 2002 DB-2, and XM2VTS databases, respectively.

The results corresponding to fuzzy commitment are shown in Figures 8, 9 and 10, 11 for the virtual and real multimodal databases, respectively. The G-S curves are obtained by varying $D_{min}$ of the error correcting code from $0.02$ to $0.6$ times the length of the binary string $N$. Similar to fuzzy vault, the performance of the fuzzy commitment multibiometric cryptosystem is significantly better than the unibiometric systems.

One interesting observation is that even though the performance of the individual modalities are different, the performance of the multimodal system combining all the three traits is nearly same for both fuzzy commitment and fuzzy vault (see Table III). As far as the individual biometric traits are concerned, their performance is better in their native representation. For example, in both the real and virtual multimodal databases, iris fuzzy commitment performs better

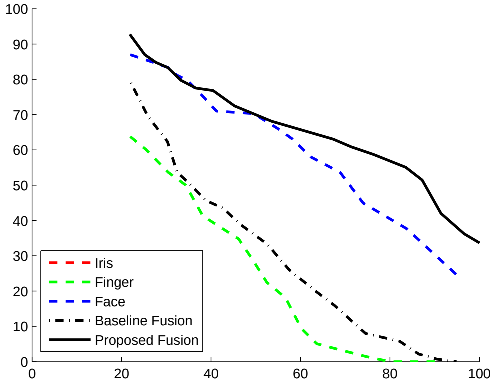


Fig. 6. The G-S curves for fuzzy vault for iris, fingerprint, and face images from WVU Multimodal database, the baseline multibiometric cryptosystem based on AND-fusion rule and the proposed multibiometric crytposystem using all three modalities.

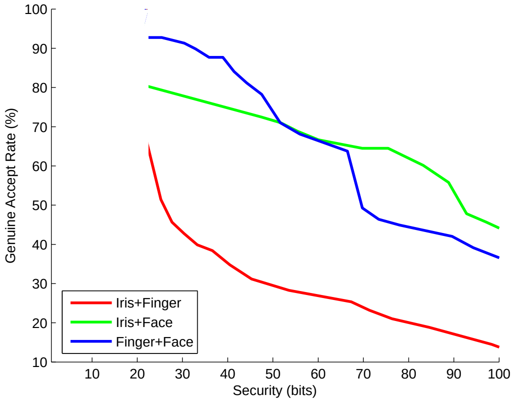


Fig. 7. The G-S curves for fuzzy vault for pairs of biometric traits from the WVU Multimodal database.

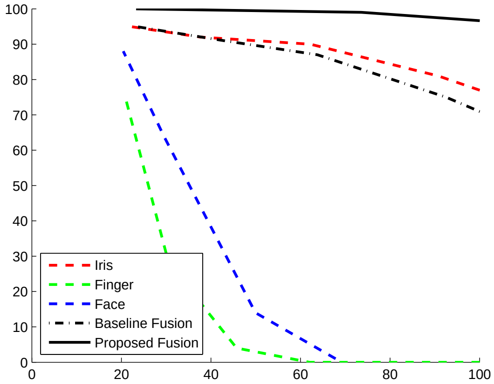


Fig. 8. The G-S curves for fuzzy commitment for iris, fingerprint, and face images from CASIA Ver-1, FVC 2002 DB-2, and XM2VTS databases, respectively, the baseline multibiometric cryptosystem based on AND-fusion rule and the proposed multibiometric crytposystem using all three modalities.

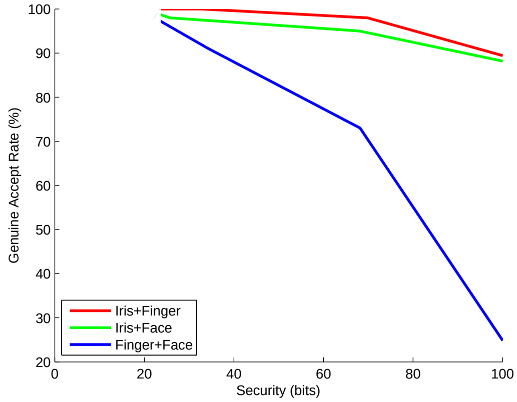


Fig. 9. The G-S curves for fuzzy commitment for pairs of biometric traits from CASIA Ver-1, FVC 2002 DB-2, and XM2VTS databases, respectively, the baseline multibiometric cryptosystem based on AND-fusion rule and the proposed multibiometric crytposystem using all three modalities.

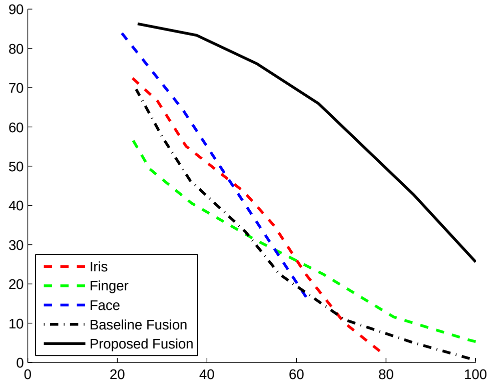


Fig. 10. The G-S curves for fuzzy commitment for iris, fingerprint, and face images from WVU Multimodal database, the baseline multibiometric cryptosystem based on AND-fusion rule and the proposed multibiometric crytposystem using all three modalities.

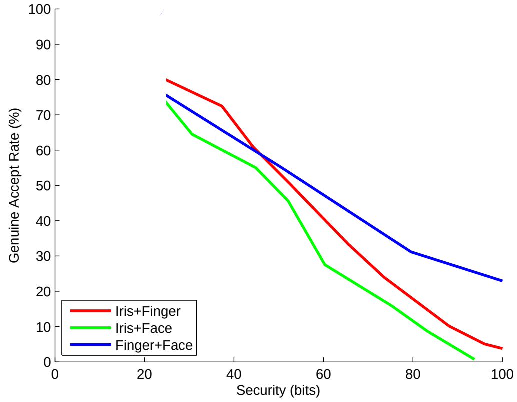


Fig. 11. The G-S curves for fuzzy commitment for pairs of biometric traits from the WVU Multimodal database.

than a iris fuzzy vault. Similarly, the performance of fingerprint fuzzy vault is generally better than a fingerprint fuzzy commitment. This could be due to possible loss of discriminatory information during feature transformation (embedding). Figure 12 shows the receiver operating characteristic (ROC) curves corresponding to the original features associated with the biometric traits along with the ROC curves obtained using the features extracted for fuzzy vault and fuzzy commitment. Note that in most of the cases the matching performance reduces as a result of using a biometric cryptosystem.

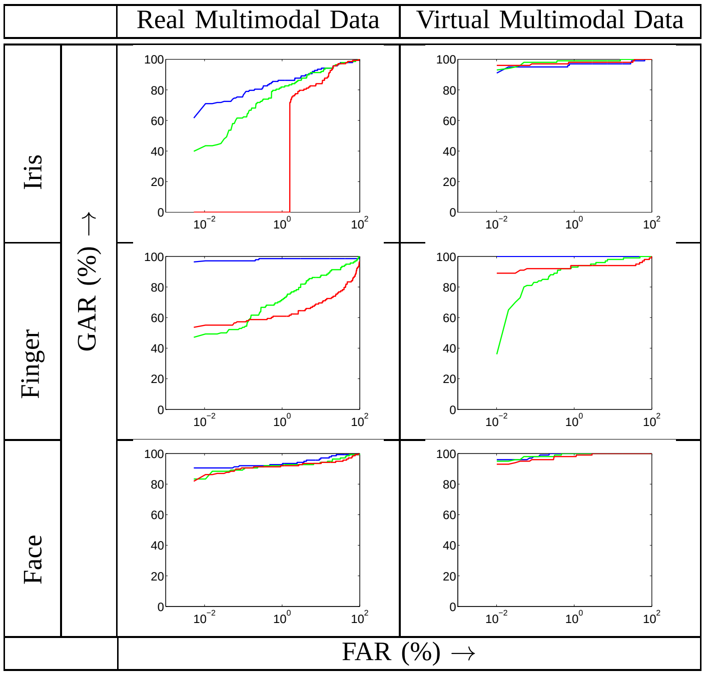


Fig. 12. ROC curves corresponding to the original features (blue), features processed for fuzzy commitment (green) and features processed for fuzzy vault (red) for all three biometric traits i.e. iris, fingerprint and face. The ROC curves corresponding to the original features for iris is based on the Hamming distance between iriscodes, for face it is based on the LDA features and for fingerprint it is based on the scores obtained from Neurotechnology Verifinger matcher using only the minutiae features. The curves corresponding to the fuzzy commitment are based on Hamming distance between $1,023$ bits of the extracted binary feature vector. The curves corresponding to the fuzzy vault are computed using the decoding complexity as the matching score when a degree-$10$ polynomial used.

For the multibiometric fuzzy vault implementation reported in [29], where iris and fingerprint templates from MSU-DBI database and CASIA Ver-1 database, respectively, were secured together, the genuine accept rate was $98.2\%$ at a security of $49$ bits. Note that the security estimate in [29] assumes uniform distribution of biometric features. In our implementation, the genuine accept rate is $99\%$ at a security of $49$ bits based on the FVC2002-DB2 and the CASIA Ver-1 databases. See Figure 5. In [30], security of the system has not been explicitly reported. In [17], the proposed technique performs fusion of two different 3D face recognition algorithms and thus cannot be directly compared to the techniques proposed here. In [16], no experimental results were reported.

To validate the constrained multibiometric cryptosystem, we implemented a system consisting of iris and fingerprint modalities, where minimum matching constraints are imposed for the fingerprint modality. We further assume that the adversary has knowledge about iris biometric, i.e., he has access to some iris image of the enrolled user. In this experiment, the *primary* biometric corresponds to the binary features thus a multibiometric fuzzy commitment is implemented. Minutiae are employed as the *free* fingerprint representation, and hence a fuzzy vault is used in the second stage. The degree of polynomial for the fuzzy vault is selected such that the sum of security in bits and GAR in percentage of the resulting system is maximized. We observed that at around $70\%$ GAR, the security of the constrained system is around $35$ bits, whereas the security is less than 22 bits when no constraints are applied.

| Traits | Real Multimodal Database | | Virtual Multimodal Database | |
|---|---|---|---|---|
| | Fuzzy vault | Fuzzy commitment | Fuzzy vault | Fuzzy commitment |
| Iris | 0% | 37% | 88% | 91% |
| Finger | 22% | 30% | 51% | 2% |
| Face | 67% | 33% | 58% | 12% |
| Baseline Fusion | 33% | 27% | 75% | 89% |
| Proposed Fusion | 68% | 75% | 99% | 99% |

TABLE III
COMPARISON OF GENUINE ACCEPT RATES OF THE DIFFERENT BIOMETRIC CRYPTOSYSTEMS AT A SECURITY LEVEL OF $53$ BITS, WHICH EQUALS THE SECURITY IMPARTED BY A RANDOMLY CHOSEN $8$ CHARACTER PASSWORD [47]. HERE, BASELINE FUSION REFERS TO SECURING INDIVIDUAL TEMPLATES USING UNIBIOMETRIC CRYPTOSYSTEMS AND COMBINING DECISIONS USING AND-RULE FUSION, WHILE THE PROPOSED FUSION SCHEME USES A SINGLE MULTIBIOMETRIC SECURE SKETCH.

## VI. Conclusions and Future Work

We have proposed a feature-level fusion framework for the design of multibiometric cryptosystems that simultaneously protects the multiple templates of a user using a single secure sketch. The feasibility of such a framework has been demonstrated using both fuzzy vault and fuzzy commitment, which are two of the most well-known biometric cryptosystems. We have also proposed different embedding algorithms for transforming biometric representations, efficient decoding strategies for fuzzy vault and fuzzy commitment, and a mechanism to impose constraints such as minimum matching requirement for specific modalities in a multibiometric cryptosystem. A realistic security analysis of the multibiometric cryptosystems has also been conducted. Experiments on two different multibiometric databases containing fingerprint, face, and iris modalities demonstrate that it is indeed possible to improve both the matching performance and template security using the multibiometric cryptosystems.

There are four critical issues that need to be investigated further: (i) Embedding schemes for transforming one biometric representation into another, while preserving the discriminative power of the original representation. (ii) A better feature fusion scheme must be developed to generate a *compact* multibiometric template that retains most of the information content in the individual templates. (iii) How to improve the security analysis by accurately modeling the biometric feature distributions. (iv) Evaluation on large multimodal databases.

## APPENDIX A
## ENTROPY OF BIOMETRIC FEATURES

By entropy of biometric features, we mean the minimum average number of bits required to represent a biometric feature vector. A simple approximation of the entropy for iriscodes was provided by Daugman [44], as

$$N_* = p(1-p)/\sigma^2 \tag{6}$$

where $p$ is the mean value of the observed normalized Hamming distances corresponding to impostor matches and $\sigma^2$ is their variance. This estimation assumes that biometric features consists of a set of Bernoulli random variables, which are independent and identically distributed (with uniform distribution). In our case, since the mean normalized Hamming distance was less than $0.5$, we assume that few bits are constant for all biometric samples. Normalized Hamming distance ($\rho_{NH}$) is thus computed as

$$\rho_{NH} = \rho_H/(2 * \mu) \tag{7}$$

where $\mu$ is the mean of the impostor Hamming distances and $\rho_H$ is the corresponding original Hamming distance. This value of normalized Hamming distance $(\rho_{NH})$ is used to computed the values of $p$ and $\sigma$ which, in turn, is used to estimate the entropy of biometric features using eq. (6).

## REFERENCES

[1] A. Ross, K. Nandakumar, and A. K. Jain, *Handbook of Multibiometrics*. Springer, 2006.

[2] http://www.morpho.com/morphotrak/MorphoTrak/mt_multi-biometrics.html.

[3] http://www.cogentsystems.com/fusion.asp.

[4] A. Jain, K. Nandakumar, and A. Nagar, “Biometric template security,” *EURASIP Journal on Advances in Signal Processing*, vol. 2008, pp. 1–17, 2008.

[5] A. Juels and M. Sudan, “A Fuzzy Vault Scheme,” in *Proc. IEEE International Symposium on Information Theory*, Lausanne, Switzerland, 2002, p. 408.

[6] A. Juels and M. Wattenberg, “A Fuzzy Commitment Scheme,” in *Proc. Sixth ACM Conference on Computer and Communications Security*, Singapore, November 1999, pp. 28–36.

[7] Y. Dodis, R. Ostrovsky, L. Reyzin, and A. Smith, “Fuzzy Extractors: How to Generate Strong Keys from Biometrics and Other Noisy Data,” Cryptology ePrint Archive, Tech. Rep. 235, February 2006, A preliminary version of this work appeared in EUROCRYPT 2004.

[8] T. Ignatenko and F. M. J. Willems, “Biometric systems: Privacy and secrecy aspects,” *IEEE Trans. on Information Forensics and Security*, vol. 4, no. 4, pp. 956–973, 2009.

[9] A. B. J. Teoh, K.-A. Toh, and W. K. Yip, “$2^N$ Discretisation of BioPhasor in Cancellable Biometrics,” in *Proc. Second International Conference on Biometrics*, Seoul, South Korea, August 2007, pp. 435–444.

[10] 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, April 2007.

[11] W. Scheirer and T. Boult, “Bio-cryptographic protocols with bipartite biotokens,” in *Proc. Biometric Symposium*, 2008.

[12] K. Nandakumar, A. Nagar, and A. K. Jain, “Hardening Fingerprint Fuzzy Vault Using Password,” in *Proc. Second International Conference on Biometrics*, Seoul, South Korea, August 2007, pp. 927–937.

[13] M. Turk and A. Pentland, “Eigenfaces for recognition,” *Journal of Cognitive NeuroScience*, vol. 3, no. 1, pp. 71–86, 1991.

[14] 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.

[15] J. Daugman, “Recognizing Persons by their Iris Patterns,” in *Biometrics: Personal Identification in Networked Society*, A. K. Jain, R. Bolle, and S. Pankanti, Eds. London, UK: Kluwer Academic Publishers, 1999, pp. 103–122.

[16] B. Fu, S. X. Yang, J. Li, and D. Hu, “Multibiometric cryptosystem: Model structure and performance analysis,” *IEEE Transactions on Information Forensics and Security*, vol. 4, no. 4, pp. 867–882, December 2009.

[17] E. Kelkboom, X. Zhou, J. Breebaart, R. Veldhuis, and C. Busch, “Multi-algorithm fusion with template protection,” in *Proc. IEEE 3rd International Conference on Biometrics: Theory, Applications, and Systems*, Washington, DC, September 2009.

[18] L. Lai, S. Ho, and H. Poor, “Privacy-security tradeoffs in biometric security systems,” in *Proc. the 46th Annual Allerton Conference on Communication, Control, and Computing*, 2008, pp. 23–26.

[19] S. Yang and I. Verbauwhede, “Automatic Secure Fingerprint Verification System Based on Fuzzy Vault Scheme,” in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 5, Philadelphia, USA, March 2005, pp. 609–612.

[20] 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, December 2007.

[21] Y. C. Feng and P. C. Yuen, “Protecting Face Biometric Data on Smartcard with Reed-Solomon Code,” in *Proc. CVPR Workshop on Biometrics*, New York, USA, June 2006, p. 29.

[22] Y. J. Lee, K. Bae, S. J. Lee, K. R. Park, and J. Kim, “Biometric Key Binding: Fuzzy Vault based on Iris Images,” in *Proc. Second International Conference on Biometrics*, Seoul, South Korea, August 2007, pp. 800–808.

[23] M. Freire-Santos, J. Fierrez-Aguilar, and J. Ortega-Garcia, “Cryptographic Key Generation Using Handwritten Signature,” in *Proc. SPIE Conference on Biometric Technologies for Human Identification*, vol. 6202, Orlando, USA, April 2006, pp. 225–231.

[24] J. Bringer, H. Chabanne, G. Cohen, B. Kindarji, and G. Zemor, “Theoretical and practical boundaries of binary secure sketches,” *IEEE Transactions on Information Forensics and Security*, vol. 3, p. 673683, 2008.

[25] T. A. M. Kevenaar, G. J. Schrijen, M. vanderVeen, A. H. M. Akkermans, and F. Zuo, “Face recognition with renewable and privacy preserving binary templates,” in *Proc. AutoID*, 2005, pp. 21–26.

[26] F. Hao, R. Anderson, and J. Daugman, “Combining Crypto with Biometrics Effectively,” *IEEE Transactions on Computers*, vol. 55, no. 9, pp. 1081–1088, September 2006.

[27] E. Maiorana and P. Campisi, “Fuzzy commitment for function based signature template protection,” *IEEE Signal Processing Letters*, vol. 17, no. 3, pp. 249–252, 2010.

[28] Y. Sutcu, Q. Li, and N. Memon, “Secure Biometric Templates from Fingerprint-Face Features,” in *Proc. CVPR Workshop on Biometrics*, Minneapolis, June 2007.

[29] K. Nandakumar and A. K. Jain, “Multibiometric template security using fuzzy vault,” in *Proc. Biometrics: Theory, Applications and Systems*, 2008.

[30] S. Cimato, M. Gamassi, V. Piuri, R. Sassi, and F. Scotti, “Privacy-aware biometrics: Design and implementation of a multimodal verification system,” in *Proc. IEEE Annual Conference on Computer Security Applications*, Los Alamitos, CA, 2008.

[31] D. Rhodes, “Methods for Binary Multidimensional Scaling,” *Neural Computation*, vol. 14, pp. 1195–1232, 2002.

[32] A. Andoni and P. Indyk, “Near-optimal hashing algorithms for approximate nearest neighbor in high dimensions,” in *IEEE Symposium on Foundations of Computer Science*, 2006, pp. 459–468.

[33] C. Chen, R. N. J. Veldhuis, T. A. M. Kevenaar, and A. H. M. Akkermans, “Biometric Quantization through Detection Rate Optimized Bit Allocation,” *EURASIP Journal on Advances in Signal Processing*, 2009.

[34] C. Chen and R. Veldhuis, “Binary Biometric Representation through Pairwise Polar Quantization,” in *Proc. International Conference on Biometrics*, 2009, pp. 72–81.

[35] A. Nagar, S. Rane, and A. Vetro, “Privacy and Security of Features extracted from Minutiae Aggregates,” in *Proceedings IEEE Intl Conf. on Acoustics, Speech and Signal Processing*, Dallas, TX, March 2010, pp. 524–531.

[36] H. Xu, R. Veldhuis, T. Kevenaar, A. Akkermans, and A. Bazen, “Spectral minutiae: A fixed-length representation of a minutiae set.” in *Proc. IEEE Computer Vision and Pattern Recognition. Workshop on Biometrics*, Anchorage, Alaska, 2008.

[37] F. Farooq, R. Bolle, T. Jea, and N. Ratha, “Anonymous and revocable fingerprint recognition,” in *Proc. IEEE Computer Vision and Pattern Recognition*, June 2007.

[38] L. Fei-Fei and P. Perona, “A Bayesian hierarchical model for learning natural scene categories,” in *Proc. of IEEE Computer Vision and Pattern Recognition*, 2005, pp. 524–531.

[39] E. R. Berlekamp, *Algebraic Coding Theory*. McGraw Hill, 1968.

[40] J. I. Hall, “Notes on coding theory,” http://www.mth.msu.edu/~jhall/classes/codenotes/GRS.pdf, 2001.

[41] R. N. Rodrigues, L. L. Ling, and V. Govindaraju, “Robustness of multimodal biometric fusion methods against spoof attacks,” *Journal of Visual Languages and Computing*, vol. 20, no. 3, pp. 169 – 179, 2009.

[42] A. Nagar and A. K. Jain, “On the Security of Non-Invertible Fingerprint Template Transforms,” in *Proc. IEEE Workshop on Information Forensics and Security*, London, UK, December 2009.

[43] E.-C. Chang, R. Shen, and F. W. Teo, “Finding the original point set hidden among chaff,” in *Proc. the 2006 ACM Symposium on Information, computer and communications security*, June 2009.

[44] J. Daugman, “The importance of being random: statistical principles of iris recognition,” *Pattern Recognition*, vol. 36, pp. 279–291, 2003.

[45] 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, April 1997.

[46] S. Shah, “Enhanced iris recognition: Algorithms for segmentation, matching and synthesis,” Department of Computer Science and Electrical Engineering, West Virginia University, Master’s Thesis, 2006.

[47] W. E. Burr, D. F. Dodson, and W. T. Polk, “Information Security: Electronic Authentication Guideline,” NIST, Technical Report Special Report 800-63, April 2006.