# A crypto-biometric scheme based on iris-templates with fuzzy extractors

R. Álvarez Mariño

*Health Centre "Nuñez Morgado"*
*Department of Health, Regional Government*
*C/ Nuñez Morgado, 28036 Madrid, Spain*

F. Hernández Álvarez, L. Hernández Encinas*

*Department of Information Processing and Coding*
*Information Security Institute (ISI), CSIC*
*C/ Serrano 144, 28006 Madrid, Spain*

## Abstract

One of the most important uses of Biometrics is the identification and authentication of individuals using one or several of their physiognomical or behavioral features. Moreover, Biometrics offers a good option to assist Cryptography for confidentiality, encryption, and decryption of messages by using some biometric traits. In this paper, a crypto-biometric scheme, based on fuzzy extractors, by using iris templates, is proposed, i.e., we propose a new system in order to permit a user to retrieve a secret or a previously saved key by using her own biometric template. The properties and efficiency for selecting the most useful parameters to provide a high level of security in the scheme are thoroughly analyzed.

*Keywords:* Biometrics, Cryptography, Iris Template, Fuzzy extractor

## 1. Introduction

It is well known that Cryptography provides secrecy and authentication of data and ensures privacy in the communication by means of different cryptosystems [28, 36].

The most important application of symmetric and asymmetric cryptography consists of encrypting and decrypting messages or plaintexts by means of specific cryptosystems and keys. In the case of symmetric cryptography, the sender

*Corresponding author

*Email addresses:* `ralvarez.gapm02@salud.madrid.org` (R. Álvarez Mariño), `fernando.hernandez@iec.csic.es` (F. Hernández Álvarez), `luis@iec.csic.es` (L. Hernández Encinas)

encrypts a plaintext by using a secret key, which is only shared with the receiver; for decrypting the ciphertext, the receiver uses the same secret key used by the sender. In the case of asymmetric cryptography, the sender encrypts the message by using the receiver's public key; to decrypt the ciphertext, the receiver uses her private key, only known to herself.

Another very extended application of cryptography is as a tool for the identification and authentication of users to be used in generating and verifying digital signatures [22, 25, 41], voting process [8], key agreement and digital signature protocols based on the identity [13, 40], accessing personal data [23], etc.

Nevertheless, the use of cryptosystems and keys in cryptography presents, currently, two important drawbacks, namely:

- *Key management problem.* It is already known that the security of the standard cryptosystems, such as the TDEA, AES, RSA, and so on [29, 30, 33], relies on the assumption that both keys, the secret and the private one, are unknown to anyone but the lawful users. If the secret key or the private key is compromised, the security of the scheme is completely broken.

- *Key bitlength problem.* As the keys used are large (128-256 bits for symmetric cryptosystems and 1024-2048 for the asymmetric ones), it is impossible to be memorized. So, they are often stored in a password-protected place. As passwords can be easily stolen, forgotten or guessed using different attacks, it can be stated that a cryptosystem is as secure as it is the password used to store its secret key [31].

For this reason it is important to develop new systems with higher level of security, so that the storing algorithms will be improved in order to avoid the above mentioned drawbacks.

Biometric authentication [17, 27] consists of verifying individuals based on their physiognomical and behavioral traits such as face [9], fingerprints [19], palmprints [26], iris [34], tongue shape [16], etc. Biometric systems offer obvious advantages over other authentication systems. They are inherently more reliable than password-based authentication, as biometric characteristics cannot be lost or forgotten. Moreover, biometric traits are extremely difficult to copy, forge, share, and distribute, and it is unlikely for a user to deny having accessed a particular content. Thus, biometrics-based authentication can be used instead of password-based authentication to assist cryptosystems to encrypt and decrypt messages by using biometric key templates.

In general, the process to identify a user by means of her biometric templates consists of two phases:

1. In the *enrollment phase*, the biometric templates are processed and stored into the database.
2. In the *verification phase*, a new biometric template (called the query template) is extracted from the user who wants to be identified, and it is compared with the data already stored (reference template). If the comparison matches, the user is identified, otherwise, her identification is rejected.

The most straightforward way to secure a biometric system, including the template, is to store all the system modules and the interfaces between them on a smart card. These systems are known as match-on-card systems and their advantage is that the biometric information never leaves the card. The drawback is that these systems are not appropriate for large-scale applications and it is possible to retrieve the template from a lost card. So, both the system and the template must be more securely protected.

Thus, to assist cryptosystems for encrypting and decrypting, biometrics-based authentication schemes, which use biometric key templates, can be considered instead of passwords-based authentication schemes. The systems obtained from the blend of both technologies, Cryptography and Biometrics, are known as biometric cryptosystems or crypto-biometric systems.

In this paper, we present a crypto-biometric scheme based on a fuzzy extractor scheme, in order to permit a user to retrieve a secret or a key, which has been previously stored, by using her own biometric templates, in particular, her iris templates. Moreover, we carry out a complete study about the behavior of that scheme by using the whole iris database CASIA [2], and by considering the standard key bitlengths for symmetric cryptosystems, i.e., from 64 to 256 bits.

The main objective of this work is to extend and to complete a preliminary study [15] in order to calculate in a more accurate way the values of the *False Acceptance Rate* and *False Rejection Rate*. The novelty of the proposed scheme stems mainly from the use of the fuzzy extractor method to perform a secure link between a secret, chosen by a user, and the iris template of the same user in secure way. The objectives to be achieved by this process are: Avoid compromising the user-extracted iris template, and assure that the correct user is the only who retrieves the secret (the one whose iris template was previously linked to the secret).

As the results show, this scheme could be considered as a good option to an in-depth study of its properties and applications in forth-coming works.

The rest of this work is organized as follows. In Section 2 an overview about the different template protection schemes is presented, explaining their main categories. The detailed description of the proposed biometric fuzzy extractor is carried out in Section 3. Section 4 deals with the explanation of the different experiments performed with the scheme, together with their results. A study about the intra-user variability using the Hamming distance among templates of the same users is carried out in Section 5; and finally, the conclusions are presented in Section 6.

## 2. Definitions and related work

The *intra-user variability* measures the differences between two biometric templates extracted from the same user, while the *inter-user variability* measures the similarities between two biometric templates extracted from different users. If the former measure is high or the latter one is low, they can lead, respectively, to reject a valid user of the system or to accept an attacker as a legal user. So, in order to characterize both variabilities, two ratios are used: False Rejection Rate (*FRR*) and False Acceptance Ratio (*FAR*), respectively.

It is well known that the most clear advantage of authentication systems based on passwords over those based on biometric templates is that the former permit canceling and changing the password, whereas in the latter, it is impossible to cancel a biometric trait. As a result, whenever a biometric template is compromised, it cannot be used anymore.

Cancelable biometrics is a way to import into biometrics the cancellation and replacement characteristics present in passwords-based systems [32]. The main objectives of these new biometrics template schemes are [18]:

- *Diversity.* Cross-matching between databases should not be allowed in order to ensure user's privacy.
- *Revocability.* A compromised biometric template should be easily revoked and replaced by a new one, based on the same biometric trait.
- *Security.* Obtaining the original biometric template from a secure-stored template should need a great computational effort. In this way physical spoof creation is prevented.
- *Performance.* The recognition performance (*FRR* and *FAR*) of the biometric system should not be degraded even if the biometric templates are distorted.

The way to provide these properties consists of distorting the biometric templates before the matching process. The variability in the distortion parameters allows the existence of different schemes. Several approaches, known as biometric template protection schemes [18], have been proposed. These schemes can be mainly classified into two categories: *feature transformation approach* and *biometric cryptosystem.*

A feature transformation approach scheme basically consists of the application of a transformation function to the biometric template and, subsequently, store the transformed template into a database. In the matching phase, the same function is first applied to the query templates and, then, the transformed query is directly matched with the transformed template in the transformed domain [18]. Depending on the properties of the transformation function, the feature transformation schemes can be divided into invertible (salting) and non-invertible transformation.

The basic idea for biometric cryptosystems [39] is either securing a cryptographic key using biometric templates or directly generating a cryptographic key from biometric templates. The main characteristic of these systems is that they need to generate public information related to the biometric template in order to perform the verification phase. This public information is called *helper data* and it should not reveal any important information about the biometric template.

Biometric cryptosystems can be further classified into different models:

- *Key release*: The biometric template and the key are stored as different entities in such a way that the key is to be released only after a successful biometric verification. Therefore the biometric verification phase and the key release mechanism are completely decoupled.
- *Key binding*: The biometric template is secured by binding it and a key within a cryptographic framework. Therefore they constitute a single entity depending on the biometric template, $\mathfrak{B}$, and the key, $K$.
- *Key generation*: A key is derived directly from the biometric template and it is stored in the database instead of the biometric template itself [18].

In a key binding cryptosystem, the biometric template is secured by binding it and a key in a cryptographic framework, and both are stored in the database as a single entity as the helper data. This system has the advantage that it is tolerant to intra-user variability, depending on the error correcting capability. But at the same time, the fact of using error correction schemes makes it necessary the use of sophisticated matchers. Several template protection technologies can be considered as key binding approaches, for example, fuzzy vault schemes [20], fuzzy commitment schemes [21], etc.

Along this work, the term *fuzziness* is used to denote the fact that two samples extracted from the same user at two distinct moments are similar but not completely equal due to the intra-user variability. In this sense, we use fuzziness as an indicator of intra-user variability and it is not related to the fuzzy set theory. An application of this theory, in particular fuzzy inference systems, to multimodal biometry is presented in [14].

A fuzzy vault scheme [20] is a cryptographic framework that binds a biometric template and a secret key to build a secure sketch of that template. In this way, the sketch can be stored in the database with no risk, because it is computationally hard to retrieve either the template or the key without knowing anything about the user's biometric data.

A fuzzy extractor scheme is a biometric tool whose basic aim [12] is to authenticate a user using her own biometric template, $\mathfrak{B}$, as the key. To do so, it makes use of another process, known as secure sketch, to allow precise reconstruction of a noisy input. The correctness of the whole procedure depends on the differences between $\mathfrak{B}$, used in the enrollment phase, and the query template, $\bar{\mathfrak{B}}$, used in the verification phase.

In general, a fuzzy extractor determines a uniformly random string $S$ from its input $\mathfrak{B}$ in a noise-tolerant way. If the input changes to some $\bar{\mathfrak{B}}$ but remains close to $\mathfrak{B}$, the string $S$ can still be exactly reproduced. As in any other biometric system, the fuzzy extractor is first used in enrollment mode, thus generating a helper string $H$, which can be safely made public without impairing the security of $S$.

The main goal of both fuzzy vault and fuzzy extractor schemes is to mitigate the fuzziness of the biometric traits. In the following paragraphs, some of the principal applications of these schemes are presented. One of them consists in using the string obtained after applying the fuzzy extractor as the secret key for an authentication system [11]. Nevertheless its main problem is the variability of the data, because it is difficult to generate keys with both high stability and high entropy at the same time [18]. Chang *et al.* propose a framework in [7] to generate stable cryptographic keys from biometric data. Their basic idea is to generate more distinguishable features by extending traditional two-class classifiers.

Another application is to associate and retrieve a committed value. In this way, Tong *et al.* [38] propose a method based on fingerprints, which consists of a fuzzy extractor and of a secure sketch [11]. The technique used to extract features of the fingerprints was FingerCode. It is an original technique which creates a stable and ordered representation of the fingerprints.

The problem of generating fuzzy extractors from continuous distributions was addressed by Buhan et al. in [5]. Secure sketches related to fuzzy extractors for fingerprints [1], face [42], and multimodal systems [37] have been proposed. Finally, protocols for secure authentication in remote applications [4, 6] have also been published.

A modified fuzzy extractor scheme was proposed in [3] where a fixed permutation is applied to the iriscode bits before hashing. It is also shown how to use a biometric secret in a remote error-tolerant authentication protocol that does not require any storage on the client side.

## 3. A crypto-biometric scheme based on iris-templates with fuzzy extractors

In this section, we propose a crypto-biometric fuzzy extractor for iris templates. The target of this scheme is to associate a secret or a key, $S$, to a user in such a way that $S$ will be returned to the user if and only if the user's identity is correctly verified. This verification is carried out using the user's iris templates.

Our scheme resembles partially the schemes proposed by Lee *et al.* [24] and by Tong *et al.* [38]. The main difference with respect to the first is that the IrisCode is generated from a set of iris features by clustering, a technique not used in our scheme; and the scheme proposed in [38] uses fingerprints whereas we use iris templates.

As most of these class of schemes, the proposed by us is divided into two phases: The enrollment phase and the verification phase.

The secret, $S$, is hidden in the coefficients of a polynomial of degree $d$ during the enrollment phase, so that for recovering this secret, that polynomial must be re-constructed in the verification phase. This re-construction is made by using Lagrange interpolation process.

### *3.1. Enrollment phase*

The inputs for this phase are the user's iris template, $\mathfrak{B}$, and the secret, $S$, chosen by the user herself. In general, $\mathfrak{B}$ is a binary sequence and $S$ is a symmetric key. The outputs are two sets of values, $H$ and $\Delta$, which can be stored in the database without impairing the security of $S$ and $\mathfrak{B}$.

The different stages of this phase are the following:

1. The secret $S$ is represented in a numerical base, $b$; for example, 10, 16, 256, 512, etc. If $S$ has $d+1$ digits in that base, these digits are considered as the coefficients of a polynomial $p(x)$ of degree $d$:

$$S = (s_0, s_1, \ldots, s_d)_b \rightsquigarrow p(x) = s_0 + s_1 x + s_2 x^2 + \ldots + s_d x^d.$$

2. Next, $n$ random integer numbers, $x_i \in \mathbb{Z}$, are generated in order to compute $n$ pairs of points, $(x_i, y_i)$, verifying $p(x)$, i.e., $y_i = p(x_i)$, $0 \leq i \leq n-1$. The parameter $n$ determines the level of fuzziness of the system, so $n$ must be greater than $d$.
3. The points are encoded into $n$ codewords, in order to mitigate the intra-user variability, by using a Reed-Solomon code:

$$C = \{c_0, c_1, \ldots, c_{n-1}\}.$$

4. A hash function, $\mathfrak{h}$, is applied to the elements of $C$ to obtain a new set

$$H = \{\mathfrak{h}(c_0), \mathfrak{h}(c_1), \ldots, \mathfrak{h}(c_{n-1})\}.$$

5. The iris template of each user, $\mathfrak{B}$, is divided into $n$ parts, so that $b_i$, $0 \leq i \leq n-1$ is a subsequence of $\mathfrak{B}$:

$$\mathfrak{B} = b_0 \parallel b_1 \parallel \ldots \parallel b_{n-1}.$$

6. Then, computing $\delta_i = c_i - b_i$ a new set is obtained:

$$\Delta = \{\delta_0, \delta_1, \ldots, \delta_{n-1}\}.$$

Finally, once the helper data ($H$ and $\Delta$) are determined, they are stored in the database. Moreover, the control parameters are made public to be able to perform the verification phase, namely: The base in which $S$ is represented, $b$; the degree of $p(x)$, $d$; the number of considered points, $n$; the Reed-Solomon code parameters, and the hash function used, $\mathfrak{h}$.

The computation complexity of the whole enrollment phase is polynomial, as all operations performed in this phase (representation of a number in a base, computation of pairs of points verifying a polynomial, calculation of a hash value, codification by an error correction code, and substraction of bits) have polynomial time.

### *3.2. Verification phase*

Since all the parameters and the helper data are made public, the retrieval of the secret $S$ is strongly conditioned by the correct verification and authentication of the user. So, recovering the secret directly depends on the matching phase where the stored and the query template are compared.

The first task of this phase is to obtain the control parameters previously stored in the enrollment phase. Then, the following stages are performed:

1. The query iris template, $\bar{\mathfrak{B}}$, is divided into $n$ parts, as it was done in the enrollment phase:
$$\bar{\mathfrak{B}} = \bar{b}_0 \parallel \bar{b}_1 \parallel \ldots \parallel \bar{b}_{n-1}.$$
2. From $\Delta$ and $\bar{\mathfrak{B}}$, the set values $\bar{c}_i = \delta_i + \bar{b}_i$ is computed:
$$\bar{C} = \{\bar{c}_0, \bar{c}_1, \ldots, \bar{c}_{n-1}\}.$$
Note that each value $\bar{c}_i$ is supposed to be similar to the corresponding value $c_i \in C$, but with some differences due to the intra-user variability between $\mathfrak{B}$ and $\bar{\mathfrak{B}}$.
3. The same hash function, $\mathfrak{h}$, is applied to the elements of the $\bar{C}$, obtaining the set $\bar{H}$, which is compared with the set $H$.
In this comparison, at least $d+1$ coincidences are necessary to continue with the process, due to the fact that Lagrange interpolation method is used to rebuild the polynomial $p(x)$ of degree $d$. These comparisons shows the importance of the parameter $n$, because it will determine the rate of errors admitted in the system due to the intra-user variability.
4. The coincident values are decoded by means of the Reed-Solomon code and $d+1$ points, $(x_i, y_i)$, at least, are obtained.
5. The Lagrange interpolation method used with the points obtained determines the polynomial $p(x)$.
6. Finally, from the coefficients of $p(x)$, the secret $S$ is directly determined and returned to the user.

As all operations performed in the verification phase, including the Lagrange interpolation method [35], are similar to those performed in the enrollment phase, the verification phase has a polynomial computation complexity.

A possible drawback of the proposed scheme could happen if the bitlength of $b_i$ is small due to fact that the sets $H$ and $\Delta$ are public. In fact, by using a trial and error procedure, a possible attacker could determine $d+1$ random values of $b_i$, which could lead to compute the necessary correct values for $c_i = \delta_i + b_i$, and consequently to obtain the secret value $S$. This drawback could be avoided just by adjusting the bitlength of each $b_i$ to make unfeasible their calculation.

## 4. Experimental results

Two experiments have been performed on the crypto-biometric scheme described, in order to measure its behavior with respect to the inter- and intra-user variability.

The first experiment measures the intra-user variability of the iris templates in order to calculate the False Rejection Rate, *FRR*, of the system, i.e., the probability of a known user to be rejected by the system.

The second experiment measures the inter-user variability of the iris templates in the scheme, in order to compute the False Acceptance Rate, *FAR*, i.e., the probability of an attacker to be mistakenly accepted as a known user by the system.

As these ratios are being calculated, the efficiency degree of the scheme is being measured. In this way it will be possible to select the best values of some of the control parameters to improve the security level of the scheme. Most of these parameters depend on the bitlength of the secret or key $S$, denoted by $|S|$. Note that as the base $b$ is fixed, the value of the degree $d$ of the polynomial $p(x)$ will be different each time $|S|$ is changed.

We consider $b = 2^9 = 512$ as the fixed value for the base. The bitlengths selected for $S$ are $|S| \in \{64, 128, 192, 256\}$ since they are the standard sizes for cryptographic symmetric keys currently in use [28]. For each value of $|S|$, a corresponding value of $d$ is obtained. Such values are shown in Table 1.

Table 1: Values of $d$ depending on the bitlength of $S$.

| bitlength of $S$: $\vert S\vert$ | 64 | 128 | 192 | 256 |
|---|---|---|---|---|
| Value of $d$ | 7 | 14 | 21 | 28 |

In this way, when the comparison is carried out in the verification phase, the number of coincidences between $H$ and $\bar{H}$ necessary to validate the user's iris template differs as the value of $d$ does. In this sense, $d$ can be deemed as an indicator of how "easy" would be to recognize a known user or an attacker.

The rest of the control parameters will be the same (or similar for the Reed-Solomon codes) in every experiment in order to do a trustworthy comparison. We consider $\mathfrak{h}$ = SHA-512 and the fuzziness parameter $n = 384$.

The results of this experiment have been obtained by using the whole CASIA database of iris images for 106 users. Each one of these users has 7 different images of her irises and the corresponding templates of all these 742 images have been extracted by using the algorithm presented in [10], which provides templates with 384 bytes each. The extraction of the templates in this case follows the following stages:

- *Pupil border detection.* The property used to detect the pupil border is the characteristic color of the pupil, since it is the part of the iris image with more level of black color.
- *Iris and sclerotic border detection.* To do this tasks, an integro-differential operator adapted to detect circles in images is defined and used.
- *Eyelid detection.* The same operator used in the previous stage is applied to locate the upper and bottom parts of the eyelid.
- *Normalization of the image obtained.* This process is a sampling of the image depending on the polar coordinates centered in the pupil, with the angle formed by the iris' centre and the pupil's centre as the initial angle.

### *4.1. Intra-user variability analysis*

The experiment is carried out in the following way: Each one of the 7 templates of the 106 users is considered as the input of the verification phase and it is compared against the stored data, $H$, of the rest of the templates of the same user. The result of these comparisons will show the level of similarity between all the templates of a single user. These coincidences determine whether the secret can be retrieved or not, and the results of this experiment will be used to measure the False Rejection Rate of the scheme.

Table 2 shows the number of success, $E_i$, $1 \leq i \leq 106$, for each user $U_i$, compared to herself, when $|S| = 192$ (and $d = 21$). Here a "success" represents a comparison with at least 22 coincidences. Note that the maximum of such comparisons for each user is 28, that is, the number of distinct combinations of 2 items selected from a pool of 7 with replacement. It can be seen that the number of success appearing more often is, in fact, 28.

The fact that no all users have 28 successes reveals the problems that the intra-user variability can cause.

Table 2: Number of intra-user successes for the set of users when $|S| = 192$ and $d = 21$.

| Number of success, $E_i$ | Number of users |
|---|---|
| 28 | 42 |
| 27 | 14 |
| 26 | 9 |
| 25 | 11 |
| 24 | 5 |
| 23 | 6 |
| 22 | 2 |
| 21 | 6 |
| 20 | 3 |
| 19 | 1 |
| 18 | 3 |
| 17 | 1 |
| 16 | 1 |
| 13 | 1 |
| 11 | 1 |
| TOTAL | 106 |

From the values $E_i$ given in Table 2, the False Rejection Rate of the scheme can be computed for the case $|S| = 192$. To do this, the rate of coincidences is calculated first, which will determine the Genuine Acceptance Rate ($GAR = 1 - FRR$):

$$GAR_{192} = \sum_{i=1}^{106} \frac{E_i}{28 \cdot 106} = \frac{2681}{2968} = 0.90330 \simeq 90.33\%$$

$$FRR_{192} = 1 - GAR = 1 - 0.90330 = 0.09669 \simeq 9.67\%.$$

Similar computations as those ones shown in Table 2 were carried out for each one of the different values of $|S|$. Table 3 shows the values of Genuine Acceptance Rate and False Rejection Rate for each value of $|S|$.

Table 3: Values of $GAR$ and $FRR$ for each value of $|S|$.

| $\lvert S\rvert$ | 64 | 128 | 192 | 256 |
|---|---|---|---|---|
| $GAR$ | 98.72% | 96.26% | 90.33% | 81.87% |
| $FRR$ | 1.28% | 3.74% | 9.67% | 18.13% |

From these results, it can be said that the lower the value of $d$, the fewer the number of necessary coincidences, and consequently more comparisons verify the Lagrange interpolation request for each user, therefore $FRR$ is lower.

### *4.2. Inter-user variability analysis*

In this second experiment we compare each one of the 7 templates of each user with the rest of the templates of the other users and mark each time the number of coincidences which is greater than the bound determined by the degree, $d$, of the polynomial under consideration.

The results of this experiment will be used to measure the False Acceptance Rate of the scheme. This measurement is crucial for the security of the scheme because it hints the chances that an attacker could be accepted by the system as a real legal user. Should this happen, the secret value of the legal user would be compromised.

Table 4 shows the number of success for $|S| = 192$ and $d = 21$ for each one of the users compared with the rest of the users. The experiment compares each one of the 7 templates of each user with all the templates of the rest of 105 users, so there are $105 \cdot 7 = 735$ comparisons for each template. Hence, the number of comparisons for each user is $735 \cdot 7 = 5,145$. Finally, the total comparisons is $735 \cdot 7 \cdot 106$ as there are 106 users.

Once all these data is collected, the next step is the calculation of $FAR$, which is for the case $|S| = 192$:

$$FAR_{192} = \frac{\sum_{i=1}^{106} D_i}{735 \cdot 7 \cdot 106} = 0.04416 \simeq 4.42\%.$$

Table 5 shows the values of False Acceptance Rate for each value of $|S|$.

From the results obtained, it can be stated that the chosen parameters seem to be appropriated from a biometric point of view for practical applications. Nevertheless, a commitment between the security parameters of the scheme and the biometric efficiency should be studied in a near future.

In this sense, if for example $n = 32$ is taken, the same values of $d$ considered in our experiments (see Table 1) could be still used since $n > d$. At the same time, this value of $n$ would lead, considering the same extraction template procedure, to a bitlength of each $b_i$ of 96. As it was mentioned, to break the proposed scheme it is necessary to determine $d + 1$ correct values of $b_i$, and in this case

Table 4: Number of inter-user successes, $D_i$, for each user $U_i$, with at least $d+1 = 22$ coincidences and $|S| = 192$.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| $U_i$ | # 1 | # 2 | # 3 | # 4 | # 5 | # 6 | # 7 |
| $D_i$ | 276 | 218 | 271 | 38 | 67 | 133 | 265 |
| $U_i$ | # 8 | # 9 | # 10 | # 11 | # 12 | # 13 | # 14 |
| $D_i$ | 90 | 78 | 86 | 73 | 398 | 293 | 140 |
| $U_i$ | # 15 | # 16 | # 17 | # 18 | # 19 | # 20 | # 21 |
| $D_i$ | 387 | 515 | 274 | 777 | 94 | 29 | 54 |
| $U_i$ | # 22 | # 23 | # 24 | # 25 | # 26 | # 27 | # 28 |
| $D_i$ | 625 | 295 | 54 | 380 | 385 | 348 | 188 |
| $U_i$ | # 29 | # 30 | # 31 | # 32 | # 33 | # 34 | # 35 |
| $D_i$ | 99 | 52 | 62 | 244 | 102 | 36 | 174 |
| $U_i$ | # 36 | # 37 | # 38 | # 39 | # 40 | # 41 | # 42 |
| $D_i$ | 177 | 260 | 73 | 154 | 62 | 72 | 473 |
| $U_i$ | # 43 | # 44 | # 45 | # 46 | # 47 | # 48 | # 49 |
| $D_i$ | 272 | 26 | 115 | 471 | 108 | 171 | 788 |
| $U_i$ | # 50 | # 51 | # 52 | # 53 | # 54 | # 55 | # 56 |
| $D_i$ | 75 | 18 | 123 | 351 | 589 | 141 | 31 |
| $U_i$ | # 57 | # 58 | # 59 | # 60 | # 61 | # 62 | # 63 |
| $D_i$ | 224 | 17 | 83 | 441 | 285 | 55 | 52 |
| $U_i$ | # 64 | # 65 | # 66 | # 67 | # 68 | # 69 | # 70 |
| $D_i$ | 49 | 82 | 379 | 93 | 420 | 100 | 443 |
| $U_i$ | # 71 | # 72 | # 73 | # 74 | # 75 | # 76 | # 77 |
| $D_i$ | 413 | 281 | 331 | 547 | 789 | 204 | 63 |
| $U_i$ | # 78 | # 79 | # 80 | # 81 | # 82 | # 83 | # 84 |
| $D_i$ | 53 | 803 | 215 | 595 | 621 | 154 | 335 |
| $U_i$ | # 85 | # 86 | # 87 | # 88 | # 89 | # 90 | # 91 |
| $D_i$ | 437 | 289 | 21 | 630 | 50 | 47 | 159 |
| $U_i$ | # 92 | # 93 | # 94 | # 95 | # 96 | # 97 | # 98 |
| $D_i$ | 18 | 21 | 127 | 83 | 173 | 105 | 61 |
| $U_i$ | # 99 | # 100 | # 101 | # 102 | # 103 | # 104 | # 105 |
| $D_i$ | 49 | 97 | 418 | 122 | 171 | 540 | 41 |
| $U_i$ | # 106 | | | | | | |
| $D_i$ | 149 | | | | | | |

Table 5: Values of $FAR$ for each value of $|S|$.

| bitlength of $S$: $\|S\|$ | 64 | 128 | 192 | 256 |
|---|---|---|---|---|
| $FAR$ | 70.38% | 22.9% | 4.42% | 0.97% |

the probability to obtain any of such values is $2^{-96}$. In other words, finding only one of the $d+1$ values of $b_i$ is equivalent to break a symmetric cryptosystem with a key of 96 bits length.

## 5. The Hamming distance and the variability

It is important to find a way to eliminate or mitigate the intra-user variability and consequently the inter-user variability, because if the different templates of the same user are very similar, it would be easier to recognize her as a legal user and not as an attacker. At the same time, it would be more difficult to mix up two different users.

In order to explain the similarities and differences of users, a Hamming distance study of the extracted templates has been performed [12]. To do this, one user is selected and each one of her templates is compared with the rest, measuring the Hamming distance between them. In this way 21 comparisons were done for each user. Table 6 shows the average of the Hamming distance for each user.

Table 6: Average of Hamming distances, $H_i$, for the templates of each user, $U_i$.

| $U_i$ | # 1 | # 2 | # 3 | # 4 | # 5 | # 6 | # 7 |
|---|---|---|---|---|---|---|---|
| $H_i$ | 821.80 | 1056.00 | 839.14 | 997.23 | 1024.47 | 1024.19 | 1133.90 |
| $U_i$ | # 8 | # 9 | # 10 | # 11 | # 12 | # 13 | # 14 |
| $H_i$ | 1169.23 | 1097.42 | 1027.71 | 1028.57 | 1128.47 | 1146.28 | 986.47 |
| $U_i$ | # 15 | # 16 | # 17 | # 18 | # 19 | # 20 | # 21 |
| $H_i$ | 1037.23 | 1010.09 | 792.19 | 1148.19 | 994.66 | 1034.57 | 925.80 |
| $U_i$ | # 22 | # 23 | # 24 | # 25 | # 26 | # 27 | # 28 |
| $H_i$ | 911.52 | 1216.38 | 1064.19 | 1060.76 | 1008.57 | 897.61 | 788.38 |
| $U_i$ | # 29 | # 30 | # 31 | # 32 | # 33 | # 34 | # 35 |
| $H_i$ | 1145.61 | 1009.04 | 872.28 | 836.85 | 976.38 | 868.66 | 1041.61 |
| $U_i$ | # 36 | # 37 | # 38 | # 39 | # 40 | # 41 | # 42 |
| $H_i$ | 1136.00 | 1110.85 | 865.42 | 1168.19 | 1458.66 | 1077.90 | 880.09 |
| $U_i$ | # 43 | # 44 | # 45 | # 46 | # 47 | # 48 | # 49 |
| $H_i$ | 1203.90 | 933.80 | 910.47 | 1004.09 | 1113.71 | 1278.19 | 892.76 |
| $U_i$ | # 50 | # 51 | # 52 | # 53 | # 54 | # 55 | # 56 |
| $H_i$ | 926.09 | 973.61 | 1056.47 | 977.71 | 1004.76 | 1001.42 | 1053.33 |
| $U_i$ | # 57 | # 58 | # 59 | # 60 | # 61 | # 62 | # 63 |
| $H_i$ | 956.47 | 1223.42 | 832.09 | 899.23 | 962.00 | 774.00 | 1064.38 |
| $U_i$ | # 64 | # 65 | # 66 | # 67 | # 68 | # 69 | # 70 |
| $H_i$ | 1078.76 | 1227.04 | 1144.38 | 1039.90 | 1094.00 | 1005.33 | 1074.09 |
| $U_i$ | # 71 | # 72 | # 73 | # 74 | # 75 | # 76 | # 77 |
| $H_i$ | 1003.04 | 894.66 | 931.71 | 1031.71 | 1071.80 | 876.00 | 1160.67 |
| $U_i$ | # 78 | # 79 | # 80 | # 81 | # 82 | # 83 | # 84 |
| $H_i$ | 1099.90 | 913.23 | 929.14 | 934.28 | 842.57 | 955.71 | 790.95 |
| $U_i$ | # 85 | # 86 | # 87 | # 88 | # 89 | # 90 | # 91 |
| $H_i$ | 899.33 | 912.19 | 1060.76 | 920.19 | 1310.76 | 922.38 | 698.47 |
| $U_i$ | # 92 | # 93 | # 94 | # 95 | # 96 | # 97 | # 98 |
| $H_i$ | 1072.95 | 915.52 | 1016.19 | 934.47 | 1060.47 | 992.66 | 894.00 |
| $U_i$ | # 99 | # 100 | # 101 | # 102 | # 103 | # 104 | # 105 |
| $H_i$ | 1220.85 | 1127.33 | 915.33 | 1190.85 | 1124.00 | 1004.66 | 1084.28 |
| $U_i$ | # 106 | | | | | | |
| $H_i$ | 966.19 | | | | | | |

This study permits one to establish a relationship between the number of corrected comparisons and the Hamming distances of each user (Table 6) for the different values of $|S|$. Subsequently, the correlation coefficients for every value of $|S|$ were computed and can be seen in Table 7.

From these coefficients, it can be stated that, in all cases, the Hamming distance is inversely related to the success in the comparison of the templates of the users, and consequently to their intra-user variability. This fact is not surprising because if two templates of a given user are far enough (measured with the Hamming distance), her templates will have many differences. In this

Table 7: Correlation coefficients between the number of comparisons and Hamming distances, for each value of $|S|$.

| bitlength of $S$: $\lvert S\rvert$ | 64 | 128 | 192 | 256 |
|---|---|---|---|---|
| Correlation coefficients | -0.459 | -0.638 | -0.783 | -0.862 |

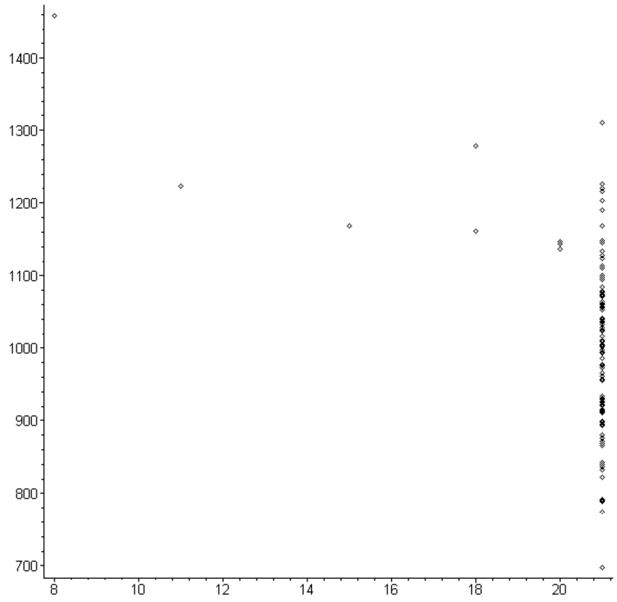

Figure 1: Scatter plot between user's number of coincidences and Hamming distance with $|S| = 64$.

situation it turns difficult to identify such a user.

Another fact that can be seen from the performed analysis is that the relationship between the Hamming distance and the success in the comparison of the templates of the users increases as $|S|$ does. The event happens due to the different values of the degree $d$.

If $d$ is small, as is the case with $|S| = 64$, where $d = 7$, the Hamming distance is not critical in the comparisons, since the user would be identified with only $d + 1 = 8$ coincidences. However, the Hamming distance becomes more relevant as $d$ increases, due to the corresponding increase in the number of necessary coincidences to successfully identify a user.

Figures 1-4 show the relation of the users, represented by the number of their coincidences, with the average of the Hamming distance of each user for the different values of $|S|$.

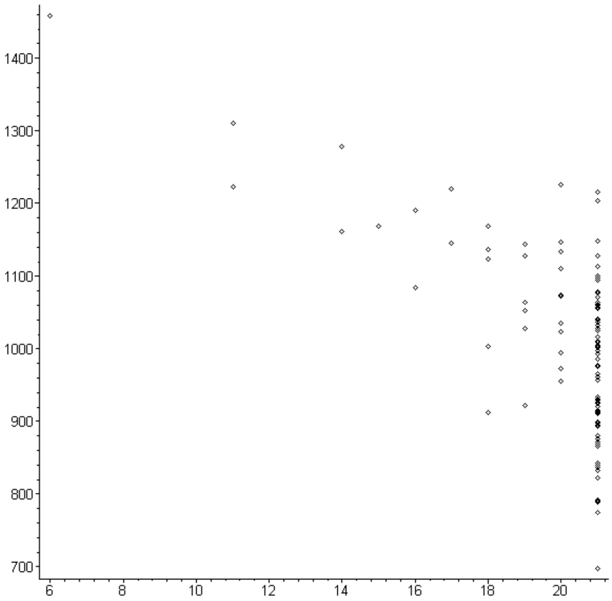

Figure 2: Scatter plot between user's number of coincidences and Hamming distance with $|S| = 128$.

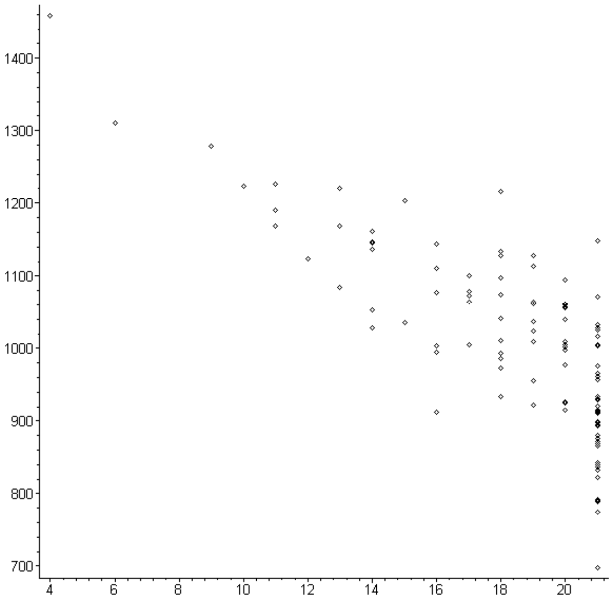

Figure 3: Scatter plot between user's number of coincidences and Hamming distance with $|S| = 192$.

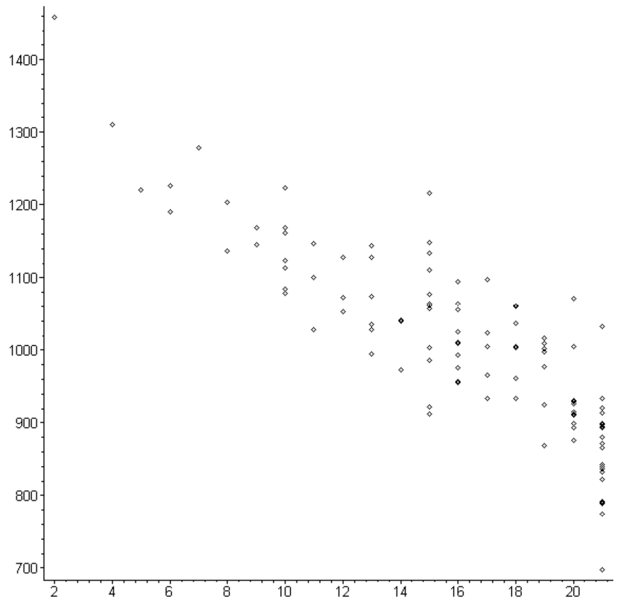

Figure 4: Scatter plot between user's number of coincidences and Hamming distance with $|S| = 256$.

## 6. Conclusions

We have presented a crypto-biometric scheme for hiding and retrieving a secret (or a key) by using the iris template of a user and a fuzzy extractor. The scheme has two phases: The enrollment and the verification phase. In the first one, the secret is hidden by using an iris template of the user; whereas in the verification phase, the secret is returned to the user if her query template is considered similar enough to the reference template used in the enrollment phase.

We have used different sets of parameters in order to study and decide which one of them provides better results in relation to the inter- and intra-user variability. The study was carried out using the whole CASIA iris data base.

Regarding to the efficiency of the scheme, it can be stated that the lower the bitlength of the secret, the easier to recognize a known user; but, at the same time, the lower the bitlength of the secret, the easier to accept an attacker as a legitimate user, as the values of $FAR$ show.

From the experimental results, considering the four different values of the bitlength of the analyzed secret, two different results can be obtained depending on the point of view:

- According to the inter-user variability, the best result is obtained when the bitlength of the secret is $|S| = 256$. In this case, the False Acceptance Rate is $FAR = 0.97\%$,

- According to the intra-user variability, the best result is obtained when the bitlength of the secret is $|S| = 64$. In this case, the value obtained for the ratio of False Rejection Rate is $FRR = 1.28\%$

Hence, there is no "optimal" length for the secret, rather it must be selected according to the security requirements of the application where the scheme will be used. In any case, the most balanced bitlength for the secret or the key, taking into account the values of $FAR$ and $FRR$ is $|S| = 192$, which provides the following values:

$$FAR = 4.42\%, \quad FRR = 9.67\%, \quad GAR = 90.33\%.$$

When the Hamming distance has been considered in order to clarify the values of $FAR$ and $FRR$, it can be stated that the Hamming distance is inversely related to the intra-user variability. Moreover, Hamming distance and success in the comparison of the templates of the users are more strongly related as the bitlength of the secret grows.

The results obtained and the possible drawback mentioned previously open two new future lines of research: The development of new methods for extracting iris templates with more precision or more bitlength than those used nowadays, and a deeper study of a commitment between the security parameters and the biometric efficiency.

### Acknowledgement

Authors would like to thank the anonymous reviewers for their helpful comments and professor R. Durán for his valuable suggestions. This work has been partially supported by Ministerio de Ciencia e Innovación (Spain) under the grant TIN2011-22668.

### References

[1] A. Arakala, J. Jeffers, K.J. Horadam, Fuzzy Extractors for Minutiae-Based Fingerprint Authentication, In: S.W. Lee, S.Z. Li (Eds.), Proc. Advances in Biometrics, Springer, New York, NY, Lecture Notes in Computer Science 4642 (2007) 760–769.

[2] BIT, Biometric Ideal Test, `http://biometrics.idealtest.org/findTotalDbByMode.do?mode=Iris`

[3] X. Boyen, Reusable Cryptographic Fuzzy Extractors, In: Proc. 11th ACM Conf. Computer and Communications Security, ACM, New York, NY, 2004, pp. 82–91.

[4] X. Boyen, Y. Dodis, J. Katz, R. Ostrovsky, A. Smith, Secure Remote Authentication Using Biometric Data, In: R.Cramer (Ed.), Proc. Advances in Cryptology, Springer, New York, NY, Lecture Notes in Computer Science 3494 (2005) 147–163.

[5] I.R. Buhan, J.M. Doumen, P.H. Hartel, R.N.J. Veldhuis, Fuzzy Extractors for Continuous Distributions, In: Proc. ACM Symp. Information, Computer and Communications Security, ACM, New York, NY, 2007, pp. 353–355.

[6] I.R. Buhan, J.M. Doumen, P.H. Hartel, R.N.J. Veldhuis, Secure Ad-hoc Pairing with Biometrics: SAfE, In: Proc. First Int. Workshop Security for Spontaneous Interaction, 2007, pp. 450–456.

[7] Y. Chang, W. Zhang, T. Chen, Biometric-based cryptographic key generation, In: Proc. IEEE Conf. Multimedia and Expo, 2004, pp. 2203–2206.

[8] X. Chen, Q. Wub, F. Zhang, H. Tian, B. Wei, B. Lee, H. Lee, K. Kim, New receipt-free voting scheme using double-trapdoor commitment, Information Sciences 181 (2011) 1493–1502.

[9] J.F. Connolly, E. Granger, R. Sabourin, An adaptive classification system for video-based face recognition, Information Sciences (in press), available online 6 March 2010, doi:10.1016/j.ins.2010.02.026.

[10] E. Díez Laiz, C. Sánchez Ávila, Sistema criptobiométrico basado en iris para esquemas Diffie-Hellman con curva elíptica (ECDH), In: Proc. Congreso de Métodos Numéricos en Ingeniería, 2009.

[11] Y. Dodis, L. Reyzin, A. Smith, Fuzzy Extractors, Security with Noisy Data 5 (2007) 93–111.

[12] Y. Dodis, R. Ostrovsky, L. Reyzin, A. Smith, Fuzzy Extractors: How to Generate Strong Keys from Biometrics and Other Noisy Data, SIAM Journal Computing 38 (1) (2008) 97–139.

[13] H. Guo, Z. Li, Y. Muc, X. Zhang, Provably secure identity-based authenticated key agreement protocols with malicious private key generators, Information Sciences 181 (2011) 628–647.

[14] D. Hidalgo, O. Castillo, P. Melin, Type-1 and type-2 fuzzy inference systems as integration methods in modular neural networks for multimodal biometry and its optimization with genetic algorithms, Information Sciences 179 (2009) 2123–2145.

[15] F. Hernández Álvarez, L. Hernández Encinas, Security Efficiency Analysis of a Biometric Fuzzy Extractor for Iris Templates, In: A. Herrero, P. Gastaldo, R. Zunino, E. Corchado (Eds.), Proc. 2nd Int. Workshop Computational Intelligence in Security for Information Systems, Springer, New York, NY, Advances in Intelligent and Soft Computing 63 (2009) 163–170.

[16] B. Huang, J. Wu, D. Zhang, N. Li, Tongue shape classification by geometric features, Information Sciences 180 (2010) 312–324.

[17] A.K. Jain, R. Bolle, S. Pankanti, Biometrics: Personal Identification in Networked Society, Springer, New York, NY, 1999.

[18] A.K. Jain, K. Nandakumar, A. Nagar, Biometric Template Security, Journal on Advances in Signal Processing 8 (2) (2008), 1–17.

[19] A.K. Jain, A. Ross, S. Prabhakar, Fingerprint matching using minutiae and texture features, In: Proc. Int. Conf. Image Processing, 2001, pp. 282–285.

[20] A. Juels, M. Sudan, A fuzzy vault scheme, Designs, Codes and Cryptography 38 (2) (2006) 237–257.

[21] A. Juels, M. Wattenberg, A fuzzy commitment scheme, In: Proc. 6th ACM Conf. Computer and Communications Security, ACM, New York, NY, 1999, pp. 28–36.

[22] F. Laguillaumie, D. Vergnaud, Time-selective convertible undeniable signatures with short conversion receipts, Information Sciences 180 (2010) 2458-2475.

[23] X.H. Le, S. Lee, Y.K. Lee, H. Lee, M. Khalid, R. Sankar, Activity-oriented access control to ubiquitous hospital information and services, Information Sciences 180 (2010) 2979–2990.

[24] Y.J. Lee, K. Bae, S.J. Lee, K.R. Park, J. Kim, Biometric Key Binding: Fuzzy Vault based on Iris Images, In: S.W. Lee, S.Z. Li (Eds.), Proc. Advances in Biometrics, Springer, New York, NY, Lecture Notes in Computer Science 4642 (2007) 800–808.

[25] C.K. Li, D.S. Wong, Signcryption from randomness recoverable public key encryption, Information Sciences 180 (2010) 549–559.

[26] H. Li, J. Zhang, Z. Zhang, Generating cancelable palmprint templates via coupled nonlinear dynamic filters and multiple orientation palmcodes, Information Sciences 180 (2010) 3876–3893.

[27] D. Maltoni, D.Maio, A.K. Jain, S. Prabhakar, Handbook of Fingerprint Recognition, Springer, New York, NY, 2003.

[28] A. Menezes, P. van Oorschot, S. Vanstone, Handbook of Applied Cryptography, CRC Press, Boca Raton, FL, 1997.

[29] National Institute of Standards and Technology, Advanced Encryption Standard (AES), Federal Information Processing Standard Publication 197, 2001.

[30] National Institute of Standards and Technology, Recommendation for the Triple Data Encryption Algorithm (TDEA) Block Cipher, Special Publication SP 800-67 Version 1.1, 2008.

[31] National Institute of Standards and Technology, Digital Signature Standard (DSS), Federal Information Processing Standard Publication 186-3, 2009.

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

[33] R.L. Rivest, A. Shamir, L.M. Adleman, A method for obtaining digital signatures and public-key cryptosystems, Commun. ACM 21 (2) (1978) 120-126.

[34] C. Sánchez Ávila, R. Sánchez-Reillo, Two different approaches for iris recognition using Gabor filters and multiscale zero-crossing representation, Pattern Recognition 38 (2) (2005) 231–240.

[35] A. Schrijver, Theory of Linear and Integer Programming, John Wiley & Sons, New York, NY, 1998.

[36] D.R. Stinson, Cryptography: Theory and Practice, $3^{rd}$ ed., Chapman & Hall/CRC, Boca Raton, FL, 2006.

[37] Y. Sutcu, Q. Li, N. Memon, Secure Biometric Templates from Fingerprint-Face Features, In: Proc. IEEE Conf. Computer Vision and Pattern Recognition 2007, pp. 1–6.

[38] V.V. Triem Tong, H. Sibert, J. Lecoeur, M. Girault, Biometric Fuzzy extractors made practical: A proposal based on FingerCodes, In: S.W. Lee, S.Z. Li (Eds.), Proc. Advances in Biometrics, Springer, New York, NY, Lecture Notes in Computer Science 4642 (2007) 604-613.

[39] U. Uludag, S. Pankanti, S. Prabhakar, A.K. Jain, Biometric Cryptosystems: Issues and Challenges, Proc. of the IEEE 92 (6) (2004) 948–960.

[40] J. Yu, R. Hao, F. Kong, X. Cheng, J. Fan, Y. Chen, Forward-secure identity-based signature: Security notions and construction, Information Sciences 181 (2011) 648–660.

[41] H. Yuan, F. Zhang, X. Huang, Y. Mud, W. Susilo, L. Zhang, Certificateless threshold signature scheme from bilinear maps, Information Sciences 180 (2010) 4714–4728.

[42] X. Zhou, Template Protection and its Implementation in 3D Face Recognition Systems, In: Proc. SPIE Conf. Biometric Technology for Human Identification IV, 2007, pp. 214–225.