IEEE Signal Processing Society | Transactions on Information Forensics and Security

Theoretical and Practical Boundaries of Binary Secure Sketches

Journal:	*Transactions on Information Forensics & Security*
Manuscript ID:	T-IFS-00616-2008.R1
Manuscript Type:	Regular Paper
Date Submitted by the Author:	n/a
Complete List of Authors:	Bringer, Julien; Sagem Sécurité Chabanne, Herve; Sagem Sécurité Cohen, Gérard; Institut TELECOM, TELECOM ParisTech Kindarji, Bruno; Sagem Sécurité; Institut TELECOM, TELECOM ParisTech Zémor, Gilles; Université de Bordeaux I, Institut de Mathématiques de Bordeaux
EDICS:	WAT-THEO Information-theoretic limits < WATERMARKING AND DATA HIDING/EMBEDDING, BIO-PROT Biometric protocols < BIOMETRICS, MOD-CHAN Channel and network models < SYSTEM MODELS

Theoretical and Practical Boundaries of Binary Secure Sketches

J. Bringer, H. Chabanne, G. Cohen, B. Kindarji and G. Zémor

Abstract

Fuzzy commitment schemes, introduced as a link between biometrics and cryptography, are a way of handling biometric data matching as an error correction issue. We focus here on finding the best error-correcting code with respect to a given database of biometric data.

We propose a method that models discrepancies between biometric measurements as an erasure and error channel, and we estimate its capacity. We then show that two-dimensional iterative min-sum decoding of properly chosen product codes almost reaches the capacity of this channel. This leads to practical fuzzy commitment schemes that are close to theoretical limits. We test our techniques on public iris and fingerprint databases and validate our findings.

Index Terms

Iris, fingerprint, biometrics, secure sketches, boundaries, min-sum decoding.

EDICS: MOD-CHAN, WAT-THEO, BIO-PROT.

I. Introduction

With the growing use of biometric recognition systems comes the need to secure and protect the privacy associated to biometric data. Juels and Wattenberg's fuzzy commitment scheme [2] uses Error Correcting Codes and was introduced to handle differences occurring between two captures of biometric data. Many papers give applications of this technique for cryptographic purposes [3], [4], [5], [6], [7], [8], [9], [10],

J. Bringer, H. Chabanne and B. Kindarji are with Sagem Sécurité, Osny, France. This work was partially supported by the French ANR RNRT project BACH.

G. Cohen and B. Kindarji are with TELECOM ParisTech, Département Informatique et Réseaux, Paris, France.

G. Zémor is with Institut de Mathématiques de Bordeaux, Université de Bordeaux I, Bordeaux, France.

A preliminary version of this work was presented at the IEEE conference Biometrics: Transactions, Applications and Systems, 2007, cf. [1].

[11], [2], [12], [13] but only a few investigate what are the best codes for this decoding problem and how to find them.

Secure sketches have been experimented with several biometrics. Applications to face recognition [14] and to fingerprints [15] are proposed that make use of BCH codes and reliable bit extraction. In a different way, Daugman *et al.* experimented with the use of a concatenated Hadamard – Reed-Solomon code for iris recognition [16].

In this paper, we explain how to estimate the theoretical performance limit of a secure sketch, applied to binary biometric data, at a given code dimension. We then describe an efficient iterative decoding algorithm on product codes, which leads to near-optimal performance in our experiments on iris and fingerprint recognition.

A. Biometric Matching and Errors Correction

1) Biometric Templates: The issue of the best codes we can expect for biometric secure sketches is addressed here, in the context of binary biometric features, as is the case for iris recognition systems, see [17] – more details on iris recognition are also available in [18]. So we focus our paper on iris biometrics but it is also relevant to fingerprints.

Indeed, our techniques are applicable to recent methods which involve transforming real-valued templates into discrete ones so as to use secure sketches (cf. [14], [15]). A nice feature of discretization is that Hamming distance becomes an efficient tool.

Note that in our setting, all templates will be binary arrays, even though our theoretical approach also applies to arrays over any finite field.

2) Matching and Error Rates: Typically, a biometric-based recognition scheme consists of two phases. First, in the enrolment phase, a biometric template b is measured from a user U and then registered in a token or a database. The second phase – the verification – captures a new biometric sample b' from U and compares it to the reference data via a matching function. According to some underlying measure μ and some recognition threshold τ, b' will be accepted as a biometric measure of U if $\mu(b, b') \leq \tau$, else rejected. Mainly two kinds of errors are associated to this scheme: False Reject (**FR**), when a matching user, i.e. a legitimate user, is rejected; False Acceptance (**FA**), when a non-matching one, e.g. an impostor, is accepted.

Note that, when the threshold increases, the **FR**'s rate (**FRR**) decreases while the **FA**'s rate (**FAR**) grows, and conversely.

3) Error Correcting Codes and Secure Sketches: Our methods will resort to information theory and coding. Some basic definitions are given hereafter. For more background, notations and classical results, the reader is referred to [19] and [20] in these two fields respectively.

Let $\mathcal{H}$ be the collection of all binary N-tuples, $\mathcal{H} = \{0,1\}^N = \mathbb{F}_2^N$, where $\mathbb{F}_2 = \{0,1\}$.

- The $\oplus$ *operator* is the canonical exclusive-or over $\mathbb{F}_2$:

$$a \oplus b = \begin{cases} 0 \text{ if } a = b \\ 1 \text{ if } a \neq b \end{cases}$$

- The *Hamming distance* over $\mathcal{H}$ is the metric distance defined as the number of binary differences between two elements, i.e.

$$d_{\mathcal{H}}(u,v) = \sum_{i=1}^{N} (u_i \oplus v_i).$$

  Equipped with the Hamming distance, $\mathcal{H}$ is called the *Hamming space* of length N.
- An *Error Correcting Code* (**ECC**) over $\mathcal{H}$ is a subset $C \subset \mathcal{H}$; elements of C are called *codewords*.
- An (N, S, d) binary **ECC** is an error correcting code C over $\mathcal{H}$ with S elements such that for all distinct codewords c_1 and c_2, $d_{\mathcal{H}}(c_1, c_2) \geq d$. N is called the length of C, S is the size of C and d, the smallest Hamming distance between two distinct codewords, is the minimum distance.
- A binary *linear* error correcting code C is a vector subspace of $\mathbb{F}_2^N$. By linearity, the minimum distance d_{min} of C is now the minimum weight among non-zero codewords, where the *weight* of a vector x is its distance to the vector $\mathbf{0}$. When k is the dimension of the subspace C, C is denoted by $[N, k, d_{min}]_2$. Here, the *correction capacity* t of C is the radius of the largest Hamming ball for which, for any $x \in \mathbb{F}_2^N$, there is at most one codeword in the ball of radius t centred on x. Clearly, $t = \lfloor (d_{min} - 1)/2 \rfloor$.

Assuming that the templates live in $\mathcal{H}$, the main idea of fuzzy schemes, as introduced in [2], is to convert the matching step into an error-correcting one. Let C be an (N, S, d) **ECC** in $\mathcal{H}$.

- During the enrolment phase, one stores $z = c \oplus b$, where c is a random codeword in C,
- During the verification phase, one tries to correct the corrupted codeword $z \oplus b' = c \oplus (b \oplus b')$. Note that when the Hamming distance $d_{\mathcal{H}}(b, b')$ is small, recovering c from $c \oplus (b \oplus b')$ is, in principle, possible.

The correction capacity of C may thus be equal to τ if we do not want to alter the **FRR** and the **FAR** of the system. Unfortunately, the difference between two measures of one biometric source can be very important, whereas the correction capacity of a code is structurally constrained.

The fuzzy commitment scheme is then an error-tolerant authentication scheme which follows the above method with the use of a committed value. The main goal is to protect the storage of biometric data involved in an authentication biometric system. Let h be a cryptographic one-way function, and let us store $h(c)$ in the enrolment phase, together with $z = c \oplus b$. The authentication will be a success if the verification returns a codeword c' such that $h(c') = h(c)$. An illustration of the scheme is provided in Fig. 1.

This construction has been formalized in [11] under the name *Secure Sketch*. Informally, a secure sketch is made of a probabilistic Sketching Function SS, which "hides" the biometric template, and a deterministic Recovery Function Rec which recovers the original template if not too many errors have occurred.

Several constraints are studied in the literature, e.g. in [2], [9], [11], to achieve the protection of b while z is publicly known. These works show that the code C must be adapted to the entropy of biometrics and it leads in fact to a trade-off between correction capacity of C and the security properties of the scheme. Moreover, the size S of C should not be too small, to prevent z from revealing too much information about the template b: indeed the probability for an attacker to "guess" b out of $z = c \oplus b$, with the computation of $z \oplus \tilde{c}$ from the choice of a random codeword $\tilde{c}$, is lower bounded by $1/S$. This issue is also discussed in Sec. IV-D.

B. Organization of this Work

We first look for theoretical limits. In Sec. II, we formalize our problem by transforming a database of biometric data into a binary erasure-and-error channel. We then give a method for finding an upper

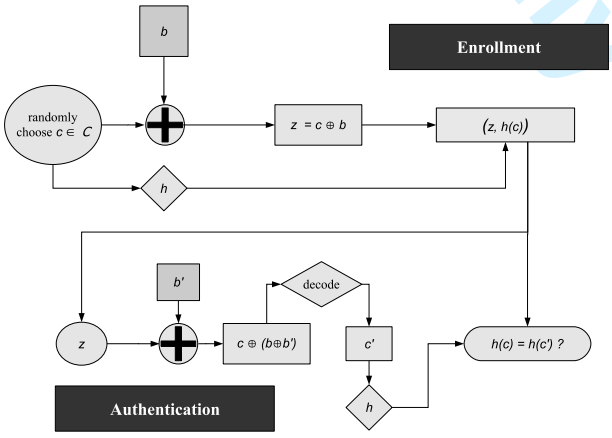


Figure 1. The Fuzzy Commitment Scheme [2]

bound on the underlying error correction capacity, and explain how to transpose this result into bounds on **FAR** and **FRR**.

Section III introduces the biometric datasets – two for iris biometric and one for fingerprints – we use in our experiments; we then present the practical bounds deduced from our result.

In Sec. IV, we illustrate our method by describing a very efficient construction with iterative min-sum decoding of product codes, and we provide parameters that put our performances close to the theoretical limit for those databases.

Section V concludes.

II. Theoretical Optimal Correction

A. Model

We consider two separate channels with a noise model based on the differences between any two biometric templates.

- The first channel, called the **matching channel**, is generated by errors $b \oplus b'$ where b and b' come from the same user U.
- The second channel, the **non-matching channel**, is generated by errors where b and b' come from different biometric sources.

In a practical biometric system, the number of errors in the **matching channel** is on average lower than in the **non-matching channel**.

Moreover, the templates are not restricted to a constant length. Indeed, when a sensor captures biometric data, we want to keep the maximum quantity of information but it is rarely possible to capture the same amount of data twice – for instance an iris may be occulted by eyelids – hence the templates are of variable length. This variability can be smoothed by forming a list of erasures, i.e. the list of coordinates where they occur. More precisely, in coding theory, an erasure in the received message is an unknown symbol at a known location. We thus have an erasure-and-error decoding problem on the **matching channel**. Simultaneously, to keep the **FAR** low, we want a decoding success to be unlikely on the **non-matching channel**: to this end we impose bounds on the correction capacity.

In the sequel, we deal with binary templates with at most N bits and assume, for the theoretical analysis that follows, that the probabilities of error and erasure on each bit are independent, i.e. we work on a binary input memoryless channel. Note that resorting to interleaving makes this hypothesis valid for all practical purposes.

B. Taking into Account Errors and Erasures

As we take into account erasures into our biometric model, we also need to slightly enhance Juels and Wattenberg's scheme. Let (b, m) and (b', m') be two biometric templates, b, b' denoting the known information, and m, m' the list of erasures, in the way IrisCodes are represented. We can represent some $(b, m) \in \{0,1\}^N \times \{0,1\}^N$ by a ternary vector $\tilde{b} \in \{0,1,\epsilon\}^N$, where the third symbol ϵ represents an erasure.

The updated **xor** rule on $\{0,1,\epsilon\}$ is very similar to the usual one: we define $x \tilde{\oplus} x'$ to be $x \oplus x'$ if x and x' are bits, and ϵ if one of x, x' is ϵ.

In order to protect c and b, the updated sketch will simply be the sum $z = c \tilde{\oplus} \tilde{b}$. The verification step will also use the $\tilde{\oplus}$ operation to combine z with $\tilde{b'}$ into $z \tilde{\oplus} \tilde{b'}$. The decoding can then proceed to correct incorrect bits and erasures.

C. Theoretical Limit

Our goal is to estimate the capacity, in the Shannon sense [21], of the matching channel when we work with a code of a given dimension. Namely, we want to know the maximum number of errors and erasures between two biometric measures that we can manage with secure sketches for this code.

Starting with a representative range of matching biometric data, the theorem below gives an easy way to estimate the lowest achievable **FRR**. The idea is to check whether the best possible code with the best generic decoding algorithm, i.e. a **maximum-likelihood** (**ML**) decoding algorithm which systematically outputs the most likely codeword, would succeed in correcting the errors.

Theorem 1: Let $k \in \mathbb{N}^*$, C be a binary code of length N and size 2^k, and m a random received message, from a random codeword of C, of length N with w_n errors and w_e erasures. Assume that C is an optimal code with respect to N and k, equipped with an **ML** decoder.

If $\frac{w_n}{N - w_e} > \theta$ then m is only decodable with a negligible probability for a large N, where θ is such that the Hamming sphere of radius $(N - w_e)\theta$ in $\mathbb{F}_2^{N-w_e}$, i.e. the set $\{x \in \mathbb{F}_2^{N-w_e},\ d_{\mathcal{H}}(x, \mathbf{0}) = (N - w_e)\theta\}$, contains 2^{N-w_e-k} elements. □

Proof: In the case of errors only (i.e. no erasures) with error-rate $p := w_n/N$, the canonical second theorem of Shannon asserts that there are families of codes with (transmission) rate $R := k/N$ coming arbitrarily close to the *channel capacity* $\kappa(p)$, decodable with ML-decoding and a vanishing (in N) word error probability P_e.

In this case, $\kappa(p) = 1 - h(p)$, where $h(p)$ is the (binary) entropy function ($\log$'s are to the base 2):

$$h(x) = -x \log x - (1-x) \log(1-x).$$

Furthermore, P_e displays a threshold phenomenon: for any rate arbitrarily close to, but above capacity and any family of codes, P_e tends to 1 when N grows.

Equivalently, given R, there exists an error-rate threshold of

$$p = h^{-1}(1-R),$$

h^{-1} being the inverse of the entropy function.

Back to the errors-and-erasures setting now. Our problem is to decode to the codeword nearest to the received word on the *non-erased* positions.

Thus we are now faced with a punctured code with length $N - w_e$, size 2^k, transmission rate $R' := k/(N-w_e)$ and required to sustain an error-rate $p' := \frac{w_n}{N-w_e}$.

By the previous discussion, if

$$p' > \theta := h^{-1}(1-R'),$$

NO code and NO decoding procedure exist with a non-vanishing probability of success.

To conclude the proof, use the classical Stirling approximation for the size of a Hamming sphere of radius αM in $\mathbb{F}_2^M$ by $2^{h(\alpha)M}$. ■

This result allows us to estimate the correcting capacity of a biometric matching channel with noise and erasures under the binary input memoryless channel hypothesis.

Indeed applying Theorem 1 to the **matching channel** gives a lower-bound on the **FRR** achievable (i.e. the *best* **FRR**), whereas applying it to the **non-matching channel** gives an upper-bound of the **FAR** (say the *worst* **FAR**).

Corollary 1: For a given biometric authentication system based on a binary secure sketch of length N and dimension k, and a given biometric database $\mathcal{B} = \{b_i\}$, let the function $f_{N,k}$ be $f_{N,k}(\tilde{y}) = \frac{w_n}{N-w_e} - h^{-1}\left(1 - \frac{k}{N-w_e}\right)$, with w_n the number of 1's occuring in $\tilde{y}$ and w_e the number of ϵ. Define $p^G_{N,k}(x)$ (resp. $p^I_{N,k}(x)$) as the probability density of results of all genuine (resp. impostor) comparisons $f_{N,k}(\tilde{b} \tilde{\oplus} \tilde{b'})$ for $b, b' \in \mathcal{B}$.

Under these hypothesis, the following inequalities stand: $FRR \geq \int_0^{+\infty} p^G_{N,k}(t)dt$ and $FAR \leq \int_{-\infty}^0 p^I_{N,k}(t)dt$.

□

In other words, Corollary 1 can lead to a kind of theoretical ROC curve which is not represented thanks to the classical matching score distributions but with the dimension of the underlying optimal code on the abscissa axis. Therefore, from a given database and a given features extraction scheme – dedicated to discrete representation, it is possible to induce an approximation of the error-rates one can expect from templates of the same quality. In particular, it may help to evaluate the efficiency of the extraction algorithm.

Practical implications of this theorem are illustrated in Sec. III-B.

III. Application to Biometric Data

To explain our approach, we now present the estimation of these optimal performances on several public biometric databases.

A. Our Setting: Data Sets and Templates

We describe here the sets on which we made experiments. We first describe the iris pictures from ICE 2005 and CASIA v1, then the FVC 2000 dataset from which we extracted binary fingerprint templates.

For each dataset, we also represent the boundaries on **FRR** and **FAR**.

1) IrisCodes *and associated databases:* We first made different experiments on iris recognition, which is a very natural target for binary error correcting codes. We chose two public databases:

- The ICE 2005 database: [22], [23].

  It contains 2953 images coming from 244 different eyes. It is taken without modification but one slight correction: the side of the eye 246260 has been switched from left to right. Hence we keep 2953 images. In this dataset the number of images for each eye is variable.
- The CASIA database: [24].

  This is the first version of the Chinese Academy of Science public iris database. It contains 756 pictures of 108 different eyes, with 7 pictures per eye.

A 256-byte (2048 bits) iris template, together with a 256-byte mask, is computed from each iris image using the algorithm reported in [17]; the mask filters out the unreliable bits, i.e. stores the erasures positions of the iris template. The resulting template is called *IrisCode*.

Note that the iris template as computed by this algorithm has a specific structure: [17] reports 249 degrees-of-freedom within the 2048 bits composing the template. As described in [25], [17], [26], the algorithm involves computation of several Gabor filters on separate and local areas of the iris picture. The

picture is normalized onto its polar representation, then divided into areas of regular size. The amplitude information is discarded and the actual bits are the phase quantization of this Gabor-domain representation of the iris image. The ordering of the bits is directly linked to the localization of the area. In practice, the iris code can be represented by an 2D bits-array.

The classical way to compare two iris codes I_1, I_2 with masks M_1, M_2 is to compute the relative Hamming distance

$$\frac{||(I_1 \oplus I_2) \cap M_1 \cap M_2||}{||M_1 \cap M_2||} \tag{1}$$

for some rotations of the second template – to deal with the iris orientation's variation – and to keep the lowest score.

This formula gives the Hamming distance distribution given on Fig. 2(a), where the scores of matching (intra-eyes) and non-matching (inter-eyes) comparisons are represented. We can see that there is an overlap between the two curves, and that the number of errors to handle in the matching channel is large. On iris matching-channel an additional difficulty originates from the number of erasures which varies, for instance for ICE, from 512 to 1977.

Although we know that all bits are not independent and that they do not follow the same distribution (see e.g. [27]), following (1) the typical matching score computation does not use any internal correlations between bits of the iris codes. So in this setting it is coherent to suppose the matching channel to be a binary input memoryless channel with independent bit errors and erasures. It will thus be possible to apply Theorem 1 in this context.

2) Fingerprint Encoding and Associated Database: Traditional fingerprint matching is made thanks to minutiae extraction [28] and comparisons of unordered sets $\mathcal{E}$, $\mathcal{E}'$ of variable length. Using the characteristic function $\chi_{\mathcal{E}}$ – as done in [11], [29] – is a way to translate minutiae into a binary vector of fixed length. The size corresponds to the number of values the coordinates could take. From a set of minutiae, the idea is to construct a vector with all coordinates equal zero except those which are associated with the position of one minutiae. The problem is that this representation is not well-suited for binary secure sketches. Indeed, the metrics associated to the set representation is the symmetric set difference, which does not take into account local distortion due to elasticity of the finger skin. Still, Secure Sketches are easier to construct for the Hamming distance with a q-ary code.

To overcome this difficulty, Tuyls *et al.* [15] describe a smart algorithm, in the line of the previous works [30], [12], to extract stable binary vectors from fingerprints and to apply secure sketches on them. We based our experiments on such a coding, more precisely on an improvement which has been proposed

in [31].

We describe a synthesis of the algorithm below. The main idea is to deal with fingerprint patterns rather than minutiae. It makes use of core-based alignment techniques and pattern features linked to directional fields, thanks to the techniques described in [32], [33], [34]. Moreover, to increase the stability of the vectors, the binary fixed-length strings are generated following some statistics by using several images per user at the enrolment.

For these experiments, the FVC2000 [35] public database (Db. 2) was tested. This data set is made of 800 pictures of 100 different fingers, 8 pictures per finger. The image size is 256 by 364 pixels.

Before the enrolment step, we first align the picture on a fixed point, such as the core, if available. For that, we preprocess the picture in order to extract the core point and an evaluation of the vertical axis. We translate the picture, then adjust it to take care of some possible rotation. In practice, we did prealign the fingerprints to solely test the binarization proposed.

These points are then executed:

1) Picture embedding: to take into account the alignment, we embed the 256×364 pixels picture into a larger picture of 768×1092 to prevent any loss of information after re-alignment.
2) Real-vector extraction: we compute several Gabor filters on the resulting picture, of which we keep only the magnitude. We also compute the directional field of the fingerprint. The concatenation of both computations gives us a real-vector of length $L = 17952$, of which 15968 positions are known to be null, due to the embedding. These positions will be marked as erasures.
3) Binarization: the enrolment is done on several pictures per user and several users; a statistical analysis gives enough information to quantize the vectors by comparing – coordinate after coordinate – the mean value of a user to the mean value of the overall enrolment database. For each user, this gives a vector from $\{0, 1, \epsilon\}^L$.
4) Reliable components selection: for each user, all enrolment vectors are combined into a bit string of fixed length N. This is done by selecting only the N most stable coordinates from the different vectors. As it is likely that real-life pictures never are pre-aligned, it is likely that the null positions will not be the same for each fingerprint capture; this enables to choose $N \geq 17952 - 15968 = 1984$. More details on component selection are given in [31].

Hence we obtain binary templates, together with a mask, of a fixed length. The verification step is quite similar; to get the fresh biometric template, we use the positions selected at the enrolment step, and then compare them with the enrolled vector.

In the sequel, we selected 6 images per finger for the enrolment phase, one 2048-bit template per

enrolled finger is obtained, possibly with some erasures, and the remaining 200 images are kept for verification. As the verification step is done on just one picture, the verification template will always contain at least $2048 - 1984 = 64$ erasures; this is well captured by the decoding algorithm. To increase the overall number of comparisons, we iterate the tests for every choices of 6 images. This gives us a genuine match count of 5600, and an impostor match count of 19800.

Any other biometrics may be used to apply Theorem 1 as soon as we succeed in getting a discrete representation of the templates associated to a Hamming distance classifier.

B. Performances Estimation on these Databases

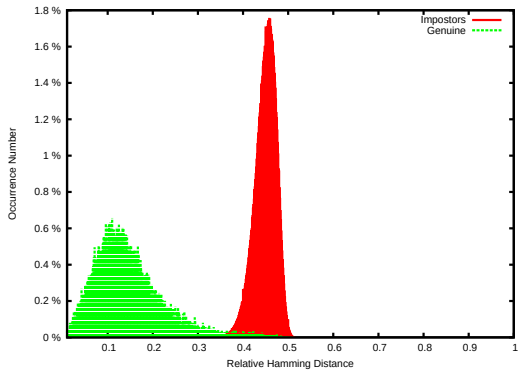
For each one of these databases we represent, in Figures 2, 3 and 4, the relative Hamming distance distribution thanks to Eq. (1) for the matching and the non-matching channel and the corresponding FRR and FAR curves. We have also estimated the optimal performances given by Corollary 1 and the results are drawn in Figures 2(c), 3(c) and 4(c). The curves correspond to the best FRR achievable with respect to the code's dimension and the greatest possible FAR as a function of this dimension.

From the Hamming Distance distributions, it is obvious that, while iris recognition performs well with the IrisCode algorithm, the chosen quantization is not as well adapted to fingerprint matching. Therefore, the different results we shall have will significantly differ.

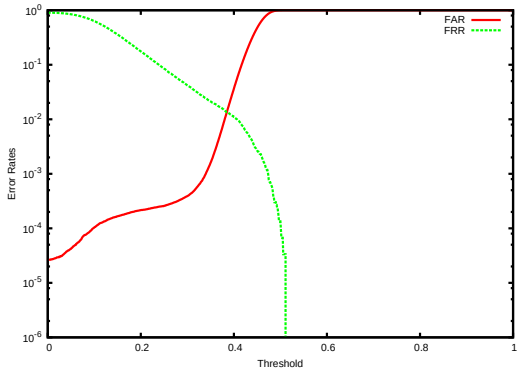
For the three datasets, we see that the ratio of errors to handle to approach the Equal Error Rate – **EER** – is very high, which is a problem for classical correcting codes as it is explained in the next section.

We summed up some of the numerical limits on **FAR** and **FRR** in table I, for dimensions likely to be chosen for practical purposes. A general consequence is that the dimension of the code can not be chosen too high in order to keep good **FR** rates.

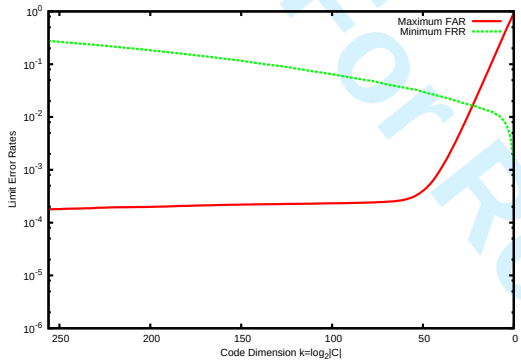
Note that Theorem 1 gives us estimations of the theoretical limits based on asymptotic analysis under a memoryless channel hypothesis, i.e. independent bits. In principle, it could be possible to expect more efficiency without resorting to bit interleaving which in practice makes the channel memoryless. However this would require highly intricate modelling of the matching channel, and it seems unreasonable to expect that the decoding problem would be within reach of present day algorithms.



(a) Hamming distance distributions



(b) FAR and FRR via the Hamming distance (Eq. 1) using a threshold



(c) Worst FAR and best FRR w.r.t. the code dimension

Figure 2. The ICE 2005 Dataset, IrisCodes

Code's dimension	Minimum **FRR**			Maximum **FAR**		
	ICE	CASIA	FVC	ICE	CASIA	FVC
42	$2.49 \cdot 10^{-2}$	$3.15 \cdot 10^{-2}$	$0.59 \cdot 10^{-2}$	$8.14 \cdot 10^{-4}$	$1.13 \cdot 10^{-4}$	$17.88 \cdot 10^{-2}$
64	$3.76 \cdot 10^{-2}$	$4.47 \cdot 10^{-2}$	$1.26 \cdot 10^{-2}$	$2.74 \cdot 10^{-4}$	0	$10.32 \cdot 10^{-2}$
80	$4.87 \cdot 10^{-2}$	$5.77 \cdot 10^{-2}$	$1.93 \cdot 10^{-2}$	$2.57 \cdot 10^{-4}$	0	$7.07 \cdot 10^{-2}$
128	$9.10 \cdot 10^{-2}$	$9.18 \cdot 10^{-2}$	$5.87 \cdot 10^{-2}$	$2.41 \cdot 10^{-4}$	0	$2.67 \cdot 10^{-2}$

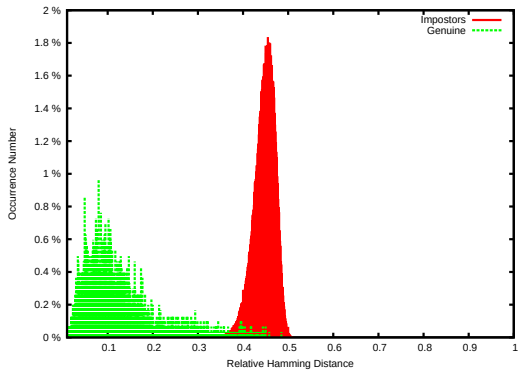
Table I

THEORETICAL LIMITS ON STUDIED DATABASES

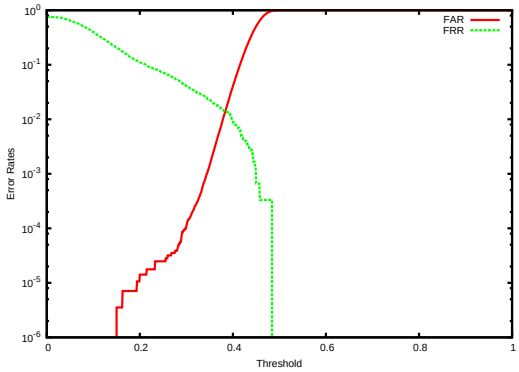
IV. A NEAR OPTIMAL CONSTRUCTION

A. Previous Works

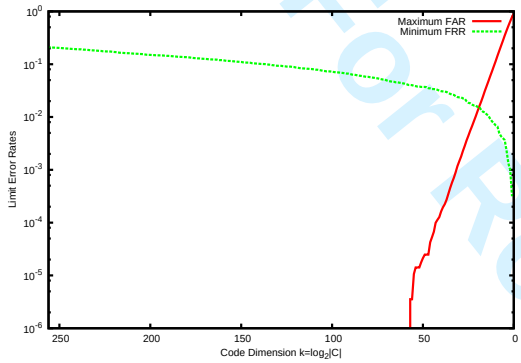
1) Quantization and BCH codes: In known applications of secure sketches to quantized biometrics, for instance [14], [15], the error correcting codes are seen directly to act as a Hamming distance classifier



(a) Hamming distance distributions



(b) FAR and FRR via the Hamming distance (Eq. 1) using a threshold



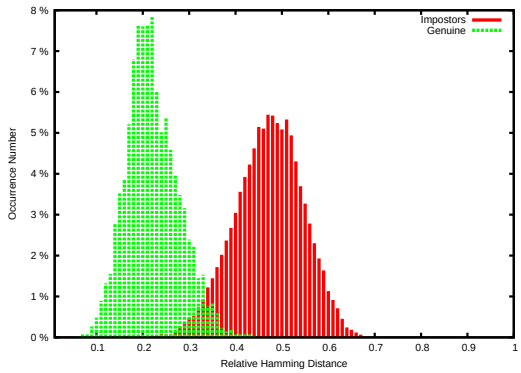
(c) Worst FAR and best FRR w.r.t. the code dimension

Figure 3. The CASIA v1 Dataset, IrisCodes

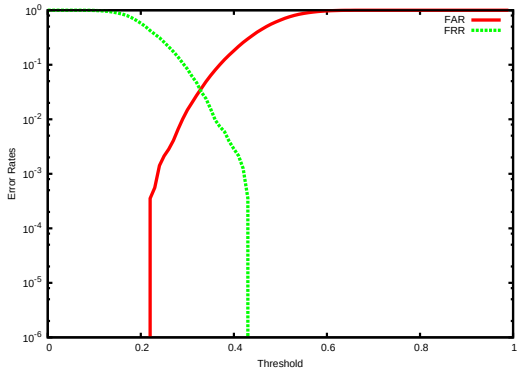
at a given threshold. Hence, the correction capacity naturally corresponds to the threshold we want to attain. To this end, the use of BCH codes is proposed: the advantage is their existence for a wide class of parameters, the main drawback is that the correction capacity is a hard constraint for the dimension.

As an illustration, in [14] the quantization technique is applied to face recognition on two databases, FERET database [36] and one from Caltech [37]. A Hamming distance classifier gives Equal Error Rates of 2.5% and 0.25% respectively for a threshold greater than 0.32 with code length 511. Unfortunately to achieve this minimal distance, the BCH code has dimension 1. A BCH of dimension 40 enables a threshold of 0.185 with a **FRR** greater than 10% and 1% respectively.

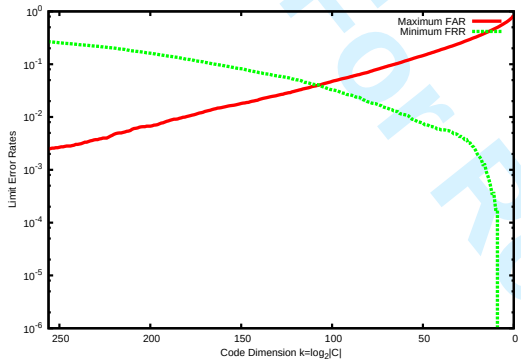
This phenomenon holds in [15] and for our first experiments on the FVC2000 dataset. Following Fig. 4(b), we remark that to achieve a **FRR** better than the EER, the threshold is high: for example, for a rate around 2%, the threshold is near 0.4 which is not realistic with non-trivial BCH codes. To overcome this limitation, we propose in the sequel to use more appropriate codes.



(a) Hamming distance distributions



(b) FAR and FRR via the Hamming distance (Eq. 1) using a threshold



(c) Worst FAR and best FRR w.r.t. the code dimension

Figure 4. The FVC 2000 (Db. 2) Dataset, Binary Encoding

2) IrisCodes *and Concatenated Codes:* More efficient codes are proposed in [16]. The secure sketch scheme is applied with a concatenated error-correcting code combining a Hadamard code and a Reed-Solomon code. More precisely, the authors use a $[32, k_{RS}, 33 - k_{RS}]_{2^7}$ Reed-Solomon code and a $[64, 7, 32]_2$ Hadamard code: a codeword of 2048 bits is in fact constructed as a set of 32 blocks of 64 bits where each block is a codeword of the underlying Hadamard code. As explained in [16], the Hadamard code is introduced to deal with the background errors and the Reed-Solomon code to deal with the bursts (e.g. caused by eyelashes, reflections, ...).

Note that in this scheme, the model is not exactly the same as ours, as the masks are not taken into account. Moreover, the quality of the database used in [16] is better than the public ones we worked with. The mean intra-eye Hamming distance reported in the paper is 3.37% whereas this number becomes 13.9% in the ICE database, which means that we must have a bigger correcting capacity. The inter and intra-eyes distributions reported by the authors is drawn on Fig. 5.

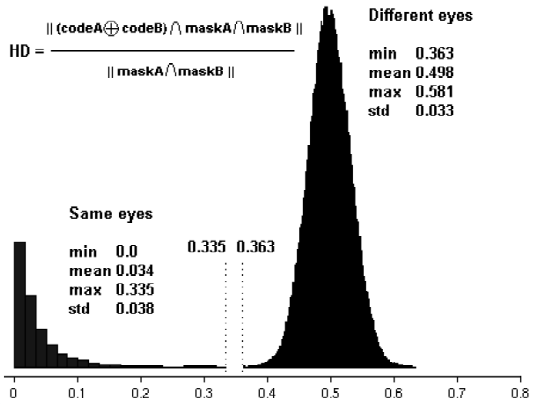


Figure 5. Hamming distance distributions from [16]

Even if [16] reports very good results on their experiments with a 700-image database , the codes do not seem appropriate in our case as the same parameters on the ICE database gave us a too large rate of **FR** (e.g. 10% of **FR** with 0.80% of **FA)**, even for the smallest possible dimension of the Reed-Solomon code when $t_{RS} = 15$.

To sum up, with respect to the Hamming distance distribution in figures 2, 3, 4, we need to find correcting codes with higher correction capacity. To achieve performances closer to the theoretical estimation given in section III-B is also a great motivation.

B. Description of the Two-Dimensional Iterative Min-Sum Decoding Algorithm

We now describe a very efficient algorithm which will help us to overcome the difficulties mentioned above.

For a linear code with a minimum distance d_{min}, we know that an altered codeword with w_n errors and w_e erasures can always be corrected, disregarding decoding complexity issues, provided that $2w_n + w_e < d_{min}$.

Classical algebraic decoding of BCH codes and concatenated Reed-Solomon codes achieve this bound, but not more. This upper bound is however a conservative estimate: it has been known since Shannon's days that it is possible in principle to correct many more errors and erasures, all the way to the channel capacity. In practice, *iterative decoding* algorithms are now known to be capable of achieving close-to-capacity performance, for such code families as LDPC or turbo codes. It is therefore natural to try and bring in iterative decoding to improve on secure sketches that use algebraic decoders. LDPC codes and

turbo codes are however not usually designed for such noisy channels as the type we have to deal with: in particular, classical turbo codes are known not to behave well under high noise. We have therefore chosen to use product codes: this is because under the high noise condition particular to biometrics, we will be dealing with codes of small dimension so that we can apply maximum-likelihood decoding (exhaustive search) to the constituent codes and alternate between both decoders with an iterative process. This will yield a particularly efficient blend of iterative decoding and exhaustive search.

We now describe product codes together with the specific iterative decoding algorithm we will use. A product code $C = C_1 \otimes C_2$ is constructed from two codes: $C_1[N_1, k_1, d_1]_2$ and $C_2[N_2, k_2, d_2]_2$. The codewords of C can be viewed as matrices of size $N_2 \times N_1$ whose rows are codewords of C_1 and columns are codewords of C_2, see Fig. 6.

This yields a $[N_1 \times N_2, k_1 \times k_2, d_1 \times d_2]_2$ code. When k_1 and k_2 are small enough for C_1 and C_2 to be decoded exhaustively a very efficient iterative decoding algorithm is available, namely the *min-sum* decoding algorithm. Min-sum decoding of LDPC codes was developed by Wiberg [38] as a particular instance of message passing algorithms. In a somewhat different setting it was also proposed by Tanner [39] for decoding generalized LDPC (Tanner) codes. The variant we will be using is close to Tanner's algorithm and is adapted to product codes. Min-sum is usually considered to perform slightly worse than the more classical sum-product message passing algorithm on the Gaussian, or binary-symmetric channels, but it is specially adapted to our case where knowledge of the channel is poor, and the emphasis

$$c = \begin{pmatrix} c_{1,1} & \dots & c_{1,j} & \dots & c_{1,n_1} \\ & & \vdots & & \\ c_{i,1} & \dots & c_{i,j} & \dots & c_{i,n_1} \\ & & \vdots & & \\ c_{n_2,1} & \dots & c_{n_2,j} & \dots & c_{n_2,n_1} \end{pmatrix}$$

$$\forall i \in [0, n_2], (c_{i,1}, c_{i,2}, \dots, c_{i,n_1}) \in C_1$$

$$\forall j \in [0, n_1], (c_{1,j}, c_{2,j}, \dots, c_{n_2,j}) \in C_2$$

Figure 6. A codeword of the product code $C_1 \otimes C_2$ is a matrix where each line is a codeword of C_1 and each column a codeword of C_2

is simply to use the Hamming distance as the appropriate basic cost function.

Let (x_{ij}) be a vector of $\{0,1\}^{N_1 \times N_2}$. The min-sum algorithm associates to every coordinate x_{ij} a cost function κ_{ij} for every iteration of the algorithm. The cost functions are defined on the set $\{0,1\}$. The initial cost function κ_{ij}^0 is defined by $\kappa_{ij}^0(x) = 0$ if the received symbol on coordinate (ij) is x and $\kappa_{ij}^0(x) = 1$ if the received symbol is $1-x$.

A *row* iteration of the algorithm takes an *input* cost function κ_{ij}^{in} and produces an *output* cost function κ_{ij}^{out}. The algorithm first computes, for every row i and for every codeword $c = (c_1 \dots c_{N_1})$ of C_1, the *sum*

$$\kappa_i(c) = \sum_{j=1}^{N_1} \kappa_{ij}^{in}(c_j)$$

which should be understood as the cost of putting codeword c on row i. The algorithm then computes, for every i, j, κ_{ij}^{out} defined as the following *min*, over the set of codewords of C_1,

$$\kappa_{ij}^{out}(x) = \min_{c \in C_1, c_j = x} \kappa_i(c).$$

This last quantity should be thought of as the minimum cost of putting the symbol x on coordinate (ij) while satisfying the row constraint.

A *column* iteration of the algorithm is analogous to a row iteration, with simply the roles of the row and column indexes reversed, and code C_2 replacing code C_1. Precisely we have

$$\kappa_j(c) = \sum_{i=1}^{N_2} \kappa_{ij}^{in}(c_i) \tag{2}$$

and

$$\kappa_{ij}^{out}(x) = \min_{c \in C_2, c_i = x} \kappa_j(c).$$

The algorithm alternates row and column iterations as illustrated by Fig. 7. After a given number of iterations (or before, if we find a codeword) it stops, and the value of every symbol x_{ij} is put at $x_{ij} = x$ if $\kappa_{ij}^{out}(x) < \kappa_{ij}^{out}(1-x)$. If $\kappa_{ij}^{out}(x) = \kappa_{ij}^{out}(1-x)$ then the value of x_{ij} stays undecided (or erased).

The following theorem is fairly straightforward and illustrates the power of min-sum decoding.

Theorem 2: If the number of errors is less than $d_1 d_2/2$, then two iterations of min-sum decoding of the product code $C_1 \otimes C_2$ recover the correct codeword. □

Proof: (Sketch)

Without loss of generality, the correct codeword is the all-zero vector. Suppose that after the second iteration the algorithm prefers 1 to 0 in some position (i, j). This means that the cost (2) $\kappa_j(c)$ of some

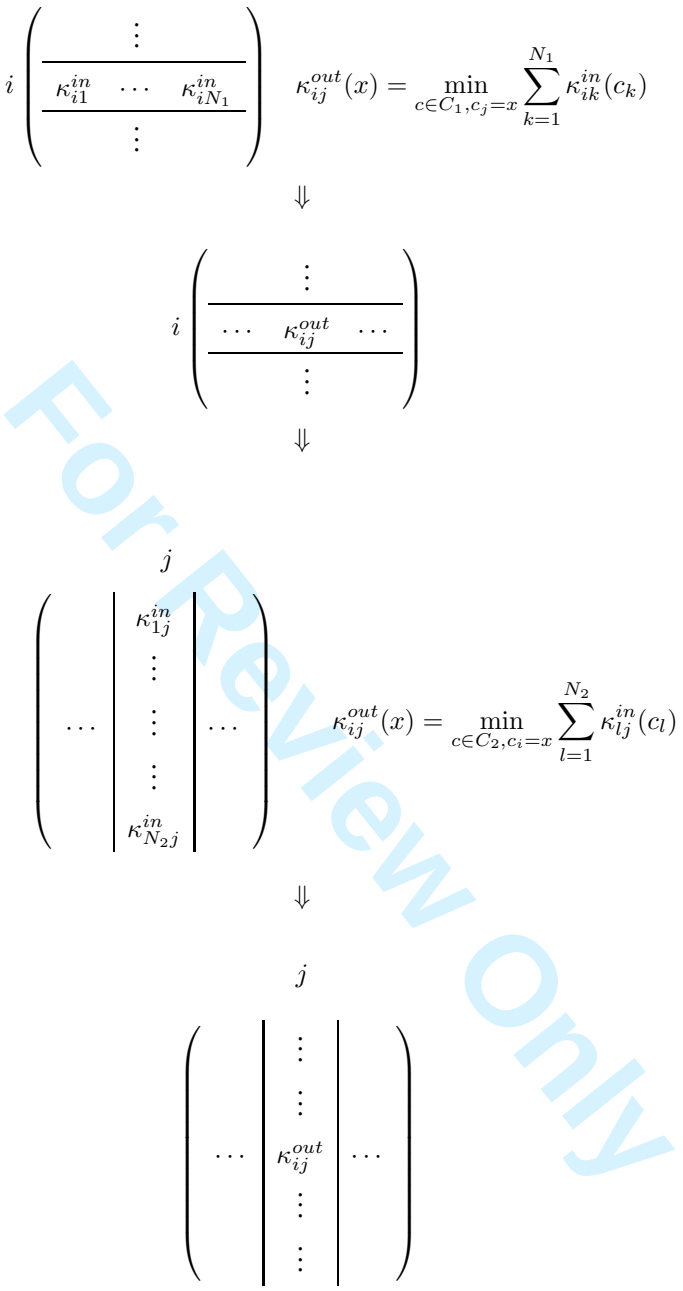


Figure 7. A row iteration followed by a column one

non-zero codeword c of C_2 is smaller than the cost $\kappa_j(0)$ of the zero column vector. Now the cost $\kappa_j(c)$ of putting codeword c in column j is equal to the Hamming distance between the received vector (x_{ij})

and a vector $\mathbf{x}_c$ that has c in column j and only rows belonging to C_1. The cost $\kappa_j(0)$ of putting the zero vector in column j is equal to the Hamming distance between the received vector (x_{ij}) and a vector $\mathbf{x}_0$ that has only zeros in column j and only rows belonging to C_1. Now since c belongs to C_2 it has weight at least d_2, therefore the Hamming distance between $\mathbf{x}_c$ and $\mathbf{x}_0$ is at least $d_2 d_1$, and $\mathbf{x}_c$ has at least d_2 rows of weight at least d_1 and at distance at least d_1 from the corresponding rows of $\mathbf{x}_0$. Therefore, if the received vector (x_{ij}) is closer to $\mathbf{x}_c$ than to $\mathbf{x}_0$, it must have weight at least $d_1 d_2/2$. ■

C. Experiments and Results

To validate the algorithm described in section IV-B, we now present the results of experiments on the public biometric databases introduced in Sec. III-A, where we succeed in obtaining some correction performances close to the theoretical limit.

We have experimented the algorithm on these databases with a particular choice for the code. In fact, the product code is constructed to fit with an array of 2048 bits, by using Reed-Muller codes [40], [41] of order 1 which are known to have good weight distributions. A binary Reed-Muller code of order 1 in m variables, abbreviated as $RM(1, m)$, is an $[2^m, m+1, 2^{m-1}]_2$ code. We chose to combine the $RM(1, 6)$ with the $RM(1, 5)$, leading to a product code of dimension 42 and codewords of length 64×32.

The overall size of the code could appear small from a cryptographic point of view, but following the theoretical analysis of section II-C, it is difficult to expect much more while achieving a low **FRR** on a practical biometric database. Achievable error rates are drawn in Sec. III-B for each database we studied.

The density of errors and erasures in an IrisCode can be very high in some regions, such as areas where eyelashes occlude the iris. The same goes for the fingerprint for which the captured area differs significantly between two measures, leading to high-erasures regions. Therefore, we also added a randomly chosen interleaver to break the biometric structure and increase the efficiency of the decoding algorithm.

- In so doing, we succeeded in obtaining for ICE a **FRR** of about 5.62% for a very small **FAR** (strictly lower than 10^{-5}). This is very close to the error rates obtained in a classical matching configuration. Note that in contrast Eq. (1) and Fig. 2(a) only give a **FAR** of about 10^{-4} for a similar **FRR**. .
- In the CASIA case, the algorithm gave us a **FRR** of 6.65% and 0 **FA**. A basic Hamming distance classifier would not give zero **FA** for a **FRR** less than 20%.
- For fingerprint from FVC2000 dataset, it yielded a **FRR** of 2.73% and a **FA** rate of 5.53%, which is also a very good result for a binary encoding scheme.

In all cases, the correction rates are relatively close to the theoretical results from Table I, and so the algorithm succeeds in achieving near-optimal results. We also noted that, unexpectedly, decoding performances are more accurate than using a basic Hamming score such as Eq. (1) with a fixed threshold for differentiating between matching and non-matching pairs. This underlines the fact that even though Hamming scores give decent results for binary matching, the associated classifier is suboptimal and can be overtaken by more elaborate techniques such as our decoding algorithm, or alternative matching functions that have been put forward recently, *e.g.* [42].

D. Cautions and Limitations

Remember that Theorem 1 is deduced from an asymptotic behaviour, thus to obtain better results, we probably need to increase the length of the templates. Moreover, the base assumption for the computation of the threshold θ is that errors and erasures occur independently and with the same probability. This assumption is far from true in practice, thus the theoretical limit on the error rates obtained by Corollary 1 should give slightly smaller False Reject Rates.

Moreover, even though we achieved near-limit results, we must not neglect some warnings for the use of Secure Sketches as a way to secure biometrics templates.

First of all, as it was noted in [3], a biometric database that would be secured thanks to Secure Sketches would not protect its users' privacy against forward verification. In a few words, if someone gains access to a biometric template b_0, it is easy for him to check whether it corresponds to a previously enrolled individual or not. As the biometrics we focused on – iris and fingerprints – are hardly private and secret, this is a flaw to seriously consider.

The error rates on secure sketches are more than just an artefact from the classical biometric behaviour: they lead to a security gap if secure sketches are used as they were presented in [2]. Indeed, to decode a sketch $(z, h(c))$ stored in a database, an attacker can try to decode every $b' \oplus z$ for b' a template from a collection of biometric measures. This collection can be an independent database the attacker collected for his personal use, or any public or secret biometric database. Whenever he obtains some codeword c', he can compare $h(c')$ with $h(c)$. If the comparison is successful, the attacker deduces $c = c'$, and thus $b = z \oplus c$. This event is likely to happen with probability **FAR**, which we can upperbound by the estimation given by Corollary 1.

Recall that a cryptographic application is nowadays considered as secure enough if the best attack known to break it takes about 2^{80} operations to be successful. If no more consolidation is done on the Fuzzy Commitment Scheme, there exists a vulnerability that gives access to b and c with about $1/\mathbf{FAR} =$

$2^{-\log_2(\mathbf{FAR})}$ operations, *i.e.* way less than what would be acceptable. We thus strongly discourage the use of Secure Sketches without further protection, such as [5], [4], [31].

V. Conclusion

This article demonstrates the inherent limits of error-correction based matching. We derived explicit upper bounds on the correction capacity of secure sketches, and we validated our theoretical results on two public iris databases and one fingerprint database. We then showed how the two-dimensional iterative min-sum decoding algorithm achieves correction performance close to the optimal decoding rate.

We believe that our techniques are also of great interest to other biometrics when the number of errors to manage and correct is quite large.

This paper shows a numerical constraint on the usual performance-security trade-off of secure sketches. Future work in this domain includes finding nearer-limit codes and decoding algorithms as well as improving the reliability of biometrics templates.

References

[1] J. Bringer, H. Chabanne, G. Cohen, B. Kindarji, and G. Zémor, "Optimal iris fuzzy sketches," in *Biometrics: Theory, Applications, and Systems, 2007. BTAS 2007. First IEEE International Conference on*, Sept 2007.

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

[3] J. Bringer, H. Chabanne, and Q. D. Do, "A fuzzy sketch with trapdoor." *IEEE Transactions on Information Theory*, vol. 52, no. 5, pp. 2266–2269, 2006.

[4] J. Bringer and H. Chabanne, "An authentication protocol with encrypted biometric data," *AFRICACRYPT*, vol. LCNS 5023, pp. 109–124, 2008.

[5] J. Bringer, H. Chabanne, M. Izabachène, D. Pointcheval, Q. Tang, and S. Zimmer, "An application of the Goldwasser-Micali cryptosystem to biometric authentication." in *ACISP*, ser. Lecture Notes in Computer Science, J. Pieprzyk, H. Ghodosi, and E. Dawson, Eds., vol. 4586. Springer, 2007, pp. 96–106.

[6] G. Cohen and G. Zémor, "Generalized coset schemes for the wire-tap channel: application to biometrics," in *IEEE International Symposium on Information Theory, Chicago*, 2004, p. 46.

[7] ——, "The wire-tap channel applied to biometrics," in *International Symposium on Information Theory and Applications, Parma*, 2004.

[8] ——, "Syndrome-coding for the wiretap channel revisited," in *ITW'06, IEEE Information Theory Workshop, Chengdu*, 2006, pp. 33–36.

[9] X. Boyen, Y. Dodis, J. Katz, R. Ostrovsky, and A. Smith, "Secure remote authentication using biometric data." in *EUROCRYPT*, ser. Lecture Notes in Computer Science, R. Cramer, Ed., vol. 3494. Springer, 2005, pp. 147–163.

[10] G. I. Davida and Y. Frankel, "Perfectly secure authorization and passive identification for an error tolerant biometric system." in *IMA Int. Conf.*, ser. Lecture Notes in Computer Science, M. Walker, Ed., vol. 1746. Springer, 1999, pp. 104–113.

[11] Y. Dodis, L. Reyzin, and A. Smith, "Fuzzy extractors: How to generate strong keys from biometrics and other noisy data." in *EUROCRYPT*, ser. Lecture Notes in Computer Science, C. Cachin and J. Camenisch, Eds., vol. 3027. Springer, 2004, pp. 523–540.

[12] P. Tuyls and J. Goseling, "Capacity and examples of template-protecting biometric authentication systems." in *ECCV Workshop BioAW*, ser. Lecture Notes in Computer Science, D. Maltoni and A. K. Jain, Eds., vol. 3087. Springer, 2004, pp. 158–170.

[13] B. Schoenmakers and P. Tuyls, *Security with Noisy Data*. Springer Verlag, 2007, ch. Computationally Secure Authentication with Noisy Data, pp. 141–152.

[14] T. A. M. Kevenaar, G. J. Schrijen, M. van der Veen, A. H. M. Akkermans, and F. Zuo, "Face recognition with renewable and privacy preserving binary templates," in *AutoID'2005, 17-18 October 2005, Buffalo, New York*. Washington, DC, USA: IEEE Computer Society, 2005, pp. 21–26.

[15] P. Tuyls, A. H. M. Akkermans, T. A. M. Kevenaar, G. J. Schrijen, A. M. Bazen, and R. N. J. Veldhuis, "Practical biometric authentication with template protection." in *Audio-and Video-Based Biometrie Person Authentication*, T. Kanade, A. K. Jain, and N. K. Ratha, Eds., vol. 3546. Springer, 2005, pp. 436–446.

[16] F. Hao, R. Anderson, and J. Daugman, "Combining crypto with biometrics effectively," *IEEE Transactions on Computers*, vol. 55, no. 9, pp. 1081–1088, 2006.

[17] J. Daugman, "The importance of being random: statistical principles of iris recognition," *Pattern Recognition*, vol. 36, no. 2, pp. 279–291, 2003.

[18] R. Wildes, *Biometric Authentication : Technologies, Systems, Evaluations and Legal Issues*. Springer, 2005, ch. 3. Iris recognition.

[19] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. Wiley, 2006.

[20] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-correcting Codes*. North-Holland, 1988.

[21] C. E. Shannon, "A mathematical theory of communication," *The Bell System Technical Journal*, vol. 27, pp. 379–423, 623–656, 1948.

[22] X. Liu, K. W. Bowyer, and P. J. Flynn, "Iris Recognition and Verification Experiments with Improved Segmentation Method," in *AutoID'2005, 17-18 October 2005, Buffalo, New York*, 2005.

[23] National Institute of Standards and Technology (NIST), "Iris Challenge Evaluation," http://iris.nist.gov/ICE, 2005.

[24] CASIA, "Chinese academy of science, institute of automation," URL : http://www.sinobiometrics.com/Database.htm.

[25] J. Daugman, "High confidence visual recognition of persons by a test of statistical independence." *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 15, no. 11, pp. 1148–1161, 1993.

[26] R. Wildes, "Automated iris recognition: An emerging biometric technology," *Proceedings of the IEEE*, vol. 85, no. 9, pp. 1348–1363, 1997.

[27] K. Hollingsworth, K. W. Bowyer, and P. J. Flynn, "All iris code bits are not created equal," in *Biometrics: Theory, Applications, and Systems, 2007. BTAS 2007. First IEEE International Conference on*, Sept 2007.

[28] S. Prabakhar, A. K. Jain, D. Maio, and D. Maltoni, *Handbook of Fingerprint Recognition*. Springer-Verlag New York, Inc., 2003.

[29] S. Draper, A. Khisti, E. Martinian, A. Vetro, and J. S. Yedidia, "Using distributed source coding to secure fingerprint biometrics," Mitsubishi Electrical Research Laboratories, Tech. Rep. TR 2007-005, Jan. 2007.

[30] J.-P. M. G. Linnartz and P. Tuyls, "New shielding functions to enhance privacy and prevent misuse of biometric templates."

in *Audio- and Video-Based Biometric Person Authentication*, ser. LNCS, J. Kittler and M. S. Nixon, Eds., vol. 2688. Springer, 2003, pp. 393–402.

[31] J. Bringer, H. Chabanne, and B. Kindarji, "The best of both worlds: Applying secure sketches to cancelable biometrics," in *Science of Computer Programming*, 2007, to appear. Presented at WISSec'07.

[32] A. M. Bazen and R. N. J. Veldhuis, "Detection of cores in fingerprints with improved dimension reduction," in *4th IEEE Benelux Signal Processing Symposium(SPS-2004), Hilvarenbeek, The Netherlands*, 2004, pp. 41–44.

[33] A. M. Bazen and S. H. Gerez, "Systematic methods for the computation of the directional fields and singular points of fingerprints." *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 905–919, 2002.

[34] A. M. Bazen and R. N. J. Veldhuis, "Likelihood-ratio-based biometric verification." *IEEE Trans. Circuits Syst. Video Techn.*, vol. 14, no. 1, pp. 86–94, 2004.

[35] "Fingerprint verification competition," http://bias.csr.unibo.it/fvc2000/.

[36] P. J. Phillips, H. Moon, S. A. Rizvi, and P. J. Rauss, "The FERET evaluation methodology for face-recognition algorithms." *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 10, pp. 1090–1104, 2000.

[37] M. Weber, "Frontal face dataset," http://www.vision.caltech.edu/html-files/archive, California Institute of Technology, 1999.

[38] N. Wiberg, "Codes and decoding on general graphs," Ph.D. dissertation, Linkoping University, Linkoping, Sweden, 1996.

[39] R. M. Tanner, "A recursive approach to low-complexity codes," *IEEE Trans. on Information Theory*, vol. 27, pp. 533–547, 1981.

[40] D. Muller, "Application of boolean algebra to switching circuit design and to error detection," *IEEE Trans. on Electronic Computers*, vol. 3, pp. 6–12, 1954.

[41] I. Reed, "A class of multiple-error-correcting codes and their decoding scheme," *IEEE Trans. on Information Theory*, vol. 4, pp. 38–42, 1954.

[42] J. Daugman, "New methods in iris recognition," *Systems, Man, and Cybernetics, Part B, IEEE Transactions on*, vol. 37, no. 5, pp. 1167–1175, Oct. 2007.

Theoretical and Practical Boundaries of Binary Secure Sketches

J. Bringer, H. Chabanne, G. Cohen, B. Kindarji and G. Zémor

***Abstract*—Fuzzy commitment schemes, introduced as a link between biometrics and cryptography, are a way of handling biometric data matching as an error correction issue. We focus here on finding the best error-correcting code with respect to a given database of biometric data.**

We propose a method that models discrepancies between biometric measurements as an erasure and error channel, and we estimate its capacity. We then show that two-dimensional iterative min-sum decoding of properly chosen product codes almost reaches the capacity of this channel. This leads to practical fuzzy commitment schemes that are close to theoretical limits. We test our techniques on public iris and fingerprint databases and validate our findings.

***Index Terms*—Iris, fingerprint, biometrics, secure sketches, boundaries, min-sum decoding.**

EDICS: MOD-CHAN, WAT-THEO, BIO-PROT.

I. Introduction

With the growing use of biometric recognition systems comes the need to secure and protect the privacy associated to biometric data. Juels and Wattenberg's fuzzy commitment scheme [2] uses Error Correcting Codes and was introduced to handle differences occurring between two captures of biometric data. Many papers give applications of this technique for cryptographic purposes [3], [4], [5], [6], [7], [8], [9], [10], [11], [2], [12], [13] but only a few investigate what are the best codes for this decoding problem and how to find them.

Secure sketches have been experimented with several biometrics. Applications to face recognition [14] and to fingerprints [15] are proposed that make use of BCH codes and reliable bit extraction. In a different way, Daugman *et al.* experimented with the use of a concatenated Hadamard – Reed-Solomon code for iris recognition [16].

In this paper, we explain how to estimate the theoretical performance limit of a secure sketch, applied to binary biometric data, at a given code dimension. We then describe an efficient iterative decoding algorithm on product codes, which leads to near-optimal performance in our experiments on iris and fingerprint recognition.

J. Bringer, H. Chabanne and B. Kindarji are with Sagem Sécurité, Osny, France. This work was partially supported by the French ANR RNRT project BACH.

G. Cohen and B. Kindarji are with TELECOM ParisTech, Département Informatique et Réseaux, Paris, France.

G. Zémor is with Institut de Mathématiques de Bordeaux, Université de Bordeaux I, Bordeaux, France.

A preliminary version of this work was presented at the IEEE conference Biometrics: Transactions, Applications and Systems, 2007, cf. [1].

A. Biometric Matching and Errors Correction

1) Biometric Templates: The issue of the best codes we can expect for biometric secure sketches is addressed here, in the context of binary biometric features, as is the case for iris recognition systems, see [17] – more details on iris recognition are also available in [18]. So we focus our paper on iris biometrics but it is also relevant to fingerprints.

Indeed, our techniques are applicable to recent methods which involve transforming real-valued templates into discrete ones so as to use secure sketches (cf. [14], [15]). A nice feature of discretization is that Hamming distance becomes an efficient tool.

Note that in our setting, all templates will be binary arrays, even though our theoretical approach also applies to arrays over any finite field.

2) Matching and Error Rates: Typically, a biometric-based recognition scheme consists of two phases. First, in the enrolment phase, a biometric template b is measured from a user U and then registered in a token or a database. The second phase – the verification – captures a new biometric sample b' from U and compares it to the reference data via a matching function. According to some underlying measure μ and some recognition threshold τ, b' will be accepted as a biometric measure of U if $\mu(b, b') \leq \tau$, else rejected. Mainly two kinds of errors are associated to this scheme: False Reject (**FR**), when a matching user, i.e. a legitimate user, is rejected; False Acceptance (**FA**), when a non-matching one, e.g. an impostor, is accepted.

Note that, when the threshold increases, the **FR**'s rate (**FRR**) decreases while the **FA**'s rate (**FAR**) grows, and conversely.

3) Error Correcting Codes and Secure Sketches: Our methods will resort to information theory and coding. Some basic definitions are given hereafter. For more background, notations and classical results, the reader is referred to [19] and [20] in these two fields respectively.

Let $\mathcal{H}$ be the collection of all binary N-tuples, $\mathcal{H} = \{0,1\}^N = \mathbb{F}_2^N$, where $\mathbb{F}_2 = \{0,1\}$.

- The $\oplus$ *operator* is the canonical exclusive-or over $\mathbb{F}_2$:

$$a \oplus b = \begin{cases} 0 \text{ if } a = b \\ 1 \text{ if } a \neq b \end{cases}$$

- The *Hamming distance* over $\mathcal{H}$ is the metric distance defined as the number of binary differences between two elements, i.e.

$$d_{\mathcal{H}}(u, v) = \sum_{i=1}^{N} (u_i \oplus v_i).$$

Equipped with the Hamming distance, $\mathcal{H}$ is called the *Hamming space* of length N.

- An *Error Correcting Code* (**ECC**) over $\mathcal{H}$ is a subset $C \subset \mathcal{H}$; elements of C are called *codewords*.
- An (N, S, d) binary **ECC** is an error correcting code C over $\mathcal{H}$ with S elements such that for all distinct codewords c_1 and c_2, $d_{\mathcal{H}}(c_1, c_2) \geq d$. N is called the length of C, S is the size of C and d, the smallest Hamming distance between two distinct codewords, is the minimum distance.
- A binary *linear* error correcting code C is a vector subspace of $\mathbb{F}_2^N$. By linearity, the minimum distance d_{min} of C is now the minimum weight among non-zero codewords, where the *weight* of a vector x is its distance to the vector $\mathbf{0}$. When k is the dimension of the subspace C, C is denoted by $[N, k, d_{min}]_2$. Here, the *correction capacity* t of C is the radius of the largest Hamming ball for which, for any $x \in \mathbb{F}_2^N$, there is at most one codeword in the ball of radius t centred on x. Clearly, $t = \lfloor (d_{min} - 1)/2 \rfloor$.

Assuming that the templates live in $\mathcal{H}$, the main idea of fuzzy schemes, as introduced in [2], is to convert the matching step into an error-correcting one. Let C be an (N, S, d) **ECC** in $\mathcal{H}$.

- During the enrolment phase, one stores $z = c \oplus b$, where c is a random codeword in C,
- During the verification phase, one tries to correct the corrupted codeword $z \oplus b' = c \oplus (b \oplus b')$. Note that when the Hamming distance $d_{\mathcal{H}}(b, b')$ is small, recovering c from $c \oplus (b \oplus b')$ is, in principle, possible.

The correction capacity of C may thus be equal to τ if we do not want to alter the **FRR** and the **FAR** of the system. Unfortunately, the difference between two measures of one biometric source can be very important, whereas the correction capacity of a code is structurally constrained.

The fuzzy commitment scheme is then an error-tolerant authentication scheme which follows the above method with the use of a committed value. The main goal is to protect the storage of biometric data involved in an authentication biometric system. Let h be a cryptographic one-way function, and let us store $h(c)$ in the enrolment phase, together with $z = c \oplus b$. The authentication will be a success if the verification returns a codeword c' such that $h(c') = h(c)$. An illustration of the scheme is provided in Fig. 1.

This construction has been formalized in [11] under the name *Secure Sketch*. Informally, a secure sketch is made of a probabilistic Sketching Function SS, which "hides" the biometric template, and a deterministic Recovery Function Rec which recovers the original template if not too many errors have occurred.

Several constraints are studied in the literature, e.g. in [2], [9], [11], to achieve the protection of b while z is publicly known. These works show that the code C must be adapted to the entropy of biometrics and it leads in fact to a trade-off between correction capacity of C and the security properties of the scheme. Moreover, the size S of C should not be too small, to prevent z from revealing too much information about the template b: indeed the probability for an attacker to "guess" b out of $z = c \oplus b$, with the computation of $z \oplus \tilde{c}$ from the choice of a random codeword $\tilde{c}$, is lower bounded by $1/S$. This issue is also discussed in Sec. IV-D.

B. Organization of this Work

We first look for theoretical limits. In Sec. II, we formalize our problem by transforming a database of biometric data into a binary erasure-and-error channel. We then give a method for finding an upper bound on the underlying error correction capacity, and explain how to transpose this result into bounds on **FAR** and **FRR**.

Section III introduces the biometric datasets – two for iris biometric and one for fingerprints – we use in our experiments; we then present the practical bounds deduced from our result.

In Sec. IV, we illustrate our method by describing a very efficient construction with iterative min-sum decoding of product codes, and we provide parameters that put our performances close to the theoretical limit for those databases.

Section V concludes.

II. Theoretical Optimal Correction

A. Model

We consider two separate channels with a noise model based on the differences between any two biometric templates.

- The first channel, called the **matching channel**, is generated by errors $b \oplus b'$ where b and b' come from the same user U.
- The second channel, the **non-matching channel**, is generated by errors where b and b' come from different biometric sources.

In a practical biometric system, the number of errors in the **matching channel** is on average lower than in the **non-matching channel**.

Moreover, the templates are not restricted to a constant length. Indeed, when a sensor captures biometric data, we want to keep the maximum quantity of information but it is rarely possible to capture the same amount of data twice – for instance an iris may be occulted by eyelids – hence

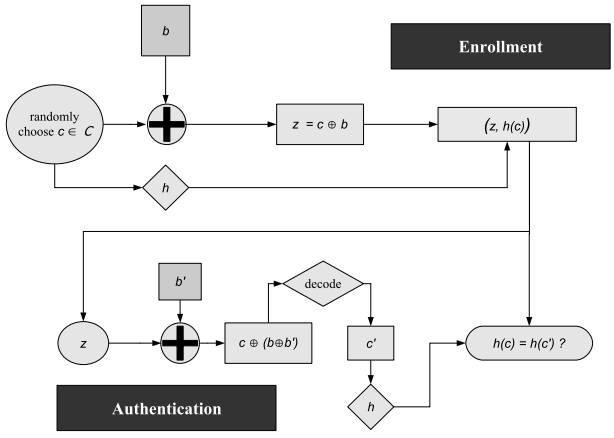


Figure 1. The Fuzzy Commitment Scheme [2]

the templates are of variable length. This variability can be smoothed by forming a list of erasures, i.e. the list of coordinates where they occur. More precisely, in coding theory, an erasure in the received message is an unknown symbol at a known location. We thus have an erasure-and-error decoding problem on the **matching channel**. Simultaneously, to keep the **FAR** low, we want a decoding success to be unlikely on the **non-matching channel**: to this end we impose bounds on the correction capacity.

In the sequel, we deal with binary templates with at most N bits and assume, for the theoretical analysis that follows, that the probabilities of error and erasure on each bit are independent, i.e. we work on a binary input memoryless channel. Note that resorting to interleaving makes this hypothesis valid for all practical purposes.

B. Taking into Account Errors and Erasures

As we take into account erasures into our biometric model, we also need to slightly enhance Juels and Wattenberg's scheme. Let (b, m) and (b', m') be two biometric templates, b, b' denoting the known information, and m, m' the list of erasures, in the way IrisCodes are represented. We can represent some $(b, m) \in \{0,1\}^N \times \{0,1\}^N$ by a ternary vector $\tilde{b} \in \{0,1,\epsilon\}^N$, where the third symbol ϵ represents an erasure.

The updated **xor** rule on $\{0,1,\epsilon\}$ is very similar to the usual one: we define $x \tilde{\oplus} x'$ to be $x \oplus x'$ if x and x' are bits, and ϵ if one of x, x' is ϵ.

In order to protect c and b, the updated sketch will simply be the sum $z = c \tilde{\oplus} \tilde{b}$. The verification step will also use the $\tilde{\oplus}$ operation to combine z with $\tilde{b'}$ into $z \tilde{\oplus} \tilde{b'}$. The decoding can then proceed to correct incorrect bits and erasures.

C. Theoretical Limit

Our goal is to estimate the capacity, in the Shannon sense [21], of the matching channel when we work with a code of a given dimension. Namely, we want to know the maximum number of errors and erasures between two biometric measures that we can manage with secure sketches for this code.

Starting with a representative range of matching biometric data, the theorem below gives an easy way to estimate the lowest achievable **FRR**. The idea is to check whether the best possible code with the best generic decoding algorithm, i.e. a **maximum-likelihood** (**ML**) decoding algorithm which systematically outputs the most likely codeword, would succeed in correcting the errors.

Theorem 1: Let $k \in \mathbb{N}^*$, C be a binary code of length N and size 2^k, and m a random received message, from a random codeword of C, of length N with w_n errors and w_e erasures. Assume that C is an optimal code with respect to N and k, equipped with an **ML** decoder.

If $\frac{w_n}{N-w_e} > \theta$ then m is only decodable with a negligible probability for a large N, where θ is such that the Hamming sphere of radius $(N-w_e)\theta$ in $\mathbb{F}_2^{N-w_e}$, i.e. the set $\{x \in \mathbb{F}_2^{N-w_e}, d_{\mathcal{H}}(x, \mathbf{0}) = (N-w_e)\theta\}$, contains 2^{N-w_e-k} elements. □

Proof: In the case of errors only (i.e. no erasures) with error-rate $p := w_n/N$, the canonical second theorem of Shannon asserts that there are families of codes with (transmission) rate $R := k/N$ coming arbitrarily close to the *channel capacity* $\kappa(p)$, decodable with ML-decoding and a vanishing (in N) word error probability P_e.

In this case, $\kappa(p) = 1 - h(p)$, where $h(p)$ is the (binary) entropy function (log's are to the base 2):

$$h(x) = -x \log x - (1-x) \log(1-x).$$

Furthermore, P_e displays a threshold phenomenon: for any rate arbitrarily close to, but above capacity and any family of codes, P_e tends to 1 when N grows.

Equivalently, given R, there exists an error-rate threshold of

$$p = h^{-1}(1-R),$$

h^{-1} being the inverse of the entropy function.

Back to the errors-and-erasures setting now. Our problem is to decode to the codeword nearest to the received word on the *non-erased* positions.

Thus we are now faced with a punctured code with length $N - w_e$, size 2^k, transmission rate $R' := k/(N-w_e)$ and required to sustain an error-rate $p' := \frac{w_n}{N-w_e}$.

By the previous discussion, if

$$p' > \theta := h^{-1}(1-R'),$$

NO code and NO decoding procedure exist with a non-vanishing probability of success.

To conclude the proof, use the classical Stirling approximation for the size of a Hamming sphere of radius αM in $\mathbb{F}_2^M$ by $2^{h(\alpha)M}$. ■

This result allows us to estimate the correcting capacity of a biometric matching channel with noise and erasures under the binary input memoryless channel hypothesis.

Indeed applying Theorem 1 to the **matching channel** gives a lower-bound on the **FRR** achievable (i.e. the *best* **FRR**), whereas applying it to the **non-matching channel** gives an upper-bound of the **FAR** (say the *worst* **FAR**).

Corollary 1: For a given biometric authentication system based on a binary secure sketch of length N and dimension k, and a given biometric database $\mathcal{B} = \{b_i\}$, let the function $f_{N,k}$ be $f_{N,k}(\tilde{y}) = \frac{w_n}{N-w_e} - h^{-1}\left(1 - \frac{k}{N-w_e}\right)$, with w_n the number of 1's occuring in $\tilde{y}$ and w_e the number of ϵ. Define $p^G_{N,k}(x)$ (resp. $p^I_{N,k}(x)$) as the probability density of results of all genuine (resp. impostor) comparisons $f_{N,k}(\tilde{b} \tilde{\oplus} \tilde{b'})$ for $b, b' \in \mathcal{B}$.

Under these hypothesis, the following inequalities stand: $FRR \geq \int_0^{+\infty} p^G_{N,k}(t)dt$ and $FAR \leq \int_{-\infty}^0 p^I_{N,k}(t)dt$. □

In other words, Corollary 1 can lead to a kind of theoretical ROC curve which is not represented thanks to the classical matching score distributions but with the dimension of the underlying optimal code on the abscissa axis. Therefore, from a given database and a given features extraction scheme – dedicated to discrete representation, it is possible to induce an

approximation of the error-rates one can expect from templates of the same quality. In particular, it may help to evaluate the efficiency of the extraction algorithm.

Practical implications of this theorem are illustrated in Sec. III-B.

III. Application to Biometric Data

To explain our approach, we now present the estimation of these optimal performances on several public biometric databases.

A. Our Setting: Data Sets and Templates

We describe here the sets on which we made experiments. We first describe the iris pictures from ICE 2005 and CASIA v1, then the FVC 2000 dataset from which we extracted binary fingerprint templates.

For each dataset, we also represent the boundaries on **FRR** and **FAR**.

1) IrisCodes *and associated databases:* We first made different experiments on iris recognition, which is a very natural target for binary error correcting codes. We chose two public databases:

- The ICE 2005 database: [22], [23].
  It contains 2953 images coming from 244 different eyes. It is taken without modification but one slight correction: the side of the eye 246260 has been switched from left to right. Hence we keep 2953 images. In this dataset the number of images for each eye is variable.
- The CASIA database: [24].
  This is the first version of the Chinese Academy of Science public iris database. It contains 756 pictures of 108 different eyes, with 7 pictures per eye.

A 256-byte (2048 bits) iris template, together with a 256-byte mask, is computed from each iris image using the algorithm reported in [17]; the mask filters out the unreliable bits, i.e. stores the erasures positions of the iris template. The resulting template is called *IrisCode*.

Note that the iris template as computed by this algorithm has a specific structure: [17] reports 249 degrees-of-freedom within the 2048 bits composing the template. As described in [25], [17], [26], the algorithm involves computation of several Gabor filters on separate and local areas of the iris picture. The picture is normalized onto its polar representation, then divided into areas of regular size. The amplitude information is discarded and the actual bits are the phase quantization of this Gabor-domain representation of the iris image. The ordering of the bits is directly linked to the localization of the area. In practice, the iris code can be represented by an 2D bits-array.

The classical way to compare two iris codes I_1, I_2 with masks M_1, M_2 is to compute the relative Hamming distance

$$\frac{||(I_1 \oplus I_2) \cap M_1 \cap M_2||}{||M_1 \cap M_2||} \tag{1}$$

for some rotations of the second template – to deal with the iris orientation's variation – and to keep the lowest score.

This formula gives the Hamming distance distribution given on Fig. 2(a), where the scores of matching (intra-eyes) and non-matching (inter-eyes) comparisons are represented. We can see that there is an overlap between the two curves, and that the number of errors to handle in the matching channel is large. On iris matching-channel an additional difficulty originates from the number of erasures which varies, for instance for ICE, from 512 to 1977.

Although we know that all bits are not independent and that they do not follow the same distribution (see e.g. [27]), following (1) the typical matching score computation does not use any internal correlations between bits of the iris codes. So in this setting it is coherent to suppose the matching channel to be a binary input memoryless channel with independent bit errors and erasures. It will thus be possible to apply Theorem 1 in this context.

2) Fingerprint Encoding and Associated Database: Traditional fingerprint matching is made thanks to minutiae extraction [28] and comparisons of unordered sets $\mathcal{E}$, $\mathcal{E}'$ of variable length. Using the characteristic function $\chi_{\mathcal{E}}$ – as done in [11], [29] – is a way to translate minutiae into a binary vector of fixed length. The size corresponds to the number of values the coordinates could take. From a set of minutiae, the idea is to construct a vector with all coordinates equal zero except those which are associated with the position of one minutiae. The problem is that this representation is not well-suited for binary secure sketches. Indeed, the metrics associated to the set representation is the symmetric set difference, which does not take into account local distortion due to elasticity of the finger skin. Still, Secure Sketches are easier to construct for the Hamming distance with a q-ary code.

To overcome this difficulty, Tuyls *et al.* [15] describe a smart algorithm, in the line of the previous works [30], [12], to extract stable binary vectors from fingerprints and to apply secure sketches on them. We based our experiments on such a coding, more precisely on an improvement which has been proposed in [31].

We describe a synthesis of the algorithm below. The main idea is to deal with fingerprint patterns rather than minutiae. It makes use of core-based alignment techniques and pattern features linked to directional fields, thanks to the techniques described in [32], [33], [34]. Moreover, to increase the stability of the vectors, the binary fixed-length strings are generated following some statistics by using several images per user at the enrolment.

For these experiments, the FVC2000 [35] public database (Db. 2) was tested. This data set is made of 800 pictures of 100 different fingers, 8 pictures per finger. The image size is 256 by 364 pixels.

Before the enrolment step, we first align the picture on a fixed point, such as the core, if available. For that, we preprocess the picture in order to extract the core point and an evaluation of the vertical axis. We translate the picture, then adjust it to take care of some possible rotation. In practice, we did prealign the fingerprints to solely test the binarization proposed.

These points are then executed:

1) Picture embedding: to take into account the alignment,

we embed the 256×364 pixels picture into a larger picture of 768×1092 to prevent any loss of information after re-alignment.

2) Real-vector extraction: we compute several Gabor filters on the resulting picture, of which we keep only the magnitude. We also compute the directional field of the fingerprint. The concatenation of both computations gives us a real-vector of length $L = 17952$, of which 15968 positions are known to be null, due to the embedding. These positions will be marked as erasures.
3) Binarization: the enrolment is done on several pictures per user and several users; a statistical analysis gives enough information to quantize the vectors by comparing – coordinate after coordinate – the mean value of a user to the mean value of the overall enrolment database. For each user, this gives a vector from $\{0, 1, \epsilon\}^L$.
4) Reliable components selection: for each user, all enrolment vectors are combined into a bit string of fixed length N. This is done by selecting only the N most stable coordinates from the different vectors. As it is likely that real-life pictures never are pre-aligned, it is likely that the null positions will not be the same for each fingerprint capture; this enables to choose $N \geq 17952 - 15968 = 1984$. More details on component selection are given in [31].

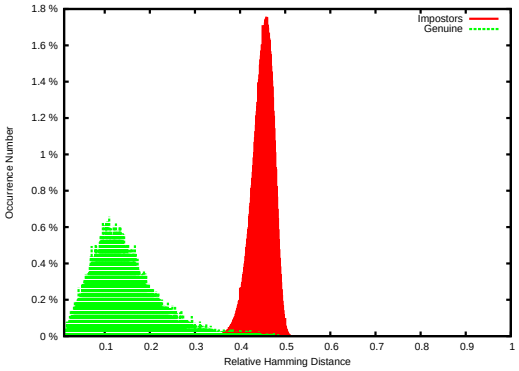
Hence we obtain binary templates, together with a mask, of a fixed length. The verification step is quite similar; to get the fresh biometric template, we use the positions selected at the enrolment step, and then compare them with the enrolled vector.

In the sequel, we selected 6 images per finger for the enrolment phase, one 2048-bit template per enrolled finger is obtained, possibly with some erasures, and the remaining 200 images are kept for verification. As the verification step is done on just one picture, the verification template will always contain at least $2048 - 1984 = 64$ erasures; this is well captured by the decoding algorithm. To increase the overall number of comparisons, we iterate the tests for every choices of 6 images. This gives us a genuine match count of 5600, and an impostor match count of 19800.

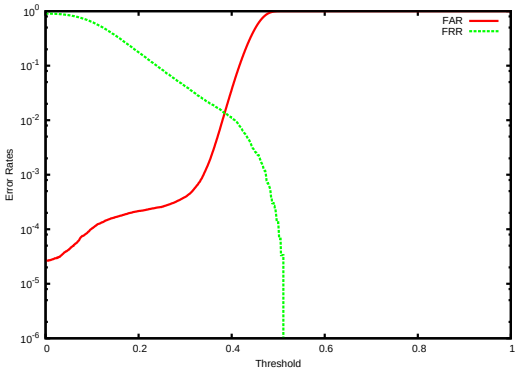
Any other biometrics may be used to apply Theorem 1 as soon as we succeed in getting a discrete representation of the templates associated to a Hamming distance classifier.

B. *Performances Estimation on these Databases*

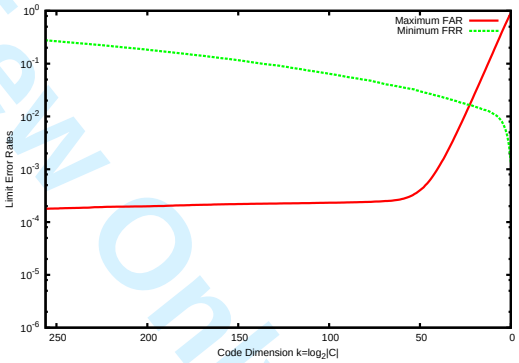
For each one of these databases we represent, in Figures 2, 3 and 4, the relative Hamming distance distribution thanks to Eq. (1) for the matching and the non-matching channel and the corresponding FRR and FAR curves. We have also estimated the optimal performances given by Corollary 1 and the results are drawn in Figures 2(c), 3(c) and 4(c). The curves correspond to the best FRR achievable with respect to the code's dimension and the greatest possible FAR as a function of this dimension.



(a) Hamming distance distributions



(b) FAR and FRR via the Hamming distance (Eq. 1) using a threshold



(c) Worst FAR and best FRR w.r.t. the code dimension

Figure 2. The ICE 2005 Dataset, IrisCodes

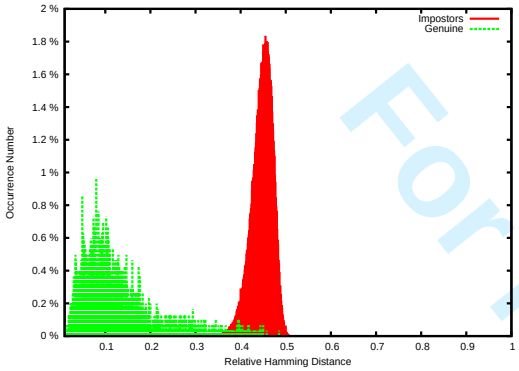
From the Hamming Distance distributions, it is obvious that, while iris recognition performs well with the IrisCode algorithm, the chosen quantization is not as well adapted to fingerprint matching. Therefore, the different results we shall have will significantly differ.

For the three datasets, we see that the ratio of errors to handle to approach the Equal Error Rate – **EER** – is very high, which is a problem for classical correcting codes as it is explained in the next section.

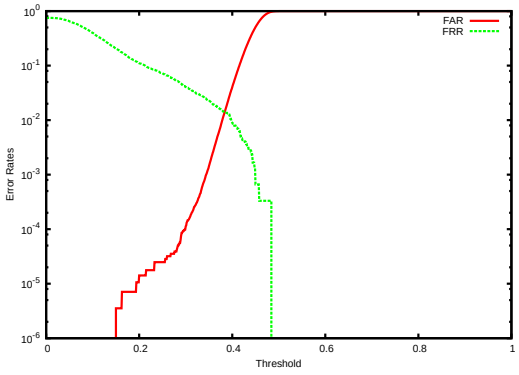
We summed up some of the numerical limits on **FAR** and **FRR** in table I, for dimensions likely to be chosen for practical purposes. A general consequence is that the dimension of the code can not be chosen too high in order to keep good **FR** rates.

Code's dimension	Minimum **FRR**			Maximum **FAR**		
	ICE	CASIA	FVC	ICE	CASIA	FVC
42	$2.49 \cdot 10^{-2}$	$3.15 \cdot 10^{-2}$	$0.59 \cdot 10^{-2}$	$8.14 \cdot 10^{-4}$	$1.13 \cdot 10^{-4}$	$17.88 \cdot 10^{-2}$
64	$3.76 \cdot 10^{-2}$	$4.47 \cdot 10^{-2}$	$1.26 \cdot 10^{-2}$	$2.74 \cdot 10^{-4}$	0	$10.32 \cdot 10^{-2}$
80	$4.87 \cdot 10^{-2}$	$5.77 \cdot 10^{-2}$	$1.93 \cdot 10^{-2}$	$2.57 \cdot 10^{-4}$	0	$7.07 \cdot 10^{-2}$
128	$9.10 \cdot 10^{-2}$	$9.18 \cdot 10^{-2}$	$5.87 \cdot 10^{-2}$	$2.41 \cdot 10^{-4}$	0	$2.67 \cdot 10^{-2}$

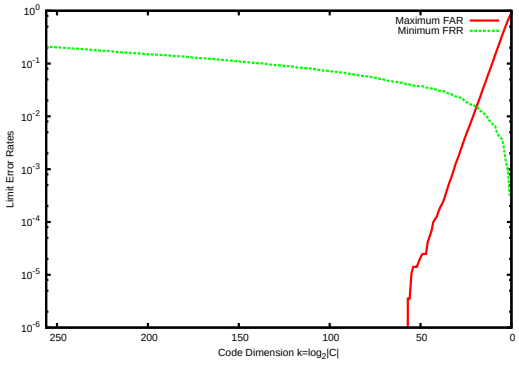
Table I
THEORETICAL LIMITS ON STUDIED DATABASES



(a) Hamming distance distributions

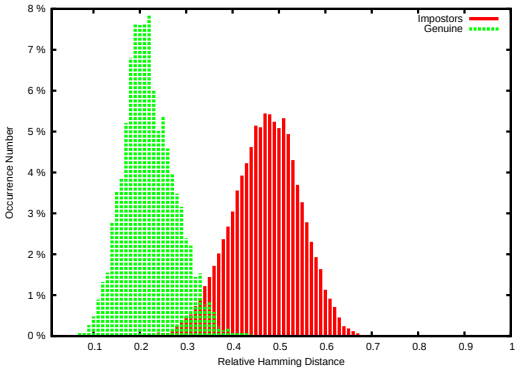


(b) FAR and FRR via the Hamming distance (Eq. 1) using a threshold

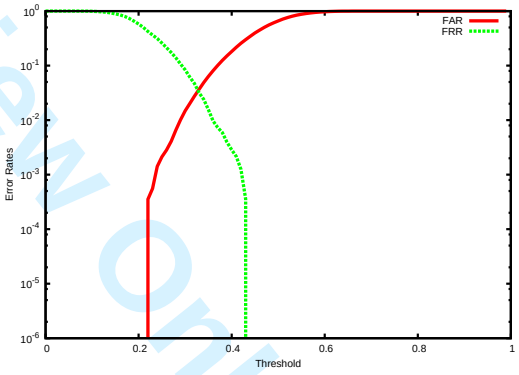


(c) Worst FAR and best FRR w.r.t. the code dimension

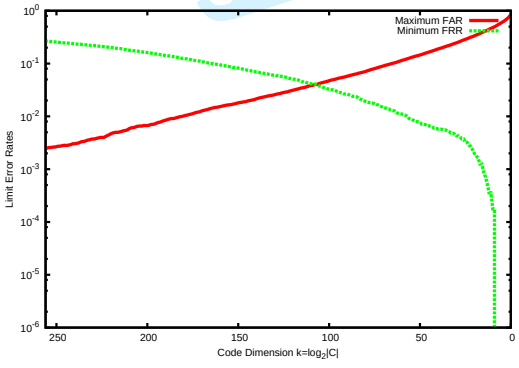
Figure 3. The CASIA v1 Dataset, IrisCodes



(a) Hamming distance distributions



(b) FAR and FRR via the Hamming distance (Eq. 1) using a threshold



(c) Worst FAR and best FRR w.r.t. the code dimension

Figure 4. The FVC 2000 (Db. 2) Dataset, Binary Encoding

Note that Theorem 1 gives us estimations of the theoretical limits based on asymptotic analysis under a memoryless channel hypothesis, i.e. independent bits. In principle, it could be possible to expect more efficiency without resorting to bit interleaving which in practice makes the channel memoryless. However this would require highly intricate modelling of the matching channel, and it seems unreasonable to expect that the decoding problem would be within reach of present day algorithms.

IV. A Near Optimal Construction

A. Previous Works

1) Quantization and BCH codes: In known applications of secure sketches to quantized biometrics, for instance [14], [15], the error correcting codes are seen directly to act as a Hamming distance classifier at a given threshold. Hence, the correction capacity naturally corresponds to the threshold we want to attain. To this end, the use of BCH codes is proposed: the advantage is their existence for a wide class of parameters, the main drawback is that the correction capacity is a hard constraint for the dimension.

As an illustration, in [14] the quantization technique is applied to face recognition on two databases, FERET database [36] and one from Caltech [37]. A Hamming distance classifier gives Equal Error Rates of 2.5% and 0.25% respectively for a threshold greater than 0.32 with code length 511. Unfortunately to achieve this minimal distance, the BCH code has dimension 1. A BCH of dimension 40 enables a threshold of 0.185 with a **FRR** greater than 10% and 1% respectively.

This phenomenon holds in [15] and for our first experiments on the FVC2000 dataset. Following Fig. 4(b), we remark that to achieve a **FRR** better than the EER, the threshold is high: for example, for a rate around 2%, the threshold is near 0.4 which is not realistic with non-trivial BCH codes. To overcome this limitation, we propose in the sequel to use more appropriate codes.

2) IrisCodes *and Concatenated Codes:* More efficient codes are proposed in [16]. The secure sketch scheme is applied with a concatenated error-correcting code combining a Hadamard code and a Reed-Solomon code. More precisely, the authors use a $[32, k_{RS}, 33 - k_{RS}]_{2^7}$ Reed-Solomon code and a $[64, 7, 32]_2$ Hadamard code: a codeword of 2048 bits is in fact constructed as a set of 32 blocks of 64 bits where each block is a codeword of the underlying Hadamard code. As explained in [16], the Hadamard code is introduced to deal with the background errors and the Reed-Solomon code to deal with the bursts (e.g. caused by eyelashes, reflections, ...).

Note that in this scheme, the model is not exactly the same as ours, as the masks are not taken into account. Moreover, the quality of the database used in [16] is better than the public ones we worked with. The mean intra-eye Hamming distance reported in the paper is 3.37% whereas this number becomes 13.9% in the ICE database, which means that we must have a bigger correcting capacity. The inter and intra-eyes distributions reported by the authors is drawn on Fig. 5.

Even if [16] reports very good results on their experiments with a 700-image database , the codes do not seem appropriate

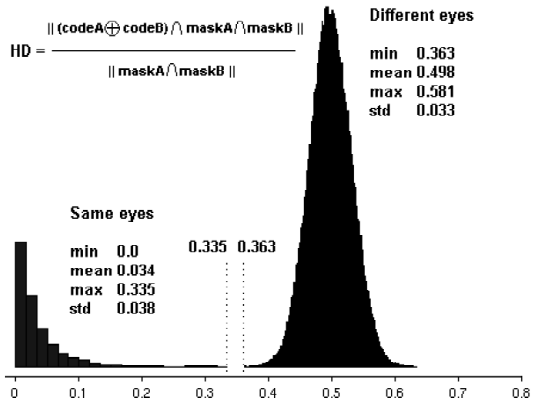


Figure 5. Hamming distance distributions from [16]

in our case as the same parameters on the ICE database gave us a too large rate of **FR** (e.g. 10% of **FR** with 0.80% of **FA**), even for the smallest possible dimension of the Reed-Solomon code when $t_{RS} = 15$.

To sum up, with respect to the Hamming distance distribution in figures 2, 3, 4, we need to find correcting codes with higher correction capacity. To achieve performances closer to the theoretical estimation given in section III-B is also a great motivation.

B. Description of the Two-Dimensional Iterative Min-Sum Decoding Algorithm

We now describe a very efficient algorithm which will help us to overcome the difficulties mentioned above.

For a linear code with a minimum distance d_{min}, we know that an altered codeword with w_n errors and w_e erasures can always be corrected, disregarding decoding complexity issues, provided that $2w_n + w_e < d_{min}$.

Classical algebraic decoding of BCH codes and concatenated Reed-Solomon codes achieve this bound, but not more. This upper bound is however a conservative estimate: it has been known since Shannon's days that it is possible in principle to correct many more errors and erasures, all the way to the channel capacity. In practice, *iterative decoding* algorithms are now known to be capable of achieving close-to-capacity performance, for such code families as LDPC or turbo codes. It is therefore natural to try and bring in iterative decoding to improve on secure sketches that use algebraic decoders. LDPC codes and turbo codes are however not usually designed for such noisy channels as the type we have to deal with: in particular, classical turbo codes are known not to behave well under high noise. We have therefore chosen to use product codes: this is because under the high noise condition particular to biometrics, we will be dealing with codes of small dimension so that we can apply maximum-likelihood decoding (exhaustive search) to the constituent codes and alternate between both decoders with an iterative process. This will yield a particularly efficient blend of iterative decoding and exhaustive search.

We now describe product codes together with the specific iterative decoding algorithm we will use. A product code $C = C_1 \otimes C_2$ is constructed from two codes: $C_1[N_1, k_1, d_1]_2$ and $C_2[N_2, k_2, d_2]_2$. The codewords of C can be viewed as matrices of size $N_2 \times N_1$ whose rows are codewords of C_1 and columns are codewords of C_2, see Fig. 6.

This yields a $[N_1 \times N_2, k_1 \times k_2, d_1 \times d_2]_2$ code. When k_1 and k_2 are small enough for C_1 and C_2 to be decoded exhaustively a very efficient iterative decoding algorithm is available, namely the *min-sum* decoding algorithm. Min-sum decoding of LDPC codes was developed by Wiberg [38] as a particular instance of message passing algorithms. In a somewhat different setting it was also proposed by Tanner [39] for decoding generalized LDPC (Tanner) codes. The variant we will be using is close to Tanner's algorithm and is adapted to product codes. Min-sum is usually considered to perform slightly worse than the more classical sum-product message passing algorithm on the Gaussian, or binary-symmetric channels, but it is specially adapted to our case where knowledge of the channel is poor, and the emphasis is simply to use the Hamming distance as the appropriate basic cost function.

Let (x_{ij}) be a vector of $\{0,1\}^{N_1 \times N_2}$. The min-sum algorithm associates to every coordinate x_{ij} a cost function κ_{ij} for every iteration of the algorithm. The cost functions are defined on the set $\{0,1\}$. The initial cost function κ^0_{ij} is defined by $\kappa^0_{ij}(x) = 0$ if the received symbol on coordinate (ij) is x and $\kappa^0_{ij}(x) = 1$ if the received symbol is $1 - x$.

A *row* iteration of the algorithm takes an *input* cost function κ^{in}_{ij} and produces an *output* cost function κ^{out}_{ij}. The algorithm first computes, for every row i and for every codeword $c = (c_1 \dots c_{N_1})$ of C_1, the *sum*

$$\kappa_i(c) = \sum_{j=1}^{N_1} \kappa^{in}_{ij}(c_j)$$

which should be understood as the cost of putting codeword c on row i. The algorithm then computes, for every i, j, κ^{out}_{ij} defined as the following *min*, over the set of codewords of C_1,

$$\kappa^{out}_{ij}(x) = \min_{c \in C_1, c_j = x} \kappa_i(c).$$

This last quantity should be thought of as the minimum cost of putting the symbol x on coordinate (ij) while satisfying the row constraint.

$$c = \begin{pmatrix} c_{1,1} & \dots & c_{1,j} & \dots & c_{1,n_1} \\ & & \vdots & & \\ c_{i,1} & \dots & c_{i,j} & \dots & c_{i,n_1} \\ & & \vdots & & \\ c_{n_2,1} & \dots & c_{n_2,j} & \dots & c_{n_2,n_1} \end{pmatrix}$$
$$\forall i \in [0, n_2], (c_{i,1}, c_{i,2}, \dots, c_{i,n_1}) \in C_1$$
$$\forall j \in [0, n_1], (c_{1,j}, c_{2,j}, \dots, c_{n_2,j}) \in C_2$$

Figure 6. A codeword of the product code $C_1 \otimes C_2$ is a matrix where each line is a codeword of C_1 and each column a codeword of C_2

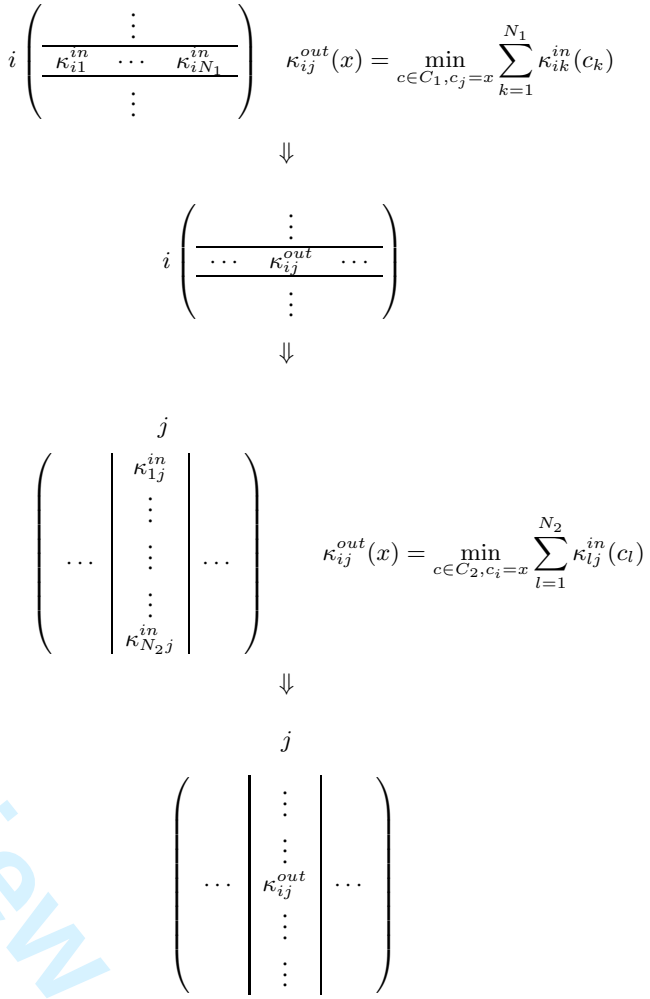


Figure 7. A row iteration followed by a column one

A *column* iteration of the algorithm is analogous to a row iteration, with simply the roles of the row and column indexes reversed, and code C_2 replacing code C_1. Precisely we have

$$\kappa_j(c) = \sum_{i=1}^{N_2} \kappa^{in}_{ij}(c_i) \tag{2}$$

and

$$\kappa^{out}_{ij}(x) = \min_{c \in C_2, c_i = x} \kappa_j(c).$$

The algorithm alternates row and column iterations as illustrated by Fig. 7. After a given number of iterations (or before, if we find a codeword) it stops, and the value of every symbol x_{ij} is put at $x_{ij} = x$ if $\kappa^{out}_{ij}(x) < \kappa^{out}_{ij}(1-x)$. If $\kappa^{out}_{ij}(x) = \kappa^{out}_{ij}(1-x)$ then the value of x_{ij} stays undecided (or erased).

The following theorem is fairly straightforward and illustrates the power of min-sum decoding.

Theorem 2: If the number of errors is less than $d_1 d_2/2$, then two iterations of min-sum decoding of the product code $C_1 \otimes C_2$ recover the correct codeword. □

Proof: (Sketch)

Without loss of generality, the correct codeword is the all-zero vector. Suppose that after the second iteration the algorithm prefers 1 to 0 in some position (i, j). This means that the cost (2) $\kappa_j(c)$ of some non-zero codeword c of C_2 is smaller than the cost $\kappa_j(0)$ of the zero column vector. Now the cost $\kappa_j(c)$ of putting codeword c in column j is equal to the Hamming distance between the received vector (x_{ij}) and a vector $\mathbf{x}_c$ that has c in column j and only rows belonging to C_1. The cost $\kappa_j(0)$ of putting the zero vector in column j is equal to the Hamming distance between the received vector (x_{ij}) and a vector $\mathbf{x}_0$ that has only zeros in column j and only rows belonging to C_1. Now since c belongs to C_2 it has weight at least d_2, therefore the Hamming distance between $\mathbf{x}_c$ and $\mathbf{x}_0$ is at least $d_2 d_1$, and $\mathbf{x}_c$ has at least d_2 rows of weight at least d_1 and at distance at least d_1 from the corresponding rows of $\mathbf{x}_0$. Therefore, if the received vector (x_{ij}) is closer to $\mathbf{x}_c$ than to $\mathbf{x}_0$, it must have weight at least $d_1 d_2/2$. ■

C. Experiments and Results

To validate the algorithm described in section IV-B, we now present the results of experiments on the public biometric databases introduced in Sec. III-A, where we succeed in obtaining some correction performances close to the theoretical limit.

We have experimented the algorithm on these databases with a particular choice for the code. In fact, the product code is constructed to fit with an array of 2048 bits, by using Reed-Muller codes [40], [41] of order 1 which are known to have good weight distributions. A binary Reed-Muller code of order 1 in m variables, abbreviated as $RM(1, m)$, is an $[2^m, m+1, 2^{m-1}]_2$ code. We chose to combine the $RM(1, 6)$ with the $RM(1, 5)$, leading to a product code of dimension 42 and codewords of length 64×32.

The overall size of the code could appear small from a cryptographic point of view, but following the theoretical analysis of section II-C, it is difficult to expect much more while achieving a low **FRR** on a practical biometric database. Achievable error rates are drawn in Sec. III-B for each database we studied.

The density of errors and erasures in an IrisCode can be very high in some regions, such as areas where eyelashes occlude the iris. The same goes for the fingerprint for which the captured area differs significantly between two measures, leading to high-erasures regions. Therefore, we also added a randomly chosen interleaver to break the biometric structure and increase the efficiency of the decoding algorithm.

- In so doing, we succeeded in obtaining for ICE a **FRR** of about 5.62% for a very small **FAR** (strictly lower than 10^{-5}). This is very close to the error rates obtained in a classical matching configuration. Note that in contrast Eq. (1) and Fig. 2(a) only give a **FAR** of about 10^{-4} for a similar **FRR**. .
- In the CASIA case, the algorithm gave us a **FRR** of 6.65% and 0 **FA**. A basic Hamming distance classifier would not give zero **FA** for a **FRR** less than 20%.
- For fingerprint from FVC2000 dataset, it yielded a **FRR** of 2.73% and a **FA** rate of 5.53%, which is also a very good result for a binary encoding scheme.

In all cases, the correction rates are relatively close to the theoretical results from Table I, and so the algorithm succeeds in achieving near-optimal results. We also noted that, unexpectedly, decoding performances are more accurate than using a basic Hamming score such as Eq. (1) with a fixed threshold for differentiating between matching and non-matching pairs. This underlines the fact that even though Hamming scores give decent results for binary matching, the associated classifier is suboptimal and can be overtaken by more elaborate techniques such as our decoding algorithm, or alternative matching functions that have been put forward recently, *e.g.* [42].

D. Cautions and Limitations

Remember that Theorem 1 is deduced from an asymptotic behaviour, thus to obtain better results, we probably need to increase the length of the templates. Moreover, the base assumption for the computation of the threshold θ is that errors and erasures occur independently and with the same probability. This assumption is far from true in practice, thus the theoretical limit on the error rates obtained by Corollary 1 should give slightly smaller False Reject Rates.

Moreover, even though we achieved near-limit results, we must not neglect some warnings for the use of Secure Sketches as a way to secure biometrics templates.

First of all, as it was noted in [3], a biometric database that would be secured thanks to Secure Sketches would not protect its users' privacy against forward verification. In a few words, if someone gains access to a biometric template b_0, it is easy for him to check whether it corresponds to a previously enrolled individual or not. As the biometrics we focused on – iris and fingerprints – are hardly private and secret, this is a flaw to seriously consider.

The error rates on secure sketches are more than just an artefact from the classical biometric behaviour: they lead to a security gap if secure sketches are used as they were presented in [2]. Indeed, to decode a sketch $(z, h(c))$ stored in a database, an attacker can try to decode every $b' \oplus z$ for b' a template from a collection of biometric measures. This collection can be an independent database the attacker collected for his personal use, or any public or secret biometric database. Whenever he obtains some codeword c', he can compare $h(c')$ with $h(c)$. If the comparison is successful, the attacker deduces $c = c'$, and thus $b = z \oplus c$. This event is likely to happen with probability **FAR**, which we can upperbound by the estimation given by Corollary 1.

Recall that a cryptographic application is nowadays considered as secure enough if the best attack known to break it takes about 2^{80} operations to be successful. If no more consolidation is done on the Fuzzy Commitment Scheme, there exists a vulnerability that gives access to b and c with about $1/\mathbf{FAR} = 2^{-\log_2(\mathbf{FAR})}$ operations, *i.e.* way less than what would be acceptable. We thus strongly discourage the use of Secure Sketches without further protection, such as [5], [4], [31].

V. Conclusion

This article demonstrates the inherent limits of error-correction based matching. We derived explicit upper bounds on the correction capacity of secure sketches, and we validated our theoretical results on two public iris databases and one fingerprint database. We then showed how the two-dimensional iterative min-sum decoding algorithm achieves correction performance close to the optimal decoding rate.

We believe that our techniques are also of great interest to other biometrics when the number of errors to manage and correct is quite large.

This paper shows a numerical constraint on the usual performance-security trade-off of secure sketches. Future work in this domain includes finding nearer-limit codes and decoding algorithms as well as improving the reliability of biometrics templates.

References

[1] J. Bringer, H. Chabanne, G. Cohen, B. Kindarji, and G. Zémor, "Optimal iris fuzzy sketches," in *Biometrics: Theory, Applications, and Systems, 2007. BTAS 2007. First IEEE International Conference on*, Sept 2007.

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

[3] J. Bringer, H. Chabanne, and Q. D. Do, "A fuzzy sketch with trapdoor." *IEEE Transactions on Information Theory*, vol. 52, no. 5, pp. 2266–2269, 2006.

[4] J. Bringer and H. Chabanne, "An authentication protocol with encrypted biometric data," *AFRICACRYPT*, vol. LCNS 5023, pp. 109–124, 2008.

[5] J. Bringer, H. Chabanne, M. Izabachène, D. Pointcheval, Q. Tang, and S. Zimmer, "An application of the Goldwasser-Micali cryptosystem to biometric authentication." in *ACISP*, ser. Lecture Notes in Computer Science, J. Pieprzyk, H. Ghodosi, and E. Dawson, Eds., vol. 4586. Springer, 2007, pp. 96–106.

[6] G. Cohen and G. Zémor, "Generalized coset schemes for the wire-tap channel: application to biometrics," in *IEEE International Symposium on Information Theory, Chicago*, 2004, p. 46.

[7] ——, "The wire-tap channel applied to biometrics," in *International Symposium on Information Theory and Applications, Parma*, 2004.

[8] ——, "Syndrome-coding for the wiretap channel revisited," in *ITW'06, IEEE Information Theory Workshop, Chengdu*, 2006, pp. 33–36.

[9] X. Boyen, Y. Dodis, J. Katz, R. Ostrovsky, and A. Smith, "Secure remote authentication using biometric data." in *EUROCRYPT*, ser. Lecture Notes in Computer Science, R. Cramer, Ed., vol. 3494. Springer, 2005, pp. 147–163.

[10] G. I. Davida and Y. Frankel, "Perfectly secure authorization and passive identification for an error tolerant biometric system." in *IMA Int. Conf.*, ser. Lecture Notes in Computer Science, M. Walker, Ed., vol. 1746. Springer, 1999, pp. 104–113.

[11] Y. Dodis, L. Reyzin, and A. Smith, "Fuzzy extractors: How to generate strong keys from biometrics and other noisy data." in *EUROCRYPT*, ser. Lecture Notes in Computer Science, C. Cachin and J. Camenisch, Eds., vol. 3027. Springer, 2004, pp. 523–540.

[12] P. Tuyls and J. Goseling, "Capacity and examples of template-protecting biometric authentication systems." in *ECCV Workshop BioAW*, ser. Lecture Notes in Computer Science, D. Maltoni and A. K. Jain, Eds., vol. 3087. Springer, 2004, pp. 158–170.

[13] B. Schoenmakers and P. Tuyls, *Security with Noisy Data*. Springer Verlag, 2007, ch. Computationally Secure Authentication with Noisy Data, pp. 141–152.

[14] T. A. M. Kevenaar, G. J. Schrijen, M. van der Veen, A. H. M. Akkermans, and F. Zuo, "Face recognition with renewable and privacy preserving binary templates," in *AutoID'2005, 17-18 October 2005, Buffalo, New York*. Washington, DC, USA: IEEE Computer Society, 2005, pp. 21–26.

[15] P. Tuyls, A. H. M. Akkermans, T. A. M. Kevenaar, G. J. Schrijen, A. M. Bazen, and R. N. J. Veldhuis, "Practical biometric authentication with template protection." in *Audio-and Video-Based Biometrie Person Authentication*, T. Kanade, A. K. Jain, and N. K. Ratha, Eds., vol. 3546. Springer, 2005, pp. 436–446.

[16] F. Hao, R. Anderson, and J. Daugman, "Combining crypto with biometrics effectively," *IEEE Transactions on Computers*, vol. 55, no. 9, pp. 1081–1088, 2006.

[17] J. Daugman, "The importance of being random: statistical principles of iris recognition," *Pattern Recognition*, vol. 36, no. 2, pp. 279–291, 2003.

[18] R. Wildes, *Biometric Authentication : Technologies, Systems, Evaluations and Legal Issues*. Springer, 2005, ch. 3. Iris recognition.

[19] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. Wiley, 2006.

[20] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-correcting Codes*. North-Holland, 1988.

[21] C. E. Shannon, "A mathematical theory of communication," *The Bell System Technical Journal*, vol. 27, pp. 379–423, 623–656, 1948.

[22] X. Liu, K. W. Bowyer, and P. J. Flynn, "Iris Recognition and Verification Experiments with Improved Segmentation Method," in *AutoID'2005, 17-18 October 2005, Buffalo, New York*, 2005.

[23] National Institute of Standards and Technology (NIST), "Iris Challenge Evaluation," http://iris.nist.gov/ICE, 2005.

[24] CASIA, "Chinese academy of science, institute of automation," URL : http://www.sinobiometrics.com/Database.htm.

[25] J. Daugman, "High confidence visual recognition of persons by a test of statistical independence." *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 15, no. 11, pp. 1148–1161, 1993.

[26] R. Wildes, "Automated iris recognition: An emerging biometric technology," *Proceedings of the IEEE*, vol. 85, no. 9, pp. 1348–1363, 1997.

[27] K. Hollingsworth, K. W. Bowyer, and P. J. Flynn, "All iris code bits are not created equal," in *Biometrics: Theory, Applications, and Systems, 2007. BTAS 2007. First IEEE International Conference on*, Sept 2007.

[28] S. Prabakhar, A. K. Jain, D. Maio, and D. Maltoni, *Handbook of Fingerprint Recognition*. Springer-Verlag New York, Inc., 2003.

[29] S. Draper, A. Khisti, E. Martinian, A. Vetro, and J. S. Yedidia, "Using distributed source coding to secure fingerprint biometrics," Mitsubishi Electrical Research Laboratories, Tech. Rep. TR 2007-005, Jan. 2007.

[30] J.-P. M. G. Linnartz and P. Tuyls, "New shielding functions to enhance privacy and prevent misuse of biometric templates." in *Audio- and Video-Based Biometric Person Authentication*, ser. LNCS, J. Kittler and M. S. Nixon, Eds., vol. 2688. Springer, 2003, pp. 393–402.

[31] J. Bringer, H. Chabanne, and B. Kindarji, "The best of both worlds: Applying secure sketches to cancelable biometrics," in *Science of Computer Programming*, 2007, to appear. Presented at WISSec'07.

[32] A. M. Bazen and R. N. J. Veldhuis, "Detection of cores in fingerprints with improved dimension reduction," in *4th IEEE Benelux Signal Processing Symposium(SPS-2004), Hilvarenbeek, The Netherlands*, 2004, pp. 41–44.

[33] A. M. Bazen and S. H. Gerez, "Systematic methods for the computation of the directional fields and singular points of fingerprints." *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 905–919, 2002.

[34] A. M. Bazen and R. N. J. Veldhuis, "Likelihood-ratio-based biometric verification." *IEEE Trans. Circuits Syst. Video Techn.*, vol. 14, no. 1, pp. 86–94, 2004.

[35] "Fingerprint verification competition," http://bias.csr.unibo.it/fvc2000/.

[36] P. J. Phillips, H. Moon, S. A. Rizvi, and P. J. Rauss, "The FERET evaluation methodology for face-recognition algorithms." *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 10, pp. 1090–1104, 2000.

[37] M. Weber, "Frontal face dataset," http://www.vision.caltech.edu/html-files/archive, California Institute of Technology, 1999.

[38] N. Wiberg, "Codes and decoding on general graphs," Ph.D. dissertation, Linkoping University, Linkoping, Sweden, 1996.

[39] R. M. Tanner, "A recursive approach to low-complexity codes," *IEEE Trans. on Information Theory*, vol. 27, pp. 533–547, 1981.

[40] D. Muller, "Application of boolean algebra to switching circuit design and to error detection," *IEEE Trans. on Electronic Computers*, vol. 3, pp. 6–12, 1954.

[41] I. Reed, "A class of multiple-error-correcting codes and their decoding scheme," *IEEE Trans. on Information Theory*, vol. 4, pp. 38–42, 1954.

[42] J. Daugman, "New methods in iris recognition," *Systems, Man, and Cybernetics, Part B, IEEE Transactions on*, vol. 37, no. 5, pp. 1167–1175, Oct. 2007.