Hindawi Publishing Corporation
EURASIP Journal on Advances in Signal Processing
Volume 2009, Article ID 784834, 16 pages
doi:10.1155/2009/784834

Research Article

Biometric Quantization through Detection Rate Optimized Bit Allocation

C. Chen,[1] R. N. J. Veldhuis,[1] T. A. M. Kevenaar,[2] and A. H. M. Akkermans[2]

[1] *Signals and Systems Group, Faculty of Electrical Engineering, University of Twente, P. O. Box 217, 7500 AE Enschede, The Netherlands*
[2] *Philips Research, High Tech Campus, 5656 AE Eindhoven, The Netherlands*

Correspondence should be addressed to C. Chen, c.chen@utwente.nl

Received 23 January 2009; Accepted 8 April 2009

Recommended by Yasar Becerikli

Extracting binary strings from real-valued biometric templates is a fundamental step in many biometric template protection systems, such as fuzzy commitment, fuzzy extractor, secure sketch, and helper data systems. Previous work has been focusing on the design of optimal quantization and coding for each single feature component, yet the binary string—concatenation of all coded feature components—is not optimal. In this paper, we present a detection rate optimized bit allocation (DROBA) principle, which assigns more bits to discriminative features and fewer bits to nondiscriminative features. We further propose a dynamic programming (DP) approach and a greedy search (GS) approach to achieve DROBA. Experiments of DROBA on the FVC2000 fingerprint database and the FRGC face database show good performances. As a universal method, DROBA is applicable to arbitrary biometric modalities, such as fingerprint texture, iris, signature, and face. DROBA will bring significant benefits not only to the template protection systems but also to the systems with fast matching requirements or constrained storage capability.

1. Introduction

The idea of extracting binary biometric strings was originally motivated by the increasing concern about biometric template protection [1]. Some proposed systems, such as fuzzy commitment [2], fuzzy extractor [3, 4], secure sketch [5], and helper data systems [6–9], employ a binary biometric representation. Thus, the quality of the binary string is crucial to their performances. Apart from the template protection perspective, binary biometrics also merit fast matching and compressed storage, facilitating a variety of applications utilizing low-cost storage media. Therefore, extracting binary biometric strings is of great significance. As shown in Figure 1, a biometric system with binary representation can be generalized into the following three modules.

Feature Extraction. This module aims to extract independent, reliable, and discriminative features from biometric raw measurements. Classical techniques used in this step are, among others, Principle Component Analysis (PCA) and Linear Discriminant Analysis (LDA) [10].

Bit Extraction. This module aims to transform the real-valued features into a fixed-length binary string. Biometric information is well known for its uniqueness. Unfortunately, due to sensor and user behavior, it is inevitably noisy, which leads to intraclass variations. Therefore, it is desirable to extract binary strings that are not only discriminative, but also have low intraclass variations. In other words, both a low false acceptance rate (FAR) and a low false rejection rate (FRR) are required. Additionally, from the template protection perspective, the bits, generated from an imposter, should be statistically independent and identically distributed (*i.i.d.*), in order to maximize the effort of an imposter in guessing the genuine template. Presumably, the real-valued features obtained from the feature extraction step are independent, reliable, and discriminative. Therefore, a quantization and coding method is needed to keep such properties in the binary domain. So far, a variety of such methods have been published, of which an overview will be given in Section 2.

Binary String Classification. This module aims to verify the binary strings with a binary string-based classifier. For

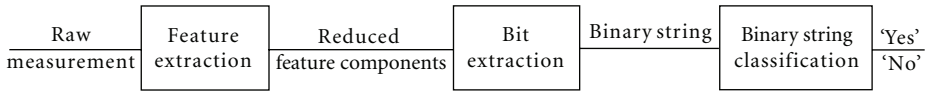

Figure 1: Three modules of a biometric system with binary representation.

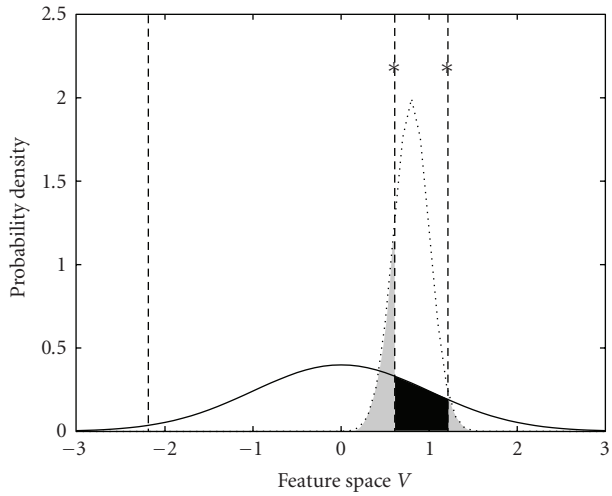

Figure 2: An illustration of the FAR (black) and the FRR (gray), given the background PDF (solid), the genuine user PDF (dot), and the quantization intervals (dash), where the genuine user interval is marked as $*$.

instance, the Hamming distance classifier bases its decision on the number of errors between two strings. Alternatively, the binary strings can be verified through a template protection process, for example, fuzzy commitment [2], fuzzy extractor [3, 4], secure sketch [5], and helper data systems [6–9]. Encrypting the binary strings by using a one-way function, these template protection systems verify binary strings in the encrypted domain. Usually the quantization methods in the bit extraction module cannot completely eliminate the intraclass variation. Thus employing a strict one-way function will result in a high FRR. To solve this problem, error correcting techniques are integrated to further eliminate the intra-class variation in the binary domain. Furthermore, randomness is embedded to avoid cross-matching.

This paper deals with the bit extraction module, for which we present a detection rate optimized bit allocation principle (DROBA) that transforms a real-valued biometric template into a fixed-length binary string. Binary strings generated by DROBA yield a good FAR and FRR performance when evaluated with a Hamming distance classifier.

In Section 2 an overview is given of known bit extraction methods. In Section 3 we present the DROBA principle with two realization approaches: dynamic programming (DP) and greedy search (GS), and their simulation results are illustrated in Section 4. In Section 5, we give the experimental results of DROBA on the FVC2000 fingerprint database [11] and the FRGC face database [12]. In Section 6 the results are discussed and conclusions are drawn in Section 7.

2. Overview of Bit Extraction Methods

A number of bit extraction methods, based on quantization and coding, have been proposed in biometric applications [6–8, 13–16]. In general, these methods deal with two problems: (1) how to design an optimal quantization and coding method for a single feature, and (2) how to compose an optimal binary string from all the features.

So far, most of the published work has been focusing on designing the optimal quantization intervals for individual features. It is known that, due to the inter- and intraclass variation, every single feature can be modeled by a background probability density function (PDF) p_b and a genuine user PDF p_g, indicating the probability density of the whole population and the genuine user, respectively. Given these two PDFs, the quantization performance of a single feature i, with an arbitrary b_i-bit quantizer, is then quantified as the theoretical FAR α_i:

$$\alpha_i(b_i) = \int_{Q_{\text{genuine},i}(b_i)} p_{b,i}(v)dv, \tag{1}$$

and FRR β_i, given by

$$\delta_i(b_i) = \int_{Q_{\text{genuine},i}(b_i)} p_{g,i}(v)dv, \tag{2}$$

$$\beta_i(b_i) = 1 - \delta_i(b_i), \tag{3}$$

where $Q_{\text{genuine},i}$ represents the genuine user interval into which the genuine user is expected to fall, and δ_i represents the corresponding detection rate. An illustration of these expressions is given in Figure 2. Hence, designing quantizers for a single feature is to optimize its FAR (1) and FRR (3).

Linnartz and Tuyls proposed a method inspired by Quantization Index Modulation [6]. As depicted in Figure 3(a), the domain of the feature v is split into fixed intervals of width q. Every interval is alternately labeled using a "0" or a "1." Given a random bit string s, a single bit of s is embedded per feature by generating an offset for v so that v ends up in the closest interval that has the same label as the bit to be embedded.

Vielhauer et al. [13] introduced a user-specific quantizer. As depicted in Figure 3(b), the genuine interval $[I_{\min}(1 - t), I_{\max}(1 + t)]$ is determined according to the minimum $I_{\min}$ and maximum $I_{\max}$ value of the samples from the genuine user, together with a tolerance parameter t. The remaining intervals are constructed with the same width as the genuine interval.

Hao and Wah [14] and Chang et al. [15] employed a user-specific quantizer as shown in Figure 3(c). The genuine interval is $[\mu - k\sigma, \mu + k\sigma]$, where μ and σ are the mean and the standard deviation of the genuine user PDF, and k

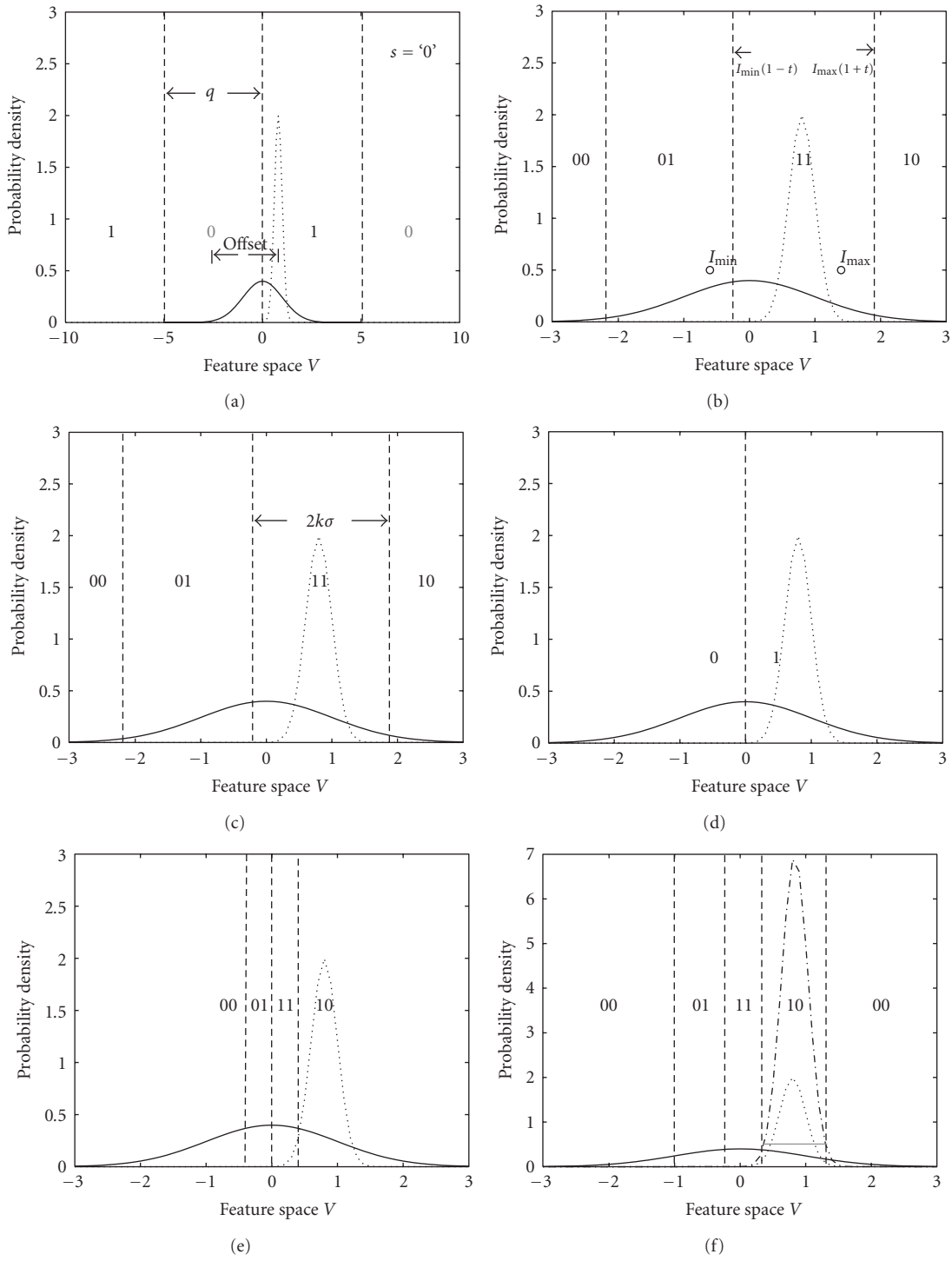

FIGURE 3: Illustration of the quantizers for a single feature i, and the corresponding Gray codes. The background PDF $p_b(v, 0, 1)$ (solid); the genuine user PDF $p_g(v, \mu, \sigma)$ (dot); the quantization intervals (dash). (a) QIM quantization; (b) Vielhauer's quantizer; (c) Chang's multibits quantizer; (d) fixed one-bit quantizer; (e) fixed two-bits quantizer; (f) likelihood ratio-based quantizer, the likelihood ratio (dash-dot), threshold (gray).

is an optimization parameter. The remaining intervals are constructed with the same width $2k\sigma$.

The quantizers in [6, 13–15] have equal-width intervals. However, considering a template protection application, this leads to potential threats, because samples tend to have higher probabilities in some quantization intervals and thus an imposter can search the genuine interval by guessing the one with the highest probability. Therefore, quantizers

with equal-probability intervals or equal-frequency intervals [7, 16] have been proposed.

Tuyls et al. [7] and Teoh et al. [17] employed a 1-bit fixed quantizer as shown in Figure 3(d). Independent of the genuine user PDF, this quantizer splits the domain of the feature v into two fixed intervals using the mean of the background PDF as the quantization boundary. As a result, both intervals contain 0.5 background probability mass. The interval that the genuine user is expected to fall into is referred to as the genuine interval.

Chen et al. [16] extended the 1-bit fixed quantizer into multibits. A b-bit fixed quantizer contains 2^b intervals, symmetrically constructed around the mean of the background PDF, with equally 2^{-b} background probability mass. Figure 3(e) illustrates an example of $b = 2$. In the same paper [16], a user-specific likelihood ratio-based multibits quantizer was introduced, as shown in Figure 3(f). For a b-bit quantizer, a likelihood ratio threshold first determines a genuine interval with 2^{-b} background probability mass. The remaining intervals are then constructed with equal 2^{-b} background probability mass. The left and right tail are combined as one wrap-around interval, excluding its possibility as a genuine interval. The likelihood ratio-based quantizer provides the optimal FAR and FRR performances in the Neyman-Pearson sense.

The equal-probability intervals in both the fixed quantizer and the likelihood ratio-based quantizer ensure independent and identically distributed bits for the imposters, which meets the requirement of template protection systems. For this reason, we take these two quantizers into consideration in the following sections. Additionally, because of the equal-probability intervals, the FAR of both quantizers for feature i becomes

$$\alpha_i(b_i) = 2^{-b_i}. \tag{4}$$

With regard to composing the optimal binary string from D features, the performance of the entire binary string can be quantified by the theoretical overall FAR α and detection rate δ:

$$\alpha(b_1, \ldots, b_D) = \prod_{i=1}^{D} \alpha_i(b_i), \tag{5}$$

$$\delta(b_1, \ldots, b_D) = \prod_{i=1}^{D} \delta_i(b_i), \qquad \sum_{i=1}^{D} b_i = L. \tag{6}$$

Given (4), the overall FAR in (5) shows a fixed relationship with L:

$$\alpha(b_1, \ldots, b_D) = 2^{-L}. \tag{7}$$

Hence composing the optimal binary string is to optimize the detection rate at a given FAR value. In [7, 8, 16], a fixed bit allocation principle (FBA)—with a fixed number of bits assigned to each feature—was proposed. Obviously, the overall detection rate of the FBA is not optimal, since we would expect to assign more bits to discriminative features and fewer bits to nondiscriminative features. Therefore, in the next section, we propose the DROBA principle, which gives the optimal overall detection rate.

3. Detection Rate Optimized Bit Allocation

(DROBA). In this section, we first give the description of the DROBA principle. Furthermore, we introduce both a dynamic programming and a greedy search approach to search for the solution.

3.1. Problem Formulation. Let D denote the number of features to be quantized; L, the specified binary string length; $b_i \in \{0, \ldots, b_{\max}\}$, $i = 1, \ldots, D$, the number of bits assigned to feature i; $\delta_i(b_i)$, the detection rate of feature i, respectively. Assuming that all the D features are independent, our goal is to find a bit assignment $\{b_i^*\}$ that maximizes the overall detection rate in (6):

$$\begin{aligned} \{b_i^*\} &= \arg \max_{\sum_{i=1}^{D} b_i = L} \delta(b_1, \ldots, b_D) \\ &= \arg \max_{\sum_{i=1}^{D} b_i = L} \prod_{i=1}^{D} \delta_i(b_i). \end{aligned} \tag{8}$$

Note that by maximizing the overall detection rate, we in fact maximize the probability of all the features simultaneously staying in the genuine intervals, more precisely, the probability of a zero bit error for the genuine user. Furthermore, considering using a binary string classifier, essentially the overall FAR α in (5) and the overall detection rate δ in (6) correspond to the point with the minimum FAR and minimum detection rate on its theoretical receiver operating characteristic curve (ROC), as illustrated in Figure 4. We know that α is fixed in (7), by maximizing δ, DROBA in fact provides a theoretical maximum lower bound for the ROC curve. Since DROBA only maximizes the point with minimum detection rate, the rest of the ROC curve, which relies on the specific binary string classifier, is not yet optimized. However, we would expect that with the maximum lower bound, the overall ROC performance of any binary string classifier is to some extent optimized.

The optimization problem in (8) can be solved by a brute force search of all possible bit assignments $\{b_i\}$ mapping D features into L bits. However, the computational complexity is extremely high. Therefore, we propose a dynamic programming approach with reasonable computational complexity. To further reduce the computational complexity, we also propose a greedy search approach, for which the optimal solution is achieved under additional requirements to the quantizer.

3.2. Dynamic Programming (DP) Approach. The procedure to search for the optimal solution for a genuine user is recursive. That is, given the optimal overall detection rates $\delta^{(j-1)}(l)$ for $j-1$ features at string length l, $l = 0, \ldots, (j-1) \times b_{\max}$:

$$\delta^{(j-1)}(l) = \max_{\sum b_i = l,\, b_i \in \{0, \ldots, b_{\max}\}} \prod_{i=1}^{j-1} \delta_i(b_i), \tag{9}$$

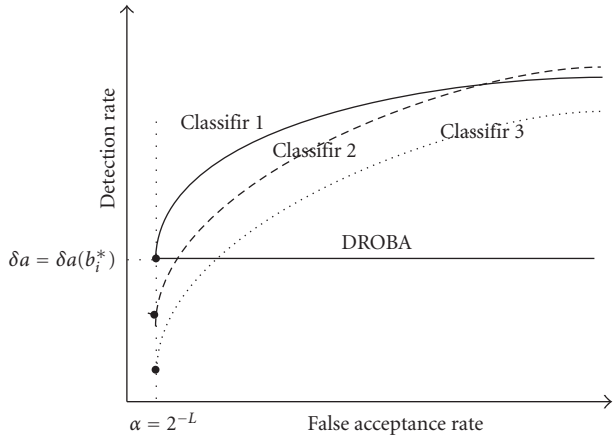

Figure 4: Illustration of the maximum lower bound for the theoretical ROC curve provided by DROBA.

the optimal detection rates $\delta^{(j)}(l)$ for j features are computed as

$$\delta^{(j)}(l) = \max_{\substack{b'+b''=l,\\ b'\in\{0,\ldots,(j-1)\times b_{\max}\},\\ b''\in\{0,\ldots,b_{\max}\}}} \delta^{(j-1)}(b')\delta_j(b''), \tag{10}$$

for $l = 0,\ldots, j\times b_{\max}$. Note that $\delta^{(j)}(l)$ needs to be computed for all string lengths $l \in \{0,\ldots, j\times b_{\max}\}$. Equation (10) tells that the optimal detection rate for j features at string length l is derived from maximizing the product of an optimized detection rate for $j-1$ features at string length b' and the detection rate of the jth feature quantized to b'' bits, while $b'+b''=l$. In each iteration step, for each value of l in $\delta^{(j)}(l)$, the specific optimal bit assignments of features must be maintained. Let $\{b_i(l)\}$, $i=1,\ldots,j$ denote the optimal bit assignments for j features at binary string length l such that the ith entry corresponds to the ith feature. Note that the sum of all entries in $\{b_i(l)\}$ equals l, that is, $\sum_{i=1}^{j} b_i(l) = l$. If $\widehat{b}'$ and $\widehat{b}''$ denote the values of b' and b'' that correspond to the maximum value $\delta^{(j)}(l)$ in (10), the optimal assignments are updated by

$$\begin{aligned} b_i(l) &= b_i\left(\widehat{b}'\right), \quad i=1,\ldots,j-1,\\ b_j(l) &= \widehat{b}''. \end{aligned} \tag{11}$$

The iteration procedure is initialized with $j=0$, $b_0(0)=0$, and $\delta^{(0)}(0)=1$ and terminated when $j=D$. After D iterations, we obtain a set of optimal bit assignments for every possible bit length $l=0,\ldots,D\times b_{\max}$, we only need to pick the one that corresponds to L: the final solution $\{b_i^*\}=\{b_i(L)\}$, $i=1,\ldots,D$. This iteration procedure can be formalized into a dynamic programming approach [18], as described in Algorithm 1.

Essentially, given L and arbitrary $\delta_i(b_i)$, the dynamic programming approach optimizes (8). The proof of its optimality is presented in Appendix A. This approach is independent of the specific type of the quantizer, which determines the behavior of $\delta_i(b_i)$. The user-specific optimal

Input:
$D, L, \delta_i(b_i),\ b_i \in \{0,\ldots,b_{\max}\},\ i=1,\ldots,D,$
Initialize:
$j=0,$
$b_0(0)=0,$
$\delta^{(0)}(0)=1,$
while $j<D$ **do**
$j=j+1,$
$\widehat{b}',\widehat{b}'' = \arg \max_{\substack{b'+b''=l,\\ b'\in\{0,\ldots,(j-1)\times b_{\max}\},\\ b''\in\{0,\ldots,b_{\max}\}}} \delta^{(j-1)}(b')\delta_j(b''),$
$\delta^{(j)}(l) = \delta^{(j-1)}(\widehat{b}')\delta_j(\widehat{b}''),$
$b_i(l) = b_i(\widehat{b}'),\ i=1,\ldots,j-1,$
$b_j(l) = \widehat{b}'',$
for $l=0,\ldots,j\times b_{\max},$
endwhile
Output:
$\{b_i^*\}=\{b_i(L)\},\ i=1,\ldots,D.$

Algorithm 1: Dynamic programming approach for DROBA.

solution $\{b_i^*\}$ is feasible as long as $0 \le L \le (D\times b_{\max})$. The number of operations per iteration step is about $O((j-1)\times b_{\max}^2)$, leading to a total number of operations of $O(D^2\times b_{\max}^2)$, which is significantly less than that of a brute force search. However, this approach becomes inefficient if $L \ll D\times b_{\max}$, because a D-fold iteration is always needed, regardless of L.

3.3. Greedy Search (GS) Approach. To further reduce the computational complexity, we introduce a greedy search approach. By taking the logarithm of the detection rate, the optimization problem in (8) is now equivalent to finding a bit assignment $\{b_i^*\}$, $i=1,\ldots,D$ that maximizes:

$$\sum_{i=1}^{D} \log(\delta_i(b_i)), \tag{12}$$

under the constraint of a total number of L bits. In [19], an equivalent problem of minimizing quantizer distortion, given an upper bound to the bit rate, is solved by first rewriting it as an unconstrained Lagrange minimization problem. Thus in our case we define the unconstrained Lagrange maximization problem as

$$\max_{b_i,\lambda\ge 0}\left[\sum_{i=1}^{D}\log(\delta_i(b_i)) - \lambda\sum_{i=1}^{D} b_i\right]. \tag{13}$$

We know that the detection rate of a feature is monotonically non-increasing with the number of quantization bits. Therefore, we can construct an L-bit binary string, by iteratively assigning an extra bit to the feature that gives the minimum detection rate loss, as seen in Algorithm 2. Suppose $\{b_i(l)\}$, $i=1,\ldots,D$ gives the bit assignments of all D features at binary string length l, we compute $\Delta_i(l)$ for

Input:
$D, L, \log(\delta_i(b_i)),\ b_i \in \{0,\ldots,b_{\max}\},\ i = 1,\ldots,D,$
Initialize :
$$l = 0,$$
$$b_i(0) = 0,$$
$$\log(\delta_i(b_i(0))) = 0,$$
while $l < L$ **do**
$$\Delta_i(l) = \log(\delta_i(b_i(l))) - \log(\delta_i(b_i(l)+1)),$$
$$i_{\max} = \arg\min_i \Delta_i(l),$$
$$b_i(l+1) = \begin{cases} b_i(l)+1, & i=i_{\max}, \\ b_i(l), & \text{otherwise.} \end{cases}$$
$$l = l+1,\ i = 1,\ldots,D,$$
endwhile
Output:
$$\{b_i^*\} = \{b_i(L)\},\ i = 1,\ldots,D.$$

Algorithm 2: Greedy search approach for DROBA.

each feature, representing the loss of the log detection rate by assigning one more bit to that feature:

$$\Delta_i(l) = \log(\delta_i(b_i(l))) - \log(\delta_i(b_i(l)+1)), \quad i = 1,\ldots,D. \tag{14}$$

Hence the extra bit that we select to construct the $(l+1)$-bit binary string comes from the feature $i_{\max}$ that gives the minimum detection rate loss, and no extra bits are assigned to the unchosen feature components:

$$i_{\max} = \arg\min_i \Delta_i(l),$$
$$b_i(l+1) = \begin{cases} b_i(l)+1, & i = i_{\max}, \\ b_i(l), & \text{otherwise.} \end{cases} \tag{15}$$

The iteration is initialized with $l = 0$, $b_i(0) = 0$, $\log(\delta_i(b_i(0))) = 0$, $i = 1,\ldots,D$ and terminated when $l = L$. The final solution is $\{b_i^*\} = \{b_i(L)\}$, $i = 1,\ldots,D$.

To ensure the optimal solution of this greedy search approach, the quantizer has to satisfy the following two conditions:

(1) $\log(\delta_i)$ is a monotonically non-increasing function of b_i,

(2) $\log(\delta_i)$ is a concave function of b_i.

The number of operations of the greedy search is about $O(L \times D)$, which is related with L. Compared with the dynamic programming approach with $O(D^2 \times b_{\max}^2)$, greedy search becomes significantly more efficient if $L \ll D \times b_{\max}^2$, because only an L-fold iteration needs to be conducted.

The DROBA principle provides the bit assignment $\{b_i^*\}$, indicating the number of quantization bits for every single feature. The final binary string for a genuine user is the concatenation of the quantization and coding output under $\{b_i^*\}$.

4. Simulations

We investigated the DROBA principle on five randomly generated synthetic features. The background PDF of each feature was modeled as a Gaussian density $p_{b,i}(v) = N(v, 0, 1)$, with zero mean and unit standard deviation. Similarly, the genuine user PDF was modeled as Gaussian density $p_{g,i}(v) = N(v, \mu_i, \sigma_i)$, $\sigma_i < 1$, $i = 1,\ldots,5$, as listed in Table 1. For every feature, a list of detection rates $\delta_i(b_i)$, $b_i \in \{0,\ldots,b_{\max}\}$ with $b_{\max} = 3$, was computed from (2). Using these detection rates as input, the bit assignment was generated according to DROBA. Depending on the quantizer type and the bit allocation approach, the simulations were arranged as follows:

Table 1: The randomly generated genuine user PDF $N(v, \mu_i, \sigma_i)$, $i = 1,\ldots,5$.

i	1	2	3	4	5
μ_i	−0.12	−0.07	0.49	−0.60	−0.15
σ_i	0.08	0.24	0.12	0.19	0.24

(i) FQ-DROBA (DP): fixed quantizer combined with DROBA, by using the dynamic programming approach;

(ii) FQ-DROBA (GS): fixed quantizer combined with DROBA, by using the greedy search approach;

(iii) LQ-DROBA (DP): likelihood ratio-based quantizer combined with DROBA, by using the dynamic programming approach;

(iv) LQ-DROBA (GS): likelihood ratio-based quantizer combined with DROBA, by using the greedy search approach;

(v) FQ-FBA (b): fixed quantizer combined with the fixed b-bit allocation principle [16];

(vi) LQ-FBA (b): likelihood ratio-based quantizer combined with the fixed b-bit allocation principle.

We computed the overall detection rate (6), based on the bit assignment corresponding to various specified string length L. The logarithm of the detection rates of the overall detection rate are illustrated in Figure 5. Results show that DROBA principle generates higher quality strings than the FBA principle. Moreover, DROBA has the advantage that an arbitrary length binary string can always be generated. Regarding the greedy search approach, we observe that the likelihood ratio based quantizer seems to satisfy the monotonicity and concaveness requirements, which explains the same optimal detection rate performance of LQ-DROBA (DP) and LQ-DROBA (GS). However, in the case of the fixed quantizer, some features in Table 1 do not satisfy the concaveness requirement for an optimal solution of GS. This explains the better performance of FQ-DROBA (DP) than FQ-DROBA (GS). Note that the performance of LQ-DROBA (DP) consistently outperforms FQ-DROBA (DP). This is because of the better performance of the likelihood ratio-based quantizer.

Table 2 gives the bit assignment $\{b_i^*\}$ of FQ-DROBA (DP) and FQ-DROBA (GS), at $L = 1,\ldots,15$. The result shows that the DROBA principle assigns more bits to discriminative features than the nondiscriminative features. We

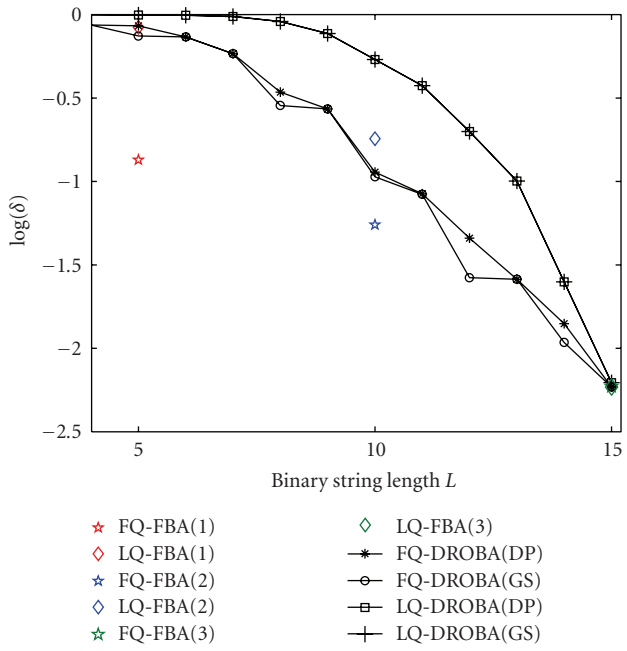

Figure 5: The log(δ) computed from the bit assignment, through model FQ-DROBA(DP), FQ-DROBA (GS), LQ-DROBA (DP), LQ-DROBA (GS), FQ-FBA (b), LQ-FBA (b), $b = 1, 2, 3$, on 5 synthetic features, at L, $L = 1, \ldots, 15$.

Table 2: The bit assignment $\{b_i^*\}$ of FQ-DROBA (DP) and FQ-DROBA (GS) at binary string length $L, L = 1, \ldots, 15$.

L	$\{b_i^*\}$ of FQ-DROBA (DP)	$\{b_i^*\}$ of FQ-DROBA (GS)
0	[0 0 0 0 0]	[0 0 0 0 0]
1	[0 0 1 0 0]	[0 0 1 0 0]
2	[0 0 1 1 0]	[0 0 1 1 0]
3	[2 0 1 0 0]	[1 0 1 1 0]
4	[2 0 1 1 0]	[2 0 1 1 0]
5	[3 0 1 1 0]	[2 0 2 1 0]
6	[3 0 2 1 0]	[3 0 2 1 0]
7	[3 0 3 1 0]	[3 0 3 1 0]
8	[3 0 2 1 2]	[3 0 3 1 1]
9	[3 0 3 1 2]	[3 0 3 1 2]
10	[3 0 3 1 3]	[3 0 3 2 2]
11	[3 2 3 1 2]	[3 0 3 3 2]
12	[3 3 3 1 2]	[3 1 3 3 2]
13	[3 2 3 3 2]	[3 2 3 3 2]
14	[3 3 3 3 2]	[3 2 3 3 3]
15	[3 3 3 3 3]	[3 3 3 3 3]

observe that the dynamic programming approach sometimes shows a jump of assigned bits (e.g., from $L = 7$ to $L = 8$ of feature 5, with $\delta = 0.34$ at $L = 8$), whereas the bits assigned through the greedy search approach have to increase one step at a time (with $\delta = 0.28$ at $L = 8$). Such inflexibility proves that the greedy search approach does not provide the optimal solution in this example.

Table 3: Training, enrollment and verification data, number of users $\times$ number of samples per user (**n**), and the number of partitionings for FVC2000, FRGCt and FRGCs.

	Training	Enrollment	Verification	Partitionings
FVC2000	$80 \times n$	$30 \times 3n/4$	$30 \times n/4$	20
FRGCt	$210 \times n$	$65 \times 2n/3$	$65 \times n/3$	5
FRGCs	$150 \times n$	$48 \times 2n/3$	$48 \times n/3$	5

5. Experiments

We tested the DROBA principle on three data sets, derived from the FVC2000 (DB2) fingerprint database [11] and the FRGC (version 1) [12] face database.

(i) *FVC2000.* This is the FVC2000 (DB2) fingerprint data set, containing 8 images of 110 users. Images are aligned according to a standard core point position, in order to avoid a one-to-one alignment. The raw measurements contain two categories: the squared directional field in both x and y directions, and the Gabor response in 4 orientations (0, $\pi/4$, $\pi/2$, $3\pi/4$). Determined by a regular grid of 16 by 16 points with spacing of 8 pixels, measurements are taken at 256 positions, leading to a total of 1536 elements [7].

(ii) *FRGCt.* This is the total FRGC (version 1) face dataset, containing 275 users with various numbers of images, taken under both controlled and uncontrolled conditions. A set of standard landmarks, that is, eyes, nose, and mouth, are used to align the faces, in order to avoid a one-to-one alignment. The raw measurements are the gray pixel values, leading to a total of 8762 elements.

(iii) *FRGCs.* This is a subset of FRGCt, containing 198 users with at least 2 images per user. The images are taken under uncontrolled conditions.

Our experiments involved three steps: training, enrollment, and verification. In the training step, we extracted D independent features, via a combined PCA/LDA method [10] from a training set. The obtained transformation was then applied to both the enrollment and verification sets. In the enrollment step, for every target user, the DROBA principle was applied, resulting in a bit assignment $\{b_i^*\}$, with which the features were quantized and coded with a Gray code. The advantage of the Gray code is that the Hamming distance between two adjacent quantization intervals is limited to one, which results in a better performance of a Hamming distance classifier. The concatenation of the codes from D features formed the L-bit target binary string, which was stored for each target user together with $\{b_i^*\}$. In the verification step, the features of the query user were quantized and coded according to the $\{b_i^*\}$ of the claimed identity, and this resulted in a query binary string. Finally the verification performance was evaluated by a Hamming distance classifier. A genuine Hamming distance was computed if the target and the query string originate from the same identity, otherwise an imposter Hamming distance was computed. The detection error tradeoff (DET)

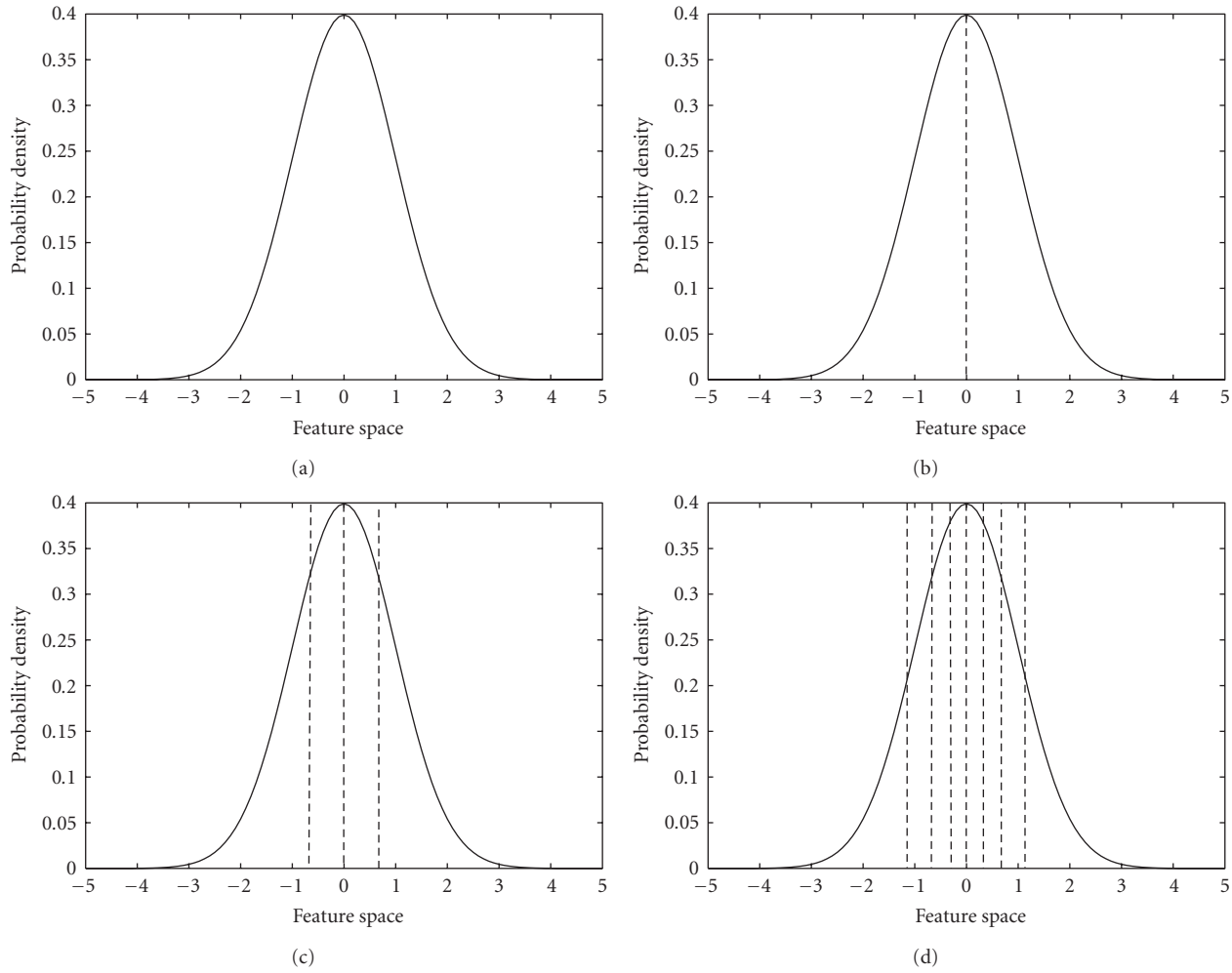

Figure 6: Illustration of the fixed quantizer with equal background probability mass in each interval: background PDF $p_{b,i}(v) = N(v, 0, 1)$ (dashed); quantization intervals (solid). (a) $b_i = 0$; (b) $b_i = 1$; (c) $b_i = 2$; (d) $b_i = 3$.

curve or the equal error rate (EER) was then constructed from these distances.

The users selected for training are different from those in the enrollment and verification. We repeated our experiment with a number of random partitionings. With, in total, n samples per user ($n = 8$ for FVC2000, n ranges from 6 to 48 for FRGCt, and n ranges from 4 to 16 for FRGCs), the division of the data is indicated in Table 3.

In our experiment, the detection rate was computed from the fixed quantizer (FQ) [7, 16]. According to the Central Limit Theorem, we assume that after the PCA/LDA transformation, with sufficient samples from the entire populations, the background PDF of every feature can be modeled as a Gaussian density $p_{b,i}(v) = N(v, 0, 1)$. Hence the quantization intervals are determined as illustrated in Figure 6. Furthermore, in DROBA, the detection rate plays a crucial role. Equation (2) shows that the accuracy of the detection rate is determined by the underlying genuine user PDF. Therefore, we applied the following four models.

(i) *Model 1.* We model the genuine user PDF as a Gaussian density $p_{g,i}(v) = N(v, \mu_i, \sigma_i)$, $i = 1, \ldots, D$. Besides, the user has sufficient enrollment samples, so that both the mean μ_i and the standard deviation σ_i are estimated from the enrollment samples. The detection rate is then calculated based on this PDF.

(ii) *Model 2.* We model the genuine user PDF as a Gaussian density $p_{g,i}(v) = N(v, \mu_i, \sigma_i)$, $i = 1, \ldots, D$, but there are not sufficient user-specific enrollment samples. Therefore, for each feature, we assume that the entire populations share the same standard deviation and thus the σ_i is computed from the entire populations in the training set. The μ_i, however, is still estimated from the enrollment samples. The detection rate is then calculated based on this PDF.

(iii) *Model 3.* In this model we do not determine a specific genuine user PDF. Instead, we compute a heuristic detection rate $\overline{\delta_i}$, based on the μ_i, estimated from the enrollment samples. The $\overline{\delta_i}$ is defined as

$$\overline{\delta_i}(b_i) = \begin{cases} 1, & d_{\mathrm{L},i}(b_i) \times d_{\mathrm{H},i}(b_i) > 1, \\ d_{\mathrm{L},i}(b_i) \times d_{\mathrm{H},i}(b_i), & \text{otherwise}, \end{cases} \tag{16}$$

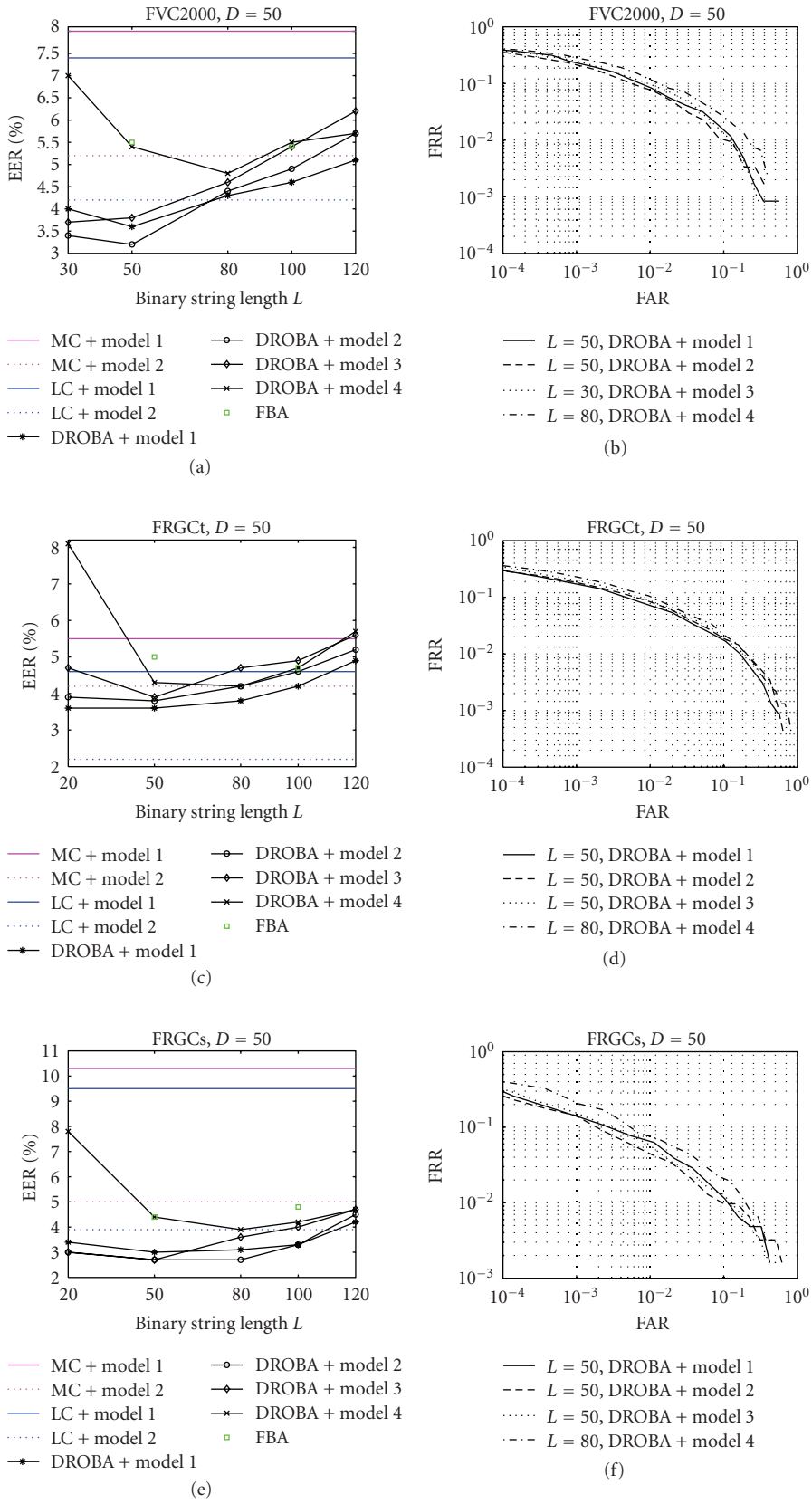

Figure 7: Experiment I: the EER performances of the binary strings generated under DROBA and FBA principles, compared with the real-value feature-based Mahalanobis distance classifier (MC) and likelihood-ratio classifier (LC), at $D = 50$, for (a) FVC2000, (c) FRGCt, and (e) FRGCs, with the DET of their best performances in (b), (d), and (f), respectively.

Table 4: Experiment II: the EER performances of DROBA + Model 1/2/3/4, FBA, MC + Model 1/2 and LC+Model 1/2, at $D = 50$, for (a) FVC2000, (b) FRGCt, and (c) FRGCs.

(a)

FVC2000	$D = 50$ EER = (%)				
	$L = 30$	50	80	100	120
DROBA + Model 1	4.0	**3.6**	4.3	4.6	5.1
DROBA + Model 2	3.4	**3.2**	4.4	4.9	5.7
DROBA + Model 3	**3.7**	3.8	4.6	5.4	6.2
DROBA + Model 4	7.0	5.4	**4.8**	5.5	5.7
FBA	—	5.5	—	5.4	—
MC + Model 1			8.0		
MC + Model 2			5.2		
LC + Model 1			7.4		
LC + Mode 2			4.2		

(b)

FRGCt	$D = 50$ EER = (%)				
	$L = 20$	50	80	100	120
DROBA + Model 1	3.6	**3.6**	3.8	4.2	4.9
DROBA + Model 2	3.9	**3.8**	4.2	4.6	5.2
DROBA + Model 3	4.7	**3.9**	4.7	4.9	5.6
DROBA + Model 4	8.1	4.3	**4.2**	4.7	5.7
FBA	—	5.0	—	4.7	—
MC + Model 1			5.5		
MC + Model 2			4.2		
LC + Model 1			4.6		
LC + Model 2			2.2		

(c)

FRGCs	$D = 50$ EER = (%)				
	$L = 20$	50	80	100	120
DROBA + Model 1	3.4	**3.0**	3.1	3.3	4.2
DROBA + Model 2	3.0	**2.7**	2.7	3.3	4.5
DROBA + Model 3	3.0	**2.7**	3.6	4.0	4.7
DROBA + Model 4	7.8	4.4	**3.9**	4.2	4.7
FBA	—	4.4	—	4.8	—
MC + Model 1			10.3		
MC + Model 2			5.0		
LC + Model 1			9.5		
LC + Model 2			3.9		

where $d_{L,i}(b_i)$ and $d_{H,i}(b_i)$ stand for the Euclidean distance of μ_i to the lower and the higher genuine user interval boundaries, when quantized into b_i bits.

(iv) *Model 4.* In this model the global detection rates are empirically computed from the entire populations in the training set. For every user, we compute the mean of feature i and evaluate this feature with the samples from the same user, at various quantization bits $b_i = 0, \ldots, b_{\max}$. At each b_i, the number of exact matches $n_{i,m}(b_i)$ as well as the total number of matches $n_{i,t}(b_i)$ are recorded. The detection rate of feature i with b_i bits quantization is then the ratio of $n_{i,m}(b_i)$ and $n_{i,t}(b_i)$ averaged over all users:

$$\hat{\delta}_i(b_i) = \frac{\sum_{\text{all users}} n_{i,m}(b_i)}{\sum_{\text{all users}} n_{i,t}(b_i)}. \tag{17}$$

We then repeat this process for all the features $i = 1, \ldots, D$. The detection rates $\hat{\delta}_i(b_i)$ are then used as input of DROBA. As a result, all the users share the same bit assignment.

Following the four models, experiments with DROBA were carried out and compared to the real-value based Mahalanobis distance classifier (MC), likelihood ratio classifier

Table 5: Experiment II: the EER performances of DROBA + Model 1/2/3/4, FBA, MC + Model 1/2, and LC + Model 1/2, at $L = 50$, for (a) FVC2000, (b) FRGCt, and (c) FRGCs.

(a)

FVC2000	$L = 50$ EER = (%)					
	$D = 20$	30	40	50	60	79
MC + Model 1	7.2	7.3	7.3	8.0	8.2	8.7
MC + Model 2	5.4	5.4	5.3	5.2	5.2	5.4
LC + Model 1	7.3	6.9	7.1	7.4	7.5	7.9
LC + Model 2	4.8	4.6	4.7	4.3	4.3	3.8
DROBA + Model 1	8.4	5.2	4.5	3.6	3.5	**2.9**
DROBA + Model 2	8.3	5.4	4.0	3.2	3.1	**2.7**
DROBA + Model 3	8.5	6.2	4.7	3.8	3.4	**2.8**
DROBA + Model 4	8.2	6.5	5.5	5.4	**5.4**	5.4

(b)

FRGCt	$L = 50$ EER = (%)				
	$D = 20$	50	80	100	120
MC + Model 1	4.9	5.5	6.9	8.1	9.0
MC + Model 2	3.8	4.2	5.7	6.2	6.9
LC + Model 1	4.5	4.6	5.3	5.8	6.3
LC + Model 2	2.7	2.2	2.2	2.2	2.2
DROBA + Model 1	7.0	3.6	3.0	3.0	**3.0**
DROBA + Model 2	7.2	3.8	3.8	3.7	**3.6**
DROBA + Model 3	7.7	4.0	**3.8**	3.9	4.2
DROBA + Model 4	7.3	4.3	**4.3**	4.3	4.3

(c)

FRGCs	$L = 50$ EER = (%)				
	$D = 20$	50	80	100	120
MC + Model 1	8.1	10.3	12.1	13.9	14.8
MC + Model 2	4.3	5.0	6.1	6.6	7.2
LC + Model 1	7.7	9.5	11.4	12.6	13.0
LC + Model 2	3.9	3.9	3.9	3.9	3.7
DROBA + Model 1	6.5	3.0	3.0	2.7	**2.4**
DROBA + Model 2	6.7	2.7	2.5	2.2	**2.1**
DROBA + Model 3	7.5	2.7	2.7	**2.6**	2.8
DROBA + Model 4	6.7	**4.4**	4.4	4.4	4.4

(LC), and the fixed bit allocation principle (FBA). Thus, in short, the experiments are described as follows.

(i) DROBA + Model 1/2/3/4: which generate the binary strings based on the fixed quantizer and the DROBA principle via the dynamic programming approach, where the detection rates are derived from Model 1/2/3/4, respectively. The binary strings are then compared with a Hamming distance classifier. Notation DROBA here refers to FQ-DROBA (DP) in Section 4.

(ii) FBA: which generates the binary strings based on the fixed quantizer and the fixed bit allocation principle [7, 8, 16], which assigns the same number of bits to all features. The binary strings are then compared with a Hamming distance classifier. Notation FBA here refers to FQ-FBA (b) in Section 4.

(iii) MC + Model 1/2: which employ a Mahalanobis (norm2) distance classifier [20] on the real-valued features, where the genuine user PDF is derived from Model 1 or 2, respectively;

(iv) LC + Model 1/2: which employ a likelihood ratio classifier [21] on the real-valued features, where the genuine user PDF is derived from Model 1 or 2, respectively.

In the experiments the maximum number of quantization bits for each feature was fixed to $b_{\max} = 3$. This allows us to investigate the impact of the $D - L$ configuration on the DROBA performances. We conducted

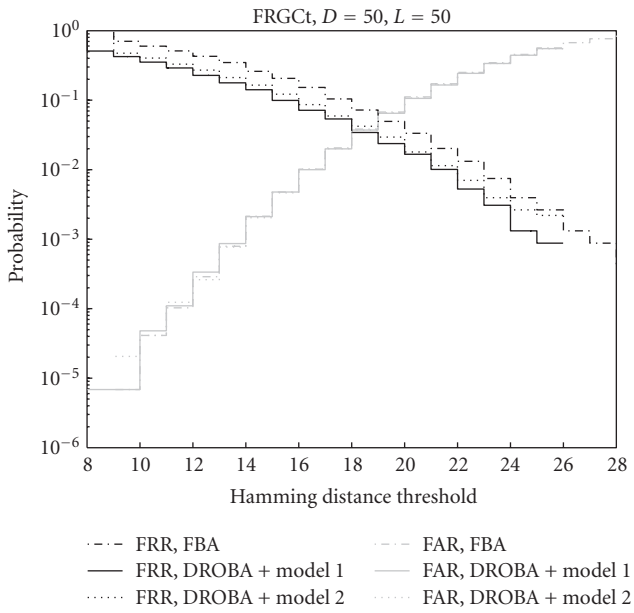

Figure 8: The FAR and FRR performances of FBA and DROBA + Model 1/2, at $D = 50$, $L = 50$.

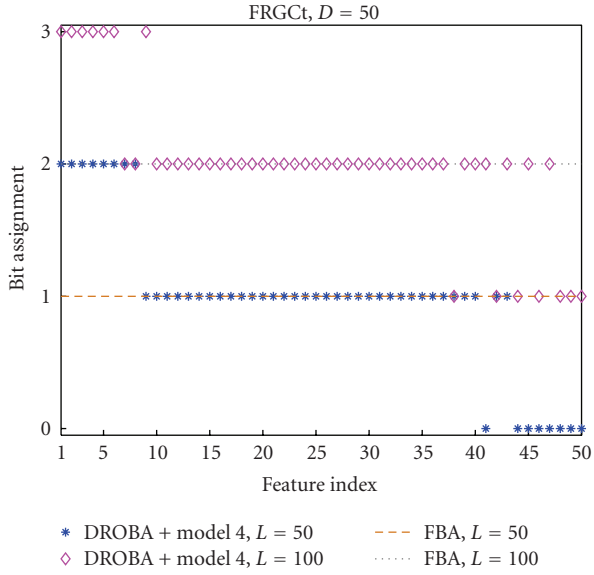

Figure 9: The bit assignment of FBA and DROBA + Model 4, at $D = 50$, $L = 50$ and 100, for FRGCt.

two experiments: in Experiment I, given D features, we evaluated the verification performances at various binary string lengths L; in Experiment II, given a budget of L bits, we investigated the verification performances with various numbers of features D. Additionally, since experimental results of DP and GS approaches are almost the same, we only present the result of DP.

In Experiment I, Figures 7(a), 7(c), 7(e), and Table 4 show the corresponding EER performances for FVC2000, FRGCt, and FRGCs, given $D = 50$ features after PCA/LDA transformation. All DROBA + Model 1/2/3/4 show similar behavior: as L increases, the performance first improves, and then starts to degrade. This could be explained by (6) and (7) that given D, a low L ends up in a high FAR bound, contrarily a high L ends up in a low detection rate bound. Therefore, a moderate L might provide a good tradeoff between FAR and FRR. For FVC2000 and FRGCs, DROBA + Model 1 and DROBA + Model 2 reveal similar performances, whereas DROBA + Model 3 has slightly worse performance. In the case of FRGCt, DROBA + Model 1 constantly outperforms DROBA + Model 2/3. As a global implementation, DROBA + Model 4 performs worse than DROBA + Model 1/2 for all three datasets, but the difference decreases as L increases. When compared to DROBA + Model 3, despite a rather poor performance at small L, DROBA + Model 4 gives comparable performances at large L. To summarize, given D features, by applying DROBA, there exists an L that gives the optimal FAR/FRR performances of a Hamming distance classifier. The optimal L depends on the Model 1/2/3/4. Furthermore, we observe that at a low bit budget, user-specific models (Model 1/2/3) have advantages over global models (Model 4). Unfortunately, when the bit budget becomes too high, all models become poor. Figures 7(b), 7(d), and 7(f) plot the DET curves of their best performances.

Comparing the performances of DROBA to FBA in Figures 7(a), 7(c), and 7(e), we observe that both DROBA + Model 1/2 outperform FBA for all three datasets. As an example of the FRR/FAR for FRGCt in Figure 8, an explanation might be that DROBA maximizes the detection rate bound of the Hamming distance classifier, leading to averagely lower FRR than FBA. At a low L, DROBA + Model 3 outperforms FBA. However, at high L, it might lose its superiority, as seen in Figures 7(a) and 7(c). This implies that at a high L, the approximate detection rates—computed only from the mean—no longer provide enough useful information for the DROBA principle. We could imagine that at high L, the bit assignment of DROBA + Model 3 tends to become "random," so that it is even not competitive to FBA, which has a uniform bit assignment. DROBA + Model 4, however, does not show great advantages over FBA. Since both DROBA + Model 4 and FBA obtain global bit assignment, we could analyze it for every feature. In Figure 9 we plot their bit assignment at $D = 50$, $L = 50$ and 100, for FRGCt. After PCA/LDA transformation, the features with lower index are generally more discriminative than those with higher index. We observe that DROBA + Model 4 consistently assigns more bits to more discriminative features than less discriminative ones. Contrarily, FBA assigns equal bits to every feature. This explains the better performances of DROBA + Model 4.

Comparing the performances of DROBA to MC and LC in Figures 7(a), 7(c), and 7(e), we observe that at some lengths L, DROBA + Model 1/2/3 outperform MC + Model 1/2 and LC + Model 1/2, except for LC + Model 2 in FRGCt. Likewise, DROBA + Model 4 obtains better performances than MC + Model 1/2 and LC + Model 1 at some lengths L, but worse performances than LC + Model 2, for all three datasets.

In Experiment II, we investigated the verification performance with various numbers of features D, given a bit

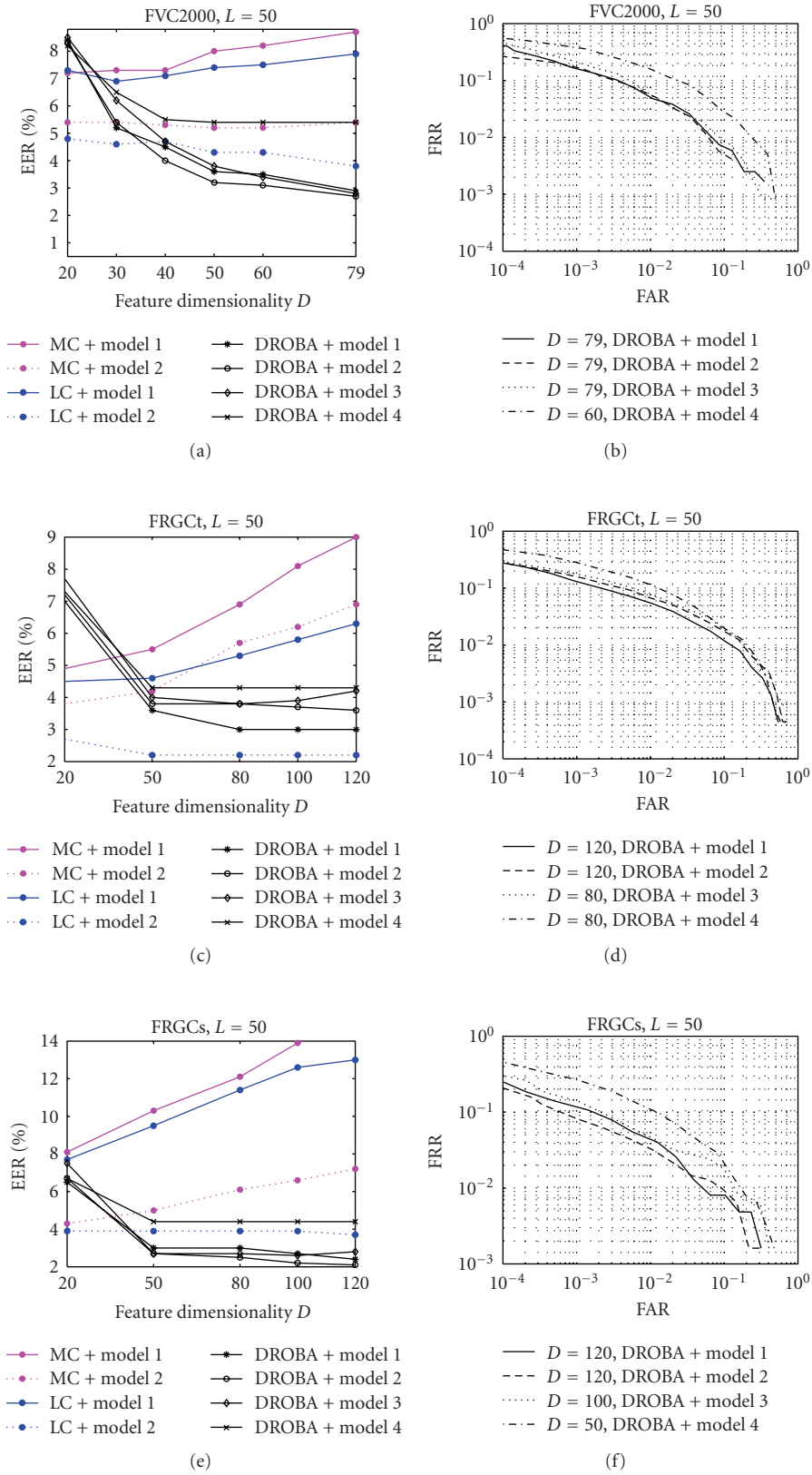

Figure 10: Experiment II: the EER performances of the binary strings generated under DROBA and FBA principles, compared with the real-value feature based Mahalanobis distance classifier (MC) and likelihood-ratio classifier (LC), at $L = 50$, for (a) FVC2000, (c) FRGCt, and (e) FRGCs, with the DET of their best performances in (b), (d), and (f), respectively.

budget $L = 50$. Figures 10(a), 10(c), 10(e), and Table 5 show the corresponding EER performances for FVC2000, FRGCt, and FRGCs. We can imagine that more features give DROBA more freedom to choose the optimal bit assignment, which theoretically should give equal or better detection rate bounded at a given string length L. On the other hand, we know that the PCA/LDA transformation yields less reliable feature components, as the dimensionality D increases. This means that at a high D, if the detection rate model we apply is not robust enough against the feature unreliability, the computed detection rate might not be accurate and consequently mislead the DROBA. Results show that the performances of DROBA + Model 1/2 on the three datasets consistently improve as D increases. This suggests that given a larger number of less reliable features, DROBA + Model 1/2 are still quite effective. Unlike DROBA + Model 1/2, DROBA + Model 3 starts to degrade at very high D, for FRGCt and FRGCs. This suggests that Model 3 is more susceptible to unreliable features. Since it only uses feature mean to predict the detection rate, when the dimensionality is high, the feature mean becomes unreliable, Model 3 no longer computes accurate detection rate. As a global implementation, DROBA + Model 4 gives relatively worse performances than DROBA + Model 1/2/3. However, we observe that when D is larger than a certain value (50 for FVC2000, 50 for FRGCt, and 20 for FRGCt), the bit assignment of DROBA + Model 4 does not change at all, leading to exactly the same performance. This result is consistent with the PCA/LDA transformation, proving that globally the features are becoming less discriminative as D increases, so that DROBA simply discards all the upcoming features. Therefore, by sacrificing the user specificity, DROBA + Model 4 is immune to unreliable features. Figures 10(b), 10(d), and 10(f) plot the DET curves of their best performances.

Comparing the performances of DROBA to MC and LC in Figures 10(a), 10(c), and 10(e), we observe that for all three data sets, DROBA + Model 1/2/3 easily outperform MC + Model 1/2 and LC + Model 1 as D increases. Similar results are obtained when comparing DROBA + Model 1/2/3 to LC + Model 2 in the context of FVC2000 and FRGCs, whereas for FRGCt, DROBA + Model 1/2/3 do not outperform LC + Model 2. Additionally, DROBA + Model 4 outperforms MC+Model 1 and LC+Model 1, as well as MC + Model 2, except for FVC2000. Unfortunately, for all three datasets, DROBA + Model 4 does not outperform LC + Model 2.

6. Discussion

Since DROBA decides the bit assignment according to the detection rate, determining the underlying genuine user PDF is crucial. However, in practice, it turns out to be difficult, due to the lack of samples. To solve this problem, we proposed three user-specific models: (1) Gaussian density (Model 1), (2) Gaussian density with approximated parameters (Model 2), and (3) heuristic model (Model 3). Experimental results suggest that FVC2000 and FRGCs obtain better performances from Model 2, while FRGCt obtains better performances from Model 1. Generally speaking, the genuine user PDF is associated with the biometric modality, as well as the feature extraction method, thus how to choose the right model (e.g., Gaussian) is important. Furthermore, how to accurately estimate the parameters (e.g., μ, σ) in the model is also a problem to solve. There is no gold standard, and choosing the right model and estimation method is a matter of how accurately it fits the features.

Apart from the user-specific models (Model 1/2/3), we also proposed a global model (Model 4). Our experimental results suggest that in a system with multiple enrollment samples per user, it is preferable to choose user-specific models. Nevertheless, Model 4 still has significant potentials: it is purely empirical and nonparametric, avoiding all problems related with model based estimation; it is robust to unreliable features; it is easily adaptable to all biometric systems.

Essentially, unlike the real-valued classifiers (e.g., MC and LC), which fully depend on or "trust" the feature density model, DROBA only partially depends on such model. Thus we might see quantization under DROBA as a model-oriented compression procedure, where the bit allocation is obtained according to the statistics of the model but the data variation within every quantization interval is ignored, leading to a binary string with compressed information. In fact, in Experiment I, we proved that Hamming distance classifier with binary strings may outperform the MC and LC with real-valued features: the applied density model (e.g., Model 1) is not accurate, so that a compressed binary representation might be less prone to overfitting. The compression can be optimized by carefully tuning the $D - L$ or even the $b_{\max}$ configurations in DROBA.

7. Conclusion

Generating binary strings from real-valued biometric measurements in fact acts as a data compression process. Thus, in biometric applications, we aim to generate binary strings that not only retain the discriminative information but also are robust to intra-class variations, so that the performance of the classification is ensured, while the binary strings can be used in various applications. Basically, there are two factors that influence the performance of the binary string: (1) the quantizer design of every feature component; (2) the principle to compose the binary string from all feature components. In this paper, independent of the quantizer design, we proposed a detection rate optimized bit allocation principle (DROBA), which can be achieved by both a dynamic programming and a greedy search approach. Consequently DROBA assigns more bits to discriminative features and fewer bits to nondiscriminative features. This process is driven by the statistics derived from the training and enrollment data, based on which we proposed four models. Experiments on the FVC2000 fingerprint and the FRGC face database show promising results.

The DROBA principle has the advantage that it is adaptable to arbitrary biometric modalities, such as fingerprint texture, iris, signature, and face. Additionally, the binary

strings can be used in any kind of binary string-based classifiers, as well as crypto systems. The practical applications of the biometric binary strings are not only limited to the template protection systems but also to systems requiring fast matching or constrained storage capability. Furthermore, combined with various detection rate estimation methods, binary strings generated under DROBA can be a new promising biometric representation as opposed to the real-valued representation.

Appendix

A. Proving Optimal of the Dynamic Programming Approach

The question that has to be answered is whether the dynamic programming approach presented above will lead to the optimal bit assignment. The proof is as follows. Denote the optimal bit allocation over D' features by

$$\left\{\hat{b}_i(l)\right\} = \arg \max_{b_i|\sum b_i=l,\, b_i\in\{0,\ldots,b_{\max}\}} \prod_{i=1}^{D'} \delta_i(b_i), \tag{A.1}$$

and denote the maximum obtained by $\delta_{\max}$. Assume that we have a partitioning of the D' features into two arbitrary sets. The sets are fully characterized by the indices of the features, so we can speak of the index sets as well. Let $\mathcal{M}$ and $\mathcal{N}$ denote the index sets, such that $\mathcal{M} \cap \mathcal{N} = \varnothing$ and $\mathcal{M} \cup \mathcal{N} = \{1, \ldots, D'\}$. Define

$$\begin{aligned} \delta^{\mathcal{M}}(l) &= \max_{b_i|\sum b_i=l,\, b_i\in\{0,\ldots,b_{\max}\}} \prod_{i\in\mathcal{M}} \delta_i(b_i), \\ &\qquad l = 0, \ldots, |\mathcal{M}|\, b_{\max}, \\ \delta^{\mathcal{N}}(l) &= \max_{b_i|\sum b_i=l,\ \ b_i\in\{0,\ldots,b_{\max}\}} \prod_{i\in\mathcal{N}} \delta_i(b_i), \\ &\qquad l = 0, \ldots, |\mathcal{N}|\, b_{\max}, \end{aligned} \tag{A.2}$$

Define

$$\begin{aligned} \hat{l}^{\mathcal{M}} &= \sum_{i\in\mathcal{M}} \hat{b}_i(l), \\ \hat{l}^{\mathcal{N}} &= \sum_{i\in\mathcal{N}} \hat{b}_i(l). \end{aligned} \tag{A.3}$$

Now

$$\begin{aligned} &\max_{l',l''|l'+l''=l,\, l'\in\mathcal{M},\, l''\in\mathcal{N}} \delta^{\mathcal{M}}(l')\delta^{\mathcal{N}}(l'') \\ &\quad \geq \delta^{\mathcal{M}}\left(\hat{l}^{\mathcal{M}}\right)\delta^{\mathcal{N}}\left(\hat{l}^{\mathcal{N}}\right) \\ &\quad \geq \prod_{i\in\mathcal{M}} \delta_i\left(\hat{b}_i(l)\right) \prod_{i\in\mathcal{N}} \delta_i\left(\hat{b}_i(l)\right) \\ &\quad = \prod_{i=1}^{D'} \delta_i\left(\hat{b}_i(l)\right) \\ &\quad = \delta_{\max}. \end{aligned} \tag{A.4}$$

The left-hand side of this inequality is a product of the form

$$\prod_{i=1}^{D'} \delta_i(b_i), \tag{A.5}$$

with b_i constrained by $\sum b_i = l$, $b_i \in \{0, \ldots, b_{\max}\}$. This cannot, by definition, be greater than $\delta_{\max}$. Therefore, it must be identical to $\delta_{\max}$.

Note that the partitioning into index sets $\mathcal{M}$ and $\mathcal{N}$ was arbitrary. If we take $D' = j$, $\mathcal{M} = \{1, \ldots, j-1\}$, and $\mathcal{N} = \{j\}$, then we have proved that the jth recursion step of the above algorithm is optimal.

Acknowledgments

This research is supported by the research program Sentinels (http://www.sentinels.nl). Sentinels is being financed by Technology Foundation STW, the Netherlands Organization for Scientific Research (NWO), and the Dutch Ministry of Economic Affairs.

References

[1] U. Uludag, S. Pankanti, S. Prabhakar, and A. K. Jain, "Biometric cryptosystems: issues and challenges," *Proceedings of the IEEE*, vol. 92, no. 6, pp. 948–960, 2004.

[2] A. Juels and M. Wattenberg, "A fuzzy commitment scheme," in *Proceedings of the 6th ACM Conference on Computer and Communications Security (CCS '99)*, pp. 28–36, Singapore, November 1999.

[3] Y. Dodis, L. Reyzin, and A. Smith, "Fuzzy extractors: how to generate strong keys from biometrics and other noisy data," in *Proceedings of the International Conference on the Theory and Applications of Cryptographic Techniques*, vol. 3027 of *Lecture Notes in Computer Science*, pp. 523–540, Interlaken, Switzerland, May 2004.

[4] I. Buhan, J. Doumen, P. Hartel, and R. N. J. Veldhuis, "Fuzzy extractors for continuous distributions," in *Proceedings of the 2nd ACM Symposium on Information, Computer and Communications Security (ASIACCS '07)*, pp. 353–355, Singapore, March 2007.

[5] E.-C. Chang and S. Roy, "Robust extraction of secret bits from minutiae," in *Proceedings of the 2nd International Conference on Advances in Biometrics (ICB '07)*, pp. 750–759, Seoul, Korea, August 2007.

[6] J.-P. Linnartz and P. Tuyls, "New shielding functions to enhance privacy and prevent misuse of biometrie templates," in *Proceedings of the 4th International Conference on Audio- and Video-Based Biometric Person Authentication (AVBPA '03)*, vol. 2688 of *Lecture Notes in Computer Science*, pp. 393–402, Guildford, UK, June 2003.

[7] P. Tuyls, T. H. M. Akkermans, T. A. M. Kevenaar, G.-J. Schrijen, A. M. Bazen, and R. N. J. Veldhuis, "Practical biometric authentication with template protection," in *Proceedings of the 5th International Conference on Audio- and Video-Based Biometric Person Authentication (AVBPA '05)*, pp. 436–446, Hilton Rye Town, NY, USA, July 2005.

[8] T. A. M. Kevenaar, G. J. Schrijen, M. van der Veen, T. H. M. Akkermans, and F. Zuo, "Face recognition with renewable and privacy preserving binary templates," in *Proceedings of the 4th IEEE Workshop on Automatic Identification Advanced*

Technologies (AUTO ID '05), pp. 21–26, New York, NY, USA, October 2005.

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

[10] R. N. J. Veldhuis, A. Bazen, J. Kauffman, and P. Hartel, "Biometric verification based on grip-pattern recognition," in *Security, Steganography, and Watermarking of Multimedia Contents VI*, vol. 5306 of *Proceedings of SPIE*, pp. 634–641, San Jose, Calif, USA, January 2004.

[11] D. Maio, D. Maltoni, R. Cappelli, J. L. Wayman, and A. K. Jain, "FVC2000: fingerprint verification competition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 3, pp. 402–412, 2002.

[12] P. J. Phillips, P. J. Flynn, T. Scruggs, et al., "Overview of the face recognition grand challenge," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '05)*, vol. 1, pp. 947–954, San Diego, Calif, USA, June 2005.

[13] C. Vielhauer, R. Steinmetz, and A. Mayerhöfer, "Biometric hash based on statistical features of online signatures," in *Proceedings of the 16th International Conference on Pattern Recognition (ICPR '02)*, vol. 1, pp. 123–126, Quebec, Canada, August 2002.

[14] H. Feng and C. C. Wah, "Private key generation from on-line handwritten signatures," *Information Management & Computer Security*, vol. 10, no. 4, pp. 159–164, 2002.

[15] Y.-J. Chang, W. Zhang, and T. Chen, "Biometrics-based cryptographic key generation," in *Proceedings of IEEE International Conference on Multimedia and Expo (ICME '04)*, vol. 3, pp. 2203–2206, Taipei, Taiwan, June 2004.

[16] C. Chen, R. N. J. Veldhuis, T. A. M. Kevenaar, and T. H. M. Akkermans, "Multi-bits biometric string generation based on the likelihood ratio," in *Proceedings of the 1st IEEE Conference on Biometrics: Theory, Applications and Systems (BTAS '07)*, pp. 1–6, Crystal City, Va, USA, September 2007.

[17] A. B. J. Teoh, A. Goh, and D. C. L. Ngo, "Random multispace quantization as an analytic mechanism for biohashing of biometric and random identity inputs," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 12, pp. 1892–1901, 2006.

[18] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*, The MIT Press, London, UK, 2nd edition, 2001.

[19] Y. Shoham and A. Gersho, "Efficient bit allocation for an arbitrary set of quantizers," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 36, no. 9, pp. 1445–1453, 1988.

[20] V. Perlibakas, "Distance measures for PCA-based face recognition," *Pattern Recognition Letters*, vol. 25, no. 6, pp. 711–724, 2004.

[21] A. M. Bazen and R. N. J. Veldhuis, "Likelihood-ratio-based biometric verification," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 14, no. 1, pp. 86–94, 2004.