EURASIP Journal on Advances in Signal Processing
a SpringerOpen Journal

RESEARCH Open Access

An analysis on equal width quantization and linearly separable subcode encoding-based discretization and its performance resemblances

Meng-Hui Lim, Andrew Beng Jin Teoh* and Kar-Ann Toh

Abstract

Biometric discretization extracts a binary string from a set of real-valued features per user. This representative string can be used as a cryptographic key in many security applications upon error correction. Discretization performance should not degrade from the actual continuous features-based classification performance significantly. However, numerous discretization approaches based on ineffective encoding schemes have been put forward. Therefore, the correlation between such discretization and classification has never been made clear. In this article, we aim to bridge the gap between continuous and Hamming domains, and provide a revelation upon how discretization based on equal-width quantization and linearly separable subcode encoding could affect the classification performance in the Hamming domain. We further illustrate how such discretization can be applied in order to obtain a highly resembled classification performance under the general L*p* distance and the inner product metrics. Finally, empirical studies conducted on two benchmark face datasets vindicate our analysis results.

1. Introduction

Explosion of biometric-based cryptographic applications (see e.g. [1-12]) in the recent decade has abruptly augmented the demand of stable binary strings for identity representation. Biometric features extracted by most current feature extractors, however, do not exist in binary form by nature. In the case where binary processing is needed, *biometric discretization* becomes necessary in order to transform such an ordered set of continuous features into a binary string. Note that discretization is referred to as a process of 'binarization' throughout this article. The general block diagram of a biometric discretization-based binary string generator is illustrated in Figure 1.

Biometric discretization can be decomposed into two essential components: biometric quantization and feature encoding. These components are governed by a static or a dynamic bit allocation algorithm, determining whether the quantity of binary bits allocated to every dimension is fixed or optimally different, respectively. Typically, given an ordered set of real-valued feature elements per identity, each single-dimensional feature space is initially quantized into a number of non-overlapping intervals according to a quantization fashion. The quantity of these intervals is determined by the corresponding number of bits assigned by the bit allocation algorithm. Each feature element captured by an interval is then mapped to a short binary string with respect to the label of the corresponding interval. Eventually, the binary output from each dimension is concatenated to form the user's final bit string.

Apart from the above consideration, information about the constructed feature space for each dimension is stored in the form of helper data to enable reproduction of the same binary string for the same user. However, it is required that such helper data, upon compromise, should neither leak any helpful information about the output binary string, nor that of the biometric feature itself.

In general, there are three aspects that can be used in assessing a biometric discretization scheme:

(1) Performance: Upon extraction of distinctive features, it is important for a discretization scheme to preserve the significance of real-valued feature elements in the Hamming domain in order to maintain

* Correspondence: bjteoh@yonsei.ac.kr
School of Electrical and Electronic Engineering, College of Engineering, Yonsei University, Seoul, South Korea

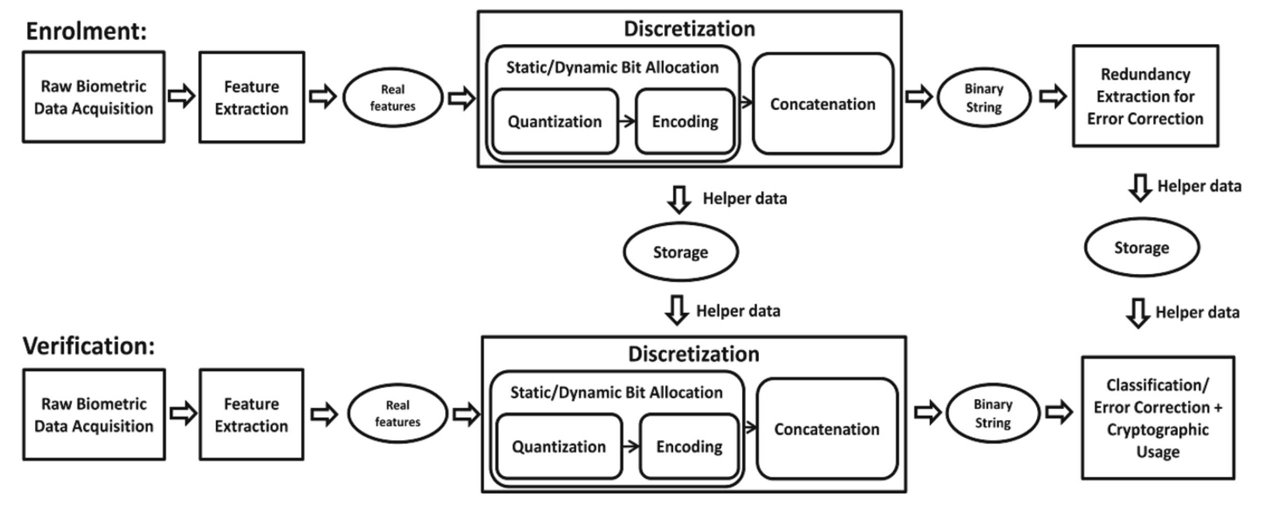


Figure 1 A biometric discretization-based binary string generator.

the actual classification performance. A better scheme usually incorporates a feature selection or bit allocation process to ensure only reliable feature components are extracted or highly weighted for obtaining an improved performance.

(2) Security: Helper data upon revelation must not expose any crucial information which may be of assistance to the adversary in obtaining a false accept. Therefore, the binary string of the user should contain adequate entropy and should be completely uncorrelated to the helper data. Generally, entropy is a measure that quantifies the expected value of information contained in a binary string. In the context of biometric discretization, the entropy of a binary string is referred to as the sum of entropy of all single-dimensional binary outputs. With the probability p_i of every binary output $i \in \{1, \ldots S\}$ in a dimension, the entropy can be calculated as $l = -\sum_{i=1}^{s} p_i \log_2 p_i$. As such, the probability p_i will be reduced when the number of outputs S is increased, signifying higher entropy and security against adversarial brute force attack.

(3) Privacy: A high level of protection needs to be exerted against the adversary who could be interested in all user-specific information other than the verification decision of the system. Apart from the biometric data applicable for discretization, it is important that unnecessary yet sensitive information such as ethnic origin, gender and medical condition should also be protected. Since biometric data is inextricably linked to the user, it can never be reissued or replaced once compromised. Therefore, helper data must be uncorrelated to such information in order to defeat any adversary's privacy violation attempt upon revealing it.

1.1 Related works

Biometric discretization in the literature can generally be divided into two broad categories: supervised and unsupervised discretization (discretization that makes use of class labels of the samples and discretization that does not, respectively).

Unsupervised discretization can be sub-categorized into threshold-based discretization [7-9,11]; equal-width quantization-based discretization [12,13]; and equal-probable quantization-based discretization [5,10,14-16]. For threshold-based discretization, each single-dimensional feature space is segmented into two intervals based on a prefixed threshold. Each interval is labeled with a single bit '0' or '1'. A feature element that falls into an interval will be mapped to the corresponding 1-bit output label. Examples of threshold-based discretization schemes include Monrose et al.'s [7,8], Teoh et al.'s [9] and Verbitsky et al.'s [11] scheme. However, determining the best threshold could be a hurdle in achieving optimal performance. On top of that, this discretization scheme is only able to produce a 1-bit output per feature dimension. This could practically be insufficient in meeting the current entropy requirement (indicating the level of toughness against brute force attacks).

On the other hand, the unsupervised equal width quantization-based discretization [12,13] partitions each single-dimensional feature space into a number of non-overlapping equal-width intervals during quantization in accordance with the quantity of bits required to be extracted from each dimension. These intervals are labeled with binary reflected gray code (BRGC) [17] for encoding, where both of which require the number of constructed intervals to be of a power of 2 in order to avoid loss of entropy. Based on the equal-width quantization and the BRGC encoding, Teoh et al. [13] have designed a user-specific equal-width quantization-based dynamic bit allocation algorithm that assigns different number of bits to each dimension based on an intra-class variation measure. Equal width quantization does not incur privacy issue. However, it could not offer maximum entropy since the probability of every quantization output in a dimension is rarely equal. Moreover, the width of quantization intervals can be easily affected by outliers.

The last subcategory of unsupervised biometric discretization, known as equal-probable quantization-based discretization [14] segments each single-dimensional feature space into multiple non-overlapping equal-probable intervals, whereby every interval is constructed to encapsulate an equal portion of background probability mass during quantization. As a result, the constructed intervals are of different widths if the background distribution is not uniform. BRGC is used for encoding. Subsequently, two efficient dynamic bit allocation schemes have further been proposed by Chen et al. in [15] and [16] based on equal-probable quantization and BRGC encoding where the detection rate (genuine acceptance rate) [15] as well as area under FRR curve [16] is used as the evaluation measure for bit allocation.

Tuyls et al. [10] and Kevenaar et al. [5] have used a similar equal-probable discretization technique but the bit allocation is limited to at most one bit per dimension. However, a feature selection technique is incorporated in order to identify reliable components based on the training bit statistics [10] or a reliability function [5] so that unreliable dimensions can be eliminated from the overall bit extraction and the discretization performance can eventually be improved. Equal probable quantization offers maximum entropy. However, information regarding the background pdf of every dimension needs to be stored so that exact intervals can be constructed during verification. This may pose a privacy threat [18] to the users.

On the other hand, supervised discretization [1,3,14,19] potentially improves classification performance by exploiting the genuine user's feature distribution or the user-specific dependencies to extract segmentations which are useful for classification. In Chang et al.'s [1] and Hao-Chan's scheme [3], single-dimensional interval defined by $[\mu_j - k\sigma_j, \mu_j + k\sigma_j]$ (also known as the *genuine interval*) is first tailored for the Gaussian user pdf (with mean μ_j and standard deviation σ_j) of the genuine user with a free parameter k. The remaining intervals of the same width are then constructed outwards from the genuine interval. Finally, the boundary intervals are formed by the leftover widths. In fact, the number of bits extractable from each dimension relies on the relative number of formable intervals in that dimension and is controllable by k. This scheme uses direct binary representation (DBR) for encoding. Chen et al. proposed a similar discretization scheme [14] except that BRGC encoding is adopted; the genuine interval is determined by the likelihood ratio pdf; and the remaining intervals are constructed equal-probably. Kumar and Zhang [19] employed an entropy-based quantizer to reduce class impurity/entropy in the intervals through recursively splitting every interval until a stopping criterion is met. The final intervals will be resulted in such a way that majority samples enclosed within each interval would belong to a specific identity.

Despite being able to achieve a better classification performance than the unsupervised approaches, a critical problem with these supervised discretization schemes is the potential exposure of the genuine measurements or the genuine user pdf, since the constructed intervals serve as a clue at which the user pdf or measurements could be located to the adversary. As a result, the number of possible locations of user pdf/genuine measurements might be reduced to the amount of quantization intervals in that dimension, thus potentially facilitating malicious privacy violation attempt.

1.2 Motivations and contributions

Past research attention was mostly devoted to proposing discretization schemes with new quantization techniques without realizing the effect of encoding towards the discretization performance. This can be seen from the recent revelation of inappropriateness of DBR and BRGC for feature encoding in classification [20], although they were the most commonly seen encoding schemes for multi-bits discretization in the literature [1,3,12-16]. For this reason, the performance of multi-bits discretization schemes remain to be a mystery when it comes to linking the classification performance in the Hamming domain (discretization performance) with the relative performance in the continuous domain (classification performance of continuous features). To date, no explicit study has been conducted to resolve such an ambiguity.

A common goal of discretization is to convert real-valued features into a binary string which at least preserve the actual classification performance without significantly compromising the security and privacy aspects. To achieve this, it is important that appropriate quantization and encoding schemes have to be adopted. A new encoding scheme known as linearly separable subcode (LSSC) has lately been proposed [20]. With this, features can be encoded much more efficiently with LSSC than with DBR or BRGC. Since combining it with an elegant quantization scheme would produce satisfactory classification results in the Hamming domain, we adopt the unsupervised equal-width quantization scheme in our analysis due to its simplicity and its less susceptibility against privacy attacks. However, a lower entropy could be achieved when the class distribution is not uniform (with respect to the equal-probable quantization approach). This shortage can simply be tackled by utilizing a larger number of feature dimensions or by allocating a larger quantity of bits to each dimension to compensate such entropy loss.

It is the objective of this article to extend the work of [20] to justify and analyze the deterministic discrete-to-binary mapping behavior of LSSC encoding; as well as the *approximate* continuous-to-discrete mapping behavior of equal-width quantization when quantization intervals in each dimension are substantial. We reveal the essential correspondence of distance between the Hamming domain and the rescaled L1 domain for an equal-width quantization and LSSC encoding-based (EW + LSSC) discretization. We further generalize this fundamental correspondence to Lp distance metrics and inner product-based classifiers to obtain desired performance resemblances. These important resemblances in fact open up possibility of applying powerful classifier in the Hamming domain such as binary support vector machine (SVM) without having to suffer from a poorer discretization performance with reference to the actual classification performance.

Empirically, we justify the superiority of LSSC over DBR and BRGC and the aforementioned performance resemblances in the Hamming domain by adopting face biometric as our subject of study. Note that such experiments could also be conducted using other biometric modalities, as long as the relative biometric features can be represented orderly in the form of a feature vector.

The organization of this paper is described as follows. In the next section, equal-width quantization and LSSC encoding are described as a continuous-to-discrete mapping and a discrete-to-binary mapping, respectively, and both mapping functions are derived. These mappings are then combined to reveal the performance resemblance of EW + LSSC discretization to that of the rescaled L1 distance-based classification. In Section 3, proper methods to extend basic performance resemblance of EW + LSSC discretization to that of different metrics and classifiers are described. In section 4, *approximate* performance of EW + LSSC discretization with respect to L1 distance-based classification performance is experimentally justified. Results showing the resemblances of altered EW + LSSC discretization to the performance of several different distance metrics/classifier are presented. Finally, several insightful concluding remarks are drawn in Section 5.

2. Biometric discretization

For binary extraction, biometric discretization can be described as a two-stage mapping process: Each segmented feature space is first mapped to the respective index of a quantization interval; subsequently, the index of each interval is mapped to a unique n-bit codeword in a Hamming space. The overall mapping process can be mathematically described by

$$b_{i^d}^d = g(i^d) = g(f(v^d)) \tag{1}$$

where v^d denotes a continuous feature, i^d denotes a discrete index of the interval, $b_{i^d}^d$ denotes a short binary string associated to i^d, $f:\mathbb{R} \to \mathbb{Z}$ denotes a continuous-to-discrete map and $g:\mathbb{Z} \to \{0, 1\}^n$ denotes a discrete-to-binary map. Note that a superscript d is used for specifying the dimension to which a variable belongs and it is by no means of being an integer power. We shall define both these functions in the following subsections.

2.1 Continuous-to-discrete mapping $f(\cdot)$

A continuous-to-discrete mapping$f(\cdot)$ is achieved through applying quantization to a continuous feature space. Recall that an equal-width quantization divides a one-dimensional feature space evenly in forming the quantization intervals and subsequently maps each interval-captured background probability density function (pdf) to a discrete index. Hence, the probability mass $p_{i^d}^d$ associated with each index i^d precisely represents the probability density captured by the interval with the same index. This equality can be described by

$$p_{i^d}^d = \int_{int_{i^d(\min)}^d}^{int_{i^d(\max)}^d} p_{bg}^d(v)\mathrm{d}v \quad \text{for} \quad i^d \in \{0, 1, \ldots, S^d - 1\} \tag{2}$$

where $p_{bg}^d(\cdot)$ denotes the d-th dimensional background pdf, $int_{i^d(\max)}^d$ and $int_{i^d(\min)}^d$ denote the upper and lower boundary of interval with index i^d in the d-th dimension, and S^d denotes the number of constructed intervals in the d-th dimension. Conspicuously, the resultant background pmf is an approximation of the original pdf upon the mapping.

Suppose that a feature element captured by an interval $\text{int}_{i^d}^d$ with an index i^d is going to be mapped to a fixed point within such an interval. Let $c_{i^d}^d$ be the fixed point in $\text{int}_{i^d}^d$ to which every feature element $v_{i^d j^d}^d$ that falls within the interval has to be mapped, where $i^d \in \{0,1, \ldots S^d - 1\}$ denotes the interval index and $j_d \in \{1,2 \ldots\}$ denotes the feature element index. The distance of $v_{i^d j^d}^d$

from $c^d_{i^d}$ is

$$\varepsilon^d_{i^d,j^d} = \left|v^d_{i^d,j^d} - C^d_{i^d}\right| \leq \max\{int^d_{i^d(\max)} - c^d_{i^d}, c^d_{i^d} - int^d_{i^d(\min)}\} = \varepsilon^d_{i^d,j^d(\max)}. \quad (3)$$

Suppose now we are to match each index i^d of the S^d intervals with the corresponding $c^d_{i^d}$ through some scaling $\hat{s}$ and translation $\hat{t}$:

$$c^d_{i^d} = \hat{s}^d(i^d + \hat{t}^d) \quad \text{for } i^d = \{0, S^d - 1\}. \quad (4)$$

To make $\hat{s}^d$ and $\hat{t}^d$ globally derivable for all intervals, it is necessary to keep distance between $c^d_{i^d}$ and $c^d_{i^d+1}$ constant for every $i^d \in$ {0, S^d - 2}. In order to preserve such a distance between any two different intervals, $c^d_{i^d}$ in every interval should, therefore, take identical distance from its corresponding $int^d_{i^d(\min)}$. Without loss of generality, we let $c^d_{i^d}$ be the central point of $\text{int}\,^d_{i^d}$, such that

$$c^d_{i^d} = \frac{int^d_{i^d(\max)} - int^d_{i^d(\min)}}{2} \quad \text{for } i^d = \{0, S^d - 1\}. \quad (5)$$

With this, the upper bound of distance of $v^d_{i^d j^d}$ from $c^d_{i^d}$ upon mapping in (3) becomes

$$\varepsilon^d_{i^d,j^d} \leq int^d_{i^d(\max)} - c^d_{i^d} = c^d_{i^d} - int^d_{i^d(\min)}\} = \varepsilon^d_{i^d,j^d(\max)}. \quad (6)$$

To obtain the parameters $\hat{s}^d$ and $\hat{t}^d$, we normalize both feature and index spaces to (0, 1) and shift every normalized index i^d by $\frac{1}{2S^d}$ to the right to fit the respective $c^d_{i^d}$, such that

$$\frac{c^d_{i^d}}{int^d_{S^d-1(\max)} - int^d_{0(\min)}} = \frac{2i^d + 1}{2S^d}. \quad (7)$$

Through some algebraic manipulation, we have

$$c^d_{i^d} = \frac{int^d_{S^d-1(\max)} - int^d_{0(\min)}}{S^d}(i^d + 0.5). \quad (8)$$

Thus, $\hat{s}^d = \frac{\text{int}\,^d_{S^d-1(\max)} - \text{int}\,^d_{0(\min)}}{S^d}$ and $\hat{t}^d = 0.5$.

Combining results from (3), (4) and (8), the continuous-to-discrete mapping function$f(\cdot)$ can be written as

$$i^d = f(v^d_{i^d,j^d}) = \begin{cases} \frac{1}{\hat{s}^d}(v^d_{i^d,j^d} - \hat{s}^d\hat{t} - \varepsilon^d_{i^d,j^d}) & \text{for} \quad v^d_{i^d,j^d} \geq c^d_{i^d} \\ \frac{1}{\hat{s}^d}(\hat{s}^d\hat{t} - v^d_{i^d,j^d} - \varepsilon^d_{i^d,j^d}) & \text{for} \quad v^d_{i^d,j^d} \geq c^d_{i^d} \end{cases} \quad (9)$$

Suppose we are to compute a L1 distance between two arbitrary points $v^d_{i^d_1,j^d_1}$ and $v^d_{i^d_2,j^d_2}$ for all $i^d_1, i^d_2 \in [0, S^d - 1], j^d_1, j^d_2 \in \{1, 2...\}$ in the d-th dimensional continuous feature space, and the relative distance between the corresponding mapped elements in the d-th dimensional discrete index space, then it is easy to find that the deviation between these two distances can be bounded below:

$$0 \leq \left|\left|v^d_{i^d_2,j^d_2} - v^d_{i^d_1,j^d_1}\right| - \left|c^d_{i^d_2,j^d_2} - c^d_{i^d_1,j^d_1}\right|\right| \leq 2\varepsilon^d_{i^d\ j^d(\max)}. \quad (10)$$

From (4), this inequality becomes

$$0 \leq \left|\left|v^d_{i^d_2,j^d_2} - v^d_{i^d_1,j^d_1}\right| - \hat{s}\left(\left|i^d_2 - i^d_1\right|\right)\right| \leq 2\varepsilon^d_{i^d\ j^d(\max)}. \quad (11)$$

Note that the upper bound of such distance deviation is equivalent to the width of an interval in (6), such that

$$2\varepsilon^d_{i^d,j^d(\max)} = int^d_{i^d(\max)} - int^d_{i^d(\min)}. \quad (12)$$

Therefore, it is clear that an increase or reduction in the width of each equal-width interval could significantly affect the upper bound of such deviation. For instance, when the number of intervals constructed over a feature space is increased/reduced by a factor of β (i.e. $S^d \to \beta S^d$ or $S^d \to \frac{1}{\beta}S^d$), the width of each equal-width interval will be reduced/increased by the same factor. Hence, the resultant upper bound for the distance deviation becomes $\frac{2\varepsilon^d_{i^d,j^d(\max)}}{\beta}$ and $2\beta\varepsilon^d_{i^d,j^d(\max)}$, respectively.

Finally, when static bit allocation is adopted where an equal number of equal-width intervals is constructed in all D feature dimensions, the total distance deviation incurred by the continuous-to-discrete mapping can be upper bounded by $2D\varepsilon^d_{i^d,j^d(\max)}$.

2.2 Discrete-to-binary mapping *g*(·)

The discrete-to-binary mapping can be defined in a more direct manner compared to the previous mapping. Suppose that in the d-th dimension, we have S^d discrete elements to be mapped from the index space. We therefore require the same amount of elements in the Hamming space to be mapped to. In fact, these elements in the Hamming space (also known as the codewords) may have different orders and indices depending on the encoding scheme being employed. With this, the direct-to-binary mapping can, therefore, be specified by

$$b^d_{i^d} = g(i^d) = \mathbb{C}(i^d) \quad \text{for } i^d \in [0, S^d - 1] \quad (13)$$

where $\mathbb{C}(i^d)$ denotes a codeword with index i^d from an encoding scheme $\mathbb{C}$. We shall look into the available options of $\mathbb{C}$ and their individual effect on the discrete-to-binary mapping in the following subsections.

2.2.1 Encoding schemes

(a) Direct binary representation (DBR)

In DBR, decimal indices are directly converted into their binary equivalent. Depending on the required size S of a code, the length of DBR is selected to be $n_{\text{DBR}} = [\log_2 S]$. A collection of DBRs in fulfilling S = 4, 8 and 16 are illustrated in Table 1.

(b) Binary reflected gray code (BRGC) [17]

BRGC is a special code that restricts the Hamming distance between every consecutive pair of codewords to unity. Similarly as DBR, each decimal index is uniquely mapped to one out of S number of n_{BRGC}-bit codewords, where $n_{\text{BRGC}} = [\log_2 S]$. If $L_{n\text{BRGC}}$ denotes the listing of n_{BRGC}-bit binary strings, then n_{BRGC}-bit BRGC can be defined recursively as follows:

$$\begin{aligned} L_1 &= 0.1 \\ L_{n_{\text{BRGC}}} &= 0L_{n_{\text{BRGC}}-1}, 1\overline{L_{n_{\text{BRGC}}-1}} \text{ for } n_{\text{BRGC}} > 1 \end{aligned} \tag{14}$$

Here, bL denotes the list constructed from L by adding bit b in front of every element of L, and $\bar{L}$ denotes the complement of list L. In Table 2, instances of BRGCs in meeting different values of S are shown.

(c) Linearly separable subcode (LSSC) [20]

Out of $2^{n_{\text{LSSC}}}$ codewords in total for any positive integer n_{LSSC}, LSSC contains (n_{LSSC} + 1) number of n_{LSSC}-bit codewords, where every adjacent pair of codewords differs by a single bit and every non-adjacent pair of codewords differs by q bits, with q denoting the corresponding index difference. Beginning with an initial codeword, say the all-zero codeword, the next n_{LSSC} number of codewords can simply be constructed by complementing a bit from the lowest order (rightmost) bit position to the highest order (leftmost) bit position one at a time. The resultant n_{LSSC} -bit LSSCs in fulfilling S = 4, 8 and 16 are shown in Table 3.

Table 2 A collection of n_{BRGC}-bit BRGCs for S = 4, 8 and 16 where [τ] denotes the codeword index

n_{BRGC} = 2		n_{BRGC} = 3		n_{BRGC} = 4			
S = 4		S = 8		S = 16			
[0]	00	[0]	000	[0]	0000	[8]	1100
[1]	01	[1]	001	[1]	0001	[9]	1101
[2]	11	[2]	011	[2]	0011	[10]	1111
[3]	10	[3]	010	[3]	0010	[11]	1110
		[4]	110	[4]	0110	[12]	1010
		[5]	111	[5]	0111	[13]	1011
		[6]	101	[6]	0101	[14]	1001
		[7]	100	[7]	0100	[15]	1000

2.2.2 Mappings and correspondences

On Hamming space where Hamming distance is crucial, a one-to-one correspondence between each binary codeword and the corresponding Hamming distance incurred with respect to any reference codeword is essentially desired. We can observe clearly from Figure 2 that even though the widely used DBR and BRGC have each of their codewords associated with a unique index, most mapped elements eventually overlap each other as far as Hamming distance is concerned. In other words, although distance deviation in prior continuous-to-discrete mapping is minimal, the deviation effect led by such an overlapping discrete-to-binary mapping could be tremendous, causing the continuous feature elements originated from multiple different non-adjacent intervals to be mapped to a common Hamming distance away from a specific codeword.

Taking DBR as an instance in Figure 2a, feature elements associated with intervals 1, 2 and 4 are mapped to codewords ‘001’, ‘010’ and ‘100’, respectively, which are all 1 Hamming distance away from ‘000’ (interval 0). This implies that if there is a scenario where we have a genuine template feature captured by interval 0, a genuine query feature by interval 1, two imposters’ query features by intervals 2 and 4, all query features will be mapped to 1 Hamming distance away from the template

Table 1 A collection of n_{DBR}-bit DBRs for S = 4, 8 and 16 where [τ] denotes the codeword index

n_{DBR} = 2		n_{DBR} = 3		n_{DBR} = 4			
S = 4		S = 8		S = 16			
[0]	00	[0]	000	[0]	0000	[8]	1000
[1]	01	[1]	001	[1]	0001	[9]	1001
[2]	10	[2]	010	[2]	0010	[10]	1010
[3]	11	[3]	011	[3]	0011	[11]	1011
		[4]	100	[4]	0100	[12]	1100
		[5]	101	[5]	0101	[13]	1101
		[6]	110	[6]	0110	[14]	1110
		[7]	111	[7]	0111	[15]	1111

Table 3 A collection of n_{LSSC}-bit LSSCs for S = 4,8 and 16 where [τ] denotes the codeword index

n_{LSSC} = 3		n_{LSSC} = 7		n_{LSSC} = 15			
S = 4		S = 8		S = 16			
[0]	000	[0]	0000000	[0]	000000000000000	[8]	000000011111111
[1]	001	[1]	0000001	[1]	000000000000001	[9]	000000111111111
[2]	011	[2]	0000011	[2]	000000000000011	[10]	000001111111111
[3]	111	[3]	0000111	[3]	000000000000111	[11]	000011111111111
		[4]	0001111	[4]	000000000001111	[12]	000111111111111
		[5]	0011111	[5]	000000000011111	[13]	001111111111111
		[6]	0111111	[6]	000000000111111	[14]	011111111111111
		[7]	1111111	[7]	000000001111111	[15]	111111111111111

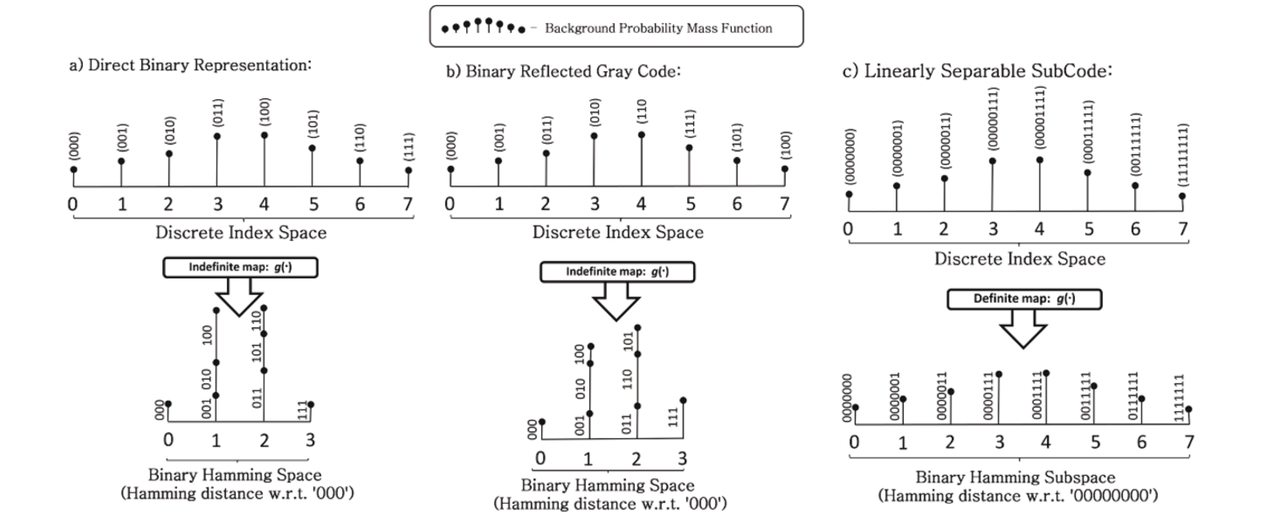


Figure 2 Discrete-to-binary mapping by different encoding techniques: (a) direct binary representation, **(b)** binary reflected gray code and **(c)** linearly separable subcode.

and could not be differentiated. Likewise, the same problem occurs when BRGC is employed, as illustrated in Figure 2b. Therefore, these imprecise mappings caused by DBR and BRGC greatly undermine the actual discriminability of the feature elements and could probably be detrimental to the overall recognition performance.

In contrast, LSSC does not suffer from such a drawback. As shown in Figure 2c, LSSC links each of its codewords to a unique Hamming distance away from any reference codeword in a decent manner. More precisely, a definite mapping behaviour can be obtained when each index is mapped to a LSSC codeword. The probability mass distribution in the discrete space is completely preserved upon the discrete-to-binary mapping and thus, a precise mapping from the L1 distance to the Hamming distance can be expected, such that given two indices $i_1^d = f\left(v_{i_1^d,j_1^d}^d\right)$, $i_2^d = f\left(v_{i_2^d,j_2^d}^d\right)$ and their respective LSSC-based binary outputs $\left|i_1^d - i_2^d\right| = H_D\left(b_{i_1^d}^d, b_{i_2^d}^d\right) \quad \forall i_1^d, i_2^d \in [0, S^d - 1]$,

$$\left|i_1^d - i_2^d\right| = H_D\left(b_{i_1^d}^d, b_{i_2^d}^d\right) \quad \forall i_1^d, i_2^d \in [0, S^d - 1] \tag{15}$$

where H_D denotes the Hamming distance operator. The only disadvantage of LSSC is the larger bit length requirement a system may need to afford in meeting a similar number of discretization outputs compared to DBR and BRGC. In the case where a total of S^d intervals need to be constructed for each dimension, LSSC introduces $R^d = S^d - \log_2 S^d - 1$ redundant bits to maintain the optimal one-to-one discrete-to-binary mapping in the d-th dimension. Thus, upon concatenation of outputs from all feature dimensions, the length of LSSC-based final binary string could be significantly larger.

2.3 Combinations of both mappings

Through combining both continuous-to-discrete and discrete-to-binary mappings, the overall mapping can be expressed as

$$b_{i_1^d}^d = g\left(f\left(v_{i_1^d,j_1^d}^d\right)\right) = \begin{cases} \mathbb{C}\left(\frac{1}{\hat{s}^d}\left(v_{i^d,j^d}^d - \hat{s}^d\hat{t} - \varepsilon_{i^d,j^d}^d\right)\right) & \text{for} \quad v_{i^d,j^d}^d \geq c_{i^d}^d \\ \mathbb{C}\left(\frac{1}{\hat{s}^d}\left(\hat{s}^d\hat{t} - v_{i^d,j^d}^d - \varepsilon_{i^d,j^d}^d\right)\right) & \text{for} \quad v_{i^d,j^d}^d < c_{i^d}^d \end{cases} \tag{16}$$

where $\hat{s}^d = \dfrac{\text{int}\,_{S^d-1\,(\max)}^{d} - \text{int}\,_{0\,(\min)}^{d}}{S^d}$ and $\hat{t}^d = 0.5\cdot$

This equation can typically be used to derive the codeword $b_{i^d}^d$ based on the continuous feature value $v_{i^d j^d}^d$.

In view of different encoding options, three discretization configurations can be deduced. They are:

- Equal Width + Direct Binary Representation (EW + DBR)
- Equal Width + Binary Reflected Gray Code (EW + BRGC)
- Equal Width + Linearly Separable SubCode (EW + LSSC)

Table 4 gives a glance of the behaviours of both mappings which we have discussed so far. Among them, a much poorer performance by EW + DBR and EW + BRGC can be anticipated due to intrinsic indefinite mapping deficiency. On contrary, only the combination

Table 4 A summary of mapping behavior of *f*(·) and *g*(·)

Continuous-to-discrete *f*(·)		**Discrete-to-binary *g*(·)**	
Quantization scheme	Mapping behaviour	Encoding scheme	Mapping behaviour
Equal-width (EW)	Approximate	DBR	Indefinite
	$\left\| v^d_{i^d_2, j^d_2} - v^d_{i^d_1, j^d_1} \right\| \cong \hat{s}\left(\left\| i^d_2 - i^d_1 \right\| \right)$		$\left(\left\| i^d_2 - i^d_1 \right\| = H_D\left(b^d_{i^d_2}, b^d_{i^d_1} \right) \right)$
		BRGC	Indefinite
			$\left(\left\| i^d_2 - i^d_1 \right\| = H_D\left(b^d_{i^d_2}, b^d_{i^d_1} \right) \right)$
		LSSC	Definite
			$\left(\left\| i^d_2 - i^d_1 \right\| = H_D\left(b^d_{i^d_2}, b^d_{i^d_1} \right) \right)$

of EW + LSSC could lead to approximate and definite discretization results. Since for LSSC, $H_D\left(b^d_{i^d_2}, b^d_{i^d_1} \right) = \left| i^d_2 - i^d_1 \right|$ and $S^d = n^d_{\text{LSSC}} + 1$, integrating these LSSC properties with (3) and (4) yield

$$\begin{aligned}
& H_D\left(b^d_{i^d_2}, b^d_{i^d_1} \right) = \left| i^d_2 - i^d_1 \right| \\
& = \frac{1}{\hat{s}^d} \left| c^d_{i^d_2} - c^d_{i^d_1} \right| \\
& = \frac{1}{\hat{s}^d} \left| v^d_{i^d_2, j^d_2} - \varepsilon^d_{i^d_2, j^d_2} - v^d_{i^d_1, j^d_1} + \varepsilon^d_{i^d_1, j^d_1} \right| \\
& \cong \frac{1}{\hat{s}^d} \left| v^d_{i^d_2, j^d_2} - v^d_{i^d_1, j^d_1} \right| \\
& \cong \frac{(n^d_{\text{LSSC}} + 1)}{\text{int}^{\,d}_{S^d-1\,(\max)} - \text{int}^{\,d}_{0\,(\min)}} \left| v^d_{i^d_2, j^d_2} - v^d_{i^d_1, j^d_1} \right| .
\end{aligned} \tag{17}$$

Here the RHS of (17) corresponds to a rescaled L1 distance.

By concatenating distances of all D individual dimensions, the overall discretization performance of EW + LSSC could, therefore, very likely to resemble the relative performance of the rescaled L1 distance-based classification:

$$\sum_{d=1}^{D} H_D\left(b^d_{i^d_2}, b^d_{i^d_1} \right) \cong \sum_{d=1}^{D} \frac{n^d_{\text{LSSC}} + 1}{\text{int}^{\,d}_{S^d-1\,(\max)} - \text{int}^{\,d}_{0\,(\min)}} \left| v^d_{i^d_2, j^d_2} - v^d_{i^d_1, j^d_1} \right| . \tag{18}$$

Hence, matching plain bitstrings in a biometric verification system guarantees a rescaled L1 distance-based classification performance when $S^d = n^d_{\text{LSSC}} + 1$ is adequately large. However, for cryptographic key generation applications where a bitstring is derived directly from the helper data of each user for further cryptographic usage, (18) then implies relation between the bit discrepancy of an identity's bitstring with reference to the template bitstring and the L1 distance of their continuous counterparts in each dimension.

3. Performance resemblances

When binary matching is performed, the basic resemblance in (18) can further be exploited to obtain resemblance with the other distance metric-based and machine learning-based classification performance. The key idea for such extension lies in how to flexibly alter the matching function or to represent each continuous feature element individually with its binary approximation in obtaining near-equivalent classification behaviour in the continuous domain. As such, rather than just confining binary matching method to pure Hamming distance calculation, these extensions significantly broaden the practicality of performing binary matching and enable a strong performance resemblance of a powerful classifier such as a multilayer perceptron (MLP) [21] or a SVM [22] when the bits allocation to each dimension is substantially large. In this section, 'ζ_φ' denotes the matching score of the 'φ' dissimilarity/similarity measure.

3.1 L*p* Distance metrics

In the case where a Lp distance metric classification performance is desired, the resemblance equation in (18) can easily be modified and applied to obtain an approximate performance in the Hamming domain by

$$\begin{aligned}
\zeta_{\text{L}p} & = \sqrt[p]{\sum_{d=1}^{D} \left| v^d_{i^d_2, j^d_2} - v^d_{i^d_1, j^d_1} \right|^p} \\
& \cong \sqrt[p]{\sum_{d=1}^{D} \left(\hat{s}^d \left| i^d_2 - i^d_1 \right| \right)^p} \\
& \cong \sqrt[p]{\sum_{d=1}^{D} \left(\left(\frac{int^d_{S^d-1\,(\max)} - int^d_{0\,(\min)}}{n^d_{\text{LSSC}} + 1} \right) \left(H_D\left(b^d_{i^d_2}, b^d_{i^d_1} \right) \right) \right)^p}
\end{aligned} \tag{19}$$

provided that the number of bits allocated to each dimension are substantially large, or equivalently, the quantization intervals in each dimension are of great number. As long as $\left|v^d_{i_2^d,j_2^d} - v^d_{i_1^d,j_1^d}\right|$ can be linked to the desired distance computation, (14) can then be modified and applied directly. According to (11), the total difference in distance of (19) is upper bounded by $\sqrt[p]{\sum_{d=1}^{D}\left(2\varepsilon^d_{i_2^d,j_2^d(\max)}\right)^p}$.

Likewise, to achieve a resembled performance of k-NN classifier [23] and RBF network [24] that use Euclidean distance (L2) as the distance metric, the RHS of (19) can simply be amended and subsequently adopted for binary matching by setting $p = 2$.

3.2 Inner product

For the inner product similarity measure which cannot be directly associated with $\left|v^d_{i_2,j_2} - v^d_{i_1,j_1}\right|$, the simplest way to obtain the approximate performance resemblance is to transform each continuous feature value into its binary approximate individually and substitute it into the actual formula. By exploiting results from (3), (8) and (15), we have

$$\begin{aligned} v^d_{i^d,j^d} &\cong \left(\frac{\text{int}^{\,d}_{S^d-1\,(\max)} - \text{int}^{\,d}_{0\,(\min)}}{n^d_{\text{LSSC}}+1}\right)(i^d + 0.5) \\ &\cong \left(\frac{\text{int}^{\,d}_{S^d-1\,(\max)} - \text{int}^{\,d}_{0\,(\min)}}{n^d_{\text{LSSC}}+1}\right)(\left|i^d - 0\right| + 0.5) \\ &\cong \left(\frac{\text{int}^{\,d}_{S^d-1\,(\max)} - \text{int}^{\,d}_{0\,(\min)}}{n^d_{\text{LSSC}}+1}\right)\left(H_D\left(b^d_{i^d}, b^d_0\right) + 0.5\right) \end{aligned} \tag{20}$$

leading to an approximate binary representation of the continuous feature value.

Considering inner product (IP) between two column feature vectors $\mathbf{v_1}$ and $\mathbf{v_2}$ as an instance, we represent every continuous feature element in each feature vector with its binary approximate to obtain an approximately equal similarity measure:

$$\begin{aligned} \zeta_{\text{IP}} &= v_2^T v_1 \\ &= \sum_{d=1}^{D} v^d_{i_2^d,j_2^d} v^d_{i_1^d,j_1^d} \\ &\cong \sum_{d=1}^{D}\left(\frac{\text{int}^{\,d}_{S^d-1\,(\max)} - \text{int}^{\,d}_{0\,(\min)}}{n^d_{\text{LSSC}}+1}\right)^2 \left(i_2^d + 0.5\right)\left(i_1^d + 0.5\right) \\ &\cong \sum_{d=1}^{D}\left(\frac{\text{int}^{\,d}_{S^d-1\,(\max)} - \text{int}^{\,d}_{0\,(\min)}}{n^d_{\text{LSSC}}+1}\right)^2 \left(H_D\left(b^d_{i_2^d}, b^d_0\right) + 0.5\right)\left(H_D\left(b^d_{i_1^d}, b^d_0\right) + 0.5\right). \end{aligned} \tag{21}$$

The total similarity deviation of (21) turns out to be upper bounded by $\sum_{d=1}^{D}\left(\varepsilon^d_{i^d,j^d(\max)}\right)^2$.

For another instance, the similarity measure adopted by SVM [22] in classifying an unknown data point appears likewise to be inner product-based. Let n_s be the number of support vectors, $y_k = \pm 1$ be the class label of the k-th support vector, $\boldsymbol{v_k}$ be the k-th D-dimensional support (column) vector, $\boldsymbol{v}$ be the D-dimensional query (column) vector, $\hat{\lambda}_k$ be the optimized Lagrange multiplier of the k-th support vector and $\hat{w}_o$ be the optimized bias. The performance resemblance of binary SVM to that of the continuous counterpart follows directly from (21) in such a way that

$$\begin{aligned} \zeta_{\text{SVM}} &= \sum_{k=1}^{n_s} y_k\hat{\lambda}_k(v^T v_k) + \hat{w}_o \\ &= \sum_{k=1}^{n_s} y_k\hat{\lambda}_k\left(\sum_{d=1}^{D} v^d_{i_2^d,j_2^d} v^d_{i_1^d,j_1^d}\right) + \hat{w}_o \\ &\cong \sum_{k=1}^{n_s}\sum_{d=1}^{D} y_k\hat{\lambda}_k\left(\frac{\text{int}^{\,d}_{S^d-1\,(\max)} - \text{int}^{\,d}_{0\,(\min)}}{n^d_{\text{LSSC}}+1}\right)^2 \left(H_D\left(b^d_{i_2^d}, b^d_0\right) + 0.5\right)\left(H_D\left(b^d_{i_1^d}, b^d_0\right) + 0.5\right) + \hat{w}_o. \end{aligned} \tag{22}$$

The expected upper bound of the total difference in similarity of (22) is then quantified by $\max_{y_k}\left|\sum_{k=1}^{n_{y_k}}\sum_{d=1}^{D} y_k\left(\varepsilon^d_{i_2^d,j_2^d(\max)}\right)^2\right|$ where $y_k = \pm 1$ and n_{y_k} denotes the number of support vectors with class label y_k.

In fact, the individual element transformation illustrated in (20) can be generalized to any other inner product-based measure and classifier such as Pearson correlation [25] and MLP [21] in order to obtain a resemblance in performance when the matching is carried out in the Hamming domain.

4. Performance evaluation

4.1 Data sets and experiment settings

To evaluate the discretization performance of the three discretization schemes (EW + DBR, EW + BRGC and EW + LSSC) and to justify the performance resemblances by EW + LSSC in particular, our experiments were conducted based on the following two popular face data sets:

AR

The employed data set is a random subset of the AR face data set [26], which contains a total of 684 images corresponding to 114 identities with 6 images per person. The images were taken under controlled illumination conditions with moderate variations in facial expressions. The images were aligned according to standard landmarks, such as eyes, nose and mouth. Each extracted raw feature vector consists of 56 × 46 grey pixel elements. Histogram equalization was applied to these images before they were processed by the feature extractor.

FERET

The employed data set is a random subset of the FERET face dataset, [27] in which the images were collected under a semi-controlled environment. It contains a total of 2400 images with 12 images for each of 200 identities. Proper alignment is applied to the images based on the standard face landmarks. Due to possible strong variation in hair style, only the face region is extracted for recognition by cropping it to the size of 61 × 73 from each raw image. The images were pre-processed with histogram equalization before feature extraction. Note that SVM performance resemblance experiments in Figures 3Ib, IIb and 4Ib, IIb only utilize images from the first 75 identities to reduce the computational complexity of our experiments.

For each identity in both datasets, half of the images are randomly selected for training while the remaining half is used for testing. In order to measure the false acceptance rate (FAR) of the system, each image of every identity is matched against a random image of every other identity within the testing partition (without overlapping selection), while for evaluating the system FRR, each image is matched against every other images of the same identity for every identity within the testing partition. In the following experiments, the equal error rate (EER) (error rate where FAR = FRR) is used to compare the classification and discretization performances, since it is a quick and convenient way to compare the accuracy of such classification and discretization. The lower the EER is, the better the performance is considered to be and vice versa.

4.2 Performance assessment

The conducted experiments can be categorized into two parts. The first part examines the performance superiority of EW + LSSC over the remaining schemes and justifies the fundamental performance resemblance with the rescaled L1 distance-based classification performance in (18). The second part vindicates the applicability of EW + LSSC discretization in obtaining a resembled performance of each different metric and a classifier including L1, L2, L3 distance metric, inner product similarity metric and a SVM classifier, as exhibited in (19) and (21). Note that in this part, features from each dimension have been min-max normalized (by dividing both sides of (19) and (21) by $\left(\text{int}\,^{d}_{S^d-1\,(\max)} - \text{int}\,^{d}_{0\,(\min)}\right)$ before they are classified/discretized.

Both parts of experiments were carried out based on static bit allocation. To ensure consistency of the results, two different dimensionality reduction techniques (principal component analysis (PCA) [28] and Eigenfeature regularization and extraction (ERE) [29]) with two well-known face data sets (AR and FERET) were used. The raw dimensions of AR (2576) and FERET (4453) images were both reduced to D = 64 by PCA and ERE in all parts of experiment.

In general, discretization based on static bit allocation assigns n bits equally to each of the D feature dimensions, thereby yielding a Dn-bit binary string in representing every identity upon concatenating short binary outputs from all individual dimensions. Note that LSSC has a code length different from DBR and BRGC when labelling a specific number of intervals. Thus, it is unfair to compare the performance of EW + LSSC with the remaining schemes through equalizing the bit length of the binary strings generated by different encoding schemes, since the dimensions utilized by LSSC-based discretization will be much lesser than that by DBR-based and BRGC-based discretization at common bit lengths.

A better way to compare these discretization schemes would be in terms of entropy L of the final bit string. By denoting the entropy of the d-th dimension as l^d and the i-th output probability of the d-th dimension as $p^d_{i^d}$, we have

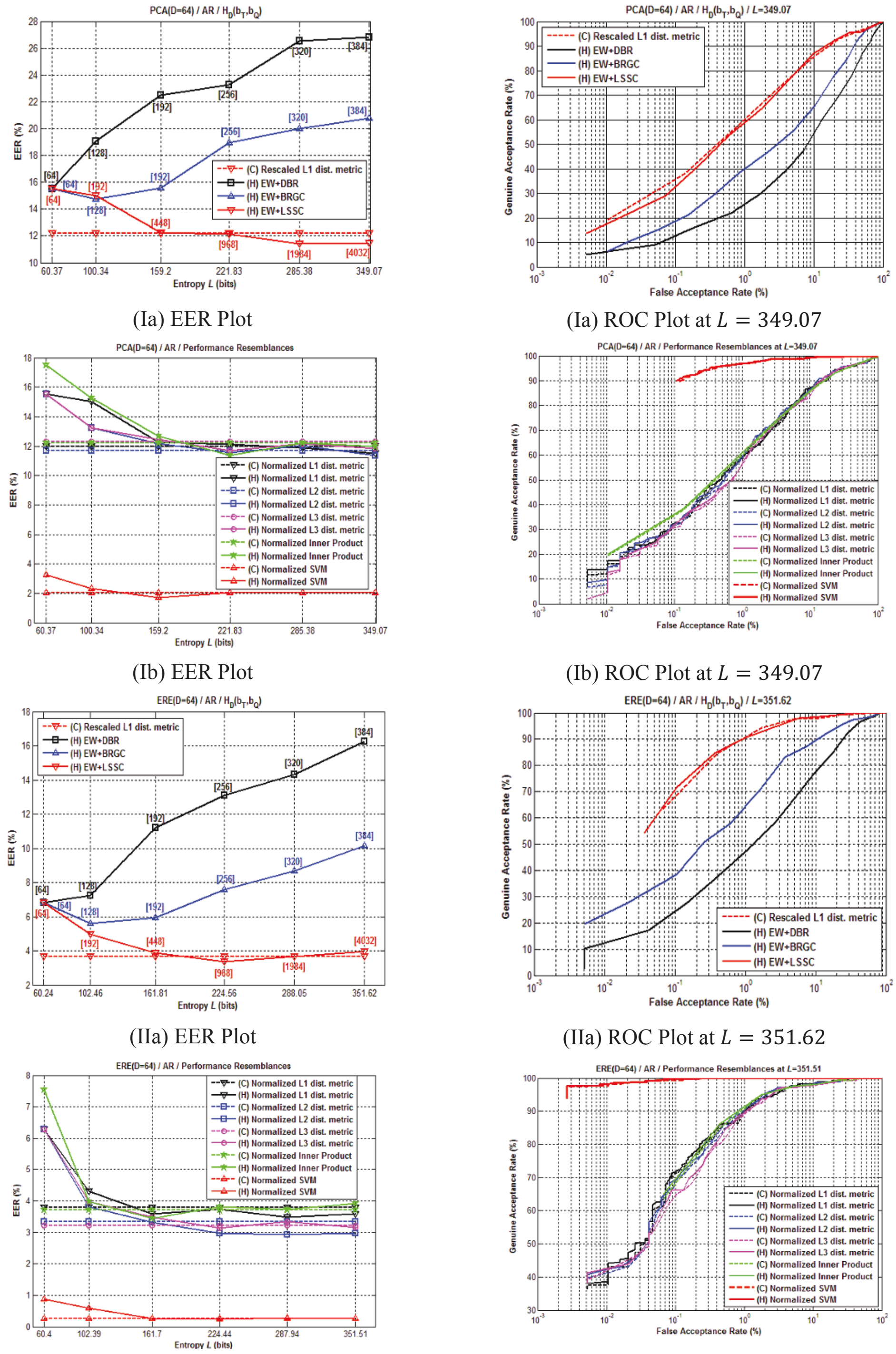
$$L = \sum_{d=1}^{D} l^d = -\sum_{d=1}^{D}\sum_{i=1}^{S^d} p^d_{i^d}\log_2 p^d_{i^d}. \tag{23}$$

Note that due to static bit allocation, S^d = S for all d. Since S^d = 2^n for BRGC & DBR while S = n_{LSSC} + 1 for LSSC, Equation 23 becomes

$$L = \begin{cases} -\sum_{d=1}^{D}\sum_{i=1}^{2^n} p^d_{i^d}\log_2 p^d_{i^d} & \text{for DBR/BRGC encoding based discretization} \\ -\sum_{d=1}^{D}\sum_{i=1}^{n_{\text{LSSC}}+1} p^d_{i^d}\log_2 p^d_{i^d} & \text{for LSSC encoding based discretization} \end{cases} \tag{24}$$

Figure 3 illustrates the EER and the ROC performances of equal-width based discretization and the performance resemblances of EW+LSSC discretization based on the AR face data set. As depicted in Figure 3Ia, IIa for experiments on PCA- and ERE-extracted features, EW + DBR and EW + BRGC discretizations fail to preserve the distances in the index space and therefore deteriorate critically as the number of quantization intervals constructed in each dimension increases, or nearly proportionally, as the entropy L increases. EW + LSSC, on the other hand, achieves not only definite, but also the lowest discretization performance among the discretization schemes especially at high L due to its capability in preserving approximately the rescaled L1 distance-based classification performance.

Another noteworthy observation is that the initially large deviation of EW + LSSC performance from the rescaled L1 distance-based performance tends to decrease as L increases at first and fluctuates trivially after a certain point of L. This can be explained by (6) that since for each dimension, the difference between each continuous value with the central point of the interval (to which we have chosen to scale the discretization output) is upper-bounded by half the width of the interval $\left(\varepsilon^{d}_{i^d,j^d(\max)}\right)$. To augment the entropy L produced by a discretization scheme, the number of intervals/possible outputs from each dimension needs to be increased. As a result, a greatly reduced upper bound of



(Ia) EER Plot

(Ia) ROC Plot at $L = 349.07$

(Ib) EER Plot

(Ib) ROC Plot at $L = 349.07$

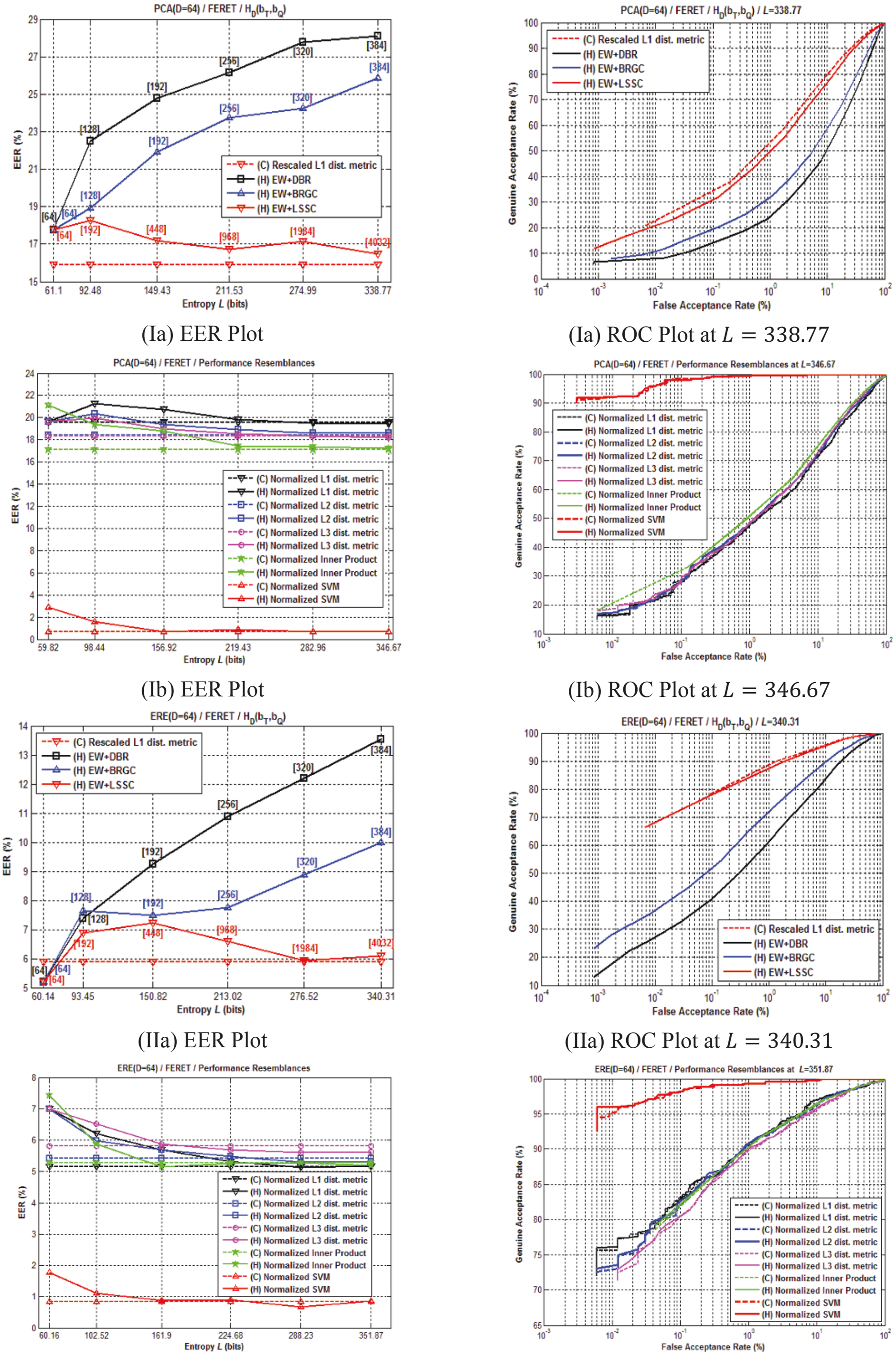
(IIa) EER Plot

(IIa) ROC Plot at $L = 351.62$

(IIb) EER Plot

(IIb) ROC Plot at $L = 351.51$

Figure 3 Results on AR data set for (I) PCA and (II) ERE experiments: (a) EER and ROC performances of EW + DBR, EW + BRGC and EW + LSSC discretizations; and the rescaled L1 distance-based classification; and (b) the performance resemblances of applying EW + LSSC. (C) and (H) denote the performance evaluation in the continuous and the Hamming domains, respectively. Classification performance evaluated in the continuous domain is irrespective to the entropy. '[α]' associated with each reading in the EER plots indicates the corresponding length α of the extracted binary strings.



(Ia) EER Plot

(Ia) ROC Plot at $L = 338.77$

(Ib) EER Plot

(Ib) ROC Plot at $L = 346.67$

(IIa) EER Plot

(IIa) ROC Plot at $L = 340.31$

(IIb) EER Plot

(IIb) ROC Plot at $L = 351.87$

Figure 4 Results on FERET data set for (I) PCA and (II) ERE experiments: (a) EER and ROC performances of EW + DBR, EW + BRGC and EW + LSSC discretizations; and the rescaled L1 distance-based classification; and (b) the performance resemblances of applying EW + LSSC. (C) and (H) denote the performance evaluation in the continuous and the Hamming domains respectively. Classification performance evaluated in the continuous domain is irrespective to the entropy. '[α]' associated with each reading in the EER plots indicates the corresponding length α of the extracted binary strings.

the overall deviation $2D\ \varepsilon^{d}_{i^{d},j^{d}(\max)}$ can eventually be obtained. Therefore, the more the number of intervals is constructed for each dimension, or in other words the higher the overall entropy is desired, the stronger the resemblances will be observed to be. Note that similar observations in Figure 3Ia, IIa illustrate the independence of the resemblances with respect to the feature extraction methods employed.

Perhaps the only limitation arose in achieving such performance resemblances is the inevitable derivation of large length binary string per user when high entropy strength is desired by a system. As shown in Figure 3Ia, IIa, bitstring that is at least four times longer than the entropy is needed to offer 222- and 224-bit entropy respectively, while bit string that is at least six times longer is required to fulfil a 285- and 288-bit system-specific entropy respectively. Indeed, these amounts of binary bits pose high processing challenges to the system capability. However, with the current state of technology advancement, it is expected that processing these binary strings would not raise so much of a critical threat to the current systems.

For performance resemblance experiments on PCA- and ERE-extracted features in Figure 3Ib, IIb, the tendency of the performance resemblances are similar to the previous case where the difference of EER performance in the continuous and the Hamming domains is noticeable at low L and an approximate performance resemblance can be noticed when $L \geq 159.2$ in Figure 3Ib and $L \geq 161.7$ in Figure 3IIb. Therefore, similar explanations applied.

Similar performance trends can be seen in Figure 4 when FERET data set was used. Note that when L increases, EW + DBR and EW + BRGC remain deteriorating badly, as shown in Figure 4Ia, IIa. Their deficit of being indefinite during the discrete-to-binary mapping process can again be justified. Contrarily, the performance of the EW + LSSC discretization remains the lowest and it resembles nearly exactly the rescaled L1 distance-based performance when $L \geq 338.77$ in Figure 4Ia and L ≥ 276.52 in Figure 4IIa. In Figure 4Ib, IIb, the initial performance deviation between each pair of schemes is slightly lower as those in Figure 4Ib, IIb, although perfect resemblance can similarly be observed at high L.

4.3 Summary and discussions

By and large, our general findings can be summarized in the following three aspects:

- When substantial quantization intervals are constructed or a large number of bits are allocated to each feature dimension, equal width (EW) quantization offers an approximate continuous-to-discrete mapping. LSSC outperforms DBR and BRGC in preserving a definite discrete-to-binary mapping behaviour. Overall, adopting equal width quantization with LSSC as a discretizer results in an approximate outcome.
- As long as EW+LSSC is concerned, the distance between two mapped elements in the Hamming domain is fundamentally associated to an approximately rescaled L1 distance between the two continuous counterparts.
- The basic performance resemblance of EW + LSSC discretization to L1 distance-based classification can be extended to Lp distance-based and inner product-based classifications either by flexibly modifying the matching function or by substituting every continuous feature element individually with its binary approximate to obtain a similar classification behaviour in the continuous domain.

We believe that the clarification of the underlying mapping behaviours of EW + LSSC discretization would benefit not only the cryptographic and biometric communities, but also communities from machine learning and data mining areas (i.e. relevant applications include image retrieval [30], image categorization [31], text categorization [32] and etc). In fact, EW + LSSC discretization can be appropriately adopted in any other application that requires transformation from continuous data to binary bitsrings and involves similarity/dissimilarity matching in the Hamming domain so as to attain a deterministically resembled performance of the continuous counterpart.

5. Conclusion

Biometric discretization aims to facilitating numerous security applications through deriving stable representative binary strings in practice. Therefore, understanding the way which discretization may influence on the classification performance is important in warranting the optimal classification performance when discretization is performed. In this paper, we have decomposed equal-width discretization into a two-stage mapping process and performed detailed analysis in the continuous, discrete and Hamming domains in view of different mapping associations among them. Our analysis yields that equal-width quantization exhibits an approximate continuous-to-discrete mapping trend when sufficiently many quantization intervals are constructed while LSSC encoding scheme offers a definite discrete-to-binary mapping behaviour. We have shown that the combination of both such quantization and encoding schemes results in a discretization scheme which offers an approximate rescaled L1 distance-based classification performance in the Hamming metric. Further, we have also illustrated how such a fundamental resemblance can be exploited to obtain other approximate classification performances when binary matching is concerned.

These analysis outcomes have been experimentally supported and the performance resemblances which we have shown are neither dependent to the feature extraction technique (PCA and ERE) nor the dataset (AR and FERET).

List of abbreviations

BRGC: binary reflected gray code; DBR: direct binary representation; EER: equal error rate; ERE: Eigenfeature regularization and extraction; EW: equal width; FAR: false acceptance rate; IP: inner product; LSSC: linearly separable subcode; MLP: multilayer perceptron; PCA: principal component analysis; SVM: support vector machine.

Acknowledgements

This work was supported by the MKE (The Ministry of Knowledge Economy), Korea, under IT/SW Creative research program supervised by the NIPA (National IT Industry Promotion Agency) (NIPA-2010-(C1810-1002-0016)).

Competing interests

The authors declare that they have no competing interests.

Received: 31 January 2011 Accepted: 10 October 2011
Published: 10 October 2011

References

1. Y Chang, W Zhang, T Chen, Biometric-based cryptographic key generation, in *Proceedings of IEEE International Conference on Multimedia and Expo (ICME 2004)*, (2004)
2. Y Dodis, R Ostrovsky, L Reyzin, A Smith, Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. in *Eurocrypt 2004, LNCS*. **3027**, 523–540 (2004). doi:10.1007/978-3-540-24676-3_31
3. F Hao, CW Chan, Private key generation from on-line handwritten signatures. Inf Manage Comp Security **10**(4), 159–164 (2002). doi:10.1108/09685220210436949
4. A Juels, M Wattenberg, A fuzzy commitment scheme, in *6th ACM Conference in Computer and Communication Security (CCS'99)*, 28–36 (1999)
5. TAM Kevenaar, GJ Schrijen, M van der Veen, AHM Akkermans, F Zuo, Face recognition with renewable and privacy preserving binary templates, in *Proceedings of 4th IEEE Workshop on Automatic Identification Advanced Technologies (AutoID '05)*, 21–26 (2005)
6. J-P Linnartz, P Tuyls, New shielding functions to enhance privacy and prevent misuse of biometric templates, in *Proceedings of 4th International Conference on Audio and Video Based Person Authentication (AVBPA 2004), LNCS*. **2688**, 238–250 (2003)
7. F Monrose, MK Reiter, Q Li, S Wetzel, Cryptographic key generation from voice, in *Proceedings of IEEE Symposium on Security and Privacy (S&P 2001)*, 202–213 (2001)
8. F Monrose, MK Reiter, Q Li, S Wetzel, Using voice to generate cryptographic keys, in *Proceedings of Odyssey 2001, The Speaker Verification Workshop* (2001)
9. ABJ Teoh, DCL Ngo, A Goh, Personalised cryptographic key generation based on FaceHashing. Comput. Security **23**(7), 606–614 (2004). doi:10.1016/j.cose.2004.06.002
10. P Tuyls, AHM Akkermans, TAM Kevenaar, G-J Schrijen, AM Bazen, NJ Veldhuis, Practical biometric authentication with template protection, in *Proceedings of 5th International Conference on Audio- and Video-based Biometric Person Authentication, LNCS*. **3546**, 436–446 (2005). doi:10.1007/11527923_45
11. E Verbitskiy, P Tuyls, D Denteneer, JP Linnartz, Reliable biometric authentication with privacy protection, in *24th Benelux Symposium on Information Theory*, 125–132 (2003)
12. WK Yip, Goh A, Ngo DCL, Teoh ABJ, Generation of replaceable cryptographic keys from dynamic handwritten signatures, in *Proceedings of 1st International Conference on Biometrics, Lecture Notes in Computer Science*. **3832**, 509–515 (2006)
13. ABJ Teoh, WK Yip, K-A Toh, Cancellable biometrics and user-dependent multi-state discretization in BioHash. Pattern Anal Appl. **13**(3), 301–307 (2009)
14. C Chen, R Veldhuis, T Kevenaar, A Akkermans, Multi-bits biometric string generation based on the likelihood ratio, in *Proceedings of IEEE International Conference on Multimedia and Expo (ICME 2004)*. **3**, 2203–2206 (2004)
15. C Chen, R Veldhuis, T Kevenaar, A Akkermans, Biometric quantization through detection rate optimized bit allocation. EURASIP J Adv Signal Process. **16** (2009)
16. C Chen, R Veldhuis, Extracting biometric binary strings with minimal area under the FRR curve for the Hamming distance classifier. Signal Process. **91**, 906–918 (2011). doi:10.1016/j.sigpro.2010.09.008
17. F Gray, Pulse code communications. U.S Patent. **2632058** (1953)
18. J Galbally, J Fierrez, J Ortega-Garcia, C McCool, S Marcel, Hill-climbing attack to an Eigenface-Based face verification system, in *1st IEEE International Conference on Biometrics, Identity and Security (BIdS)*, 1–6 (2009)
19. A Kumar, D Zhang, Hand geometry recognition using entropy-based discretization. IEEE Trans Inf Forens Security, 2, 181–187 (2007)
20. M-H Lim, ABJ Teoh, Linearly separable subcode, A novel output label with high separability for biometric discretization, in *Proceedings of 5th IEEE Conference on Industrial Electronics and Applications (ICIEA'10)* (2010)
21. H Simon, *Neural Networks: A Comprehensive Foundation*, Second Edition (Prentice Hall, New York, 1998)
22. C Cortes, V Vapnik, Support-vector networks. Mach Learn. **20**(3), 273–297 (1995)
23. TM Cover, PE Hart, Nearest neighbor pattern classification. IEEE Trans Inform Theory **13**(1), 21–27 (1967)
24. MD Buhmann, *Radial Basis Functions: Theory and Implementations* (Cambridge University, Cambridge, United Kingdom, 2003)
25. K Pearson, Notes on the history of correlation. Biometrika **13**(1), 25–45 (1895)
26. AM Martinez, R Benavente, The AR Face Database. CVC Technical Report # 24 (1998)
27. PJ Philips, H Moon, PJ Rauss, S Rizvi, The FERET evaluation methodology for face recognition algorithms. IEEE Trans Pattern Anal Mach Intell. **22**(10) (2000)
28. M Turk, A Pentland, Eigenfaces for recognition. J Cognit, Neurosci. **3**(1), 71–86 (1991). doi:10.1162/jocn.1991.3.1.71
29. XD Jiang, B Mandal, A Kot, Eigenfeature regularization and extraction in face recognition. IEEE Trans Pattern Anal Mach Intell. **30**(3), 383–394 (2008)
30. R Datta, D Joshi, J Li, JZ Wang, Image retrieval: ideas, influences, and trends of the new age. ACM Comput Surveys **40**(2), 1–60 (2008)
31. Y Chen, JZ Wang, Image categorization by learning and reasoning with regions. J Mach Learn Res. **5**, 913–939 (2004)
32. F Sebastiani, Machine learning in automated text categorization. ACM Comput Surveys **34**(1), 1–47 (2002). doi:10.1145/505282.505283

doi:10.1186/1687-6180-2011-82
Cite this article as: Lim *et al.*: An analysis on equal width quantization and linearly separable subcode encoding-based discretization and its performance resemblances. *EURASIP Journal on Advances in Signal Processing* 2011 **2011**:82.