Biometric Hashing Based on Genetic Selection and Its Application to On-Line Signatures

Manuel R. Freire, Julian Fierrez, Javier Galbally, and Javier Ortega-Garcia

Biometric Recognition Group - ATVS,
Escuela Politecnica Superior, Universidad Autonoma de Madrid
C/ Francisco Tomas y Valiente 11, E-28049 Madrid, Spain
{m.freire,julian.fierrez,javier.galbally,javier.ortega}@uam.es

Abstract. We present a general biometric hash generation scheme based on vector quantization of multiple feature subsets selected with genetic optimization. The quantization of subsets overcomes the dimensionality problem of other hash generation algorithms, while the feature selection step using an integer-coding genetic algorithm enables to exploit all the discriminative information found in large feature sets. We provide experimental results of the proposed hashing for verification of on-line signatures. Development and evaluation experiments are reported on the MCYT signature database, comprising $16,500$ signatures from 330 subjects.

Keywords: Biometric Hashing, Biometric Cryptosytems, Feature Selection, Genetic Algorithms.

1 Introduction

The application of biometrics to cryptography is receiving increasing attention from the research community. Cryptographic constructions known as *biometric cryptosystems* using biometric data have been recently proposed, exploiting the advantages of authentication based on something that you are (e.g., your fingerprint or signature), instead of something that you know (e.g., a password) [1,2].

A review of the state of the art in biometric cryptosystems is reported in [2]. It establishes a commonly accepted classification of biometric cryptosystems, namely: (*i*) *key release*, where a secret key and a biometric template are stored in the system, the key being released after a valid biometric match, and (*ii*) *key generation*, where a template and a key are combined into a unique token, such that it allows reconstructing the key only if a valid biometric trait is presented. This last scheme has the particularity that it is also a form of cancelable biometrics [3] (i.e., the key can be changed), and it is secure against system intruders since the stored token does not reveal information from neither the key nor the biometric.

Within key generation biometric cryptosytems, the biometric template can be extracted using a *biometric hashing* scheme, where a binary string is obtained from the biometric sample (see Fig. 1). In this arquitecture, biometric cryptosystems have stronger security constraints than biometric hashing schemes, where the extraction of a stable binary representation of the biometric is generally priorized.

S.-W. Lee and S.Z. Li (Eds.): ICB 2007, LNCS 4642, pp. 1134–1143, 2007.

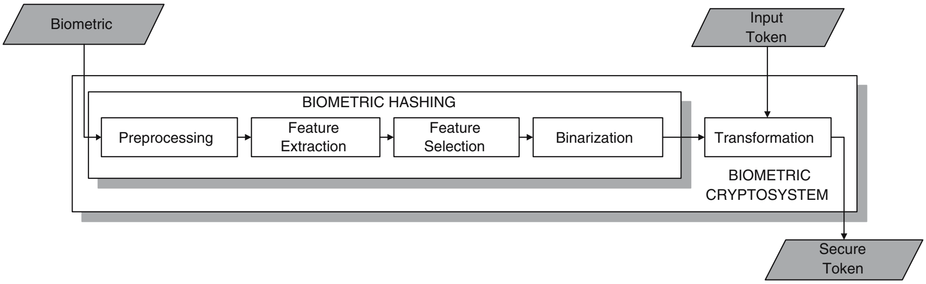


Fig. 1. A generic biometric cryptosystem, where the biometric is binarized using a biometric hashing scheme

We present a biometric hashing scheme based on genetic selection, extending the idea of feature subset concatenation presented in [4]. In that previous work, there was no clear indication of which of all the possible feature subsets should be used for the biometric hash. Moreover, there is a need for a dimensionality reduction criterion when dealing with high-dimensional vectors. We provide a solution to this problem using feature subset selection based on a genetic algorithm (GA) [5], leading to a practical implementation of biometric hashing.

The proposed hash generation scheme can be applied to any biometric trait represented as a fixed-sized feature vector. In this work, we present a case study for the application of this scheme to the verification of on-line signatures, where dynamic information of the signing process is available. Within biometric hashing, handwritten signature has an interesting application in authentication and identity management, due to its widespread social and legal acceptance [6,7].

This paper is structured as follows. In Sect. 2 we outline related work on hashing and biometric cryptosystems. The proposed hashing scheme is presented in Sect. 3.1, and the feature selection algorithm using GA is detailed in Sect. 3.2. A case study in biometric hash generation from on-line signatures is reported in Sect. 4. Finally, some conclusions and future work are discussed in Sect. 5.

2 Related Work

Several biometric cryptosystems for key generation have been proposed in the literature. The *fuzzy vault* scheme [8] establishes a framework for biometric cryptosystems. In this construction, a secret (typically, a random session key) is encoded using an unordered set of points A, resulting in an indivisible vault V. The original secret can only be reconstructed if another set B is presented and overlaps substantially with A. The fuzzyness of this construction fits well with the intra-variability of biometrics. Uludag et al. [9] proposed a biometric cryptosystem for fingerprints based on the fuzzy vault, where the encoding and the decoding sets were vectors of minutiae data. Their scheme was further developed in [10], where the fuzzy vault for fingerprints is enhanced with helper data extracted from the orientation field flow curves. Other works have applied the fuzzy vault to on-line signature data using function-based information [11].

Hoque et al. [4] present a biometric hashing scheme for biometrics, where the generated hash plays the role of a cryptographic key. Their work identifies the problem of intra-variability and proposes a hashing based on vector quantization of feature subsets. However, the problem of the high dimensionality in the feature vector is not considered in their contribution. Also, their evaluation considers the hash as a final cryptographic key and not as a building block of a biometric cryptosystem, and therefore performance is measured in terms of exact matching among two hashes. Another approach was presented in [12], where Vielhauer et al. propose a biometric hashing scheme for statistical features of on-line signatures. Their work is based on user-dependent helper data, namely an Interval Matrix. Vielhauer and Steinmetz further applied this scheme to biometric hash generation using handwriting [13].

Another approach to crypto-biometrics using handwritten signature is Bio-Hashing, where pseudo-random tokens and biometrics are combined to achieve higher security and performance [14,15]. This scheme has also been applied to face biometrics in [16].

Information-theoretical approaches to crypto-biometrics have also been presented. One example is the work of Dodis et al. [17], where a theoretical framework is presented for cryptography with fuzzy data (here, biometrics). They propose two primitives: a *secure sketch*, which produces public information about a biometric signal that does not reveal details of the input, and a *fuzzy extractor*, wich extracts nearly uniform randomness from a biometric input in an error-tolerant way helped by some public string. Also, they propose an extension of the fuzzy vault which is easier to evaluate theoretically than the original formulation of Juels and Sudan [8].

3 Biometric Hash Generation

We present a biometric hash generation scheme based on the concatenation of binary strings extracted from a set of feature vector subsets. We extend the previous work by Hoque et al. [4], where vector quantization is applied to feature subsets, which are further concatenated to form the biometric hash. We provide a solution for high-dimensional vectors by means of feature selection based on an integer-coding genetic algorithm.

3.1 Feature Subset Concatenation

Given a feature vector $\mathbf{x} = [x_1, \ldots, x_N]$ with $x_i \in \mathcal{R}$, a biometric hash $\mathbf{h} = [h_1, \ldots, h_L]$ with $h_i \in \{0, 1\}$ of dimension L is extracted. Let $\mathbf{x}^j$ with $j = 1, \ldots, D$ be formed by a subset of features of $\mathbf{x}$ of dimension M ($M < N$), with possibly overlapping features for different j. Let C^j be a codebook obtained by vector quantization of feature subset $\mathbf{x}^j$ using a development set of features $\mathbf{x}^j_{k=1,\ldots,K}$. We define $\mathbf{h}$ for an input feature vector $\mathbf{x}_T$ as:

$$\mathbf{h}(\mathbf{x}_T) = \underset{j=1,\ldots,D}{\text{concat}} \left(f(\mathbf{x}^j_T, C^j) \right) \tag{1}$$

where f is a function that assigns the nearest-neighbour codewords, and concat$(\cdot)$ denotes the concatenation of binary strings.

The codebooks C^j are computed with vector quantization as follows. Let $\mathbf{x}^j_{k=1,\ldots,K}$ be feature vector subsets forming a development set. The k-means algorithm is used to compute the centroids of the underlying clusters, for a given number of clusters Q. Then, centroids are ranked based on their distance to the mean of all centroids. Finally, binary codewords of size $q = \log_2 Q$ are defined as the position of each centroid in the ranking using Gray coding [18].

3.2 Feature Selection Using Genetic Algorithms

GA are non-deterministic methods inspired in natural evolution, which apply the rules of selection, crossover and mutation to a population of possible solutions in order to optimize a given fitness function [5]. In the present work, a GA with integer coding is implemented in order to obtain the best subsets of M features. Integer coding has been used instead of binary coding, since the last one does not fit well when the number of features is fixed.

Algorithm 1. Feature subset selection using GA

```
Input: $n, S, \theta$
Output: $A$
  $F \leftarrow S$
  $A \leftarrow \emptyset$
  for $i \leftarrow 1$ to $n$ do
    $B \leftarrow \mathrm{GA}(F)$ {Call GA, returns a sorted list of candidate subsets}
    for all $b \in B$ do
      if $b \cap a \leq \theta, \forall a \in A$ then
        $A \leftarrow A \cup b$
      end if
    end for
    $N \leftarrow \emptyset$
    for all $a \in A$ do
      $N \leftarrow N \cup a$
    end for
    $N \leftarrow \mathrm{unique}(N)$ {Remove repeated items}
    $F' \leftarrow \emptyset$
    for $j \leftarrow 1$ to $|F|/2$ do
      $F' \leftarrow F' \cup N_j$
    end for
    $F \leftarrow F - F'$
  end for
```

The proposed iterative algorithm for feature subset selection using GA is presented in Algorithm 1. Note that the proposed algorithm can be easily modified to use a different feature selection technique such as SFFS [19].

In words, Algorithm 1 receives the number of iterations n, the initial feature set S, and the threshold θ, which represents the maximum number of overlapped features permitted among different subsets. For the feature set F, initially equal to S, the feature selection algorithm is called (here, the GA). From the output (a sorted list of subsets of size M), subsets are selected iteratively if no previously selected subset overlaps with the one at hand in more than a certain threshold θ. This way, the threshold settles the degree of correlation allowed among the different subsets.

In the proposed scheme, the fitness function of the GA has been defined as $f = \mathrm{EER}^{-1}$, where the EER is computed for skilled forgeries from a development set different to the training set used for vector quantization (see Sect. 4.1).

After the subsets selection, the half of the features with the best performance are removed from F, and the algorithm is iterated. This strategy was followed in order to avoid the possible loss of not so discriminative sets of features, that nevertheless provide complementary information to the previously selected features.

4 Case Study: Biometric Hash Generation from Feature-Based Information of On-Line Signatures

4.1 Signature Database and Experimental Protocol

The MCYT on-line signature corpus is used for the experiments [20]. This database contains 330 users with 25 genuine signatures and 25 skilled forgeries per user, captured in four acquisition sites. Forgers were asked to imitate after observing the static image of the signature to imitate, they tried to copy them at least 10 times, and then they wrote the forgeries naturally without breaks or slowdowns.

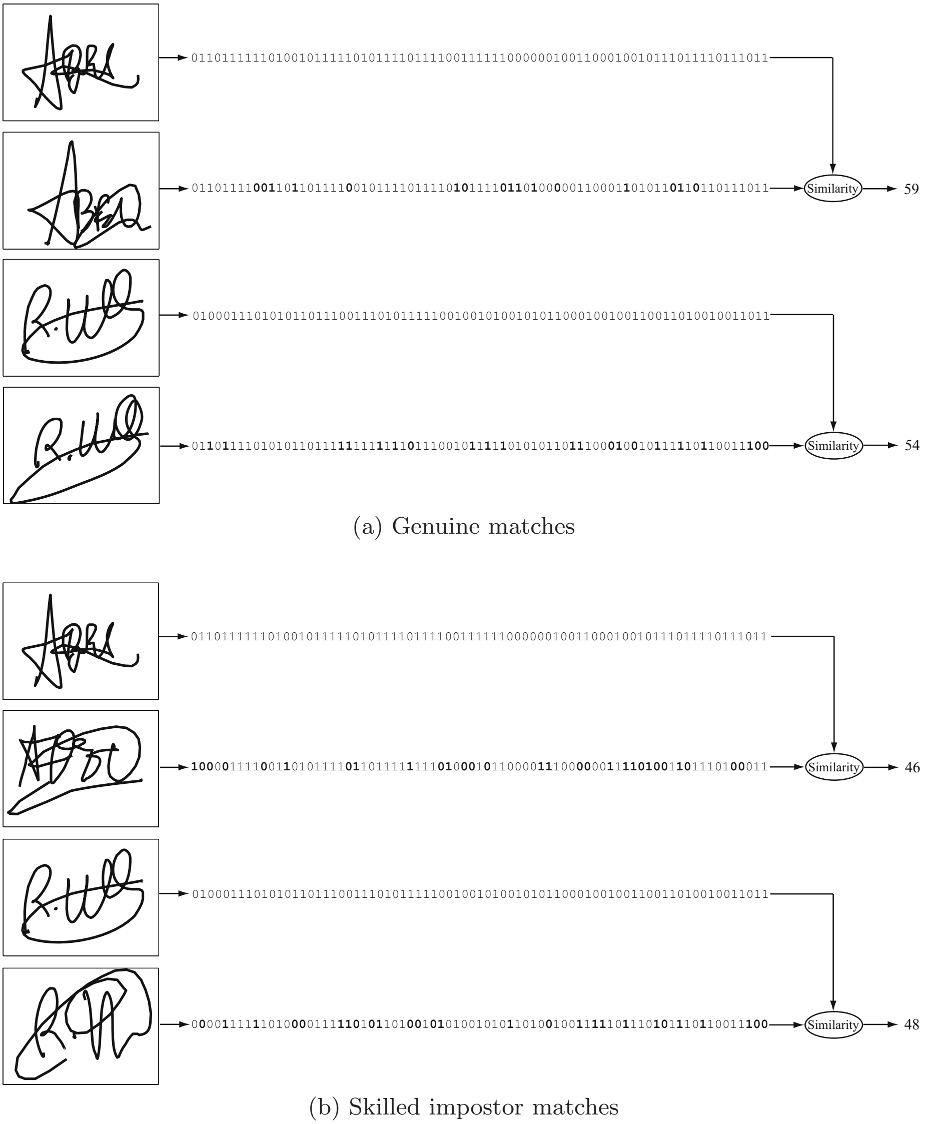
For the experiments presented here, we have followed a 2-fold cross-validation strategy. The database has been divided into two sets: a *development* set, formed by the even users, and an *evaluation* set, with the odd users. The development set has been further partitioned into: *training* set (for vector quantization), with the even users of the development set, and *testing* set (for GA optimization), with the rest.

Evaluation experiments were conducted as follows. A binary hash was generated for each genuine signature in the database, and compared to the hashes of the remaining genuine signatures of the user at hand, all her skilled forgeries, and the first genuine signature of the remaining users (random forgeries).

Genuine and impostor matching scores were calculated using the similarity function $s_H(\mathbf{h}_1, \mathbf{h}_2) = q_{\max} - d_H(\mathbf{h}_1, \mathbf{h}_2)$, where d_H represents Hamming distance and $\mathbf{h}_1$ and $\mathbf{h}_2$ are binary vectors of size $q_{\max}$ [21]. The EER between genuine and either skilled of random forgeries using this similarity measure is used as the performance criterion of the hashing scheme. Examples of the matching of genuine and skilled hashes are included in Fig. 2.

4.2 Feature Extraction from On-Line Signatures

For the experiments we use an on-line signature representation based on global features [7]. In particular, a 100-dimensional global feature vector is extracted from each on-line signature [22], including features based on timing information, number of strokes, geometry, etc.



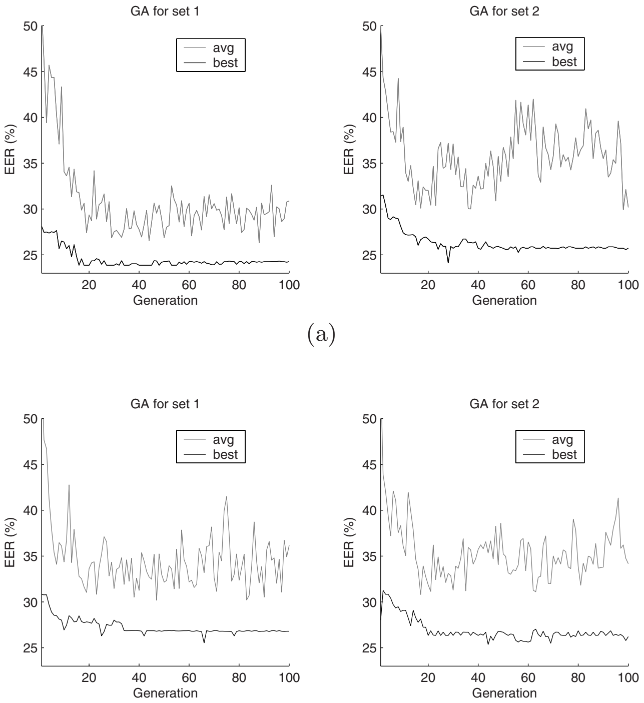
(a) Genuine matches

(b) Skilled impostor matches

Fig. 2. Examples of the matching between a genuine template and a genuine test hash (a), and between a genuine template and a skilled forgery (b). The binary strings correspond to the real hashes obtained with overlap $\theta = 0$. Different bits are represented in **bold**.

Each feature x_i is normalized into the range $[0, 1]$ using *tanh*-estimators [23]. The normalization is given by:

$$x'_i = \frac{1}{2}\left\{\tanh\left(0.1\left(\frac{x_i - \mu_i}{\sigma_i}\right)\right) + 1\right\} \tag{2}$$



(a)

(b)

Fig. 3. Results of the first (a) and second (b) execution of the GA. Following the 2-fold cross-validation approach, set 1 (left) corresponds to the development set formed by the odd users, and set 2 (right) by the even users.

Table 1. Biometric hashing scheme performance for different overlapping threshold θ

Overlap (θ)	Number of subsets	Hash length	EER skilled (%)	EER random (%)
0	25	75	18.83	8.02
1	171	513	12.99	3.50
2	530	1590	12.15	3.16

where μ_i and σ_i are the mean and the standard deviation of the feature x_i for the genuine signatures in the database.

4.3 Implementation of the Genetic Algorithm

In the proposed integer-coding genetic selection, a string of the genetic population is represented by a vector of length M, where M is the dimension of the subsets to be found. Each element of the string is an integer in the range $[1, 100]$ and corresponds to a feature in the original set.

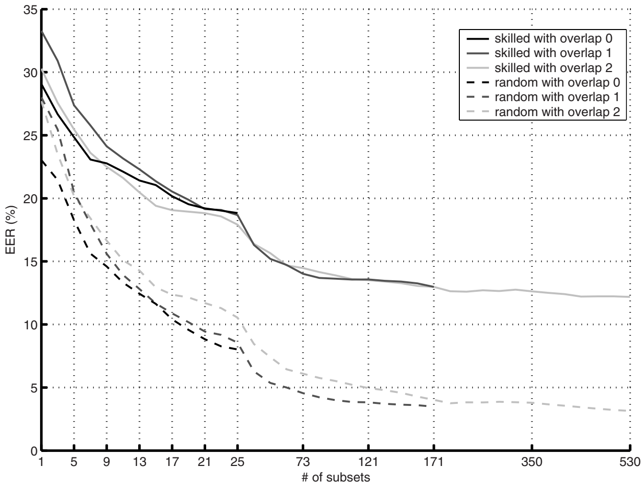


Fig. 4. EER (in %) for increasing number of subsets of the biometric hash for overlapping threshold $\theta = 0$, 1 and 2, respectively

The configuration set of the genetic algorithm is the following:

- Population size: 100, randomly generated in the first generation.
- String size: 4, representing the number M of features in the subset.
- Stop condition: after completing 100 generations.
- Selection: binary tournament.
- Crossover: one-point, with 85% probability.
- Mutation: mutated elements are randomly assigned a value in the range $[1, 100]$ that is not present in the string, with 5% probability.

The output of our genetic algorithm is the whole set of the strings produced along the evolution of the GA, sorted by descending order of fitness.

4.4 Experimental Results

Development Experiments. Algorithm 1 was executed for parameters $n = 2$, $S = [1, 100]$ and $\theta = 0$, 1, and 2, respectively. The number of clusters in the k-means algorithm was fixed to 8. As a result, codewords extracted from each subset have a bit size of $\log_2 8 = 3$.

In Fig. 3(a) we present the evolution of the first iteration of the feature subset selection algorithm, corresponding to an execution of the genetic algorithm. We observe that the best string (feature subset) converges to an EER value of about 24% for skilled forgeries. Results of the second execution are presented in Fig. 3(b). Interestingly, the best EER is only slightly worse when considering the 50 features discarded from the first iteration.

Evaluation Results. The proposed hashing scheme was evaluated for the subsets and the codebooks obtained in the development experiments. Matching scores were computed using the similarity measure described in Sect. 4.1.

EERs using an increasing number of subsets are presented in Fig. 4 for $\theta = 0$, 1 and 2. The evaluation results are summarized in Table 1. We observe that the best EER for skilled and random forgeries is achieved when a high number of subsets is considered ($\theta = 2$). However, it is worth noting that a big hash length does not imply a higher security, since large hashes include redundant information.

5 Conclusions

We have proposed a general hash generation scheme for fixed-length feature-based approaches to biometrics. Our scheme includes a feature selection step based on genetic algorithms, providing a practical solution for high-dimensional feature vectors. The proposed scheme can be used as a building block of biometric cryptosystems.

Experiments have been conducted on signature data from the MCYT database using 2-fold cross-validation. We have studied the effect of using subsets with a variable number of overlapping features. We observed that the best EER is achieved with the configuration that involves more feature subsets (overlapping threshold of 2).

A future direction of this research will be the comparison of the GA with other feature selection strategies. Also, the effect of other parameters as the quantization size or the subset length will be considered. The redundancy of the resulting hashes is also yet to be studied, as well as the application of the proposed hashing to other biometrics.

Acknowledgements

This work has been supported by Spanish Ministry of Education and Science (project TEC2006-13141-C03-03) and BioSecure NoE (IST-2002-507634). M. R. F. is supported by a FPI Fellowship from Comunidad de Madrid. J. F. is supported by a Marie Curie Fellowship from European Commission. J. G. is supported by a FPU Fellowship from the Spanish Ministry of Education and Science.

References

1. Jain, A.K., Ross, A., Pankanti, S.: Biometrics: A tool for information security. IEEE Trans. on Information Forensics and Security 1(2), 125–143 (2006)
2. Uludag, U., Pankanti, S., Prabhakar, S., Jain, A.K.: Biometric cryptosystems: Issues and challenges. Proceedings of the IEEE 92(6), 948–960 (2004)
3. Bolle, R.M., Connell, J.H., Ratha, N.K.: Biometric perils and patches. Pattern Recognition 35(12), 2727–2738 (2002)

4. Hoque, S., Fairhurst, M., Howells, G., Deravi, F.: Feasibility of generating biometric encryption keys. Electronics Letters 41(6), 309–311 (2005)
5. Goldberg, D.E.: Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley Longman Publishing Co. Inc., Boston, MA, USA (1989)
6. Plamondon, R., Srihari, S.N.: On-line and off-line handwriting recognition: A comprehensive survey. IEEE Trans. PAMI 22(1), 63–84 (2000)
7. Fierrez, J., Ortega-Garcia, J.: On-line signature verification. In: Jain, A.K., Ross, A., Flynn, P. (eds.) Handbook of Biometrics (to appear)
8. Juels, A., Sudan, M.: A fuzzy vault scheme. Design Code. Cryptogr. 38(2), 237–257 (2006)
9. Uludag, U., Pankanti, S., Jain, A.K.: Fuzzy vault for fingerprints. In: Kanade, T., Jain, A., Ratha, N.K. (eds.) AVBPA 2005. LNCS, vol. 3546, pp. 310–319. Springer, Heidelberg (2005)
10. Uludag, U., Jain, A.K.: Securing fingerprint template: Fuzzy vault with helper data. In: Proc. CVPRW, p. 163. IEEE Computer Society, Los Alamitos (2006)
11. Freire-Santos, M., Fierrez-Aguilar, J., Ortega-Garcia, J.: Cryptographic key generation using handwritten signature. In: Proc. SPIE., vol. 6202, pp. 225–231 (2006)
12. Vielhauer, C., Steinmetz, R., Mayerhoefer, A.: Biometric hash based on statistical features of online signatures. In: Proc. ICPR., vol. 1, pp. 123–126 (2002)
13. Vielhauer, C., Steinmetz, R.: Handwriting: Feature correlation analysis for biometric hashes. EURASIP JASP 2004(4), 542–558 (2004)
14. Teoh, A.B., Goh, A., Ngo, D.C.: Random multispace quantization as an analytic mechanism for biohashing of biometric and random identity inputs. IEEE Trans. PAMI 28(12), 1892–1901 (2006)
15. Lumini, A., Nanni, L.: An improved biohashing for human authentication. Pattern Recognition 40(3), 1057–1065 (2007)
16. Ngo, D.C.L., Teoh, A.B.J., Goh, A.: Biometric hash: high-confidence face recognition. IEEE Trans. Circ. Syst. Vid. 16(6), 771–775 (2006)
17. Dodis, Y., Reyzin, L., Smith, A.: Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. In: Cachin, C., Camenisch, J.L. (eds.) EUROCRYPT 2004. LNCS, vol. 3027, pp. 523–540. Springer, Heidelberg (2004)
18. Lin, S., Costello, D.J.: Error Control Coding, 2nd edn. Prentice-Hall, Inc. Upper Saddle River, NJ, USA (2004)
19. Pudil, P., Novovicova, J., Kittler, J.: Floating search methods in feature selection. Pattern Recogn. Lett. 15(11), 1119–1125 (1994)
20. Ortega-Garcia, J., et al.: MCYT baseline corpus: A bimodal biometric database. IEE Proc. Vision, Image and Signal Processing 150(6), 395–401 (2003)
21. Theodoridis, S., Koutroumbas, K.: Pattern Recognition. Academic Press, San Diego (2006)
22. Fierrez-Aguilar, J., et al.: An on-line signature verification system based on fusion of local and global information. In: Roli, F., Vitulano, S. (eds.) ICIAP 2005. LNCS, vol. 3617, pp. 523–532. Springer, Heidelberg (2005)
23. Jain, A.K., Nandakumar, K., Ross, A.: Score normalization in multimodal biometric systems. Pattern Recognition 38(12), 2270–2285 (2005)