Fingerprint Indexing Based on Combination of Novel Minutiae Triplet Features

Wei Zhou[1], Jiankun Hu[1,⋆], Song Wang[2], Ian Petersen[1], and Mohammed Bennamoun[3]

[1] School of Engineering and Information Technology, The University of New South Wales, Canberra, Australia ACT 2600
wei.zhou@student.adfa.edu.au, {J.Hu,i.petersen}@adfa.edu.au

[2] School of Engineering and Mathematical Sciences, LaTrobe University, Melbourne, Australia VIC 3086
song.wang@latrobe.edu.au

[3] School of Computer Science and Software Engineering, The University of Western Australia, Perth, Australia WA 6009
m.bennamoun@csse.uwa.edu.au

Abstract. Fingerprint indexing is a process of pre-filtering the template database before matching. The most common features used for fingerprint indexing are based on minutiae triplets. In this paper, we investigated the indexing performance based on some commonly used features of minutiae triplets and proposed to combine these features with some novel features of minutiae triplets for fingerprint indexing. Experiments on FVC 2000 DB2a and 2002 DB1a show that the proposed indexing method can perform better than state-of-the-art schemes for full fingerprint indexing, meanwhile, experimental results on NIST SD 14 show that the performance is improved significantly after the new features are added to the feature space, and is fairly good even for partial fingerprint indexing.

1 Introduction

Biometrics such as fingerprint, face, iris, keystroke dynamics and ECG have been widely used for authentication [1] [2] [3] [4] [5], of which fingerprint is an ideal biometric trait for many applications in modern security systems, ranging from access control, criminal identification, to the emerging bio-cryptography [6] [7] [8] [9] [10] [11] [12] [13]. Among these applications, access control is a fingerprint verification (or fingerprint authentication) process, which is used to verify whether the fingerprint of a claimed identity matches the corresponding fingerprint enrolled and stored in a database; criminal identification is a process of

⋆ Corresponding author.

M.H. Au et al. (Eds.): NSS 2014, LNCS 8792, pp. 377–388, 2014.

fingerprint identification (or fingerprint retrieval), which is used to identify an unknown person by searching a template fingerprint database. Fingerprint identification is a process of one-to-many comparisons and is usually time-consuming if the template database is large. Conventional solutions are based on exclusive classification techniques whereby fingerprints are first classified into several classes to reduce the search space, such as Arch, Loop and Whorl. Such exclusive classification based schemes are not effective enough because more than 90% of fingerprints belong to only three super classes (left loops, right loops and whorls).

To address these problems, fingerprint indexing (or continuous fingerprint classification) was developed, whereby instead of classifying fingerprints into limited and predefined classes, fingerprint indexing techniques use feature vectors to describe fingerprints. Through similarity preserving transformations, these feature vectors form a multidimensional feature space, where similar fingerprints characterized by similar features are arranged as neighbors in the feature space. For identification, the query fingerprint is mapped into a point in the same feature space, and the neighboring fingerprints are compared one by one until a match is found or a certain number of the neighbors have been compared. The penetration rate, defined as the percentage of database searched until a match is found, and the hit rate, defined as the probability of retrieving the correct identity, are commonly used to measure the performance of an indexing scheme. Higher hit rate together with lower penetration rate means better indexing performance.

Among various fingerprint features, level 1 features (Orientation Field, Singular Points, Ridge Frequency) and level 2 features (minutiae) are often used for fingerprint indexing, wherein features extracted from minutiae triplets formed by minutia points are most popular. In this paper, we investigated the performance of fingerprint indexing based on some common features of minutiae triplets and proposed an indexing scheme by combining these features with some novel features which are easily obtainable from minutiae information. We carried out a series of experiments on both full and partial fingerprint databases to evaluate the proposed approach. Experimental results on FVC 2000 DB2a and 2002 DB1a show that the proposed indexing approach can achieve better performance than state-of-the-art methods for full fingerprint indexing. Meanwhile, the experimental results of the indexing scheme on partial fingerprints, which were generated from NIST SD 14, by adding new features incrementally to the common feature set, show that the performance is much better after new features are considered, and can even be comparable to that on full fingerprint indexing if the parameters are chosen properly.

The rest of this paper is organized as follows. Section 2 is a brief introduction to the related work on fingerprint indexing. Section 3 elaborates on the generation of the minutiae triplets features including both common features and the newly proposed features, and the indexing scheme based on these features. Experimental evaluations on several public fingerprint databases are demonstrated in Section 4 and Section 5 concludes the whole work.

2 Related Work

2.1 Full Fingerprint Indexing Techniques

A number of fingerprint indexing schemes based on level one features have been proposed since 1990s. Lumini et al. [14] were the first to propose the idea of indexing for fingerprint identification and used orientation field as feature vectors. Cappelli et al. [15] used fingerprint prototype masks to generate feature vectors and studied several different strategies. In addition to these level one features which are primarily used for fingerprint indexing, some other features have also been investigated. Bhanu and Tan [16] proposed to index fingerprints using minutiae triplets. Boer et al. [17] investigated the use of the orientation field, FingerCode and minutiae triplets as the input feature vectors and concluded that the orientation field performs the best if only a single type of feature were to be used. Wang et al. [18] proposed to use FOMFE (Fingerprint Orientation Model based on 2D Fourier Expansion) coefficients as feature vectors for fingerprint indexing, which has achieved the fastest feature generation speed. All of these indexing techniques are reported to be able to achieve a high hit rate at a low penetration rate when applied on full fingerprint images [16].

2.2 Partial Fingerprint Indexing Techniques

Different from full fingerprint images, the missing part of a partial fingerprint may contain significant information which is hardly feasible, therefore, researchers attempted to make full use of all levels of features that can be extracted from the existing segments. Feng and Jain [19] [20] developed a multi-staged filtering system to reduce the search space while retrieving the potential candidates for large-scale latent fingerprint matching. Yuan et al. [21] used the number of matched minutiae polygons derived from matching information of minutiae triplets as well as minutiae triplets themselves to speed up the indexing. Alessandra et al. [22] proposed to index latent fingerprint by fusion of level 1 and level 2 features. These filtering or indexing schemes either depend on the singular points which are hardly found in the partial fingerprint segment, or involve excessive computation on the minutiae information. Wang and Hu [23] applied FOMFE model [18] to address partial fingerprint identification from another angle. They developed algorithms to extend the partial ridge flows smoothly into the unknown segment and used the reconstructed features to form the indexing space. This approach has shown very promising results in reducing the size of candidate lists for matching when applied in fingerprint indexing, and the feature set does not include singular points.

3 The Proposed Indexing Scheme

In this paper, we employ the local features derived from each noncollinear minutiae triplet for fingerprint indexing.

3.1 Fingerprint Representation

The features of a minutia extracted from a fingerprint image usually include its coordinates (x, y), local ridge orientation θ and minutia type (ridge bifurcation or ending denoted by 1 or 0). In our approach, a commercial fingerprint verification software VeriFinger SDK [24] was adopted to extract minutiae information for both full fingerprint images and partial fingerprint images. After extraction, the resulting minutiae are sorted according to their y coordinates in an ascending order, which will benefit the choose of minutiae points to form minutiae triplets.

Common Features of Minutiae Triplets

- **Triangle handedness [16]:** Suppose P_1, P_2, P_3 are the three minutiae to form a triangle and their y coordinates are in an ascending order. We choose P_1 as the first vertex and use (x_i, y_i) to denote the coordinates of minutiae $P_i, i = 1, 2, 3$. Define $\phi = (x_2 - x_1) \times (y_3 - y_1) - (y_2 - y_1) \times (x_3 - x_1)$. If $\phi > 0$, P_1, P_2, P_3 are in counter-clockwise order, then we set the vertices as $\{P_1, P_2, P_3\}$; otherwise, we order the vertices as $\{P_1, P_3, P_2\}$. By this means, we make sure that the vertices of all triangles are arranged in the counter-clockwise direction. Before the extraction of other features, we suppose P_1, P_2, P_3 are already in counter-clockwise order.
- **Lengths of each side [25]:** Let $Z_i = x_i + jy_i$ be the complex number $(j = \sqrt{-1})$ corresponding to the coordinate (x_i, y_i) of $P_i, i = 1, 2, 3$. Define $Z_{21} = Z_2 - Z_1$, $Z_{32} = Z_3 - Z_2$, and $Z_{13} = Z_1 - Z_3$. The length of each side is defined as $\{L_1, L_2, L_3\}$, wherein $L_1 = |Z_{21}|$, $L_2 = |Z_{32}|$, and $L_3 = |Z_{13}|$.
- **Triangle type [16]:** Let $\gamma = 4\gamma_1 + 2\gamma_2 + \gamma_3$, where γ_i is the type of minutiae $P_i, i = 1, 2, 3$. If P_i is a bifurcation point, $\gamma_i = 1$, or else $\gamma_i = 0$. We have $\gamma \in \{0, 1, 2, 3, 4, 5, 6, 7\}$.

New Features of Minutiae Triplets. According to our investigation, when the above features are used for fingerprint indexing, a triangle in the top left of a query sample may map to another triangle in the bottom right of a template fingerprint image, which will introduce errors. Enlightened by the feature of triangle type, we define a type of new feature, namely triangle position. Besides, to make full use of the local ridge orientation θ, we introduce another feature, namely orientation differences.

- **Triangle position:** Suppose the fingerprint image is aligned roughly. We divide the segment into 4 equal-sized blocks. Similar to quadrant partition, we let 1 denote the upper right block, 2 denote the upper left block, 3 denote the lower left block and 4 denote the lower right block. Let $\rho_i, i = 1, 2, 3$ be the block type of minutiae $P_i, i = 1, 2, 3$, $\rho_i \in \{1, 2, 3, 4\}$. Define $\varrho = 100\rho_1 + 10\rho_2 + \rho_3$ as the triangle position, then the number of triangle positions is 4^3.
- **Orientation differences:** Let θ_i be the local orientation of minutiae $P_i, i = 1, 2, 3$. We represent orientation difference between each pair of adjacent vertices as $\alpha_i, i = 1, 2, 3$, wherein $\alpha_1 = \theta_2 - \theta_1$, $\alpha_2 = \theta_3 - \theta_2$, and $\alpha_3 = \theta_1 - \theta_3$.

The final feature set of a minutiae triplet is in the form of an eight tuple $\{L_1, L_2, L_3, \gamma, \varrho, \alpha_1, \alpha_2, \alpha_3\}$. Among these features, L_1, L_2, L_3 and γ are the commonly used features of minutiae triplets for indexing [16] [21], and $\alpha_1, \alpha_2, \alpha_3$ and ϱ are the newly designed ones since they are simple, discriminative and easy to obtain even with the singular areas missing (see Section 4).

3.2 Indexing Scheme

Parameters. To reduce the number of false correspondences obtained from querying the indexing space, some parameters on length and orientation difference are introduced.

- **Relative length difference:** Assume the length of each side of a triangle formed by minutiae triplet does not change much in different impressions of the same finger. Let L and $L^{'}$ be L_1, L_2, or L_3 in a query image and a template image, respectively. We constrain $|L - L^{'}| < \delta_L$.
- **Relative rotation:** Assume the orientation difference does not change much in different impressions of the same finger. Let α and $\alpha^{'}$ be α_1, α_2, or α_3 in a query image and a template image, respectively. We constrain $|\alpha - \alpha^{'}| < \delta_O$.

Therefore, δ_L and δ_O are the main parameters for the indexing process.

Registration Process. Since certain distortion in the sides of triangles should be allowed, we adopt quantization to implement feature space clustering. During the registration process, each triangle in a template image is characterized by an eight-tuple vector, which means each fingerprint is viewed as a collection of points distributed in the index space with each point characterizing an eight-dimensional feature vector. Then, we quantize the triangles by the lengths of their three sides. Suppose the maximum side of all the triangles in the database is L_{max}, then the indexing space is partitioned into $(L_{max}/\delta_L)^3$ clusters. Each of the points is assigned to one of the pre-defined clusters based on the quantization rule. This process is repeated for every template fingerprint in the database. Thus, a cluster in the index space will have a list of fingerprint indices that have at least one point assigned to that cluster. Besides, the cluster also stores the remaining features ($\{\gamma, \varrho, \alpha_1, \alpha_2, \alpha_3\}$) in the eight-tuple vector for further processing except for the lengths of each side.

Query Process. During the query process, when a query sample fingerprint q is presented, it is first represented as a set of points with eight-dimensional feature vectors. Next, these points are mapped to individual clusters in the index space. A set of possible matching indices corresponding to a small number of clusters are then determined. After that, each point of the query fingerprint is further compared with the possible matching points in the clusters, and those points that satisfy the following requirements will be chosen:

Table 1. Performance Evaluation on FVC 2000 DB2a – Hit Rate

δ_L	δ_O	*HR* (%)						
		$PR = 1\%$	$PR = 2\%$	$PR = 3\%$	$PR = 4\%$	$PR = 5\%$	$PR = 10\%$	$PR = 20\%$
4	15	85	87	88	89	89	90	92
	30	84	86	87	87	88	90	92
	60	81	85	86	87	87	89	92
5	15	86	88	89	90	**91**	**92**	**94**
	30	85	86	88	89	89	91	**94**
	60	82	84	86	87	88	91	93
6	15	**88**	**90**	**90**	**91**	**91**	**92**	**94**
	30	86	88	89	89	90	**92**	**94**
	60	82	86	88	88	89	91	93

Table 2. Performance Evaluation on FVC 2002 DB1a – Hit Rate

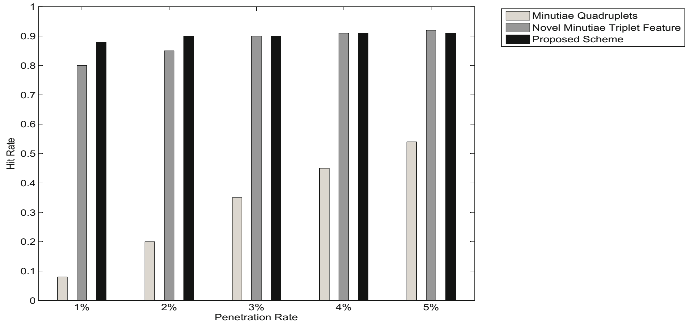
δ_L	δ_O	*HR* (%)						
		$PR = 1\%$	$PR = 2\%$	$PR = 3\%$	$PR = 4\%$	$PR = 5\%$	$PR = 10\%$	$PR = 20\%$
4	15	89	91	92	92	92	93	95
	30	88	90	91	91	92	93	94
	60	84	87	87	88	89	92	93
5	15	90	**93**	**94**	**94**	**94**	**95**	**96**
	30	89	91	91	92	93	**95**	**96**
	60	85	88	88	89	91	92	94
6	15	**91**	92	93	93	**94**	**95**	95
	30	88	90	91	92	92	93	95
	60	84	86	87	88	89	91	94

- The triangle types γ and $\gamma^{'}$ are the same.
- The triangle positions ϱ and $\varrho^{'}$ are the same.
- $|\alpha - \alpha^{'}| < \delta_O$.

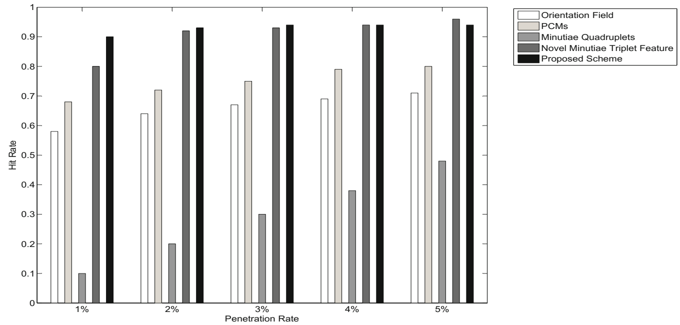
Finally, the qualified indices are sorted in the candidate list by their occurring frequency in descending order.

4 Experimental Results

To evaluate the proposed fingerprint indexing approach, statistical experiments have been carried out on several popular public databases. Section 4.1 illustrates the performance of the proposed scheme on full fingerprint databases including both FVC 2000 DB2a and 2002 DB1a. Section 4.2 demonstrates the performance on partial fingerprint database generated from NIST SD 14. The whole experiment was implemented in Matlab on a workstation PC with the following configurations: Intel(R) Core(TM)i7 3.4GHz, 16GB memory, 64-bit Operating System.



(a) FVC 2000 DB2a



(b) FVC 2002 DB1a

Fig. 1. Performance comparison of different indexing schemes on FVC databases

4.1 Performance on Full Fingerprint Databases

Most of the published techniques for full fingerprint indexing have been evaluated on FVC 2000 DB2a and FVC 2002 DB1a. FVC 2000 DB2a contains 800 fingerprints from 100 subjects (8 impressions per subject) captured using a capacitive fingerprint scanner. FVC 2002 DB1a also contains 800 fingerprints from 100 fingers (8 impressions per finger), but it was captured using a optical fingerprint scanner. In our experiment, we chose the first impression of each subject (100 in total) to form the template database and the rest as the query samples (700 in total) for both FVC 2000 DB2a and FVC 2002 DB1a.

The performance of the fingerprint indexing scheme is evaluated by reporting the hit rate (HR) at certain penetration rates (PR). We tested the proposed indexing scheme using different parameter settings, wherein distortion scale of the triangle sides δ_L is set to be 3, 4, or 5, distortion scale of the orientation difference δ_O is set to be 15, 30, or 60, and the maximum number of candidates to be considered is 20.

Table 1 and Table 2 show the performance of the proposed indexing approach on FVC 2000 DB2a and FVC 2002 DB1a, respectively, wherein the best performance at a certain penetration rate is highlighted in bold. We can see from these

Table 3. Average penetration rate on FVC 2002 DB1a when hit rate is 100%

Minutiae Triplets [16]	38.1%
Low-order Delaunay Triangle [26]	18.1%
Minutiae Quadruplets [27]	11.8%
Novel Minutiae Triplets [21]	9.9%
Proposed Scheme	**3.51%**

Table 4. Average penetration rate on FVC 2000 DB2a when hit rate is 100%

SIFT Features [28]	Minutiae Quadruplets [27]	Novel Minutiae Triplets [21]	Proposed Scheme
91%	26%	22%	**5.24%**

tables that even if the penetration rate is very low (e.g. 1%), the hit rate is high (above 80%). Different choice of δ_L and δ_O results in different performance. For FVC 2000 DB2a, the best choice of δ_L and δ_O is 6 and 15, respectively, and for FVC 2002 DB1a, the best choice of δ_L and δ_O is 5 and 15, respectively.

Fig. 1(a) shows the comparison of the indexing performance of our approach on FVC 2000 DB2a with other methods, including minutiae quadruplets based indexing [27] and indexing with novel minutiae triplet feature [21]. Fig. 1(b) shows the comparison of indexing performance of the proposed method on FVC 2002 DB1a with other techniques based on orientation field [29], PCMs [30], minutiae quadruplets [27] and novel minutiae triplet feature [21]. We can see that the proposed scheme outperforms other state-of-the-art methods, especially when the penetration is very low (1% and 2%).

Table 3 and Table 4 show the results on FVC 2002 DB1a and FVC 2000 DB2a for comparisons with other methods using another measurement, respectively, that is the average penetration rate when the hit rate is 100%. We can see from both tables that the proposed indexing scheme can achieve much better performance than other method evaluated using the same measurement.

4.2 Performance on Partial Fingerprint Database

Related works on partial fingerprint indexing have used NIST special database 27 (SD 27) as the query image set, however, feature extraction in NIST SD 27 is manually done [20] [22] [21] and cannot be operated automatically. Since the objective of this experiment is fingerprint indexing on partial fingerprints, we use another public database NIST special database 14 (SD 14) [31] in this test.

NIST SD 14 consists of 54000 ink-rolled prints scanned from fingerprint cards. There are two impressions recorded for each finger, namely, the F (First) prints ranging from F00001 to F27000 and the S (Second) prints ranging from S00001 to S27000. In our experiments, we chose the last 2000 F prints to constitute the template database and the last 2000 S images as the query samples. Before indexing, we segmented both the template and sample images to remove peripheral regions and make the remainder frame lie in a north-south direction as much as possible. The image size after segmentation is 480×512 pixels.

(a) Full fingerprint image

(b) Partial fingerprint image

Fig. 2. A typical partial fingerprint image in our experiment and its corresponding full image

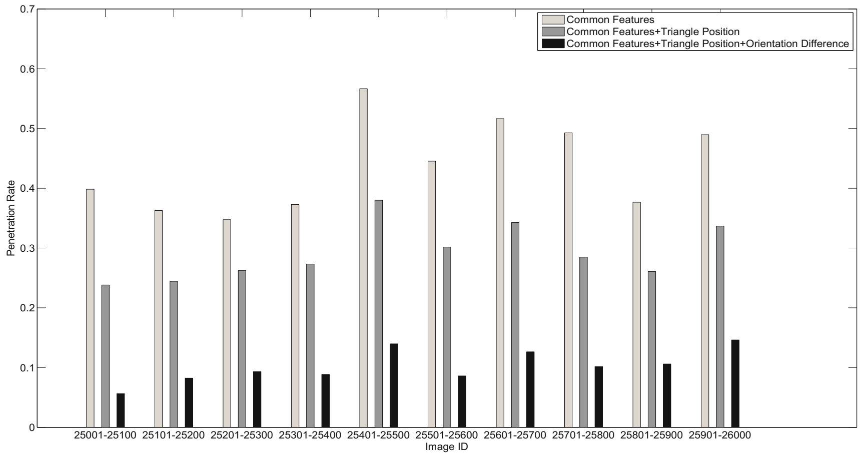


Fig. 3. Performance improvement of using new features incrementally on NIST SD 14

For each sample image, we used a routine of NIST Biometric Image Software [32], namely Mindtct, to obtain a quality map marking reliability of local fingerprint image areas at different levels. Then, we extracted an image foreground with the highest quality level and produced a partial fingerprint segment by keeping only the high quality areas. Figure 2 shows a typical example of such partial fingerprint images generated in the test and its corresponding full fingerprint. Therefore, partial fingerprints generated in our experiment do not contain any singularity, and even singularity regions are usually removed. In this way, we can generate a sample image set composed of partial fingerprints.

Fig. 3 illustrates the performance improvement of fingerprint indexing on NIST SD 14 when the new features are used incrementally. In this experiment, the last 2000 F prints ($F25001 \sim F27000$) constitute the template database and 1000 S prints ($S25001 \sim S26000$) are divided into 10 groups as the query samples. Parameters δ_L and δ_O were set to be 4 and 15, respectively. As is

Table 5. Performance Evaluation on NIST SD 14 – Penetration Rate

δ_L	Penetration Rate (%)		
	$\delta_O = 15$	$\delta_O = 30$	$\delta_O = 60$
4	15	13.28	14.63
5	12.25	11.48	13.29
6	10.25	10.12	10.27

shown in Fig. 3, the penetration rate decreased by at least 1/3 when the triangle position was used as an extended feature, and the the penetration rate further decreased by at least 1/2 when the orientation difference was used as another extended feature.

Table 5 is the performance of the indexing approach on NIST SD 14 with different choice of δ_L and δ_O. In this experiment, the last 2000 F prints ($F25001 \sim F27000$) form the template database and the last 2000 S prints ($S25001 \sim S27000$) are used as the query samples. We can see that when δ_L and δ_O are 6 and 30 respectively, the performance is the best (nearly 10%) in this test.

As mentioned before, existing techniques on partial fingerprint indexing approaches were evaluated on NIST SD 27, which need human involvement to extract features. However, the partial sample images used in our experiments are generated from full fingerprint images by erosion and are not used elsewhere, so there is no related comparable work. According to the indexing performance of other methods on full fingerprint databases in Table 3 and Table 4, we can see that 10% penetration rate is fairly good for partial fingerprint indexing.

5 Conclusion

In this paper, we proposed an indexing scheme by combining some common features with some novel features of minutiae triplets for fingerprint indexing. Experimental evaluation on full fingerprint databases FVC 2000 DB2a and 2002 DB1a show that the proposed indexing approach can achieve better performance than state-of-the-art methods. We also investigated the performance improvement on partial fingerprint database generated from NIST SD 14 when these new features were added to the feature space incrementally. The experimental results show that the performance is improved significantly after new features are considered, and can be comparable to that on full fingerprint indexing under certain parameter settings.

Acknowledgment. The work in this paper was supported by Australian Research Council (ARC) Linkage Project LP120100595.

References

1. Xi, K., Hu, J., Han, F.: Mobile device access control: an improved correlation based face authentication scheme and its java me application. Concurrency and Computation: Practice and Experience 24(10), 1066–1085 (2012)

2. Xi, K., Tang, Y., Hu, J.: Correlation keystroke verification scheme for user access control in cloud computing environment. Comput. J. 54(10), 1632–1644 (2011)
3. Sufi, F., Khalil, I.: Faster person identification using compressed ecg in time critical wireless telecardiology applications. J. Network and Computer Applications 34(1), 282–293 (2011)
4. Sufi, F., Khalil, I.: An automated patient authentication system for remote telecardiology. In: International Conference on Intelligent Sensors, Sensor Networks and Information Processing, ISSNIP 2008, pp. 279–284 (December 2008)
5. Sufi, F., Khalil, I., Hu, J.: Ecg-based authentication. In: Stavroulakis, P., Stamp, M. (eds.) Handbook of Information and Communication Security, pp. 309–331. Springer, Heidelberg (2010)
6. Ahmad, T., Hu, J., Wang, S.: Pair-polar coordinate-based cancelable fingerprint templates. Pattern Recogn. 44(10-11), 2555–2564 (2011)
7. Wang, S., Hu, J.: Alignment-free cancelable fingerprint template design: A densely infinite-to-one mapping (ditom) approach. Pattern Recogn. 45(12), 4129–4137 (2012)
8. Xi, K., Ahmad, T., Han, F., Hu, J.: A fingerprint based bio-cryptographic security protocol designed for client/server authentication in mobile computing environment. Journal of Security and Communication Networks 4(5), 487–499 (2011)
9. Xi, K., Hu, J.: Introduction to Bio-cryptography. In: Handbook of Information and Communication Security. Springer (2010)
10. Yang, W., Hu, J., Wang, S., Stojmenovic, M.: An alignment-free fingerprint biocryptosystem based on modified voronoi neighbor structures. Pattern Recognition 47(3), 1309–1320 (2014)
11. Wang, S., Hu, J.: Design of alignment-free cancelable fingerprint templates via curtailed circular convolution. Pattern Recognition 47(3), 1321–1329 (2014)
12. Yang, W., Hu, J., Wang, S., Yang, J.: Cancelable fingerprint templates with delaunay triangle-based local structures. In: Wang, G., Ray, I., Feng, D., Rajarajan, M. (eds.) CSS 2013. LNCS, vol. 8300, pp. 81–91. Springer, Heidelberg (2013)
13. Yang, W., Hu, J., Wang, S.: A finger-vein based cancellable bio-cryptosystem. In: Lopez, J., Huang, X., Sandhu, R. (eds.) NSS 2013. LNCS, vol. 7873, pp. 784–790. Springer, Heidelberg (2013)
14. Lumini, A., Maio, D., Maltoni, D.: Continuous versus exclusive classification for fingerprint retrieval. Pattern Recogn. Lett. 18(10), 1027–1034 (1997)
15. Cappelli, R., Lumini, A., Maio, D., Maltoni, D.: Fingerprint classification by directional image partitioning. IEEE Trans. Pattern Anal. Mach. Intell. 21(5), 402–421 (1999)
16. Bhanu, B., Tan, X.: Fingerprint indexing based on novel features of minutiae triplets. IEEE Trans. Pattern Anal. Mach. Intell. 25(5), 616–622 (2003)
17. de, J.B., Bazen, A.M., Gerez, S.H.: Indexing fingerprint databases based on multiple features. In: Proceedings SAFE, ProRISC, SeSens 2001, Utrecht, The Netherlands, STW, pp. 300–306 (November 2001)
18. Wang, Y., Hu, J., Phillips, D.: A fingerprint orientation model based on 2D fourier expansion (fomfe) and its application to singular-point detection and fingerprint indexing. IEEE Trans. Pattern Anal. Mach. Intell. 29(4), 573–585 (2007)
19. Feng, J., Jain, A.K.: Filtering large fingerprint database for latent matching. In: Proc. Int. Conf. on Pattern Recognition (ICPR 2008), pp. 1–4 (2008)
20. Jain, A.K., Feng, J.: Latent fingerprint matching. IEEE Transactions on Pattern Analysis and Machine Intelligence 33(1), 88–100 (2011)

21. Yuan, B., Su, F., Cai, A.: Fingerprint retrieval approach based on novel minutiae triplet features. In: 2012 IEEE Fifth International Conference on Biometrics: Theory, Applications and Systems (BTAS), pp. 170–175 (2012)
22. Paulino, A.A., Liu, E., Cao, K., Jain, A.K.: Latent fingerprint indexing: Fusion of level 1 and level 2 features. In: Biometrics: Theory, Applications and Systems, Washington, D.C. (2013)
23. Wang, Y., Hu, J.: Global ridge orientation modeling for partial fingerprint identification. IEEE Trans. Pattern Anal. Mach. Intell. 33(1), 72–87 (2011)
24. VeriFinger: Verifinger sdk (2013), `http://www.neurotechnology.com/verifinger.html`
25. Germain, R., Califano, A., Colville, S.: Fingerprint matching using transformation parameter clustering. IEEE Computational Science Engineering 4(4), 42–49 (1997)
26. Liang, X., Bishnu, A., Asano, T.: A robust fingerprint indexing scheme using minutia neighborhood structure and low-order delaunay triangles. IEEE Transactions on Information Forensics and Security 2(4), 721–733 (2007)
27. Iloanusi, O., Gyaourova, A., Ross, A.: Indexing fingerprints using minutiae quadruplets. In: 2011 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 127–133 (2011)
28. Shuai, X., Zhang, C., Hao, P.: Fingerprint indexing based on composite set of reduced sift features. In: 19th International Conference on Pattern Recognition, ICPR 2008, pp. 1–4 (2008)
29. Jiang, X., Liu, M., Kot, A.: Fingerprint retrieval for identification. IEEE Transactions on Information Forensics and Security 1(4), 532–542 (2006)
30. Liu, M., Yap, P.T.: Invariant representation of orientation fields for fingerprint indexing. Pattern Recogn. 45(7), 2532–2542 (2012)
31. SD14: Nist special database 14 (2013), `http://www.nist.gov/srd/nistsd14.cfm`
32. NBIS: Nist biometric image software (2013), `http://www.nist.gov/itl/iad/ig/nbis.cfm`