A Novel Fingerprint Matching Algorithm Using Ridge Curvature Feature*

Peng Li, Xin Yang, Qi Su, Yangyang Zhang, and Jie Tian

Institute of Automation, Chinese Academy of Sciences, Beijing 100190 China
tian@ieee.org, jie.tian@ia.ac.cn
http://www.fingerpass.net

Abstract. Fingerprint matching based on solely minutiae feature ignore the abundant ridge information in fingerprint images. We propose a novel fingerprint matching algorithm which integrates minutiae feature with ridge curvature map(RCM). The RCM is approximated by a polynomial model which is computed by Least Square(LS) method. In the matching stage, phase-only correlation matching method is employed to match two RCMs. Then sum fusion rule is selected to combine the minutiae matching score and the RCM matching score. Experiments conducted on FVC2002 and FVC2004 databases show that proposed algorithm can obtain more promising performance than solely minutiae-based algorithm and several other multi-feature fusion algorithms.

Keywords: Fingerprint matching, Polynomial model, RCM, Phase-only correlation, Sum rule.

1 Introduction

Most classical Automatic Fingerprint Identification Systems(AFIS) employ minutiae information, including the position, orientation and type, as the distinctive feature. However, using solely minutiae, which belongs to micro feature in fingerprint representation, misses the abundant macro features, for example orientation field, ridge frequency, ridge curvature and so on. Pankanti et al.[1] analyzed and computed the performance's theoretical upper bound of solely minutiae-based fingerprint recognition systems. So more and more researchers begin to focus on exploring the supplementary features for minutiae feature in AFIS[2][3][4].

Fingerprint pattern is constituted by a group of ridge and valley curves. The ridge and valley have the same trend in nature, so only the ridge curve is taken into consideration in this paper. Ridge curvature is a kind of important macro

* This paper is supported by the Project of National Natural Science Foundation of China under Grant No. 60875018 and 60621001, National High Technology Research and Development Program of China under Grant No. 2008AA01Z411, Chinese Academy of Sciences Hundred Talents Program, Beijing Natural Science Foundation under Grant No. 4091004.

M. Tistarelli and M.S. Nixon (Eds.): ICB 2009, LNCS 5558, pp. 607–616, 2009.

feature in fingerprint representation and has been widely used in singular point detection[5][6], classification[7], alignment[8] and matching[9] of fingerprints. Liu et al.[5] and Koo et al.[6] compute fingerprint's ridge curvature using the orientation field information and multi-resolution method. They define the ridge point with the maximum curvature value as the reference point of fingerprint. Wang et al.[7] apply the fingerprint ridge curvature into classification with fixed class number. In their algorithm, 25 sampling points are selected and the corresponding curvature values are computed. Then the curvature values are used for composing a 25-dimension feature vector, which are input into the Radial Basis Function Neural Networks(RBFNN) to obtain the classification result. Yager and Amin[8] proposed a two-stage fingerprint alignment method, in the first stage of which the ridge curvature is employed as one of the features. The other two are fingerprint orientation field and ridge frequency and all of them belong to macro feature. Saleh and Adhami transformed all the entire fingerprint feature, including minutiae, ridge curvature, relative position etc., into the so-called Angel-Curvature domain. In their method, the curvature of some ridge points are employed.

In this paper, we explore the fingerprint ridge curvature feature for supplementing the minutiae-based matching algorithm, which is first proposed to our knowledge. The fingerprint ridge curvature is approximated using a polynomial model, whose coefficients are stored, with little extra memory cost(typically 50 bytes), for generating the whole curvature map in the matching phase. We compute the matching score of curvature maps by using correlation matching method based on Fourier-Mellin Transformation[10]. The curvature maps' matching score is fused with the minutiae matching score using sum rule in the decision level.

2 Fingerprint Ridge Curvature Map

Fingerprint ridge contains abundant information representing the flow direction and the orientation's change amount to a length-fixed ridge segment. Introducing ridge curvature into minutiae-based matching algorithm can help improve the accuracy of AFIS. We compute the curvature values of every pixel in the fingerprint image and display all of them using density in another gray scale image, which we call Ridge Curvature Map(RCM). The RCM we propose is obtained by two steps: coarse curvature value extraction and polynomial model fitting.

2.1 Coarse Curvature Value Computing

Ridge curvature is defined as the orientation's change amount along an length-fixed arc and commonly computed by tracing the thinned ridge. Yager et al.[8] represented each thinned ridge with a cubic B-Spline to approximately calculate the change rate on some pixel's tangent direction. And then a Gaussian low-pass filter is used to obtain the whole RCM. The disadvantage of Yager's method lies on the difficulty of computing the minutiae's curvature and coarse curvature value's separative distribution due to 1-pixel width ridge. In our method, the

ridge contour line, rather than thinned ridge, is used to compute the coarse curvature value. And we define the curvature as the orientation's difference along a arc with fixed-number pixels. Thus the effective coarse curvature data increase by about one time and the minutiae's curvature can also be computed correctly.

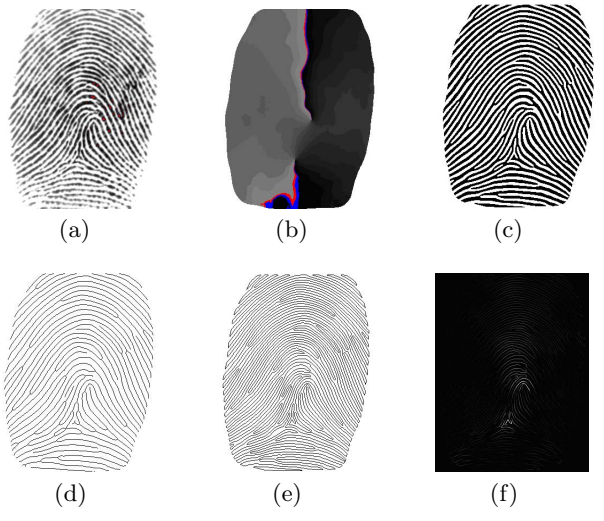


Fig. 1. Intermediate images in coarse ridge curvature value computing procedure((a) is from FVC2002 DB1). (a) input fingerprint image; (b) orientation image; (c) binary image; (d) thinned image; (e) contour image; (f) contour curvature image.

The coarse curvature is calculated by the following steps:

1) Gradient-based orientation filed computation(See Fig.1(b));
2) Image enhancement and binarization. Here we adopt a structural anisotropic filter[11] to enhance the input fingerprint image I_{IN}. The filter kernel is defined in space domain as:

$$h(x, y, \psi) = c_1 + c_2 \cdot exp(-\frac{x_\psi^2}{2\sigma_1^2} - \frac{y_\psi^2}{2\sigma_2^2}) \cdot \frac{\sin f \cdot x_\psi}{f \cdot x_\psi} \tag{1}$$

where,

$$\begin{aligned} x_\psi &= x \cos \psi + y \sin \psi \\ y_\psi &= -x \sin \psi + y \cos \psi. \end{aligned} \tag{2}$$

The parameters c_1, c_2, σ_1^2 and σ_2^2 are all selected empirically as $c_1 = -1$, $c_2 = 2$, $\sigma_1^2 = 4$ and $\sigma_2^2 = 2$. And the parameter f is defined by the fingerprint ridge frequency. After obtaining the enhancement image I_E, global threshold based binarization method is performed and the threshold equals 128. The binarization image is shown in Fig.1(c).

3) Extracting fingerprint ridge contour image using Sobel operator. The resultant image I_C is shown in Fig.1(e). It doubles the curvature information amount compare to Fig.1(d), which shows the thinned ridge image I_T. The spacing between lines of I_T is much larger than I_C.
4) Coarse curvature value computation. We approximately compute the ridge contour line pixel's curvature using the orientation difference of a arc with fixed-number pixels. As is shown in Fig.2, a contour pixel p_0's curvature is decided by the orientation difference of pixel p_1 and p_2, i.e. $c(i_0, j_0) = |o(i_1, j_1) - o(i_2, j_2)|$. For the minutiae p_3, the same computation method is applied to obtain the curvature, $c(i_3, j_3) = |o(i_4, j_4) - o(i_5, j_5)|$. In Fig.2, the fixed pixel length $\triangle L$ is 10 pixels. The resultant coarse curvature map is shown in Fig.1(f).

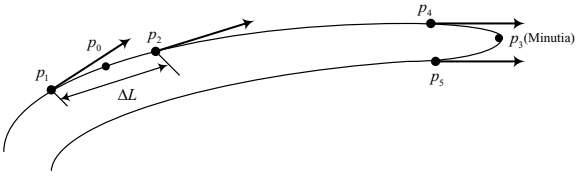


Fig. 2. Sketch map of a ridge contour line and the ridge curvature computation method

2.2 RCM Generation Based on Polynomial Model

The coarse curvature data of all the contour pixels are huge and very memory-costing to store them, which is not necessary at all because there exists a mass of data redundancy. In fingerprint feature analysis and representation, polynomial model is an effective tool to fit the macro feature[12]. We employ a bivariate polynomial function to model the fingerprint ridge curvature. The Least Square(LS) algorithm is used to compute the model's coefficients. The polynomial model is written as Equ.3.

$$\tilde{p}(x, y) = \sum_{i=0}^{k} \sum_{j=0}^{k} a_{ij} x^i y^j \tag{3}$$

where k is the polynomial degree. Let $A = \{a_{ij}\}$ denotes the coefficient matrix, $i = 1, 2, \cdots, k, j = 1, 2, \cdots, k$. A can be obtained by minimizing the square error between the polynomial value and the coarse curvature computed from the given fingerprint image, i.e., minimizing the following function:

$$\varepsilon = \sum_{i=1}^{n} [\tilde{p}(x_i, y_i) - p(x_i, y_i)]^2 \tag{4}$$

where n is the number of sampling points and $p(x_i, y_i)$ denotes the coarse curvature value of the i-th sampling point. The element $a_{i,j}$ of A can be computed by solving the following equation:

$$\frac{\partial \varepsilon}{\partial a_{i,j}} = 0, i \in [1, k], j \in [1, k] \tag{5}$$

In order to avoid ill solution, Singular Value Decomposition(SVD) method is used to solve Equ.5. In addition, k in Equ.3 is selected as 4 according to the tradeoff between the computing cost and the accuracy.

In the end, the 25 coefficients($k = 4$) of A are stored into the fingerprint template to generate RCM in the matching phase by using Equ.3. The total memory cost is $2bytes \times 25 = 50bytes$. Fig.3 compares the RCMs generated by polynomial model and by Yager's method[8], respectively. It shows that the polynomial model can generate smoother RCMs.

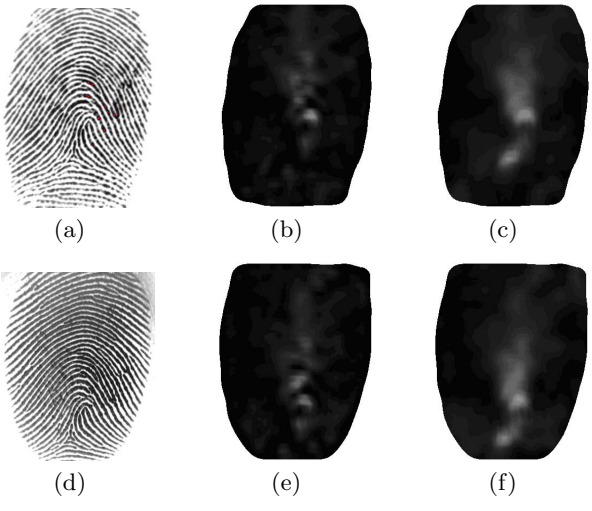


Fig. 3. RCMs. (a)&(d) input fingerprint images(from FVC2002 DB1); (b)&(e) RCMs extracted by method[8]; (c)&(f) RCMs extracted by proposed method.

3 Fingerprint Matching Fusing Ridge Curvature and Minutiae

In this section, we will clarify the matching scheme combining RCM and minutiae information. Fourier-Mellin Transform(FMT) based correlation matching algorithm is utilized for the RCM matching. And its matching score will be combined with the minutiae matching algorithm's with sum rule in the decision level.

3.1 RCM Matching Algorithm

An important method in fingerprint matching is phase-only correlation matching [13], which employs gray-scale images to compute the affine transform's translation parameter and the correlation matching score of two images. The translation parameter and matching score can be computed in either space domain or frequency domain. However, due to large time consuming in space domain, correlation matching is commonly performed using Fourier Transform in frequency domain.

In this paper, we compute the correlation matching score by performing Fourier-Mellin Transform on RCMs. Let f_1 and f_2 denote two RCMs in space domain and their corresponding FMT maps are F_1 and F_2 respectively. Let the relation between f_1 and f_2 be written as(assuming two images have the same size here):

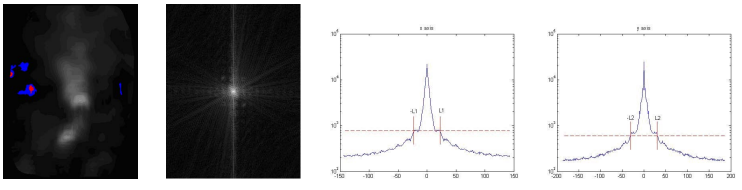
$$f_2(x, y) = f_1[(x\cos\theta_0 + y\sin\theta_0) - x_0, (-x\sin\theta_0 + y\cos\theta_0) - y_0] \quad (6)$$

where (x_0, y_0) and θ_0 denote the translation and rotation parameter, respectively. Correspondingly, the relation between F_1 and F_2 can be written as:

$$F_2(u, v) = F_1(u\cos\theta_0 + v\sin\theta_0, -u\sin\theta_0 + v\cos\theta_0) \cdot exp(-j2\pi(ux_0 + vy_0)) \quad (7)$$

Equ.7 shows that the translation parameter (x_0, y_0) and the rotation parameter θ_0 can be computed separately. According to the property of Fourier Transform, (x_0, y_0) can be indicated with cross-power spectrum of f_1 and f_2, as shown in Equ.8.

$$\hat{R}_{F_1F_2}(u, v) = \frac{F_1(u, v) \cdot F_2^*(u, v)}{|F_1(u, v) \cdot F_2^*(u, v)|} = e^{-j2\pi(ux_0 + vy_0)} \quad (8)$$



(a) (b) (c) (d)

Fig. 4. RCMs in band-limited frequency domain. (a) A fingerprint RCM; (b) Amplitude spectrum of (a); (c) Band limitation in x-axis projection; (d) Band limitation in y-axis projection.

Inverse Fourier Mellin Transform(IFMT) is performed on Equ.8 to obtain a correlation image $I_{cor} = \hat{r}_{f_1 f_2}(x, y)$ in space domain. If f_1 and f_2 are from the same finger, in I_{cor} there will be an aculeated single peak value, similar to monoimpulse, which denotes the position of (x_0, y_0). Otherwise, the peak value's distribution is relatively gentle. This is so-called Phase-Only Correlation(POC) matching. In order to sharpen the peak value, we utilize Band-limited POC(LDPOC)[13] to compute the phase of the cross power spectrum density of f_1 and f_2. LDPOC selects the mean of the power spectrum as the threshold, based on which domain energy region is decided. Fig.4 illuminates the decision of main energy region. After LDPOC, the peak value $v_p (0 < v_p < 1)$ could be viewed as the matching score S_c. If there are several peaks, the sum of them will be S_c.

3.2 Minutiae Matching Algorithm and Fusion Strategy

We use similarity histogram based minutiae matching algorithm, proposed by Zhang et al.[14], to obtain the minutiae matching score. Let a minutiae $m_i =$

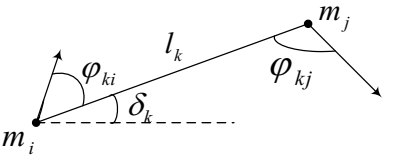


Fig. 5. Structure of minutiae pair

$(x_i, y_i, \theta_i, t_i)$, where (x_i, y_i) denotes the position, θ_i the orientation and t_i the type. All the minutiae in a fingerprint constitute the template $M = \{m_i | i \in [1, n]\}$($n$ is the minutiae's total number). The basic local structure of minutiae pair, illuminated as Fig.5, is utilized in minutiae matching algorithm.

We search the minutiae template to find all the basic structures, which is subject to Equ.9.

$$\begin{aligned} &l(m_i, m_j) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \\ &d_{min} < l(m_i, m_j) < d_{max}, \end{aligned} \tag{9}$$

where $l(m_i, m_j)$ is the distance between m_i and m_j; d_{min} and d_{max} are two distance thresholds.

Given the template fingerprint T and the query fingerprint Q, we extract two local structure sets $P^T = \{p_k^t(l_t^t, \delta_k^t, \phi_{ki}^t, \phi_{kj}^t) | k \in [1, N_1]\}$ and $P^Q = \{p_k^q(l_t^q, \delta_k^q, \phi_{ki}^q, \phi_{kj}^q) | k \in [1, N_2]\}$, where N_1 and N_2 both are the number of local structure. For $p_m^t \in P^T$ and $p_n^q \in P^Q$, the similarity between them is defined as:

$$\begin{aligned} s(m, n) &= \begin{cases} 1 - \frac{\Delta p_{mn}}{c_0} & \Delta p_{mn} < c_0 \\ 0 & otherwise \end{cases} \\ \Delta p_{mn} &= \eta_1 \cdot |l_m^t - l_n^q| + \eta_2 \cdot |\delta_m^t - \delta_n^q| + \eta_3 \cdot |\varphi_{mi}^t - \varphi_{mi}^q| + \eta_4 \cdot |\varphi_{ni}^t - \varphi_{ni}^q|, \end{aligned} \tag{10}$$

where, empirically the constants are selected as $c_0 = 100, \eta_1 = 0.2, \eta_2 = 0.2, \eta_3 = 0.3, \eta_4 = 0.3$.

Before matching is performed, the alignment procedure and minutiae structure pairing between the template and the query need to be conducted first[14]. For arbitrary $p_m^T \in P^T$ and $p_n^Q \in P^Q$, all the local similarity $s(m, n)$ are computed and the similarity histogram[14] is plotted to decide the reference minutiae structure and the corresponding relation of all local minutiae structures. Afterwards, all the corresponding minutiae structure's similarity degrees are computed and accumulated to get the overall similarity score S_{TQ}(where k is the total number of basic structures).

$$S_{TQ} = \sum_{m,n} s(m, n)/k \tag{11}$$

Fusing different features and their corresponding classifiers can obtain better matching performance than using single feature. We will fuse the ridge curvature matching score S_c and the minutiae matching score S_{TQ} in the decision level, where several fusion strategies can be considered, including max rule, min rule, product rule, sum rule. Experiments in section 4 show the sum rule performs best among all of them.

4 Experimental Results and Analysis

In this section, we select fingerprint data sets from FVC2002 and FVC2004 to test the performance of proposed fusion matching algorithm. Both FVC2002 and FVC2004 include 4 data sets(3 real fingerprint sets from different sensors and 1 synthetic set), which we denote as DB1-DB4. Each of these 8 sets consists 800 fingerprints, i.e., 8 samples for each of 100 fingers, respectively. All the real samples are collected under natural condition and the synthetic ones simulate the fingerprints' natural quality and distribution. For each set, we conduct $\frac{8\times7}{2} \times 100 = 2800$ genuine matches and $\frac{100\times99}{2} = 4950$ imposter matches. The indicator EER(Equal Error Rate) is used to evaluate the performance and the test platform is a P4 2.4G PC with 512M RAM. The algorithm is implemented with C and Matlab computing engine hybrid programming.

Table 1. Accuracy comparison of fusion strategies on FVC2002

	EER_c	EER_{TQ}	EER_{max}	EER_{min}	$EER_{product}$	EER_{sum}
DB1	9.7%	2.1%	11.2%	1.4%	1.3%	1.1%
DB2	8.7%	1.3%	7.8%	0.69%	0.64%	0.59%
DB3	15%	4.2%	16%	3.9%	3.6%	3.3%
DB4	12%	1.7%	13.5%	1.5%	1.3%	1%

4.1 Fusion Rule Selection

Using multi-classifier correlation computation method proposed by Kuncheva et al.[15], we calculate the correlation coefficient between RCM matching and minutiae matching on DB1 and DB2 of FVC2002. Respectively, they are 0.18 and 0.19, which are lower than the ones(0.28 and 0.24) reported by[12]. This shows the independence between RCM matching and minutiae matching is higher than the two classifiers used in[12].

We test the performance of the four fusion rules(maximum rule, minimum rule, product rule and sum rule) on 4 sets from FVC2002. Table 1 shows the results, which can tell that sum rule performs best among all the fusion rules. So we select sum rule to test the final performance of proposed fusing matching method.

4.2 Results on Data Sets from FVC2002

Table 2 shows the performance comparison of proposed fusing algorithm and solely minutiae-based matching method on data sets from FVC2002. Combining modeled RCM with minutiae leads average EER decrease by 0.83% on 4 data sets. Due to using of C and Matlab computing engine hybrid programming, average enroll time and match time consumed by proposed method are somewhat longer than solely minutiae-based method. However, if the entire algorithm is transplanted to an AFIS and implemented with C, the time consuming can be much shorter.

Table 2. Performance comparison of fusing method and minutiae-base method on FVC2002

	Image description	Algorithm	EER	Enroll time	Match time
DB1	388 × 374,500dpi	Minutiae	2.1%	0.09s	0.019s
		RCM+Minutiae	1.1%	0.7s	0.26s
DB2	296 × 560,569dpi	Minutiae	1.3%	0.1s	0.019s
		RCM+Minutiae	0.59%	0.9s	0.27s
DB3	300 × 300,500dpi	Minutiae	4.2%	0.07s	0.019s
		RCM+Minutiae	3.3%	0.5s	0.26s
DB4	288 × 384,500dpi	Minutiae	1.7%	0.08s	0.019s
		RCM+Minutiae	1.0%	0.5s	0.26s

Table 3. Accuracy comparison of different matching algorithms on FVC2002

	DB1	DB2	DB3	DB4
Zhang et al.[14]	2.6%	1.32%	4.55%	3.13%
Yager et al.[8]	2.6%	3.0%	–	–
Wan and Zhou[12]	4%	3.5%	–	–
Proposed method	1.1%	0.59%	3.3%	1.0%

We also compare proposed fusing matching method with other minutiae-based or fusing method on data sets from FVC2002. The results are shown in Table 3 and our proposed method performs best.

4.3 Results on Data Sets from FVC2004

Data sets from FVC2004 are more difficult to deal with than ones from FVC2002. We select FVC2004 DB1 and DB2 to test proposed method's performance on large distortion or low quality data set. Fingerprints of DB1 are captured by CrossMatch V300 optical sensor, and the image size is 640 × 480 with 500dpi. There exists relatively large distortion in images of DB1. Fingerprints of DB2 are captured by URU4000 optical sensor, and the image size is 328×364 with 500dpi. Much images in DB2 fall into low quality images and difficult to deal with. Respectively, proposed fusing matching method obtains EER 7.3% and 5.9% on DB1 and DB2, compared with 7.6% and 6.4% obtained by solely minutiae-based method. The results show that performance improvement on FVC2004 data sets is weaker than FVC2002 data sets. We can conclude that proposed fusing method is not able to deal with large distortion and low quality fingerprint images effectively, i.e., RCM is easy influenced by the distortion and image quality.

5 Conclusion

Fingerprint matching algorithm combining minutiae feature and ridge curvature feature is proposed in this paper. More refined ridge curvature is extracted

from fingerprint ridge contour map. Then a polynomial model to approximate global ridge curvature is used to obtain resultant RCMs, which are matched by phase-only correlation method in the matching phase. Experiments conducted on FVC2002 and FVC2004 databases show the promising performance of proposed fusing matching algorithm.

References

1. Pankanti, S., Prabhakar, S., Jain, A.K.: On the Individuality of Fingerprints. IEEE Trans. Pattern Anal. Mach. Intell. 24(8), 1010–1025 (2002)
2. He, Y., Tian, J., Luo, X., Zhang, T.: Image Enhancement and Minutiae Matching in Fingerprint Verification. Pattern Recognition Letters 24(9), 1349–1360 (2003)
3. Tico, M., Kuosmanen, P.: Fingerprint Matching Using an Orientation-Based Minutia Descriptor. IEEE Trans. Pattern Anal. Mach. Intell. 25(8), 1009–1014 (2003)
4. Jain, A.K., Prabhakar, S., Lin, H., Pankanti, S.: Filterbank-based Fingerprint Matching. IEEE Trans. Image Processing 9(5), 846–859 (2000)
5. Liu, M., Jiang, X., Kot, A.C.: Fingerprint Reference-point Detection. EURASIP Journal on Applied Signal Processing 2005(4), 498–509 (2005)
6. Koo, W.M., Kot, A.: Curvature-Based Singular Points Detection. In: Proceedings of International Conference on Audio-and Video-Based Biometric Person Authentication, pp. 229–234. Springer, Heidelberg (2001)
7. Wang, X., Li, J., Niu, N.: Fingerprint Classification Based on Curvature Sampling and RBF Neural Networks. In: Wang, J., Liao, X.-F., Yi, Z. (eds.) ISNN 2005. LNCS, vol. 3497, pp. 171–176. Springer, Heidelberg (2005)
8. Yager, N., Amin, A.: Fingerprint Alignment Using a Two Stage Optimization. Pattern Recognition Letters 27(5), 317–324 (2006)
9. Saleh, A.A., Adhami, R.R.: Curvature-based Matching Approach for Automatic Fingerprint Identification. In: Proceedings of the 33rd Southeastern Symposium on System Theory, pp. 171–175 (2001)
10. Chen, Q., Defrise, M., Deconinck, F.: Symmetric Phase-only Matched Filtering of Fourier-Mellin Transforms for Image Registration and Recognition. IEEE Trans. Pattern Anal. Mach. Intell. 16(12), 1156–1168 (1994)
11. Chen, X., Tian, J., Zhang, Y., Yang, X.: Enhancement of Low Quality Fingerprints Based on Anisotropic filter. In: Zhang, D., Jain, A.K. (eds.) ICB 2006. LNCS, vol. 3832, pp. 302–308. Springer, Heidelberg (2006)
12. Wan, W., Zhou, J.: Fingerprint Recognition Using Model-based Density Map. IEEE Transactions on Image Processing 15(6), 1690–1696 (2006)
13. Ito, K., Nakajima, K., Kobayashi, K., Aoki, T., Higuchi, T.: A Fingerprint Matching Algorithm Using Phase-Only Correlation. IEICE Trans. Fundamentals E87-A(3) (2004)
14. Zhang, T., Tian, J., He, Y., Cheng, J., Yang, X.: Fingerprint Alignment Using Similarity Histogram. In: Proceedings of International Conference on Audio- and Video-Based Biometric Person Authentication, pp. 854–861. Springer, Heidelberg (2003)
15. Kuncheva, L.I., Whitaker, C.J.: Measures of Diversity in Classifiers Ensembles and Their Relationship with the Ensemble Accuracy. Mach. Learn. 51, 182–207 (2003)