A Robust Algorithm for Fingerprint Singular Point Detection and Image Reference Direction Determination Based on the Analysis of Curvature Map

Jin Qi, Mei Xie
Electrical Engineering Department
University of Electronic and Science Technology of China
Chengdu, China
jqichina@hotmail.com; qijin@ee.uestc.edu.cn; xiemei@ee.uestc.edu.cn

***Abstract*—A novel fingerprint singular point (SP) detection algorithm is proposed in this paper based on the analysis of curvature maps computed from fingerprint images. And a fingerprint image reference direction (FIRD) is defined and computed using curvature map and orientation field. Because each fingerprint image owns one and only on FIRD, FIRD can be used as the reference frame orientation when performing fingerprint match. A series of experiments are done on a public domain collection of fingerprint images, Set B(training database) of DB3 in FVC2002, to evaluate our method proposed here. The results demonstrate the good accuracy and efficiency of the algorithm in this paper.**

***Index Terms*—Fingerprint Singular Point; Fingerprint Reference Direction; Curvature.**

I. Introduction

Biometric recognition is an emerging subject in application like high-security wireless access and secure transactions across computer networks. Among many available biometric features like fingerprint, face, iris, voice, hand geometry, retina, etc., fingerprints are easy to use and provide relatively good performance. Fingerprint recognition has been extensively studied by private and academic institutions.

In this paper, our focus is on both the detection of fingerprint singular points, which play an important role in fingerprint classification and match, as well as the definition, computation of FIRD. Reliable FIRD can be used as the reference registration when performing fingerprint matching operation. There exist many singular point detection algorithms in the literature, including Poincare Index based techniques [3], [4], [5], directional field coherence based algorithms [2], [7]. The methods based on Poincare Index value are time-consuming and the latter based on coherence analysis is sensitive to noise from various sources. We propose to use fingerprint curvature map to detect the positions of SP and give the definition and computation method for finding FIRD. Curvature-based SP detection algorithm is also proposed by W.M. Koo and A. Kot [8]. But in [8], fingerprint ridge curvature has not been computed reasonably, the author seems to use the local orientation variance to take place the desired curvature. Furthermore, the method [8] is sensitive to noise such that to find many spurious candidate singular points. Our curvature computation here is simple and effective. Different from the approach [8], this paper contains the definition and computation of FIRD.

This paper is organized as follows: Curvature computation algorithm is proposed in Section 2; In Section3, we introduce the SP detection algorithm based on our curvature map; FIRD is defined and computed in Section 4; Experimental results are provided in Section 5; Finally conclusion is drawn in the last section.

[1]This work is supported by Natural Science Foundation of China with grant number 60605007 and Youth Foundation of UESTC

II. Computation of Ridge Curvature

In this paper we take advantage of the preprocessing, orientation field computation algorithms in [6]. After getting the fingerprint orientation field and minutiae, our focus is on the curvature computation process. We propose the novel and simple way to calculate the ridge curvature for each pixel in this section.

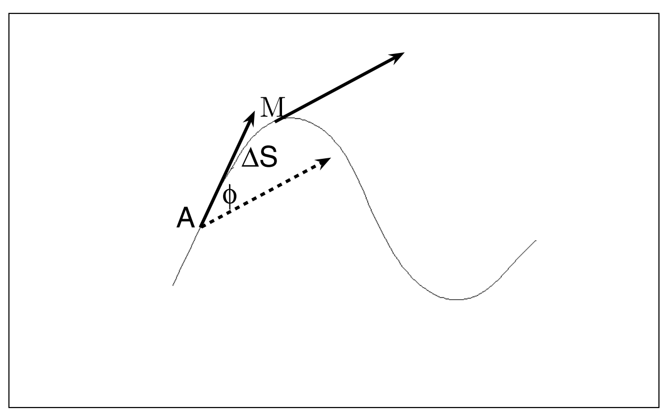

Fig. 1. An illustration for the curvature computation of a spatial curve

Before giving our ridge curvature computation algorithm, we introduce the general formula for curvature calculation of spatial curves [1]. A general spatial curve is shown in Figure 1. There are two points on the curve: Point A and Point M. ΔS is the length of the curve AM. The letter φ in the figure denotes the acute angle between the two tangent lines which are tangential to the curve at the point A and the point B, respectively. The curve curvature at the point A is calculated by the following formula:

$$k = \lim_{\Delta S \to 0} \frac{\varphi}{\Delta S}. \quad (1)$$

The equation indicates that the curvature of curve at point A is the limit when ΔS approaches to zero value. The curvature k indicates the changing rate of the orientation of the tangent line at the point M when the point M moves away from the point A along the curve.

We divide a fingerprint image into blocks with $W \times W$ size to compute the fingerprint block orientation instead of pixel orientation. Each pixel in the same block has the same orientation. Correspondingly, only block curvature can be obtained based on the block orientation. According to the general curvature computation formula 1, each block curvature can be calculated in its 3×3 block neighborhood (see Figure 2) around the block using its neighboring blocks orientation. The curvature of center block is half of the difference between the orientations of the two neighboring blocks to which the direction of the center block points. The formula for computing center block curvature Bk is as follows:

$$Bk = \frac{|O(B1) - O(B2)|}{2}, \quad (2)$$

where the two blocks, $B1$ and $B2$, are the blocks pair to which the orientation of center block in the neighborhood points. For instance, in Figure 2 the four neighboring block pairs, $P15$, $P26$, $P37$, $P48$, are respectively distributed in four directions: $-45°$, $90°$, $45°$, $0°$. The orientation of center block $B0$ is $30°$. It is obvious that among the four directions the direction $45°$ has the minimum difference $15°$ with respect to the orientation of center block. Hence, the curvature of the center block $B0$ is half of the difference between the two orientations of the two blocks: Block $B3$, Block $B7$. Figure 3 shows the result of fingerprint curvature computation using our proposed method here. The brighter the block, the bigger its curvature. According to the shown result in Figure 3, our curvature computation algorithm is effective.

III. Detection of Singular Points

As we know, the singular points are in the areas where ridge directions are not continuous and change abruptly. It means that the singular points must be contained in the areas with relatively large ridge curvature values. If no noise presents at fingerprint images or relatively less image noise exists, the point with the maximum curvature must be a singular point.

With the curvature map computed in the above section, we propose the SP detection algorithm as follows:

1) Set two curvature thresholds T_c, T_s and distance threshold T_d. Let $T_c > T_s$.

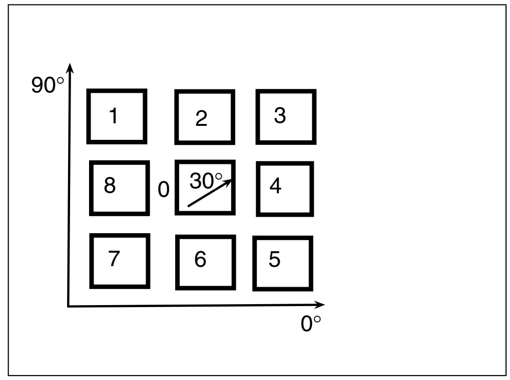

Fig. 2. An illustration of size 3×3 block neighborhood

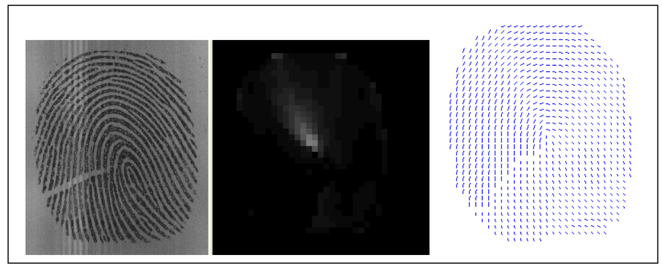

Fig. 3. Curvature map computed using our method: from left to right, original fingerprint image, curvature map and orientation field. The more brighter the block in middle curvature map, the more bigger the corresponding curvature value.

2) Denote by S_c the collection of candidate blocks of which curvature is more than T_c.
3) Choose any two candidate blocks B_i and B_j from the set S_c. If the distance between them is less than T_c, the block with the smaller curvature is discarded from the set S_c and go to step 5.
4) If the curvature of any one block on the line segment joining the two blocks B_i and B_j is bigger than T_s, then the block with the smaller curvature is discarded from the set S_c.
5) Return to the step 3 until the size of set S_c is stable.
6) Finally, the set S_c is the collection of singular points.

The size of the set S_c in above algorithm is the number of singular points in one fingerprint image. Any block in set S_c corresponds to one singular point. So we got the singular point set S_c using the algorithm mentioned above. It is noted here that we use distance threshold and curvature threshold to ensure that each singular point has relatively big curvature and that there is reasonable distance between any two singular points.

In order to determinate the type of singular points in set S_c, an Poincare Index based technique [3], [4], [5] is adopted in this paper. Different from those methods in [3], [4], [5] taking much time to compute the Poincare Index value of each pixel in the whole image, our method here is only to calculate the Poincare Index values of a few (normally less than four) singular points in set S_c. Hence our time spent in determining singular point type can be neglected. The process of computing Poincare Index value is detailed in [3], [4], [5]. There exist two kinds of singular points: core point and delta point. The Poincare Index values of core point and delta point are π and $-\pi$, respectively.

It should be noted that our singular points are obtained with block precision. In other words, we can only determine whether some block contains one singular point. But we can't determine whether some pixel is singular point. This seems not to be a problem because the definition of singular points is not very accurate.

IV. Computation of FIRD

With singular points obtained in above section, we can conduct the calculation of FIRD. First, we give the definition of core point orientation. Usually, the definition of core point orientation is various. Here in order to define FIRD, our definition of core point orientation is the following:

> **Definition 1** The orientation of core point is the orientation of the block with the least curvature in the 5×5 block neighborhood of the core point block.

We think that delta point has no orientation. Second, we give the definition of FIRD as follows:

> **Definition 2** FIRD is the average of all core point orientations.

According to the above definition of FIRD, our algorithm for computing FIRD is as follows:

1) Choose any one core point P_i from core point set S_c and delete P_i from the set S_c.
2) Find the block B_i with the least curvature in the 5×5 block neighborhood of core point P_i.
3) The orientation O_i of the block B_i is the orientation of core point P_i.
4) Return to step 1 until the set S_c is empty.
5) Suppose N is the number of core points. FIRD is the average value: $\sum_{i=1}^{N} O_i$.

It should be known that if there isn't core point in fingerprint image, FIRD is not computed. The algorithm for determining FIRD of fingerprint image without core point is our future task.

V. Experimental Results

To evaluate the performance of our algorithm proposed in this paper, we conduct a series of experiments on the public domain collection of fingerprint images: Set B of DB3 in FVC 2002, which is the training set of DB3 in FVC 2002 for algorithm optimization. The Set B contains 80 fingerprint images from 10 fingers with 8 fingerprint images for each finger. The size of fingerprint images is 300×300 pixels with resolution of 500dpi. We use our algorithm in this paper to detect fingerprint singular points contained in fingerprint images in the Set B. The total number of core points and delta points contained in Set B is 88 and 7, respectively. Our algorithm can detect 86 core points and lose two core points. It is because that the two lost core points locate in the margin of fingerprint image. The results obtained by us is tabulated in Table I. The results show that our algorithm can determine the number and the type of fingerprint singular points accurately.

True SP		Detected SP	
Type	Number	Core	Delta
Core	88	86	0
Delta	7	0	7

TABLE I
Results of singular points detection in Set B of DB3 of FVC2002 using our algorithm

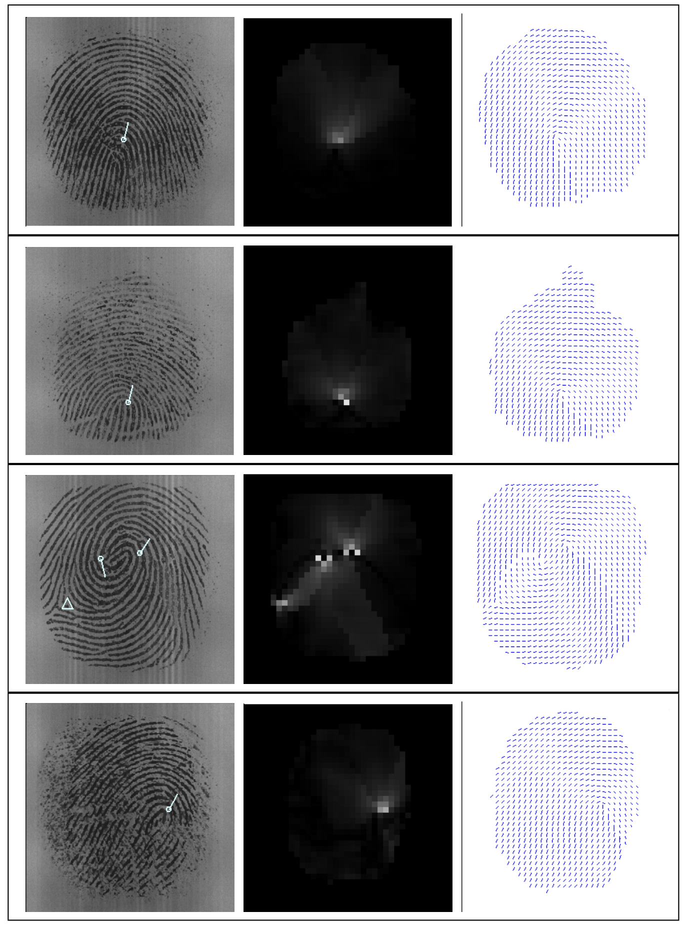

Fig. 4. Results of singular point detection and FIRD computation. From left to right: original image, corresponding curvature map, corresponding orientation field. In the left images white circles denote core points, and white triangle and white line segments indicate delta point and FIRD, respectively.

Figure 4 below shows some fingerprint images in Set B with singular points and FIRD using the proposed curvature-based algorithm. It indicates that the position of singular points and the FIRD are computed coarsely. This is due to the inaccurate computation algorithm of fingerprint orientation field in [6], which is adopted here to compute fingerprint orientation field. The corresponding coarse orientation fields and curvature maps are also showed in Figure 4. From Figure 4, we can see that orientation fields are computed in a coarse level. The algorithm for singular point detection and FIRD computation in this paper can be improved by enhancing orientation field calculation algorithm. However, the position of singular points and FIRD is useful for fingerprint match. For example, singular points and FIRD can determine the coarse correspondence between two fingerprint images or two minutiae sets, which can reduce the matching time when matching. In addition to, our algorithm can be used to classify fingerprint images by detecting singular points.

VI. Conclusion

In this paper, we propose a method to detect singular points and compute FIRD by the analysis of curvature map. The computation of curvature is simple and fast. Our method only needs to compute a few Poincare Index values for each fingerprint image rather than many Poincare Index values in those Poincare Index based methods in [5], [4], [3]. The FIRD defined and computed by us is very useful for fingerprint coarse registration. Experimental results show that our algorithm is effect and efficient.

References

[1] A. Aleksandrov, *Mathematics, its essence, methods and role*, Publishers of USSR Academy of Sciences, Moscow, 1956.

[2] A. Bazen, and S. Gerez, Extractin of singular points from directinal fields of fingerprint. *Mobile Communications in Perspective, Annual CTIT Workshop*. Enschede, The Netherlands.

[3] A. Bazen, and S. Gerez, Systematic methodsfor the computation of the directional fields and singular points of fingerprints, *IEEE Trans. PAMI*, 24(7), 905–919, 2002.

[4] B. Cho, J. Kim, J. Bae, I. Bae, and K. Yoo, Core-based fingerprint image classification. *Proc. of ICPR 2000*, Spain, 2000.

[5] L. Hong, and A. Jain, Classification of fingerprint images. *Proceedings of 11th Scandinavian Conference on Image Analysis*. Jun 7–11, Kangerlussuaq, Greenland, 1999.

[6] A. Jain, L. Hong, and R. Bolle, On-line fingerprint verification. *IEEE Trans Patt Anal Mach Intell*, 19(4):302–314, 1997.

[7] X. Jiang, M. Liu, and A. Kot, Reference piont detection for fingerprint recognition. *Proc. of ICPR04*. 2004.

[8] W. Koo and A. Kot, Curvature-based singular points detection. *AVBPA 2001, LNCS 2091*, 229–2345, 2001.