Structure Matching Algorithm of Fingerprint Minutiae Based on Core Point[1)]

ZHANG Wei-Wei WANG Sen WANG Yang-Sheng
(*National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing* 100080)
(E-mail: {wwzhang, swang, wys}@nlpr. ia. ac. cn)

Abstract Fingerprint matching algorithm is a key issue of the fingerprint recognition, there already exist many study about fingerprint matching algorithms. According to dependence on core point, fingerprint matching algorithms can be grouped into two categories: core-based match algorithms and structure-based match algorithms. Most of the structure-based matching algorithms are time consuming, therefore they are not suitable for on-line application. Meanwhile, core-based matching algorithm is more efficient than the structure-based matching algorithm, but it highly depends on core point detection precision. In this paper, we present a new core-based structure matching algorithm which both considers efficient and precision. Firstly, we use core point detection algorithm to find the core point and define some local minutiae structures around core point. Through matching these local minutiae structures, we can get some corresponding points of the two fingerprint images. Secondly, we use the corresponding points to match the global feature of fingerprints, where a match table is constructed to prevent one to many false minutiae match. Finally, we promote the global match distance and global match distance variance to help make the final decision. Experimental result shows that the performance of the proposed algorithm is good.

Key words Fingerprint recognition, core point, matching algorithm, structure matching

1 Introduction

Fingerprint-based personal identification has been used for a very long time owning to their uniqueness and immutability. Today, fingerprints are the most widely used biometrics features in automatic verification and identification systems. Most automatic fingerprint identification systems (AFIS) are based on local ridge features, such as ridge endings and ridge bifurcations[1].

The key issue of the fingerprint recognition is the minutiae matching algorithm. Although there exist a lot of matching algorithms in literature, most of them are not suitable for on-line application because of the following difficulties.

1) The alignment of two minutiae features vectors. After the feature extraction of fingerprint, we get a minutia feature vector. And we need alignment the feature vectors before matching them. According to usage of core point, the alignment methods can be divided into two groups: core-based alignment and structure-based alignment methods.

2) Due to the noise of the fingerprint image, the preprocessing algorithm can not remove all of the noise from feature vector. Therefore, the match algorithm should robust to noise.

3) The geometric distortion of the fingerprint image may greatly affect the match algorithm, and there still is no effective method to deal with this nonlinear distortion.

Moreover, the on-line system requires the matching speed as fast as possible, together with small size template. In order to solve the difficulties mentioned above, there are a lot of researches dealing with some of them in the literature.

In [2], Z. Chen proposed a topology-based matching algorithm of fingerprint authen-

1) Supported by Creative Foundation of Chinese Academy of Sciences, the Project of Institute of Automation
Received May 9, 2002; in revised form April 22, 2002
收稿日期 2002-05-09; 收修改稿日期 2003-04-22

tication. In their algorithm, they constructed a structure for minutiae in a pre-specified neighborhood, and then used a tree matching algorithm to match two fingerprint templates. Their algorithm is invariant to translation, rotation and does not depend on core point. Whereas, their algorithm will be greatly affected by noise. When noise increases, the performance of their algorithm degrades rapidly. Furthermore, they constructed a structure for each minutia, this lead to a large size template. Andrew K Hrechak proposed a structural matching algorithm[3], which used the local structure of the minutia to describe the characteristics of the minutiae. And in [4] some improvements for this approach are proposed. Both of those two methods just used the local structure information of the fingerprint. However the local structures from the same finger may have less similarity due to noise, and they need to build the local structure during the feature extraction stage, which will lead to a large size template. Therefore their algorithm is not suitable for the on-line application. In [1], Jain proposed an alignment-based elastic matching algorithm in which the ridge link to the minutiae is used to alignment the feature vectors. In order to do this alignment, they save ten sample points of the ridge which link the minutiae. This also leads to a large size template.

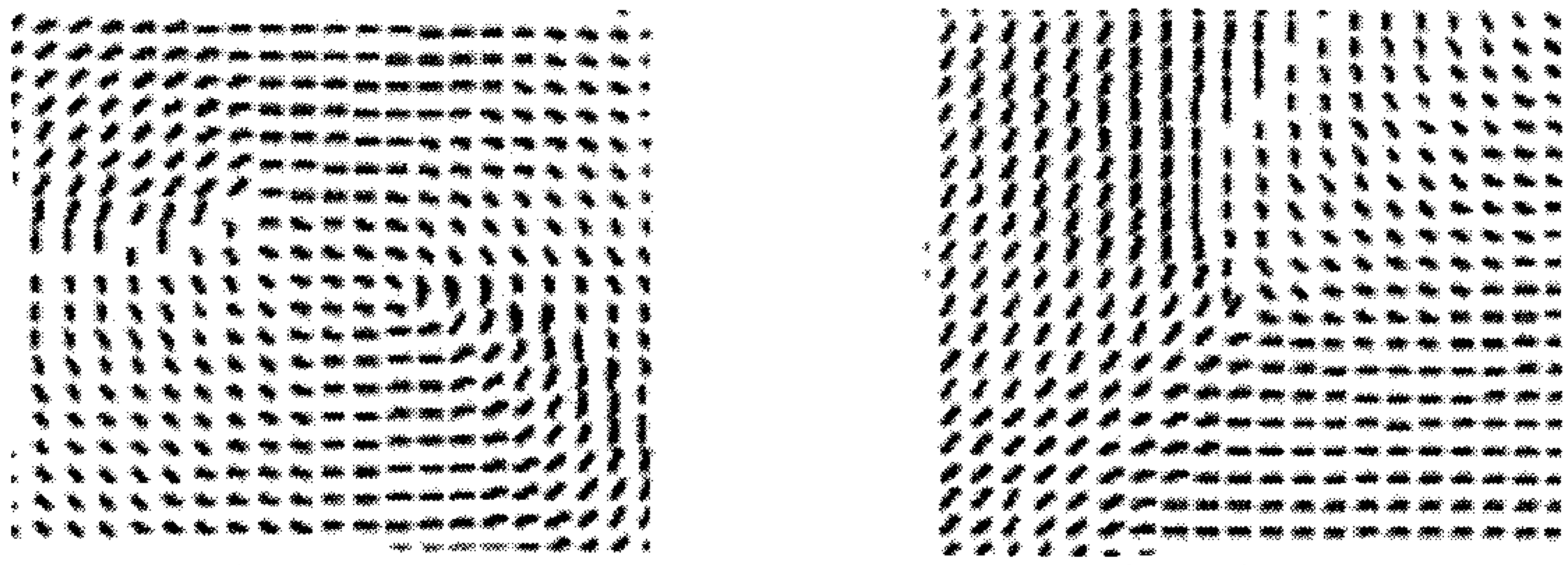
In [5], Xudong. J proposed a matching algorithm which is based on local and global structures. First they used the k-nearest neighborhood minutiae of each minutia form the local minutia structure, and defined a similarity level to measure the similarity of two local structures. Then they used the most similar minutiae pair as the corresponding point to alignment the feature vectors, and matched the global feature vectors. This method has some advantages in processing speed and robustness to rotation. However, they need to extract the minutiae type and ridge line count between each minutiae and its k-nearest neighborhood exactly, therefore the performance of their algorithm will be greatly affected by the noise. In [6], Nalini K. Ratha proposed a fingerprint authentication algorithm using local structural information. Firstly, they obtained a minimum set of matched node pairs by matching their neighborhood structures. Then they included more pairs in the later matching stage by comparing distances with respect to matched pairs obtained in the first phase. Their methods are robustness in large fingerprint database, but their processing speed is low, therefore this method is not suitable for the online application.

When considering an on line application, none of the above mentioned method is suitable for this type application because of the template size and the system efficiency. In this paper, we propose a new core-based structure matching algorithm which considers both efficiency and precision. Firstly, we use core detection algorithm to get the core position, and then define some local structure for the minutiae near the core point. Through matching those local structures, we can get some corresponding points of the two fingerprints. Secondly, we use the corresponding points in the first stage to match the global feature of the fingerprint. Finally, we use the global matching distance to help the final decision making. The core point detection method is present in Section 2. In Section 3, the method of local structure matching is presented. The global feature matching algorithm and final decision making algorithm is presented in Section 4. In Section 5, experimental results of the algorithm are shown. Finally, the conclusions are given in Section 6.

2 Core point detection

In the earlier work, we developed a fast multi-resolution based core point detection algorithm[7]. It mainly involves two steps: Firstly, we use the low resolution direction field to localize a singular area which includes the core point. Secondly, we use high resolution direction field of singular area to precisely localize the core point. Using this multi-resolution method, we can localize the core point in block levels fast and precisely.

Due to various image qualities, our algorithm can not localize the core point in pixel level; therefore, we can not directly use the core point as the reference point for the matching algorithm. But in most cases, the core point detection is very precise, which will locate the singular point within 8×8 pixels. If we use this reference information, we can greatly reduce the search space, thus improving the efficiency of the matching algorithm. According to this idea, we develop a core based matching algorithm in this paper, which uses the core point to select some minutiae points, and uses those minutiae points to construct a lot of local structures. Local and global matching is based on those local structures.



(a) Core Point (b) Delta Point

Fig. 1 Direction field near the singular soint

We choose core point as the reference point for the following three reasons: Firstly, there exists rich minutia information in close around core point region than others, which will help to construct uniquely local structure. Secondly, due to our early work, we can easily localize the core point, the details of the core point detection algorithm are found in [7]. Finally, owning to the guidance of core point, we only need to construct local structure for a small amount of minutiae. This will greatly reduce the template size and improve the efficient of our matching algorithm.

3 Local structure matching

Usually, when an expert recognizes a fingerprint, he would first finds correspond of fingerprint features and then correlates the features based on minutia type, position, orientation and location relative to other features. Usually, we want to let the computer do as the human expert does. That is, firstly we use the local structure of the fingerprint to find some corresponding point pairs through structure matching, and then we match the global feature vector of the two fingerprints based on those corresponding point pairs.

As we know, local structure of the fingerprint minutiae has some properties which help us to match the fingerprint:

1) Local structures in a small area from the same fingerprint are similar in minutiae type, distance and angle;

2) Local structures in a small area from different fingerprints are often different in minutiae type, distance, angle and topology information;

3) Fingerprints which come from the same finger often have many similar local structures;

4) Fingerprints which come from different fingers often have less similar local structures than those from the same one.

Usually there are 30～60 minutiae in one fingerprint. If we construct local structure for every minutia, the template size will be very large, and large template will decrease the system's efficiency. In our algorithm, we only select the minutia near the core point of the fingerprint, the selection rule is:

$$|M - Core| < R \tag{1}$$

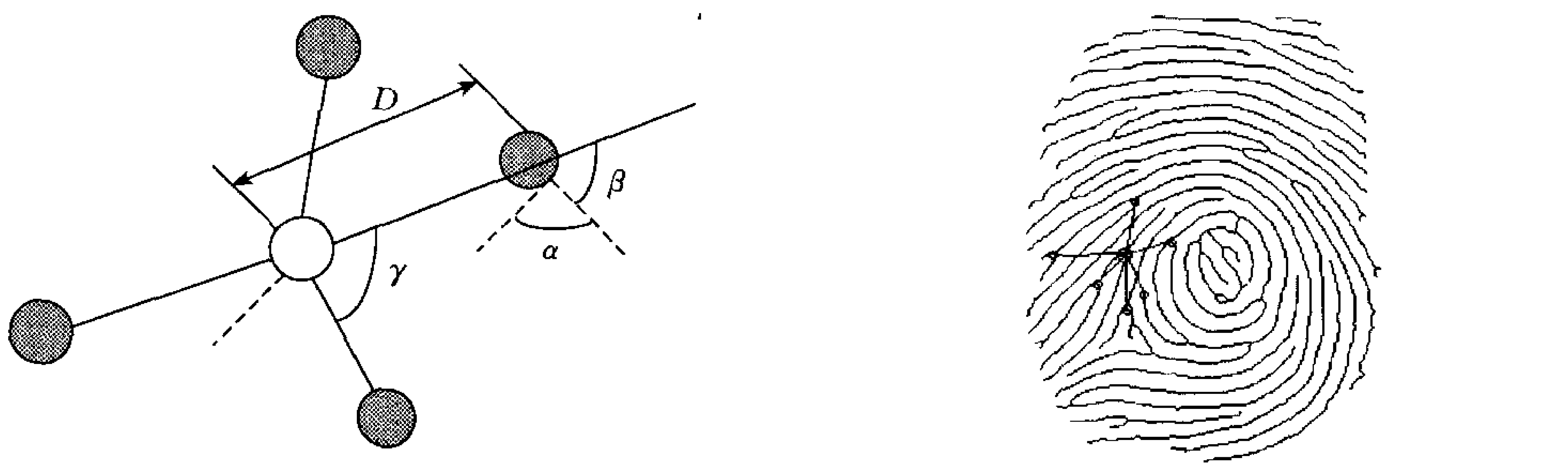
where M and Core denote the minutia and core point respectively, $|\cdot|$ denote the distance function, R is a constant which is determined by experiment. Usually, 10～15 minutiae are enough for the local structure. Therefore, reader can adjust the R according to the minutiae count.

3.1 Definition of local structure

In this section, we present the local structure definition of minutiae that are used in our match algorithm. Usually, there are two ways to construct local structure.

The first one is use all of the minutiae around a minutia within a constant distance to construct a local structure. The problem of this method is how to decide the constant distance. As we know, if the distance is too small, very few minutiae will be included for the local structure. This will lead to false match with the noise influence. But with a large distance, the structure will be seriously affected by the elastic distortion. The second method is use the k-nearest neighborhood of minutiae to construct a local structure.

In our matching algorithm, the second method is used to construct a local structure. An example of local structure is shown in Fig. 2, which demonstrate an example of four-nearest neighbor local structure. We call the line that connects the center minutia and the neighbor minutia as edge, and construct local structure uses the follow information:



(a) four-nearest neighbor local structure (b) local structure in the thinned fingerprint

Fig. 2 Example of local structure

The distance between the center minutiae and the neighbor minutiae, as D shown in Fig. 2.

The angle α between center minutiae's orientation and neighbor minutiae's orientation in Fig. 2.

The angle β between the minutiae orientation and the edge. As shown in Fig. 2.

The angle γ between the neighboring two edges, as shown in Fig. 2.

It is known that the elastic distortion of fingerprint in a local small area always similar. Therefore, we can regard that the distances between the center minutia and the neighbor minutia are invariable to translation and rotation. It is apparently that the angles α, β, γ are invariable to translation, rotation. Therefore, we can use those four features to match the local structure.

3.2 The local structure matching algorithm

In this section, we introduce the local structure matching algorithm which is used to find corresponding point pairs of the two fingerprints.

There are two methods to match the local structure: The first one is to compute the matching cost of the two local structures. Then select the center minutia of the matching pair which has the minimal cost as the correspondence pairs. The second one is to compute the matching edge of the local structure. We can compare the edges from the two local structures. If the edges are similar in distance, angle etc, we say those two edges are matched. Moreover, if two edges of one local structure are matched with the other two edges of the other local structure, the angles should be similar. Through matching all edges of the local structure, we can get the matching edge count, and set the match score of the two local structures as the matching edge count.

Through experiment we find that the second method is both efficient and robust to noise, therefore we select the second as our local structure matching algorithm.

Now we describe our local structure matching algorithm in detail. For each edge of the two local structures we compute:

1) The relative distance difference R_d of the two edges, with D_1 and D_2 being the distances of two edge's from different local structures as

$$R_d = |D_1 - D_2| / \min(D_1, D_2) \tag{2}$$

the min(• , •) function means to select the minimal of the two values;

2) The angle difference:

$$\begin{aligned} D_\alpha &= \alpha_1 - \alpha_2 \\ D_\beta &= \beta_1 - \beta_2 \end{aligned} \tag{3}$$

If R_d, D_α, D_β are below a given threshold, we say the two edges are matching.

If two edges of the template match the other two edges of the input image, let St_1 and St_2 denote the local structures of the two fingerprints, $E_{11}, E_{12}, E_{21}, E_{22}$ be edges of those two local structures. And

$$E_{11}, E_{12} \in St_1; \qquad E_{21}, E_{22} \in St_2$$

Moreover, let γ_1 *and*γ_2 denotes the angles between E_{11}, E_{12} and E_{21}, E_{22}, respectively. Then we compute the difference between those two angles:

$$D_\gamma = \gamma_1 - \gamma_2 \tag{4}$$

If D_γ is below a given threshold, we say that both of those two edges are really a match, and increase the match score of the two local structures with two.

After those steps, we will get a match score of the two local structures. In order to avoid one edge from a local structure matching more than one edge with the other local structure, we set a flag with the compared edge, and use the edge which has not be matched to start the new edge match.

3.3 The confirmation of the matched structures

After the local structure matching stage, we have obtained some similar structures. We will use the center minutiae of those structures in the global matching stage. But before we do the global matching, we need to confirm the correctness of those matched local structures in order to decrease the false matching rate. As we know, the minutiae from the same finger holds stable relationship with core point, this is to say, the center points of those local structures are very similar in angle, position and distance with the core point. According to this hypothesis, we come to the following confirmation method of the matched local structure.

Let S_1 and S_2 be the center points of the two matched local structure in different fingerprint image respectively, andC_1, C_2 be their core points. We compute:

$$\begin{aligned} D_1 &= |S_1 - C_1|, \quad \alpha_1 = O_{s_1} - O_{c_1} \\ D_2 &= |S_2 - C_2|, \quad \alpha_2 = O_{s_2} - O_{c_2} \end{aligned} \tag{5}$$

where O denotes the orientation of the core point and minutia, D denotes the distance between the minutiae and core point.

$$\text{If } |D_1 - D_2| < T_D \text{ and } |\alpha_1 - \alpha_2| < T_\alpha \tag{6}$$

then we say that those two local structures are real matched. Here we used the orientation estimation method given in [8]. This step just confirms the correctness of matched local structures, the threshold can be a little loose to decrease the FRR in some sense.

4 Global matching algorithm

In Section 3, we match the local structure around the core point, however, just matching the local structure is not enough for making verification decision, because the local structures usually tend to be similar among different fingerprints. Therefore in this section, we match the global feature based on the local structure match result. From the local structure matching results, we select the two structures which have the maximum match score, and use the center minutia of the two structures as the corresponding point pair of the two fingerprints. After that, we normalize the global feature based on the corresponding point pairs.

Let

$$P = ((x_1^p, y_1^p, \theta_1^p)^{\mathrm{T}}, \cdots, (x_m^p, y_m^p, \theta_m^p)^{\mathrm{T}})$$

denote the minutiae feature vector of the template and

$$Q = ((x_1^q, y_1^q, \theta_1^q)^{\mathrm{T}}, \cdots, (x_n^q, y_n^q, \theta_n^q)^{\mathrm{T}})$$

denote the minutiae feature vector of the input image. If we denote the reference point as (x^r, y^r, θ^r), we have the normalization formulation:

$$\begin{bmatrix} r_i \\ \theta_i \\ o_i \end{bmatrix} = \begin{bmatrix} \sqrt{(x_i - x^r)^2 + (y_i - y^r)^2} \\ \arctan\left(\dfrac{y_i - y^r}{x_i - x^r}\right) \\ \theta_i - \theta^r \end{bmatrix} \tag{7}$$

where (x_i, y_i, θ_i) denotes the minutiae from both the template and input image. After the normalization, we match the minutiae vectors through an elastic box, as suggested by Jain[1].

In order to avoid one to many false minutia matching, we construct a match table M for each matched minutia pair P_i, Q_j and we compute the "distance" of those match minutia pairs. Let

$$\Delta d = r_i - r_j$$
$$\Delta \alpha = abs(\theta_i - \theta_j) + abs(o_i - o_j) \tag{8}$$

And save this "distance" into the match table,

$$M(i,j) = \Delta d + \Delta \alpha$$

After the entire global match finishes, we search the match table with the search rule as follows.

For each row, if there is nonzero value, we get the minimum value, and set the associated column and row to zero except this minimum value. After this operation, the matched minutiae number is the nonzero count k in the match table. We define a match rate for two fingerprints;

$$R = \sqrt{k \times k / m \times n} \tag{9}$$

where R is the match rate, and k is the matched pair count, m, n are the minutiae count from template and input fingerprint image.

4.2 Global matching decision

After the global matching, we need to make final decision that whether the two fingerprints are same or not. There are many drawbacks if we just use the matched minutiae point count to make decision. Therefore, in this paper we introduce a global distance based decision method. We find that if we can introduce some efficient constrains, we can greatly decrease the false match risk. In order to do that, we introduce global matching distance (GMD) and global matching distance variance (GMDV).

Let $S = \{s_1, s_2, \cdots, s_n\}$, $R = \{r_1, r_2, \cdots, r_n\}$

where S denotes the minutiae vector of the saved template, R denotes the minutiae vector of the input template. There must exist the following relationship between the real matched minutiae if we neglect the nonlinear distortion:

$$R = A \times S + b$$

where A and b are the affine matrixes of the two fingerprints.

$$S^* = A \times S + b$$

where S^* is the minutiae vector after affine transform under the affine matrix.

We define the global distance GMD as:

$$E = \frac{1}{n} \sum (S^* - R) \tag{10}$$

Under the ideal case, the GMD should be zero. Due to noise and nonlinear transform, the GMD is always a positive value. In order to tolerate those errors, we let GMD below a given threshold:

$$E < T_\eta \tag{11}$$

Meanwhile, we compute the variance of the GMD

$$\sigma = \sqrt{\frac{1}{n} \sum (S^* - R - E)^2} \tag{12}$$

and let σ below a given threshold

$$\sigma < T_\sigma \tag{13}$$

If the matched minutiae meet those two conditions as well as a given minimal matched minutiae count, we regard those two fingerprints as a match.

5 Experiment result and analysis

In order to evaluate our algorithm, we test our algorithm on our fingerprint image database. The image size is 300×300 pixels. Our fingerprint image database includes two parts. Part I has 100 images per finger, and 4000 images altogether. Part II has 20 images per finger, and 4000 images altogether too. We test the FRR on Part I, and test FAR on Part II. Our fingerprint image database includes various type quality fingerprint images, some are shown in Fig. 3.

Fig. 3 Example of different quality fingerprint image

We do altogether 80000 times matching. The test results are shown in Table 1. The comparison with Jain's algorithm[1] is also given in Table 1.

Table 1 Experiment results of our algorithm compared with Jain's algorithm

	Our algorithm		Jain algorithm[1]	
	False Accept Rate(%)	False Refuse Rate(%)	False Accept Rate(%)	False Refuse Rate(%)
1	5.0	0.08	5.0	0.8
2	4.0	0.09	4.0	1.7
3	3.0	0.35	3.0	1.8
4	2.0	0.58	2.0	3.2
5	1.0	0.80	1.0	5.6

From Table 1 we can see that, our algorithm has better performance with our live fingerprint database. Moreover, the average template size is 256 Bytes which only include the position and orientation of the minutiae points (6 Byte per minutia), and the total time consuming of our matching algorithm is 0.004 seconds on PIII 450M Hz. From the test result we can see that our algorithm has both better performance and efficiency, therefore is more suitable for on-line application.

The reason leading to the verification failed is the poor quality fingerprint image as shown in Fig. 3. Those types of fingerprint images are confusing even to manual verification. If we add some strict image quality estimate algorithm, the performance can be greatly improved, therefore the future work of our system is to develop some strict image quality estimate algorithm as well as improve the minutiae extraction precision. Meanwhile, in order to totally verify the performance of our algorithm, we need to test our algorithm in the standard fingerprint database such as NIST, FVC2002. Therefore our future work should test and improve the performance according to those standard fingerprint databases.

6 Conclusion

In this paper, we propose a core based structure matching algorithm of fingerprint, which uses the minutiae near core point of the fingerprint to construct some local structures. Through matching the local structures, we can get corresponding minutiae. The global match is carried out based on those corresponding minutiae. Final decisions are making based on global distance and matching minutiae point rate. Experiment results show that the performance and efficiency are significant improved with our algorithm.

References

1 Anil Jain, Lin Hong, Ruud Boole. On-line fingerprint verification. *IEEE Transactions on PAMI*, 1997, **19**(4):302~314

2 Chen Z, Kou C H. A toplogy-based matching algorithm for fingerprint authentication. In: Proceeding of IEEE International Carnahan Conference on Security Technology, Taipei: IEEE Press, 1991. 82~85

3 Andrew K Hrechak, James A Mchugh. Automated fingerprint recognition using structural matching. *Pattern Recognition*, 1990, **23**(8): 839~904

4 Wahab A. Novel approach to automated fingerprint recognition. *IEE Proceedings on Visual Image Signal Process.*, 1998, **145**(3):160~166

5 Willis A J, Myers L. A cost-effective fingerprint recognition system for use with low-quality prints and damaged fingerprint. *Pattern Recognition*, 2001, **34**:255~270

6 Srinivasan V S, Murthy N N. Detection of singular point in fingerprint images. *Pattern Recognition*, 1992, **25**(2): 139~153

7 Zhang Wei-Wei. Singular point detection in fingerprint image. In: Proceeding of the 5th Asian Conference on Computer Vision, Melbourne, Australia, 2002. 793~796

8 Asker M Bazen, Sabih HGerez. Extraction of singular points from directional fields of fingerprints. In: Proceeding of CTIT Workshop on Mobile communications, Enschede, The Netherlands: University of Twente, 2001. 41~44

ZHANG Wei-Wei Received his master degree in Automation form Beijing University of Aeronautics and Astronautics in 2000. He is currently a Ph. D. candidate in National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Science, Beijing, P. R. China. His research interests include digital signal processing, pattern recognition, and fingerprint recognition.

WANG Sen Received his bachclor in Automation from Shandong University. He is currently a master student in National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, P. R. China. His research interests include pattern recognition, computer vision, and digital image processing.

WANG Yang-Sheng Received his master(1984) and Ph. D. (1989) degree in Electronic Engineering from Huazhong University of Science and Technology. In October 1989, he joined the University of Edinburgh and did his Post-Doctor research. From 1990, he worked in Brunel University as a senior researcher. From 1999 he worked as a senior researcher in Nortel Networks (UK). In 2000, he returned to P. R. China and joined the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Science, Beijing, P. R. China. He is currently Professor of National Laboratory of Pattern Recognition. Dr. Wang is also the professor of Beijing University of Posts and Telecommunications, Huazhong University of

Science and Technology, and CEO Beijing Pattek Co. His research interests include biometrics, robotics, computer vision, and pattern recognition.

基于中心点的指纹细节结构匹配算法

张伟伟　　王　森　　王阳生

(中国科学院自动化研究所模式识别国家重点实验室　北京 100080)

(E-mail: {wwzhang, swang, wys}@nlpr.ia.ac.cn)

摘　要　指纹细节匹配算法是自动指纹识别系统(AFIS)中一项关键的任务,目前存在大量的研究和算法.依据算法是否依赖中心点,指纹细节点匹配算法可以分为两类:基于中心点的匹配算法和非中心点匹配算法.大多数非中心点匹配算法都非常耗时,因此不适合在线应用.而基于中心点方法的效率相对较高,但是这类算法极度依赖于中心点的定位精度.在本文中,提出了一种全新的基于中心点的指纹细节结构匹配算法,该算法综合了基于中心点匹配算法和非中心点匹配算法的优点,同时又避免了二者的缺点.首先利用中心点检测算法获得中心点的位置,然后在中心区域定义了一些局部的结构,同时利用这些局部结构寻找指纹细节的对应点,并通过对应点和中心点的相对关系来确认这些对应细节点.其次利用这些细节对应点匹配全局的细节信息,最后,利用匹配细节的全局距离和距离方差来判决最终匹配结果.实验结果表明,算法的匹配效果非常好,同时匹配效率较高,非常适合在线指纹识别系统的应用.

关键词　指纹识别,中心点,匹配,结构匹配

中图分类号　TP391.4