# Fingerprint Matching Method based on Minutiae and Direction of Minutiae

Chen Tang

School of Hunan Railway Professional Technology College, Zhuzhou 412001, China

372384937@qq.com

## Abstract

Nowadays identity recognition based on fingerprint has been received more and more attention. And it has more and more been put into application. The matching method of Fingerprint is the key of fingerprint recognition, so the quality matching method affects the accuracy rate of recognition directly. Because of the translation and rotation of the fingerprint, the coordinates of the same feature points can be different in two homologous fingerprints. But the difference of distance and direction between the feature points and center point cannot be changed, and the type of feature points also cannot be changed. Based on this rule, I put forward a matching method which is base on the difference of distance and direction between the feature points and center and the type of the feature points. The experiments confirm the method can effective avoid the influence of translation and rotation.

## Keywords

Fingerprint recognition; Translation and rotation; Feature point.

## 1. Introduction

Traditionally, passwords and badges have been used to restrict access to secure systems. However, the security systems can be easy got access when a password is deciphered or a badge is stolen. The emergences of biometrics [1] solve the problems that exist in traditional verification methods. Fingerprint is the most widely used verification method among biometrics because it is stability, safety and easy to capture. Automatic fingerprint identification systems (AFIS) are in great demand. AFIS is usually used in criminal identification but now it is more and more popular using in civilian applications [2]. There are many fingerprint matching algorithms, some of them are based on graphs and images or ridge structures [3]. But the most popular matching algorithm is based on the minutiae(ending points and bifurcation) which is also used by the FBI because it is balancing robustness and effective. Basically, an AFIS consists of three parts: preprocessing, feature extraction and matching. The preprocessing include segmentation, enhancement, binarization and thinning. Here is a simple flow chart of an AFIS. Most minutia-based matching algorithms [3-11] need to compute the rotation and translation parameters to align the input image and the template image. This algorithm can acquire high matching accuracy rate, however it is too difficult to compute the rotation and translation parameters, and alignment consume a lot of time to. We present a novel matching algorithm that do not need to compute the rotation and translation parameters and also do not need to align the input image and the template image. Firstly we find out the center point of the fingerprint. The coordinate and orientation field of same minutiae would be different in the input image and the template image because of translation and rotation, but the distance between the center and minutia will equal in the input image and the template image, and the difference of orientation field between the center and minutia will also equal in the input image and the template image. We can calculate the score which is number of same minutiae existing between in the input image and the template image, then we can calculate a threshold through experiment. If the score is greater than the threshold, it demonstrates that the input image and the template image match, otherwise not match.

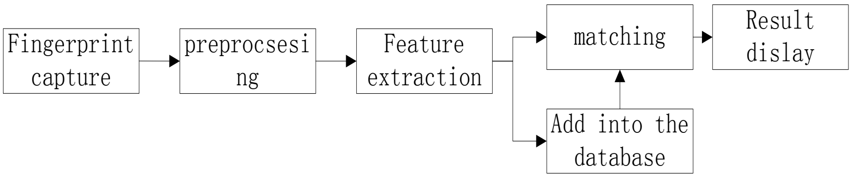


Figure 1.A simple flow chart of an AFIS.

## 2. Fingerprint feature extraction

### 2.1 Minutiae extraction.

In this paper, the method of minutia is based on the template matching on the thinning fingerprint. Firstly, we need to thin the fingerprint and build the 3×3 neighborhood of pixels as illustrated in Figure 2. P is the pixel under detection. $P_1, P_2, ...P_8$ are the neighborhood of P. $C_n(P)$ is the changing time of neighborhood pixels' gray value. The fingerprint is binary, so the gray value of pixels only can change from 1 to 0 and 0 to 1. $S_n(P)$ is the total number that the gray value of pixels are 1 in the neighborhood. The $S_n(P)$, $C_n(P)$ are:

$$C_n(P) = \sum_{i=1}^{8} P_{i+1} - P_i \ (P_1 = P_9) \tag{1}$$

$$S_n(P) = \sum_{i=1}^{8} P_i \tag{2}$$

If Cn(P)=2, Sn(P) =1, P is ending point ,as A illustrated in Figure 3 and the blue points in Figure 4.

If Cn(P)=4, Sn(P) =2, 3or 4, P is not minutia, as C1, C2 and C3 illustrated in Figure 3.

If Cn(P)=6, Sn(P) =3, P is bifurcation, as B illustrated in Figure 3 and the red points in Figure 4.

| | | |
|---|---|---|
| P1 | P2 | P3 |
| P8 | P | P4 |
| P7 | P6 | P5 |

Figure 2.The pixels neighborhood

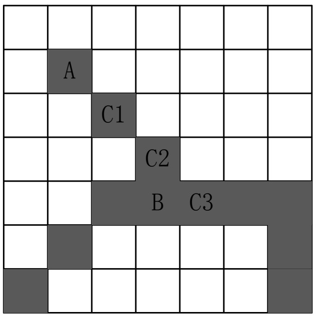


Figure 3. 3 type of minutiae

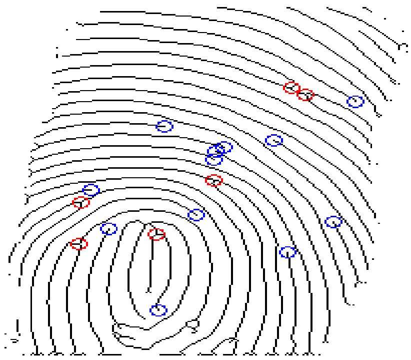

Figure 4.Feature points of fingerprint

### 2.2 Orentation Of minutiae.

The orientation field reflects the information that fingerprint ridge goes and it also can reflect the texture structure of fingerprint. The orientation field of minutiae can reflect the direction of fingerprint ridge that the minutia lies on. Orientation field of minutiae reserves important information of the fingerprint. The algorithm to calculate the orientation field of each minutiae are:

Get out the 8 neighborhood of minutiae: $P_1, P_2, ...P_8$, as illustrated in figure 2.

Using the sobel calculates the grads of minutia on x side and y side. Gx(u,v) represent the grads of minutiae on x side and Gy(u,v) represent the grads of minutiae on y side.The sobel is:

$$Sobel_x = \begin{bmatrix} -2 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}; Sobel_y = \begin{bmatrix} -1 & -2 & 1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

*The method to* calculate *$G_x(u,v)$ and $G_y(u,v)$ are:*

$$Gx(u,v) = 2(u, v+1) + (u-1, v+1) + (u+1, v+1) - 2(u, v-1) - (u-1, v-1) - (u+1, v-1)$$

$$Gy(u,v) = 2(u-1, v) + (u-1, v+1) + (u-1, v-1) - 2(u+1, v) - (u+1, v+1) - (u+1, v-1)$$

*u and v are coordinates of the minutiae.*

The formulas to calculate the orientation field of minutiae $O(i, j)$ are:

$$V_X(i, j) = \sum_{u=i-\omega/2}^{i+\omega/2} \sum_{v=j-\omega/2}^{j+\omega/2} 2G_x^2(u,v) - G_y^2(u,v) \tag{3}$$

$$V_y(i, j) = \sum_{u=i-\frac{w}{2}}^{i+\frac{w}{2}} ? \sum_{v=j-\frac{w}{2}}^{j+\frac{w}{2}} (G_x^2(u,v) - G_y^2(u,v)) \tag{4}$$

$$O(i, j) = \frac{1}{2} \tan^{-1} \left\{ \frac{V_y(i, j)}{V_x(i, j)} \right\} \tag{5}$$

w xis 8 which represent the length of the neighborhood path.

## 3. Fingerprint matching

Let minutiae feature vector Vb={b1,b2,b3,…,bn} denote the template fingerprint and minutiae feature vector Va={a1,a2,a3,…,an} denote the input fingerprint. $a_n$ and $b_n$ consist of the coordinates , type and orientation field of minutiae.

$$a_n = \{x_{an}, y_{an}, \theta_{an}, t_{an}\} \tag{6}$$

$$b_n = \{x_{bn}, y_{bn}, \theta_{bn}\} \tag{7}$$

We select the point which has the minimum vertical coordinate when it has the biggest rotation angle as the center point. The matching algorithm of template fingerprint and input fingerprint are:

Select a minutia $a_1$ from the vector Va. Calculate the distance of $a_1$ to the center of a. $c_a = \{x_{ca}, y_{ca}, \theta_{ca}\}$ denote the center of a.

$$D_{a1} = (x_{ca} - x_{a1})^2 + (y_{ca} - y_{a1})^2 \tag{8}$$

Calculate the orientation field difference of $a_1$ and $c_a$.

$$\alpha_{a1} = \theta_{ca} - \theta_{a1} \tag{9}$$

Calculate the distance of all the minutiae to the center of b.

$$D_{bn} = (x_{cb} - x_{bn})^2 + (y_{cb} - y_{an})^2 \tag{10}$$

Array $D_{bn}$ according to the value of it.

Calculate the orientation field difference of all the minutiae to the center of b.

$$\alpha_{bn} = \theta_{cb} - \theta_{an} \tag{11}$$

Find out the minutiae that they minutiae type are same as $a_1$ from Vb. Match the distance of minutia to center using bisection method. If there is a minutia render:

$$D_{a1} \approx D_{bn} \tag{12}$$

$$\theta_{a1} \approx \theta_{bn} \tag{13}$$

The score plus one.

Repeat the method in step 3 to find out if there are minutiae in Vb can match the minutiae in Va except $a_1$.

The score denote how many same minutiae existing between Va and Vb. Find out a threshold value according to the experiments. If the score greater than threshold value, it denote matching, otherwise not matching.

## 4. Conclusion

In the fingerprints, the situation may happen that the minutia from template fingerprint has same minutia type, distance to center and difference of orientation field to center with input fingerprint when the two fingerprints are not captured from the same finger. But this situation seldom happens. So it can avoid the error matching when set reasonable threshold. We use matlab to implement our algorithm and the fingerprint images are from the FVC2002. We finished 1000 times matching experiment. Each matching cost 0.00127 second. We find that the scores are greater than 9 in the majority situation when the two fingerprints are captured from the same finger, and the scores are less than 5 when the two fingerprints are not captured from the same finger. The error matching rate is 12.3% when sets the threshold to 5. The error matching rate is 6.5% when sets the threshold to 6. The error matching rate is 1.6% when sets the threshold to 7. The error matching rate is 2.8% when sets the threshold to 8. The error matching rate is 5.2% when sets the threshold to 9. The error matching rate is 9.2% when sets the threshold to 10. So we set the threshold to 7.

The experiment confirm that algorithm can effective avoid the influence of translation and rotation when we set reasonable threshold. The algorithm is a easy and effective matching algorithm.

## References

[1] Anil K. Jain, Ruud Bolle, and Sharat Pankanti, Eds., Biometrics: Personal Identification in Networked Society, Kluwer Academic Publishers, 1999.

[2] Federal Bureau of Investigation, The Science of Fingerprints: Classification and Uses. Washington, D.C.:U.S. Government Printing OEce, 1984.

[3] Zang J, Yuan J, Shi F, et al. A Fingerprint Matching Algorithm of Minutia Based on Local Characteristic[C] Natural Computation, 2008. ICNC08. Fourth International Conference on. IEEE, 2008, 4: 13-17.

[4] TICO M, KUOSMANEN P. Fingerprint matching using an orientation-based minutia descriptor [J]. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 2003, 25(8): 1009-14.

[5] JAIN A, ROSS A, PRABHAKAR S. Fingerprint matching using minutiae and texture features; proceedings of the Image Processing, 2001 Proceedings 2001 International Conference on, F, 2001 [C]. IEEE.

[6] LUO X, TIAN J, WU Y. A minutiae matching algorithm in fingerprint verification; proceedings of the Pattern Recognition, 2000 Proceedings 15th International Conference on, F, 2000 [C]. IEEE.

[7] Zhang W, Wang Y. Core-based structure matching algorithm of fingerprint verification[C] Pattern Recognition, 2002. Proceedings. 16th International Conference on. IEEE, 2002, 1: 70-74.

[8] Jia C, Xie M, Li Q. A fingerprint minutiae matching approach based on vector triangle method and ridge structure [C] Communications, Circuits and Systems, 2004. ICCCAS 2004. 2004 International Conference on. IEEE, 2004, 2: 871-875.

[9] Wen W, Qi Z, Li Z, et al. A Robust and Efficient Minutia-Based Fingerprint Matching Algorithm [C]. Pattern Recognition (ACPR), 2013 2nd IAPR Asian Conference on. IEEE, 2013: 201-205.

[10] Xiaolong Zheng. A Scheme for Minutiae Scoring and Its Application to Fingerprint Matching[C]. Proceedings of the 7th World Congress on Intelligent Control and Automation June 25 - 27, 2008, Chongqing, China

[11] R. Cappelli, D. Maio, D. Maltoni, J. L. Wayman, and A. K. Jain, Performance evaluation of fingerprint verification systems, IEEE Trans. Pattern Anal. Mach. Intell., vol. 28, no. 1, pp. 3–18, 2006