

A Matching Algorithm of Minutiae for Real Time Fingerprint Identification System

Shahram Mohammadi, Ali Frajzadeh

Abstract—A lot of matching algorithms with different characteristics have been introduced in recent years. For real time systems these algorithms are usually based on minutiae features. In this paper we introduce a novel approach for feature extraction in which the extracted features are independent of shift and rotation of the fingerprint and at the meantime the matching operation is performed much more easily and with higher speed and accuracy. In this new approach first for any fingerprint a reference point and a reference orientation is determined and then based on this information features are converted into polar coordinates. Due to high speed and accuracy of this approach and small volume of extracted features and easily execution of matching operation this approach is the most appropriate for real time applications.

Keywords—Matching; Minutiae; Reference point; Reference orientation.

I. INTRODUCTION

THE use of fingerprints as a biometric is the oldest method of personal identification [1]. There is expectation that a recent combination of factors will favor the use of fingerprints for the much larger market of personal authentication. These factors include: small and inexpensive fingerprint capture devices, fast computing hardware, recognition rate and speed to meet the needs of many applications, the explosive growth of network and Internet transactions. To find specific features for a fingerprint it is required to have a unique reference point for each fingerprint. Based on this reference point it is possible to enhance feature extraction and matching steps of fingerprint identification. This reference point will eliminate any dependency of features to rotation and displacement transformations and because of the resulting low volume features it increases the matching speed of the fingerprints. Fingerprint characteristics are divided into two general groups:

- 1-Local characteristics and
- 2-Global characteristics

Global characteristics are a general feature of any fingerprint based on which classification of fingerprints into one of defined classes is performed before matching. This will cause the matching operation to be performed easily and

with high speed. Another important fingerprint feature is the local ridge characteristics. There are two most prominent local ridge characteristics: ridge ending and ridge bifurcation. Both of them are called minutiae (Fig.1). Based on extracted features different methods have been proposed for minutiae pattern matching which are followed briefly. The approaches to fingerprint matching can be coarsely classified into three families. One family uses Correlation-based matching [2], [3] and [4]. Correlation matching is less tolerant to rotational and translational variances of the fingerprint and of extra noise in the image. Another family uses Minutiae-based matching [5], [6] and [7]. Minutiae matching are certainly the most well-known and widely used method for fingerprint matching. In general minutiae matching is considered by most to have a higher recognition accuracy. The last family uses Ridge feature-based matching [8]. Jain et al [8] proposed a local texture analysis technique where the fingerprint area of interest is tessellated with respect to the core point and Finger Code (feature) obtained. Finger Codes are not as distinctive as minutiae. These features may be used in conjunction with minutiae to increase system accuracy and robustness. The performance of a fingerprint recognition method involves a tradeoff among different performance measures: accuracy, efficiency, template size, and so on. Different applications desire different properties in the fingerprint matching algorithms (e.g., template size, matching speed, memory requirements, etc.).

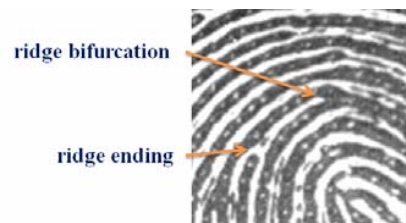


Fig. 1 Types of minutiae

II. METHODOLOGY

The proposed methodology consists of several steps. First a reference point will be obtained for fingerprint. Then reference orientation will be estimated. Feature extraction in polar coordinate is the next step. Last step is the matching.

A. Reference Point Detection

Reference Point detection is performed in several steps as follows. Based on the method proposed by Hong [9], it is possible first to find the orientation field (Fig.2 (a)). If we

Shahram Mohammadi is with Faculty of Engineering of Zanjan university, Zanjan, Iran (phone: 0098241-5152495, fax: 0098241-5152413, Email: Shahram@znu.ac.ir).

Ali Farajzadeh is with Faculty of Engineering of Zanjan University, Zanjan, Iran (Email: alikwr@gmail.com).

This research was funded in part by Iran Telecommunication Research Center ITRC.

consider the orientation field as an intensity image, the absolute angle of the ridge at each point is interpreted as a grey level (Fig.2 (b)). By applying edge detection on the obtained intensity image (F. 3(b)) a binary image is provided (Fig.2(c)). The next step is curvature measurement which is performed based on the method proposed by [10]. If the orientation field obtained by previous step is called θ then curvature measurement is derived from scattering of the orientations within a local region and is given by:

$$C = \frac{1}{N^2} \sum_{u=i-\frac{N}{2}}^{i+\frac{N}{2}} \sum_{v=j-\frac{N}{2}}^{j+\frac{N}{2}} (\phi_x(i, j, u, v) + \phi_y(i, j, u, v)) \quad (1)$$

Where

$$\phi_x(i, j, u, v) = |\cos(2\theta(i, j)) - \cos(2\theta(u, v))| \quad (2)$$

$$\phi_y(i, j, u, v) = |\sin(2\theta(i, j)) - \sin(2\theta(u, v))| \quad (3)$$

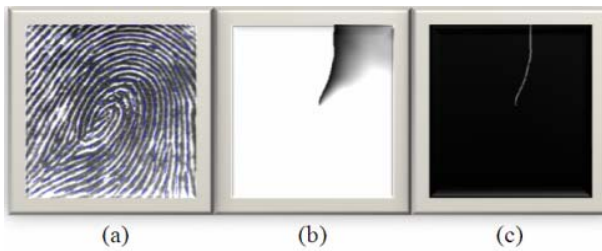


Fig. 2 Orientation Field Estimation

The higher the curvature is, the larger the value of C will be. Fig.3 illustrates the curvature field shown by the gray scale image, where higher gray level means larger curvature measurement value.

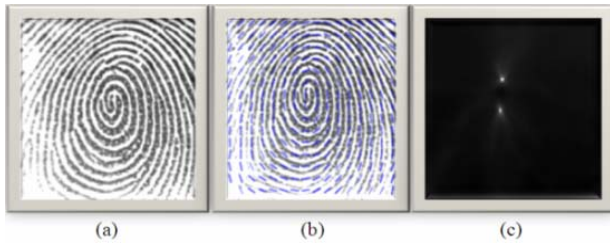


Fig. 3(a) Original fingerprint image. (b) Orientation field. (c) Curvature field displayed by gray-scale image

In this step singular points and then the reference point will be calculated. Since there is no singular point for an arch class fingerprint, so another method is used for that purpose. We first compute a maximum value for the C (see Formula 1) and call it m .

$$m = \max(c) \quad (4)$$

In the used empirical data if $m > 0.25$ then the fingerprint includes a singular point otherwise there is no singular point. Based on the computed value there are two cases:

a) *Fingerprint classes with singular points.*

For fingerprints which have singular points (Fig.4 (a)), a threshold value T_1 is defined as follows:

$$T_1 = 0.2 \times m \quad (5)$$

If C is processed according to threshold value of T_1 then Fig.4 (d) will result. The resultant image will be multiplied by edged image (Fig.4(c)), obtained from Fig. 4(b), which results in Fig.4 (e). This will provide the approximate location of singular points. By getting the average value of points in close vicinity of singular points, the exact value of singular points will be obtained (red points in Fig. 4 (f)). If just one singular point is obtained (core point that usually exists in every fingerprint) it will be considered as the reference point. Otherwise the singular point which is situated on the highest position is considered as the reference point (green points in Fig.4 (g)).

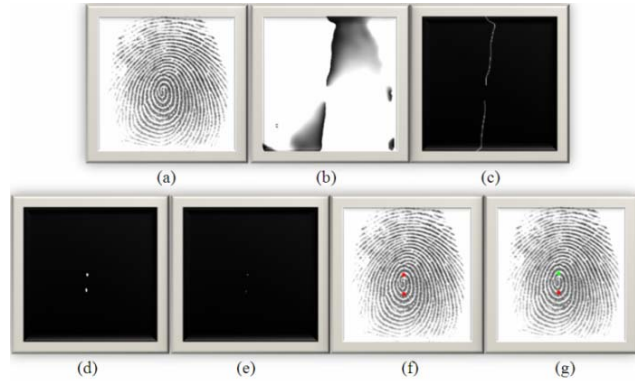


Fig. 4 Reference Point detection for fingerprint classes with singular points.

b) *Fingerprint classes without Singular points*

For fingerprints without singular points, such as arch class fingerprints, a new threshold value T_2 is defined as follows:

$$T_2 = 0.3 \times m \quad (6)$$

By performing edge detection algorithms on the intensity image Fig.5 (b), Fig.5(c) will be obtained. Dilation conversion [11], i.e. Morphological operations with structuring element=30, will then be applied to the image obtained in the previous step (Fig.5(c)). The resultant image will then be multiplied by image obtained from matrix C with threshold T_2 . Finally the maximum value of the image (resulted from the previous step) will be taken as the reference point (Green point in Fig.5 (d)).

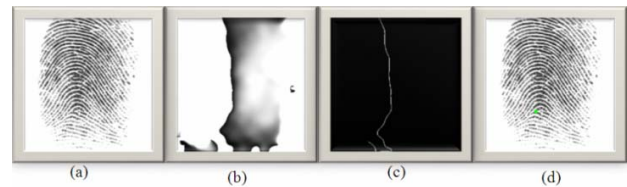


Fig. 5 Reference point detection for fingerprint classes without singular points.

To filter out those points which can be considered mistakenly as singular points and a reference point Apply a limitation on variance of the intensity input image, performing Morphological operations (Erosion conversion)[11] and choose suitable size for the low pass Gaussian filter in orientation field estimation step.

B. *Reference Orientation Computation*

The main image (Fig. 6(a)) will be enhanced by using Gabor filter and a binary image will result (Fig. 6(b)) [9].

Now consider a disk neighborhood centered at reference point which has a radius proportional to image dimensions and reference point position (Fig. 6(c)). If intensity of up core=0 (up core on the ridge) (Fig. 6 (d)); then intensity of every pixel in the previous step is complemented (Fig. 6(e)). For the resultant image from previous step a dilation conversion is performed using a squared neighborhood centered at reference point with dimension of 15×15 [11]; i.e. Morphological operations with structuring element=5 (Fig. 6(f)). Now from the resulted image all connecting points with up core are obtained (Fig. 6 (g)). Multiplication of the



resulting image with a circular mask with previous radius of one pixel thickness Fig. 6 (i) will be obtained which provides the approximate position of ends of ridges or valleys. Mean value of lengths and widths of each position will provide the exact position for each of them (red points in Fig. 6(g)). Getting the mean value of widths and lengths of red points will give a blue point. Angle of this point with respect to reference point (green point) is the reference orientation of the image (Fig. 6(j)).

Fig. 6 Reference orientation computation

C. Feature Extraction

Since during scanning the fingerprint it may have been translated and/or oriented, the extracted features must be independent of these operations. In our new method the features are extracted somehow that are independent from these operations. Features are extracted based on the reference point and reference orientation computed in previous steps. For every minutia of each fingerprint e.g. nth minutia, feature vector is extracted as follows:

$$(d_n, t_n, \theta_{1(n)}, \theta_{2(n)})$$

d_n Is the distance of nth minutiae from reference point (Fig. 7). If characteristics of the reference point are (x_{rp}, y_{rp}) and characteristics of nth minutiae are $(x_{m(n)}, y_{m(n)})$ then:

$$d_n = \sqrt{(x_{rp} - x_{m(n)})^2 + (y_{rp} - y_{m(n)})^2} \quad (7)$$

t_n is the type of minutiae which can be of ridge ending or ridge bifurcation type. For ridge ending number 1 and for bifurcation number 2 has been considered. $\theta_{1(n)}$ is:

$$\theta_{1(n)} = \theta_n - \theta_r \quad (8)$$

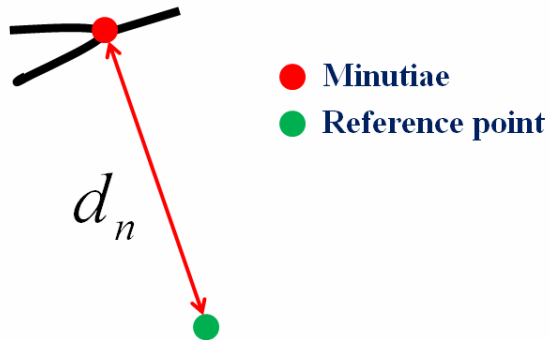


Fig. 7 Distance of nth minutiae from reference point.

In the above relation θ_n is the angle of the line connecting reference point and the nth minutiae with respect to horizontal axis and θ_r is the angle of the line connecting the reference point and reference orientation with respect to horizontal axis and both are between 0 and 2π radians (Fig. 8). For every minutia a minutiae directional angle is defined. This angle can be between 0 and 2π radians (Fig. 9). $\theta_{2(n)}$ is obtained as follows:

$$\theta_{2(n)} = \theta_{m(n)} - \theta_r \quad (9)$$

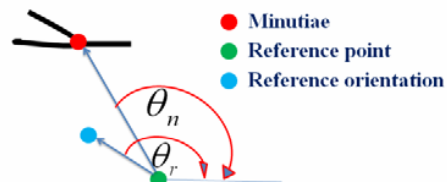


Fig. 8 Angle of minutiae respect to reference point and reference orientation.

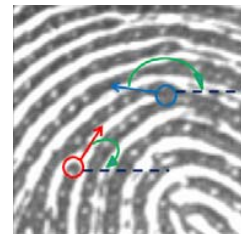


Fig. 9 Minutia directional angle (red circle is ridge ending and blue circle is ridge bifurcation)

In which $\theta_{m(n)}$ is the minutia directional angle with respect to horizontal axis and θ_r is the angle of the line connecting the reference point and reference orientation with respect to

horizontal axis and $\theta_{m(n)}$ and θ_r are between 0 and 2π radians(Fig.10).

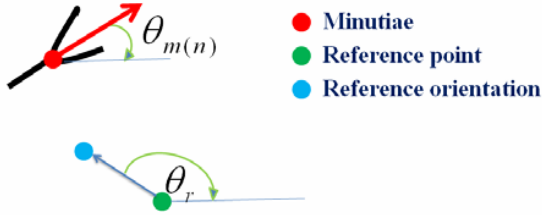


Fig. 10 Minutia directional angle with respect to reference point and reference orientation.

In addition to mentioned features which are known as local features there are other features which are known as global features (such as singular points) and the classification is performed based on type and position of the singular points[12]. These help in increasing accuracy and speed of operation and decreasing the complexity of the algorithm. Here all images are classified into five classes (Fig.11).



Fig. 11 Example Fingerprint image classes: (a) Whorl, (b) Right Loop, (c) Left Loop, (d) Arch, and (e) Tent Arch

D. Matching

Matching for each fingerprint is performed based on the extracted features. If the template fingerprint is T and the input fingerprint is I then for a template fingerprint image, T denoting the M minutiae points set as:

$$T = ((d_1^T, t_1^T, \theta_{1(1)}^T, \theta_{2(1)}^T), \dots, (d_M^T, t_M^T, \theta_{1(M)}^T, \theta_{2(M)}^T))$$

For an input fingerprint image, I denote the N minutiae set as:

$$I = ((d_1^I, t_1^I, \theta_{1(1)}^I, \theta_{2(1)}^I), \dots, (d_N^I, t_N^I, \theta_{1(N)}^I, \theta_{2(N)}^I))$$

Now to match image I with image T , features of each of minutia I is compared with features of each of minutia T . If feature of i th minutia from image I is:

$$(d_i^I, t_i^I, \theta_{1(i)}^I, \theta_{2(i)}^I)$$

and feature of minutia j th from image T is:

$$(d_j^T, t_j^T, \theta_{1(j)}^T, \theta_{2(j)}^T)$$

then matching of two minutia is as follows:

- 1- First d_j^T and d_i^I must satisfy the following condition

i.e.:

$$|d_j^T - d_i^I| \leq T_{distance} \quad (10)$$

In this relation $T_{distance}$ is the distance threshold which the value 8 can be considered for it.

- 2- The type of minutia i th and j th must be the same i.e.:

$$t_j^T = t_i^I \quad (11)$$

- 3- $\theta_{1(i)}^I$ is matched with $\theta_{1(j)}^T$ if the following is hold:

$$|\theta_{1(j)}^T - \theta_{1(i)}^I| \leq T_{1(ang)} \text{ or } |\theta_{1(j)}^T - \theta_{1(i)}^I| \geq T'_{1(ang)} \quad (12)$$

$T_{1(ang)}$ is angle threshold which its value for minutia close to reference point is about $\frac{\pi}{6}$ and for the rest, proportional to distance from the reference point, its value decreases (Fig. 12).

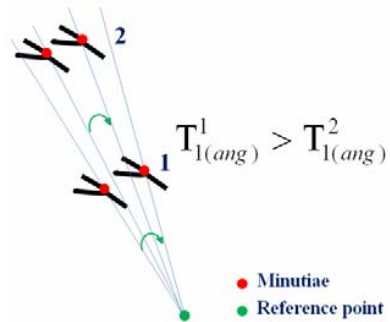


Fig. 12 Value of angle threshold decreases with increasing of distance from the reference point.

Since angles 0 and 2π radians show the same position so for matching in this case another threshold angle called $T'_{1(ang)}$ is used which its value for minutia close to reference point is about 1.9π radians and for the rest, proportional to the distance from minutia, its value is increased (final value is 1.99π radians).

- 4- $\theta_{2(i)}^I$ and $\theta_{2(j)}^T$ are matched if the following is hold:

$$|\theta_{2(j)}^T - \theta_{2(i)}^I| \leq T_{2(ang)} \text{ or } |\theta_{2(j)}^T - \theta_{2(i)}^I| \geq T'_{2(ang)} \quad (13)$$

$T_{2(ang)}$ and $T'_{2(ang)}$ are other angle threshold that for the first about $\frac{\pi}{6}$ radians and for the second about 1.9π radians is considered.

If the total minutia of the two images examined with this algorithm and the number of matched minutia is M_{mn} and the total number of minutia of image I is N and that for image T is M then:

$$K = \frac{M_{mn}}{\sqrt{M \times N}} \quad (14)$$

If K is higher than a threshold value then I is matched with T otherwise it doesn't matched. Depending on different applications this threshold value can be selected a number between 0 and 1.

III. EXPERIMENTAL RESULT

To evaluate the algorithm we have used the method proposed in [13] and [7]. Evaluation is performed using the FVC2004 fingerprint database DB1 set A. There are 100 fingerprints including 5 images from each 20 fingers. First of all, for every fingerprint in the database, it is considered as an input fingerprint image and it is matched with other 4 coming from the same finger. Denote *reject_num* as the number of rejected times. If the matching score is lower than a threshold then increase *reject_num* by one. Denote the *match_num* as the number of total matching times. Now FRR (False Reject Rate) is defined as follows:

$$FRR = \frac{\text{reject_num}}{\text{match_num}} \times 100\% \quad (15)$$

In the second place, for every fingerprint in the database, match it with other fingerprint (100-5) coming from other fingers. Denote *accept_num* as the number of false accepted times. If the matching score is higher than a threshold, then increase *accept_num* by one. Denote the *match_num* as the number of total matching times. FAR (False Accept Rate) is defined as follows:

$$FAR = \frac{\text{accept_num}}{\text{match_num}} \times 100\% \quad (16)$$

To evaluate our method we refer to the methods proposed in [14], [15] and [16] which are called standard minutiae methods. The result of evaluating the algorithm is shown in Table 1. As it can be seen the proposed method is superior to standard methods both from speed and accuracy point of views and this is in spite of existence of noise in the used database.

TABLE I
THE PERFORMANCE OF PROPOSED MATCHING ALGORITHM COMPARED
WITH STANDARD MATCHING ALGORITHM.

	FRR	FAR	Time
Standard methods	17.09%	0.84%	98ms
Proposed method	1.37%	0.50%	24ms

IV. CONCLUSION

In this paper a new method for fingerprint matching was presented which is more robust to shift and rotation of the fingerprints while it is of high accuracy. Due to low volume of extracted features and low complexity of the matching algorithm this method is of high speed and so makes it very appropriate for real time applications. In our proposed approach we have also used new methods for reference point and reference orientation detections which are of high accuracy.

REFERENCES

- [1] A. K. Jain, R. Bolle and S. Pankanti, "Biometrics- personal identification in networked society," Kluwer Academic Publishers, New York, 2002.
- [2] R. Bahuguna, "Fingerprint verification using hologram matched filterings," Biometric Consortium Eighth Meeting, San Jose, CA, 1996.
- [3] L. Coetzee and E.C. Botha, "Fingerprint recognition in low quality images," Pattern Recognition, vol. 26, no. 10, 1993, pp. 1441-1460.
- [4] R. A. Marsh, and G. S. Petty, "Optical Fingerprint Correlator," US Patent 5050220, 1991.
- [5] M. Tico and P. Kuosmanen, "Fingerprint matching using an orientation-based minutia descriptor," IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 25, no. 8, 2003, pp. 1009-1014.
- [6] A.K. Jain, L. Hong and R. Bolle, "On-line fingerprint verification," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 19, no. 4, 1997, pp. 302-314.
- [7] Y. Jie, Y. Yifang, Z. Renjie and S. Qifa, "Fingerprint minutiae matching algorithm for real time system," Pattern Recognition, 2006, pp. 143-146.
- [8] A.K. Jain, S. Prabhakar, L. Hong and S. Pankanti, "Filterbank-based fingerprint matching," IEEE Trans. Image Processing, 9(5):846-859, 2000.
- [9] L. Hong, Y. Wan and A.K. Jain, "Fingerprint image enhancement: Algorithm and performance evaluation," IEEE Trans. Machine Intelligence, 20 (8), 1998, 777-789.
- [10] W. Wang, J. Li, F. Huang and H. Feng, "Design and implementation of Log-Gabor filter in fingerprint image enhancement," Pattern Recognition Letters 29 (2008), pp. 301-308.
- [11] R. C. Gonzalez, R. E. Wood and S. Leddins, "Digital image processing using matlab," Pearson Prentice Hall, New Jersey, 2004.
- [12] A. K. Jain and S. Pankanti, "Fingerprint classification and matching," In A. Bovik, Ed., Handbook for Image and Video Processing. Academic Press, 2000.
- [13] Z. W. Bo, N. X. Bao and W. C. Jian, "A fingerprint matching algorithm based on relative topological relationship among minutiae," IEEE Int. Conference Neural Networks & Signal Processing Zhenjiang, China, 2008.
- [14] Y. He, J. Tian, X. Luo and T. Zhang, "Image enhancement and minutiae matching in fingerprint verification," Pattern Recognition Letters, no. 24, 2003, pp. 1349-1360.
- [15] N. K. Ratha, K. Karu, S. Chen and A. K. Jain, "A Real-Time Matching System for Large Fingerprint Databases," PAMI, vol. 18, no. 8, 1996, pp. 799-813.
- [16] J. Yang, L. Liu and T. Jiang, "An Efficient Fingerprint Matching Algorithm for Integrated Circuit Cards," Journal of Computer Science and Technology, vol. 19, no. 4, 2004, pp. 510-520.

Shahram Mohammadi received the BSc in Electronic Engineering from Iran University of Science and Technology IUST in 1977, MSc in Digital Techniques from Heriot-Watt University in UK in 1986 and PhD in Fault-Tolerant computing from Manchester University in 1990. From then on he was working as a lecturer and researcher in Zanjan University and Iran Telecommunications Research center respectively.

His research interests include Dependable systems, computer networking and pattern recognition.

Ali Farajzadeh received his BSc in Electronic Engineering from Urmieh University Iran in 2001 and currently doing his MSc in Zanjan University. His research interests include pattern recognition and Image processing.