IJREAT International Journal of Research in Engineering & Advanced Technology, Volume 2, Issue 1, Feb-Mar, 2014
ISSN: 2320 - 8791
www.ijreat.org

Minutiae Points Extraction using Biometric Fingerprint-Enhancement

Vishal Wagh[1], Shefali Sonavane[2]

[1] **Computer Science and Engineering Department, Walchand College of Engineering, Sangli, Maharashtra-416415, India**

[2] **Computer Science and Engineering Department, Walchand College of Engineering, Sangli, Maharashtra-416415, India**

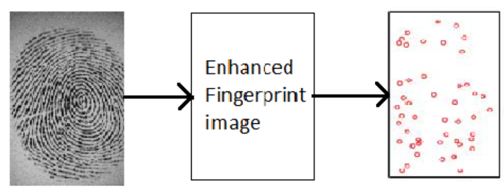
Abstract
Security mechanisms always insist the wide usage of fingerprints as common biometrics. The various features of the fingerprint like Minutiae points, Orientation, Reference points and arch's are always the focus of the concern. However, if the database of the stored fingerprints is hacked; the entire authentication system will be collapsed. Hence, to provide security to the security marks i.e. fingerprints, it becomes essential to encode those by any of the means. One of the solutions suggests working with the combined fingerprint template, which can be built by the fusion of the test fingerprints. In this direction, security researchers try to design crypt complex fingerprint template by extracting features from fingerprints and minutiae structures. In fingerprint combination systems, the feature extraction and its correct orientation is necessary. The preprocessing steps in terms of segmentation, normalization, orientation estimation, binarization, thinning and minutiae points' extraction are discussed in this paper.

Keywords: *Fingerprint, Enhancement, Minutiae Extraction, Protection.*

1. Introduction

Fingerprints are important parameters used for personal identification . Fingerprint identification is used in criminal investigations; and is widely used for biometric identification and verification. Unique fingerprint is determined by the local ridge characteristics [1]. A fingerprint is an unique pattern of the ridges and valleys on the surface of the individual finger [4]. A ridge is a single curved or line shape segment and a valley is region between two adjacent ridges. 'Ridge ending' is the point where a ridge ends and 'Ridge bifurcation' is the point where a ridge divides into branch ridges. A good fingerprint contains about 40 to 100 minutiae. The various features of the point representations are present. To identify these points and features from fingerprint image priority is to find Minutiae Positions, Pre-processing is the important step which denoises the image and concentrates on the target contents [2]. In this attempt, various pre-processing methods and extraction of minutiae positions are proposed in [3].

Fig.1 represents
a) Original biometric fingerprint image and
b) Extracted minutiae positions



a) b)

Fig. 1 Minutiae Point Extraction from fingerprint image

The remainder of this paper is organized as follows
Section 2: Definitions
Section 3: Literature review
Section 4: Proposed methodology
Section 5: Conclusions

2. Definitions

2.1 Pre-processing

Following five steps procedure is suggested under pre-processing task:
1) Segmentation
2) Normalization
3) Orientation and ridge frequency estimation
4) Binarization
5) Thinning

2.2 Minutiae Points

The points connected in different shapes in thinned image are called as Minutiae positions. Minutiae Points are the tiny and unique characteristics of fingerprint ridges that are used for positive identification [2][5][6][13].
Features of Minutiae Points

1) Ridge Ending
2) Ridge Bifurcation
3) Isolated Point
4) Normal Ridge
5) Crossing Point

3. Literature Review

Fingerprint image enhancement [3] undergoes with removal of noise, improving clarity of image smoothening, thinning and extracting the required features. To smooth the image, filter is applied in orientation estimation like Gabor filter. The algorithm discussed in [6] is a very ground idea about how minutiae points are extracted from thin fingerprint image.
VeriFinger6.3 SDK; a freeware is simultaneously operated for the same purpose to compare and verify the obtained result from algorithm [12]. FVC_2002 fingerprint database [8] is used as test sample. In Fingerprint image enhancement [2] the image is normalized using mean and variance and then orientation estimation algorithm is applied to the normalized image.

4. Methodology

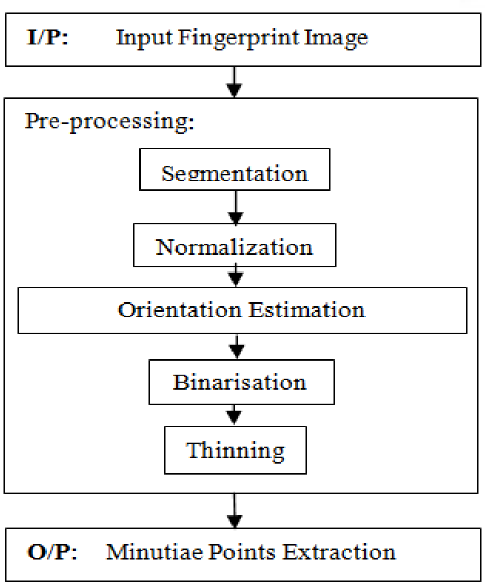


Fig. 2 Detailed input and output of proposed work

4.1 Pre-processing

1. Segmentation

It separates the foreground region from background region. The foreground area is the area of fingerprint ridges and valleys and background region is the area outside the boundary ridges of fingerprint. Here, the grey-scale variance is calculated, if it is greater than threshold then foreground region is assigned otherwise background region is declared.

$$V(k) = \frac{1}{w^2}\sum_{i=0}^{w-1}\sum_{j=0}^{w-1}(I(i,j) - M(k))^2 \quad (1)$$

where V(k) is the variance for block k, I(i;j) is the grey-level value at pixel(i;j), and M(k) is the mean grey-level value for block k [3].

2. Normalization

Normalization standardize the intensity value of an image by adjusting the range of grey-level value of N(i,j) pixel so that it lies between specific range of values. It is the process which removes the noise from an image.

$$N(i,j) = \begin{cases} M_0 + \sqrt{\frac{VAR_0(I(i,j)-M)^2}{VAR}} & I(i,j) > M \\ M_0 - \sqrt{\frac{VAR_0(I(i,j)-M)^2}{VAR}} & otherwise \end{cases} \quad (2)$$

$$M(I) = \frac{1}{N^2}\sum_{i=0}^{N-1}\sum_{j=0}^{N-1} I(i,j) \quad (3)$$

$$VAR(I) = \frac{1}{N^2}\sum_{i=0}^{N-1}\sum_{j=0}^{N-1}(I(i,j) - M(I))^2 \quad (4)$$

where N(i,j) is normalized gray level value at I(i,j), M_0 and VAR_0 are desired mean and variance [3].

3. Orientation Estimation

The orientation of a fingerprint image represents the directionality of ridges or it represents as a slope of line. It is required to match the fingerprint impression with authenticated fingerprint database. Because, if the direction of a fingerprint impression is not aligned, there may be chances that same fingerprint impression can vary to different orientation which may reject fingerprint authentication.i.e. It defines the local orientation of ridge

in an image. In every image, the ridges' flows are in different directions [7]. Filtering is applied to smooth the image using Gabor Filter, which improves or enhances the image to a finer structure. The overall description is given in [2] and [3]. In Fingerprint image enhancement [5] orientation estimation algorithm is proposed. For orientation estimation, gray level fingerprint image normalized using it's mean (M) and variance VAR(I).
There are five steps for orientation estimation:

1) Divide N into blocks of size w×w (16×16).
2) Compute the gradients $\partial_x(i,j)$ and $\partial_y(i,j)$ at each pixel (i,j).
3) Estimate the local orientation at each block centered at pixel (i,j).
4) Remove noise applying low pass filter [5].
5) Compute the local ridge orientation at (i,j) using

$$O(i,j) = \frac{1}{2}\tan^{-1}\left(\frac{\oint'_y(i,j)}{\oint'_x(i,j)}\right) \quad (5)$$

4. Binarisation

Binarization converts grey-level image into binary image. Binary image has only two level of interest; one is black pixels that are ridges and white pixels that are valleys. To improve contrast between ridges and valleys binarization is performed using global threshold; if pixel value is greater than threshold then it is set to one otherwise set as zero [3].

$$S_i = \begin{cases} 255 \text{ if } \dfrac{1}{8}\sum_{\substack{x=0 \\ x\neq i}}^{8} S_x \geq M_j \\ 0 \quad \text{otherwise} \end{cases} \quad (6)$$

where, S_i is the center pixel, S_x is foreground pixel and M_j is threshold.

5. Thinning

It is the operation that successively reduces the foreground pixels to one pixel wide. Using thinning algorithm [10] binary image is converted into thinned image. Thin fingerprint image has the connection between ridge structures from skeletonized version of the binary image [9]. Using thin fingerprint image proposed system extracts minutiae features. In paper [13] all fingerprint enhacement process is decribed with minutiae extraction.

4.2 Minutiae Points

Minutia Point has number of attributes; like its location, orientation, type (ridge termination, ridge bifurcation, etc.) [11]. Minutiae Point type is called as Crossing Number (CN). Crossing Numbers relating various properties are described in table 1.

Table 1: Property types of Crossing Number

CN	Property
0	Isolated point
1	Ridge ending point
2	Continuing ridge point
3	Bifurcation point
4	Crossing point

The CN value is calculated, which is the half sum of the differences of pairs between adjacent pixels in the eight neighborhoods [6].

$$CN = 0.5\sum_{i=1}^{8}|P_i - P_{i+1}| \ , \ P_9 = P_1 \quad (7)$$

Table 2: Rules for Calculating Crossing Number

Supporting Process for feature Extraction		
Rule No.	***Filter/Mask***	***Feature Description***
Rule 1	0 0 0 / 0 1 0 / 0 0 0	If central pixel is 1 and has exactly no (or zero) one-value neighbors, then central pixel is an **isolated point**
Rule 2	0 0 0 / 0 1 0 / 0 0 1	If central pixel is 1 and has exact 1 one-value across neighbors, then central pixel is **ridge Termination**
Rule 3	0 0 1 / 0 1 0 / 1 0 0	If central pixel is 1 and has exact 2 one-value across neighbors, then central pixel is **normal ridge**

Rule 4	0 1 0 0 1 0 1 0 1	If central pixel is 1 and has exact 3 one- value across neighbors, then central pixel is **ridge bifurcation**.
Rule 5	1 0 1 0 1 0 1 0 1	If central pixel is 1 and has exact 4 one-value across neighbors, then central pixel is **crossing point**

Table 2 describes the rules for calculating crossing number in which the center pixel value is calculated based on its foreground pixels. Various filters are applied to find crossing number CN.

Table 3: (3X3) Pixel matrix representation

P4	P3	P2
P5	P	P1
P6	P7	P8

Table 3 describes the center pixel P value is calculated based on its foreground pixel P1 to P9 (P1=P9).

Table 4: Snapshot results for Minutiae point extraction methods

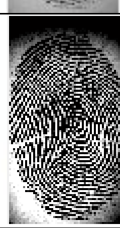
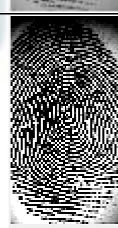
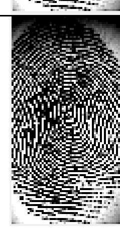
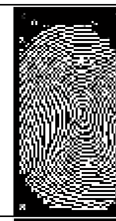
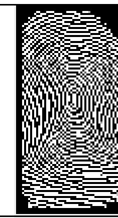
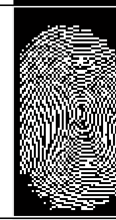
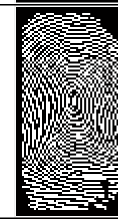
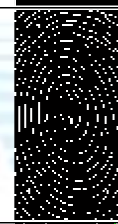
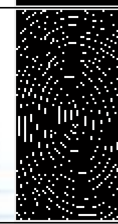
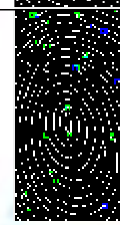
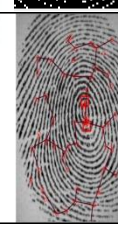
Preprocessing steps for Minutiae Extraction	***Fingerprint Image 1***	***Fingerprint Image 2***
Original Images		
Segmentation		
Normalization		
Enhancement		
Binarization		
Thinning		
Minutiae Points Extraction using proposed method		
Minutiae Points Extraction using VeriFinger6.3 SDK Tool		

Table 4 shows fingerprint enhancement pre-processing methods in the form of snapshot of results. Different fingerprint images are tested for experimentation based on the pre-processing steps shown in figure 1 then output in the form of minutiae points is obtained. The result obtained using algorithm [8] is compared with the results extracted by VeriFinger6.3 SDK [12] and it is observed that both the results are similar.

5. Conclusions

It is necessary to pass the test biometric fingerprint information through preprocessing to extract the required noiseless information which is useful for feature extraction. Once the preprocessing is performed; the accuracy for estimating correct minutiae feature increases. Ridge Endings and Ridge Bifurcations minutiae points are most commonly used features for minutiae point's extraction.

Further, the attempts can be made to ‘fix’ the orientation of the fingerprint. It is required to have same orientation and size to fuse the test fingerprints to create the combined template. This designed combined template needs to be completely different than the original fingerprint; which can be the remarkable attempt in the field of biometric security.

References

[1] D. Maltoni, D. Maio, A. K. Jain, and S. Prabhakar, "Handbook of Fingerprint Recognition", Springer Verlag, June 2003.
[2] L. Hong, Y. F. Wan, and A. Jain, “Fingerprint image enhancement: Algorithm and performance evaluation”, IEEE Trans. Pattern Anal. Mach. Intell., vol. 20, no. 8, pp. 777–789, Aug. 1998.
[3] A.B.Kayode, Olabode Olatubosun and A.O.Charles, “Fingerprint image enhancement: “Segmentation to Thinning”,(IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 3, No. 1, 2012
[4] Raffaele Cappelli and Alessandra Lumini, “Fingerprint Classification by Directional Image Partitioning”, IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, VOL. 21, NO. 5, MAY 1999.
[5] M. Gamassi, V. Pivri and F. Scotti, “Fingerprint local analysis for high performance minutiae extraction”, IEEE International Conference on Image Processing (ICIP) vol. 3, 2005, pp. 265-268.
[6] Sunny Arief SUDIRO and Maulana KUSUMA , “Simple Fingerprint Minutiae Extraction Algorithm Using Crossing Number On Valley Structure”, IEEE Trans. Pattern Anal. Mach. Intell,july 2007.
[7] Y. Wang and J. Hu, “Global ridge orientation modeling for partial fingerprint identification,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 33, no. 1, pp. 72–87, Jan. 2011.
[8] bias.csr.unibo.it/fvc2002/.
[9] F. Zhao and X. Tang, "Pre-processing for skeleton-based fingerprint minutiae extraction", Proc. Int'l Conf Imaging Science, Systems, and Technology (CISST), pp. 742-745, June 2002.
[10]Sunny Arief Sudiro, " Thinning Algorithm for Image Converted in Fingerprint Recognition System", National Seminar of Soft Computing, Intelligent Systems & Information Technology 2005, Petra University.
[11]S. Kasaei, M. Deriche, B. Baashash, "Fingerprint Feature Extraction Using Block-Direction on Reconstructured Images", IEEE Tencon(Spech and Image Technologies for Computing and Telecommunications), Brisbane Australia, pp. 303-306. 1997.
[12]VeriFinger6.3 SDK [Online] Available: http://www.neurotechnology.com
[13]R. Thai: Fingerprint Image Enhancement and Minutiae Extraction, PhD Thesis Submitted to School of Computer Science and Software Engineering, 2003, University of Western Australia.

.