ISSN : 2231-0711
Mayank Tripathi et al | IJCSET(www.ijcset.net) | May 2015 | Vol 5, Issue 5,120-126

Designing of Fingerprint Recognition System Using Minutia Extraction and Matching

Mayank Tripathi [#1], Deepak Shrivastava [*2]

[#1] *M. Tech Scholar, Dept. of CSE, Disha Institute of Management and Technology, Raipur, India*

[*1] *Assistant Professor, Dept. of CSE, Disha Institute of Management and Technology, Raipur, India*

Abstract— **A biometric system provides automatic identification of an individual based on a unique feature or characteristic possessed by the individual. This work deals with the development of a highly robust and efficient biometric person identification system based on fingerprint features. Human fingerprints are rich in details called minutiae, which can be used as identification marks for fingerprint verification. The goal of this project is to develop a complete system for fingerprint verification through extracting and matching minutiae. To achieve good minutiae extraction in fingerprints with varying quality, pre-processing in form of image enhancement and binarization is first applied on fingerprints before they are evaluated. Many methods have been combined to build a minutia extractor and a minutia matcher. Minutia-marking with false minutiae removal methods are used in the work. An alignment-based elastic matching algorithm has been developed for minutia matching. This algorithm is capable of finding the correspondences between input minutia pattern and the stored template minutia pattern without resorting to exhaustive search. The developed work utilizes finger print minutia as a feature for finger print identification. The software platform used for the implementation of the research work is MATLAB. Total 80 fingerprint images have been used to present the effectiveness of the developed work from fingerprint database of the FVC2000 (Fingerprint Verification Competition 2000). After the complete analysis it is found that the fingerprint recognition efficiency of the developed system is very high about 95 %, which is quit higher to the available fingerprint recognition techniques.**

Keywords— **Biometric system, fingerprint, minutia, feature extraction, feature matching.**

I. INTRODUCTION

A fingerprint is the feature pattern of one finger (Fig.1). It is believed with strong evidences that each fingerprint is unique. Each person has his own fingerprints with the permanent uniqueness. So fingerprints have being used for identification and forensic investigation for a long time.

Fig.1. A fingerprint image acquired by an Optical Sensor

A fingerprint is composed of many ridges and furrows. These ridges and furrows present good similarities in each small local window, like parallelism and average width. However, shown by intensive research on fingerprint recognition, fingerprints are not distinguished by their ridges and furrows, but by Minutia, which are some abnormal points on the ridges (Fig.2).

Among the variety of minutia types reported in literatures, two are mostly significant and in heavy usage: one is called termination, which is the immediate ending of a ridge; the other is called bifurcation, which is the point on the ridge from which two branches derive.

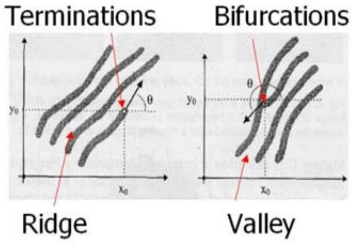


Fig.2 A view of Minutia.

A. *Foundations of Fingerprint Recognition*

The fingerprint recognition problem can be grouped into two sub-domains: one is fingerprint verification and the other is fingerprint identification (Fig.3).

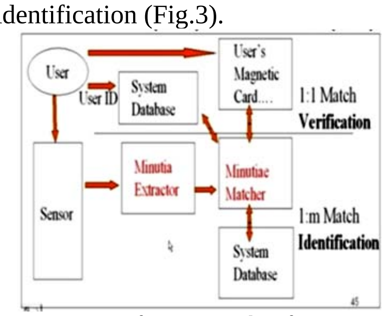


Fig.3 Verification vs. Identification

Fingerprint verification is to verify the authenticity of one person by his fingerprint. The user provides his fingerprint together with his identity information like his ID number. The fingerprint verification system retrieves the fingerprint template according to the ID number and matches the template with the real-time acquired fingerprint from the user.

Fingerprint identification is to specify one person's identity by his fingerprint(s). Without knowledge of the person's identity, the fingerprint identification system tries to match his fingerprint(s) with those in the whole fingerprint database.

II. PROPOSED METHODOLOGY AND SYSTEM DESIGN

A. *System Level Design*

A fingerprint recognition system constitutes of fingerprint acquiring device, minutia extractor and minutia matcher [Fig.4].

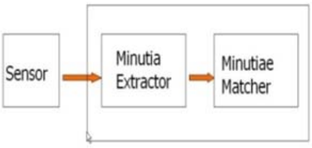


Fig.4 Simplified Fingerprint Recognition System

For fingerprint acquisition, optical or semi-conduct sensors are widely used. They have high efficiency and acceptable accuracy except for some cases that the user's finger is too dirty or dry. However, the testing database for project is from the available fingerprints provided by FVC2002 (Fingerprint Verification Competition 2002). So no acquisition stage is implemented. The minutia extractor and minutia matcher modules are explained in detail in the next part for algorithm design and other subsequent sections.

B. Algorithm Level Design

To implement a minutia extractor, a three-stage approach is widely used by researchers. They are preprocessing, minutia extraction and post processing stage [Fig.5].

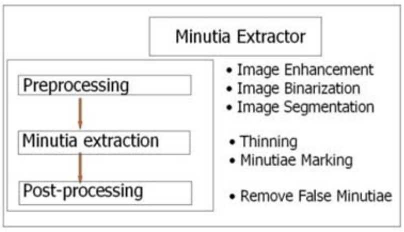


Fig.5 Minutia Extractor

For the fingerprint image preprocessing stage, this work utilized Histogram Equalization and Fourier Transform to do image enhancement [9]. And then the fingerprint image is binarized using the locally adaptive threshold method [12]. The image segmentation task is fulfilled by a three-step approach: block direction estimation, segmentation by direction intensity [4] and Region of Interest extraction by Morphological operations. The morphological operations for extraction ROI are introduced to fingerprint image segmentation. For minutia extraction stage, Morphological thinning operation is utilized. The minutia marking is a simple task as most literatures reported but one special case is found during implementation and an additional check mechanism is enforced to avoid such kind of oversight.

For the post processing stage, a more rigorous algorithm is developed to remove false minutia based on [12][1]. Also a novel representation for bifurcations is proposed to unify terminations and bifurcations.

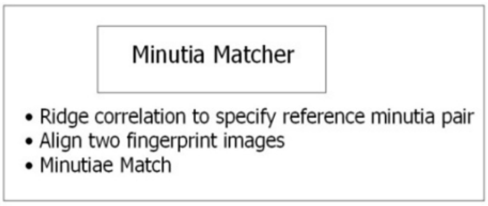


Fig.6 Minutia Matcher

The minutia matcher chooses any two minutia as a reference minutia pair and then matches their associated ridges first. If the ridges match well [1], two fingerprint images are aligned and matching is conducted for all remaining minutia [Fig.6].

C. Fingerprint Image Pre-processing

1) Fingerprint Image Enhancement

Fingerprint Image enhancement is to make the image clearer for easy further operations. Since the fingerprint images acquired from sensors or other Medias are not assured with perfect quality, those enhancement methods, for increasing the contrast between ridges and furrows and for connecting the false broken points of ridges due to insufficient amount of ink, are very useful for keep a higher accuracy to fingerprint recognition. Two Methods are adopted in this work fingerprint recognition system: the first one is Histogram Equalization; the next one is Fourier Transform.

Histogram Equalization:

Histogram equalization is to expand the pixel value distribution of an image so as to increase the perceptional information. The original histogram of a fingerprint image has the bimodal type [Fig.7], the histogram after the histogram equalization occupies all the range from 0 to 255 and the visualization effect is enhanced [Fig.8].

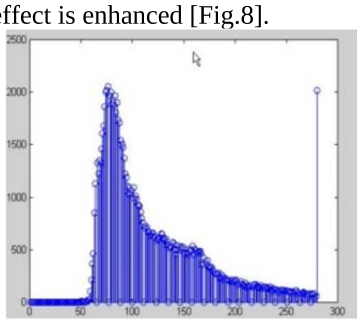


Fig.7 The Original histogram

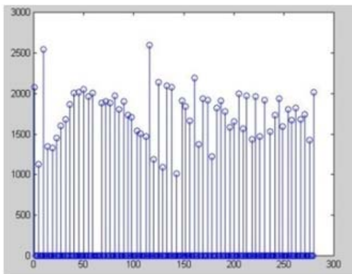


Fig.8 Histogram after the Histogram Equalization

The right side of the following Figure [Fig.9] is the output after the histogram equalization.

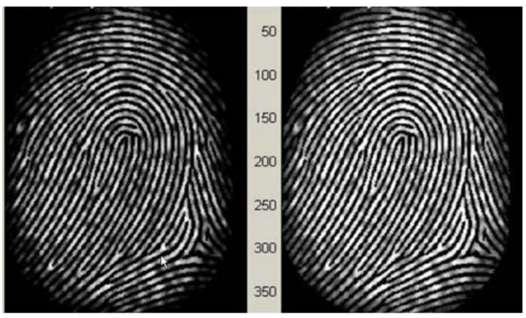


Fig.9 Histogram Enhancement, Original Image (Left), Enhanced image (Right)

Fingerprint Enhancement by Fourier Transform

The image is divided into small processing blocks (32 by 32 pixels) and performs the Fourier transform according to:

$$F(u,v)=\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)\times \exp\left\{-j2\pi\times\left(\frac{ux}{M}+\frac{vy}{N}\right)\right\} \quad ...(1)$$

For u = 0, 1, 2... 31 and v = 0, 1, 2... 31.

In order to enhance a specific block by its dominant frequencies, we multiply the FFT of the block by its magnitude a set of times. Where the magnitude of the original FFT = abs (F (u,v)) = |F(u,v)|. Get the enhanced block according to

$$g(x,y)=F^{-1}\left\{F(u,v)\times\left|F(u,v)\right|^{K}\right\} \quad ...(2)$$

Where F^{-1}(F(u,v)) is done by:

$$f(x,y)=\frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} F(u,v)\times \exp\left\{j2\pi\times\left(\frac{ux}{M}+\frac{vy}{N}\right)\right\} \quad ...(3)$$

For x = 0, 1, 2... 31 and y = 0, 1, 2... 31.

The k in formula (2) is an experimentally determined constant, which we choose k=0.45 to calculate. While having a higher "k" improves the appearance of the ridges, filling up small holes in ridges, having too high a "k" can result in false joining of ridges. Thus a termination might become a bifurcation. Fig.10 presents the image after FFT enhancement.

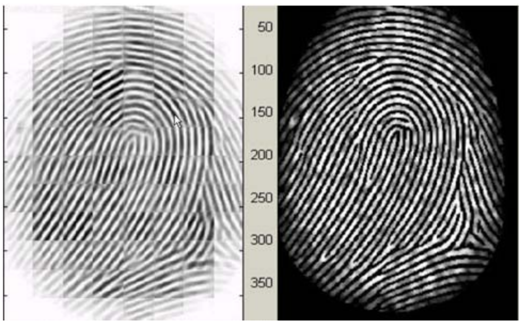


Fig.10 Fingerprint enhancement by FFT Enhanced image (left), Original image (right)

The enhanced image after FFT has the improvements to connect some falsely broken points on ridges and to remove some spurious connections between ridges. The shown image at the left side of Fig.10 is also processed with histogram equalization after the FFT transform.

Fingerprint Image Binarization

Fingerprint Image Binarization is to transform the 8-bit Gray fingerprint image to a 1-bit image with 0-value for ridges and 1-value for furrows. After the operation, ridges in the fingerprint are highlighted with black color while furrows are white.

A locally adaptive binarization method is performed to binarize the fingerprint image. Such a named method comes from the mechanism of transforming a pixel value to 1 if the value is larger than the mean intensity value of the current block (16x16) to which the pixel belongs [Fig.11].

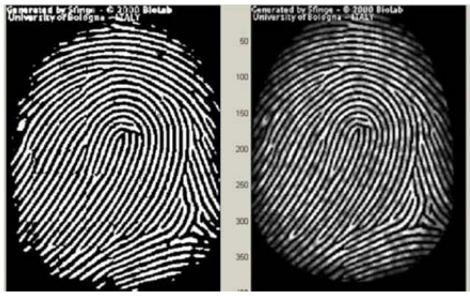


Fig.11 the Fingerprint image after adaptive binarization Binarized image (left), Enhanced gray image (right).

2) Fingerprint Image Segmentation

In general, only a Region of Interest (ROI) is useful to be recognized for each fingerprint image. The image area without effective ridges and furrows is first discarded since it only holds background information. Then the bound of the remaining effective area is sketched out since the minutia in the bound region is confusing with those spurious minutia's that are generated when the ridges are out of the sensor. To extract the ROI, a two-step method is used. The first step is block direction estimation and direction variety check [1], while the second is intrigued from some Morphological methods.

Block direction estimation

i. Estimate the block direction for each block of the fingerprint image with WxW in size (W is 16 pixels by default). The algorithm is:

Calculate the gradient values along x-direction (g_x) and y-direction (g_y) for each pixel of the block. Two Sobel filters are used to fulfill the task.

ii. After finished with the estimation of each block direction, those blocks without significant information on ridges and furrows are discarded based on the following formulas:

$$E = \{2\sum\sum(g_x*g_y)+\sum\sum(g_x^2-g_y^2)\}/W*W*\sum\sum(g_x^2+g_y^2)$$

For each block, if its certainty level E is below a threshold, then the block is regarded as a background block. The direction map is shown in the following diagram. It is assumed there is only one fingerprint in each image.

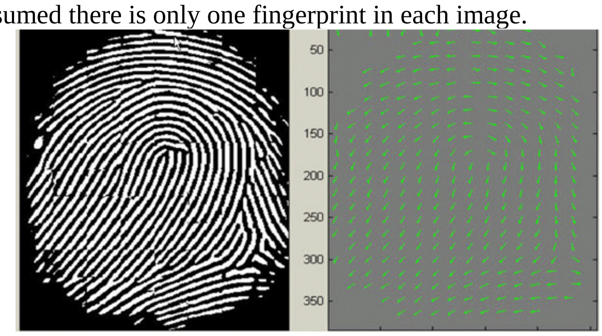


Fig.12 Direction map. Binarized fingerprint (left), Direction map (right)

ROI extraction by Morphological operations

Two Morphological operations called 'OPEN' and 'CLOSE' are adopted. The 'OPEN' operation can expand images and remove peaks introduced by background noise [Fig.13]. The 'CLOSE' operation can shrink images and eliminate small cavities [Fig.14].

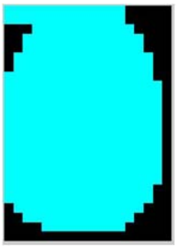


Fig.13 Original Image Area

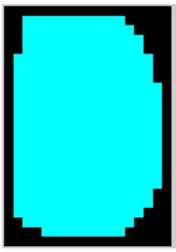


Fig.14 After CLOSE operation

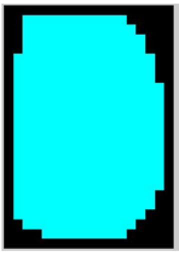


Fig.15 After OPEN operation

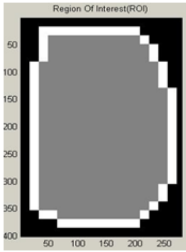


Fig.16 ROI + Bound

Fig.16 shows the interest fingerprint image area and it's bound. The bound is the subtraction of the closed area from the opened area. Then the algorithm throws away those leftmost, rightmost, uppermost and bottommost blocks out of the bound so as to get the tightly bounded region just containing the bound and inner area.

3) *Minutia Extraction*

Fingerprint Ridge Thinning

Ridge Thinning is to eliminate the redundant pixels of ridges till the ridges are just one pixel wide. [12] Uses an iterative, parallel thinning algorithm. In each scan of the full fingerprint image, the algorithm marks down redundant pixels in each small image window (3x3). And finally removes all those marked pixels after several scans. In our testing, such an iterative, parallel thinning algorithm has bad efficiency although it can get an ideal thinned ridge map after enough scans. [2] Uses a one-in-all method to extract thinned ridges from gray-level fingerprint images directly. Their method traces along the ridges having maximum gray intensity value. However, binarization is implicitly enforced since only pixels with maximum gray intensity value are remained. Also in our testing, the advancement of each trace step still has large computation complexity although it does not require the movement of pixel by pixel as in other thinning algorithms. The thinned ridge map is then filtered by other three Morphological operations to remove some H breaks, isolated points and spikes.

Minutia Marking

After the fingerprint ridge thinning, marking minutia points is relatively easy. But it is still not a trivial task as most literatures declared. In general, for each 3x3 window, if the central pixel is 1 and has exactly 3 one-value neighbors, then the central pixel is a ridge branch [Fig.17]. If the central pixel is 1 and has only 1 one-value neighbor, then the central pixel is a ridge ending [Fig.18].

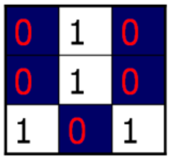


Fig.17 Bifurcation

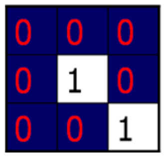


Fig.18 Termination

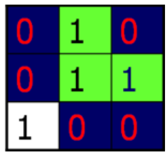


Fig.19 Triple counting branch

Fig.19 illustrates a special case that a genuine branch is triple counted. Suppose both the uppermost pixel with value 1 and the rightmost pixel with value 1 have another neighbor outside the 3x3 window, so the two pixels will be marked as branches too. But actually only one branch is located in the small region. So a check routine requiring that none of the neighbors of a branch are branches is added.

Also the average inter-ridge width D is estimated at this stage. The average inter-ridge width refers to the average distance between two neighboring ridges. The way to approximate the D value is simple. Scan a row of the thinned ridge image and sum up all pixels in the row whose value is one. Then divide the row length with the above summation to get an inter-ridge width. For more accuracy, such kind of row scan is performed upon several other rows and column scans are also conducted, finally all the inter-ridge widths are averaged to get the D.

Together with the minutia marking, all thinned ridges in the fingerprint image are labeled with a unique ID for further operation. The labeling operation is realized by using the Morphological operation: BWLABEL.

4) *Minutia Post-processing*

False Minutia Removal

The preprocessing stage does not totally heal the fingerprint image. For example, false ridge breaks due to insufficient amount of ink and ridge cross-connections due to over inking are not totally eliminated. Actually all the earlier stages themselves occasionally introduce some artifacts which later lead to spurious minutia. This false minutia will significantly affect the accuracy of matching if they are simply regarded as genuine minutia. So some mechanisms of removing false minutia are essential to keep the fingerprint verification system effective. Seven types of false minutia are specified in following diagrams:

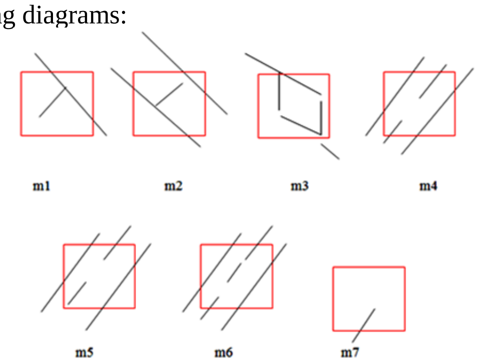


Fig.20 False Minutia Structures

Fig.20 False Minutia Structures. m1 is a spike piercing into a valley. In the m2 case a spike falsely connects two ridges. m3 has two near bifurcations located in the same ridge. The two ridge broken points in the m4 case have nearly the same orientation and a short distance. m5 is alike the m4 case with the exception that one part of the broken ridge is so short that another termination is generated. m6 extends the m4 case but with the extra property that a third ridge is found in the middle of the two parts of the broken ridge. m7 has only one short ridge found in the threshold window. [4] Only handles the case m1, m4,m5 and m6. [9] And [2] have not false minutia removal by simply assuming the image quality is fairly good. [12] Has not a systematic healing method to remove those spurious minutias although it lists all types of false minutia shown in Fig. 20 except the m3 case?

The procedures in removing false minutia are:

- If the distance between one bifurcation and one termination is less than D and the two minutias are in the same ridge (m1 case). Remove both of them. Where D is the average inter-ridge width representing the average distance between two parallel neighboring ridges.
- If the distance between two bifurcations is less than D and they are in the same ridge, remove the two bifurcations. (m2, m3 cases).
- If two terminations are within a distance D and their directions are coincident with a small angle variation. And they suffice the condition that no any other termination is located between the two terminations. Then the two terminations are regarded as false minutia derived from a broken ridge and are removed. (Case m4, m5, m6).
- If two terminations are located in a short ridge with length less than D, remove the two terminations (m7).

The proposed procedures in removing false minutia for this work have two advantages. One is that the ridge ID is used to distinguish minutia and the seven types of false minutia are strictly defined comparing with those loosely defined by other methods. The second advantage is that the order of removal procedures is well considered to reduce the computation complexity. It surpasses the way adopted by [12] that does not utilize the relations among the false minutia types. For example, the procedure3 solves the m4, m5 and m6 cases in a single check routine. And after procedure 3, the number of false minutia satisfying the m7 case is significantly reduced.

Unify terminations and bifurcations

Since various data acquisition conditions such as impression pressure can easily change one type of minutia into the other, most researchers adopt the unification representation for both termination and bifurcation. So each minutia is completely characterized by the following parameters at last: 1) x-coordinate, 2) y-coordinate, and 3) orientation.

The orientation calculation for a bifurcation needs to be specially considered. All three ridges deriving from the bifurcation point have their own direction, [9] represents the bifurcation orientation using a technique proposed in [12] [. [1] Simply chooses the minimum angle among the three anticlockwise orientations starting from the x-axis. Both methods cast the other two directions away, so some information loses. Here this work proposes a novel representation to break a bifurcation into three terminations. The three new terminations are the three neighbor pixels of the bifurcation and each of the three ridges connected to the bifurcation before is now associated with a termination respectively [Fig.21].

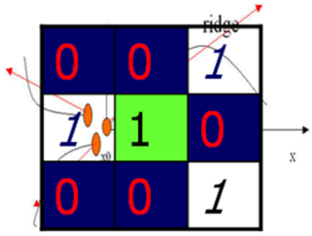


Fig.21 A bifurcation to three terminations Three neighbors become terminations (Left) Each termination has their own orientation (Right)

Track a ridge segment who's starting point is the termination and length is D. Sum up all x-coordinates of points in the ridge segment. Divide above summation with D to get sx. Then get sy using the same way. Get the direction from: atan((sy-ty)/(sx-tx)).

5) Minutia Match

Given two set of minutia of two fingerprint images, the minutia match algorithm determines whether the two minutia sets are from the same finger or not. An alignment-based match algorithm partially derived from the [1] is used in this work. It includes two consecutive stages: one is alignment stage and the second is match stage.

Alignment stage: Given two fingerprint images to be matched, choose any one minutia from each image; calculate the similarity of the two ridges associated with the two referenced minutia points. If the similarity is larger than a threshold, transform each set of minutia to a new coordination system whose origin is at the referenced point and whose x-axis is coincident with the direction of the referenced point.

Match stage: After we get two set of transformed minutia points, we use the elastic match algorithm to count the matched minutia pairs by assuming two minutia having nearly the same position and direction are identical.

Alignment Stage

i. The ridge associated with each minutia is represented as a series of x-coordinates (x_1, $x_2 \ldots x_n$) of the points on the ridge. A point is sampled per ridge length L starting from the minutia point, where the L is the average inter-ridge length. And n is set to 10 unless the total ridge length is less than 10*L.

   So the similarity of correlating the two ridges is derived from:

   $S = \sum^{m}_{i=0} x_i X_i / [\sum^{m}_{i=0} x_i^2 X_i^2]$^0.5,

   Where ($x_{i\sim}x_n$) and ($X_{i\sim}X_N$) are the set of minutia for each fingerprint image respectively. And m is minimal one of the n and N value. If the similarity score is larger than 0.8, then go to step 2, otherwise continue to match the next pair of ridges.

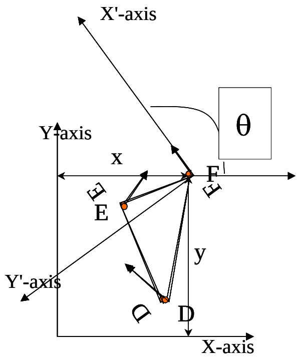
ii. For each fingerprint, translate and rotate all other minutia with respect to the reference minutia according to the following formula:

$$\begin{pmatrix} xi\_new \\ yi\_new \\ \theta i\_new \end{pmatrix} = TM * \begin{bmatrix} (xi - x) \\ (yi - y) \\ (\theta i - \theta) \end{bmatrix},$$

Where (x,y,θ) is the parameters of the reference minutia, and TM is

$$TM = \begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

The following diagram illustrates the effect of translation and rotation:



The new coordinate system is originated at minutia F and the new x-axis is coincident with the direction of minutia F. No scaling effect is taken into account by assuming two fingerprints from the same finger have nearly the same size.

The proposed approach transforms each according to its own reference minutia and then do match in a unified x-y coordinate. Therefore, less computation workload is achieved through proposed method.

Match Stage

The matching algorithm for the aligned minutia patterns needs to be elastic since the strict match requiring that all parameters (x, y, θ) are the same for two identical minutia is impossible due to the slight deformations and inexact quantization's of minutia.

The approach of this work to elastically match minutia is achieved by placing a bounding box around each template minutia. If the minutia to be matched is within the rectangle box and the direction discrepancy between them is very small, then the two minutias are regarded as a matched minutia pair. Each minutia in the template image either has no matched minutia or has only one corresponding minutia. The final match ratio for two fingerprints is the number of total matched pair over the number of minutia of the template fingerprint. The score is 100*ratio and ranges from 0 to 100. If the score is larger than a pre-specified threshold, the two fingerprints are from the same finger. However, the elastic match algorithm has large computation complexity and is vulnerable to spurious minutia.

A graphical user interface is also developed for the proposed system to make the project work user friendly. The snapshot of the developed GUI is shown below.

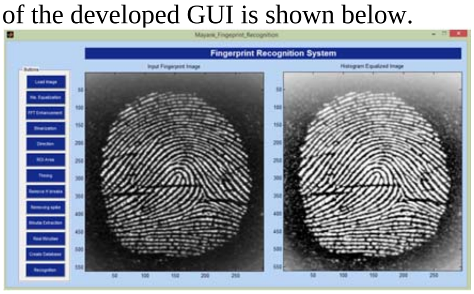


Fig.22 the snapshot of the developed GUI.

III. Experimentation Results

A. *Experimentation Results*

A fingerprint database from the FVC2000 (Fingerprint Verification Competition 2000) is used to test the experiment performance. This work tests all the images without any fine-tuning for the database. The experiments show developed program can differentiate imposturous minutia pairs from genuine minutia pairs in a certain confidence level. Furthermore, good experiment designs can surely improve the accuracy. Here table-I show the tabulated results for Correct and Incorrect fingerprint recognition of all the 80 images.

TABLE I

Tabulated results for Correct and Incorrect fingerprint recognition of all 80 images

S. No.	Fingerprint Image Name	Recognition Status	Correct Recognition	S. No.	Fingerprint Image Name	Recognition Status	Correct Recognition
1	1.tiff	Yes	Recognized	**41**	41.tiff	Yes	Not Recognized
2	2.tiff	Yes	Recognized	**42**	42.tiff	Yes	Recognized
3	3.tiff	Yes	Recognized	**43**	43.tiff	Yes	Recognized
4	4.tiff	Yes	Recognized	**44**	44.tiff	Yes	Recognized
5	5.tiff	Yes	Recognized	**45**	45.tiff	Yes	Recognized
6	6.tiff	Yes	Recognized	**46**	46.tiff	Yes	Recognized
7	7.tiff	Yes	Recognized	**47**	47.tiff	Yes	Recognized
8	8.tiff	Yes	Recognized	**48**	48.tiff	Yes	Recognized
9	9.tiff	Yes	Recognized	**49**	49.tiff	Yes	Recognized
10	10.tiff	Yes	Recognized	**50**	50.tiff	Yes	Recognized
11	11.tiff	Yes	Recognized	**51**	51.tiff	Yes	Recognized
12	12.tiff	Yes	Recognized	**52**	52.tiff	Yes	Recognized
13	13.tiff	Yes	Recognized	**53**	53.tiff	Yes	Recognized
14	14.tiff	Yes	Recognized	**54**	54.tiff	Yes	Recognized
15	15.tiff	Yes	Recognized	**55**	55.tiff	Yes	Recognized
16	16.tiff	Yes	Recognized	**56**	56.tiff	Yes	Recognized
17	17.tiff	Yes	Recognized	**57**	57.tiff	Yes	Recognized
18	18.tiff	Yes	Not Recognized	**58**	58.tiff	Yes	Recognized
19	19.tiff	Yes	Recognized	**59**	59.tiff	Yes	Recognized
20	20.tiff	Yes	Recognized	**60**	60.tiff	Yes	Recognized
21	21.tiff	Yes	Recognized	**61**	61.tiff	Yes	Recognized
22	22.tiff	Yes	Recognized	**62**	62.tiff	Yes	Recognized
23	23.tiff	Yes	Recognized	**63**	63.tiff	Yes	Recognized
24	24.tiff	Yes	Recognized	**64**	64.tiff	Yes	Not Recognized
25	25.tiff	Yes	Recognized	**65**	65.tiff	Yes	Recognized
26	26.tiff	Yes	Recognized	**66**	66.tiff	Yes	Recognized
27	27.tiff	Yes	Recognized	**67**	67.tiff	Yes	Recognized
28	28.tiff	Yes	Recognized	**68**	68.tiff	Yes	Recognized
29	29.tiff	Yes	Recognized	**69**	69.tiff	Yes	Recognized
30	30.tiff	Yes	Not Recognized	**70**	70.tiff	Yes	Recognized
31	31.tiff	Yes	Recognized	**71**	71.tiff	Yes	Recognized
32	32.tiff	Yes	Recognized	**72**	72.tiff	Yes	Recognized
33	33.tiff	Yes	Recognized	**73**	73.tiff	Yes	Recognized
34	34.tiff	Yes	Recognized	**74**	74.tiff	Yes	Recognized
35	35.tiff	Yes	Recognized	**75**	75.tiff	Yes	Recognized
36	36.tiff	Yes	Recognized	**76**	76.tiff	Yes	Recognized
37	37.tiff	Yes	Recognized	**77**	77.tiff	Yes	Recognized
38	38.tiff	Yes	Recognized	**78**	78.tiff	Yes	Recognized
39	39.tiff	Yes	Recognized	**79**	79.tiff	Yes	Recognized
40	40.tiff	Yes	Recognized	**80**	80.tiff	Yes	Recognized

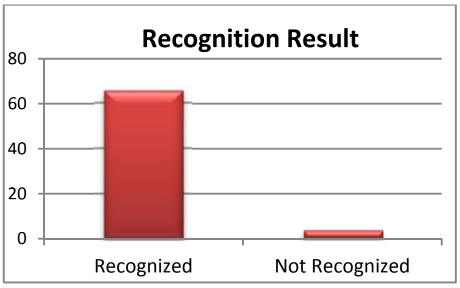


Fig.23 Recognition Result

Now Fig. 23 shows the plot of fingerprint recognition of developed fingerprint recognition system. From Fig. 23 it is clearly observable that the recognition efficiency of the developed project is very high and equal to 95 %.

IV. CONCLUSIONS

This proposed work has combined many methods to build a minutia extractor and a minutia matcher. The combination of multiple methods comes from a wide investigation into research paper. Also some novel changes like segmentation using Morphological operations, minutia marking with special considering the triple branch counting, minutia unification by decomposing a branch into three terminations, and matching in the unified x-y coordinate system after a two-step transformation are used in this project.

In addition to this, in the result section it is found that the fingerprint recognition efficiency of the developed system is very high about 95 %, which is quit higher to the available fingerprint recognition techniques.

REFERENCES

[1] Lin Hong. "Automatic Personal Identification Using Fingerprints", Ph.D. Thesis, 1998.

[2] D.Maio and D. Maltoni. Direct gray-scale minutiae detection in fingerprints. IEEE Trans. Pattern Anal. And Machine Intell., 19(1):27-40, 1997.

[3] Jain, A.K., Hong, L., and Bolle, R.(1997), “On-Line Fingerprint Verification,” IEEE Trans. On Pattern Anal and Machine Intell, 19(4), pp. 302-314.

[4] N. Ratha, S. Chen and A.K. Jain, "Adaptive Flow Orientation Based Feature Extraction in Fingerprint Images", Pattern Recognition, Vol. 28, pp. 1657-1672, November 1995.

[5] Alessandro Farina, Zsolt M.Kovacs-Vajna, Alberto leone, Fingerprint minutiae extraction from skeletonized binary images, Pattern Recognition, Vol.32, No.4, pp877-889, 1999.

[6] Lee, C.J., and Wang, S.D.: Fingerprint feature extration using Gabor filters, Electron. Lett., 1999, 35, (4), pp.288-290.

[7] M. Tico, P.Kuosmanen and J.Saarinen. Wavelet domain features for fingerprint recognition, Electroni. Lett., 2001, 37, (1), pp.21-22.

[8] L. Hong, Y. Wan and A.K. Jain, "Fingerprint Image Enhancement: Algorithms and Performance Evaluation", IEEE Transactions on PAMI, Vol. 20, No. 8, pp.777-789, August 1998.

[9] Image Systems Engineering Program, Stanford University. Student project By Thomas Yeo, Wee Peng Tay, Ying Yu Tai.

[10] L.C. Jain, U.Halici, I. Hayashi, S.B. Lee and S.Tsutsui. Intelligent biometric techniques in fingerprint and face recognition. 1999, the CRC Press.

[11] M. J. Donahue and S. I. Rokhlin, "On the Use of Level Curves in Image Analysis," Image Understanding, VOL. 57, pp 652 - 655, 1992.

[12] Jin Fei Lim; Chin, R.K.Y., "Enhancing Fingerprint Recognition Using Minutiae-Based and Image-Based Matching Techniques," Artificial Intelligence, Modelling and Simulation (AIMS), 2013 1st International Conference on, vol., no., pp.261, 266, 3-5 Dec. 2013.

[13] Sudheesh, K.V.; Patil, C.M., "An approach of cryptographic for estimating the impact of fingerprint for biometric," Pattern Recognition, Informatics and Medical Engineering (PRIME), 2012 International Conference on , vol., no., pp.167,171, 21-23 March 2012.

[14] Patil, A.R.; Zaveri, M.A., "A Novel Approach for Fingerprint Matching Using Minutiae," Mathematical/Analytical Modelling and Computer Simulation (AMS), 2010 Fourth Asia International Conference on, vol., no., pp.317,322, 26-28 May 2010.

[15] Ito, K.; Morita, A.; Aoki, T.; Higuchi, T.; Nakajima, H.; Kobayashi, K., "A fingerprint recognition algorithm using phase-based image matching for low-quality fingerprints," Image Processing, 2005. ICIP 2005. IEEE International Conference on , vol.2, no., pp.II,33-6, 11-14 Sept. 2005.

[16] Malathi, S.; Meena, C., "An efficient method for partial fingerprint recognition based on local binary pattern," Communication Control and Computing Technologies (ICCCCT), 2010 IEEE International Conference on , vol., no., pp.569,572, 7-9 Oct. 2010.

[17] Liming Zhang; Yilong Yin, "Fingerprint Matching Based on Ternary Vector," Pattern Recognition, 2009. CCPR 2009. Chinese Conference on, vol., no., pp.1,5, 4-6 Nov. 2009.

[18] Wen Wen; Zhi Qi; Zhi Li; Junhao Zhang; Yu Gong; Peng Cao, "A Robust and Efficient Minutia-Based Fingerprint Matching Algorithm," Pattern Recognition (ACPR), 2013 2nd IAPR Asian Conference on , vol., no., pp.201,205, 5-8 Nov. 2013.

[19] Zin Mar Win; Sein, M.M., "Fingerprint recognition system for low quality images," SICE Annual Conference (SICE), 2011 Proceedings of, vol., no., pp.1133, 1137, 13-18 Sept. 2011.

[20] Kaur, R.; Sandhu, P.S.; Kamra, A., "A novel method for fingerprint feature extraction," Networking and Information Technology (ICNIT), 2010 International Conference on , vol., no., pp.1,5, 11-12 June 2010.

Mr. MAYANK TRIPATHY is studying M.TECH (IV SEM) in information Security in Disha Institute of Management and Technology, Raipur Chhattisgarh India. He has completed B.E. (Information technology) in session 2011-12 from K.I.T. Collage Raigarh (C.G.) university of Chhattisgarh swami Vivekananda technical university Bhilai (Chhattisgarh). His research interests are in Information and Network Security, genetic algorithm, Cryptography, artificial intelligence, Digital Signal Processing and Image Processing.

Mr. Deepak Shrivastava received his M. Tech. in Information Security, Branch of Computer Science and Engineering degree from Disha Institute of Management and Technology, Raipur, Chhattisgarh, India, affiliated to Chhattisgarh Swami Vivekananda Technical University, Bhilai, Chhattisgarh, India in 2014 and Master in Computer Applications (MCA) degree from Indira Gandhi National Open University in 2008. He is Assistant Professor, Department of Computer Science and Engineering in Disha Institute of Management and Technology, Raipur, Chhattisgarh, India. His research interests are in Information and Network Security, Cloud Computing, Cryptography, Artificial Intelligence, Digital Signal Processing and Image Processing.