ISSN No. 0976-5697

Volume 5, No. 8, Nov-Dec 2014

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

Performance Improvement for Fingerprint Recognition System using Shape and Orientation Descriptors

A. E. Amin
Department of Computer Science
Mansoura University
Mansoura - Egypt

A. F. Elgamal
Department of Computer Science
Mansoura University
Mansoura - Egypt

Abstract: One of the oldest methods to human identification is fingerprints, there are many and varied systems are used to recognition and verification. The fingerprint recognition systems are similar in all implementation steps which represented by pre and post processing of fingerprint images and extract features. In the preprocessing stage, the image has been enhanced by using Histogram equalization and Fourier transform. The gray fingerprint image is converted to 0-value for ridge and 1-value for bifurcation that is called image binarization. The binarized fingerprint image is used to separate the region of actual fingerprint from the background to produce Region of Interest (ROI). Minutiae extraction stage is consist of two steps namely Fingerprint Ridge Thinning and Minutia Marking. The post-processing stage eliminates spurious feature points based on the structural and spatial relationships of the minutiae, and then validates the minutiae points. Speed and high accuracy in the performance are the distinguishing parameters of fingerprint recognition systems. So, the continuing developments of fingerprints matching algorithms have the greatest impact on the development of biometric identification systems. In spite of the different techniques in the matching methods but were similar in the two basic steps are the alignment andscoring. In this paper, fingerprint matching algorithm based on shape and orientation descriptors is used. The proposed algorithm is designed to address the spurious and missing minutiae and the inability to detect minutiae to guide pre-alignment.

Keywords: Fingerprint identification, Minutiae extraction, Fingerprint image processing, Scoring matching.

1. INTRODUCTION

Automatic authentication of a person based on his physiological or behavioral characteristics is called biometric recognition. Examples of traits used in biometric recognition are: face, fingerprint, iris, signature, and voice characteristics [1]. A fingerprint is the feature pattern of one finger which is unique, so it has being used for identification and investigation. It is the marks made by ridges and furrows in the finger. Fingerprint features are generally categorized into three levels as shown in Figure 1:

1. Refer to ridge orientation field and its derived features, i.e., singular points and pattern type.
2. Refer to ridge skeleton and its derived features, i.e., ridge bifurcations and endings.
3. Include ridge contours, position, and shape of sweat pores and incipient ridges.

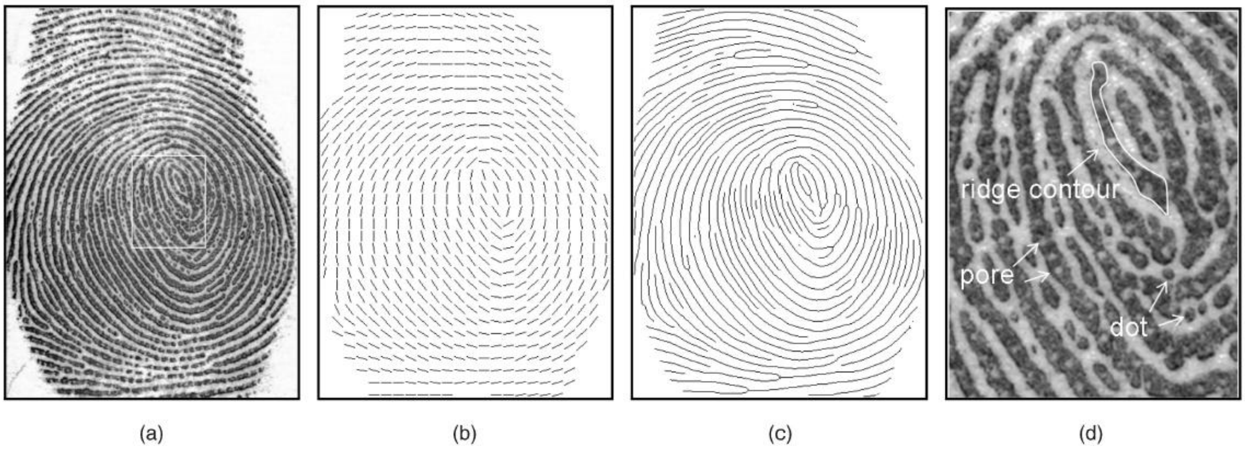


Fig. 1:Features at three levels. (a) Grayscale image (b) Level 1 feature (orientation field), (c) Level 2 feature (ridge skeleton), and (d) Level 3 features (ridge contour, pore, and dot) [**2**].

Fingerprint recognition systems were mostly used in forensic sciences. Now the current popularity of these systems is civilian applications such as the control of logical access to software, and voters during elections [3], where a person needs to be verified or identified with high confidence. Fingerprint Verification System determines the correspondence of an input fingerprint (query image) with a template fingerprint stored in data base [4]. A key component in fingerprint recognition systems is the fingerprint matching algorithm [3]. Most fingerprint matching systems are based on four types of fingerprint representation schemes: gray scale image, phase image, skeleton image, and minutiae as shown in Figure 2.

.

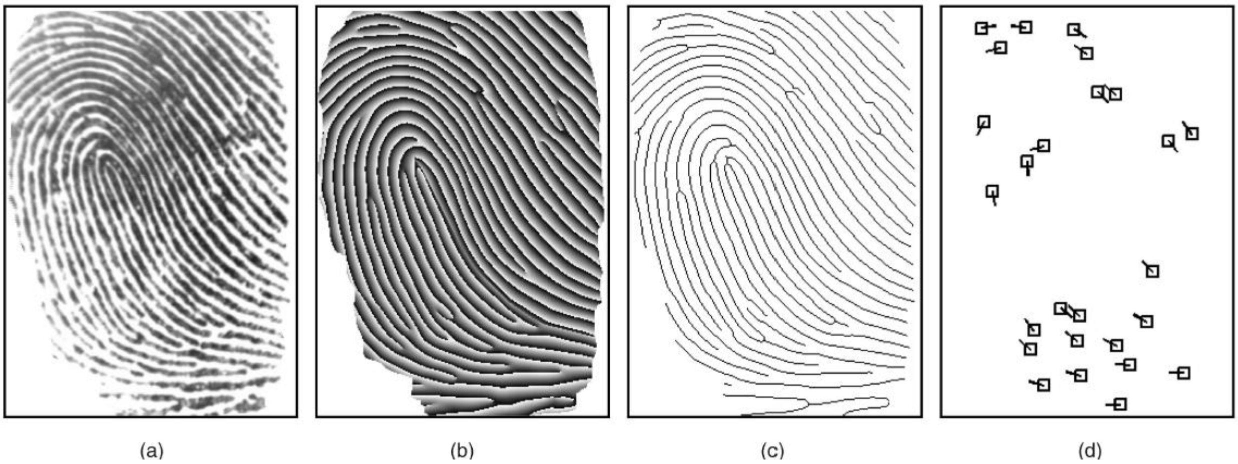


Fig. 2: Fingerprint representation schemes. (a) Grayscale image, (b) phase image, (c) skeleton image, and (d) minutiae.[2]

There are many fingerprint matching algorithms which can be classified into three categories: correlation-based matching [5, 6], minutia-based matching [7] and non-minutiae -based feature matching [8, 9].Correlation-based matching (CM) involves superimposing 2 fingerprint images together and calculating pixel-wise correlation for different displacement and rotations. Correlation-based methods are characterized by abilities to save most information in fingerprints although it is fragile to nonlinear distortion. Minutia-based matching uses extracted minutiae from both fingerprints in order to help perform alignment and retrieve minutiae pairings between both fingerprint minutiae sets. Minutiae-based matching can be viewed as a point-pattern matching problem with theoretical roots in pattern recognition and computer vision. Non-minutiae feature based matching use non-minutiae features, such as ridge shape, orientation and frequency images in order to perform alignment and matching.

The methods which based on minutiae are the most common method due to the close similarity with the forensic methods in fingerprints comparison that are considered acceptable as identification guide legal in many countries [10]. This methodis extremely uniquespatial distributionof eachfingerprintfrom the other, which makesit ideal features forfingerprint matching. In addition,the sets of minutiaepoints which extracted by this method, are characterized bydifferentlevels ofrepresentationinthe fingerprintimage, such as ridge orientation and skin pores.

Shape matching algorithms are similar to minutiae-based fingerprint matching algorithms, because they usually combine the use of descriptors with dynamic programming, greedy, simulated annealing, and energy minimization based algorithms, in order to register shapes and calculate the measure of similarity.

The required characteristics in fingerprint matching algorithms, which using shape matching methods, not only stability to rotation and scale but also don’t influence by amounts of distortion and outlier point samples.

The main purpose of this research is to produce a better performance and accuracy for the fingerprint recognition system. The paper is organized as follows: section 2 describes the proposed method; section 3 introduces fingerprint matching; section 4 illustrates implementation and experiments; finally, section 5 concludes the research.

2. THE PROPOSED METHOD

There are many proposed techniques and algorithms in feature extraction of human fingerprints for specific or different applications. The proposed method is consists of two main parts namely fingerprint minutiae extractor and minutiae matcher.

The acquisition of fingerprint images was performed by two main methods namely off-line sensing and live scan. The off-line scan based on ink-technique where the subject's finger was spread with black ink and pressed against a paper card. The card was scanned by using a common paper scanner to producing the final digital image. Whereas, the live scan based on directly sensing where digital images is acquired by the finger surface with an electronic fingerprint scanner.

The fingerprint image which produced by both method are not of optimum quality results ink distortion or scanner noise. So, both distortion and noise must be removing and enhanced fingerprint image quality by using preprocessing and post-processing stages.

A. Fingerprint Image Preprocessing

Aim of fingerprint image preprocessing is enhancement of fingerprint image by using Histogram Equalization and Fourier Transform. Then, adaptive threshold method is used to binarized the fingerprint image. To implement that the fingerprint image must be segmented by an approach called block direction estimation, and Region of Interest is extracted by Morphological operations.

I) Fingerprint Image Enhancement:

Fingerprint Image enhancement is to make the image clearer byincreasing the contrast between ridges and valleys and for connecting the false broken points of ridges for easy further operations.

- *Histogram Equalization:*

Histogram equalization is to expand the pixel value distribution of an image so as to increase the perceptional information [11]. Figure 3 is showing, the original histogram (b) for fingerprint image_1 (a), the histogram equalization (d) occupies all the range from 0 to 255 and visualization effect is enhanced in fingerprint image_2 (c).

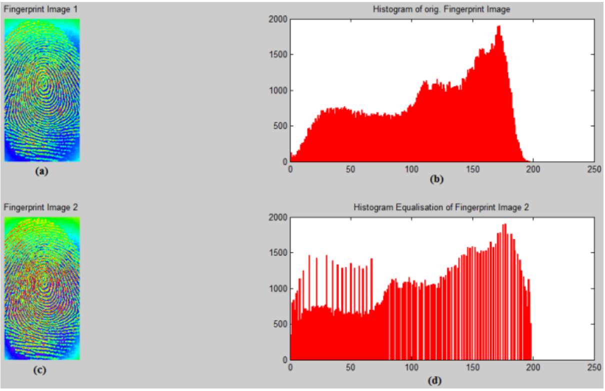


Fig 3: Effect of histogram equalization.

- *Fingerprint Enhancement by Fourier Transform:*

Fourier transform is a technique used to transform the input image from spatial domain to frequency domain. The following steps are involved in this technique: Normalization, Segmentation, Orientation image estimation, 2D Fourier transformation, Butterworth filter, Inverse 2D Fourier transform and reconstruction.

Fourier transform (FT) is used to enhancement fingerprint image by divide the image into small processing blocks (32×32) and perform FT according to the following equations [12]:

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

Where, $u \; and \; v = 0,1, \ldots ,31$.

In order to enhance a specific block by its dominant frequencies, we multiply the FT of the block by its magnitude a set of times. Where the magnitude of the original:

$$FFT = abs\big(F(u,v)\big) = |F(u,v)|$$

Get the enhanced block according to:

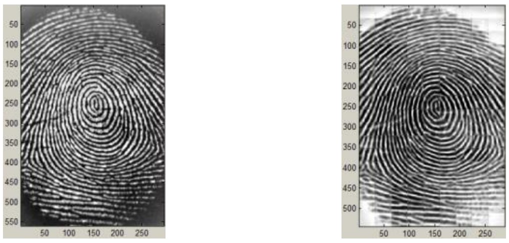
$$g(x,y) = F^{-1}\{F(u,v) \times |F(u,v)|^{K}\}$$

The k 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. Whereas, $\mathrm{F}^{-1}\big(\mathrm{F}(\mathrm{u},\mathrm{v})\big)$is done by:

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

Where x and y $= 0,1, \ldots ,31$

Figure 4 presents original fingerprint image (a) and the image after Fingerprint FT (FFT) enhancement (b).



(a) Original image **(b) FFT Enhanced image**

Fig 4: Effect of Fourier Transform.

After the enhancement of the image through FFT it is quite easy to connect the falsely broken points on ridges and it becomes simpler to remove some unwanted cross connections between ridges.

II)Fingerprint image binariztion:

The aim of binerization processing is transform the 8-bit Gray fingerprint image to a 1-bit image with black color for ridge (0-value) and white color for bifurcation (1-value). A locally adaptive binarization method is used to implement the fingerprint image binarized, that extracted the brightness and density of the (16×16) block from image (as a pixel), then set range of threshold value are detected by subtracted the selected pixels after added the sensitivity value from the Y value of selected pixels. By shifting the block and repeating, a new set of threshold value range are determined with a new pixel [13]. Each pixel is transformed to 1 if the value is larger than the mean intensity value of the current block to which the pixel belongs as shown in the figure 5.

ISSN No. 0976-5697

Volume 5, No. 8, Nov-Dec 2014

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

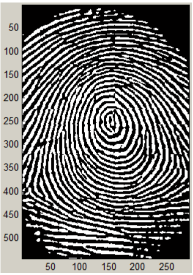


Fig 5: Fingerprint image binarization.

III) Fingerprint Image Segmentation

Fingerprint image segmentation means separate the region of actual fingerprint from the background which one causes distortion in statistical data in further processing. Region of Interest (ROI) is useful to be recognized for each fingerprint image. There are two- step method used to extract the ROI namely blocks direction estimation and direction variety check, while the second is intrigued from some Morphological methods [14].

- *Block direction estimation*

Block direction estimation depends on the orientations at pixel scale, the orientation of ridge is orthogonal to average phase angle of changes pixels value indicated by gradients. The orientation of an image block is estimated by averaging the squared gradients to avoid the orientation ambiguity. Estimate the block direction for each block of the fingerprint image with (16×16) pixel, by the algorithm:

Block Direction Estimation Algorithm

I. Calculating G_x and $G_y$gradients at each pixel of fingerprint image, where gradient operator is estimated as Sobel operator:

$$S_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}_{G_x} \quad , \quad S_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}_{G_y}$$

II. Estimate the local orientation in ω×ω blocks, centered at pixel (x ,y) using the following equations:

$$\overline{G_{Sy}}(x,y) = \sum_{i=-w/2}^{w/2} \sum_{j=-w/2}^{w/2} 2G_x(x+i,y+j)G_y(x+i,y+j)$$

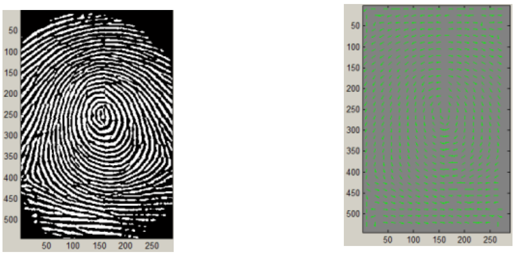
$$\overline{G_{Sx}}(x,y) = \sum_{i=-w/2}^{w/2} \sum_{j=-w/2}^{w/2} \left(G_x(x+i,y+j)^2 - G_y(x+i,y+j)^2\right)$$

$$\bar{\varphi}(x,y) = \frac{1}{2}\tan^{-1}\frac{\overline{G_{Sy}}(x,y)}{\overline{G_{Sx}}(x,y)}$$

$$\theta_{gb}(x,y) = \bar{\varphi}(x,y) + k\pi$$

$$k = \begin{cases} \frac{1}{2}\ when\ \left(\left(\bar{\varphi}(x,y) < 0 \cap \overline{G_{Sy}}(x,y) < 0\right) \cup \left(\bar{\varphi}(x,y) \geq 0 \cap \overline{G_{Sy}}(x,y) > 0\right)\right) \\ 1\ when\ \bar{\varphi}(x,y) < 0 \cap \overline{G_{Sy}}(x,y) \geq 0 \\ 0\ when\ \bar{\varphi}(x,y) \geq 0 \cap\ \overline{G_{Sy}}(x,y) \leq 0 \end{cases}$$

θ_{gb}Conducted experiments show that the size of ω×ω blocks should be at least twice the average distance between the fingerprint ridges [15]. Computed orientation field can be presented as color hue values as shown in figure 6.



(a) Binarized Image **(b)** θ_{gb}Oriented field

Fig 6: Binarized and Oriented field θ_{gb} Fingerprint image

- *ROI extraction by Morphological operations:*

Morphological operators [16, 17] are the mathematical operators work in two stages, first stage, it is used to perform the edge detection and secondly they are used as the post-processing stage to perform whole filling. Morphological operators are able to analyze the geometrical structure and take the adaptive decision regarding the size, shape, convexity of the image. There are two morphological operations “OPEN” and “CLOSED” are used to extract ROI [18]. The OPEN operation expands the image and removes the picks introduced by background noise and CLOSE operation shrinks the image and eliminates small cavities. Figure 7 shows images obtained after morphological operations. The bound is obtained by subtracting closed area from the open area. The blocks out of the bound are described to give tightly bounded fingerprint ROI.

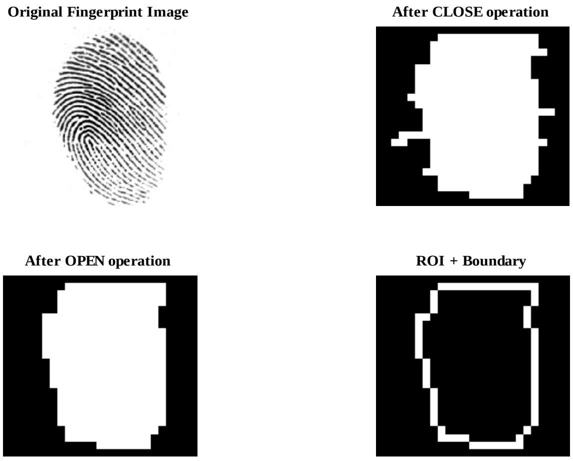


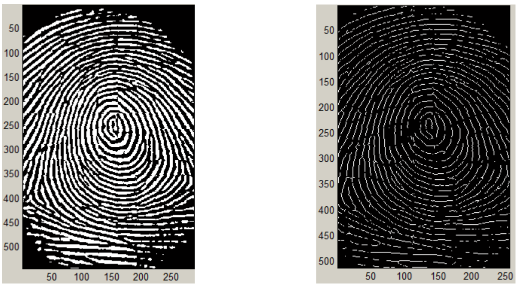
Fig 7: morphological operations Fingerprint image.

B. Minutiae Extraction

Many methods have been proposed for the minutia extraction, the traditional method consist of the following steps:

I) Fingerprint Ridge Thinning

The objective of thinning is transforming a digital binary pattern to a connected skeleton of unit width as shown in figure 8.



(a) Binarized Image **(b)** Thinned Image

Fig 8: Thinned Image

Two basic implementations available for this approach are sequential and parallel methods [19]. Thinning process characterized by does not change the location and orientation of minutiae points compared to original fingerprint which ensures accurate estimation of minutiae points. But the most important disadvantages the distortions in skeleton of thinned image (spikes) which lead to spurious ridge bifurcation and endings. Therefore some procedure aiming the elimination of these artifacts must be performed after the thinning.

II)Enhanced Thinning

Therefore, before minutiae extraction, there is a need to a validation algorithm to eliminate the erroneous pixels while preserving the skeleton connectivity at the fork regions. For this purpose an enhanced thinning flowchart is shown in figure 9.

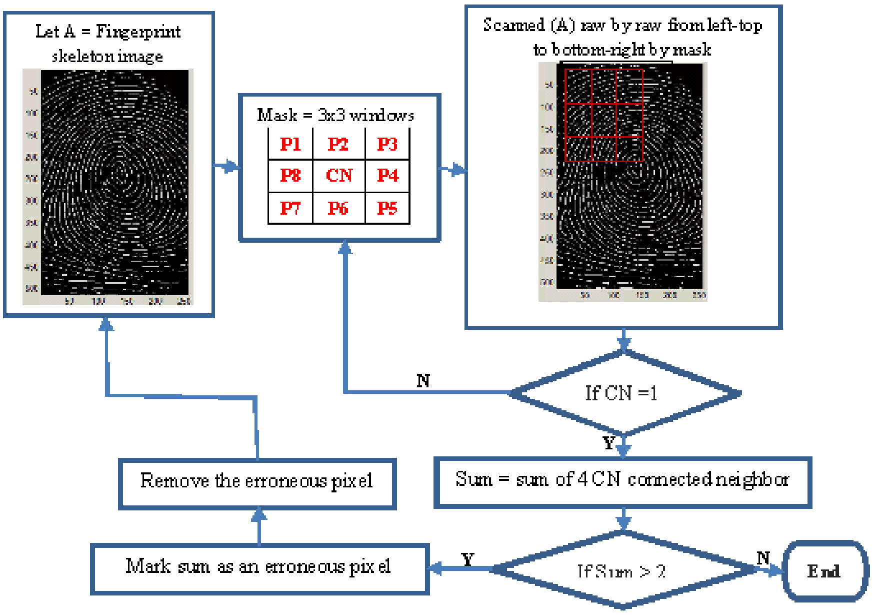

Let A = Fingerprint skeleton image
Mask = 3x3 windows
P1
P2
P3
P8
CN
P4
P7
P6
P5
Scanned (A) raw by raw from left-top to bottom-right by mask
N
If CN =1
Y
Remove the erroneous pixel
Sum = sum of 4 CN connected neighbor
Mark sum as an erroneous pixel
Y
If Sum > 2
N
End

Fig 9: Thinning process flowchart.

III) Minutia Marking

Locating minutia points in the thinned image is implemented by using a Crossing Number (CN) method [20]. In the CN method, the minutia extraction is performed through the scanning of the 3 x 3 neighborhood of each ridge pixel in the thinned image. The CN value is then calculated from half the sum of the differences between pairs of adjacent pixels in the eight-neighborhood as shown in the following equation.

$$CN = 0.5\sum_{i=1}^{8} P_i - P_{i+1} \quad , \qquad P_i = P_9$$

Where, p_i is the pixel value in the neighborhood of p . For a pixel p , its eight neighboring pixels are scanned in an anti-clockwise direction as follows:

p_4	p_3	p_2
p_5	p	p_1
p_6	p_7	p_8

The minutiae points are determined according to the value of the CN. Using the properties of the CN as shown in Table 1, the ridge pixel can then be classified as a ridge ending, bifurcation or non-minutiae point.

Table 1: Properties of the Crossing Number.

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

A CN of one corresponds to a ridge ending point and a CN of three corresponds to a bifurcation point as shown in figure 11.

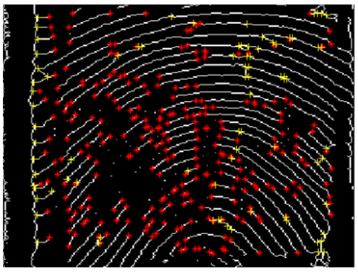


Fig 11: Located Minutiae

All the ridge end points and ridge bifurcation points detected through this method are not always true features, but the method does seem to identify most of the true feature point.

A post-processing stage filters out the undesired feature points based on their structural characteristics.

IV) Post-Processing

The post-processing phase eliminates spurious feature points based on the structural and spatial relationships of the minutiae, and then validates the minutiae points. Minutiae Post-processing steps are:

- *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 [21]. Actually all the earlier stages themselves occasionally introduce some artifacts which later lead to spurious minutia. Figure 12 shows the types of false minutia (m_i).

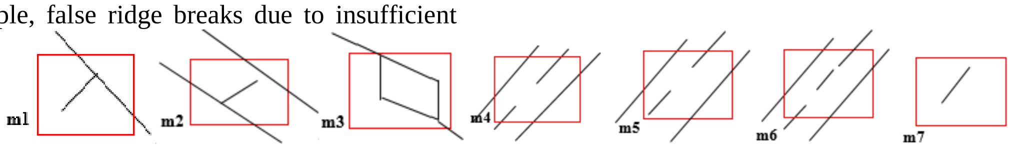
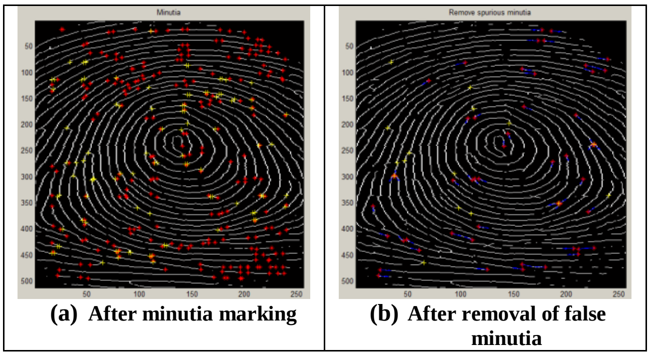


Fig. 12: False minutia structures.



(a) After minutia marking

(b) After removal of false minutia

Fig. 13: fingerprint minutia marking and removal false minutia

Figure 13 shows the post processing algorithm filters out the undesired minutiae points.

- *Unify termination and bifurcation*

Types of minutia are similar with each other to a fingerprint which happens as a result of contact with the different types of data acquisition conditions. So the minutiae must be saved in some form which distinction between them.

So each minutia is completely characterized by x-coordinate, y- coordinate, orientation and associated ridge. Each bifurcation can be divided into three terminations, each termination having a characterization of minutia [22]. The orientation for each termination$(\boldsymbol{t_x}, \boldsymbol{t_y})$is estimated by using the following algorithm:

Termination Orientation Estimation Algorithm

I. Determine the starting point for termination by tracking a ridge segment, and calculate its length (D).

II. In the particular ridge segment, all the points of X-coordinates are summed up.

III. S_x, S_y are obtained by divided the above summation with D, sequentially.

IV. The direction is obtained from:

$$\tan^{-1}\left(\frac{S_y - t_y}{S_x - t_x}\right)$$

Minutia triplets (MT) is minutiae structures which commonly represents the minutiae in minutiae-based matching. MT is described as:

$$MT_j = (x, y, \theta)$$

Where, x, y are representing x-y coordinate of minutia and θ the angular direction of the main ridge.

To a template image (T) minutiae set:

$$\mathrm{T} = \{\mathrm{MT_{T_1}}, \mathrm{MT_{T_2}}, \dots, \mathrm{MT_{T_r}}\}$$

Where, $\mathrm{MT}_{\mathrm{T}_j} = \left\{\mathrm{x_{T_j}}, \mathrm{y_{T_j}}, \theta_{\mathrm{T_j}}\right\}$ and $1 \le \mathrm{j} \le \mathrm{r}$

3. FINGERPRINT MATCHING TECHNIQUES:

The proposed fingerprint matching technique is consisting of three main stages, beginning by hybrid shape and orientation descriptor and ends up similarity score passing through registration stage.

A. Shape and orientation descriptor stage:

For each minutia,$\mathrm{MT}_{\mathrm{T}_\mathrm{j}} \in \mathrm{T}$and $\mathrm{MT}_{\mathrm{Q}_\mathrm{i}} \in \mathrm{Q}$, the best matching minutia is wanted to be found. The spatial geometric regions which represents by Cartesian plan (x, y) are transformed to log-polar regions (ξ, η) by:

$$\begin{bmatrix} \xi \\ \eta \end{bmatrix} = \begin{bmatrix} \log \mathrm{r} \\ \theta \end{bmatrix} = \begin{bmatrix} \log \sqrt[2]{\mathrm{x}^2 + \mathrm{y}^2} \\ \tan^{-1} \dfrac{\mathrm{y}}{\mathrm{x}} \end{bmatrix}$$

The shape context descriptor [23] is constructed for each minutia in template and query fingerprint by providing a localized spatial survey of the minutiae distributions for each fingerprint.

The minutia matching cost $(\mathrm{C}_{\mathrm{ij}})$ between$\mathrm{MT}_{\mathrm{T}_\mathrm{j}} \in \mathrm{T}$ and $\mathrm{MT}_{\mathrm{Q}_\mathrm{i}} \in \mathrm{Q}$is calculated as:

$$\mathrm{C}_{\mathrm{ij}} = \mathrm{C}\left(\mathrm{MT}_{\mathrm{T}_\mathrm{i}}, \mathrm{MT}_{\mathrm{Q}_\mathrm{j}}\right)$$

to find the optimal mapping of minutiae. The distributed histogram for shape context is modified by χ^2 statistic:

$$\mathrm{C}_{\mathrm{ij}} = \mathrm{C}\left(\mathrm{MT}_{\mathrm{T}_\mathrm{i}}, \mathrm{MT}_{\mathrm{Q}_\mathrm{j}}\right) = \frac{1}{2}\sum_{\mathrm{k}=1}^{\mathrm{K}} \frac{\left(\mathrm{h}_{\mathrm{MT}_{\mathrm{T}_\mathrm{i}}}(\mathrm{K}) - \mathrm{h}_{\mathrm{MT}_{\mathrm{Q}_\mathrm{j}}}(\mathrm{K})\right)^2}{\mathrm{h}_{\mathrm{MT}_{\mathrm{T}_\mathrm{i}}}(\mathrm{K}) + \mathrm{h}_{\mathrm{MT}_{\mathrm{Q}_\mathrm{j}}}(\mathrm{K})}$$

Where,$h_{MT_{T_i}}$and$\mathrm{h}_{\mathrm{MT}_{\mathrm{Q}_\mathrm{j}}}$ denote the K-bin histograms of pointsMT_{T_i}andMT_{Q_j},

respectively which calculated by:

$$\mathrm{h}_{\mathrm{MT}_{\mathrm{T}_\mathrm{i}}}(\mathrm{K}) = \mathrm{No.}\left\{\mathrm{MT}_{\mathrm{T}_\mathrm{J}} \neq \mathrm{MT}_{\mathrm{T}_\mathrm{i}} : \left(\mathrm{MT}_{\mathrm{T}_\mathrm{J}} - \mathrm{MT}_{\mathrm{T}_\mathrm{i}}\right)\right\} \in \mathrm{bin}(\mathrm{K})$$

To improve overall accuracy of the minutia matching cost, there is extra relevant information added like minutia type and minutia angle. So, the log-polar histogram cost is modified as:

$$\mathrm{C}^*_{\mathrm{ij}} \equiv \mathrm{C}^*\left(\mathrm{MT}_{\mathrm{T}_\mathrm{i}}, \mathrm{MT}_{\mathrm{Q}_\mathrm{j}}\right)$$

$$= \left(1 - \gamma \mathrm{C}^{\mathrm{type}}_{\mathrm{ij}} \mathrm{C}^{\mathrm{angle}}_{\mathrm{ij}}\right)\left(\frac{1}{2}\sum_{\mathrm{k}=1}^{\mathrm{K}} \frac{\left(\mathrm{h}_{\mathrm{MT}_{\mathrm{T}_\mathrm{i}}}(\mathrm{K}) - \mathrm{h}_{\mathrm{MT}_{\mathrm{Q}_\mathrm{j}}}(\mathrm{K})\right)^2}{\mathrm{h}_{\mathrm{MT}_{\mathrm{T}_\mathrm{i}}}(\mathrm{K}) + \mathrm{h}_{\mathrm{MT}_{\mathrm{Q}_\mathrm{j}}}(\mathrm{K})}\right)$$

With constrains, $0 \le \gamma \le 1$

$$\mathrm{C}^{\mathrm{type}}_{\mathrm{ij}} = \begin{cases} -1, & \text{if type}\left(\mathrm{MT}_{\mathrm{T}_\mathrm{i}}\right) = \text{type}\left(\mathrm{MT}_{\mathrm{Q}_\mathrm{j}}\right) \\ 0, & \text{if type}\left(\mathrm{MT}_{\mathrm{T}_\mathrm{i}}\right) \neq \text{type}\left(\mathrm{MT}_{\mathrm{Q}_\mathrm{j}}\right) \end{cases}$$

$$\mathrm{C}^{\mathrm{angle}}_{\mathrm{ij}} = -\frac{1}{2}\left(1 + \cos\left(\left|\theta_{\mathrm{initial}} - \theta_{\mathrm{warped}}\right|\right)\right)$$

Where;$\left(\left|\theta_{intial} - \theta_{warped}\right|\right)$is absolute angle difference between the beginning and after warping iteration for orientation tangent for template and query minutia. Hungarian algorithm [24] is used to compute the total matching cost $H(\pi)$ by:

$$\mathrm{H}(\pi) = \sum_{\mathrm{i}} \mathrm{C}\left(\mathrm{MT}_{\mathrm{T}_\mathrm{i}}, \mathrm{MT}_{\mathrm{Q}_{\pi(\mathrm{i})}}\right)$$

Outlier dummy points can be added to the fingerprint minutiae set to conforming one to one mapping. For more robust handling, dummy point mappings can be extended to minutiae that have a minimum cost greater than a desired threshold δ_d.

B. Registration stage:

When the fingerprint minutiae extracted, the Thin Plate Spline (TPS) [25] is used to register the corresponding points together. TPS is simple and sufficient mathematical model based on algebraically expressing the physical bending energy of a thin metal plate on point constraints. TPS is used affine transform [26] to decompose the fingerprint minutiae into linear and non-linear components. Linear component can be considered as the transformation that expresses the global geometric dependence of the point sets, whereas non-affine transform component is known as individual transform components in order to fine tune the interpolation of the point sets. Affine transform is characterized by its component allows TPS to be invariant under both rotation and scale.

The TPS model describes the transformed coordinates (x_{new}, y_{new})both independently as a function of the original coordinates (x, y):

$$\mathrm{x}_{\mathrm{new}} = \mathrm{f}_{\mathrm{x}(\mathrm{x},\mathrm{y})}$$
$$\mathrm{y}_{\mathrm{new}} = \mathrm{f}_{\mathrm{y}}(\mathrm{x}, \mathrm{y})$$

Given the displacements of a number of minutiae points, the TPS model interpolates those points, while maintaining maximal smoothness. The smoothness is represented by the bending energy of a thin metal plate. At each landmark point(x, y), the displacement is represented by an additional z-coordinate, and, for each point, the thin metal plate is fixed at position (x, y, z). The bending energy is given by the integral of the second-order partial derivatives over the entire surface and can be minimized by solving a set of linear equations. Therefore, the algebraic crux of TPS is presented in terse overview. The TPS model for one of the transformed coordinates is given by parameter vectors$(a)$and(w):

$$f(x, y) = a_1 + a_2 x + a_3 y + \sum_{i=1}^{n} w_i U(|P_i - (x, y)|)$$

Where, $\mathrm{U}(\mathrm{r}) = \mathrm{r}^2 \log \mathrm{r}^2$is the basis function, $(a) \xrightarrow{defines}$ the affine part of the transformation, $(w) \xrightarrow{gives}$ an additional non-linear deformation, P_i are the minutiae that the TPS interpolates, and n is the number of minutiae. The TPS parameters that minimize the bending energy can be found by solving a set of linear equations:

$$\mathrm{Kw} + \mathrm{Pa} = \mathrm{v}$$

$$\mathrm{P}^{\mathrm{T}}\mathrm{w} = 0$$

Where, $\mathrm{w} = [\mathrm{w}_\mathrm{x}(1)\mathrm{w}_\mathrm{x}(2) \;\; \ldots \;\; \mathrm{w}_\mathrm{x}(\mathrm{n})]^{\mathrm{T}}$,

$$\mathrm{v} = [\mathrm{q}_\mathrm{x}(1)\mathrm{q}_\mathrm{x}(2) \;\; \ldots \;\; \mathrm{q}_\mathrm{x}(\mathrm{n})]^{\mathrm{T}} \,,$$

$$\mathrm{a} = [\mathrm{a}_\mathrm{x}(1)\mathrm{a}_\mathrm{x}(2)\mathrm{a}_\mathrm{x}(3)]^{\mathrm{T}}$$

$$P = \begin{bmatrix} 1 & p_x(1) & p_y(1) \\ 1 & p_x(2) & p_y(2) \\ \vdots & \vdots & \vdots \\ 1 & p_x(n) & p_y(n) \end{bmatrix}$$

$$K = \begin{bmatrix} U(|P_1 - P_1|) & U(|P_1 - P_2|) & \dots & U(|P_1 - P_n|) \\ U(|P_2 - P_1|) & U(|P_2 - P_2|) & \dots & U(|P_2 - P_n|) \\ \vdots & \vdots & \ddots & \vdots \\ U(|P_n - P_1|) & U(|P_n - P_2|) & \dots & U(|P_n - P_n|) \end{bmatrix}$$

$P_i = \left(P_x(i), P_y(i)\right)$is the set of minutiae points in the first image, $P_x(i)$is the x-coordinate of point i in set P_i, $Q_i = \left(q_x(i), q_y(i)\right)$is the set of corresponding points in the second image, and n is the number of minutiae points.

A method is presented to estimate approximating thin-plate splines. These splines do not exactly interpolate all given points, but are allowed to approximate them in favor of a smoother transformation. The smoothness is controlled by a parameter λ, which weights the optimization of minutiae distance and smoothness. For λ=0, there is full interpolation, while for very large λ, there is only an affine transformation left. For equal isotropic errors at all minutiae, the optimal TPS parameters can be found by solving the following system of equations:

$$(K + \lambda I) + Pa = v$$

Where, I: the $(n \times n)$ identity matrix.

For instance, the first plot in Figure (14-a) shows two configurations of points, both approximately in the shape of five-pointed stars but at different scale, rotation and location. The shape vector to be aligned is referred to as the input, whereas the shape vector to which the inputs are aligned is called the base. The second plot (14-b) shows them after one has been aligned to the other by using registration stage.

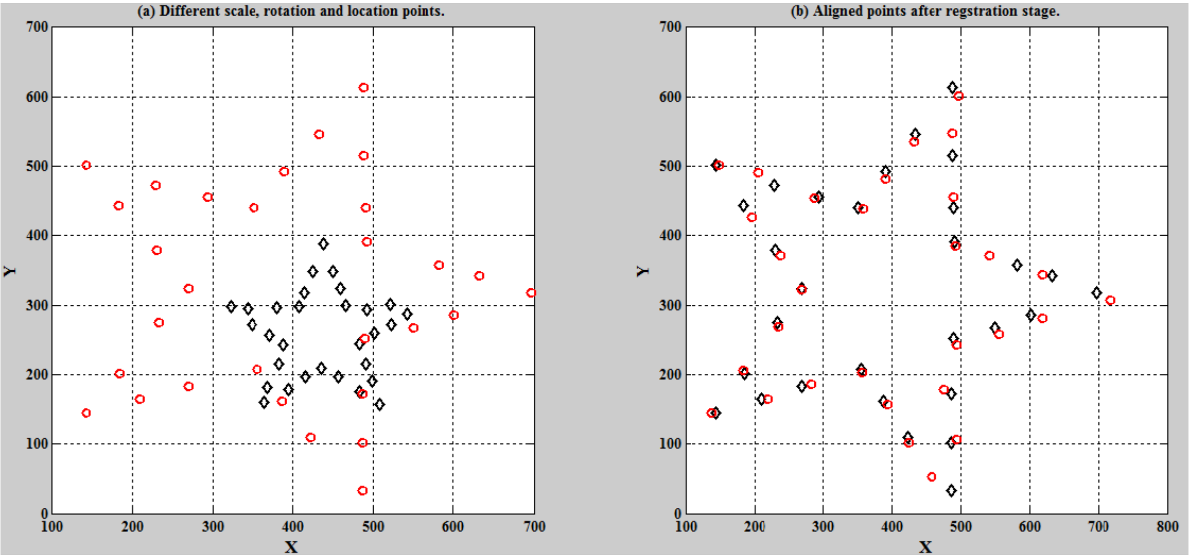


Fig. 14: Alignment of two shapes by using registration stage.

C. Similarity score:

Once the *n* iterations are performed, the final pairs have now been established. From this, the shape similarity distance measure can be calculated as:

$$D_{sc}(P,Q) = \frac{1}{k}\sum_{p\in P} arg \min_{q\in Q} C\left(p, T_{(q)}\right) + \frac{1}{r}\sum_{q\in Q} arg \min_{p\in P} C\left(p, T_{(q)}\right)$$

Where, $T(.)$ denotes the TPS transformed representative of the contour point q. In addition, an appearance term,$D_{sc}(P,Q)$, measuring pixel intensity similarity and a bending energy term, $D_{be}(P,Q) = I_f$, can be added to the similarity score.

$$D_{sc}^{*} = D_{sc} + \beta D_{be}$$

4. Implementation and Experiments:

a) Databases of fingerprints:

Fingerprint Verification Competition (FVC) was a worldwide competition for fingerprint based biometric authentication algorithms. The FVC2002 database Db1 set *A* [27] which contains 800 fingerprint images with 100 fingers having 8 impressions for each. The FVC2002 protocol was used in experimentation, comprising of$\frac{(100\times8\times7)}{2} = 2800$ genuine and$100 \times \frac{100-1}{2} = 4950$ imposter attempts for the FVC2002 database. And FVC2004 are consists of four databases which collected from three different scanners and the synthetic generator SFinGe [28, 29]. each database have a total of 120 fingers and 12 impressions per finger (1440 impressions) using 30 volunteers. The size of each database to be used in the test was established as 110 fingers, 8 impressions per finger (880 impressions). The experiments are done on windows 8, processor core i5 and RAM 6 GB to obtain high accuracy and performance.

b) Fingerprint feature extraction processing:

Fingerprint verificationsystem is implemented to extract background information without noising and that requires better image quality in order to produce match scores with higher accuracy. Figures 15-20 show fingerprint recognition system processing begin from load the original fingerprint image and ends up minutiae extraction passing through the traditional processing which represents in pre-processing such as bineraization, thinning and minutia marking, post-processing such as removal false minutia. This system is allowed to select a region of interest manually and automatically, therefore it is characterized by the possibility to identify the all fingerprints kinds partial and total as shown in figure 18.

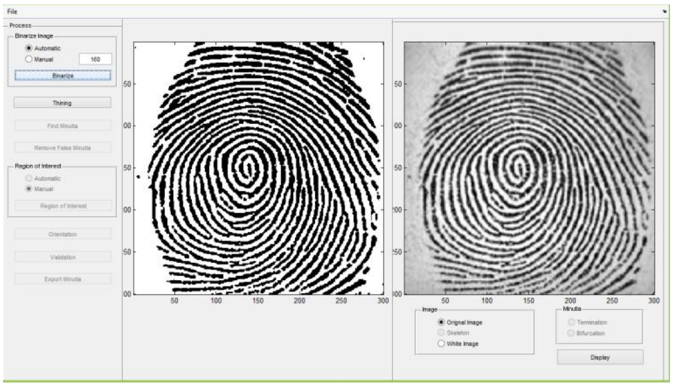

Binarize Image: ● Automatic ○ Manual 160
Region of Interest: ○ Automatic ● Manual
Image: ● Orignal Image ○ Skeleton ○ White Image
Minutia: ○ Termination ○ Bifurcation

Fig. 15: original fingerprint and its binarization image.

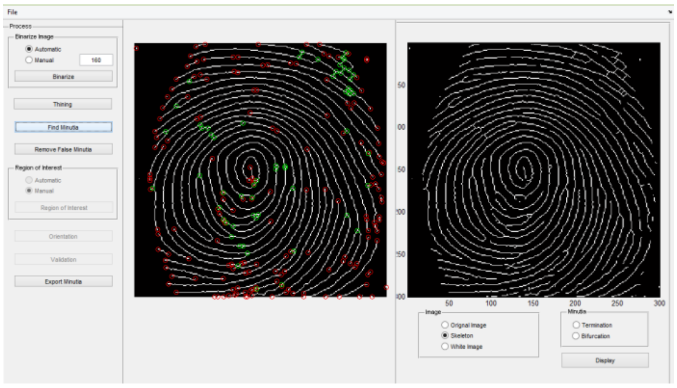

Binarize Image: ● Automatic ○ Manual 160
Region of Interest: ○ Automatic ● Manual
Image: ○ Orignal Image ● Skeleton ○ White Image
Minutia: ○ Termination ○ Bifurcation

Fig. 16: skeleton fingerprint image and minutia marked image.

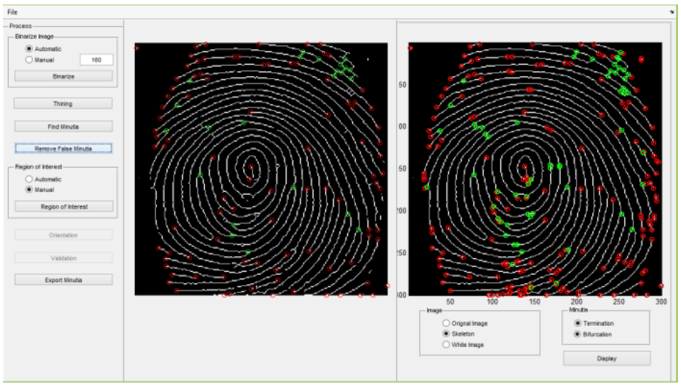

Binarize Image: ● Automatic ○ Manual 160
Region of Interest: ○ Automatic ● Manual
Image: ○ Orignal Image ● Skeleton ○ White Image
Minutia: ● Termination ● Bifurcation

Fig. 17: fingerprint minutia marking and removal false minutia

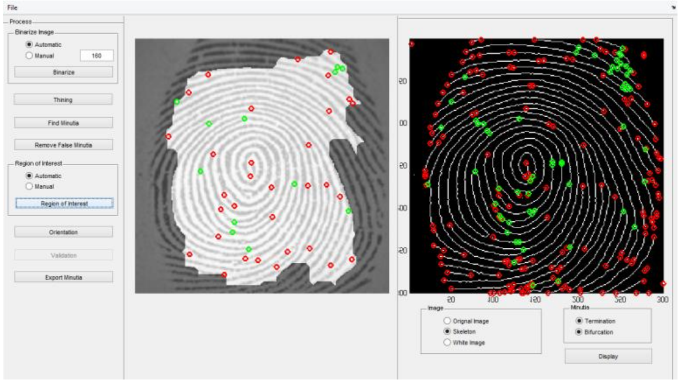
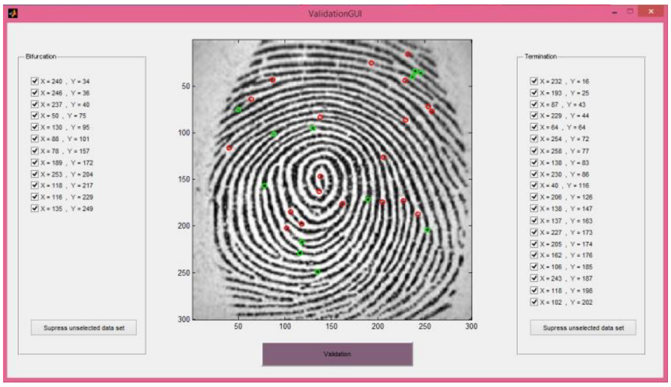

Binarize Image: ● Automatic ○ Manual 160
Region of Interest: ● Automatic ○ Manual
Image: ○ Orignal Image ● Skeleton ○ White Image
Minutia: ● Termination ● Bifurcation

Fig. 18: removal false minutia image ROI.



Bifurcation

☑ X = 240 , Y = 34
☑ X = 246 , Y = 36
☑ X = 237 , Y = 40
☑ X = 50 , Y = 75
☑ X = 130 , Y = 95
☑ X = 88 , Y = 101
☑ X = 78 , Y = 157
☑ X = 189 , Y = 172
☑ X = 253 , Y = 204
☑ X = 118 , Y = 217
☑ X = 116 , Y = 229
☑ X = 135 , Y = 249

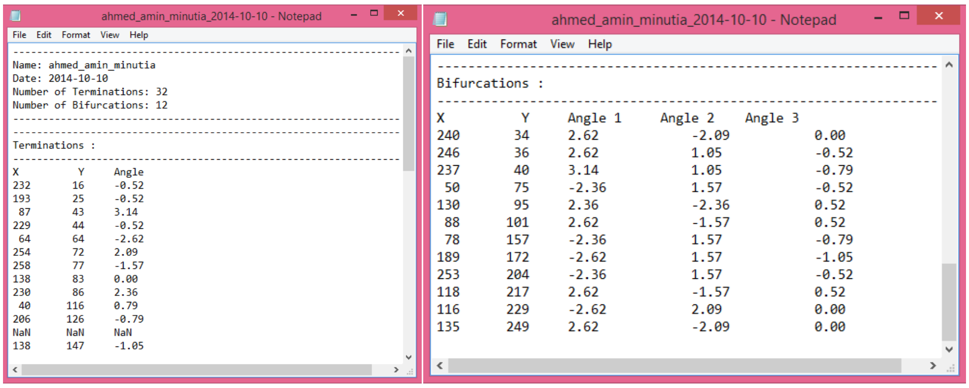
Supress unselected data set

Termination

☑ X = 232 , Y = 16
☑ X = 193 , Y = 25
☑ X = 87 , Y = 43
☑ X = 229 , Y = 44
☑ X = 64 , Y = 64
☑ X = 254 , Y = 72
☑ X = 258 , Y = 77
☑ X = 138 , Y = 83
☑ X = 230 , Y = 86
☑ X = 40 , Y = 116
☑ X = 206 , Y = 126
☑ X = 138 , Y = 147
☑ X = 137 , Y = 163
☑ X = 227 , Y = 173
☑ X = 205 , Y = 174
☑ X = 162 , Y = 176
☑ X = 106 , Y = 185
☑ X = 243 , Y = 187
☑ X = 118 , Y = 198
☑ X = 102 , Y = 202

Supress unselected data set

Fig. 19: validation fingerprint minutiae.


ahmed_amin_minutia_2014-10-10 - Notepad
File Edit Format View Help

```
-----------------------------------------------------------
Name: ahmed_amin_minutia
Date: 2014-10-10
Number of Terminations: 32
Number of Bifurcations: 12
-----------------------------------------------------------
-----------------------------------------------------------
Terminations :
-----------------------------------------------------------
X          Y      Angle
232       16      -0.52
193       25      -0.52
 87       43      3.14
229       44      -0.52
 64       64      -2.62
254       72      2.09
258       77      -1.57
138       83      0.00
230       86      2.36
 40      116      0.79
206      126      -0.79
NaN      NaN      NaN
138      147      -1.05
```

ahmed_amin_minutia_2014-10-10 - Notepad
File Edit Format View Help

```
-----------------------------------------------------------
Bifurcations :
-----------------------------------------------------------
X          Y      Angle 1     Angle 2     Angle 3
240       34      2.62        -2.09       0.00
246       36      2.62        1.05        -0.52
237       40      3.14        1.05        -0.79
 50       75      -2.36       1.57        -0.52
130       95      2.36        -2.36       0.52
 88      101      2.62        -1.57       0.52
 78      157      -2.36       1.57        -0.79
189      172      -2.62       1.57        -1.05
253      204      -2.36       1.57        -0.52
118      217      2.62        -1.57       0.52
116      229      -2.62       2.09        0.00
135      249      2.62        -2.09       0.00
```

Fig. 20: minutia location report.

c) Fingerprint Matching:

Minutiae-based matching is focusing on perform of pairing of minutiae points from a query image (Q) minutia set:

$$Q = \{MT_{Q_1}, MT_{Q_2}, \dots, MT_{Q_k}\}$$

Where,

$$\mathrm{MT}_{\mathrm{Q_i}} = \{\mathrm{x}_{\mathrm{Q_i}}, \mathrm{y}_{\mathrm{Q_i}}, \theta_{\mathrm{Q_i}}\} \ \text{ and } \ 1 \le \mathrm{i} \le \mathrm{k}$$

Forming the minutiae$\left(\mathrm{MT}_{\mathrm{Q_U}}, \mathrm{MT}_{\mathrm{T}_{\tau(\mathrm{U})}}\right)$ pairs with $\tau(\mathrm{U})$ as the mapping permutation of pairs from set Q to T.

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.As mentioned early, it includes three consecutive stages,the first is shape and orientation descriptor stage, the second is registration stage and the third is similarity score calculation.

Figure 21 shows the aim of the first stage to align the query fingerprint minutiae to the reference minutiae which represents as template fingerprint minutiae so as to minimize the total sum of squared Euclidean distances between the corresponding points. The three alignments of translation, scaling and rotation are successively applied, each independently satisfying this least-squares criterion.

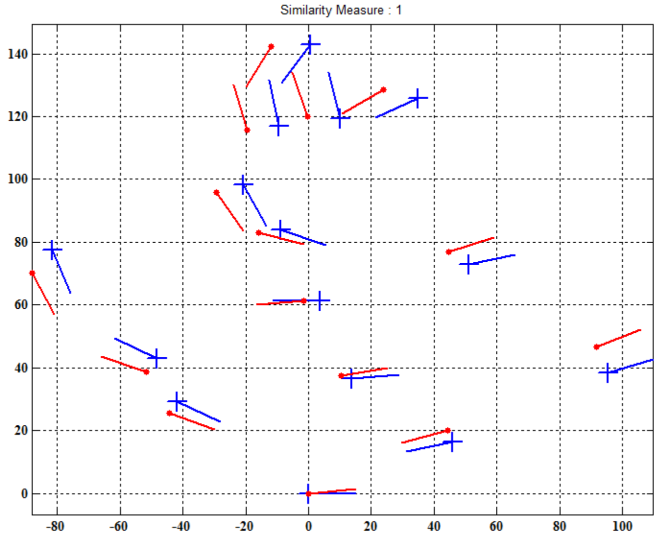


Fig. 21: Improved alignment of query (*---) and template (+---) minutiae pairs.

In the second stage, which consisting of integration between shape context method with orientation descriptor is contributed by two-fold: firstly to prune outlier minutiae pairs, and secondly to provide more information to use in similarity assessment. This method has proven through experimentation its suitability and provides acceptable performance in assessment of fingerprint similarity. However, the resulting minutiae mapping from the application of the algorithm on the contextually based cost histograms produced some unnatural pairs, as illustrated in Figure 22.

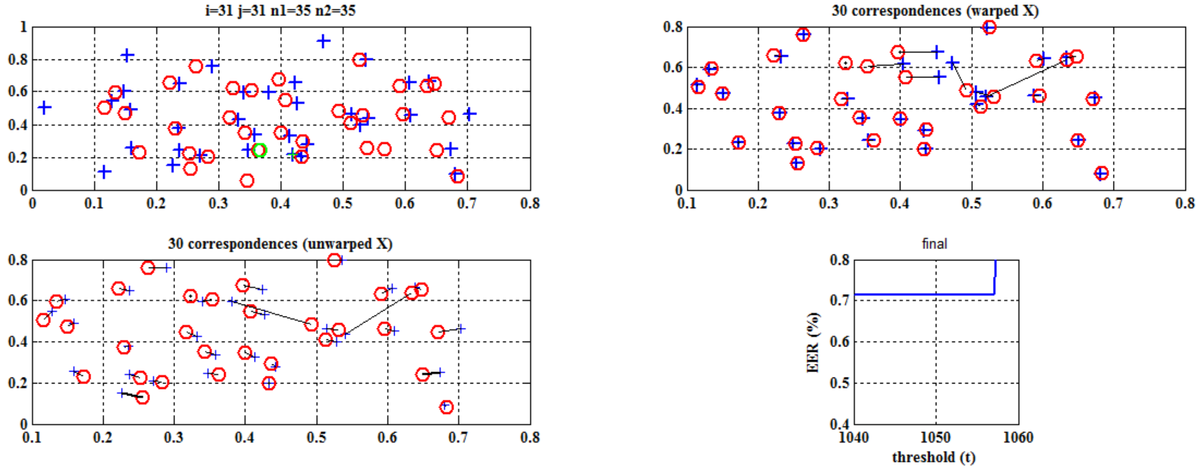


Fig. 22: improved registration between query and template set.

d) Performance Evaluation:

To evaluate any matching algorithm performance [30], some important quantities have to be measured such as: False Non-Match Rate (FNMR) often referred to as False Rejection Rate (FRR), False Match Rate (FMR) often referred to as False Acceptance Rate (FAR), Equal Error Rate (EER), ZeroFNMR , and ZeroFMR.

The FAR is the fraction of impostor fingerprints which are accepted and is calculated as:

$$FAR = \frac{No. of\ impostor\ fingerprints\ accepted}{Total\ No. of\ impostor\ tests}$$

That means, for all databases the first sample of each finger is matched against the first sample of the remaining fingers in the same database. Whereas, each sample in a database is matched against the remaining sample of the same finger to compute FRR by applied the:

$$FRR = \frac{No. of\ genuine\ fingerprints\ rejected}{Total\ No. of\ genuine\ tests}$$

The Equal Error Rate is computed as the point where FAR = FRR. From figure 23, to determine the equal error rate, the intersection line between the two curves is drawn in x-y plane which x and y is representing similarity score and Error Rate respectively. ZeroFMR is defined as the lowest FNMR at which no False Matches occur and ZeroFNMR as the lowest FMR at which no False Non-Matches occur.

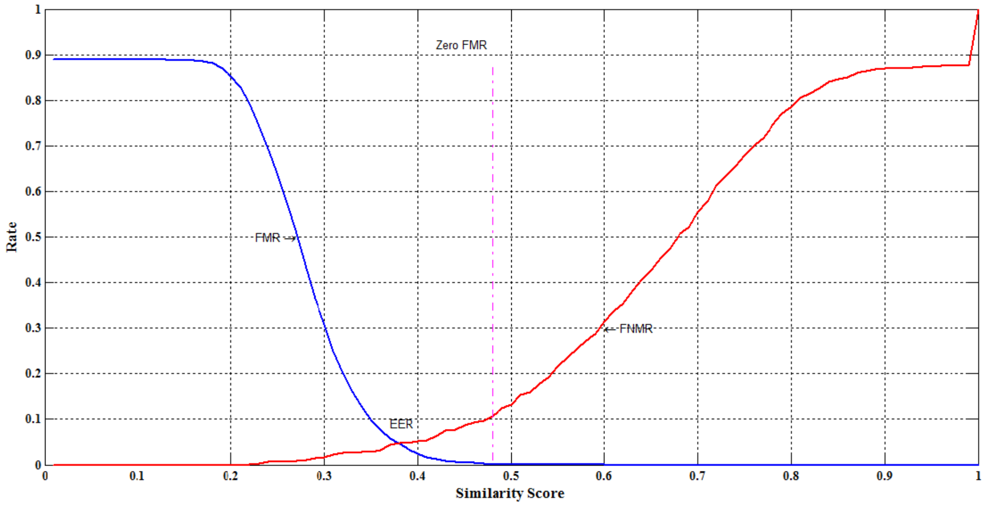


Fig. 23: Similarity score via Error Rate.

A ROC (Receiving Operating Curve) is given where FRR is plotted as a function of FAR as shown in figure 24. As can be shown, the recognition performance is good by comparison with the curve of a good recognition performance system. It is noted that the curve in figure 24 is going to the top left portion of the plotting area and that represents the good recognition performance curve.

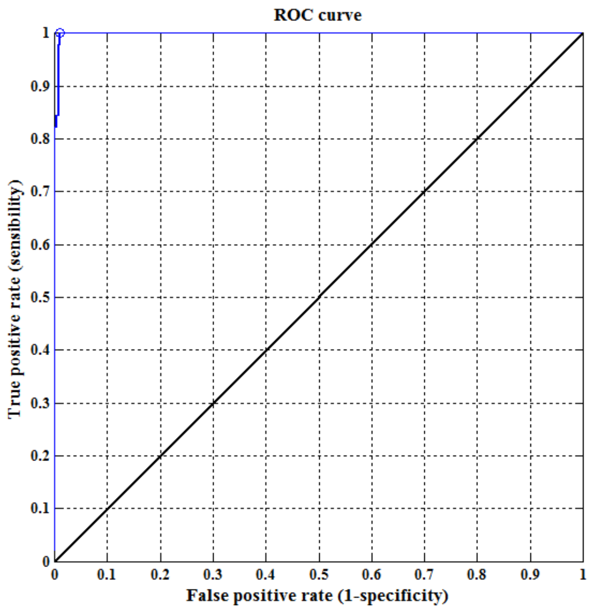


Fig. 24: ROC curve

5. Conclusion:

A fingerprint verification system has been proposed and implemented,to produce a better performance and accuracy. The proposed system is composed of two main parts namely fingerprint minutiae extractor and minutiae matcher. The first part consists of fingerprint image preprocessing, minutia extraction, and post-processing. The preprocessing contains: enhancement of fingerprint image by using Histogram Equalization and Fourier Transform. Then, a locally adaptive binarization method is used to implement the binarized image. The fingerprint image is segmented by two-step approach called block direction estimation, and Region of Interest extraction is done by Morphological operations. Minutia extraction contains fingerprint ridge thinning, enhanced thinning; minutia marking where locating minutia points in the thinned image is implemented by using a Crossing Number (CN) method. The post-processing phase eliminates spurious feature points and then validates the minutiae points. This done through two steps: false minutia removal, and unify termination and bifurcation. The second part of the proposed system is minutia matcher, where the algorithm includes three consecutive stages, shape and orientation descriptor stage, registration stage, and similarity score calculation.

The experimental data is obtained from FVC2002 and FVC2004databases;the size of each database to be used in the test was established as 110 fingers, 8 impressions per finger (880 impressions). The experiments are done on windows 8, processor core i5 and RAM 6 GB. Some statistics are reported to confirm the high performance of the proposed system.

6.REFERENCES

[1] Sarat C. Dass and Anil K. Jain,"Fingerprint-Based Recognition", Technometrics, Vol.49, No.3, pp. 262-276, Aug 2007.

[2] Feng, Jianjiang, and Anil K. Jain, "Fingerprint reconstruction: from minutiae to phase" Pattern Analysis and Machine Intelligence, IEEE Transactions on biometrics compendium, vol. (33), issue (2), pp 209-223, 2011.

[3] M. A. Medina-Pérez, M. García-Borroto, A. E. Gutierrez-Rodríguez, and L. Altamirano-Robles, "Improving Fingerprint Verification Using Minutiae Triplets," Sensors, vol. 12, pp. 3418–3437, 2012.

[4] Chander Kant &RajenderNath, "Reducing Process-Time for Fingerprint Identification System", International Journals of Biometric and Bioinformatics, Volume (3): Issue (1), pp.1-9, 2009.

[5] T. Hatano, T. Adachi, S. Shigematsu, H. Morimura, S. Onishi, Y. Okazaki & H. Kyuragi, "A fingerprint verification algorithm using the differential matching rate". International Conference on Pattern Recognition, vol(3), pp. 30799, 2002.

[6] A. Lindoso , L. Entrena, J. LiuJimenez& E. SanMillan, "Correlation-based fingerprint matching with orientation field alignment", International Conference ICB, Seoul, Korea, pp.713–721, August 27-29, 2007.

[7] G. Eckert, S. M¨uller, T. Wiebesiek, "Efficient Minutiae-Based Fingerprint Matching", MVA2005 IAPR Conference on Machine VIsion Applications, Tsukuba Science City, Japan, pp.554-557, May 16-18, 2005.

[8] J. C. Yang & D.-S. Park, "A fingerprint verification algorithm using tessellated invariant moment features"; Neuro-computing, Volume (71), Issue (10-12), pp.1939–1946, June, 2008.

[9] L. Nanni& A. Lumini, "Descriptors for image-based fingerprint matchers", Expert Systems with Applications, vol(36), issue(10),pp.12414–12422, December 2009.

[10] N. K. Ratha& R. Bolle, Automatic Fingerprint Recognition Systems. Springer Verlag, 2003.

[11] PrateekVerma;YogeshBahendwar; Amrita Sahu; MaheedharDubey," Feature Extraction Algorithm of Fingerprint Recognition", International Journal of Advanced Research in Computer Science and Software Engineering, Vol. 2, issue 10,pp.292-297, October 2012.

[12] Indira Chakravarthy, VVSSS.Balaram, and B.Eswara Reddy, "Overview of FingerprintImage Enhancement & Minutia Extraction", International Journal of Computer Science and Information Technologies, Vol. 4 (1), pp.5 – 9, 2013.

[13] R.Dharmendra Kumar, KaliyaperumalKarthikeyan, and T.Ramakrishna, "Finger Print Image Enhancement Using FFT For Minutia Matching With Binarization", International Journal of Engineering Research & Technology (IJERT), Vol. 1 Issue 8, pp. 1-6,October – 2012.

[14] Komal Sharma, Vinod Kumar Singla, "A Review on Fingerprint Recognition Technique Using Real Minutia Identification"; International Journal of Engineering and Computer Science, Volume - 3 Issue – 8, pp. 7798-7804, August, 2014**.**

[15] WIĘCŁAW Ł., "A Review on Fingerprint Orientation Estimation Methods", Journal of Medical Informatics & Technologies Vol. 21, pp. 95-102,2012.

[16] Ashwin Kumar, Pradeep Kumar., "A New Framework for Color Image Segmentation Using Watershed Algorithm"; Journal of Electronic Imaging, Vol. 2, No. 3, pp. 41-46, 2011.

[17] Salem Saleh Al-amri, N.V. Kalyankar and Khamitkar S.D " Image Segmentation By Using Edge Detection" International Journal on Computer Science and Engineering ,Vol. 02, No. 03, pp. 804-807, 2010.

[18] WUZHILI, "Fingerprint recognition" Student project. Hong Kong Baptist University, April 2002.

[19] B. K. Jang, R. T. Chin, "One-Pass Parallel Thinning: Analysis, Properties and Quantitative Evaluation", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 14, No. 11, pp. 1129-1140, 1992.

[20] RoliBansal, PritiSehgal, and PunamBedi. "Minutiae Extraction from Fingerprint Images-a Review", IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 5, No 3, pp. 74-85, September 2011.

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

[22] NEETA MURMU, and ABHA OTTI, "FINGERPRINT RECOGNITION", Thesis at Electrical Engineering, National Institute of Technology, Rourkela-769008, 2008.

[23] Shu, Xin, and Xiao-Jun Wu., "A novel contour descriptor for 2D shape matching and its application to image retrieval.", Image and vision Computing 29.4 (2011): 286-294.

[24] Cope, James S., and Paolo Remagnino., "Utilizing the hungarian algorithm for improved classification of high-dimension probability density functions in an image recognition problem", Advanced Concepts for Intelligent Vision Systems. Springer Berlin Heidelberg; 2012.

[25] Crum, William R., Thomas Hartkens, and D. L. G. Hill., "Non-rigid image registration: theory and practice." (2014).

[26] Myronenko, Andriy, and Xubo Song., "Point set registration: Coherent point drift." Pattern Analysis and Machine Intelligence, IEEE Transactions on 32.12 (2010): 2262-2275.

[27] Zhu, Yongfang, Sarat C. Dass, and Anil K. Jain., "Statistical models for assessing the individuality of fingerprints." , Information Forensics and Security, IEEE Transactions on 2.3 (2007): 391-401.

[28] D. Maltoni, D. Maio, A.K. Jain and S. Prabhakar., Handbook of Fingerprint Recognition., Springer (New York), 2003.

[29] R. Cappelli, D. Maio and D. Maltoni., "Synthetic Fingerprint-Database Generation", Proc. 16th ICPR, 2002.

[30] D. Maio, D. Maltoni, R. Cappelli, J. L.Wayman, & A. K. Jain, (2002)., FVC2000: Fingerprint Verification Competition. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 24, No. 3, pp. 402-412.