FINGERPRINT IMAGE ENHANCEMENT AND COMPARISON OF METHODS TO LOCATE VALID MINUTIAE

Sumitha.S.M[1], A.N.Mukunda Rao[2]

[1]PG Scholar, [2]Associate Professor, Dept. of Electronics and Communication,
Siddaganga Institute of Technology, Tumkur,Karnataka, (India)

ABSTRACT

This paper presents a method to improve quality of fingerprint image before extracting minutiae points from image. Also it compare the two approaches for false minutiae removal. Fingerprint image undergoes pre-processing steps which include normalisation, segmentation, orientation estimation, freuency estimation, Gabor filtering, binarization and thinning. Crossing number method is used for minutiae extraction. False minutiae removal is done based on average inter-ridge distance and based on connected component labeling. Implementation is done in Visual studio 2010 in C++ language using OpenCV libraries.

Keywords: Enhancement, Fingerprint, False Minutiae, Minutiae, Segmentation

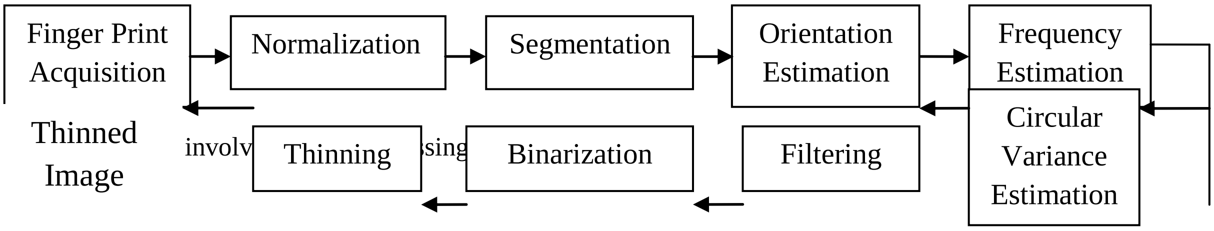
I. INTRODUCTION

Fingerprint identification is one of the most important biometric technologies which have been used in many areas like security, forensics, banking etc. A fingerprint is the pattern of ridges and valleys on the surface of a fingertip. Fingerprints are unique for each individual and remains same throughout the life. User acceptability is higher for fingerprint identification. Challenges involved in fingerprint identification are: (i)High displacements/rotation, (ii)Distortion, (iii)Different pressure and skin conditions, (iv)Feature extraction errors. These may result in spurious or missing features. So, there is requirement for efficient fingerprint enhancement technique to remove noise and improves the clarity of ridges and valleys. The one of the local ridge characteristics is minutiae. There are two types of minutiae points one is ridge ending and other is ridge bifurcation. A ridge ending is defined as the point where a ridge ends abruptly. A ridge bifurcation is defined as the point where a ridge forks or diverges into branch ridges. A good quality fingerprint typically contains about 40–100 minutiae.

A critical step in Automatic Fingerprint matching system is to automatically and reliably extract minutiae from input finger print images. However the performance of the Minutiae extraction algorithm relies heavily on the quality of the input fingerprint image. In order to ensure to extract the true minutiae points it is essential to incorporate the enhancement algorithm. There are two ways in which we can enhance the input fingerprint image. 1. Binarization method. 2. Direct gray-level enhancement. The steps included in Binarization method are local histogram equalization, Wiener filtering, Binarization and thinning. The steps included in direct gray-level enhancement are normalization, orientation estimation, frequency estimation and filtering. Usually crossing number method is used to locate minutiae points.

II. FINGERPRINT ENHANCEMENT

Fingerprint image is captured using fingerprint scanner. Fingerprint image contains noise due to scanner, different pressure or skin conditions. So, it is not suitable to directly extract the features. It has to be enhanced before feature extraction.



2.1 Normalization

Let $I(i,j)$ denote the gray level value of the pixel (i,j) for an $N \times N$ image. The mean M and variance VAR is given by the equations (1) and (2) respectively. Let $G(i,j)$ denotes the normalized gray level value at pixel(i,j), M_0 and VAR_0 are desired mean and variance values respectively, then the normalized image is defined as in the equation (3). Normalization doesn't change the clarity of ridge and valley structures. The main purpose of normalization is to reduce the variations in the gray level values along ridges and valleys [1].

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

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

$$G(i,j) = \begin{cases} M_0 + \sqrt{\dfrac{VAR_0(I(i,j)-M)^2}{VAR}} & if\ I(i,j) > M \\ M_0 - \sqrt{\dfrac{VAR_0(I(i,j)-M)^2}{VAR}} & otherwise \end{cases} \qquad 3$$

2.2 Segmentation

Segmentation is the process of separating the foreground regions in the image from the background regions. The foreground regions correspond to the clear fingerprint area containing the ridges and valleys, which is the area of interest. The background corresponds to the regions outside the borders of the fingerprint area, which do not contain any valid fingerprint information. When minutiae extraction algorithms are applied to the background regions of an image, it results in the extraction of noisy and false minutiae. Thus, segmentation is employed to discard these background regions, which facilitates the reliable extraction of minutiae.

In a fingerprint image, the background regions generally exhibit a very low grey-scale variance value, whereas the foreground regions have a very high variance. Hence, a method based on variance thresholding can be used to perform the segmentation [2]. Firstly, the image is divided into blocks and the grey-scale variance is calculated for each block in the image. If the variance is less than the global threshold, then the block is assigned to be a background region. Otherwise, it is assigned to be part of the foreground.

2.3 Orientation estimation

The orientation field of a fingerprint image defines the local orientation of the ridges contained in the fingerprint. The orientation estimation is a fundamental step in the enhancement process as the subsequent Gabor filtering stage relies on the local orientation in order to effectively enhance the fingerprint image. The least mean square estimation method is used to compute the orientation image [1]. The steps for calculating the orientation at pixel (i, j) are as follows:

(1) Input image I is divided into non-overlapping blocks of size $w \times w$.

(2) The gradients $\partial_x(i, j)$ and $\partial_y(i, j)$ at each pixel (i, j) are computed. The gradient operator may vary from simple *Sobel* operator to more complex *Marr Hildreth* operator.

(3) The local orientation of each block centered at pixel (i, j) is estimated using the following equations.

$$V_x(i, j) = \sum_{u=i-w/2}^{i+w/2} \sum_{v=j-w/2}^{j+w/2} 2\, \partial_x(i, j) \partial_y(i, j) \quad 4$$

$$V_y(i, j) = \sum_{u=i-w/2}^{i+w/2} \sum_{v=j-w/2}^{j+w/2} \left(\partial_x^2(u, v) - \partial_y^2(u, v)\right) \quad 5$$

$$\theta(i, j) = \frac{1}{2} tan^{-1}\left(\frac{V_y(i, j)}{V_x(i, j)}\right) \quad 6$$

Where $\theta(i, j)$ is the least square estimate of the local orientation at the block centered at pixel (i, j).

(4) Smooth the orientation field in a local neighborhood using a Gaussian filter. The orientation image is firstly converted into a continuous vector field, which is defined as:

$$\emptyset_x(i, j) = \cos(2\, \theta(i, j)) \quad 7$$

$$\emptyset_y(i, j) = \sin(2\, \theta(i, j)) \quad 8$$

Where $\emptyset_x$ and $\emptyset_y$ are the *x* and *y* components of the vector field, respectively.

(5) After the vector field has been computed, Gaussian smoothing is then performed as follows:

$$\emptyset_x'(i, j) = \sum_{u=-w_\emptyset/2}^{w_\emptyset/2} \sum_{v=-w_\emptyset/2}^{w_\emptyset/2} W(u, v) . \emptyset_x(i - uw, j - vw) \quad 9$$

$$\emptyset_y'(i, j) = \sum_{u=-w_\emptyset/2}^{w_\emptyset/2} \sum_{v=-w_\emptyset/2}^{w_\emptyset/2} W(u, v) . \emptyset_y(i - uw, j - vw) \quad 10$$

Where *G* is a Gaussian low-pass filter of size $w_\emptyset \times w_\emptyset$.

(6) The final smoothed orientation field O at pixel (i, j) is defined as:

$$O(i, j) = \frac{1}{2} tan^{-1}\left(\frac{\emptyset_y'(i, j)}{\emptyset_x'(i, j)}\right) \quad 11$$

2.4 Frequency estimation

In addition to the orientation, another important parameter that is used in the construction of the Gabor filter is the local ridge frequency. The frequency image represents the local frequency of the ridges in a fingerprint [1].

Let G and O be the normalized image and oriented image respectively, then the steps involved in local ridge frequency estimation are

(1) Divide G into blocks of size $w \ x \ w$.

(2) For each block centered at pixel (i,j) compute an oriented window of size $l \ x \ w$ and then compute the signature $X[0], X[1], X[2] \dots \dots \dots . . X[l-1]$, the ridges and valleys within oriented window where

$$X[k] = \frac{1}{w}\sum_{d=0}^{w-1} G(u,v), \qquad k = 0,1, \dots \dots, l-1 \tag{12}$$

$$u = i + \left(d - \frac{w}{2}\right) cosO(i,j) + \left(k - \frac{l}{2}\right) sinO(i,j) \tag{13}$$

$$v = j + \left(d - \frac{w}{2}\right) sinO(i,j) + \left(\frac{l}{2} - k\right) cosO(i,j) \tag{14}$$

If no minutiae or singular points appear in the oriented window, the x-signature forms a discrete sinusoidal shape wave, which has the same frequency as that of the ridges and valleys in the oriented window. Therefore, the frequency of ridges and valleys can be estimated from the x-signature. Let $T[i,j]$ be the average number of pixels between two consecutive peaks in the x-signature, then the frequency $\Omega(i,j)$ computed as $\Omega(i,j) = \frac{1}{T(i,j)}$. If no consecutive peaks can be detected from the x-signature , then the frequency is assigned a value of -1 to separate it from the valid frequency values.

(3) For a fingerprint image scanned at a fixed resolution the value of the frequency of the ridges and valleys in a local neighborhood lies in a certain range. If the estimated value of the frequency is out of this range, then the frequency is assigned a value of -1 to indicate that the valid frequency cannot be obtained.

(4) The blocks in which minutiae and/or singular points appear and /or ridges and valleys are corrupted do not form a well-defined sinusoidal shaped wave. The frequency values for these blocks need to be interpolated from the frequency of the neighboring blocks which have a well-defined frequency. The interpolation is performed as follows:

(a)For each block centered at (i,j)

$$\Omega'(i,j) = \begin{cases} \Omega(i,j) & if \Omega(i,j) \neq -1 \\ \dfrac{\sum_{u=-w_\Omega/2}^{w_\Omega/2} \sum_{v=-w_\Omega/2}^{w_\Omega/2} W_g(u,v)\mu((i-uw, j-vw))}{\sum_{u=-w_\Omega/2}^{w_\Omega/2} \sum_{v=-w_\Omega/2}^{w_\Omega/2} W_g(u,v)\delta((i-uw, j-vw)+1)} & otherwise \end{cases} \tag{15}$$

Where,

$$\mu(x) = \begin{cases} 0 & if \ x \leq 0 \\ x & otherwise \end{cases} \tag{16}$$

$$\delta(x) = \begin{cases} 0 & if \ x \leq 0 \\ 1 & otherwise \end{cases} \tag{17}$$

Wg is a discrete Gaussian kernel with fixed mean and variance and w_Ω is the size of the kernel.

(b) If there exists at least one block with the frequency value of -1, then swap Ω and Ω' and repeat step(a).

(5) Inter-ridge distances change slowly in a local neighborhood. A low –pass filter can be used to remove the outliers occurred due to such changes.

$$F(i,j) = \sum_{u=-w_\Omega/2}^{w_l/2} \sum_{v=-w_\Omega/2}^{w_l/2} W_l(u,v)\Omega'(i-uw, j-vw) \tag{18}$$

Where, W_l is two dimensional low –pass filter and w_l is the size of the filter.

The configuration of parallel ridges and valleys with well-defined frequency and orientation in a fingerprint image provide useful information which helps in removing undesired noise. The sinusoidal –shaped waves of ridges and valleys vary slowly in a local constant orientation. Therefore a suitable band-pass filter that is tuned to the corresponding frequency and orientation may be used if necessary to remove the undesired noise and preserve the true ridge and valley structures.

2.5 Filtering

Once the ridge orientation and ridge frequency information has been determined, these parameters are used to construct the even-symmetric Gabor filter. A two dimensional Gabor filter consists of a sinusoidal plane wave of a particular orientation and frequency, modulated by a Gaussian envelope. Gabor filters are employed because they have frequency-selective and orientation-selective properties. These properties allow the filter to be tuned to give maximal response to ridges at a specific orientation and frequency in the fingerprint image. Therefore,

a properly tuned Gabor filter can be used to effectively preserve the ridge structures while reducing noise [1].

An even-symmetric Gabor filter has general form as:

$$h(X,Y:\emptyset,f) = exp\left\{-\frac{1}{2}\left[\frac{X_\emptyset^2}{\delta_X^2}+\frac{Y_\emptyset^2}{\delta_Y^2}\right]\right\}\cos(2\pi f X_\emptyset) \quad 19$$

$$X_\emptyset = X\cos\emptyset + Y\sin\emptyset \quad 20$$

$$Y_\emptyset = -X\sin\emptyset + Y\cos\emptyset \quad 21$$

Where, $\emptyset$ is the orientation of Gabor filter, f is the frequency of a sinusoidal plane wave, δ_X and δ_Y are the standard deviations of Gaussian envelope along X and Y axes, respectively.

Let G be normalized fingerprint image, O be the orientation image, F be the frequency image and S be the segmentation mask, the enhanced image E is obtained as follows:

$$E(i,j) = \begin{cases} 255 & if\ S(i,j) = 0 \\ \sum_{u=-w_g/2}^{w_g/2}\sum_{v=-w_g/2}^{w_g/2} h(u,v:O(i,j),F(i,j))G(i-u,j-v) & otherwise \end{cases} \quad 22$$

Where, w_g is the size of Gabor filter.

2.6 Binarization

Binarization is the process that converts a grey level image into a binary image. This improves the contrast between the ridges and valleys in a fingerprint image and consequently facilitates the extraction of minutiae. A locally adaptive binarization method is performed. It is the mechanism of transforming a pixel value to1 if the value is larger than the mean intensity value of selected block. Otherwise it is 0.

2.7 Thinning

Thinning is a morphological operation that successfully erodes away the foreground pixels until they are one pixel wide. The thinning operation is done using two sub-iterations [3]. Each sub-iterations begins by examining the neighborhood of each pixel in the binary image and based on a particular set of pixel deletion criteria it checks whether the pixel can be deleted or not. These sub-iterations continue until no more pixels can be deleted. It also preserves the connectivity of ridges. Guo-Hall thinning algorithm is used and steps are as follows:

P_9	P_2	P_3
P_8	P_1	P_4
P_7	P_6	P_5

(1) Calculate $C(P_1)$

$$C(P_1) = !P_2\&(P_3|P_4) + !P_4\&(P_5|P_6) + !P_6\&(P_7|P_8) + !P_8\&(P_9|P_2) \quad 23$$

It indicates the number of distinct 8-connected component of 1's in $P_1's$ 8-neighborhood.

(2) Calculate $N(P_1)$

$$N(P_1) = MIN[N_1(P_1), N_2(P_1)] \quad 24$$

$where,$

$$N_1(P_1) = (P_9|P_2) + (P_3|P_4) + (P_5|P_6) + (P_7|P_8) \quad 25$$

$$N_2(P_1) = (P_2|P_3) + (P_4|P_5) + (P_6|P_7) + (P_8|P_9) \quad 26$$

$N_1(P_1)$ and $N_2(P_1)$ each break the ordered set of $P_1's$ neighboring pixel into four pairs of adjoining pixels and count the number of pairs which contain one or two 1's.

(3) Calculate $(P_2|P_3|!P_5)|P_4$ for add iteration or $(P_6|P_7|!P_9)\&P_8$ for even iteration.

(4) Delete the pixel $P(i,j)$, if $C(P_1) = 1$ and $2 \leq N(P_1) \leq 3$ and

a. $(P_2|P_3|!P_5)|P_4 = 0$ for add iteration.

b. $(P_6|P_7|!P_9)\&P_8 = 0$ for even iteration.

(5) Carry out the step 1 to 4 for all the pixels in the image.

(6) If pixel is deleted, go back to step 1 else stop the thinning process.

III. MINUTIAE

Minutiae are the minute, precise and trivial details. These mainly include ridge endings and ridge bifurcations which are shown in figure (2).

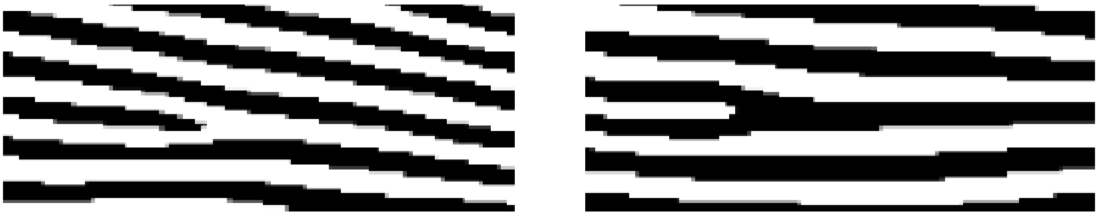


Figure 2: Left Ridge Ending and Right Ridge Bifurcation

3.1 Minutiae marking

The most commonly used method of minutiae extraction is the crossing number concept [4]. In this method, the thinned image is used where the ridge pattern is eight connected. The minutiae are extracted by scanning the local neighborhood of each ridge pixel in the eight-neighborhood. Using the properties of the crossing number the ridge pixel can then be classified as a ridge ending, bifurcation or non-minutiae point. The crossing number CN for a ridge pixel P is given by,

$$CN = \frac{\sum_{i=1}^{8} |P_i - P_{i+1}|}{2} \quad (27)$$

Where, P_i is the pixel value in the neighborhood of P.

P_4	P_3	P_2
P_5	P	P_1
P_6	P_7	P_8

Figure 3: Order of Scan Around Pixel P

For a pixel P, its eight neighboring pixels are scanned as shown in figure (3). After obtaining the crossing number CN, minutiae are marked. If $CN = 1$ then pixel P is ridge ending point and if $CN = 3$ then pixel P is ridge bifurcation point.

3.2 Removal of False Minutiae Using Connected Component Labeling Approach

The pre-processing stage does not totally heal the fingerprint image. Through the scanned fingerprint image gets better clarity through pre-processing stage, each of all the earlier stages occasionally introduce some small amount of errors which later lead to spurious minutiae. Therefore removal of false minutiae is essential to keep the fingerprint verification system effective.

To removing false minutiae following steps are carried out for each candidate minutiae (ridge ending or ridge bifurcation)[5]:

1. Create and initialize with 0 an image L of size $W \times W$. Each pixel of L corresponds to a pixel of the thinned image which is located in a $W \times W$ neighborhood centered in the candidate minutia.

2. Label with -1 the central pixel of L (Figure (4a), Figure (5a)). This is the pixel corresponding to the candidate minutia point in the thinned ridge map image.

3. If the candidate minutia is a ridge ending then:

(a) Label with 1 all the pixels in L which correspond to pixels connected with the candidate ridge ending in the thinned ridge map image (Figure (4b)).

(b) Count the number of 0 to 1 transitions (T_{01}) met when making a full clockwise trip along the border of the L image (Figure (2c)).

(c) If $T_{01} = 1$, then validate the candidate minutia as a true ridge ending.

4. If the candidate minutia is a ridge bifurcation then:

(a) Make a full clockwise trip along the 8 neighborhood pixels of the candidate ridge bifurcation, and label in L with 1, 2 and 3 respectively the three connected components met during this trip (Figure (5b)).

(b) For each l = 1, 2, 3 (Figure (5c), (5d), (5e)), label with l all pixels in L which:

i. have the label 0, ii. are connected with an l labeled pixel ,iii. correspond to 1 valued pixels in the thinned ridge map.

(c) Count the number of 0 to 1, 0 to 2 and 0 to 3 transitions met when making a full clockwise trip along the border of the L image. The above three numbers are denoted by T_{01}, T_{02}, T_{01} respectively as shown in Figure (5f).

(d) If T_{01} = 1 & T_{02} = 1 & T_{03} = 1, then validate the candidate minutia as a true ridge bifurcation.

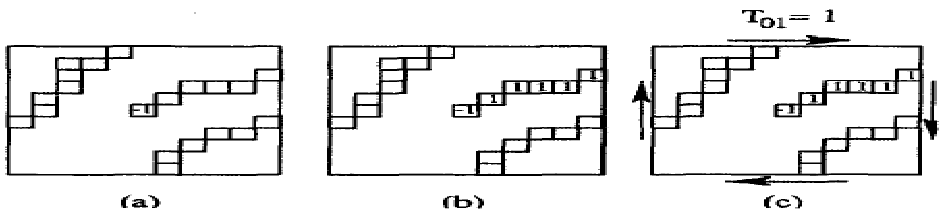


Figure 4: Steps to Validate Ridge Ending

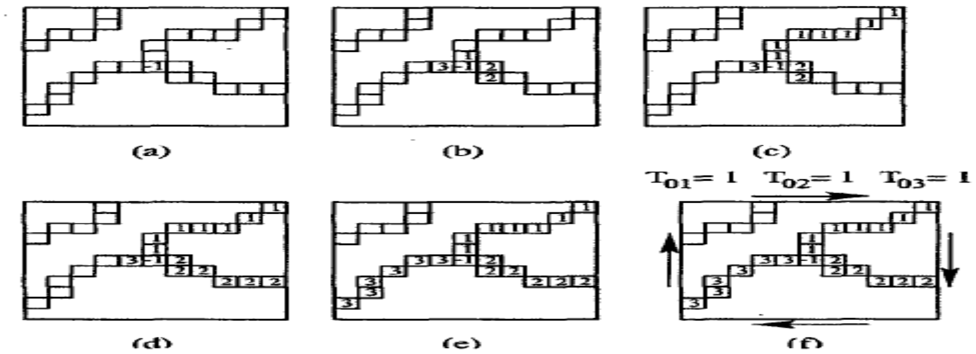


Figure 5: Steps to Validate Ridge Bifurcation

The dimension W of the neighborhood analyzed around each candidate minutia is chosen larger than two times the average distance between two neighborhood ridges. By this minutiae belonging to the same ridge are canceled.

3.3 Removal of False Minutiae Using Plus Rule

There is existence of false minutiae at boundary of fingerprint even after the above method of false minutiae removal. Some of pixels in border of fingerprint is marked as ridge ending but they are not actual ridge ending. These false minutiae are removed using plus rule [6]. The rule creates a plus sign on each minutia. It works on each minutia, to find a white pixel (i.e. '1' value) across these lines. If a white pixel is not detected, in any line, the minutia is marked as boundary minutia and hence removed.

3.4 Removal of False Minutiae Based on Average Inter-Ridge Distance

False ridge breaks due to insufficient amount of ink & ridge cross connections due to over inking may occur. Also some of the pre-processing steps introduce some spurious minutia points in the image. So to keep the recognition system consistent these false minutiae need to be removed.

The inter ridge distance D which is the average distance between two neighboring ridges is considered. For this scan each row to calculate the inter ridge distance using the formula [7]:

$$\text{Inter ridge distance} = \frac{\text{sum all pixels with value } 1}{\text{row length}} \qquad 28$$

Finally an averaged value over all rows gives D.

1. If d(bifurcation, termination) < D & the 2 minutia are in the same ridge then remove both of them.
2. If d(bifurcation, bifurcation) < D & the 2 minutia are in the same ridge them remove both of them.
3. If d(termination, termination) < D & the 2 minutia are in the same ridge them remove both of them.

where $d(X, Y)$ is the distance between 2 minutia points.

IV. EXPERIMENTAL RESULTS

Database of fingerprint is collected using optical fingerprint sensor Secugen Hamster Plus. The size of the ingerprint image captured is pixels and the resolution is 500 DPI. Fingerprint image captured is in bitmap format (gray image). Fingerprints of right hand forefinger are collected from 10 persons and from each person 20 fingerprints of same finger are collected.

Figure (6) is input fingerprint image captured using an optical fingerprint sensor. It has varied range of gray levels which is reduced using normalization, Figure (7) is the normalized image of input fingerprint, in which gray level variations are low in fingerprint ridges. Next is to separate fingerprint from the background. Figure (8) shows the segmentation mask where black region indicated the background and white region indicates fingerprint, Figure (9) is segmented normalized image obtained using segmentation mask on normalized image. Figure (10) is enhanced image obtained by gabor filtering the segmented normalized image using orientation and frequency. Figure (11) is inverse binarized image of enhanced image. Figure (12) is the thinnied image obtained by applying morphological operation on binarized image. The size of ridge is reduced to 1 pixel. Figure (13) is the minutiae marked image obtained using cross number method. Here, green circles indicates ridge endings and red circle indicates ridge bifurcation. In this lot of false minutiae points are present.In Figure (14) and Figure (16) false minutiae present over the fingerprint is removed leaving valid minutiae using average inter-ridge distance based method. In Figure (15) and Figure (17) false minutiae present over the fingerprint is removed leaving valid minutiae using connected component labeling method. we can see that lot of valid minutiae are detected and lot of false minutiae are retained using average inter-ridge distance based method. But almost all valid minutiae are detected with very few false minutiae using connected componet labeling approach.

Figure 6:Input Image

Figure 7: Normalised Image

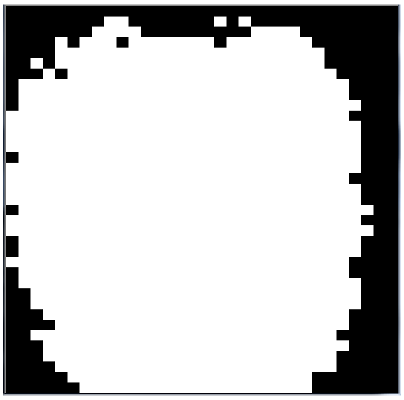


Figure 8: Segmentation Mask

Figure 9: Segmented Normalised Image

Figure 10: Enhanced Image

Figure 11: Binarised Image

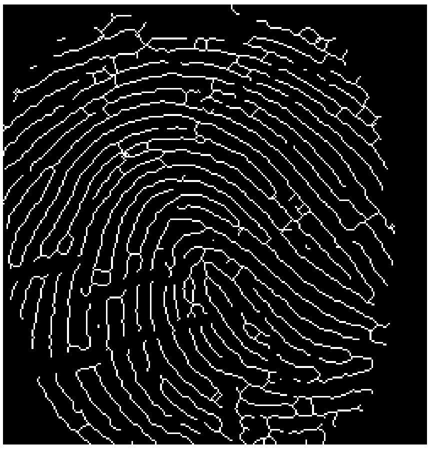


Figure 12: Thinned Image

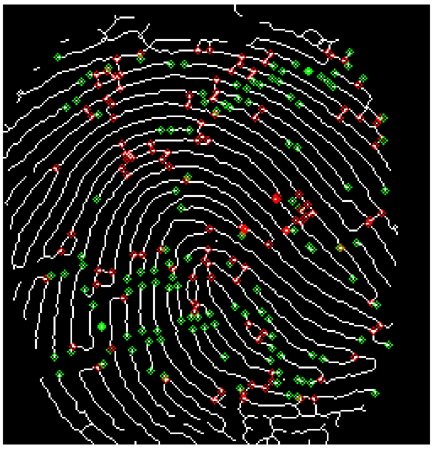


Figure 13: Minutiae Marked Image

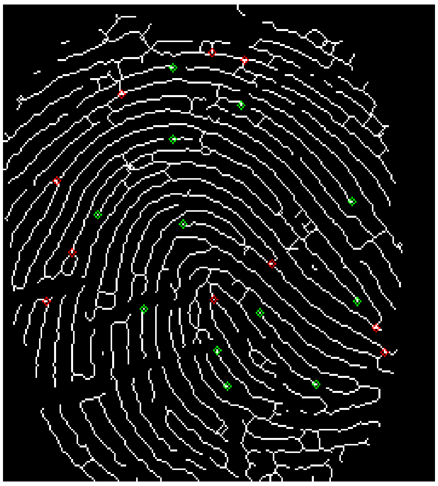


Figure 14: Valid Minutiae Using Average Connected Component inter-ridge distance Method

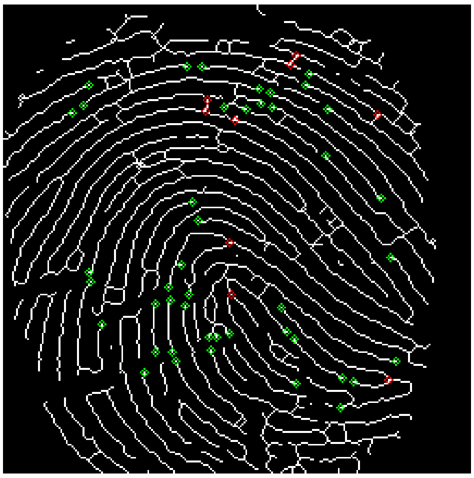


Figure 15: Valid minutiae using labeling method

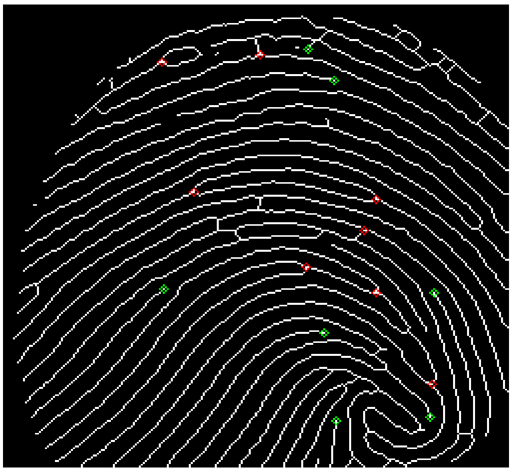


Figure 16: Valid minutiae using average connected component inter-ridge distance method

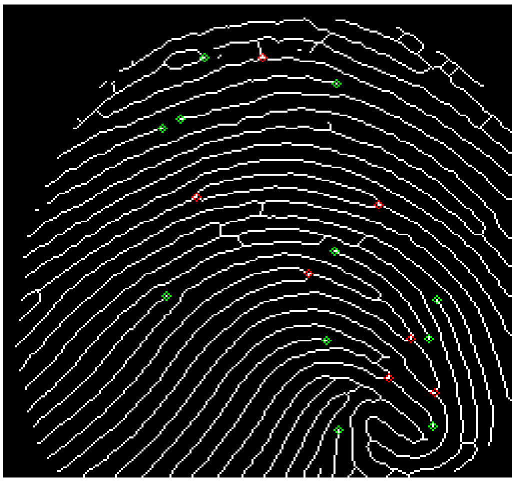


Figure 17: Valid minutiae using labeling method

V. CONCLUSION

The pre-processing, minutiae marking of fingerprint image is implemented. Also the two approaches for false minutiae removal is implemented and their performance is compared. Results obtained show that fingerprint image is enhanced better after pre-processing. Performance of removal of false minutiae is better using connected component labeling approach when compared to average inter-ridge distance based approach. It retains most valid minutiae with very few false minutiae. Enhancement of fingerprint image avoids existence of some false minutiae. But still some false minutiae will occur even after enhancement is removed by false minutiae removal methods. Using best method for false minutiae removal accuracy of fingerprint authenticaction system can be increased. i.e.false acceptance rate and false rejection rate can be achieved low.

REFERENCES

[1] Lin Hong, Yifei Wan, and Anil Jain,"Fingerprint Image Enhancement: Algorithm and Performance Evaluation", IEEE Transactions on Pattern Analysis And Machine Intelligence, Vol. 20, No. 8, August 1998.

[2] Ishmael S. Msiza, Mmamolatelo E. Mathekga, Fulufhelo V. Nelwamondo and Tshilidzi Marwala2," Fingerprint Segmentation: An Investigation Of Various Techniques And A Parameter Study Of A Variance-Based Method", International Journal of Innovative Computing, Information and Control, Volume 7, Number 9, September 2011.

[3] Z. Guo and R. Hall, "Parallel thinning with two-subiteration algorithms," Communications of the ACM, vol. 32, pp. 359–373, Mar 1989.

[4] Manvjeet Kaur, Mukhwinder Singh, Akshay Girdhar, and Parvinder S. Sandhu," Fingerprint Verification System using Minutiae Extraction Technique, World Academy of Science, Engineering and Technology,Vol:2, Oct 2008.

[5] Marius Tico, Pauli Kuosmanen, "An Algorithm for Fingerprint Image Post processing", IEEE Conference Record of the Thirty-Fourth Asilomar Conference on Signals, Systems and Computers, pp. 1735 1739, vol.2, Nov 2000.

[6] Shahida Jabeen, Shoab A. Khan, "A Hybrid False Minutiae Removal Algorithm with Boundary Elimination", IEEE International Conference on System of Systems Engineering, pp. 1-6, June 2008.

[7] Sachin Harne, K. J. Satao ," Minutiae Fingerprint Recognition Using Hausdorff Distance ",UNIASCIT, Vol 1 (1), pp. 16-22, 2011.