Fingerprint Image Enhancement based on Threshold Fast Discrete Curvelet Transform (FDCT) and Gabor Filters

Hany Hashem Ahmed
Faculty of Engineering,
Mansoura University, Egypt

Hamdy M. Kelash
Faculty of Electronic
Engineering, Menoufia
University, Menouf, Egypt

Maha Tolba, Mohamed
Badawy
Faculty of Electronic
Engineering, Menoufia
University, Menouf, Egypt

ABSTRACT

The most important stages of Automatic Fingerprint Identification System (AFIS) are enhancement stage, features extraction stage, and matching stage. The main purpose of the enhancement stage is to increase the clarity of the fingerprint image, convert poor quality image to good quality image, and prepare the image for features extraction stage. Both of the two last stages (features extraction & matching stage) depend heavily on the enhancement stage, therefore this paper focus on the enhancement stage. Practically most of the input fingerprint images are corrupted by noise, body conditions, and environmental factors. Therefore it is necessary to use an effectively enhancement method. This paper present new efficient fingerprint image enhancement algorithm works by performing threshold on Fast Discrete Curvelet Transform (FDCT) domain and applying Gabor Filters. The proposed algorithm reduce the fingerprint image noise by using threshold fast discrete curvelet transform, then apply Gabor filters to enhance and increase the clarity of the image. The performance of proposed enhancement method is evaluated on the basis of Peak Signal to Noise Ratio (PSNR), Absolute Mean Brightness Error (AMBE) and the Computation Time (CT). Experimental results, and the comparison table between proposed enhancement algorithm against traditional methods at the end of this paper show that the proposed method is computationally efficient, with the same level of the enhancement performance. The proposed enhancement method is compared with other traditional fingerprint enhancement techniques. The performance is evaluated on the basis of Peak Signal to Noise Ratio (PSNR), Absolute Mean Brightness Error (AMBE) and the Computation Time (CT). Experimental results show that the proposed method is computationally efficient, with the same level of the enhancement performance.

General Terms

Pattern Recognition – Image Processing

Keywords

AFIS – Fingerprint image enhancement – HE – FFT Enhancement – Gabor filter – FDCT – PSNR – AMBE.

1. INTRODUCTION

The increasing awareness of society toward security threat has great effects in building new ways to protect software, hardware, building as well as network system from outside party attacks. Biometrics are the most popular approach for data security, they refer to identifying a person based on his physiological or behavioral characteristics. Types of existing biometrics are fingerprint, iris-scan, speech, face and voice.

Fingerprint identification is one of the oldest and most popular biometric technologies and it is used in criminal investigations, civilian, commercial applications, and so on. The fingerprint of each person is unique and remains unchanged over a lifetime. A fingerprint consists of a pattern of ridges (lines across fingerprints) and valleys (spaces between ridges) in a finger. The pattern of the ridges and valleys is unique and permanent for each individual. A ridge is defined as a single curved segment, and a valley is the region between two successive ridges as shown in figure 1. The minutiae, which are the local discontinuities in the ridge flow pattern, provide the features that are used for identification. The two most prominent local ridge characteristics (minutiae) are: (1) ridge ending (termination), and (2) ridge bifurcation. A ridge ending is defined as the point where the ridge curve terminates. A ridge bifurcation is defined as the point where a ridge splits from a single path into branch ridges [1].
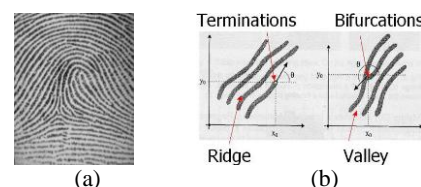

Fig 1: (a) Fingerprint pattern (b) minutiae

The most popular application for fingerprint identification is the Automated Fingerprint Identification System (AFIS). AFIS is a type of biometric identification system that uses digital imaging technology to analyze, store, and capture a fingerprint image, which then can be compared to a database of fingerprint records to help determine the identity of an individual. AFIS is commonly used in law enforcement where sets of prints are recorded at crime scenes, digitally captured and then compared to fingerprint records in the AFIS system. Civilian uses of AFIS include fingerprint scanners for security purposes. In the past the acquisitions of fingerprint images was performed by using ink-technique. Now days they are captured as live-scan digital images acquired by directly sensing the fingerprint surface with an electronic fingerprint scanner. Some of the captured fingerprint images are poor in quality, which corrupts the accuracy of fingerprint recognition. Hence, fingerprint image enhancement is the first step in most AFIS, which improves the quality of the fingerprint image and increasing the reliability of fingerprint recognition [2, 3]. The main purpose of the fingerprint image enhancement step is to improve the clarity of ridge structures of fingerprint images in recoverable regions. This paper is organized as follows: Section 2 discuss some of the existing

fingerprint image enhancement techniques such as histogram equalization HE, FFT Enhancement, and Gabor filter. Section 3 deals with the proposed algorithm of the fingerprint image enhancement which use fast discrete curvelet transform with Gabor filter. Section 4 deals with the comparison of the enhancement techniques and experiment results. Section 5 gives a brief summery and conclusion which summarizes the paper, and finally outlines future work.

2. RELATED WORKS

This section presents a survey on traditional and classical fingerprint image enhancement techniques such as Histogram Equalization, Fast Fourier Transform, and Gabor Filter.

2.1 Fingerprint Enhancement using Histogram Equalization (HE)

Histogram equalization is a computer image processing technique used to improve contrast in the images. It is the oldest and simplest technique used for enhancing fingerprint images. HE is a method used for adjusting image intensities to enhance contrast of the fingerprint image. The main idea of this technique is to stretch or re-estimate the intensity values of image pixels to make the intensity distribution uniform [4, 5]. Consider *(R)* be a given grayscale image represented as $(m \times n)$ matrix of integer pixels intensity values ranging from *0* to *(L-1)* gray level. Where *L* is the number of possible intensity values, often *256*.

Let (P_r) is the probability of an occurrence of a pixel of level *i* in the image. Then

$$P_r(i) = \frac{n_i}{N} \qquad 0 \le i \le 255 \qquad (1)$$

Where n_i is the number of pixels with intensity *i*, and *N* is the total number of pixels (if the image 8x8 matrix, i.e. *N = 64*).
Let us also define the cumulative distribution function (*cdfr*) as:

$$cdf_r(i) = \sum_{j=0}^{i} P_r(j) \qquad (2)$$

Then the general histogram equalization formula is:

$$h(r) = round\left(\frac{cdf_r(i) - cdf_{min}}{(m \times n) - cdf_{min}} \times (L-1)\right) \qquad (3)$$

Where cdf_{min} is the minimum value of the cumulative distribution function, $(m \times n)$ gives the total number of image's pixels, and L is the number of grey level, often 256. Fingerprint image before and after HE is shown in Figure 2.
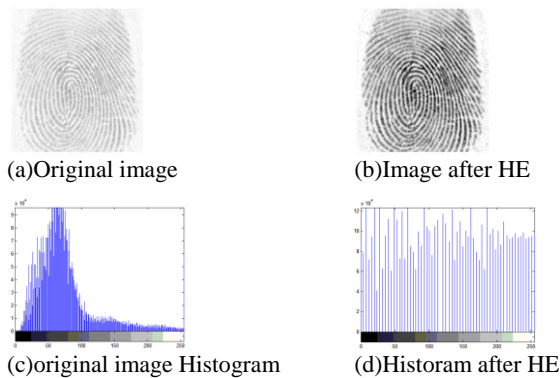
(a)Original image

(b)Image after HE

(c)original image Histogram

(d)Historam after HE

Fig 2: (a) Original image. (b) Image after applying HE. (c) Histogram of original image. (d) Histogram after applying HE.

2.2 Fingerprint Enhancement using Adaptive Histogram Equalization (AHE)

Adaptive Histogram Equalization (AHE) is an improved version of the previous HE enhancement algorithm, in which the adaptive method computes several histograms, each histogram corresponding to a distinct section of the image, and uses them to redistribute the lightness values of the image. For this reason, the adaptive method is more suitable than ordinary HE for improving the local contrast of an image and bringing out more detail. Ordinary histogram equalization works well when the distribution of pixel values is similar throughout the image. However, when the image contains regions that are significantly lighter or darker, the contrast in those regions will not be sufficiently enhanced. In this case adaptive histogram equalization (AHE) is most suitable than ordinary histogram equalization (HE). Figure 3 shows the difference between HE and AHE [6].
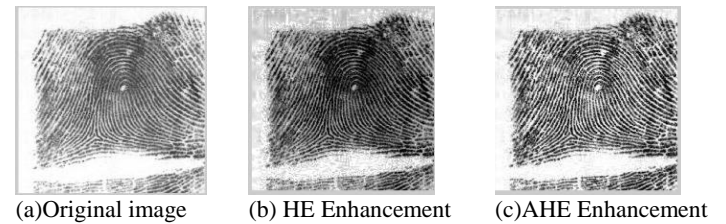
(a)Original image   (b) HE Enhancement   (c)AHE Enhancement
Fig 3: AHE versus HE

2.3 Fingerprint Enhancement using 2D Fourier Transform (2DFT)

The Two Dimensions Fourier Transform (2DFT) is a transformation technique that transforms the image from spatial domain to frequency domain. This algorithm consists of the following steps: Normalization, Segmentation, Orientation Estimation, Fourier Transform, Butterworth Filters, Inverse Fourier Transform, and Reconstruction. These steps are shown in Figure 4 [7].
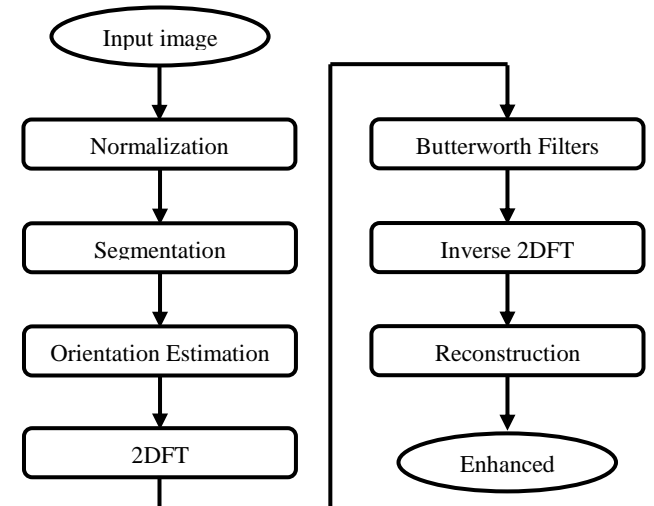
Input image

Normalization

Segmentation

Orientation Estimation

2DFT

Butterworth Filters

Inverse 2DFT

Reconstruction

Enhanced

Fig 4: The algorithm of enhancement using 2DFT

2.3.1 Normalization
The first step in this algorithm is the normalization of the fingerprint image so that it has a pre-specified mean and variance. Normalization step reduces the differences in the grey level values along the ridges and valleys, and enhances the contrast of the fingerprint image, so that the ridges and valleys can be easily distinguished.

2.3.2 Segmentation

It is done to separate the foreground regions (area of fingerprint containing the ridges and valleys) from the background regions in the image.

2.3.3 Orientation Estimation

Orientation image estimation is done to estimate the dominant direction of the ridges and valleys in the fingerprint image. An image window is used to estimate the orientation of each point in the raw image. The window is rotated in 16 equally spaced directions and then the projections of these directions are calculated along the y-direction. The projection with the maximum variance is fixed as the orientation of the pixel. This process is repeated to estimate the orientation for each pixel in the image.

2.3.4 2DFT

The main purpose of this step is to convert the image from spatial domain to frequency domain.

2.3.5 Butterworth Filters

The frequency domain image is then filtered through 16 Butterworth filters with each filter tuned to a particular orientation. Each directional filter is independently applied to the image in frequency domain. The number of Butterworth directional filters equals to the number of directions used in orientation estimation step to calculate the orientation image.

2.3.6 Inverse 2DFT

The inverse of 2DFT is then used to convert each image from frequency domain to the spatial domain, producing a set of directionally filtered images called as pre-filtered images.

2.3.7 Reconstruction

The last stage of this algorithm is to reconstruct the final enhanced image from the pre-filtered images by using the ridge orientation value at each pixel in the raw image, and the filtering direction of each pre-filtered image. Each point in the final enhanced image is then computed by selecting from the pre-filtered images the pixel value whose filtering direction most closely matches the actual ridge orientation[6] , [14] , [16].

2.4 Fingerprint Enhancement using Gabor filters

Enhancement using Gabor filter is an enhanced version of the previous enhancement algorithm, because in the 2D Fourier Transform the frequency content was taken to be constant throughout, but in Gabor filter technique the frequency content of the image is taken into account. Steps of Gabor filter algorithm are: Normalization, Segmentation, Orientation Image Estimation, Orientation Smoothing, Frequency Image Estimation, and Filtering. The algorithm of this technique is shown in figure 5 [7].

2.4.1 Normalization

As explained in the previous section 2.3. 1

2.4.2 Segmentation

As explained in the previous section 2.3.2

2.4.3 Orientation Image Estimation

This step is used to represent the local orientation $\theta(i,j)$ of the ridges contained in the fingerprint image. It is a critical process because any errors occurring at this step will be propagated into the frequency estimation and filtering stages.

2.4.4 Orientation Smoothing

Orientation smoothing is very important step because fingerprint images may be have some noises and hence the estimated local ridge orientation $\theta(i,j)$ in the previous step may be incorrect.

2.4.5 Ridge Frequency Estimation

The local ridge frequency is defined as the frequency of the ridge and valley structures in a local neighborhood along a direction orthogonal to the local ridge orientation.

2.4.6 Filtering

In this step the noise in the fingerprint image will be eliminated and preserve the ridge and valley structures.
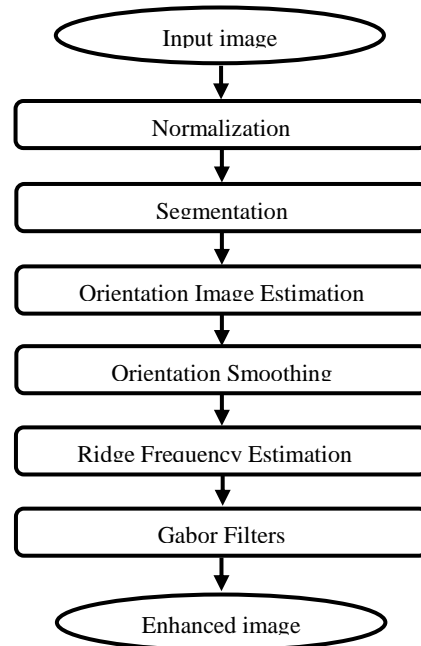

Fig 5: Steps of enhancement using Gabor Filters technique

3. PROPOSED FINGERPRINT IMAGE ENHANCEMENT USING FDCT WITH GABOR FILTERS

Fingerprint images usually have different kinds of noise such as random noise, salt noise, speckle noise and Gaussian noise. Removing noise from fingerprint images is an important problem in image processing [8]. Methods of removing noises are divided into:

1. Space domain field: is data operation carried on the original image, and processes the image gray value, like neighborhood average method, Total Variation (TV) filter, wiener filter and so on.

2. Transform domain filed:

   a. In the time-frequency domain: Fourier transform is used.

   b. In the time-scale domain: multi-resolution transforms like wavelet/curvelet/contourlet transforms are used.

The most effective tool for image denoising is curvelet transform. It has good orientation characteristic due to its structural elements. Curvelet transform is a powerful tool multi-scale representation which is more suited for edges and other singularity curves. The curvelet construction is based on combining the following three components [9]:

1. **Ridgelets**: is a method of image analysis suitable for objects with discontinuities across straight lines

2. **Multi-scale Ridgelets**: a pyramid of windowed ridgelets is used, renormalized and transported to a wide range of scales and locations.

3. **Band pass Filter**: is a method of separating out an image into a series of disjoint scales.

By assembling the above components, Curvelets are based on multi-scale ridgelets combined with a spatial band pass filtering operation to isolate different scales.

There are two implementations that fast discrete curvelet transform (FDCT) based on:

1. Unequally Spaced Fast Fourier Transform (USFFT).

2. Wrapping Algorithm (this implementation is used here).

Wrapping based fast discrete curvelet transform is faster in computation time and more robust, simple, and good edge capturing than USFFT based FDCT, therefore this technique is used in this paper. The following steps are involved in enhancing the fingerprint image using fast discrete curvelet transform over Gabor filters is shown in figure 6 [10, 11].
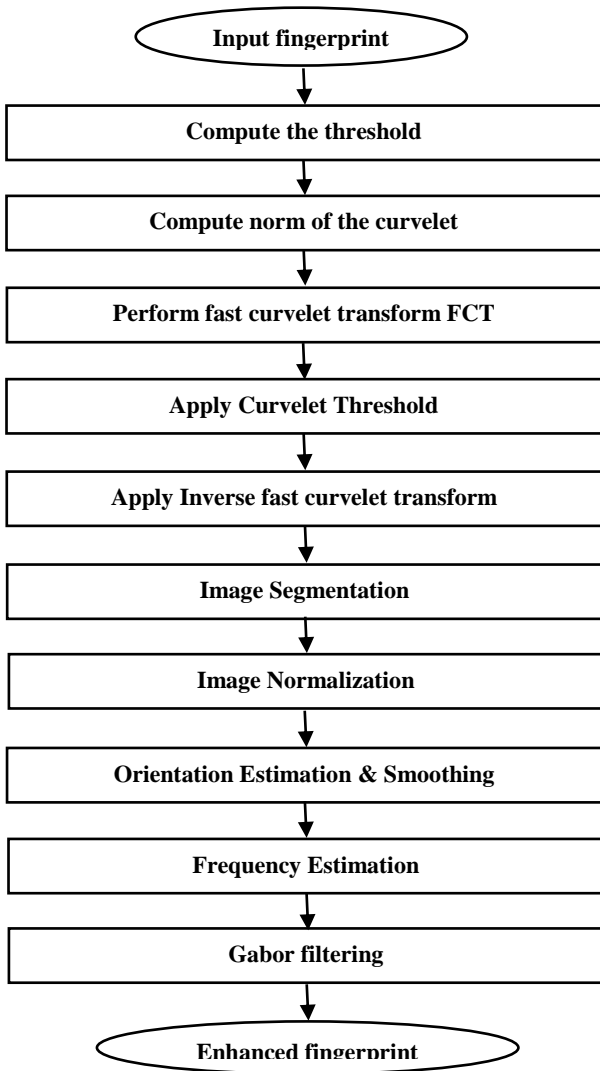
Input fingerprint

↓

Compute the threshold

↓

Compute norm of the curvelet

↓

Perform fast curvelet transform FCT

↓

Apply Curvelet Threshold

↓

Apply Inverse fast curvelet transform

↓

Image Segmentation

↓

Image Normalization

↓

Orientation Estimation & Smoothing

↓

Frequency Estimation

↓

Gabor filtering

↓

Enhanced fingerprint

Fig 6: FDCT over Gabor Filters algorithm

3.1.1 Compute threshold
Compute all thresholds of curvelets which will be applied to the image curvelet coefficient in the fourth step in this algorithm.

3.1.2 Compute norm of the curvelet
The curvelet coefficients are normalized and computed by the following equation [8]:

$$C^{D,N}(j,l,k) = \frac{n}{\sqrt{L_{1,j}L_{2,j}}}C^D(j,l,k) \tag{4}$$

Where $L_{1,j}$, $L_{2,j}$ are the side lengths of the parallelogram.

3.1.3 Perform fast discrete curvelet transform
Perform warping Fast Discrete Curvelet Transform (FDCT) to the noisy image to move it from spatial domain to curvelet domain.

3.1.4 Apply curvelet threshold
Apply the computed threshold in step1 to the curvelet coefficients.

3.1.5 Apply inverse FDCT
Apply inverse FDCT to the result in the previous step (after apply threshold on it) to move image from curvelet domain to spatial domain.

3.1.6 Segmentation
The process of separating the foreground regions (contain fingerprint information) from the background regions (noisy area) in the image. The foreground regions refer to the actual fingerprint area containing the ridges and valleys, which is also called the Region of Interest (ROI). Figure 7 shows the original image and image after segmentation [12]. Steps of segmentation process are as follow:

(a) Divide the input image $I(i,j)$ into blocks of sizen × n (8 × 8).

(b) Compute the mean $M(I)$ using the equation:

$$M(I) = \frac{1}{n^2}\sum_{i=0}^{n-1}\sum_{j=0}^{n-1}I(i,j) \tag{5}$$

(c) Calculate the variance V(I) using the equation:

$$V(I) = \frac{1}{n^2}\sum_{i=0}^{n-1}\sum_{j=0}^{n-1}\{I(i,j) - M(I)\}^2 \tag{6}$$

(d) Calculate the standard deviation std(I) by the equation:

$$std(I) = \sqrt{V(I)} \tag{7}$$

(e) A threshold value (thr) is empirically selected to exclude the background from the fingerprint area (Try a value 0.1 - 0.2).

If $\begin{cases} std(I) > thr & \text{then the block is considered as foreground} \\ std(I) < thr & \text{then the block is considered as background} \end{cases}$
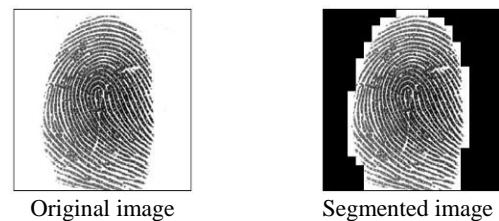

Original image   Segmented image

Fig 7: Image before and after segmentation process

3.1.7 Normalization

The main purpose of normalization is to reduce the differences in the grey level values along the ridges and valleys so that the pixels values in the image have a specified mean and variance. The normalization stage enhances the contrast of the fingerprint image, so that the ridges and valleys of the normalized image are easily distinguished. Let a gray level image (I) be defined as $(m \times n)$ matrix where, $I(i, j)$ represents the intensity of the pixel at the ith row and jth column. The first step is to calculate the mean (M), and the variance (V) of the given image (I), then the equation of the normalized image $N(i, j)$ is defined as:

$$N(i,j) = \begin{cases} M_0 + \sqrt{\dfrac{V_0(I(i,j) - M)^2}{V}} & \text{if } I(i,j) > M \\ \\ M_0 - \sqrt{\dfrac{V_0(I(i,j) - M)^2}{V}} & \text{otherwise} \end{cases} \quad (8)$$

Where $I(i, j)$ is the gray-scale value at pixel(i, j), M_o and V_o are the desired mean and variance respectively, and N is the normalized image. Let $M_0 = 0$ and $V_0 = 1$, so that the new intensities of the pixels for the normalized image would be between -1 and 1. Figure 8 shows the image before and after normalization [12].

(a)   (b)
Fig 8: (a) Original image (b) Image after normalization.

3.1.8 Orientation estimation

The orientation image defines the local orientation of the ridges contained in the fingerprint image. The orientation estimation is a critical process and any errors occurring at this stage will be propagated into the frequency estimation and filtering stages. The orientation image (O) represents the instantaneous ridge orientation at every pixel in the fingerprint image. However local ridge orientation is usually estimated for a block rather than at every pixel, since the ridge orientation varies very slowly across the fingerprint image. Note, there is no difference between a local ridge orientation of 90° and 270°, since the ridges oriented at 90o and 270o in a local neighborhood cannot be distinguished from each other. The steps of estimating the orientation are as follow:

(a) Divide the normalized image into non-overlapping blocks of size $(W \times W)$

(b) Compute the gradient of each pixel in the block by using the Sobel operator which is given in table 1.

Table 1 Horizontal and vertical Sobel operators

Description	Operator
Horizontal Sobel operator is convolved with the normalized image to get the gradients along x-direction $\partial_x(i,j)$.	$\begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}$
Vertical Sobel operator is convolved with the normalized image to get the gradients along y-direction $\partial_y(i,j)$.	$\begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$

(c)   Calculate the local orientation of each block centered at pixel (i, j) using the following equations:

$$V_x(i,j) = \sum_{u=i-\frac{W}{2}}^{i+\frac{W}{2}} \sum_{v=j-\frac{W}{2}}^{j+\frac{W}{2}} 2\partial_x(u,v)\partial_y(u,v) \quad (9)$$

$$V_y(i,j) = \sum_{u=i-\frac{W}{2}}^{i+\frac{W}{2}} \sum_{v=j-\frac{W}{2}}^{j+\frac{W}{2}} \partial_x^2(u,v)\partial_y^2(u,v) \quad (10)$$

$$\theta(i,j) = \frac{1}{2}\tan^{-1}\frac{V_y(i,j)}{V_x(i,j)} \quad (11)$$

Where $V_x(i,j)$, and $V_y(i,j)$ are the moments in x and y direction respectively. $\theta(i,j)$ Is the local ridge orientation at the block centered at pixel (i, j) [12, 13].

3.1.9 Orientation Smoothing

Fingerprint images may be have some noises such as corrupted ridge and valley structures, minutiae, etc. so that the estimated local ridge orientation $\theta(i,j)$ in the previous step may not always be correct. Since local ridge orientation varies slowly in a local neighborhood, a Gaussian low-pass filter can be used to modify the incorrect local ridge orientation, which is also called as orientation smoothing. Figure 9 shows an example of orientation smoothing [12]. The first step in orientation smoothing is to convert the orientation image into a continuous vector field, which is defined as follows:

$$\varphi_x(i,j) = \cos(2\theta(i,j))$$
$$\text{and} \qquad\qquad\qquad\qquad (12)$$
$$\varphi_y(i,j) = \sin(2\theta(i,j))$$

Where φ_x and φ_y are the x and y components of the vector field, respectively. The Gaussian low-pass filter (G) of size $(w_\varphi \times w_\varphi)$ can then be applied as follows:

$$\varphi'_x(i,j) = \sum_{u=-\frac{w_\varphi}{2}}^{\frac{w_\varphi}{2}} \sum_{v=-\frac{w_\varphi}{2}}^{\frac{w_\varphi}{2}} G(u,v)\varphi_x(i - uw, j - vw) \quad (13)$$

$$\varphi'_y(i,j) = \sum_{u=-\frac{w_\varphi}{2}}^{\frac{w_\varphi}{2}} \sum_{v=-\frac{w_\varphi}{2}}^{\frac{w_\varphi}{2}} G(u,v)\varphi_y(i - uw, j - vw) \quad (14)$$

The final smoothed orientation field is computed by using the following equation:

$$O(i,j) = \frac{1}{2}\tan^{-1}\frac{\varphi'_y(i,j)}{\varphi'_x(i,j)} \quad (15)$$

Fig 9: example of orientation smoothing.

3.1.10 Frequency estimation

The local ridge frequency is defined as the frequency of the ridge and valley structures in a local neighborhood along a direction orthogonal to the local ridge orientation. The ridge frequency is also varying slowly and hence it is computed only once for each non-overlapping block of the image. The first step for ridge frequency estimation is to divide the image into non-overlapping blocks of size$(W \times W)$. The second step

is to estimate the projection sum of the grey-level values of all the pixels inside each block along a direction normal to the local ridge orientation. The projection sum forms a sinusoidal shaped signal and the distance between any two peaks provides the inter-ridge distance (see Figure. 10).
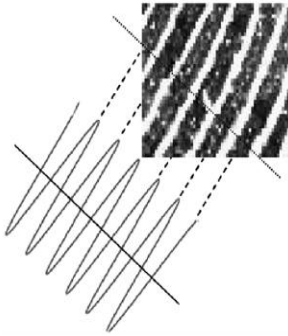
Fig 10: show the pixels intensity values projection along a direction orthogonal to the local ridge orientation.

The steps of estimating the local ridge frequency are as follow:

(a) The normalized image *(G)* is divided into blocks of size $(w \times w)$.

(b) For each block centered at pixel (i, j), compute an oriented window of size $(l \times w)$, which is rotated to align the y-axis of the window with the local ridge orientation.

(c) Compute the X−signature of the gray-levels for each block by using the equation:

$$X[k] = \frac{1}{w} \sum_{p=0}^{w-1} G(u, v), \qquad k = 0,1,2, \dots \dots \dots l - 1 \qquad (16)$$

Where $u \; and \; v$ is the ridge coordinates as follows:

$$u = i + \left(p - \frac{w}{2}\right) \cos\left(O(i,j)\right) + \left(k - \frac{l}{2}\right) \sin(O(i,j)) \qquad (17)$$

$$v = j + \left(p - \frac{w}{2}\right) \sin\left(O(i,j)\right) - \left(k - \frac{l}{2}\right) cos(O(i,j)) \qquad (18)$$

Where $O(i, j)$ is the local ridge orientation at the block centered at pixel (i, j).

(d) Calculate the average number of pixels $T(i, j)$ between two consecutive peaks in the x-signature

(e) The ridge frequency for block centered at (i, j) is:

$$F(i,j) = \frac{1}{T(i,j)} \qquad (19)$$

3.1.11 Gabor filter

The main purpose of this stage is to remove noise and preserve the ridge and valley structures. One of the most important advantages of Gabor filters that they have both orientation-selective and frequency-selective properties. This advantage allows Gabor filters to be tuned to the corresponding local orientation and frequency for each local neighborhood and give optimal response to ridges in the fingerprint image. The necessary parameters of Gabor filter are:

a. Local ridge orientation.

b. The ridge frequency.

c. The standard deviations of the Gaussian envelope $\sigma_x \; and \; \sigma_y$.

The filter is mathematically defined by (20).

$$G(x, y; \theta, f) = e^{-\frac{1}{2}\left[\frac{x_\theta^2}{\sigma_x^2} + \frac{y_\theta^2}{\sigma_y^2}\right]} \cos(2\pi f x_\theta) \qquad (20)$$

$$x_\theta = x\cos\theta + y\sin\theta \quad \& \quad y_\theta = -x\sin\theta + y\cos\theta \qquad (21)$$

Where θ the Gabor filter orientation, f is is the frequency of a sinusoidal plane wave, and $x_\theta \; and \; y_\theta$ define the x and y axes of the filter coordinate frame, respectively. The enhanced image E is then obtained by spatially convolving the image with the filter and is mathematically defined as:

$$E(i,j) = \sum_{u=-\frac{w_x}{2}}^{\frac{w_x}{2}} \sum_{v=-\frac{w_y}{2}}^{\frac{w_y}{2}} G\left(u, v; O(i,j), F(i,j)\right) N(i-u, j-v) \qquad (22)$$

Where O is the orientation image, F is the frequency image, and N is the normalized fingerprint image [13].

4. EXPERIMENTAL RESULTS

The experimental tests of the different methods of fingerprint enhancement techniques are conducted on 500dpi resolution fingerprint images derived from FVC2002 fingerprint database. FVC2002 fingerprint database contains 800 fingerprint images. Fingerprint images in the database were selected from NIST Special Database, FVC database samples, and captured with a live scanner. The implementation of fingerprint enhancement techniques was done using Matlab2010 [14-16].

4.1 Experimental results of enhancement methods on various samples

In order to explain the performance and effectiveness of the proposed approach, Table 2 compare the proposed method against the other enhancement techniques. The 1st column in table 2 shows the poor quality original input images, the 2nd and 3rd column give the output enhanced images using HE and AHE respectively, the 4th column shows the output enhanced images using FFT, the 5th column shows enhanced image using Gabor Filters, and the 6th column shows the enhanced image using the proposed algorithm. As shown in table 2 the output of proposed algorithm give the best fingerprint image enhancement results.

Table 2 Output images from different enhancement methods

	Input image	HE	AHE	FFT	Gabor Filters	FDCT & Gabor
Sample 1						
Sample 2						
Sample 3						
Sample 4						

4.2 Experimental results of enhancement methods on Noisy fingerprint images

This section represents the experimental results when the proposed algorithm is applied to noisy fingerprint images with different level of noise. Table 3 shows the experimental results of applying proposed algorithm to noisy fingerprint images. The first column contain the noise level which inserted to the fingerprint image, the second column contain the finger image after adding noise, the third column represent the enhanced fingerprint image, and the fourth column show the estimated value of PSNR.

Table 3 experimental results with noisy images

Noise level in decibel	Original image with noise	Fingerprint image enhancement using FDCT & Gabor	PSNR
5			62.9618
10			57.3354
15			54.047
20			51.5463
25			49.7315
30			48.1599
35			47.0731
40			45.1021

After applying the traditional enhancement methods on noisy fingerprint images, and evaluate the PSNR value for each method. Figure 11 describe the relation between PSNR & Noise level for each enhancement technique.
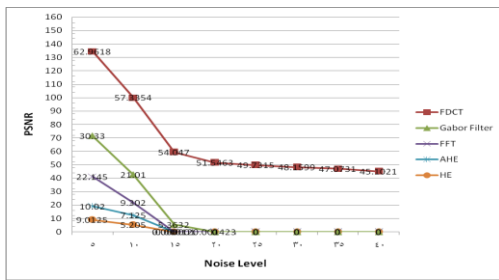
Fig 11: Relation between PSNR and Noise Level

5. SUMMARY, CONCLUSION AND FUTURE WORK

The main purpose of fingerprint image enhancement stage is to increase the clarity, and improve feature detection of the fingerprint image as much as possible, and hence avoiding undesired side effects in the subsequent stages. The performance parameters that are used for the comparative study of the enhancement techniques explained above can be represented by peak signal-to-noise ratio, absolute mean brightness error, and Computation time [5, 17].

5.1 Peak Signal-to-Noise Ratio (PSNR)

PSNR is the ratio between the maximum possible value (power) of the image and the power of distorting noise that affects the quality of its representation. PSNR is usually measured in terms of the logarithmic decibel scale. Let the original gray level image *(I)* be defined as $(m \times n)$ matrix, and *(E)* is the enhanced or restored image. Then the PSNR is mathematically computed by equation (23).

$$PSNR = 10 \log_{10}\left(\frac{L_I^2}{MSE}\right) \tag{23}$$

Where *L* is the maximum possible pixel value of the image (for 8-bit gray level image $L = 2^8 - 1 = 255$), and MSE (mean squared error) is defined by (24)

$$MSE = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}[I(i,j) - E(i,j)]^2 \tag{24}$$

Note: the greater PSNR, the better the output image quality.

5.2 Absolute Mean Brightness Error (AMBE)

AMBE is used to estimate the degree of brightness preservation. AMBE can be expressed by (25).

$$AMBE = |I_m - E_m| \tag{25}$$

Where I_m: input image mean intensity, & E_m: output image mean intensity. The lower AMBE is the better brightness.

5.3 Computation Time (CT)

CT is used to estimate the processing time taken by the fingerprint image enhancement stage. This is done by calculating the central processing unit time of the Matlab commands. After applying the discussed enhancement methods on some fingerprint images, and estimate the average value of PSNR, AMBE, and CT. Table 4 describes a comparison between the enhancement methods with respect to PSNR, AMBE, and CT.

Table 4 Experiment values and comparison of the above discussed enhancement techniques

Technique	PSNR (dB)	AMBE	CT (sec)
HE	31.4288	13.7799	0.25374
AHE	33.5622	10.5738	0.3149
FT	35.2857	10.5664	1.0507
Gabor Filters	42.5877	0.71306	1.7113
FDCT & Gabor	54.7809	0.14017	2.0246

6. SUMMARY, CONCLUSIONS AND FUTURE WORK

There are various techniques used for enhancing a fingerprint image which include HE, AHE, 2DFT, Gabor Filter, and FDCT with Gabor Filter for enhancing the gray scale Fingerprint image. Various performance metrics like AMBE, PSNR, and CT are used to evaluate and estimate each and every fingerprint enhancement technique. Higher PSNR and lower AMBE is considered to be favourable one. After demonstrating the aforementioned techniques, we concluded that the enhancement of an image depends upon the purpose of application, while previous figures, graphs, and tables indicate that FDCT with Gabor filter technique is performing better in terms of AMBE, PSNR and CT since its AMBE has the least value, its PSNR has the highest value, and take an acceptable average computation time. It has been observed that, FDCT & Gabor filters is the best algorithm for fingerprint images which has different kinds of noise such as random noise, salt and Gaussian noise. The future work is to create a fingerprint recognition system which contains both fingerprint identification and verification systems. The purpose of the identification system process is to compare an input fingerprint with other fingerprints stored in database. The fingerprint recognition system depends on the extracted features from fingerprint image. Therefore the above discussed fingerprint enhancement methods is used in fingerprint recognition system in order to improve the clarity of fingerprint image and facilitate the features extraction process.

7. REFERENCES

[1] D. Davide Maltoni, D. Dario Maio, Anil K. Jain, and Salil Prabhakar, "Handbook of Fingerprint Recognition Second Edition," New York: Springer-Verlag, 2009.

[2] http://en.wikipedia.org/wiki/Automated_Fingerprint_Ide ntification_System

[3] Zain S. Barham, Supervised by Dr. Allam Mousa, " Fingerprint Recognition using MATLAB ," Graduation project 5/17/2011

[4] http://en.wikipedia.org/wiki/Histogram_equalization

[5] Ekta Thirani, "A Comparative Analysis of Fingerprint Enhancement Technique through Absolute Mean Brightness Error and Entropy, " IJRITCC, International Journal on Recent and Innovation Trends in Computing and Communication ISSN: 2277- 4804 Volume: 1 Issue: 2 | JAN 2013

[6] http://en.wikipedia.org/wiki/Adaptive_histogram_equaliz ation

[7] M.Manju, and V.Kavitha "Survey on Fingerprint Enhancement Techniques," International Journal of Computer Applications (0975 – 8887) Volume 62– No.4, January 2013.

[8] Aliaa A.A.Youssif, A.A.Darwish, and A.M.M.Madbouly, "Adaptive Algorithm for Image Denoising Based on Curvelet Threshold, " IJCSNS International Journal of Computer Science and Network Security, VOL.10 No.1, January 2010

[9] E. J. Candès and D. L. Donoho, "Curvelets," [Online] Available: http://www-stat.stanford.edu/~donoho/Reports/1999/curvelets.pdf.

[10] E.Cand`es, L.Demanet, D.Donoho and L.Ying, "Fast Discrete Curvelet Transforms", March 2006. http://www.curvelet.org/papers/FDCT.pdf

[11] J.Candes, L.Demanet, D. Donoho and .Ying. http://www.curvelet.org.

[12] Iwasokun Gabriel Babatunde, Akinyokun Oluwole Charles, Alese Boniface Kayode, and Olabode Olatubosun "Fingerprint Image Enhancement: Segmentation to Thinning," (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 3, No. 1, 2012

[13] Somsak Choomchuay, and Keokanlaya Sihalath,"An Application of Second Derivative of Gaussian Filters in Fingerprint Image Enhancement, " 978-1-4244-4713-8/10/ ©2010 IEEE.

[14] http://bias.csr.unibo.it/fvc2002/download.asp

[15] http://www.mathworks.com

[16] http://www.imageprocessingbasics.com/

[17] N. Phanthuna, F. Cheevasuvit and S. Chitwong, "CONTRAST ENHANCEMENT FOR MIMIMUM MEAN BRIGHTNESS ERROR FROM HISTOGRAM PARTITIONING," ASPRS 2009 Annual Conference Baltimore, Maryland, March 9-13, 2009.