Permit to Work (P.T.W) Using Thumb-print Recognition System

by

Tuan Mohd Irwan bin Tuan Daud

Project dissertation submitted in partial fulfilment of
the requirements for the
Bachelor of Engineering (Hons)
(Electrical & Electronics Engineering)

June 2009

Universiti Teknologi PETRONAS
Bandar Seri Iskandar
31750 Tronoh
Perak Darul Ridzuan

CERTIFICATION OF APPROVAL

PERMIT TO WORK (P.T.W) USING THUMB-PRINT RECOGNITION SYSTEM

by

Tuan Mohd Irwan bin Tuan Daud

A project dissertation submitted to the
Electrical & Electronics Engineering Programme
Universiti Teknologi PETRONAS
in partial fulfilment of the requirement for the
Bachelor of Engineering (Hons)
(Electrical & Electronics Engineering)

Approved:

(Ms. Lila Iznita Izhar)
Project Supervisor

UNIVERSITI TEKNOLOGI PETRONAS
TRONOH, PERAK

June 2009

CERTIFICATION OF ORIGINALITY

This is to certify that I am responsible for the work submitted in this project, that the original work is my own except as specified in the references and acknowledgements, and that the original work contained herein have not been undertaken or done by unspecified sources or persons.

Tuan Mohd Irwan bin Tuan Daud

ABSTRACT

Fingerprint is one of the oldest biometric identification that has been researched. The uniqueness of fingerprint minutiae which can differentiate everybody in this world without duplication has made it most suitable for authentication system. Permit to Work (P.T.W) system has been widely used in petroleum or plant related industry which expose to highly dangerous activity and hazards. By taking the two systems into consideration, the purpose of this project is to integrate the fingerprint authentication in P.T.W approval system. The challenge in this project is to understand the basic knowledge of fingerprint and do some analysis to enhance the image and matching process. Since the fingerprint images are rarely of perfect quality and they may be degraded or corrupted due to variations in skin and impression conditions, the enhancement process for fingerprint is essential. There are three enhancement method have been carry out on the fingerprint image and the result are being monitored. The three methods are adaptive histogram equalization, filtering using Gabor Filter and filtering using Ridge Filter. The purpose of the enhancement analysis is to get the better image to be used for authentication system later. Analysis of fingerprint image is then carried out by using minutiae extraction. The purpose of minutiae extraction is to detect the minutiae features which are termination and bifurcation. From these three enhancement methods, we can see that the best method which produces better output is by using Ridge Filter. The application to use this system has been created and approval processes have been tested.

ACKNOWLEDGEMENTS

I would like to take the opportunity to express my utmost gratitude to the individuals who have taken the time and effort to assist me in completing this project. Without the cooperation of these individuals, I would undoubtedly have faced complications throughout the project.

First and foremost my utmost gratitude goes to my supervisor, Mrs. Lila Iznita Izhar. Without her guidance and patience, I would not be succeeded to complete this project. To all the technicians in Electrical Engineering Departments, thank you for being ever helpful in providing assistance and giving constructive criticisms to help improve the project. In this chance also, I would like to take this opportunity to express my deepest gratitude to all parties involved in conducting this project, ranging from Universiti Teknologi PETRONAS lecturers, and graduate assistants to external establishments who have put in a large effort in turning this project into a reality.

I also would like to dedicate this project to family and friends for giving their tireless support and continuous motivation throughout this last one year in completing this project.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1
INTRODUCTION

1.1 Background of Study

The possibilities of serious accidents are very high in petroleum industry because they are holding large quantities of toxic and flammable materials. When accidents occur, human factors, such as failure to implement procedures are often the cause. To avoid failures, a system known as the Permit to Work (P.T.W) should be in place.

A P.T.W is an essential part of a system which determines how that job can be carried out safely and it is not simply permission to carry out a dangerous job. We should understand that the permit should not be regarded as a statement that all hazards and risks have been eliminated from the work area. The issue of a permit does not make a job safe and we need to understand that the safety at workplace can only be achieved by those preparing for the work and those carrying it out [6].

The P.T.W system should ensure that suitable precautions by authorized and properly trained people about probable risks are avoided. For those who carrying out the job, they should think about and understand what they are doing and how their work may interface with others. They must also take the necessary precautions which they have been trained to take and for which they have been made responsible [6].

Fingerprints are one of those bizarre twists of nature. Human beings happen to have built-in, easily accessible identity cards which called fingerprints. It has a unique design, which represents everybody alone, with no duplication literally at our fingertips.

Everybody has different pattern of fingerprints even they are identical twins and this ridges structure will not change for the rest of our life. Injuries such as burns or scrapes will not change the ridge structure and when new skin grows in, the same pattern will come back. People have tiny ridges of skin on their fingers because this particular adaptation was extremely advantageous to the ancestors of the human species. The pattern of ridges and "valleys" on fingers make it easier for the hands to grip things, in the same way a rubber tread pattern helps a tire grip the road.

A fingerprint is made of a series of ridges and furrows on the surface of the finger. The uniqueness of a fingerprint can be determined by the pattern of ridges and furrows as well as the minutiae points. Minutiae points are local ridge characteristics that occur at either a ridge bifurcation or a ridge ending [2].

1.2 Problem Statement

Before any work subject to a permit is allowed to commence, certain signatures will be required. The number and designation of the signatories will be determined by the type of permit and the nature of the work to be undertaken. This should be specified within the P.T.W. system. As a minimum, the permit issuer and the person in charge of the work should sign the permit. Other personnel involved in the permit preparation, such as gas tester, should also sign the permit. The person who needs to be aware of the permit or of aspects of the particular task may also be required to sign. Where a transfer of responsibilities takes place example a new Supervisor assumes responsibility for the permit or for the work, provision should be made for this person to sign the valid permit.

Even though fingerprint recognition can differentiate everybody in this world alone, but it have several flaws that we need to overcome to make sure the system is applicable and to reduce the error. The wet or dry fingers are the major factor that may affect the fingerprint recognition process. When these situations occur, the fingerprint detected by the device may not be perfect and degrade. A poor quality of fingerprint acquired will have the problems such as noisy, low in contrast, smudgy or broken will cause spurious ad missing minutiae. This degradation may be caused by the cuts, creases, or cruises on fingertips and also damaged or unclean fingerprint reader.

Based on my experience during internship at one of the petrochemical company at Kertih, there are two (2) signature required for every P.T.W before the workers can do their work. The frequent problem occurs for this P.T.W system is during acquiring the signature process from the person in charge for the related work. This problem leads to delaying the work progress because the work couldn't be start without the approval.

1.3 Objectives and Scope of Study

The main objective of this project is:

- To improve the P.T.W. approval system by developing a software integrating the thumb-print recognition intelligence

The approaches that will be used in order to achieve the objective are.

- To extract thumbprint features i.e. minutiae, ridge
- To perform thumbprint recognition
- To integrate the recognition system with P.T.W. approval system

The scope of work for this project is divided into two (2) parts which are the analysis part and the software part. For analysis part, MATLAB is used to perform simulation on thumb-print image. These analyses are used to determine methods to extract features from thumb-print to be integrated with the software part. We also need to study all the required techniques for the thumb-print recognition to compare several features of print pattern. For the analysis part, we should study on how to enhance the thumb-print image in order to improve the clarity of ridge and furrow structures of input fingerprint images.

For software part, the Microsoft® Visual Basic is used to design and develop the Graphical User Interface (GUI) for the approval process. This GUI will contain the function for the user to create new P.T.W, submit the permit for approval and function to approve the permit. This GUI will contain the approval part which only can be access by the authorized person and will use the thumb-recognition during login process. The other function that will also include in the GUI is file searching task for the user to search existing record of the P.T.W.

This project also needs to use the hardware or device to capture the image of the thumb-print for processing. The selection of the device as well need to be consider in order to make sure that the GUI and the device can integrate together for this project.

CHAPTER 2
LITERATURE REVIEW

2.1 Theory

The first part of the project will focus on the analysis for the thumb-print. Before starting the analysis part, we first need to understand the basic knowledge of the thumb-print. Fingerprints are the patterns formed on the inside and the tips of fingers. The ridges of skin, also known as friction ridges, together with the valleys between them form unique patterns on the fingers [1]. The analysis of fingerprint is a biometric technique comparing scanned image of prints with a database of fingerprints. The fact that they do not change during a person's life and the uniqueness of prints, form the basis for fingerprint analysis.

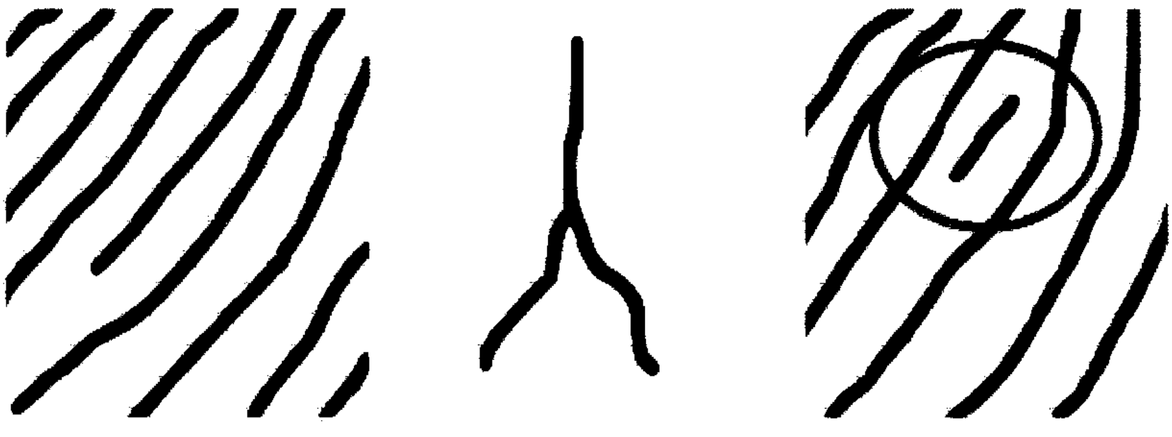
Existence of pores on the surface of the ridges of the fingers results in the accumulation of perspiration on the fingertips. This moisture will remains on the surface of the object a person touches and leaving prints. Depending on the surface touched, prints can be visible to the naked eye (e.g. metal, glass or plastic) or invisible (paper, cardboard or timber). Prints left on non-porous surfaces such as metal can be visualized with powders and lifted with tape. In contrast, the prints on porous objects require special lighting, such as lasers or x rays [1].

There are two major methods of the identification of fingerprints. The first one is comparison of lifted prints and the second method is live scanning. The first method is mainly used in forensics by collecting the fingerprints at a crime scene, or on items of evidence from a crime. This method can be used to identify suspects, victims and other persons who touched a surface leaving the prints.

The second method is used for authentication purposes. This method is widely used in security applications such as to gain access to a building or areas within the building, or computers and network access. Some companies, police offices, and high-security government buildings require fingerprint identification for access to the building or its selected parts. In this project, the second method is used and the fingerprints are collected by the fingerprint sensor.

Ridges present on the fingers are classified based on the patterns they form. The most important features are ridge endings and bifurcations (separation of a ridge into two). The fingerprints patters were shown by Figure 1. These features are called minutiae and form the basis for further classification and identification. Based on the forms created by the minutiae, fingerprints are further sub-classified into many more distinct patterns [2].

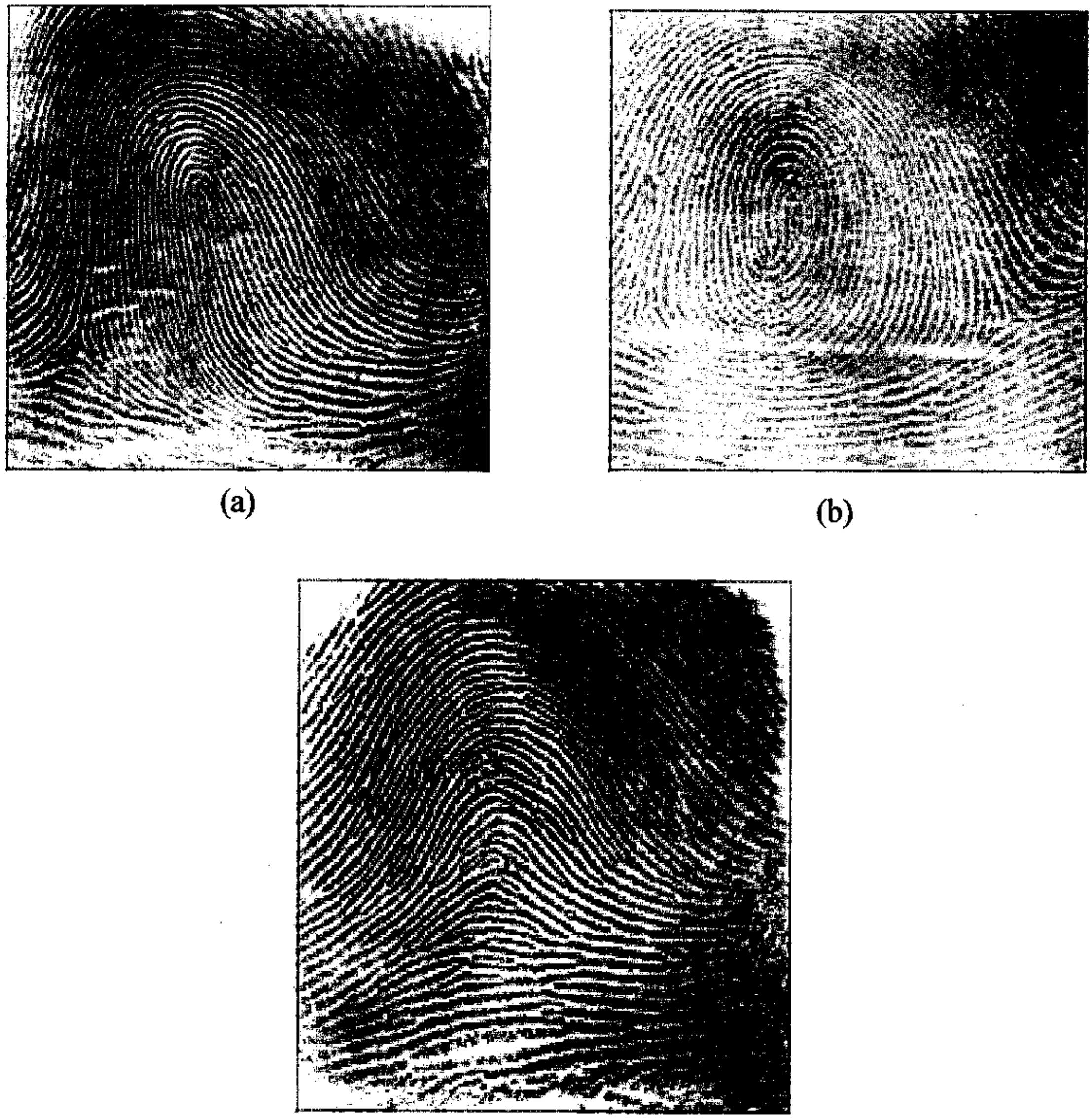
The ridge ending is the point at which a ridge terminates. Bifurcations are points at which a single ridge splits into two ridges. Short ridges (or dots) are ridges which are significantly shorter than the average ridge length on the fingerprint. Minutiae and patterns are very important in the analysis of fingerprints since no two fingers have been shown to be identical [7].



(a) (b) (c)

Figure 1: (a) Ridge Ending; (b) Ridge Bifurcation; (c) Short Ridge

The three basic patterns of fingerprint ridges are the loop, whorl and arch. The loop is a pattern where the ridges enter from one side of a finger, form a curve, and tend to exit from the same side they enter as shown in Figure 2(a). In the whorl pattern, ridges form circularly around a central point on the finger (Figure 2(b)). An arch is a pattern where the ridges enter from one side of the finger, rise in the center forming an arc, and then exit the other side of the finger shown by Figure 2(c). Scientists have found that family members often share the same general fingerprint patterns, leading to the belief that these patterns are inherited.



(a) (b) (c)

Figure 2: (a) Loop Pattern; (b) Whorl Pattern; (c) Arch Pattern

Fingerprint image acquisition is considered the most critical step of an automated fingerprint authentication system, as it determines the final fingerprint image quality, which has drastic effects on the overall system performance. In order to get the better image for processing, the image needs to be enhanced. Since the fingerprint images acquired from sensors or other media are not assured with perfect quality, those enhancement methods, for increasing the contrast between ridges and furrows and for connecting the false broken points of ridges due to insufficient amount of ink, are very useful for keep a higher accuracy to fingerprint recognition. The image enhancement process will be discussed in methodology part.

2.2 Fingerprint Reader Accuracy

The accuracy of fingerprint readers are depends on their False Acceptance Rate (FAR) and False Rejection Rate (FRR). FAR is the measure of the likelihood that the biometric security system will incorrectly accept an access attempt by an unauthorized user. A system's FAR typically is stated as the ratio of the number of false acceptances divided by the number of identification attempts. FAR claimed for today's biometric access systems range from 0.0001% to 0.1%. FRR is the measure of the likelihood that the biometric security system will incorrectly reject an access attempt by an authorized user. A system's FRR typically is stated as the ratio of the number of false rejections divided by the number of identification attempts. The False Reject Rates quoted for current biometric systems range from 0.00066% to 1.0%.

There are several factors that may affect the accuracy of the fingerprint reader. The factors are:

1. **Wet fingers**

   Especially capacitive sensors may have problems with wet fingers. Wet fingers often occur with young users, in warm environment or with excited users. The situation with wet fingers may partly be defused by the user (drying fingers).

2. **Dry fingers**

   Capacitive sensors may have problems with dry fingers. Dry fingers often happen with elder users. The situation with dry fingers may partly be defused by the user (increase pressure and wait longer).

3. **Minutia scarcity**

   Some users may have too few minutiae to be detected depending on sensor area. To reduce this type of error, we need to increase the sensor area.

4. **Skin disease**

   Some kind of skin disease may destroy or disturb the natural finger structure.

5. **Skin abrasion**

   Depending on sensor type, many handicraft activities may decrease the ridge heights such that many sensors deliver only small-contrast pictures. This effect is reversible.

6. **Wrong finger pressure**

   If the pressure on the sensor is too high, the image quality may degrade. If the pressure is non-uniform and non-vertical, warping may prevent proper recognition.

7. **Wrong finger positioning**

   Usually rotation and translation is limited (this limitations may be adjusted by software) because of limited sensor area and because of fake protection. Sometimes a finger guide is not accepted by a user because it may be insufficient or too difficult to be understood.

8. **Finger contamination**

   Contaminated or even dirty fingers degrade image quality of fingerprints. For different sensor types the type of harmful contamination may be different. Frequent contaminations come from skin care substances such as cream.

9. **Sensor contamination**

   Some types of fingerprint sensors are sensitive to sensor surface contaminations, e.g. skin cream. Surface contaminations superimpose nonlinearly with actual fingerprints and may prevent recognition. Moving the finger slightly on the sensor surface or cleaning the sensor may solve the problem.

2.3 Fingerprint Recognition Enhancement Method

There are many previous researches in enhancement process for the fingerprint image. Sherlock, Monroe and Millard [5] proposed a fingerprint enhancement method in the Fourier domain. For this approach, a fingerprint image is convolved with the pre-computed filters and it will results in a set of filtered images. Then, the enhanced fingerprint image is constructed by selecting each pixel from the filtered image whose orientation is the closest to that of the original pixel.

FFT based fingerprint enhancement method are proposed by Willis and Myers [8]. For this technique, enhancement is achieved by multiplying the Fourier transform of the block by magnitude of power, k instead of explicitly computing the local ridge direction and frequency. Chikkerur [12] have proposed an algorithm based on Short Time Fourier Transform (STFT), and probabilistic approximation of dominant ridge orientation and frequency was used instead of the maximum response of the Fourier spectrum. While performing STFT analysis, the ridge orientation image, ridge frequency image, and foreground region image are generated simultaneously.

CHAPTER 3
METHODOLOGY

3.1 Overall Flow of the Project (Flow Chart of Project)

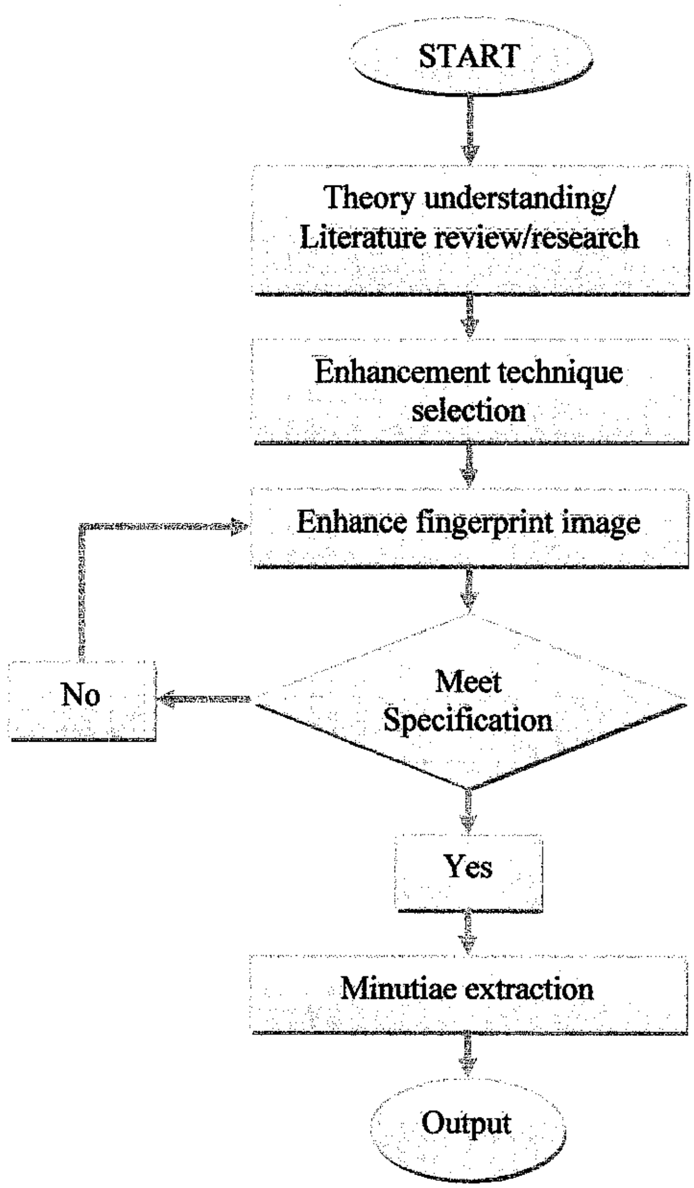


Figure 3: Overall Flow of the Project

3.2 Enhancement Process Using Ridge Filter

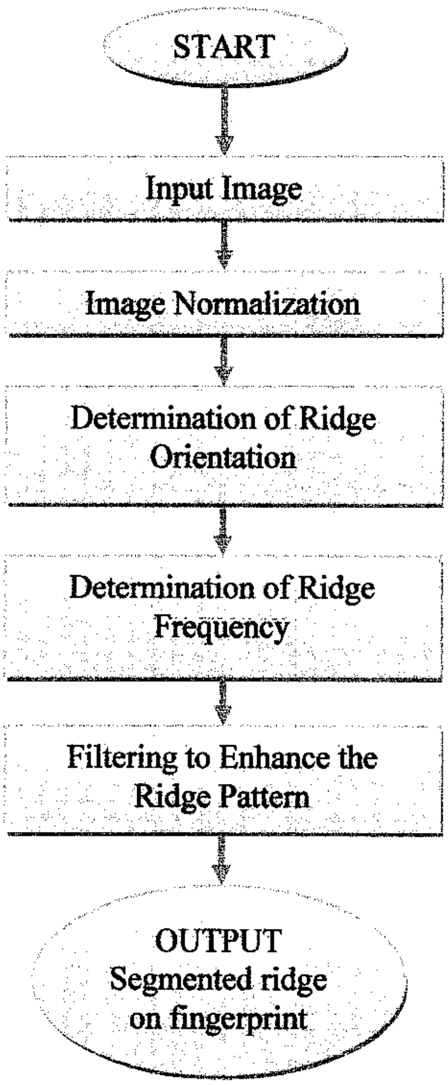


Figure 4: Enhancement Process Using Ridge Filter

3.3 Graphical User Interface (GUI) Flowchart

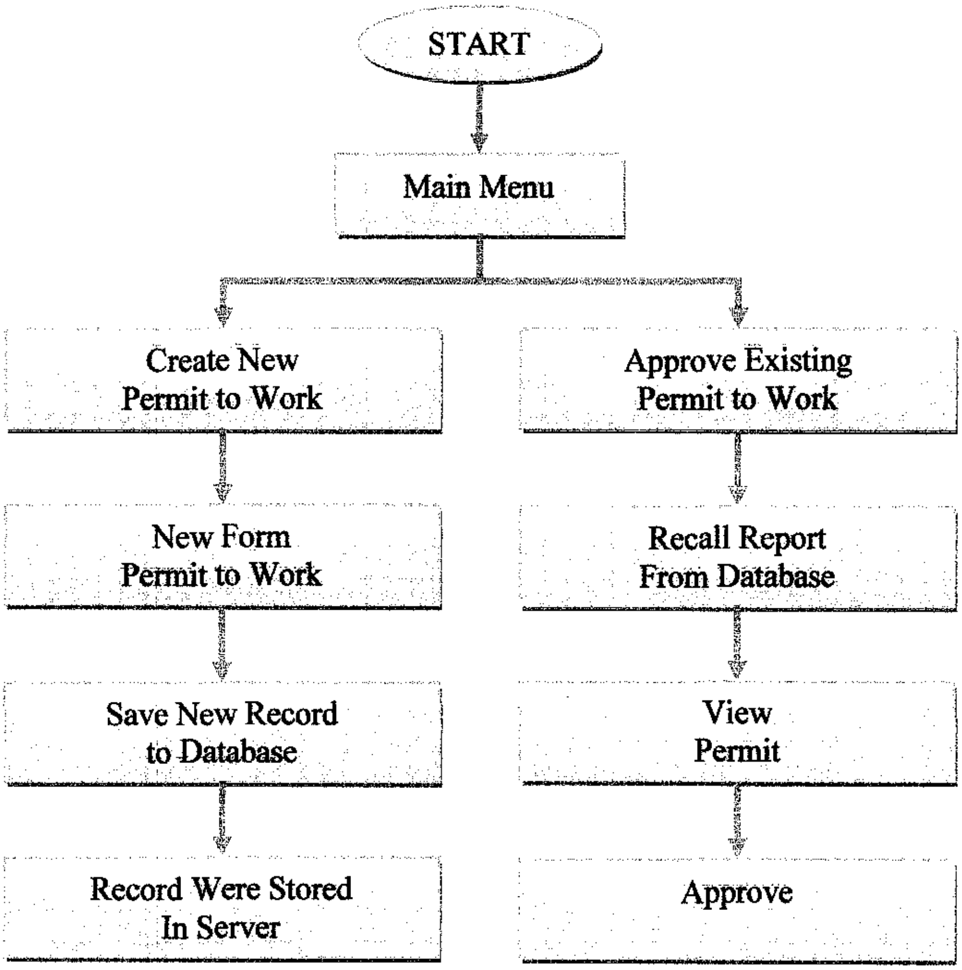


Figure 5: Graphical User Interface Flowchart

From the Main Menu, user can choose two major tasks which are create new PTW or approve existing PTW. If user chooses to create new PTW, the new form of PTW will appear and user can fill up required information of PTW. Then user can choose whether to save or discard the record by clicking the Save Button or Cancel Button. When user chooses the Save Button, it will save the record into the database which is in Microsoft ® Access file format. When user chooses Cancel Button, the program will prompt the confirmation message to user before showing Main Menu again. To approve the existing records of PTW, user can choose Approve Button at Main Menu and it will guide the user through the process.

3.3 Approval Process Flowchart

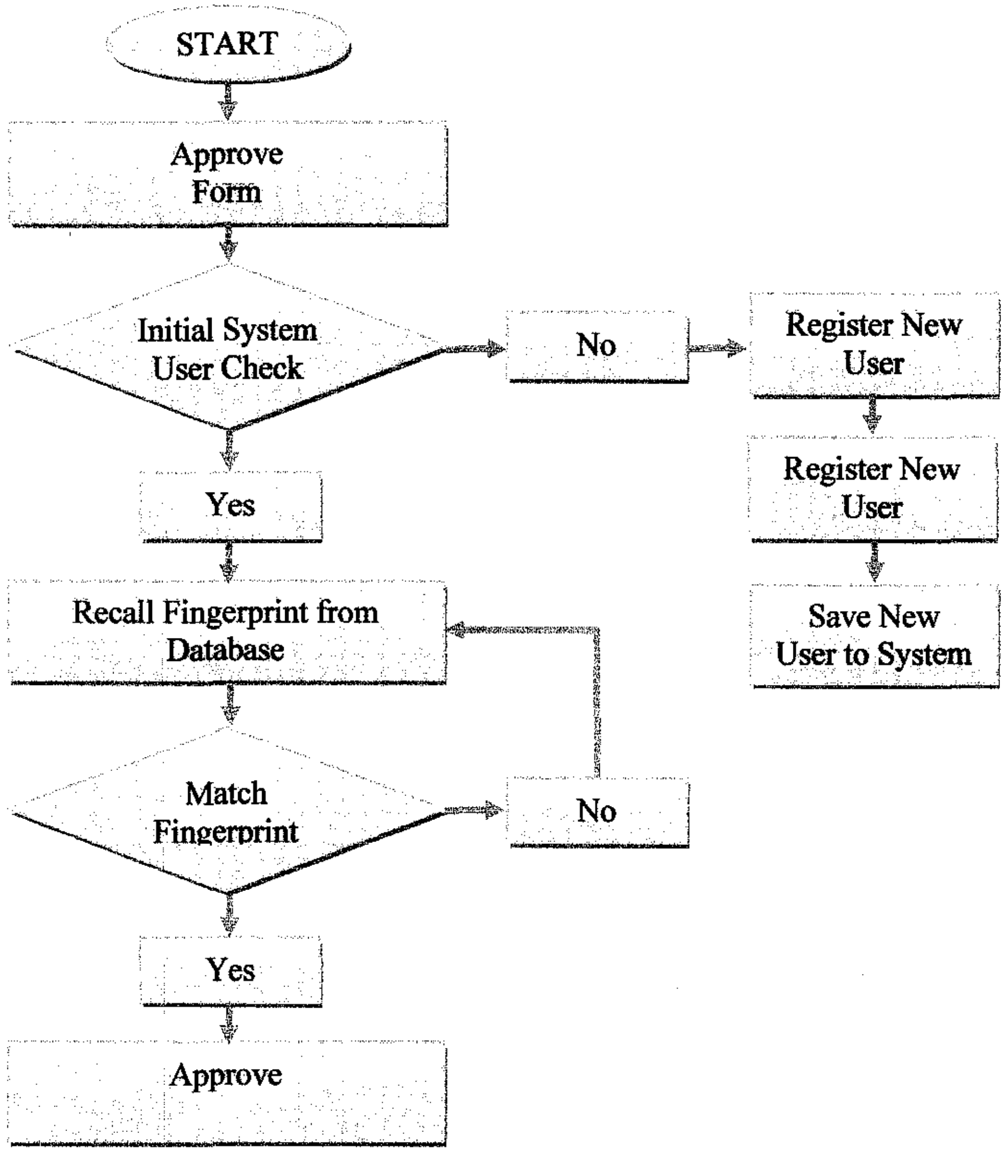


Figure 6: Approval Process Flowchart

The approval process starts exactly when user clicks Approve Button on Main Menu. When the Approve form load, it will check the initial user in database. It is because, without record of authorized person in the system, the approval could not be done. During approval process, the program will recall fingerprint from database and match with the user fingerprint. The PTW only can be approved when the user fingerprint and the fingerprint from database are identical. To perform this action the program will match the minutiae of the fingerprint.

3.4 Tools

A fingerprint image enhancement algorithm receives an input fingerprint image, applies a set of intermediate steps on the input image and finally outputs the enhanced image. In order to introduce our fingerprint image enhancement algorithm, we are using the MATLAB base analysis. The flow of the enhancement process is shown by the flowchart provided.

Several methods have been used for the enhancement process and some of them are the build in function in MATLAB. The methods that were used in fingerprint enhancement process are:

- Adaptive histogram equalization

  The *adapthisteq* function performs contrast-limited adaptive histogram equalization (CLAHE). This function uses a contrast-enhancement method that works significantly better than regular histogram equalization for most images. Adaptive histogram equalization operates on small regions in the image, called tiles. Each tile's contrast is enhanced, so that the histogram of the output region approximately matches a specified histogram. After performing the equalization, adaptive histogram equalization combines neighboring tiles using bilinear interpolation to eliminate artificially induced boundaries.

- Filtering using Gabor Filter

  The Gabor filter is basically a Gaussian (with variances sx and sy along x and y-axes respectively) modulated by a complex sinusoid (with centre frequencies U and V along x and y-axes respectively). The purpose of using Gabor Filter on fingerprint image is because the filters are able to detect bars and lines in image. The visualization of the Gabor filter is shown in the Figure 7.

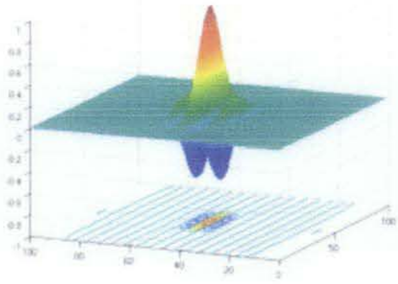


Figure 7: Gabor Filter Visualization

- Filtering using Ridge Filter

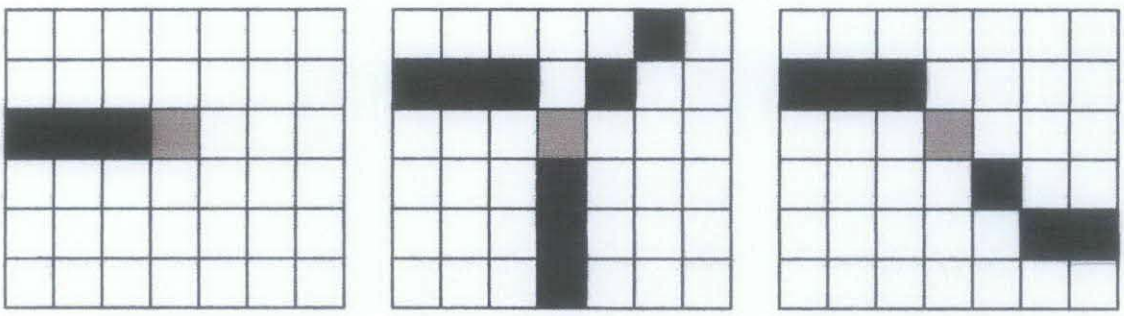
  The function of the ridge filter is to enhance fingerprint image via oriented filters. For this analysis, the ridge regions in the image are identified and normalized, ridge orientations are determined, local ridge frequencies calculated, and then contextual filters with the appropriate orientation and frequency are applied

The next step for fingerprint analysis is the minutiae extraction. Before we can extract the minutiae, several step need to be completed.

1. Image binarization
2. Image thinning
3. Filtering to detect termination and bifurcation

In this analysis we consider three (3) conditions to detect whether it is termination or bifurcation.

- First Condition: If the central is 1 and has only 1 one-value neighbor, then the central pixel is a termination
- Second Condition: If the central is 1 and has 3 one-value neighbor, then the central pixel is a bifurcation
- Third Condition: If the central is 1 and has 2 one-value neighbor, then the central pixel is a usual pixel



(a) (b) (c)

Figure 8: Minutiae Extraction (a) Termination (b) Bifurcation (c) Usual Pixel

All the fingerprint images that are used in the analysis were obtained from the internet. All result and discussion regarding the project work were explained in the Result and Discussion part.

CHAPTER 4
RESULT AND DISCUSSION

4.1 Adaptive histogram equalization (*adapthisteq*)

Adaptive histogram equalization is one of the build-in image processing or enhancement method in MATLAB. The *adapthisteq* function performs contrast-limited adaptive histogram equalization (CLAHE). This function uses a contrast-enhancement method that works significantly better than regular histogram equalization for most images. Adaptive histogram equalization operates on small regions in the image, called tiles. Each tile's contrast is enhanced, so that the histogram of the output region approximately matches a specified histogram. After performing the equalization, adapthisteq combines neighboring tiles using bilinear interpolation to eliminate artificially induced boundaries.

The results using original function with default MATLAB variable for adaptive histogram equalization shown in Figure 9 (b)

(a) (b)

Figure 9: (a) Original Image; (b) Equalized Image

The result that we get from the first syntax of adapthisteq doesn't show major differences between original and equalized image, hence we try to change the string for Distribution. The distribution specifies the desired histogram shape for the image tiles. There are three type of distribution which are:

- *Rayleigh* - Flat histogram
- *Uniform* - Bell-shaped histogram
- *Exponential* - Curved histogram

The result for *rayleigh* distribution compared with the original image shown by the Figure 10 below.

(a) (b)

Figure 10: (a) Original Image; (b) Equalized Image

The result for *uniform* distribution compared with the original image shown by the Figure 11 below.

(a) (b)

Figure 11: (a) Original Image; (b) Equalized Image

Next we try *exponential* distribution and the output result compared with the original image shown by the Figure 12.

(a) (b)

Figure 12: (a) Original Image; (b) Equalized Image

After testing all shape of distribution, we can see that the most suitable distribution is uniform. While using uniform, the ridge can be seen clearly and it also amplifies the ridges that were not so clear before.

Next we try to change the value of ClipLimit over the NumTiles. The function of ClipLimit is to specify a contrast enhancement limit for the image while the NumTiles specify the number of tiles by row and column. We need to adjust the value of ClipLimit to find suitable value because lower ClipLimit will result in insufficient detail while over ClipLimit will amplify unwanted artifacts.

When comparing the result, the effect that we get is more satisfying when the value of ClipLimit is set to 0.05 and the value of NumTiles equal to 8. At this stage, the ridge can be seen clearly compared to others. As we increase the number of tiles, the output result shows finer ridge. The summary of the result comparing ClipLimit over the NumTiles shows by the Table 1.

Table 1: Comparison Clip Limit over Number of Tiles

CLIPLIMIT	NUMBER OF TILES	
	2	4
0.05		
0.10		
0.15		

CLIPLIMIT	NUMBER OF TILES	
	6	8
0.05		
0.10		
0.15		

4.2 Filtering using Gabor Filter

The Gabor filter is basically a Gaussian (with variances sx and sy along x and y-axes respectively) modulated by a complex sinusoid (with centre frequencies U and V along x and y-axes respectively). The purpose of using Gabor Filter on fingerprint image is because the filters are able to detect bars and lines in image.

For analysis purpose we try to vary the value for Sx and Sy parameter over U and V parameter. As we reduce the value of Sx and Sy, it will results in reducing the background intensity. The most suitable value for Sx and Sy in order for us to see the ridge clearly is when Sx=Sy=0.35. U and V parameter does not affect the differences while we vary the value although we have tried large range. The syntax to perform the tasks are as follows.

The summary of the result comparing Sx and Sy parameter over U and V parameters shows by the Table 2.

Table 2: Comparison *Sx* and *Sy* Parameter Over *U* and *V* Parameter

Sx and Sy Parameter	U and V Parameter	
	U = V = 0.50	U = V = 50.0
0.50		
0.35		
0.25		

4.3 Filtering using Ridge Filter

The function of the ridge filter is to enhance fingerprint image via oriented filters. For this analysis, the ridge regions in the image are identified and normalized, ridge orientations are determined, local ridge frequencies calculated, and then contextual filters with the appropriate orientation and frequency are applied. [4] Before we can apply the filter to the image, several step must be followed:

1. normalize image
2. determine ridge orientation
3. determine ridge frequency values across the image

The filtering process by using ridge filter starts by load the input image. The input image for ridge filter shown in Figure 13

Figure 13: Original Image

The next step is normalize the fingerprint image. To normalize image, we must set the block size and threshold value. The result that we get for normalized image shown in Figure 14.

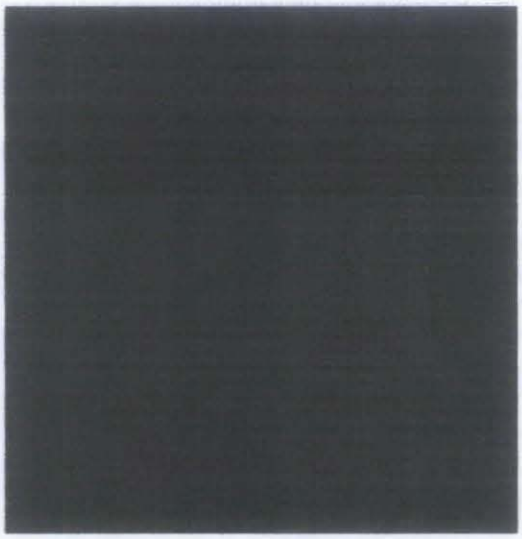


Figure 14: Normalize Image

Next we define the fingerprint ridge orient. The ridge orient shows the orientation for the fingerprint ridge. The orientation will be plotted on the original image and is shown by Figure 15.

Figure 15: Fingerprint Ridge Orient

Then we need to define the fingerprint frequency. The medium frequency will be used in the next step. Figure 16 shows the result for fingerprint ridge frequency.

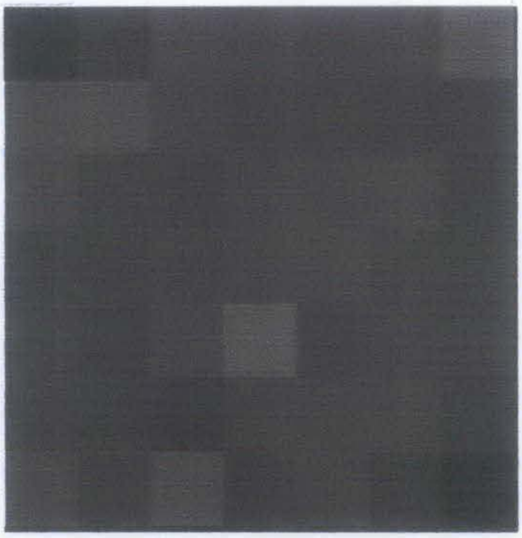


Figure 16: Fingerprint Ridge Frequency

The final step for this process is applying the filter to the image. The previous result which is normalized image, orientation image and medium frequency will be used. The filtered image is shown by Figure 17.

Figure 17: Filtered Image

From the output filtered image, we can see that the white region and the black region are separated entirely. The white region is actually represents the ridge and the black region represents furrows.

4.4 Minutiae Extraction

In order to use the minutiae extraction method, we first need to binarize the image. For binary images, there are only two levels of interest which are the black pixels that represent ridges, and the white pixels that represent valleys. 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.

The binarization process involves examining the grey-level value of each pixel in the enhanced image, and, if the value is greater than the global threshold (T), then the pixel value is set to a binary value one (1) otherwise, it is set to zero (0). The outcome is a binary image containing two levels of information, the foreground ridges and the background valleys.

$$g(x,y) \begin{cases} 1 , & \text{If } \mathbf{f(x,y)} \geq \mathbf{T} \\ 0 , & \text{If } \mathbf{f(x,y)} < \mathbf{T} \end{cases}$$

The next step for minutiae extraction is the thinning process. The purpose of thinning process is to eliminate the redundant pixels of ridges till the ridges are just one pixel wide. It is important to detect whether it is bifurcation or termination.

After we thinned the ridge, then we need to filter the ridge in order to detect whether it is termination or bifurcation. For this task, we need to perform general sliding-neighborhood operations.

We start the minutiae extraction by load the input image. Figure 18 below shows the input image to be used in minutiae extraction process.

Figure 18: Original Image

Then we binarize the image. The output from binarization process is the picture with only white and black pixel. The ridge is shown by black pixel while furrows shown by white pixel. The binarize image shown by Figure 19 below.

Figure 19: Binarize Image

After the image has been binarized, then we need to thin the ridge of the fingerprint image. The purpose of thinning the image is to make sure that the ridge is only one pixel wide to be used while detecting the minutiae later. Thinned fingerprint image shown by Figure 20 below.

Figure 20: Thinned Image

The last step in minutiae extraction is to find the termination and bifurcation by filtering the thinned image. To define termination, we find the pixel with only one neighbor and for bifurcation we find the pixel with three neighbors. The termination will be mark with the red circle while bifurcation will be marked with green circle. The minutiae with termination and bifurcation are shown in Figure 21.

Figure 21: Minutiae Mark with Termination and Bifurcation

The result that we get contains too many mark and some of them are spurious minutiae. To make our analysis simpler and reduce error, we need to remove these spurious minutiae. To remove minutiae, we consider the Distance, D where it is the distance between the pixels of thinned image. We will use three (3) conditions to process the minutiae are as follows:

1. if the distance between a termination and a bifurcation is smaller than D, we remove this minutiae
2. if the distance between two bifurcations is smaller than D, we remove this minutia
3. if the distance between two terminations is smaller than D, we remove this minutia

We have tested and vary the value of distance, D for termination and bifurcation. The results for the analysis are shown in Table 3.

Table 3: Comparison Minutiae Detected Using Various Distance, D

(a) D = 6

(b) D = 8

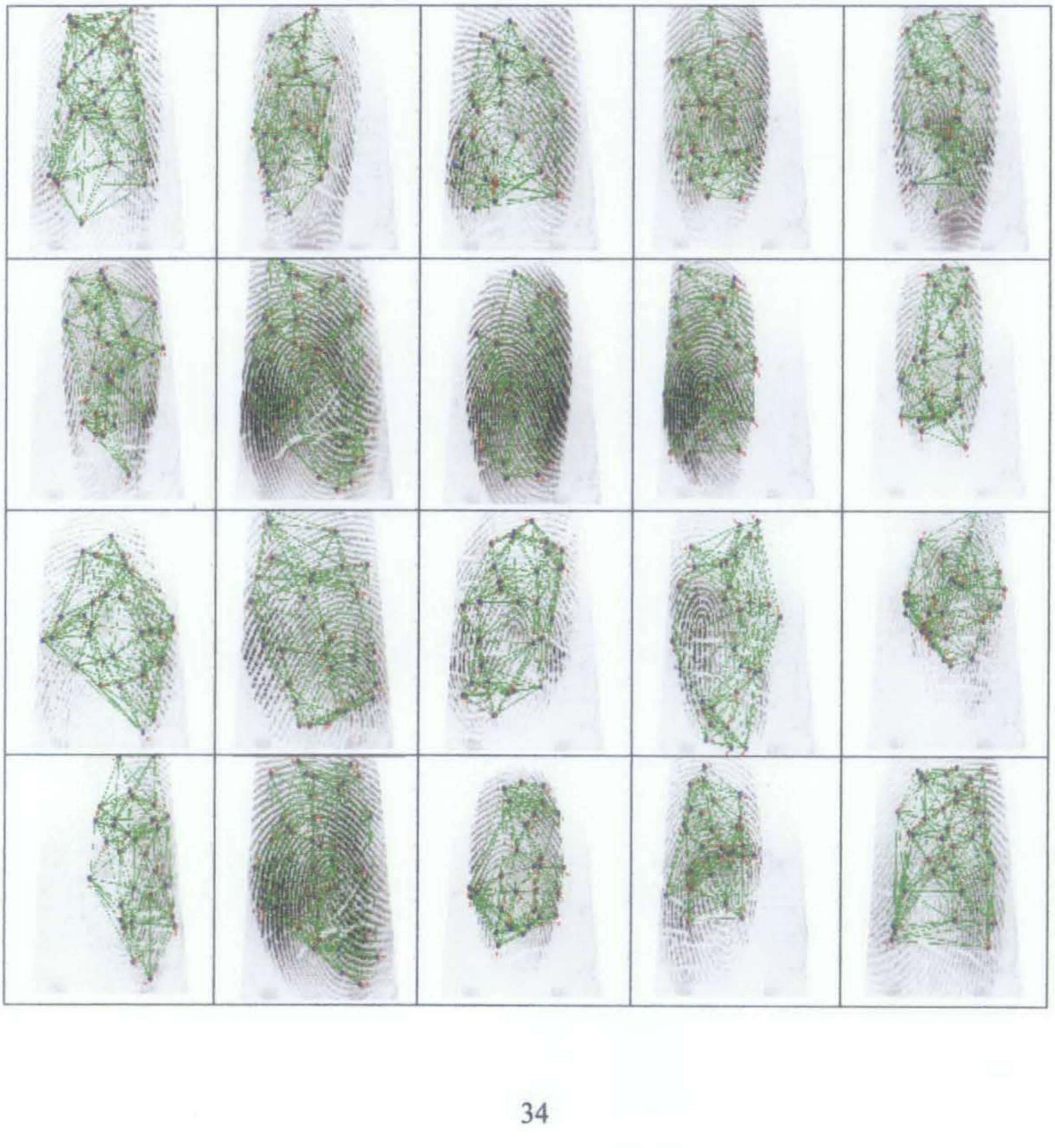
(c) D = 10

(d) D = 12

In order to make sure that this system is applicable in detecting the fingerprints with high accuracy, we have tested it with 20 samples where every samples represented by one person. It is found that this recognition system is able to recognize all the 20 fingerprint accurately. Hence it can be said that the system achieve a 100% accuracy at this stage. The result for the fingerprint recognition shows by Table 4.

During fingerprint registration process for this system, users need to put their fingerprint three times. This is to increase the accuracy of the fingerprint recognition system.

Table 4: Fingerprint Recognition



4.5 Graphical User Interface (GUI)

The GUI of the program was constructs using Microsoft® Visual Basic 2008 Express Edition. On the Main Menu of the program, there will be four buttons which are Create New PTW, Search PTW Records, Approve PTW and User Management. The visual of the main menu are like below.

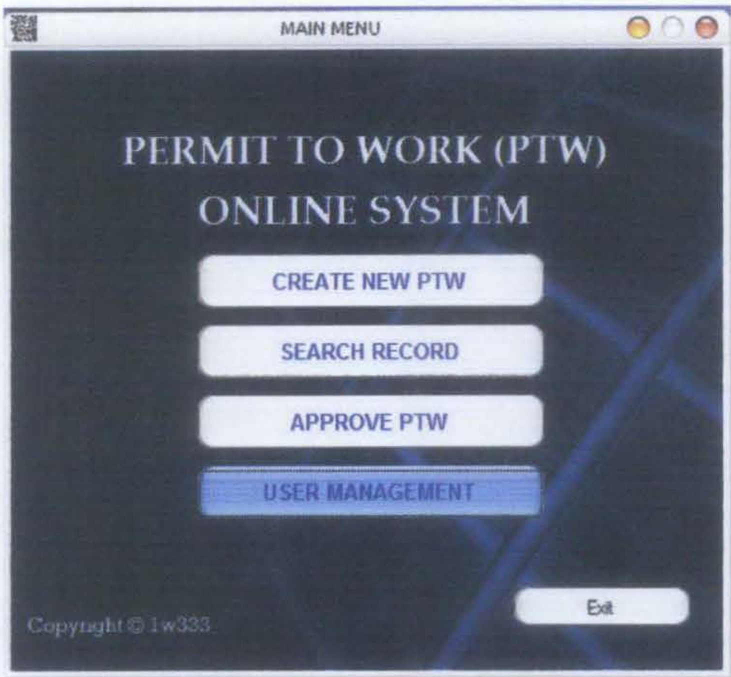


Figure 22: Main Menu of the Program

To create the new PTW, user can click the Create Button and the new form of PTW will appear for user to fill the required information. User can save the PTW by clicking the Save Button at the bottom of the form after completing all required information. There are also Cancel Button next to the Save Button for user to discard the changes made to the form. When user click the save button the message box will appear to notify the user that the record have been save to the database. After the record being saved, the PTW form will close then the Main Menu will appear again. The Approve Button will be used by the authorized person to approve the PTW. The visual of the PTW form are shown on the next page.

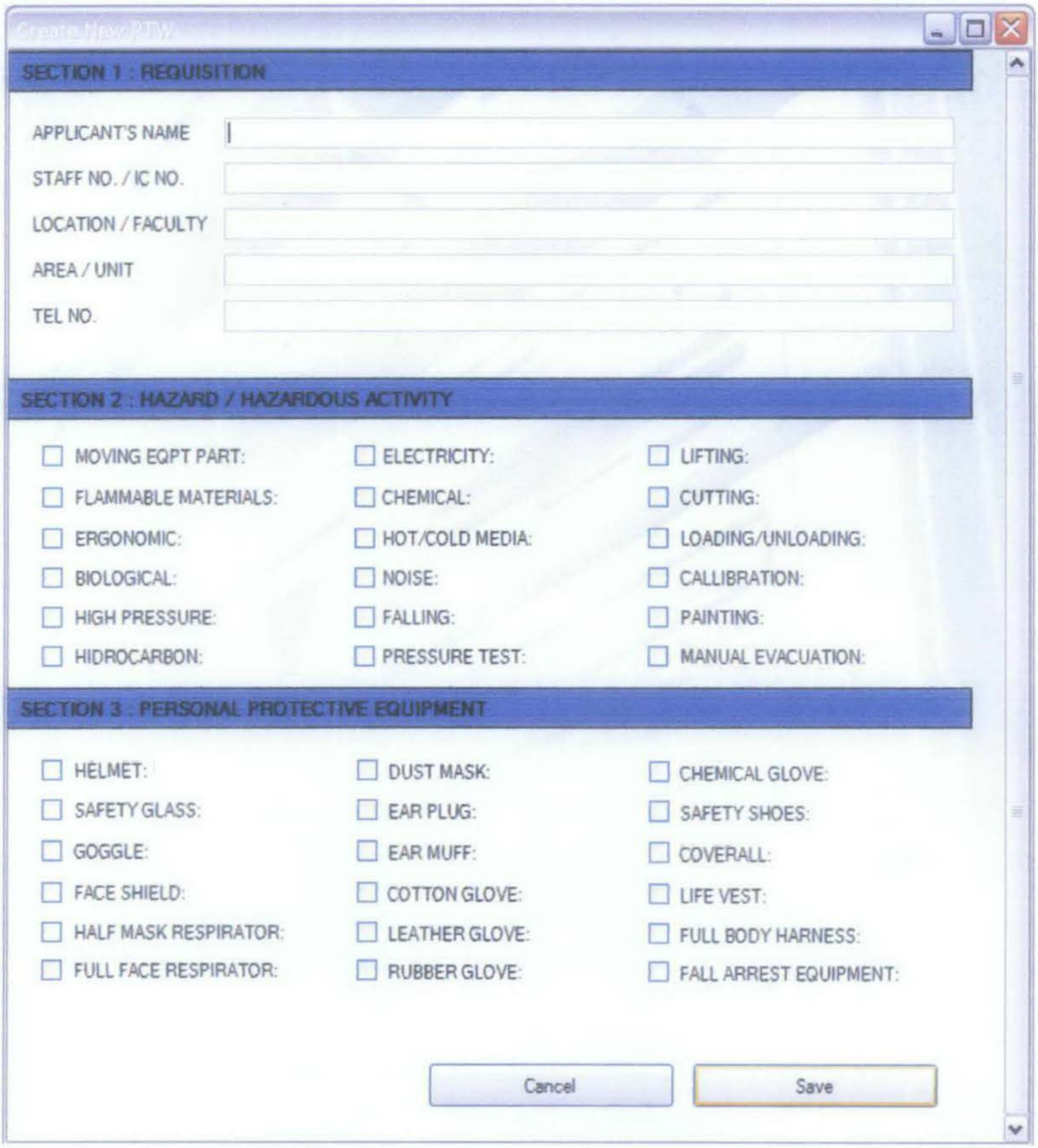


Figure 23: New Form for PTW

4.5.1 User Registration

User registration can be performed by clicking the User Management button on the Main Menu. In order to make sure that only system administrator can register new user, password will be required to continue the process. If the correct password were entered, then user needs to fill required information which is name, office and position. After that, user needs to register the fingerprint by putting the finger over the sensor three times. The purpose of taking fingerprint three times is to make sure that all minutiae were detected in order to increase the accuracy of the recognition system. The register button only enable after the third read of fingerprint. The flow of the registration process was shown by the figure below.

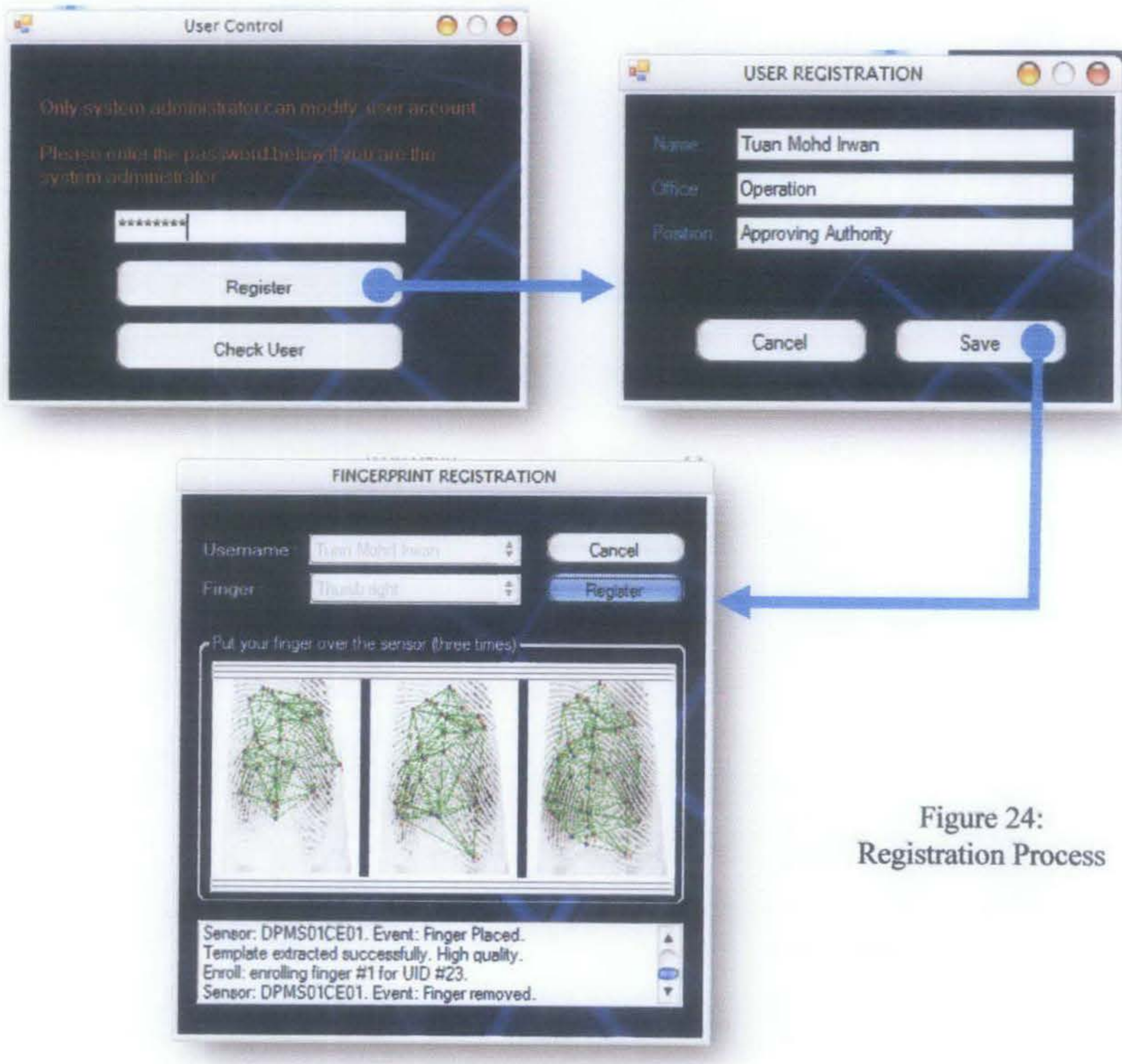


Figure 24: Registration Process

4.5.2 Approval Process

For approval process, the system will compare the input fingerprint with the existing records. To compare the fingerprint, the system will use the minutiae detected from the input image and compare it with the fingerprint on the database. The approval process only can be done by Approving Authority. After the system verifies that the user is approving authority, then the approval process will be permitted. The PTW that have been approved cannot be edited or reapproved.

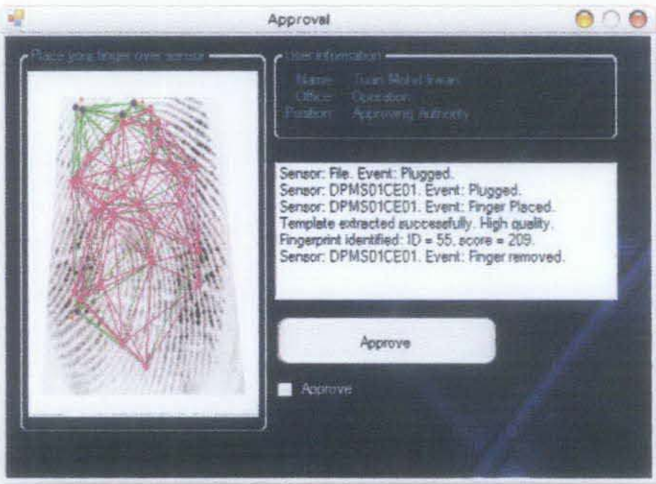


Figure 25:
Approval Form

CHAPTER 5

CONCLUSION AND RECOMMENDATION

5.1 Conclusion

The enhancement process that have been carry out is basically to make sure that the performance of an automatic fingerprint identification or verification system will be robust with respect to the quality of input of fingerprint images. The subsequent minutiae extraction process has been perform to extract the minutiae from fingerprint image. Experiments conducted have shown that this method is able to accurately detect all valid bifurcations and ridge endings from the thinned image. All analysis and enhancement of thumb-print image are being carried out using MATLAB platform and some of them are actually part of the MATLAB build in function. The best enhancement technique is filtering using ridge filter and the result shows the ridge in the fingerprint image clearly and it separate ridge and furrows. Analysis of fingerprint image is then carried out by using minutiae extraction to extract minutiae features on the image. The graphical user interface to use the system have been built and tested. From the analysis and obtained result, it shows that this system is applicable in P.T.W approval system. The security and reliability of the P.T.W. approval process will be increase since fingerprint is highly differentiated compared to signature. This system will reduce the time consumed because it converts manual operation into fully automated system.

5.2 Recommendation

This project still in the early stage and can be expand further. The GUI of the software also can be further utilizing in order to make sure that it is user friendly as possible. This project also can be use in other suitable application and not only for PTW approval. Some of the application that can be implement in the plant by using thumb-print recognition is accessing the store or workshop area. The accuracy of the system also can be increased to make sure that it is applicable and the system is more secure. The device that being used in the system also can be replace by other device which have low FRR and FAR. This PTW approval system can be implement in petroleum industry which apply permit system in the daily operation.

REFERENCES

[1] Rajibul Islam, Md. Shohel Sayeed, Andrews Samraj, "Multimodality to improve Security and Privacy in Fingerprint Authentication System"

[2] Shlomo Greenberg, Mayer Aladjem, Daniel Kogan and Itshak Dimitrov, "Fingerprint Image Enhancement using Filtering Techniques"

[3] S. Greenberg, "Adaptive anisotropic filter applied for Fingerprint enhancement"

[4] Hong, L., Wan, Y., and Jain, A. K. "Fingerprint image enhancement: Algorithm and performance evaluation". "IEEE Transactions on Pattern Analysis and Machine Intelligence 20, 8 (1998), pp 777-789"

[5] B. G. Sherlock, D. M. Monro, and K. Millard. "Fingerprint Enhancement by directional Fourier Filtering" Vision, Image and Signal Processing, IEEE Proceedings 141:87-94, April 2004

[6] N. Petkov and M.B. Wieling, "Gabor filter for image processing and computer vision" http://matlabserver.cs.rug.nl/edgedetectionweb/web/index.html

[7] International Association of Oil and Gas Producers, "Guidelines on Permit to Work (P.T.W) System. (1993).

[8] Andrew John Wilis and L. Mayers. A Cost-effective Fingerprint recognition system for use with low quality prints and damaged fingertips. Pattern Recognition, 34(2):255-270, 2001

APPENDICES

APPENDICES A: ADAPTIVE HISTOGRAM EQUALIZATION (*adapthisteq*)

APPENDICES B: FILTERING USING GABOR FILTER

APPENDICES C: FILTERING USING RIDGE FILTER

APPENDICES D: MINUTIAE EXTRACTION

APPENDICES A: ADAPTIVE HISTOGRAM EQUALIZATION *(adapthisteq)*

The syntax of using this function are :

- `J = adapthisteq(I)`
- `J = adapthisteq(I,param1,val1,param2,val2...)`

```
I=imread('19_7.bmp');
J=adapthisteq(I);
imshow(I);
figure,imshow(J);

J=adapthisteq(I,'clipLimit',0.02,'Distribution','rayleigh');
figure,imshow(J);
J=adapthisteq(I,'clipLimit',0.02,'Distribution','uniform');
figure,imshow(J);
J=adapthisteq(I,'clipLimit',0.02,'Distribution','exponential');
figure,imshow(J);

J=adapthisteq(I,'clipLimit',0.05,'numtiles',[2 2]);
figure,imshow(J)
J=adapthisteq(I,'clipLimit',0.05,'numtiles',[4 4]);
figure,imshow(J)
J=adapthisteq(I,'clipLimit',0.10,'numtiles',[2 2]);
figure,imshow(J)
J=adapthisteq(I,'clipLimit',0.10,'numtiles',[4 4]);
figure,imshow(J)
J=adapthisteq(I,'clipLimit',0.15,'numtiles',[2 2]);
figure,imshow(J)
J=adapthisteq(I,'clipLimit',0.15,'numtiles',[4 4]);
figure,imshow(J)


J=adapthisteq(I,'clipLimit',0.05,'numtiles',[6 6]);
figure,imshow(J)
J=adapthisteq(I,'clipLimit',0.05,'numtiles',[8 8]);
figure,imshow(J)
J=adapthisteq(I,'clipLimit',0.10,'numtiles',[6 6]);
figure,imshow(J)
J=adapthisteq(I,'clipLimit',0.10,'numtiles',[8 8]);
figure,imshow(J)
J=adapthisteq(I,'clipLimit',0.15,'numtiles',[6 6]);
figure,imshow(J)
J=adapthisteq(I,'clipLimit',0.15,'numtiles',[8 8]);
figure,imshow(J)
```

Parameter and Value for the *adapthisteq* function

PARAMETER	VALUE
'NumTiles'	Two-element vector of positive integers specifying the number of tiles by row and column, [M N]. Both M and N must be at least 2. The total number of tiles is equal to M*N. Default: [8 8]
'ClipLimit'	Real scalar in the range [0 1] that specifies a contrast enhancement limit. Higher numbers result in more contrast. Default: 0.01
'NBins'	Positive integer scalar specifying the number of bins for the histogram used in building a contrast enhancing transformation. Higher values result in greater dynamic range at the cost of slower processing speed. Default: 256
'Range'	String specifying the range of the output image data. 'original' -- Range is limited to the range of the original image, [min(I(:)) max(I(:))]. 'full' -- Full range of the output image class is used. For example, for uint8 data, range is [0 255]. Default: 'full'
'Distribution'	String specifying the desired histogram shape for the image tiles. 'uniform' -- Flat histogram 'rayleigh' -- Bell-shaped histogram 'exponential' -- Curved histogram Default: 'uniform'
'Alpha'	Nonnegative real scalar specifying a distribution parameter. Default: 0.4 Note: Only used when 'Distribution' is set to either 'rayleigh' or 'exponential'.

APPENDICES B: FILTERING USING GABOR FILTER

The Gabor Filter function is :

```
[G,gabout] = gaborfilter(I,Sx,Sy,U,V);
```

Where :

G :	The output filter
gabout :	The output filtered image
I :	Input image
Sx & Sy :	Variances along x and y-axes respectively
U & V :	Centre frequencies along x and y-axes respectively

The MATLAB coding to use Gabor Filter for this project are

```
I=imread('19_7.bmp');
U=0.5; V=0.5;
Sx=0.5; Sy=0.5;
[G,gabout] = gaborfilter(I,Sx,Sy,U,V);
figure,imshow(gabout)

U=50.0; V=50.0;
Sx=0.5; Sy=0.5;
[G,gabout] = gaborfilter(I,Sx,Sy,U,V);
figure,imshow(gabout)

U=0.5; V=0.5;
Sx=0.35; Sy=0.35;
[G,gabout] = gaborfilter(I,Sx,Sy,U,V);
figure,imshow(gabout)

U=50.0; V=50.0;
Sx=0.5; Sy=0.5;
[G,gabout] = gaborfilter(I,Sx,Sy,U,V);
figure,imshow(gabout)

U=0.5; V=0.5;
Sx=0.25; Sy=0.25;
[G,gabout] = gaborfilter(I,Sx,Sy,U,V);
figure,imshow(gabout)

U=50.0; V=50.0;
Sx=0.25; Sy=0.25;
[G,gabout] = gaborfilter(I,Sx,Sy,U,V);
figure,imshow(gabout)
```

APPENDICES C: FILTERING USING RIDGE FILTER

```
im = imread('19_7.bmp');
imshow('19_7.bmp');

% Identify ridge-like regions and normalise image
blksze = 16; thresh = 0.1;
[normim, mask] = ridgesegment(im, blksze, thresh);
figure,imshow(normim);

% Determine ridge orientations
[orientim, reliability] = ridgeorient(normim, 1, 5, 5);
plotridgeorient(orientim, 20, im, 2)


% Determine ridge frequency values across the image
blksze = 36;
[freq,medfreq]=ridgefreq(normim,mask,orientim,blksze,5,5,15);
figure,imshow(freq)


% use medium frequency
freq = medfreq.*mask;

% apply filters to enhance the ridge pattern
newim = ridgefilter(normim, orientim, freq, 0.5, 0.5, 1);
figure,imshow(newim);
```

APPENDICES D: MINUTIAE EXTRACTION

```
% Load Image
I=imread('37_5_2.bmp');
imshow(I)

% Binarize image
J=im2bw(I,graythresh(I));
figure,imshow(J)

% Thinning binarize image
K=bwmorph(~J,'thin','inf');
figure,imshow(~K)

% Filter thinned image to find termination and bifurcation
fun = @minutie;
L = nlfilter(K,[3 3],fun);

%% Termination
LTerm=(L==1);
figure,imshow(LTerm)
LTermLab=bwlabel(LTerm);
propTerm=regionprops(LTermLab,'Centroid');
CentroidTerm=round(cat(1,propTerm(:).Centroid));
figure,imshow(~K)
hold on
plot(CentroidTerm(:,1),CentroidTerm(:,2),'ro')

%% Bifurcation
LBif=(L==3);
LBifLab=bwlabel(LBif);
propBif=regionprops(LBifLab,'Centroid','Image');
CentroidBif=round(cat(1,propBif(:).Centroid));
plot(CentroidBif(:,1),CentroidBif(:,2),'go')
```

Removing spurious minutiae

```
%% Process 1
Distance=DistEuclidian(CentroidBif,CentroidTerm);
SpuriousMinutae=Distance<D;
[i,j]=find(SpuriousMinutae);
CentroidBif(i,:)=[];
CentroidTerm(j,:)=[];

%% Process 2
Distance=DistEuclidian(CentroidBif);
SpuriousMinutae=Distance<D;
[i,j]=find(SpuriousMinutae);
CentroidBif(i,:)=[];

%% Process 3
Distance=DistEuclidian(CentroidTerm);
SpuriousMinutae=Distance<D;
[i,j]=find(SpuriousMinutae);
CentroidTerm(i,:)=[];


hold off
figure,imshow(~K)
hold on
plot(CentroidTerm(:,1),CentroidTerm(:,2),'ro')
plot(CentroidBif(:,1),CentroidBif(:,2),'go')
hold off
```