Fingerprint Validated Mailing System

Project Report submitted in partial fulfillment of the requirement for the degree of

Bachelor of Technology

In

INFORMATION TECHNOLOGY

Under the Supervision of

Dr. Vivek Sehgal

By

Charuta Jassal (111462)

To

Jaypee University of Information and Technology

Waknaghat, Solan -173234, Himachal Pradesh

Certificate

This is to certify that project report entitled “Fingerprint validated mailing system”, submitted by Charuta Jassal in partial fulfillment for the award of degree of Bachelor of Technology in Computer Science & Engineering to Jaypee University of Information Technology, Waknaghat, Solan has been carried out under my supervision.

This work has not been submitted partially or fully to any other University or Institute for the award of this or any other degree or diploma.

Date: **Dr. Vivek Sehgal**

Associate Professor

Acknowledgement

The satisfaction that accompanies that the successful completion of any task would be incomplete without the mention of people whose ceaseless cooperation made it possible, whose constant guidance and encouragement crown all efforts with success. I am grateful to my project guide
Dr. Vivek Sehgal for the guidance, inspiration and constructive suggestions that helpful me in the preparation of this project.

I also thank my colleagues who have helped in successful completion of the project.

Date: Charuta Jassal

Table of Content

S. No.	Topic	Page No.
1.	Introduction	1
2.	Biometrics	4
2.1	Fingerprints	6
3.	Fingerprint sensors	8
4.	Feature extraction	10
4.1	Features	11
4.2	Preprocessing of image	14
4.3	Minutiae extraction	16
5.	Database module	26
6.	Matching module	28
7.	Why fingerprints?	35
8.	Mailing site	38
9.	Conclusion and future work	49
10.	Biblography	50

List of Figures

S.No.	Title	Page No.
1.	Fingerprint based personal authentication	2
2.	Fingerprint image acquired by optical sensor	7
3.	Sensor working	9
4.	Ridges and valleys in fingerprint	10
5.	Termination and bifurcation	11
6.	Major fingerprint pattern types	12
7.	Level 2 features	13
8.	Level 3 features	13
9.	Thinned image	16
10.	Extraction steps overview	17
11.	Position and orientation convention	18
12.	Example of ridge ending and bifurcation pixel	20
13.	Removing spurious minutiae	22
14.	Minutiae points extracted	25
15.	Database created snippet	27
16.	Flowchart of minutiae matching algorithm	28
17.	Minutiae set alignment	30
18.	Minutia descriptor	31
19.	Fingerprint matching	32

List of Tables

S.No.	Title	Page No.
1.	Properties of the Crossing Number	19
2.	Comparison of different biometrics	35

Abstract

Fingerprint matching is the process used to determine whether two sets of fingerprint Ridge detail come from the same finger. There exist multiple algorithms that do Fingerprint matching in many different ways. Some methods involve matching minutiae Points between the two images, while others look for similarities in the bigger structure of the fingerprint .In order to protect users of computer systems and to secure network-based transactions, demand is in- creasing for improved user authentication procedures to establish the identity of an actual user and to bar access to a terminal to anyone who is unauthorized.

Among various modalities in biometrics, such as fingerprints, face, iris, etc., fingerprints are the most widely used and have the longest history in real- world law enforcement applications. Stable, reliable and highly accurate identification software is currently available even for use on personal computers. Fingerprint sensors can be made small and thin enough to be implemented easily on small computers and even on pocket-sized terminals. Fingerprint as a password is a good layer of security added to the mailing accounts which will reduce the attacks of intruders.

The scanner that will be used in this project is the Fingerprint Reader. The primary problem in using this particular scanner comes from the security features. The signal caring all the information from the fingerprint scan is encrypted, and the biggest challenge will come from deciphering this signal. Another major challenge of this project will be understanding the methods used to analyze the information from the scan.

In this project we propose a method for fingerprint matching based on minutiae matching. However, unlike conventional minutiae matching algorithms our algorithm also takes into account region and line structures that exist between minutiae pairs. This allows for more structural information of the fingerprint to be accounted for thus resulting in stronger certainty of matching minutiae. Also, since most of the region analysis is preprocessed it does not make the algorithm slower. Evidence from the testing of the preprocessed images gives stronger assurance that using such data could lead to faster and stronger matches.

CHAPTER 1
INTRODUCTION

In order to protect computer users and to secure network-based transactions, demand of improved user authentication procedures to create actual user’s identity and to bar access to one who is unauthorized is increasing. Personal identification using biometrics, i.e. a person’s physical or behavioral characteristics, has come to be a possible solution to this issue and one that might offer reliable systems at a reasonable cost. While traditionally this technology has been available only with such expensive, high-end systems as those used in law enforcement and other government applications, today many personal-level applications have also become possible thanks to the advancements in pattern recognition technology.

Among various categories in biometrics, such as fingerprints, face, iris, etc., fingerprints are the most widely used and have the longest history in real- world applications. Research into automated fingerprint identification began in the 1960’s, and the resulting AFISs i.e. Automated Finger- print Identification Systems have been used world- wide with established dependability. Millions of identifications over a century of actual forensic history have clearly shown that fingerprints are unique and permanent and thus that fingerprint identification is extremely reliable. Recent technical advances have made identification systems low enough in cost for civilian applications.

Fingerprints have, among many, the following two advantages when compared with other modalities:

Stable, reliable and highly accurate identification software is currently available even for use on personal computers.

Fingerprint sensors can be made small and thin enough to be implemented easily on small computers and even on pocket-sized terminals.

A fingerprint-based personal authentication system operates in two distinct modes: enrollment and authentication as is shown in Fig. 1. During enrollment, a fingerprint image is acquired from a finger presented by an authorized user using a “fingerprint sensor,” and relevant features are extracted by the features extractor. The set of extracted features, also referred to as a “template” is stored in a database, along with the user’s information necessary for granting service, and some form of ID assigned for the user.

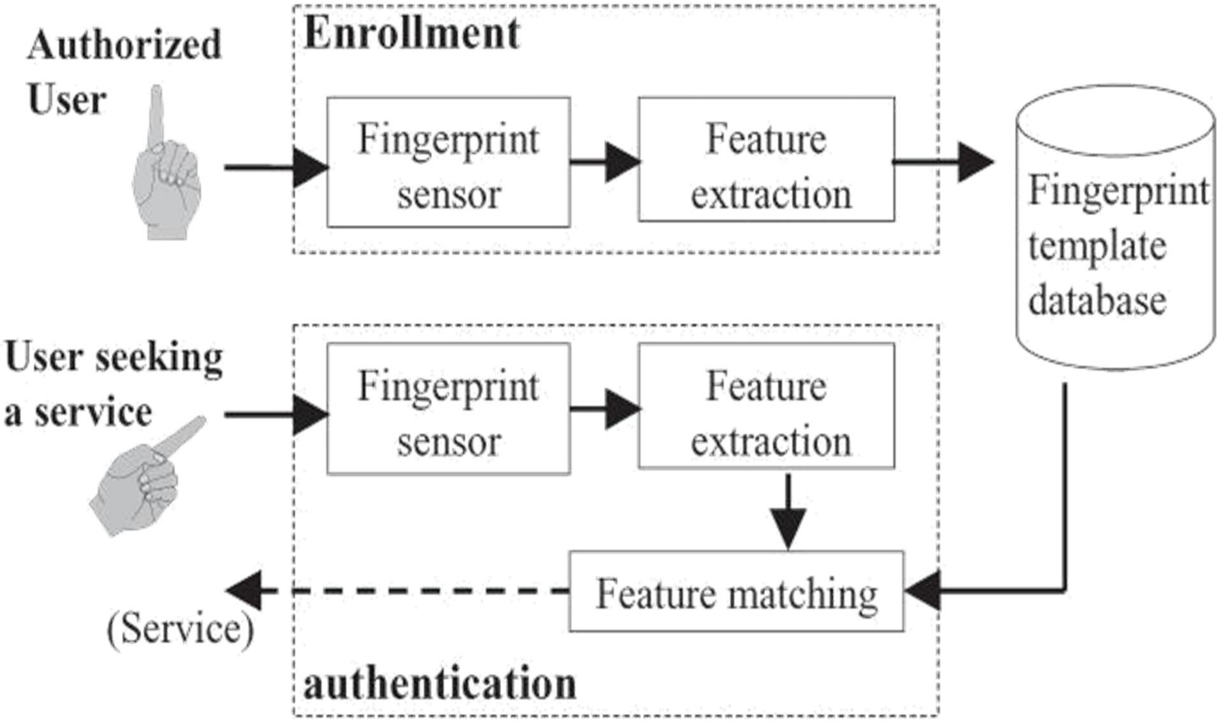


Fig. 1 Fingerprint-based personal authentication.

When the user seeks for a service, i.e. in authentication mode, the user inputs his assigned ID and presents his fingerprint to the sensor. The system captures the image, extracts (input) features from it, and attempts to match the input features to the tem-plate features corresponding to the subject’s ID in the system database.

If the calculated similarity score between the input and the template is larger than the predetermined threshold, the system determines that the subject is who he claims to be and offer the service; otherwise would reject the claim.

In identification mode, on the other hand, the user who seeks for a service presents his fingerprint only without his ID, and the system may either be able to determine the identity of the subject or decide the person is not enrolled in the database.

CHAPTER 2

BIOMETRICS

Biometrics is the science of verifying the identity of an individual through physiological measurements or behavioral characteristics. Since biometric identifiers are associated permanently with the user they are more reliable than knowledge based authentication methods.

Biometrics offers several advantages over traditional security measures. Some of them are presented below.

1. Accuracy and Security

Biometrics based security systems are far most secure and accurate than traditional password or token based security systems. For example a password based security system has always the threat of being stolen and accessed by the unauthorized user. Furthermore the traditional security systems are always prone to accuracy as compared to biometrics which is more accurate.

2. One individual, Multiple IDs

Traditional security systems face the problem that they don't give solution to the problem of individuals having multiple IDs. For examples a person having multiple passports to enter a foreign country. They give us a system in which an individual can't possess multiple IDs and can't change his ID throughout his life time. Each individual is identified through a unique Biometric identity throughout the world.

3. One ID, multiple individuals

In traditional security systems one ID can be used by multiple individuals. For example in case of a password based security system a single password can be shared among multiple individuals and they can share the resources allotted to a single individual. Biometric based security system doesn't allow such a crime. Here each individual has a single unique ID and it can't be shared with any other individual.

Biometrics categories

Biometrics can be categorized in various categories as follow.

1. Physical biometrics
2. Behavioral biometrics

Physical biometrics

This biometrics involves measurement of physical characteristics of individuals. The most prominent of these include

- Fingerprints
- Face
- Hand geometry
- Iris scans

Behavioral biometrics

This category of biometrics is temporal in nature. They are evolved during the lifetime of an individual. It involves measuring the way in which an individual performs certain tasks. Behavioral biometrics include

- Gait
- Handwriting
- Speech
- Signature

2.1 FINGERPRINTS

Fingerprints have been scientifically studied for many years in our society. The characteristics of fingerprints were studied as early as 1600s. Meanwhile, using fingerprints as a means of identification first occurred in the mid-1800s. Sir William Herschel, in 1859, discovered that fingerprints do not change over time and that each pattern is unique to an individual. With these findings, he was the first to implement a system using fingerprints and handprints to identify an individual in 1877. By 1896, police forces in India realized the benefit of using fingerprints to identify criminals, and they began collecting the fingerprints of prisoners along with their other measurements

With a growing database of fingerprint images, it soon became desirable to have an efficient manner of classifying the various images. Between 1896 and 1897, Sir Edward Henry developed the Henry Classification System, which quickly found worldwide acceptance within a few years. This system allows for logical categorization of a complete set of the ten fingerprint images for a person. By establishing groupings based on fingerprint pattern types, the Henry System greatly reduces the effort of searching a large database. Until the mid-1990s, many organizations continued to use the Henry Classification System to store their physical files of fingerprint images.

As fingerprints began to be utilized in more fields, the number of requests for fingerprint matching began to increase on a daily basis. At the same time, the size of the databases continued to expand with each passing day. Therefore, it soon became difficult for teams of fingerprint experts to provide accurate results in a timely manner. In the early 1960s, the FBI, Home Office in the United Kingdom, and Paris Police Department began to devote a large amount of resources in developing automatic fingerprint identification systems. These systems allowed for an improvement in operational productivity among law enforcement agencies. At the same time, the automated systems reduced funding requirements to hire and train human fingerprint experts. Today, automatic fingerprint recognition technology can be found in a wide range of civilian applications.

What is a Fingerprint?

A fingerprint is the feature pattern of one finger. It is believed with strong evidences that each fingerprint is unique. Each person has his own fingerprints with the permanent uniqueness. So fingerprints have being used for identification and forensic investigation for a long time. Fingerprint recognition is one of the most reliable identification techniques .Fingerprint technology is the most widely used for security purposes. The Technology is being frequently used in criminal investigation purpose.

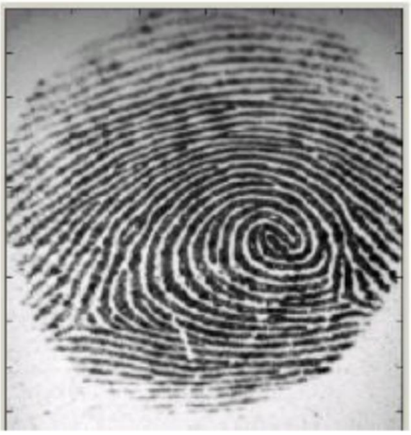


Fig. 2 Fingerprint image acquired by an Optical Sensor.

CHAPTER 3

FINGERPRINT SENSORS

A fingerprint is a pattern of fine ridges and valleys i.e. spaces between ridges on the surface of a finger, and a fingerprint sensor makes a digitized image of it. The sensing resolution is 500ppi (pixel per inch; also known as 500dpi, i.e., dots per inch) in most cases, which is equivalent to 20 pixels in 1 millimeter. The obtained image size is typically in the range of between 300 300 and 512 512 pixels, which makes the area covering the fingerprint between 15 to 25 millimeters square.

The most important part of a fingerprint scanner is the sensor (or sensing element), which is the component where the fingerprint image is formed. Almost all the existing sensors belong to one of the three families: optical, solid-state, and ultrasound:

Optical sensors: Frustrated Total Internal Reflection (FTIR) is the oldest and most used live-scan acquisition technique. The finger touches the top side of a glass prism, but while the ridges enter in contact with the prism surface, the valleys remain at a certain distance; the left side of the prism is illuminated through a diffused light. The light entering the prism is reflected at the valleys, and absorbed at the ridges. The lack of reflection allows the ridges to be discriminated from the valleys. The light rays exit from the right side of the prism and are focused through a lens onto a CCD or CMOS image sensor.

Solid-state sensors. Solid-state sensors (also known as silicon sensors) became commercially available in the middle 1990s. All silicon-based sensors consist of an array of pixels, each pixel being a tiny sensor itself. The user directly touches the surface of the silicon: neither optical components nor external CCD/CMOS image sensors are needed. Four main effects have been proposed to convert the physical information into electrical signals: capacitive, thermal, electric field, and piezoelectric.

Ultrasound sensors. Ultrasound sensing is as a kind of echography. A characteristic of sound waves is the ability to penetrate materials, giving a partial echo at each impedance change. This technology is not yet mature enough for large-scale production.

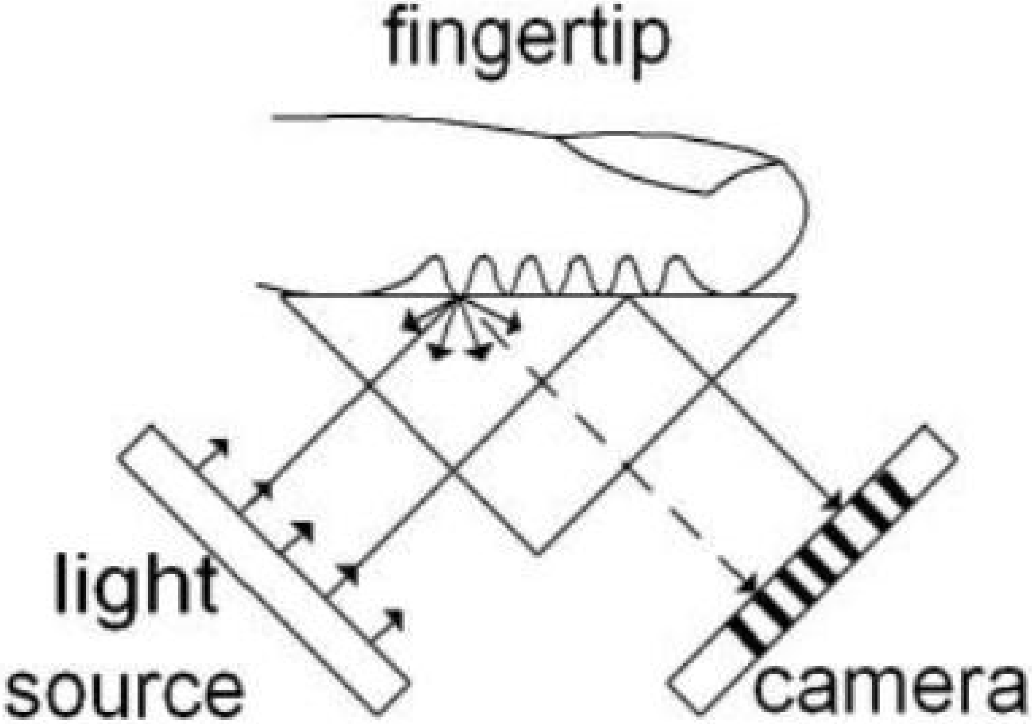


Fig. 3 Sensor working

When user puts his/her finger on the scanner then accordingly the light waves from the source strike to the surface and refract and strike the camera lens and a image is generated.

CHAPTER 4

FEATURE EXTRACTION

In a fingerprint image, ridges (also called ridge lines) are dark whereas valleys are bright (see Figure 2.1a). Ridges and valleys often run in parallel; sometimes they bifurcate and sometimes they terminate. When analyzed at the global level, the fingerprint pattern exhibits one or more regions where the ridge lines assume distinctive shapes. These regions (called singularities or singular regions) may be classified into three typologies: loop, delta, and whorl (see Figure 2.1b). Singular regions belonging to loop, delta, and whorl types are typically characterized by ∩, Δ, and O shapes, respectively. The core point (used by some algorithms to pre-align fingerprints) corresponds to the center of the north most (uppermost) loop type singularity.

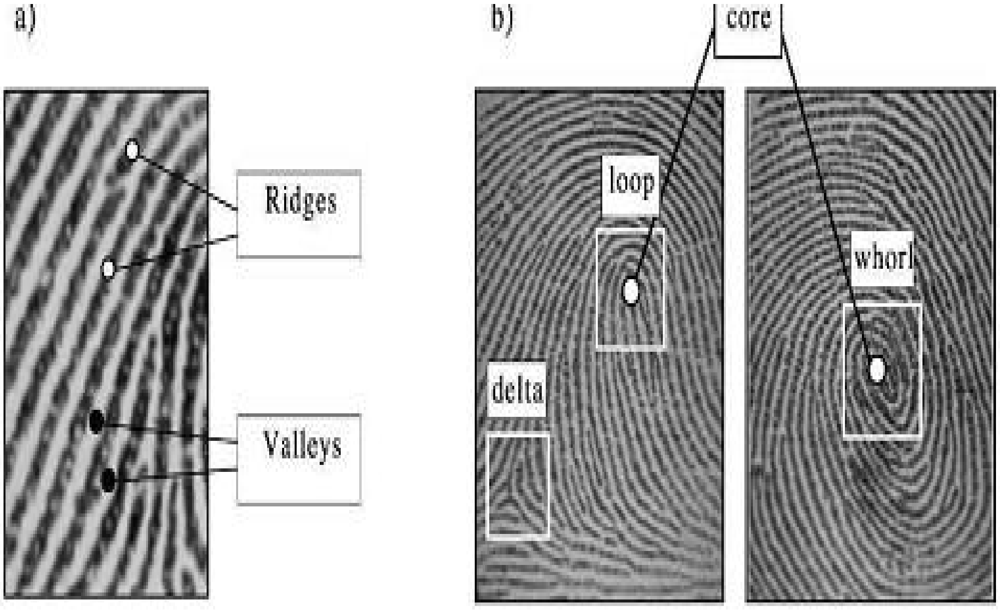


Fig. 4 a) Ridges and valleys in a fingerprint image; b) singular regions (white boxes) and core points (circles) in fingerprint images.

At the local level, other important features, called minutiae can be found in the fingerprint patterns. Minutia refers to the various ways in which the ridges can be discontinuous. For example, a ridge can abruptly come to an end (termination), or can divide into two ridges (bifurcation) (Figure 2.2). Although several types of minutiae can be considered, usually only a coarse classification (into these two types) is adopted to deal with the practical difficulty in automatically discerning the different types with high accuracy.

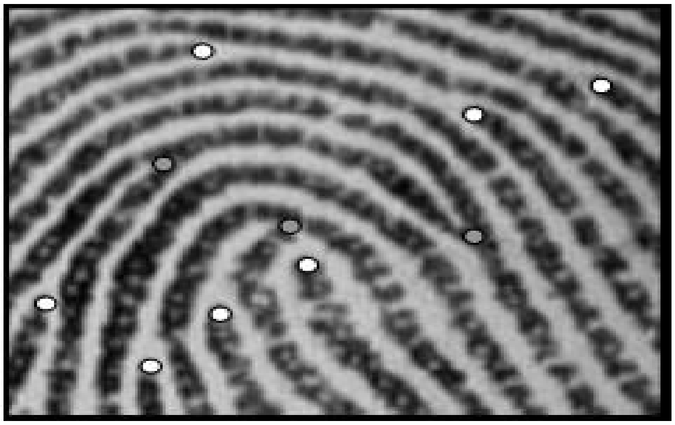


Fig. 5. Termination (white) and bifurcation (gray) minutiae in a sample fingerprint.

4.1 Features

The details in a fingerprint can be characterized at three different levels ranging from coarse to fine. Under ideal conditions, coarse level features can be derived from the finer levels of fingerprint representation.

Level 1 features

At the first (coarsest) level, a fingerprint is represented as a ridge orientation map which records the local ridge orientation at each location of the fingerprint, and a ridge frequency map, which records the local ridge frequency at each location in the fingerprint. A fingerprint is often referred to as an oriented texture pattern since its global shape and structure can be defined by the orientation and frequency of its ridges. In Level 1 detail, only the ridge flow and ridge frequency are observed; the exact location and dimensional details of ridges are ignored. Thus, low-resolution image sensors capable of scanning 250 pixels per inch (ppi) can be used to observe the Level 1 details of a fingerprint.

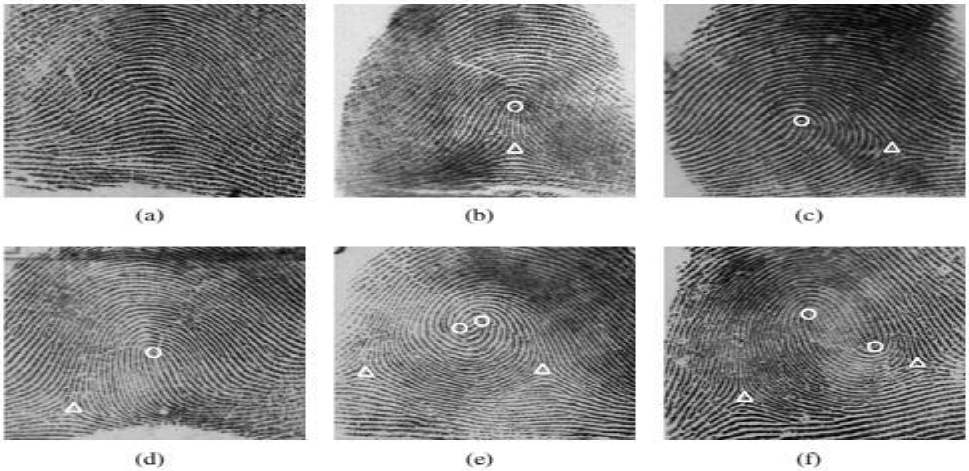


Fig. 6 Major fingerprint pattern types. (a) Plain arch, (b) tented arch, (c) left loop, (d) right loop, (e) whorl, and (f) twin loop.

Level 2 features

In the second (middle) level, a fingerprint is represented as a ridge skeleton image in which each ridge is only one-pixel wide At this level, the exact locations of the ridges are recorded, but the geometric and dimensional details of the ridges are ignored. The locations where a ridge emerges, ends, splits, or merges with another ridge are termed as ridge characteristics or minutiae. In addition to its location, a minutia generally has two other properties: direction and type. The direction of a minutia is along the local ridge orientation. There are two basic types of minutiae: ending (also called ‘termination’) and bifurcation .Thus, each minutia can be characterized by its (a) location in the image, (b) direction and (c) type. Level 2 details of a fingerprint can be easily observed in images acquired at a resolution of 500 ppi.

The number of minutiae found in a fingerprint varies a lot according to the acquisition method and other factors.

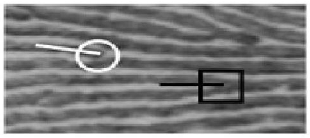


Fig. 7 There are two types of minutiae used to represent the Level 2 details in a fingerprint: ridge ending (denoted as white circle) and ridge bifurcation (denoted as black box).

Level 3 features

In the third (finest) level, a fingerprint is represented using both the inner holes (sweat pores) and outer contours (edges) of the ridges. So the ridges are no longer viewed as being simple, one-pixel wide skeletal images. Rather, information embedded within the ridges are observed in detail. Incipient ridges and dots are also included at this level.

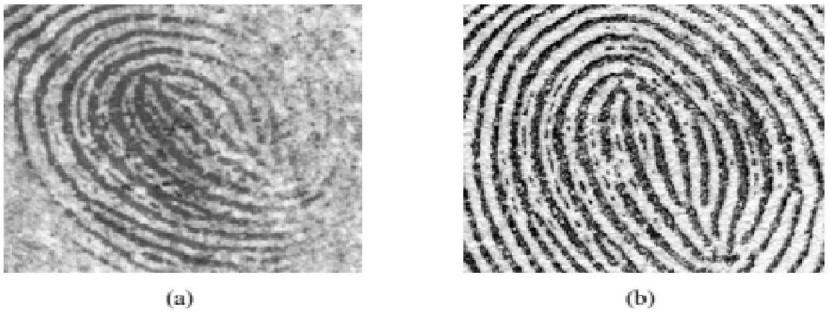


Fig. 8 Level 3 features observed in a latent and its mated rolled fingerprint. (a) Latent fingerprint with pores and incipient ridges and (b) the mated rolled print with the same configuration of the Level 3 features observed in (a)

Incipient ridges are immature ridges, which are thinner than mature ridges and contain no sweat pores. A dot is a very short ridge containing only a single ridge unit. With advances in fingerprint sensing technology, many sensors are now equipped with 1000 ppi scanning capability that is needed to capture the Level 3 details in a fingerprint.

4.2 Preprocessing of image

Usually, the raw biometric data from the sensor is subjected to pre-processing operations before features are extracted from it.

The three commonly used pre-processing steps are

(a) Quality assessment,

(b) Segmentation, and

(c) Enhancement.

First, the quality of the acquired biometric samples needs to be accessed to determine its suitability for further processing. If the raw data is not of sufficient quality, there are two options. One can either attempt to re-acquire the data from the user or trigger an exception (failure alarm) alerting the system administrator to activate suitable alternate procedures (typically involving some form of manual intervention by the system operator).

The next pre-processing step is known as segmentation, where the goal is to separate the required biometric data from the background noise. Detecting a face in a cluttered image is a good example of segmentation.

Finally, the segmented biometric data is subjected to a signal quality enhancement algorithm in order to improve its quality and further reduce the noise. In the case of image data, enhancement algorithms like smoothing or histogram equalization may be applied to minimize the noise introduced by the camera or illumination variations.

In some cases, the above pre-processing steps may be inseparable from the actual feature extraction step. For example, quality assessment in itself may entail the extraction of some features from the acquired biometric data.

Feature extraction refers to the process of generating a compact but expressive digital representation of the underlying biometric trait, called a template .The template is expected to contain only the salient discriminatory information that is essential for recognizing the person. For example, the position and orientation of minutia points (locations where the friction ridges in a fingerprint pattern exhibit some anomalies) are believed to be unique for each finger. Therefore, detecting the minutia points in a fingerprint image is a key feature extraction step.

Function used in implementation:

```
edge(RGB2GRAY(img), 'canny');
```

Edge detection is an image processing technique for finding the boundaries of objects within images. It works by detecting discontinuities in brightness. Edge detection is used for image segmentation and data extraction in areas such as image processing, computer vision, and machine vision.

This function looks for places in the image where the intensity changes rapidly, using one of these two criteria:

Places where the first derivative of the intensity is larger in magnitude than some threshold.

Places where the second derivative of the intensity has a zero crossing.

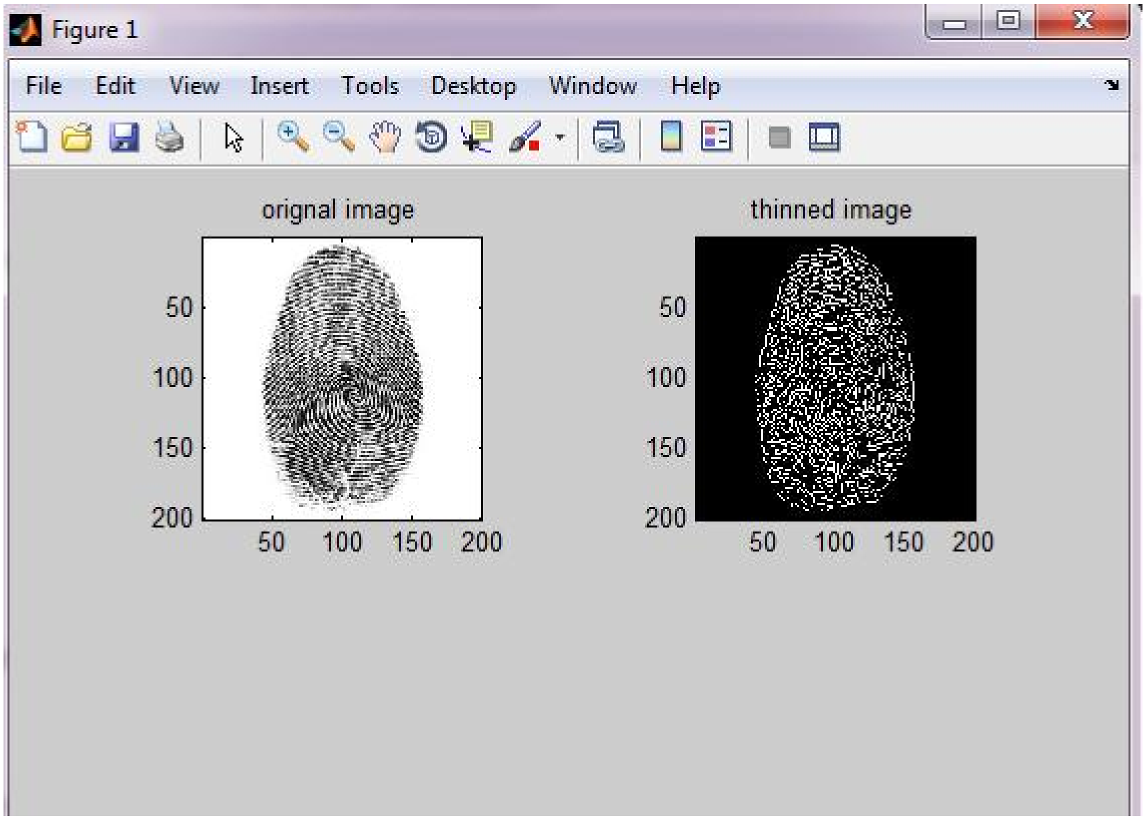


Fig 9. Thinned image of an original image after using function

4.3 Minutiae Extraction

Minutia are the most essential feature. Most of the proposed methods require the fingerprint gray-scale image to be converted into a binary image. The binary images obtained by the binarization process are submitted to a thinning stage which allows for the ridge line thickness to be reduced to one pixel. Finally, a simple image scan allows the detection of pixels that correspond to minutiae through the pixel-wise computation of crossing number.

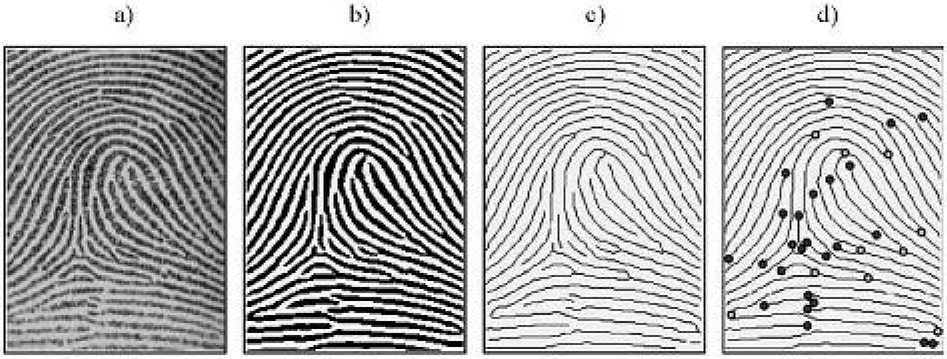


Fig. 10. a) A fingerprint gray-scale image; b) the image obtained after enhancement and binarization; c) the image obtained after thinning; d) termination and bifurcation minutiae detected through the pixel-wise computation of the crossing number.

Some authors have proposed minutiae extraction approaches that work directly on the gray-scale images without binarization and thinning. This choice is motivated by the following considerations:

i) A significant amount of information may be lost during the binarization process;

ii) Thinning may introduce a large number of spurious minutiae;

iii) Most of the binarization techniques do not provide satisfactory results when applied to low-quality images. Maio and Maltoni proposed a direct gray-scale minutiae extraction technique, whose basic idea is to track the ridge lines in the gray-scale image, by ‘sailing’ according to the local orientation of the ridge pattern.

A post-processing stage (called minutiae filtering) is often useful in removing the spurious minutiae detected in highly corrupted regions or introduced by previous processing steps (e.g., thinning).

Minutiae points are extracted during the enrollment process and then for each authentication. In a fingerprint, they correspond to either a ridge ending or a bifurcation (figure 3.7). There is a duality between the two types of minutiae: if the pixel brightness is inverted, ridge endings become bifurcations and vice versa. The position of the minutia point is at the tip of the ridge or the valley. The orientation is given by the orientation of the arrow formed by the ridge or the valley according to figure 3.7. First, the local orientation field need to be computed. This will allow to enhance the print using oriented Gabor filter, and then better detect minutiae point using template matching procedure.

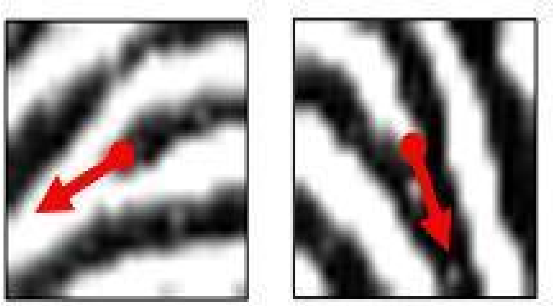
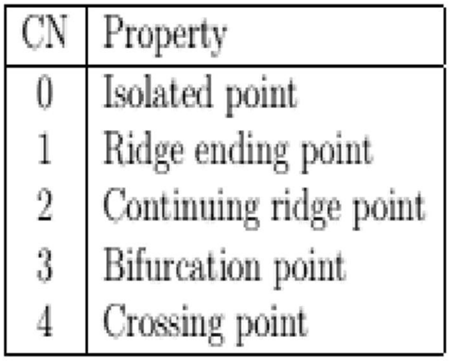


Figure 11. Ridge ending and bifurcation - Position and orientation convention

The most commonly employed method of minutiae extraction is the Crossing Number (CN) concept .This method involves the use of the skeleton image where the ridge flow pattern is eight-connected. The minutiae are extracted by scanning the local neighborhood of each ridge pixel in the image using a window.

The CN value is then computed, which is defined as half the sum of the differences between pairs of adjacent pixels in the eight-neighborhood. Using the properties of the CN as shown in Table 3.1, the ridge pixel can then be classified as a ridge ending, bifurcation or non-minutiae point. For example, a ridge pixel with a CN of one corresponds to a ridge ending, and a CN of three corresponds to a bifurcation.



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

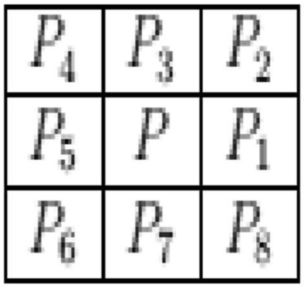
Table 1: Properties of the Crossing Number.

The Crossing Number (CN) method is used to perform minutiae extraction. This method extracts the ridge endings and bifurcations from the skeleton image by examining the local neighborhood of each ridge pixel using a 3 £ 3 window. The

CN for a ridge pixel P is given by

$$CN = 0.5\sum_{i=1}^{8}|P_i - P_{i+1}|, \qquad P_9 = P_1$$

where Pi is the pixel value in the neighborhood of P . For a pixel P, its eight neighboring pixels are scanned in an anti-clockwise direction as follow:



P_4	P_3	P_2
P_5	P	P_1
P_6	P_7	P_8

After the CN for a ridge pixel has been computed, the pixel can then be classified according to the property of its CN value. As shown in Figure 3.2, a ridge pixel with a CN of one corresponds to a ridge ending, and a CN of three corresponds to a bifurcation. For each extracted minutiae point, the following information is recorded: x and y coordinates orientation of the associated ridge segment, and type of minutiae (ridge ending or bifurcation).

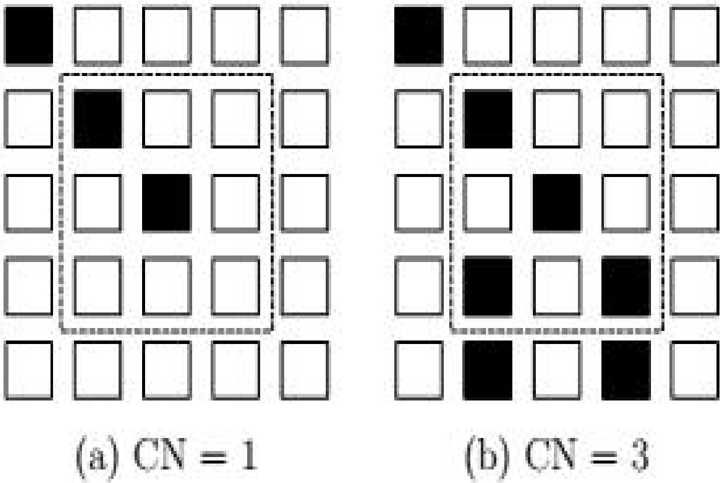

(a) CN = 1
(b) CN = 3

Figure 12. Examples of a ridge ending and bifurcation pixel. (a) A Crossing

Number of one corresponds to a ridge ending pixel. (b) A Crossing Number of three corresponds to a bifurcation pixel.

In order to eliminate false minutiae, I have chosen to implement the minutiae validation algorithm by Tico and Kuosmanen. This algorithm tests the validity of each minutiae point by scanning the skeleton image and examining the local neighbourhood around the point. The first step in the algorithm is to create an image M of size W X W , where M corresponds to the W £ W neighbourhood centred on the candidate minutiae point in the skeleton image. The central pixel of M corresponds to the minutiae point in the skeleton image, and so this pixel is labelled with a value of ¡1. The rest of the pixels in M are initialised to values of zero, as shown in Figure 3.3(a) and Figure 3.4(a). The subsequent steps of the algorithm depend on whether the candidate minutiae point is a ridge ending or a bifurcation.

1. For a candidate ridge ending point:

(a) Firstly, label with a value of 1 all the pixels in M , which are eight-connected with the ridge ending point .

(b) The next step is to count in a clockwise direction, the number of 0 to 1 transitions (T01) along the border of image M . If T01= 1, then the candidate minutiae point is validated as a true ridge ending.

For a candidate bifurcation point:

(a) Firstly, examine the eight neighbouring pixels surrounding the bifurcation point in a clockwise direction. For the three pixels that are connected with the bifurcation point, label them with the values of 1, 2, and 3, respectively.

(b) The next step is to label the rest of the ridge pixels that are connected to these three connected pixels. This labelling is similar to the ridge ending approach, however, instead of labelling a single ridge branch, three ridge branches are now labelled. Let l = 1, 2 and 3 represent the label for each ridge branch. For each l, label with l all the ridge pixels that

have a label of 0, and are connected to an l labelled pixel.

(c) The last step is to count in a clockwise direction, the number of transitions from 0 to 1 (T01), 0 to 2 (T02), and 0 to 3 (T03) along the border of image M. If T01= 1 ^ T02= 1 ^ T03= 1, then the candidate minutiae point is validated as a true bifurcation.

In practice, some of the minutiae detected using the above approach may be spurious due to artifacts in image processing and noise in the fingerprint image. To remove these spurious minutiae, a minutiae filtering algorithm is employed, which typically consists of a number of heuristic rules. For instance, minutiae satisfying any of the following conditions are deemed to be spurious minutiae and discarded:

(i) Minutiae that do not have an adjacent ridge on either side (mainly the endpoints of ridges along the finger border);

(ii) Minutiae that are close in location and almost opposite in direction (namely, the difference between two minutiae directions is close to 180◦);

Too many minutiae in a small neighborhood.

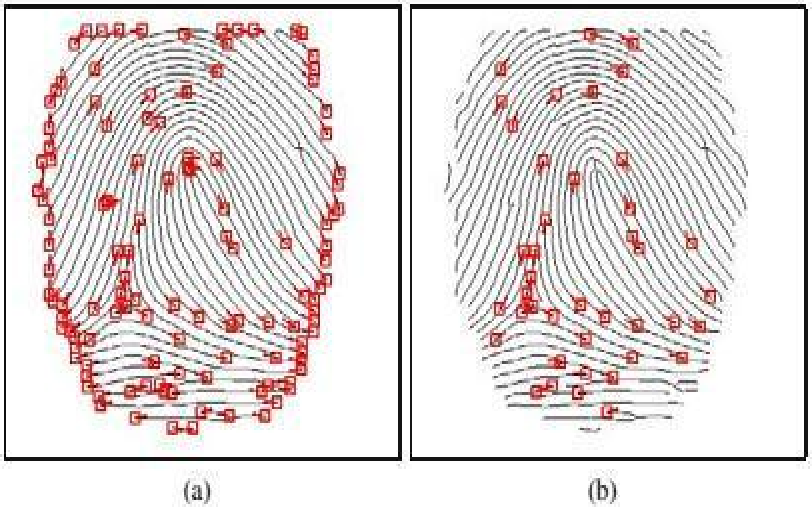


Fig 13. Removing spurious minutiae. (a) Before minutiae filtering and (b) after minutiae filtering.

Functions used are:

1. Main File:

img = imread('testcases/a.jpg');

imread reads the image from the file specified by filename, inferring the format of the file from its contents,here image name a.jpg is read from the folder testcases.

thinned = edge(RGB2GRAY(img), 'canny');

This function is used to detect the edges of the image and converting it from colour to gray scale image. It is basically technique for finding the boundaries of objects within images.it works by detecting discontinuities in brightness.it is used for image segmentation and data extraction.

Canny : It finds edges by finding local maxima of the gradient of I.

minu_count = 1;

the variable is set to 1,so that pixels can be calculated.

minutiae(minu_count, :) = [0,0,0,0];

Array of pixels is created.

minutiae(minu_count, :) = [x, y, CN, theta];

In this function values calculated whether the pixel is white or black ,angle calculated and crossing number are defined

minu_count = minu_count + 1;

counter is incremented.

```
minutiae_img = uint8(zeros(size(img, 1),size(img, 2), 3));
```

uint8 converts the elements of an array into unsigned 8-bit integers of class uint8,so that ridges and bifurcations can be detected.

```
 minutiae_img(x1, y1,:) = [255, 0, 0];
```

In this the ridge that is detected is colored red.

```
 minutiae_img(x1, y1,:) = [0, 0, 255];
```

using this bifurcations are colored blue.

```
combined = uint8(minutiae_img);
combined(x,y,:) = [255,255,255];
combined(x,y,:) = [0,0,0];
```

using these lines the images are combined and red colored ridges and blue colored bifurcations are created.

```
%imwrite(thinned,'testcases/a_thin.bmp');
%imwrite(minutiae_img,'testcases/a_minutiae.bmp');
   imwrite(combined,'match/a_combine.bmp');
```

imwrite(B,filename.fmt) function writes the image in B to filename in the format specified by fmt.A can be either a grayscale image or a truecolor image.filename specifies the name of the output file.

```
subplot(2,2,1), subimage(img), title('orignal image')
subplot(2,2,2), subimage(thinned), title('thinned image')
subplot(2,2,3), subimage(minutiae_img), title('minutiae points')
subplot(2,2,4), subimage(combined), title('thinned and minutiae points')
```

subplot is used to plot the images generated in a graph to show the output.

2. FindTheta :

This is a user defined function used to calculate the angle of the patterns.

function theta = FindTheta(img, x, y, CN)
in this the image object , x and y pixels detected and the crossing number is passed.
After that according to the conditions using switch case angle is detected.
Snippet below shows the result of the code using these functions.

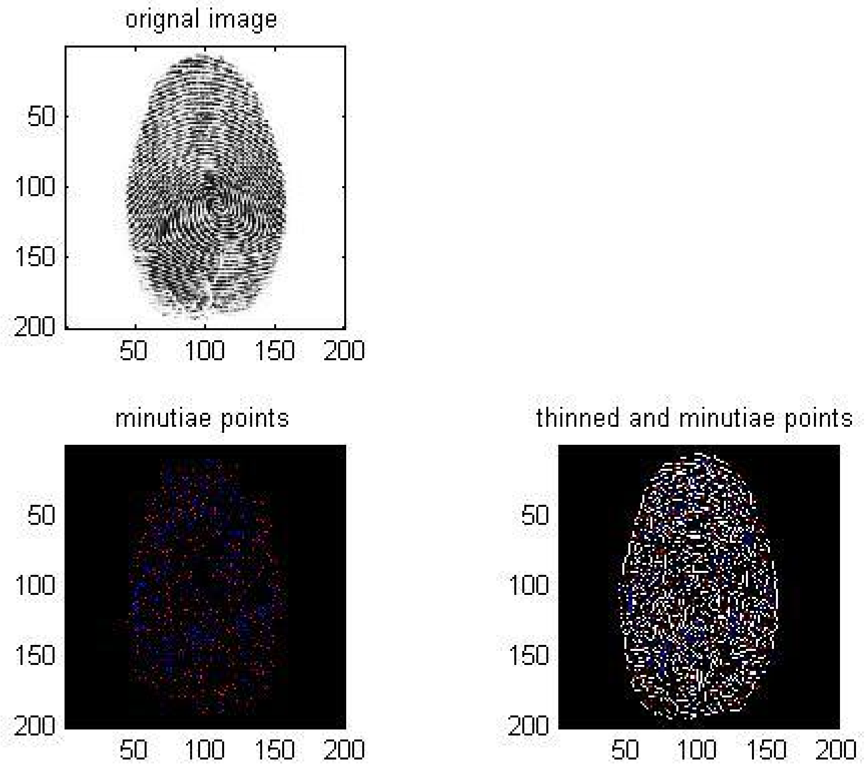


Fig 14. Minutiae points extracted from fingerprint image shown

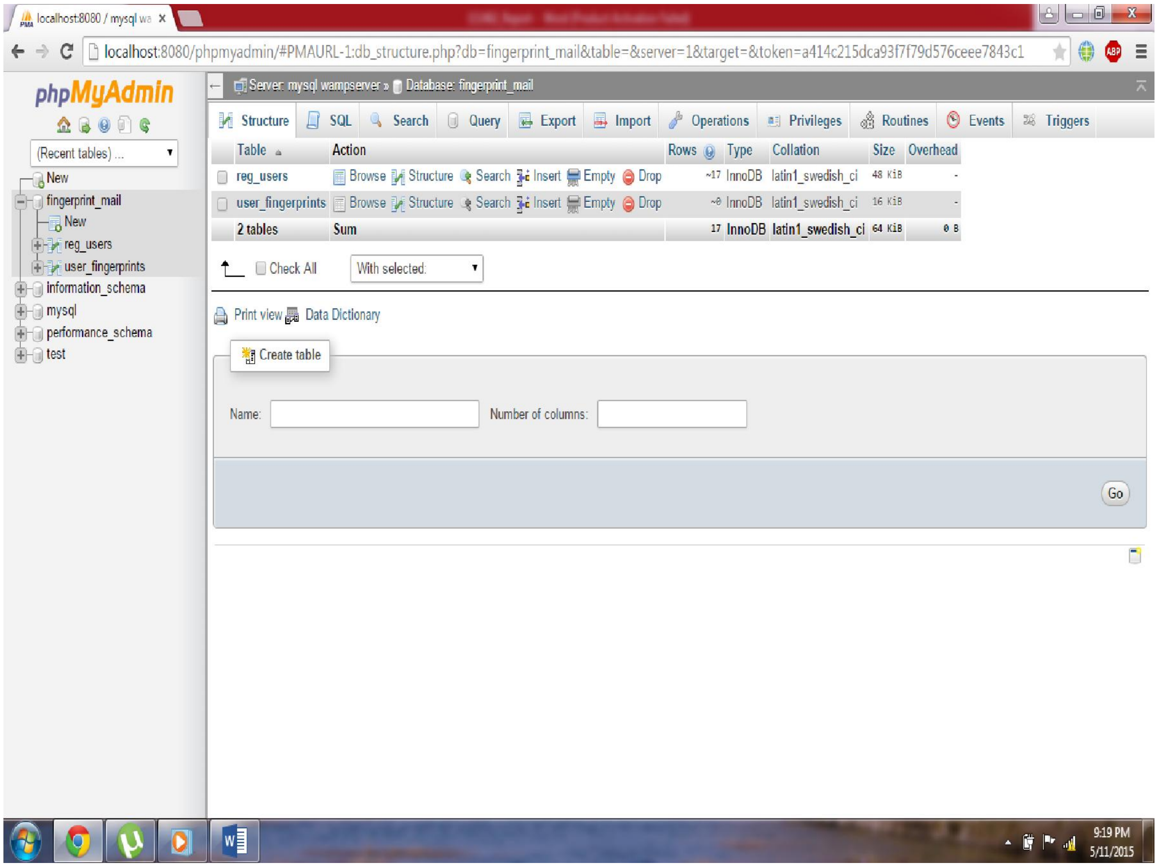
CHAPTER 5

DATABASE MODULE

The database acts as the repository of biometric information. During the enrollment process, the feature set extracted from the raw biometric sample (i.e., the template) is stored in the database (possibly) along with some biographic information (such as name, Personal Identification Number (PIN), address, etc.) characterizing the user. The data capture during the enrollment process may or may not be supervised by a human depending on the application.

For example, a user attempting to create a new computer account in her biometric enabled workstation may proceed to enroll her biometrics without any supervision; a person desiring to use a biometric-enabled ATM, on the other hand, will have to enroll her biometrics in the presence of a bank officer after presenting her non-biometric credentials.

The minutiae features extracted are stored in the form of blob variables in php myadmin along with person’s id and another alphanumeric password if user requires further layer of security.



localhost:8080 / mysql wa
localhost:8080/phpmyadmin/#PMAURL-1:db_structure.php?db=fingerprint_mail&table=&server=1&target=&token=a414c215dca93f7f79d576ceee7843c1
phpMyAdmin
(Recent tables) ...
New
fingerprint_mail
New
reg_users
user_fingerprints
information_schema
mysql
performance_schema
test
Server: mysql wampserver » Database: fingerprint_mail
Structure
SQL
Search
Query
Export
Import
Operations
Privileges
Routines
Events
Triggers

Table	Action	Rows	Type	Collation	Size	Overhead
□ reg_users	Browse Structure Search Insert Empty Drop	~17	InnoDB	latin1_swedish_ci	48 KiB	-
□ user_fingerprints	Browse Structure Search Insert Empty Drop	~0	InnoDB	latin1_swedish_ci	16 KiB	-
2 tables	Sum	17	InnoDB	latin1_swedish_ci	64 KiB	0 B

Fig 15. Database created

CHAPTER 6

MATCHING MODULE

Given the minutiae set of a query fingerprint with M minutiae and the minutiae set of a template fingerprint with N minutiae, i now describe a simple matching algorithm which consists of three steps:

Alignment: Determine the geometric transformation between the two minutiae sets so that they are in the same coordinate system.

Correspondence: Form pairs of corresponding minutiae.

Score generation: Compute the match score based on the corresponding minutiae points.

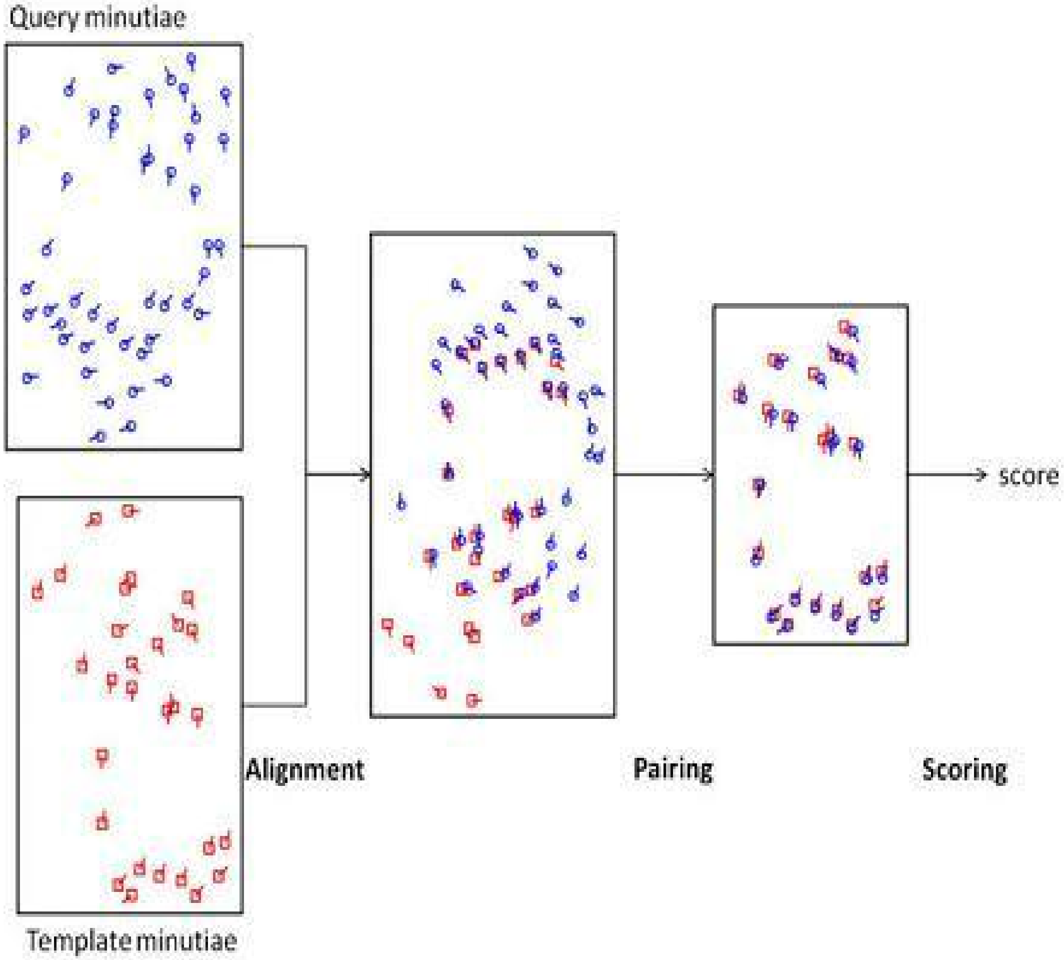


Fig 16. Flowchart of a minutiae matching algorithm.

6.1 Alignment

Since two impressions of the same finger taken at different instances could differ due to different placement of the finger on the sensor, an alignment process is required to transform them to the same coordinate system. This process, also known as registration, transforms one image in such a way that it is geometrically aligned with the other. First, we need to specify a spatial transformation model. Generally, a rigid transformation is sufficient for fingerprint matching unless a severe nonlinear deformation is introduced during fingerprint acquisition. Generalized Hough transform is a well-known algorithm for estimating the spatial transformation between two point sets. The pseudo-code of the Generalized Hough transform algorithm is given in Algorithm 1.

Another popular fingerprint alignment algorithm is to first find a pair of matched minutiae, and then compute the rotation and translation parameters between two fingerprints based on this pair of minutiae. Since the basic properties of minutiae, namely location, direction and type, do not contain sufficient information for determining matched minutiae, additional information in the neighborhood of minutiae needs to be associated with each minutia to increase its distinctiveness. This additional information is termed as a minutia descriptor [15]. A widely used minutia descriptor is based on the set of minutiae in the neighborhood of the central minutia (namely, the minutia whose descriptor needs to be computed). The location and direction of neighboring minutiae are defined in the local minutia coordinate system using the central minutia as the origin and its direction as the x axis.

This way, the descriptor is invariant with respect to the rigid transformation of fingerprints. The similarity between two minutiae descriptors is computed by (a) first a pairing of neighboring minutiae is established (using an algorithm similar to the algorithm described in the next subsection) and then (b) computing the product of the percentages of matched minutiae in the local region of each minutia .

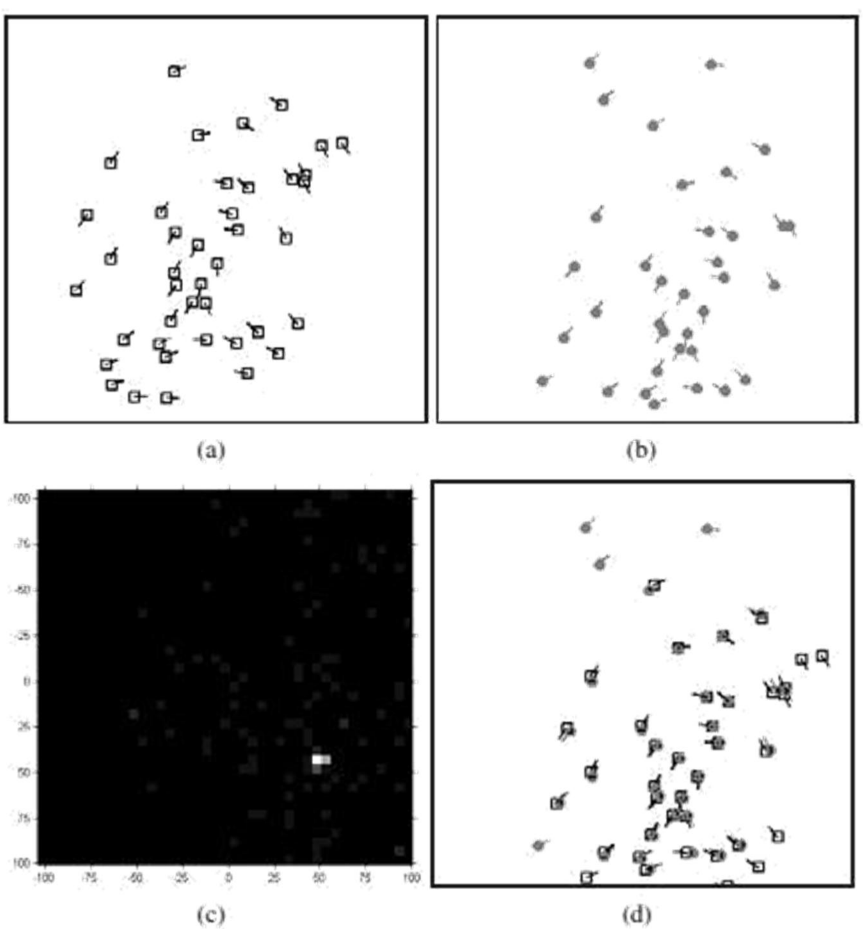


Fig. 17. Minutiae set alignment using the Hough transform. (a) query minutiae set, (b) template minutiae set, (c) accumulator array or the Hough space image, and (d) aligned minutiae sets. The "bright" spot in the Hough space in (c) indicates the cell that receives the most votes. Thexandy translation corresponding to this cell is used for aligning the two minutiae sets.

6.2 Pairing minutiae

After the two minutiae sets are aligned, the corresponding minutiae are paired. A Minutia in the template (reference) minutiae set is said to be in correspondence with minutia b in the query minutiae set if and only if their distance is within a predefined distance threshold (say, 15 pixels) and the angle between their directions is within another predefined angle threshold (say, 20 degrees). One minutia in the template fingerprint is allowed to match to at most one minutia in the query finger print and vice versa.

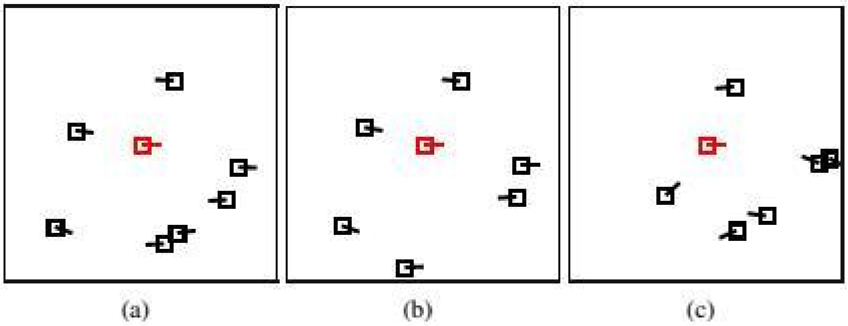
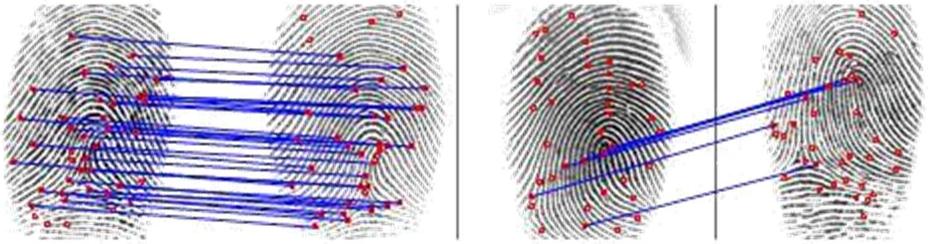


Fig. 18. Minutia descriptor. (a) Descriptor of a minutia (the central one) in a query fingerprint image, (b) descriptor of the mated minutia in a template fingerprint, and (c) descriptor of another minutia.

6.3 Match score generation

In this final step, we need to compute a match score, which is then compared to a predefined threshold to classify the two fingerprints as a genuine match or an impostor match (they come from two different fingers). This problem can be viewed as a two-class classification problem with genuine match as class-1 and impostor match as class-2. For this classification problem, several potential features for distinguishing genuine matches from impostor matches can be examined. The first feature is the number of paired minutiae. It is intuitive that genuine matches should have more paired minutiae than the impostor matches.

The second useful feature is the percentage of matched minutiae in the overlapped area between the two fingerprints. Again, it is intuitive that this percentage be larger for genuine matches than for impostor matches. Given a set of minutiae, the fingerprint area can be approximated by the convex hull of its minutiae points.



(a) Match score = 614

(b) Match score = 7

Fig. 19. Fingerprint matching by a commercial matcher. (a) A genuine pair of fingerprints with 31 matched minutiae, and (b) an imposter pair with 6 matched minutiae. Corresponding minutiae between the two images are connected by lines. The match score is computed as some function of the number of matched minutiae and some other parameters that are proprietary to the commercial matcher.

According to the match score user will be identified and will be able to use his\her mailing account.

A percent match score is generated and it is compared with a threshold of 80% and according to fingerprint user is accepted or rejected and the user get logged in or asked to try again.

Function used:

Main function:

```
img1 = imread('a_combine.bmp');
img2= imread('a.jpg');
```

two objects img1 and img2 are created using imread function which is used to read the images from the location,these are the fingerprints to match.

```
percent_match=validtest(img1,img2);
```

percent_match variable is used to store the match score calculated by the function validtest.

Validtest:

```
function [percent_match]=validtest(pic1,pic2)
```

it is a user defined function to calculate the percentage of match score between two images.

```
[x,y,z] = size(pic1);
```

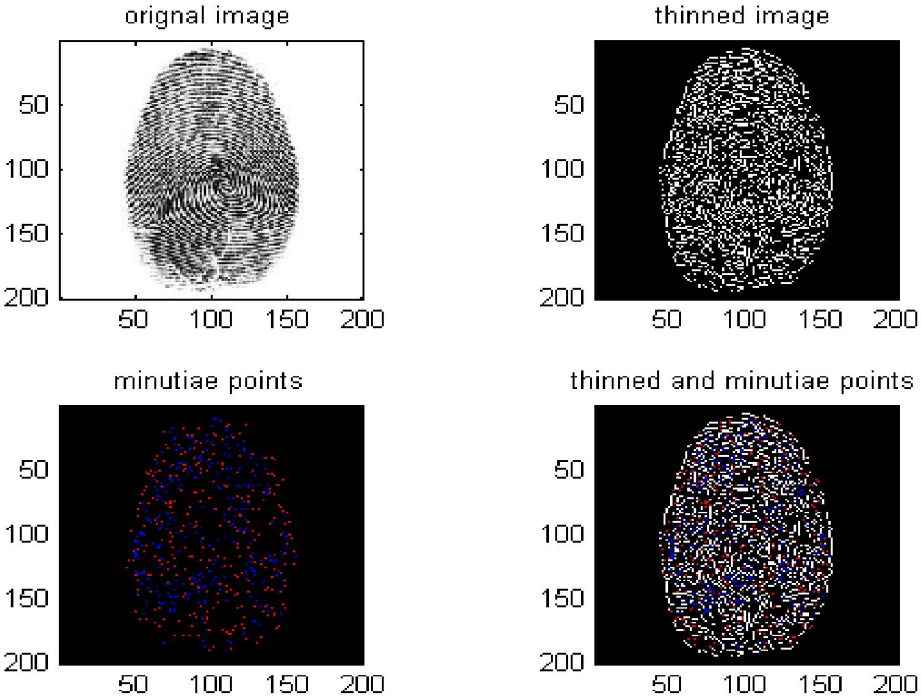
Size function gives the two element row vector containing the number of rows and columns in the object(pic 1).

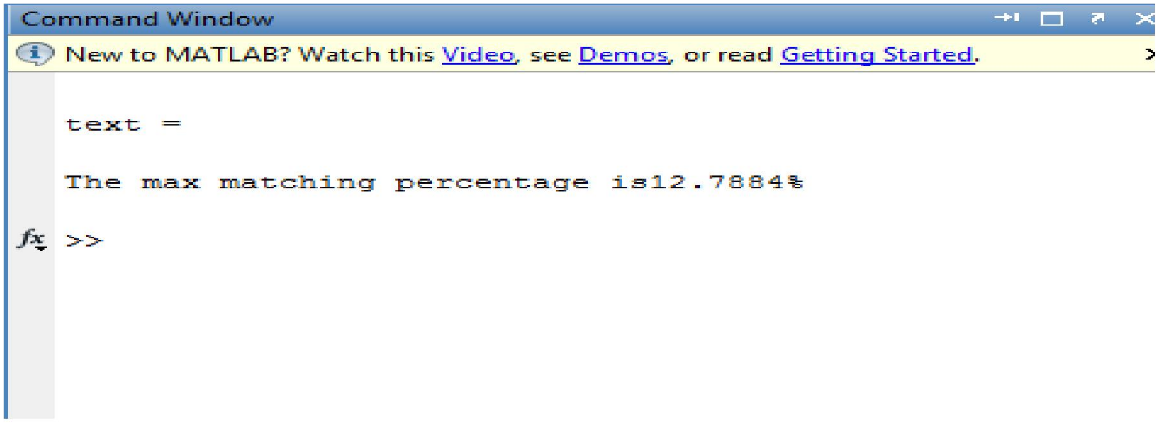
```
pic1 = rgb2gray(pic1);
pic2 = rgb2gray(pic2);
```

rgb2gray converts the truecolor image RGB to the gray scale intensity image by eliminating the hue and saturation information while retaining the luminance.

```
edge_det_pic1 = edge(pic1,'prewitt');
edge_det_pic2 = edge(pic2,'prewitt');
```

edge function is used to detect the edges of the objects ,in this using prewitt algorithm. Prewitt algorithm finds edges using the prewitt approximation to the derivative,it return edges at those points where the gradient of I is maximum.





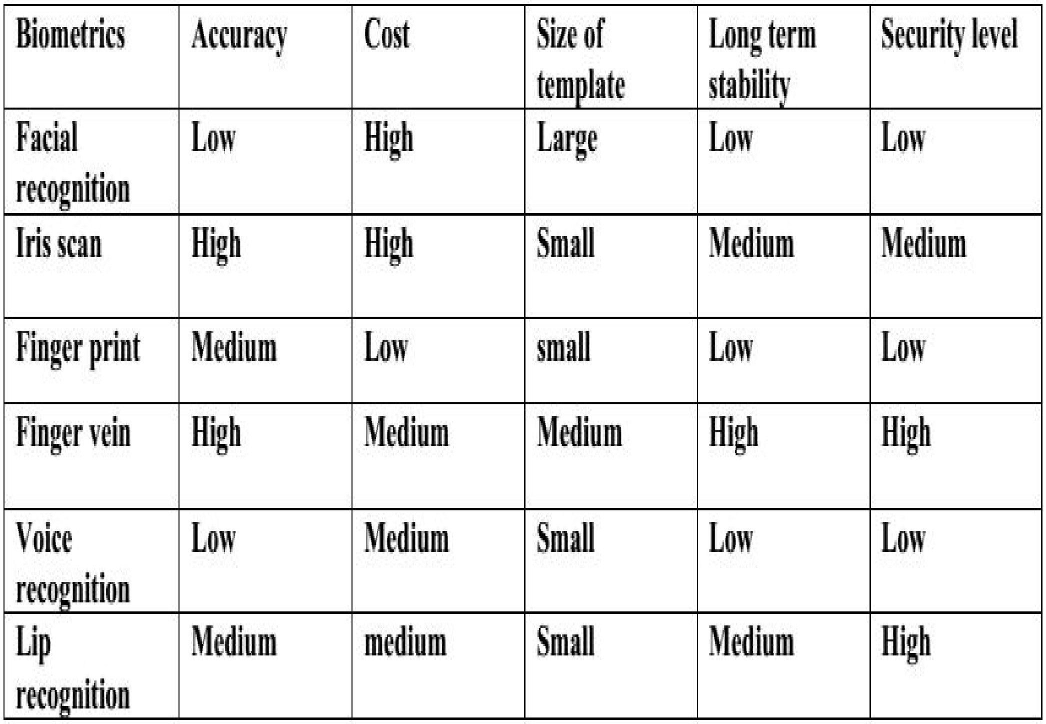
New to MATLAB? Watch this Video, see Demos, or read Getting Started.

```
text =

The max matching percentage is12.7884%

>>
```

These snippets shows the result of extracting minutiae and generating the match score.

CHAPTER 7

WHY FINGERPRINTS?



Biometrics	Accuracy	Cost	Size of template	Long term stability	Security level
Facial recognition	Low	High	Large	Low	Low
Iris scan	High	High	Small	Medium	Medium
Finger print	Medium	Low	small	Low	Low
Finger vein	High	Medium	Medium	High	High
Voice recognition	Low	Medium	Small	Low	Low
Lip recognition	Medium	medium	Small	Medium	High

Table 2: Comparison of different biometrics

Fingerprint identification has a number of advantages which make it a popular method of identification in settings ranging from police stations to secured facilities. This method of identification is accomplished by comparing fingerprints from someone against a database of known fingerprints. If the sample fingerprints match fingerprints in the database, it is considered a positive match. It is important to note that many identification systems which use fingerprints go for a statistically significant match; rather than matching the whole fingerprint, they look for key markers which can be used for comparison.

One big advantage of fingerprint identification is that it is very well accepted in the legal community, among law enforcement, and the general public. It has a long history in forensic science, complete with numerous studies which back up the use of fingerprints for identification. This venerable history gives it weight and credence which are not available to newer identification systems. In addition, fingerprint identification is widely perceived as highly accurate and very reliable, since the statistical chance of two people on Earth having identical fingerprints is very low.

The accuracy factor is important, as mistaken identity is hard to do when fingerprints are collected and studied properly. However, this can also be a pitfall; because people rely so heavily on fingerprint identification, if fingerprint evidence is not collected, stored, or handled properly it may result in a false identification which people will believe is valid because they view fingerprinting as highly reliable.

Fingerprint is the cheapest, fastest, most convenient and most reliable way to identify someone. That's why fingerprint alone has 2/3 of the biometric world market (according to an International Biometric Group independent report). And the tendency, due to scale, easiness and the existing foundation, is that the use of fingerprint will only increase. Cars, cell phones, PDAs, personal computers and dozens of products and devices are using fingerprints more and more.

When compared with the conventional authentication methods that are based on “what only the person possesses” or “what only the person knows,” biometrics authentication offers two distinctive advantages:

Enhanced convenience: By merely presenting his biometric features, a user can easily prove himself or herself. There are no troubles such that authorized users are denied access because of loss of a card or forgetting a password.

Augmented security: The reliable rejection of impostors, who might attempt to gain access either by stealing or forging cards or by guessing or fraudulently obtaining passwords, becomes possible.

CHAPTER 8

MAILING SITE

The aim is to implement the necessary functionalities to the users such as receiving and organizing mails through SMTP and PHP MAIL FUNCTION and Sending mails.

All these are provided with the graphical interface so that the users can do their desired work smoothly. Apart from that, the administrator can create and delete user accounts through the server. The system starts with the welcome page prompting the user to establish his authentication. He is allowed to see his mail boxes, read mails and send mails to the other users only after his/her account is verified using his\her fingerprints .

Different functionalities of site:

Login page: There should be a login page for the existing user where the username and password are verified and then if he is a valid user, he is allowed for further advancements.

Inbox: The logged in users should be able to see the lists of new mails as well as the existing ones.

Compose Mail: User should be able to compose mails and send them to the other

Users.

Reply/Forward/delete: The user should be able to reply to mails, forward mails and also delete mails from his mailboxes. The deleted mails should be moved to the Thrash mailbox.

What is Simple Mail Transfer Protocol (SMTP)?

SMTP stands for Simple Mail Transfer Protocol. It's a set of communication guidelines that allow software to transmit email over the Internet. Most email software is designed to use SMTP for communication purposes when sending email, and It only works for outgoing messages. When people set up their email programs, they will typically have to give the address of their Internet service provider's SMTP server for outgoing mail. There are two other protocols - POP3 and IMAP - that are used for retrieving and storing email.

SMTP provides a set of codes that simplify the communication of email messages between servers. It's a kind of shorthand that allows a server to break up different parts of a message into categories the other server can understand. Any email message has a sender, a recipient - or sometimes multiple recipients - a message body, and usually a title heading. From the perspective of users, when they write an email message, they see the slick interface of their email software, but once that message goes out on the Internet, everything is turned into strings of text. This text is separated by code words or numbers that identify the purpose of each section. SMTP provides those codes, and email server software is designed to understand what they mean.

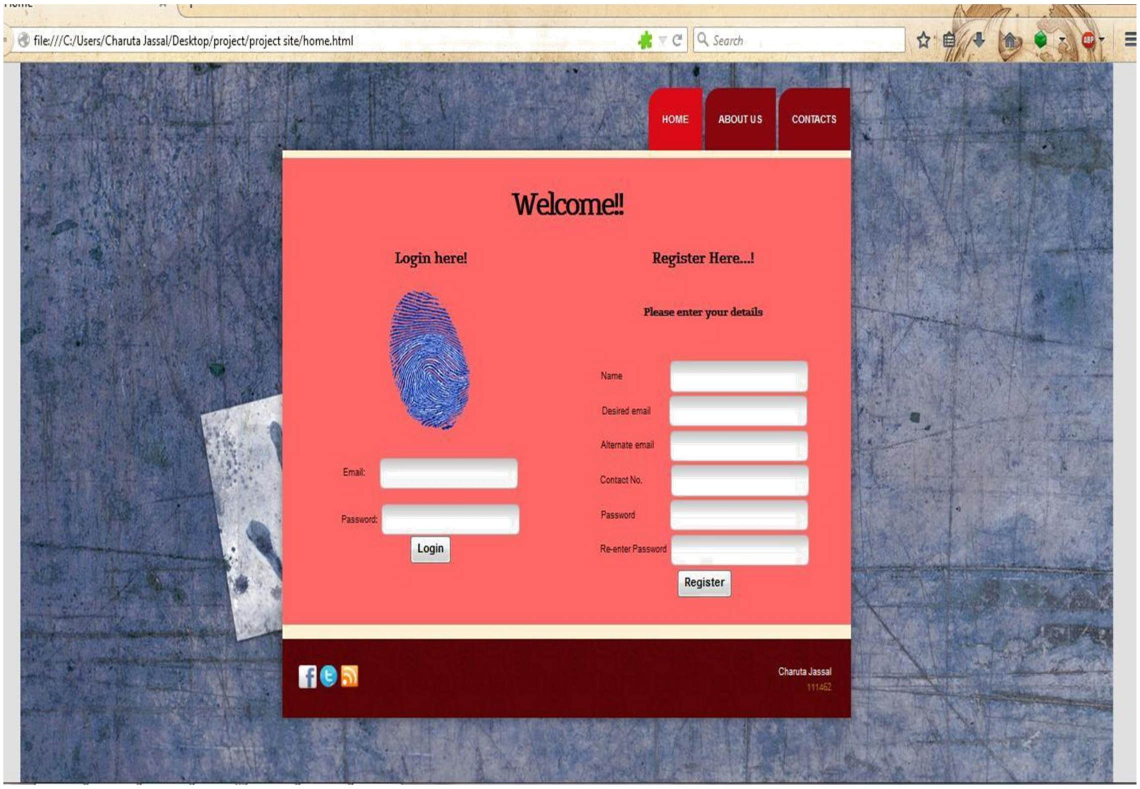
The other purpose of SMTP is to set up communication rules between servers. For example, servers have a way of identifying themselves and announcing what kind of communication they are trying to perform. There are also ways to handle errors, including common things like incorrect email addresses. In a typical SMTP transaction, a server will identify itself, and announce the kind of operation it is trying to perform. The other server will authorize the operation, and the message will be sent. If the recipient address is wrong, or if there is some other problem, the receiving server may reply with an error message of some kind.

To receive e-mail, you need an account on a mail server. This is similar to having a postal box where you receive letters. One advantage over regular mail is that you can retrieve your e-mail from any location on earth, provide that you have Internet access. Once you connect to your mail server, you either download your messages to your computer or wireless device, or use your web browser to read them online.

To send e-mail, you need a connection to the Internet and access to a mail server that forwards your mail to its final destination. The standard protocol used for sending Internet e-mail is called SMTP, short for Simple Mail Transfer Protocol. It works in conjunction with POP--Post Office Protocol--servers. Almost all Internet service providers and all major online services offer at least one e-mail address with every account.

When you send an e-mail message, your computer routes it to an SMTP server. The server looks at the e-mail address (similar to the address on an envelope), then forwards it to the recipient's mail server, where it's stored until the addressee retrieves it. You can send e-mail anywhere in the world to anyone who has an e-mail address. In fact astronauts on the international space station use e-mail to keep in touch with their earth-bound colleagues.

At one time, you could only send text messages without attachments via the Internet. With the advent of MIME, which stands for Multipurpose Internet Mail Extension, and other types of encoding schemes, such as UUencode, you can now send formatted documents, photos, audio and video files. Just make sure that the person to whom you send the attachment has the software capable of opening it.


file:///C:/Users/Charuta Jassal/Desktop/project/project site/home.html
Search
HOME
ABOUT US
CONTACTS

Welcome!!

Login here!

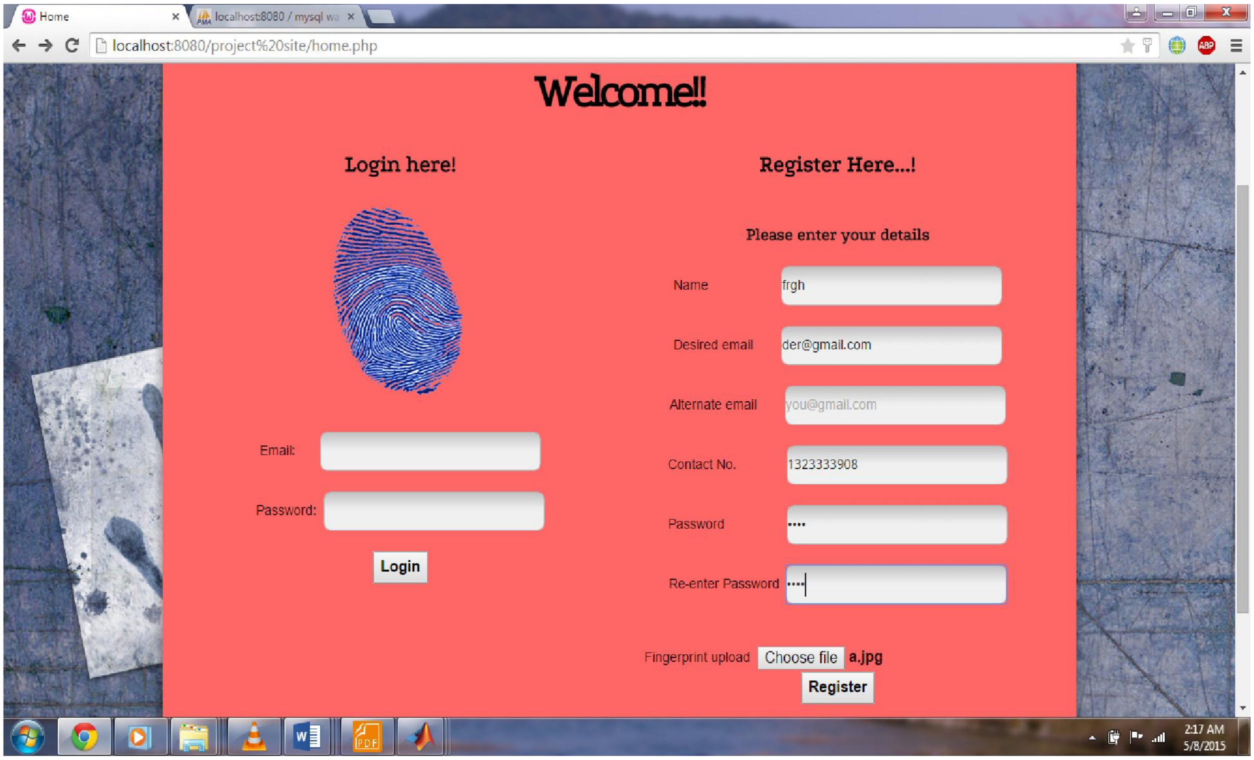
Register Here...!

Please enter your details

Charuta Jassal
111462

The homepage of the website is shown.

In his webpage there are two divisons one is the background and other one is tha panel containing two subdivisions. One form is for the registered users to login into their accounts and other form is for the new users to provide necessary details and get registered.



Home
localhost:8080 / mysql wa
localhost:8080/project%20site/home.php
2:17 AM
5/8/2015

Welcome!!

Login here!

Email:

Password:

Login

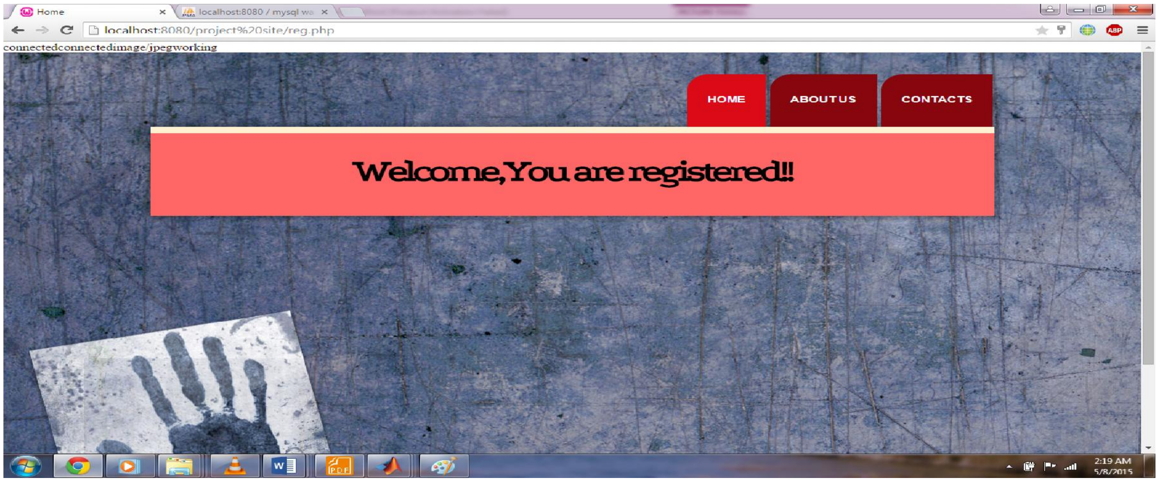
Register Here...!

Please enter your details

Name	frgh
Desired email	der@gmail.com
Alternate email	you@gmail.com
Contact No.	1323333908
Password	••••
Re-enter Password	••••
Fingerprint upload	Choose file a.jpg

Register

Webpage showing the details being entered of a new user , so that all the details are stored in the database.

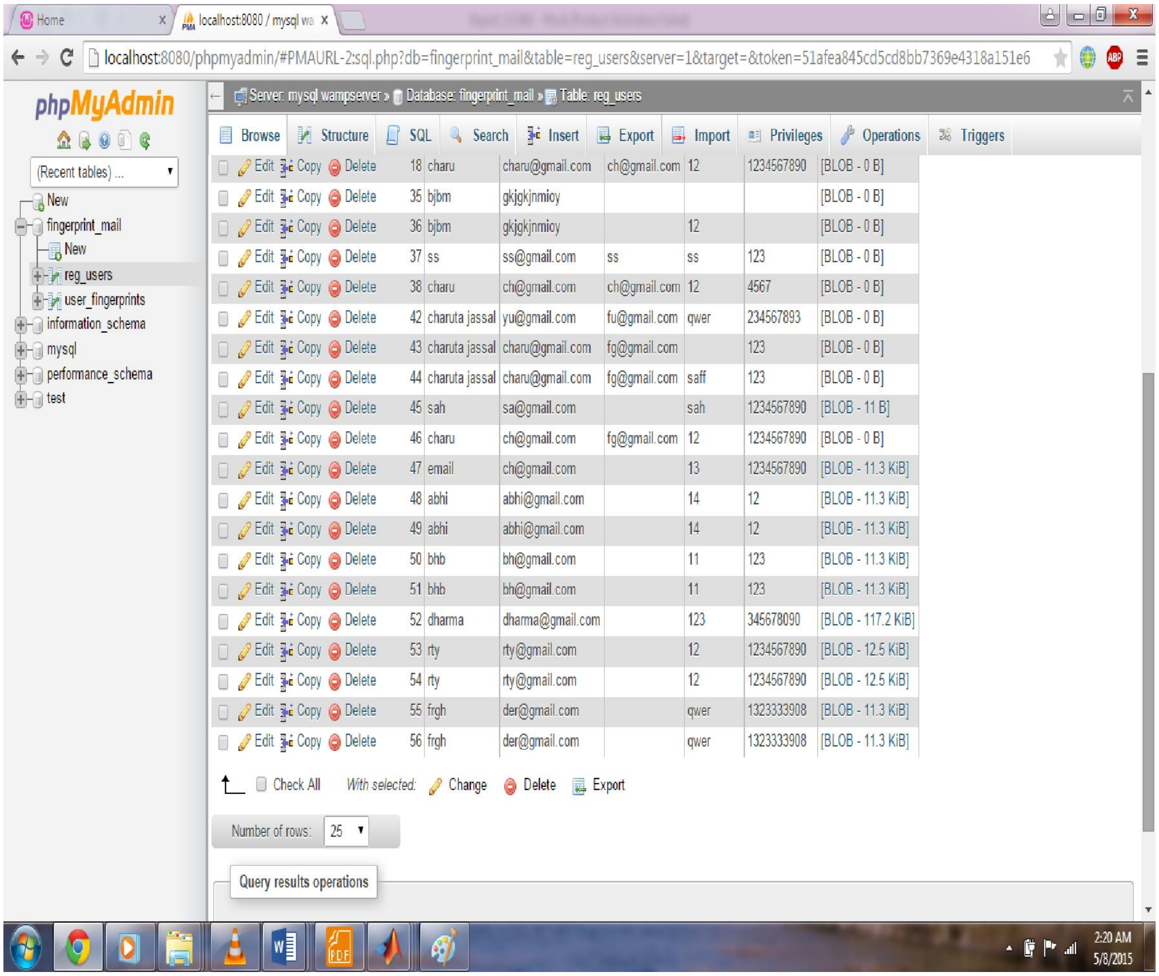


Home
localhost:8080 / mysql wa
localhost:8080/project%20site/reg.php
HOME
ABOUTUS
CONTACTS
2:19 AM
5/8/2015

connectedconnectedimage/jpegworking

Welcome,You are registered!!

After giving the correct information a success webpage is shown dscribing the user has been added.



localhost:8080/phpmyadmin/#PMAURL-2:sql.php?db=fingerprint_mail&table=reg_users&server=1&target=&token=51afea845cd5cd8bb7369e4318a151e6

Server: mysql wampserver » Database: fingerprint_mail » Table: reg_users

□	Edit	Copy	Delete	18	charu	charu@gmail.com	ch@gmail.com	12	1234567890	[BLOB - 0 B]
□	Edit	Copy	Delete	35	bjbm	gkjgkjnmioy				[BLOB - 0 B]
□	Edit	Copy	Delete	36	bjbm	gkjgkjnmioy		12		[BLOB - 0 B]
□	Edit	Copy	Delete	37	ss	ss@gmail.com	ss	ss	123	[BLOB - 0 B]
□	Edit	Copy	Delete	38	charu	ch@gmail.com	ch@gmail.com	12	4567	[BLOB - 0 B]
□	Edit	Copy	Delete	42	charuta jassal	yu@gmail.com	fu@gmail.com	qwer	234567893	[BLOB - 0 B]
□	Edit	Copy	Delete	43	charuta jassal	charu@gmail.com	fg@gmail.com		123	[BLOB - 0 B]
□	Edit	Copy	Delete	44	charuta jassal	charu@gmail.com	fg@gmail.com	saff	123	[BLOB - 0 B]
□	Edit	Copy	Delete	45	sah	sa@gmail.com		sah	1234567890	[BLOB - 11 B]
□	Edit	Copy	Delete	46	charu	ch@gmail.com	fg@gmail.com	12	1234567890	[BLOB - 0 B]
□	Edit	Copy	Delete	47	email	ch@gmail.com		13	1234567890	[BLOB - 11.3 KiB]
□	Edit	Copy	Delete	48	abhi	abhi@gmail.com		14	12	[BLOB - 11.3 KiB]
□	Edit	Copy	Delete	49	abhi	abhi@gmail.com		14	12	[BLOB - 11.3 KiB]
□	Edit	Copy	Delete	50	bhb	bh@gmail.com		11	123	[BLOB - 11.3 KiB]
□	Edit	Copy	Delete	51	bhb	bh@gmail.com		11	123	[BLOB - 11.3 KiB]
□	Edit	Copy	Delete	52	dharma	dharma@gmail.com		123	345678090	[BLOB - 117.2 KiB]
□	Edit	Copy	Delete	53	rty	rty@gmail.com		12	1234567890	[BLOB - 12.5 KiB]
□	Edit	Copy	Delete	54	rty	rty@gmail.com		12	1234567890	[BLOB - 12.5 KiB]
□	Edit	Copy	Delete	55	frgh	der@gmail.com		qwer	1323333908	[BLOB - 11.3 KiB]
□	Edit	Copy	Delete	56	frgh	der@gmail.com		qwer	1323333908	[BLOB - 11.3 KiB]

Database is created using MySQL and all the information entered by the new user is stored here.

Validations in registration:

```
var n=document.forms["myform"]["name"].value;
var de=document.forms["myform"]["demail"].value;
var ae=document.forms["myform"]["aemail"].value;
var c=document.forms["myform"]["contact"].value;
var reg=document.forms["myform"]["regpass"].value;
var re=document.forms["myform"]["repass"].value;
var atpos=de.indexOf("@");
var dotpos=de.lastIndexOf(".");
```

all the values from the form are extracted and stored in the variables name.

```
if (n=="") //|| de=="" || ae=="" || c=="" || reg=="" || re="")
  {
  alert("Name must be filled!");
  return false;
  }
```

If any field is empty its going to show an box alerting field should be filled.

```
if (atpos<1 || dotpos<atpos+2 || dotpos+2>=x.length)
  {alert("Not a valid e-mail address");
  return false;
  }
```

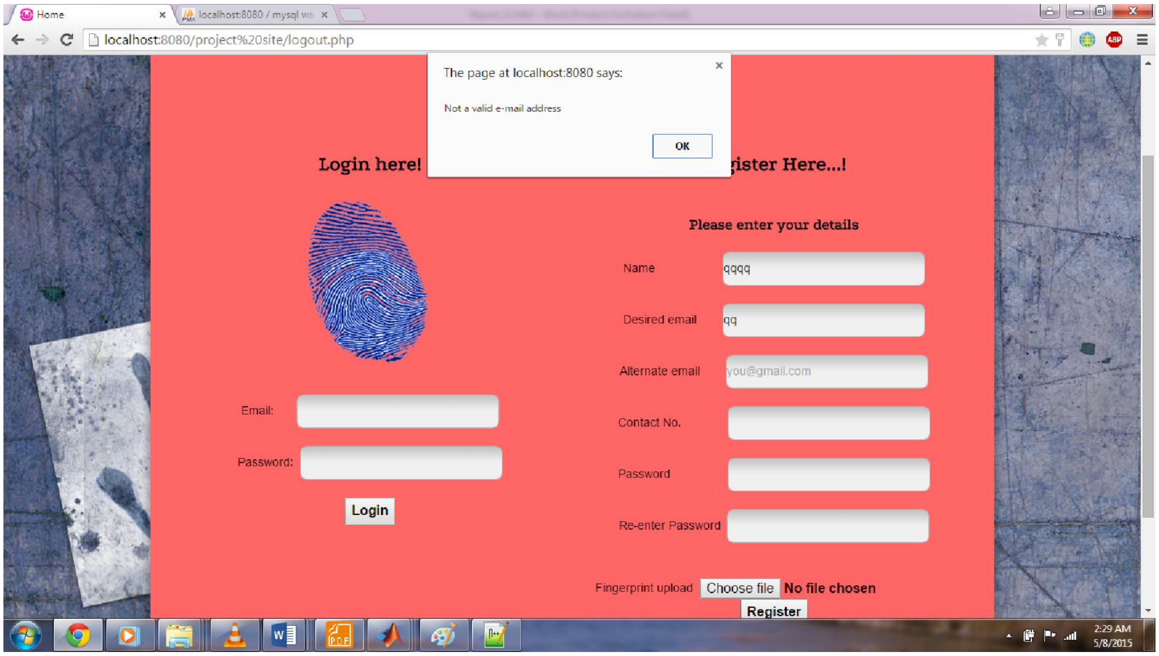
These lines are checking if an email id entered is valid or not and hence if not an alert box is displayed.

```
 if (c=="")
if(c.toString().length()!=10)
```

these lines are checking whether the contact number is of 10 digits or not. If the field is filled or epmpty is also checked.

```
if (reg!=re )
  {
  alert("Re-entered password doesnot match!");
  return false;
  }
```

This is used to check if both the fields enter password and re enter password contains the same string , otherwise an error box is shown.



localhost:8080/project%20site/logout.php

The page at localhost:8080 says:

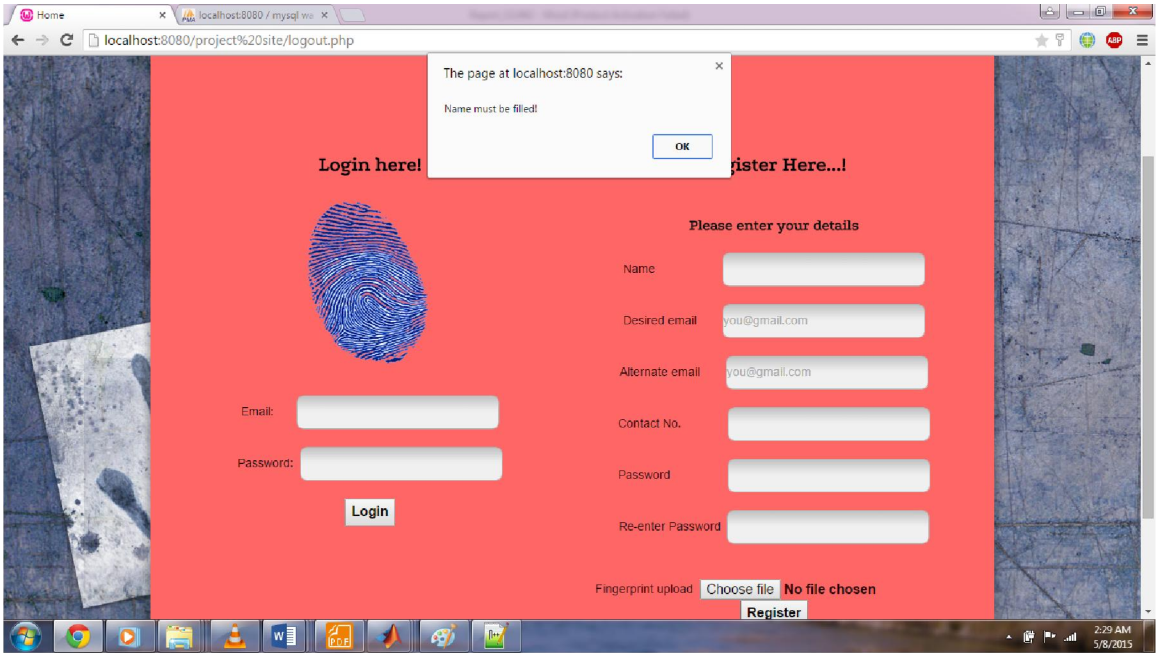
Not a valid e-mail address

Name: qqqq

Desired email: qq

Alternate email: you@gmail.com

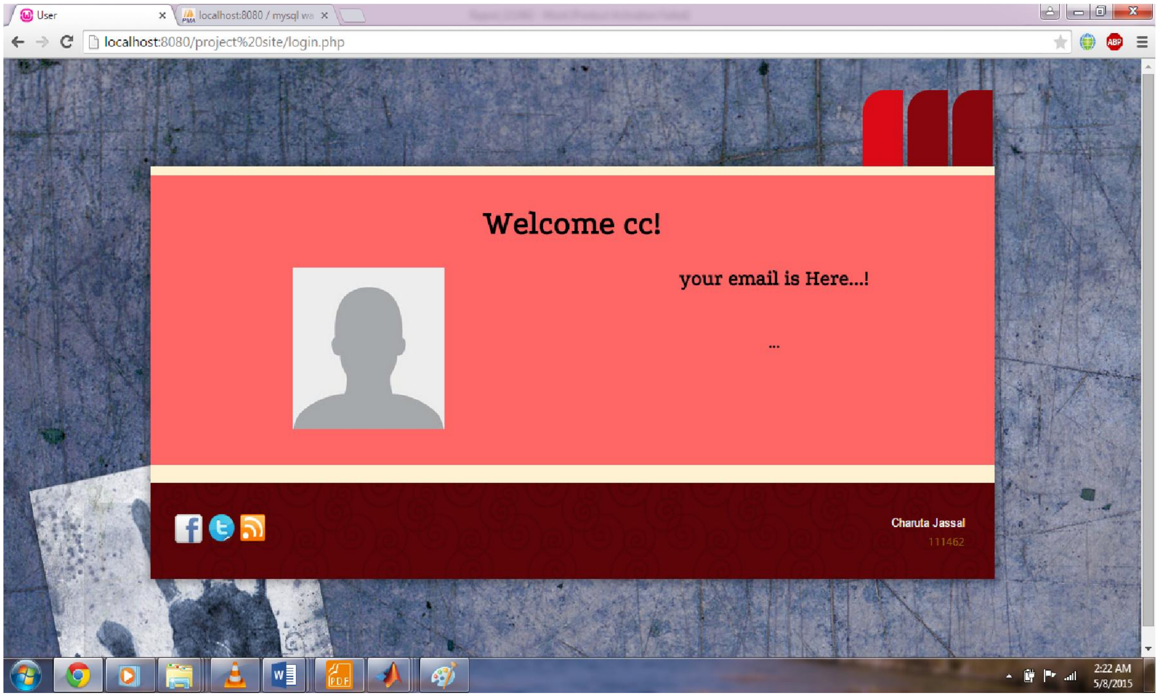
Fingerprint upload: No file chosen



The page at localhost:8080 says:

Name must be filled!

OK


User
localhost:8080 / mysql we
localhost:8080/project%20site/login.php
Charuta Jassal
111462
2:22 AM
5/8/2015

Welcome cc!

your email is Here...!

...

After correctly registering, user can login and a welcome page is shown.

In my website user can register in the form designed by providing necessary details name, desired email id, alternate email id, password, contact number and uploading the fingerprint image in the any format .jpeg, .png, .bmp, etc.

Various validations are implied in JavaScript so that no field is empty and email is in correct form and contact number should be of ten digits.

If all these conditions are verified then the information of the user are stored in the database created using MySQL php myadmin and the tables are created in the database ,and the information is stored in the tabular form. Images are stored in blob type.

In case of error then error page is shown warning the user about the error occurred.
If the user successfully enters his/her information then a page showing registration successful is displayed.

At the time of login user can then enter his/her registered email id and password and the fingerprint image.

Then the program matches the image entered at the login time and the one stored in database are matched and according to the matching percentage the user is authorized and logged in to the site.

CHAPTER 9

CONCLUSION AND FUTURE WORK

I have surveyed key technologies about fingerprint identification and have described in detail the working of fingerprint-based systems, the most widely-employed of all systems based on biometrics technology. I have also illustrated some actual systems based on these technologies in use, and presented some new activities that are taking place internationally. I have also briefly discussed the enhanced user interface which takes advantage of fingerprint identification technology to broaden the scope of its potential real-world application. Fingerprint as a password is a good layer of security added to the mailing accounts which will reduce the attacks of intruders. In future intend to add mailing feature also.

CHAPTER 10

BIBLOGRAPGHY

1. Anil k. Jain, Arun A. Ross, Karthik NandaKumar, “Introduction to Biometrics”,3rd edition,Springer,311

2. Jianbo Shi and Jitendra Malik (2000): "Normalized Cuts and Image Segmentation", IEEE Transactions on pattern analysis and machine intelligence, Vol. 22, No. 8 pp 888-905.

3. Davide Maltoni, Dario Maio, Anil K. Jain, Salil Prabhakar; “Handbook of Fingerprint Recognition”, second Edition, Springer-Verlag London Limited, 2009.

4. N. Ratha, S. Chen and A.K. Jain, "Adaptive Flow Orientation Based Feature Extraction in Fingerprint Images", Pattern Recognition, , November 2010,Vol. 28, pp. 1657-1672.

5. Prateek Verma,Yogesh Bahendwar,Amrita Sahu,Maheedhar Dubey,”Feature Extraction Algorithm of Fingerprint Recognition”, International Journal of Advanced Research in Computer Science and Software Engineering, October 2012,Vol. 10,6.

6. Gayathri S and Dr V Sridhar, “An Improved Fast Thinning Algorithm for Fingerprint Image”, International Journal of Engineering Science and Innovative Technology, Issue 1- January 2013,Vol. 2.