FINGERPRINT VERIFICATION SYSTEM

A PROJECT

Submitted in partial fulfillment of the requirements for the award of the degree of

BACHELOR OF TECHNOLOGY
IN

COMPUTER SCIENCE ENGINEERING

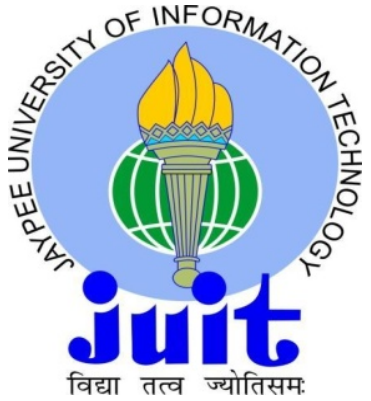
Under the supervision of

Dr. Amit Kumar Singh

By

Disha Gupta(121293)

To



JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY
WAKNAGHAT SOLAN – 173 234
HIMACHAL PRADESH INDIA

CERTIFICATE

This is to certify that the work which is being presented in the project title **"Fingerprint Verification System"** in fulfillment of the requirements for the award of the degree of Bachelor of technology and submitted in Computer Science & Engineering Department, Jaypee University of Information Technology, Waknaghat is an authentic record of work carried out by Disha Gupta during a period from August 2015 to May 2016 under the supervision of Dr. Amit Kumar Singh, Computer Science Engineering Department, Jaypee University of Information Technology, Waknaghat.

The above statement made is correct to the best of my knowledge.

Date: - ……

Dr. S P Grera
Professor and Head of Department
Computer Science Engineering
JUIT Waknaghat

Dr. Amit Kumar Singh
Assistant Professor
Computer Science Engineering
JUIT Waknaghat

ACKNOWLEDGEMENT

It is our proud privilege to epitomize our deepest sense of gratitude and indebtedness to our guide, Dr. Amit Kumar Singh, for his valuable instructions, guidance and support throughout our project work. His inspiring assistance and affectionate care enabled us to complete our work smoothly and successfully. This report is a dedicated contribution towards that greater goal.

Date :

Disha Gupta
(121293)

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

Biometric Recognition System: An Introduction

Personal identification is to associate a particular individual with an identity. It plays a critical role in our society, in which questions related to identity of an individual such as “Is this the person who he or she claims to be?”, “Has this applicant been here before?”, “Should this individual be given access to our system?” “Does this employee have authorization to perform this transaction?” etc are asked millions of times every day by hundreds of thousands of organizations in financial services, health care, electronic commerce, telecommunication, government, etc. With the rapid evolution of information technology, people are becoming even more and more electronically connected. As a result, the ability to achieve highly accurate automatic personal identification is becoming more critical.

Traditionally, passwords (knowledge-based security) and ID cards (token-based security) have been used to restrict access to systems. The major advantages of this traditional personal identification are that

(i) They are very simple

(ii)They can be easily integrated into different systems with a low cost.

However these approaches are not based on any inherent attributes of an individual to make a personal identification thus having number of disadvantages like tokens may be lost, stolen, forgotten, or misplaced; PIN may be forgotten or guessed by impostors. Security can be easily breached in these systems when a password is divulged to an unauthorized user or a card is stolen by an impostor. In the world of computer security, biometrics refers to authentication techniques that rely on measurable physiological and individual characteristics that can be automatically verified. In other words, we all have unique personal attributes that can be used for distinctive identification purposes, including a fingerprint, the pattern of a retina, and voice characteristics. Strong or two-factor authentication—identifying oneself by two of the three methods of something you know (for example, a password), have (for example, a swipe card), or is (for example, a fingerprint)—is becoming more of a genuine standard in secure computing environments. Some personal computers today can include a fingerprint scanner where you place your index finger to provide authentication. The computer analyzes your fingerprint to determine

who you are and, based on your identity followed by a pass code or pass phrase, allows you different levels of access. Access levels can include the ability to open sensitive files, to use credit card information to make electronic purchases, and so on.

Figure 1: Example of Fingerprint

1.1 CHARACTERISTICS OF BIOMETRIC

These are the important factors necessary for any effective biometric system: accuracy, speed and throughput rate, acceptability to users, uniqueness of the biometric organ and action, resistance to counterfeiting, reliability, data storage requirements, enrollment time, intrusiveness of data collection, and subject and system contact requirements.

1.1.1 Accuracy

Accuracy is the most critical characteristic of a biometric identifying verification system. If the system cannot accurately separate authentic persons from impostors, it should not even be termed a biometric identification system.

1.1.2 False Reject Rate

The rate, generally stated as a percentage, at which authentic, enrolled persons are rejected as unidentified or unverified persons by a biometric system is termed the false reject rate. False rejection is sometimes called a Type I error. In access control, if the requirement is to keep the “bad guys” out, false rejection is considered the least important error. However, in other

biometric applications, it may be the most important error. When used by a bank or retail store to authenticate customer identity and account balance, false rejection means that the transaction or sale (and associated profit) is lost, and the customer becomes upset. Most bankers and retailers are willing to allow a few false accepts as long as there are no false rejects. False rejections also have a negative effect on throughput, frustrations, and unimpeded operations, because they cause unnecessary delays in personnel movements.

1.1.3 False Accept Rate

The rate, generally stated as a percentage, at which unenrolled or impostor persons are accepted as authentic, enrolled persons by a biometric system is termed the false accept rate. False acceptance is sometimes called a Type II error. This is usually considered to be the most important error for a biometric access control system.

1.1.4 Crossover Error Rate (CER)

This is also called the equal error rate and is the point, generally stated as a percentage, at which the false rejection rate and the false acceptance rate are equal. All biometric systems have sensitivity adjustment capability. If false acceptance is not desired, the system can be set to require (nearly) perfect matches of enrollment data and input data. If tested in this configuration, the system can truthfully be stated to achieve a (near) zero false accept rate. If false rejection is not desired, this system can be readjusted to accept input data that only approximate a match with enrollment data. If tested in this configuration, the system can be truthfully stated to achieve a (near) zero false rejection rate. However, the reality is that biometric systems can operate on only one sensitivity setting at a time. The crossover error rate (CER) provides a single measurement that is fair and impartial in comparing the performance of the various systems. In general, the sensitivity setting that produces the equal error will be close to the setting that will be optimal for field operation of the system. A biometric system that delivers a CER of 2% will be more accurate than a system with a CER of 5%.

1.1.5 Speed and Throughput Rate

The speed and throughput rate are the most important biometric system characteristics. Speed is often related to the data processing capability of the system and is stated as how fast the accept

or reject decision is annunciated. In actuality, it relates to the entire authentication procedure: stepping up to the system; inputting the card or PIN (if a verification system); input of the physical data by inserting a hand or finger, aligning an eye, speaking access words, or signing a name; processing and matching of data files; annunciation of the accept or reject decision; and, if a portal system, movement through and closing the door.

Generally accepted standards include a system speed of 5 seconds from startup through decision annunciation. Another standard is a portal throughput rate of 6 to 10/minute, which equates to 6 to 10 seconds/person through the door.

1.1.6 Acceptability to Users

System acceptability to the people who must use it has been a little noticed but increasingly important factor in biometric identification operations. Biometric system acceptance occurs when those who must use the system — organizational managers and any union present — all agree that there are assets that need protection, the biometric system effectively controls access to these assets, system usage is not hazardous to the health of the users, system usage does not inordinately impede personnel movement and cause production delays, and the system does not enable management to collect personal or health information about the users.

1.2 APPLICATIONS

The applications of Biometric Verification is shown in the table below. It is categorized into three main areas.

Table 1: Applications of Biometric Verification

Forensic	Government	Commercial
Corpse Identification	National Id Card	Computer Network Logon
Criminal Investigation	Correctional Facility	Electronic Data Security
Terrorist Investigation	Driver’s License	E-Commerce
Parents Determination	Social Security	Internet Access,ATM
Missing Children	Border Control	Cellular Phnones

1.3 PROBLEM STATEMENT

To design a fingerprint verification system to reduce EER and improve the accuracy of system. Fingerprint recognition technology extracts features from impressions made by the distinct ridges on the fingertips. The fingerprints can be either flat or rolled. A flat print captures only an impression of the central area between the fingertip and the first knuckle; a rolled print captures ridges on both sides of the finger.

1.4 OBJECTIVES

The objectives of fingerprint enhancement process as used by me are as follows:

1) We propose a simple and effective approach for fingerprint image enhancement and minutiae extraction based on the frequency and orientation of the local ridges and thereby extracting correct minutiae.

2) We propose a simple and effective approach for Biometric fingerprint image enhancement and minutiae extraction based on the frequency and orientation of the local ridges and thereby extracting correct minutiae points.

3) Automatic and reliable extraction of minutiae from fingerprint images is a critical step in fingerprint matching. The quality of input fingerprint images plays an important role in the performance of automatic identification and verification algorithms. In this project we presents a fast fingerprint enhancement and minutiae extraction algorithm which improves the clarity of the ridge and valley structures of the input fingerprint images based on the frequency and orientation of the local ridges and thereby extracting correct minutiae.

1.5 BIOMETRIC METHODOLOGY

Biometrics are security mechanism used to authenticate and provide access to a facility or system based on the automatic and instant verification of an individual's physical characteristics. The study of automated identification, by use of physical or behavioral traits. An authentication can be divided into two modules:

a.) Enrollment module

b.) Identification or Verification module

1.5.1 Enrollment module

In enrollment, a biometric system is trained to identify a specific person. The person first provides an identifier, such as an identity card. The biometric is linked to the identity specified on the identification document. He or she then presents the biometric (e.g., fingertips, hand, or iris) to an acquisition device. The distinctive features are located and one or more samples are extracted, encoded, and stored as a reference template for future comparisons. Depending on the technology, the biometric sample may be collected as an image, a recording, or a record of related dynamic measurements.

Minute changes in positioning, distance, pressure, environment, and other factors influence the generation of a template. Consequently, each time an individual's biometric data are captured, the new template is likely to be unique. Depending on the biometric system, a person may need to present biometric data several times in order to enroll.

Either the reference template may then represent an amalgam of the captured data or several enrollment templates may be stored. The quality of the template or templates is critical in the overall success of the biometric application. Because biometric features can change over time, people may have to reenroll to update their reference template.

1.5.2 Biometric Verification System

In verification systems, the step after enrollment is to verify that a person is who he or she claims to be (i.e., the person who enrolled). After the individual provides an identifier, the biometric is presented, which the biometric system captures, generating a trial template that is based on the vendor's algorithm. The system then compares the trial biometric template with this person's reference template, which was stored in the system during enrollment, to determine whether the individual's trial and stored templates match. Verification is often referred to as 1:1 (one-to-one) matching. Verification systems can contain databases ranging from dozens to millions of enrolled templates but are always predicated on matching an individual's presented biometric against his or her reference template. Nearly all verification systems can render a match–no-

match decision in less than a second.

One of the most common applications of verification is a system that requires employees to authenticate their claimed identities before granting them access to secure buildings or to computers.

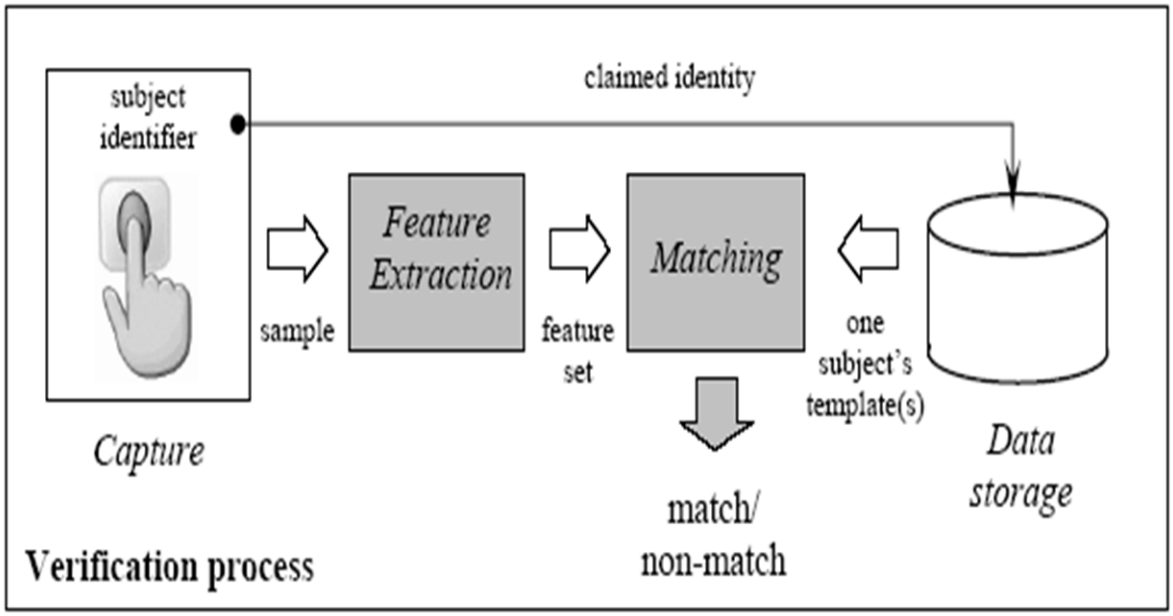


Figure 2: Verification Process

1.5.3 Biometric Identification System

In identification systems, the step after enrollment is to identify who the person is. Unlike verification systems, no identifier is provided. To find a match, instead of locating and comparing the person's reference template against his or her presented biometric, the trial template is compared against the stored reference templates of all individuals enrolled in the system. Identification systems are referred to as 1: M (one-to-M, or one-to-many) matching because an individual's biometric is compared against multiple biometric templates in the system's database. There are two types of identification systems: positive and negative. Positive identification systems are designed to ensure that an individual's biometric is enrolled in the database. The anticipated result of a search is a match.

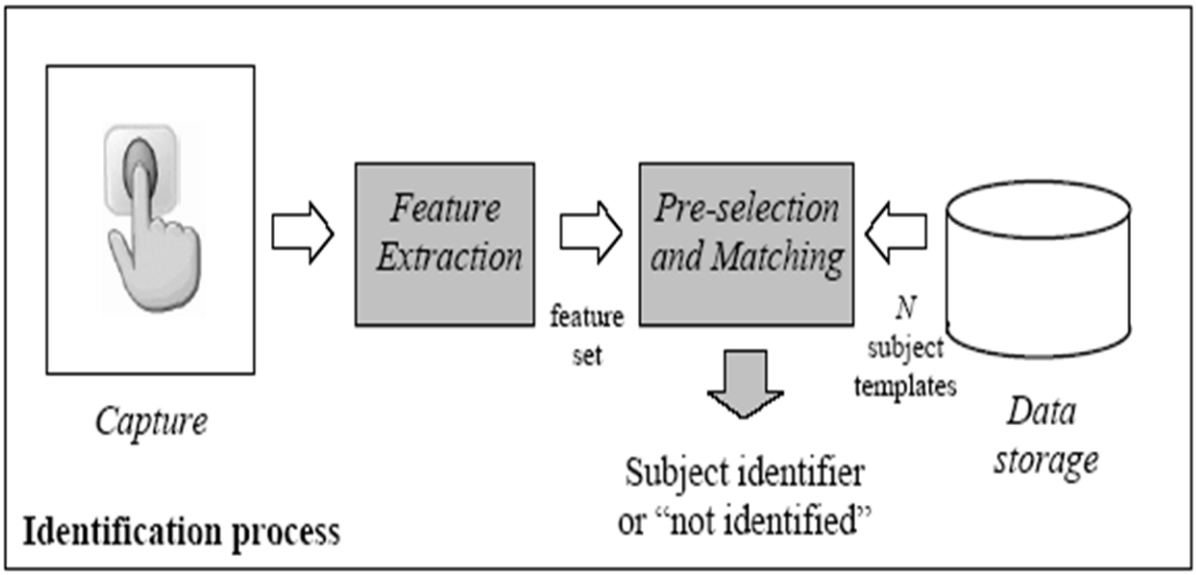


Figure 3: Identification Process

1.5.4 Matches Are Based on Threshold Settings

No match is ever perfect in either verification or identification system, because every time a biometric is captured, the template is likely to be unique. Therefore, biometric systems can be conFigured to make a match or no-match decision, based on a predefined number, referred to as a threshold, which establishes the acceptable degree of similarity between the trial template and the enrolled reference template. After the comparison, a score representing the degree of similarity is generated, and this score is compared to the threshold to make a match or no-match decision. Depending on the setting of the threshold in identification systems, sometimes several reference templates can be considered matches to the trial template, with the better scores corresponding to better matches.

1.5.5 Leading Biometric Technologies

A growing number of biometric technologies have been proposed over the past several years, but only in the past 5 years have the leading ones become more widely deployed.

Some technologies are better suited to specific applications than others, and some are more acceptable to users. We describe seven leading biometric technologies: [1]

1)Facial Recognition: Captures and compares facial patterns and is suitable for border control.

2) Fingerprint Recognition : Captures and compares fingertip patterns suitable for border control.

3) Hand Geometry : It measures and compares dimensions of hand and fingers and is suitable for border control(verification only).

4) Iris Recognition : It captures and compares iris patterns and is suitable for border control.

5)Signature Recognition : It captures and compares rhythm, acceleration, and pressure flow of signature and is suitable for border control.

6)Speaker Recognition : It captures and compares cadence, pitch, and tone of vocal tract and is suitable for border control.

Table 2: Comparison of different biometric traits [Jain, Prabhakar, & Ross, 2004]

Biometric identifier	Universality	Distinctiveness	Permanence	Collectability	Performance	Acceptability	Circumvention
DNA	H	H	H	L	H	L	L
Ear	M	M	H	M	M	H	M
Face	H	L	M	H	L	H	H
Facial thermogram	H	H	L	H	M	H	L
Fingerprint	M	H	H	M	H	M	M
Gait	M	L	L	H	L	H	M
Hand geometry	M	M	M	H	M	M	M
Hand vein	M	M	M	M	M	M	L
Iris	H	H	H	M	H	L	L
Keystroke	L	L	L	M	L	M	M
Odor	H	H	H	L	L	M	L
Palmprint	M	H	H	M	H	M	M
Retina	H	H	M	L	H	L	L
Signature	L	L	L	H	L	H	H
Voice	M	L	L	M	L	H	H

1.5.6 Fingerprint Recognition

Fingerprint recognition is one of the best known and most widely used biometric technologies. Automated systems have been commercially available since the early 1970s, and at the time of our study, we found there were more than 75 fingerprint recognition technology companies. Until recently, fingerprint recognition was used primarily in law enforcement applications.

Fingerprint recognition technology extracts features from impressions made by the distinct ridges on the fingertips. The fingerprints can be either flat or rolled. A flat print captures only an impression of the central area between the fingertip and the first knuckle; a rolled print captures ridges on both sides of the finger.

An image of the fingerprint is captured by a scanner, enhanced, and converted into a template. Scanner technologies can be optical, silicon, or ultrasound technologies. Ultrasound, while potentially the most accurate, has not been demonstrated in widespread use. In 2002, we found that optical scanners were the most commonly used. During enhancement, “noise” caused by such things as dirt, cuts, scars, and creases or dry, wet or worn fingerprints is reduced, and the definition of the ridges is enhanced. Approximately 80 percent of vendors base their algorithms on the extraction of minutiae points relating to breaks in the ridges of the fingertips. Other algorithms are based on extracting ridge patterns.

Among all biometric traits, fingerprints have one of the highest levels of reliability and have been extensively used by forensic experts in criminal investigations. A fingerprint refers to the flow of ridge patterns in the tip of the finger. The ridge flow exhibits anomalies in local regions of the fingertip (Figure), and it is the position and orientation of these anomalies that are used to represent and match fingerprints.

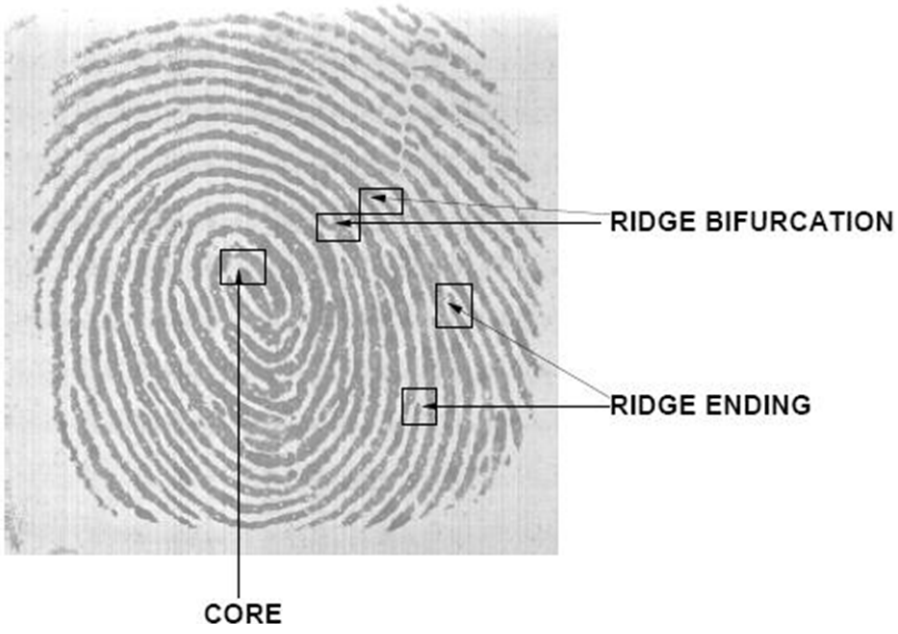


Figure 4: Ridge Flow

1.5.7 Fingerprint Representation

The uniqueness of a fingerprint is determined by the topographic relief of its ridge structure and the presence of certain ridge anomalies termed as minutiae points. Typically, the global configuration defined by the ridge structure is used to determine the class of the fingerprint, while the distribution of minutiae points is used to match and establish the similarity between

two fingerprints.

Automatic fingerprint identification systems, that match a query print against a large database of prints (which can consist of millions of prints), rely on the pattern of ridges in the query image to narrow their search in the database (fingerprint indexing), and on the minutiae points to determine an exact match (fingerprint matching). The ridge flow pattern itself is rarely used for matching fingerprints.

1.5.8 Minutiae

Minutiae, in fingerprinting terms, are the points of interest in a fingerprint, such as bifurcations (a ridge splitting into two) and ridge endings. Examples are:

a.) ridge endings - a ridge that ends abruptly

b.) ridge bifurcation - a single ridge that divides into two ridges

c.) short ridges, island or independent ridge - a ridge that commences, travels a short distance and then ends

d.) ridge enclosures - a single ridge that bifurcates and reunites shortly afterward to continue as a single ridge.

e.) spur - a bifurcation with a short ridge branching off a longer ridge

f.) crossover or bridge - a short ridge that runs between two parallel ridges

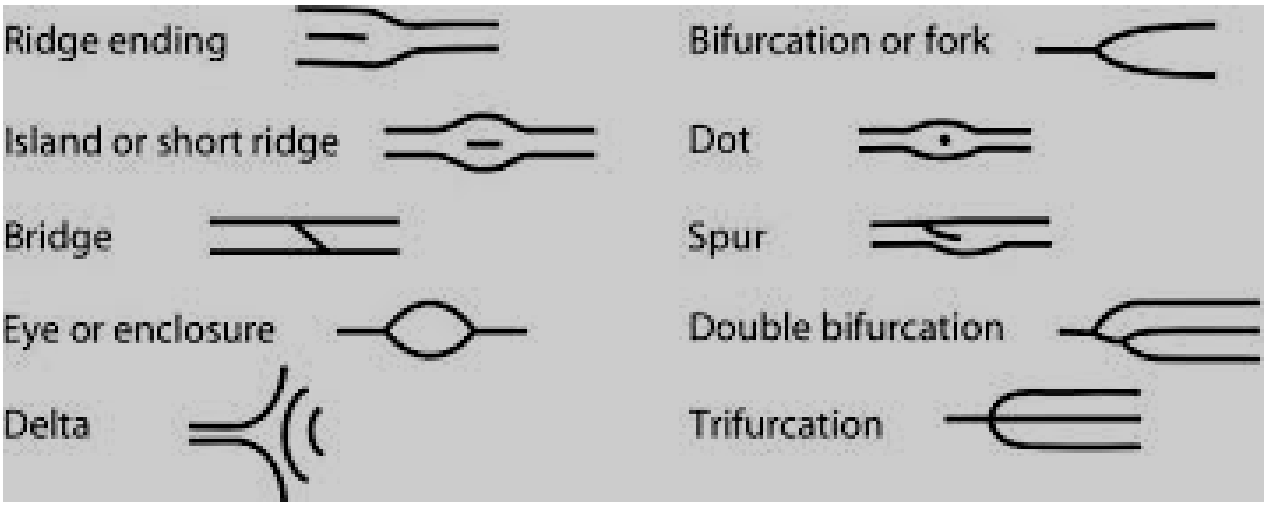


Figure 5: Different types of Minutiae[International Journal of Emerging Technology and Advanced Engineering]

CHAPTER 2
Fingerprint Based Verification system: A Brief Literature Survey

2.1Hong et al.[1999] proposed a method based on fingerprint image enhancement: algorithm and performance evaluation. A critical step in automatic fingerprint matching is to automatically and reliably extract minutiae from the input fingerprint images. However, the performance of a minutiae ex- traction algorithm relies heavily on the quality of the input fingerprint images. In order to ensure that the performance of an automatic fingerprint identification/verification system will be robust with respect to the quality of input fingerprint images, it is essential to incorporate a fingerprint enhancement algorithm in the minutiae extraction module. We present a fast fingerprint enhancement algorithm, which can adaptively improve the clarity of ridge and furrow structures of input fingerprint images based on the estimated local ridge orientation and frequency.

Algorithm used:

A gray-level fingerprint image, I, is defined as a N*N matrix , where I(i,j)represents the intensity of the pixel at the i th row and j th column. We assume that all the images are scanned at a resolution of 500 dots per inch (dpi), which is the resolution recommended by FHI. The man and variance of a gray level fingerprint image, I are defined as

M(I) = 1/N^2 sigma I(I,j)

VAR(I)= 1/N^2 sigma I(I,j) – M(I)^2

Respectively.

An orientation image, O, is define as a N N image, where O(i; j) represents the local ridge orientation at pixel (i; j). Local ridge orientation is usually specified for a block rather than at every pixel; an image is divided into a set of non-overlapping blocks and a single local ridge orientation is defined for each block.

IMAGE ENHANCEMENT TECHNIQUES

1) Frequency domain techniques

2) Spatial domain techniques

FREQUENCY DOMAIN TECHNIQUES

It is based on computing fourier transform of the image. The computing of the fourier transform of the image to be enhanced simply involves multiplying the result by a filter rather than by convolving.

FILTERS USED:

A. LOW PASS FILTER

It involves the elimination of the high frequency components from the image resulting in sharp transition reduction that are associated with noise.

DISADVANTAGES:-

Blurring and ringing.

These are caused by the shapes of frequency domain filters.

B. LOW PASS FILTER

These make image appear more sharper.They emphasis on fine details of the image. But they can degrade the quality of the image.

SPATIAL DOMAIN TECHNIQUES

They uses two operations-

1) Point processing operation
   - ☐ Can only be used for linear stretchin.
   - ☐ Cannot produce much effective results.
2) Histogram equalization
   - ☐ It is used for visual perception especially when image have close contrast data.
   - ☐ It results in noise amplification

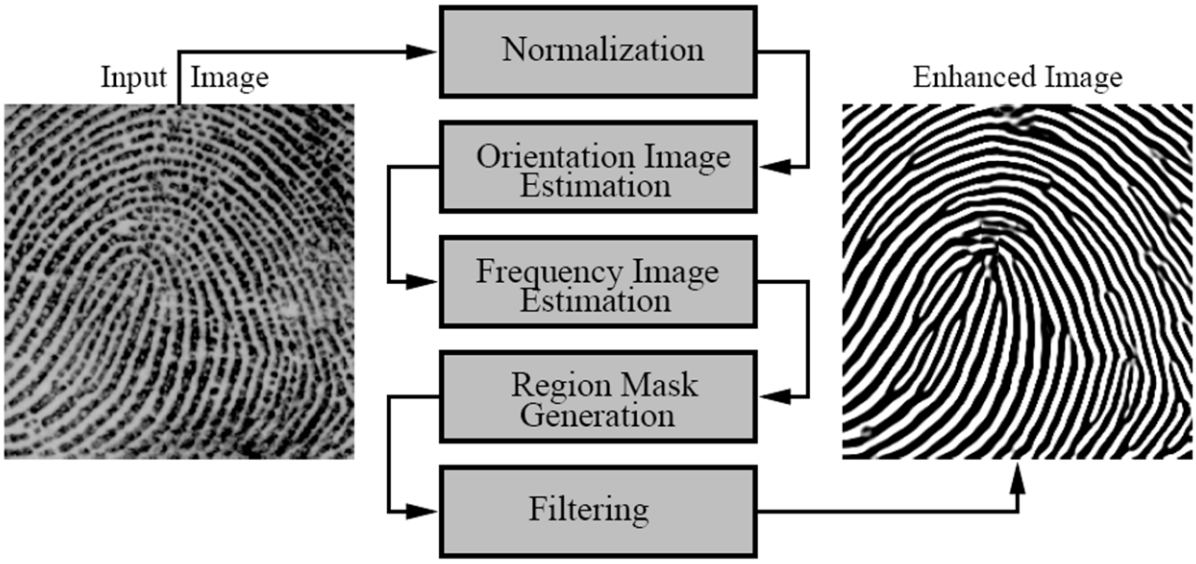


Figure 6: Minutiae Ex-traction Algorithm

2.2 K.B Raja et al.[1999] proposed a method based on a survey of biometric recognition methods. A brief overview of biometric methods and their advantages and disadvantages.

A wide variety of systems requires reliable personal recognition schemes to either confirm or determine the identity of an individual requesting their services. The purpose of such schemes is to ensure that the rendered services are accessed only by a legitimate user and no one else. Examples of such applications include secure access to buildings, computer systems, laptops, cellular phones, and ATMs. In the absence of robust personal recognition schemes, these systems are vulnerable to the wiles of an impostor. Biometric recognition or, simply, biometrics refers to the automatic recognition of individuals based on their physiological and/or behavioral characteristics. By using biometrics, it is possible to confirm or establish an individual's identity based on "who she is," rather than by "what she possesses" (e.g., an ID card) or "what she remembers" (e.g., a password). In this paper, we give a brief overview of the field of biometrics and summarize some of its advantages, disadvantages, strengths, limitations, and related privacy concerns.

2.3Sonavane et al.[2007] proposed Noisy Fingerprint Image Enhancement Technique for

Image Analysis: A Structure Similarity Measure Approach. Fingerprint images vary in quality. In order to ensure that the performance of an automatic fingerprint identification system (AFIS) will be robust with respect to the quality of input fingerprint images, it is essential to incorporate a fingerprint enhancement module in the AFIS system. A technology for recognizing fingerprints for security purposes is proving as regards as reliable but efficient recognition is depending on the quality of input fingerprint image. Recognition of the fingerprint becomes a complex computer problem while dealing with noisy and low quality images. In this Paper work the focus is on the special domain biometric System of noisy and low quality images, which will be beneficial for recognition system.

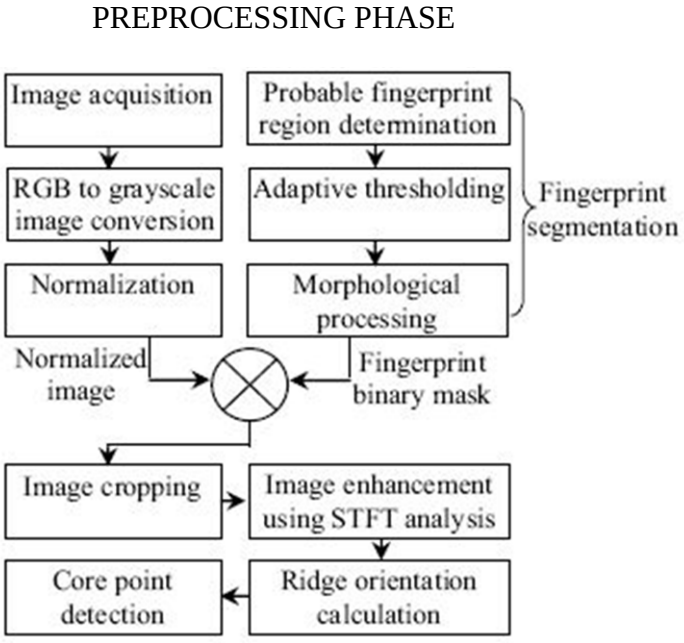


Figure 7: Image Improvement

2.4 KB Raja and K.R et al.[2011] proposed a method based on a Fingerprint Recognition Using Binarization. The Binarization step is basically stating the obvious, which is that the true information that could be extracted from a print is simply binary; ridges vs. valleys. But it is a really important step in the process of ridge extracting, since the prints are taken as grayscale images, so ridges, knowing that they're in fact ridges, still vary in intensity. So, binarization transforms the image from a 256-level image to a 2-level image that gives the same information. Typically, an object pixel is given a value of "1" while a background pixel is given a value of

“0.” Finally, a binary image is created by coloring each pixel white or black, depending on a pixel's label (black for 0, white for 1).

The difficulty in performing binarization is that not all the fingerprint images have the same contrast characteristics, so a single intensity threshold (global thresholding) cannot be chosen.

A locally adaptive binarization method is performed to binarize the fingerprint image. In this method, the image is divided into blocks (16x16), and the mean intensity value is calculated for each block, then each pixel is turned into 1 if its intensity value is larger than the mean intensity value of the current block to which the pixel belongs.

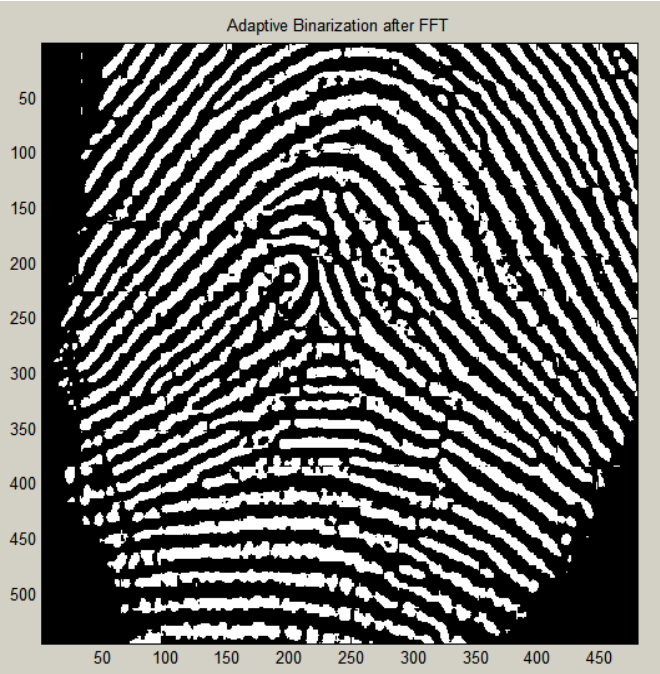
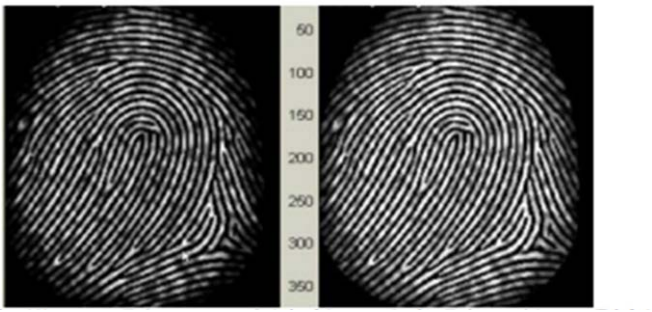


Figure 8: Adaptive Binarization After Fft

2.5 Verma et al.[2012] proposed Feature Extraction Algorithm of Fingerprint Recognition. Most fingerprint matching algorithms are based on finding correspondences between minutiae in two fingerprints. Minutiae based matching approach considers the overall minutiae distribution pattern between the two fingerprints. Neighbourhood correlation score represents the local similarity between the matched pair of minutiae. Edge correlation score gives the resemblance of areas that in between the two corresponding minutiae pairs. With identity fraud in our society reaching unprecedented proportions and with an increasing emphasis on the emerging automatic

personal identification applications, biometrics-based verification, especially fingerprint-based identification, is receiving a lot of attention Biometrics deals with identifying individuals with help of their biological data.



Original image Enhanced image

Figure 9: Feature Extraction

2.6 Thai et al.[2009] One of the most widely cited fingerprint enhancement techniques is the method employed by Hong et al. [8], which is based on the convolution of the image with Gabor filters tuned to the local ridge orientation and ridge frequency. The main stages of this algorithm include normalisation, ridge orientation estimation, ridge frequency estimation and filtering.

1) The first step in this approach involves the normalisation of the fingerprint image so that it has a prespecified mean and variance. Due to imperfections in the fingerprint image capture process such as non-uniform ink intensity or 3 non-uniform contact with the fingerprint capture device, a fingerprint image may exhibit distorted levels of variation in grey-level values along the ridges and valleys. Thus, normalisation is used to reduce the effect of these variations, which facilitates the subsequent image enhancement steps. An orientation image is then calculated, which is a matrix of direction vectors representing the ridge orientation at each location in the image.

The widely employed gradient-based approach is used to calculate the gradient [18, 20, 22], which makes use of the fact that the orientation vector is orthogonal to the gradient. Firstly, the image is partitioned into square blocks and the gradient is calculated for every pixel, in the x and y directions. The orientation vector for each block can then be derived by performing an averaging operation on all the vectors orthogonal to the gradient pixels in the block.

Due to the presence of noise and corrupted elements in the image, the ridge orientation may not

always be correctly determined. Given that the ridge orientation varies slowly in a local neighbourhood, the orientation image is then smoothed using a low-pass filter to reduce the effect of outliers.

2) The next step in the image enhancement process is the estimation of the ridge frequency image. The frequency image defines the local frequency of the ridges contained in the fingerprint. Firstly, the image is divided into square blocks and an oriented window is calculated for each block. For each block, an x-signature signal is constructed using the ridges and valleys in the oriented window. The xsignature is the projection of all the grey level values in the oriented window along a direction orthogonal to the ridge orientation. Consequently, the projection forms a sinusoidal-shape wave in which the centre of a ridge maps itself as a local minimum in the projected wave. The distance between consecutive peaks in the x-signature can then be used to estimate the frequency of the ridges. Fingerprint enhancement methods based on the Gabor filter have been widely used to facilitate various fingerprint applications such as fingerprint matching [17, 19] and fingerprint classification [12]. Gabor filters are bandpass filters that have both frequency-selective and orientation-selective properties [4], which means the filters can be effectively tuned to specific frequency and orientation values.

3) One useful characteristic of fingerprints is that they are known to have well defined local ridge orientation and ridge frequency. Therefore, the enhancement algorithm takes advantage of this regularity of spatial structure by applying Gabor filters that are tuned to match the local ridge orientation and frequency. Based on the local orientation and ridge frequency around each pixel, the Gabor filter is applied to each pixel location in the image. The effect is that the filter enhances the ridges oriented in the direction of the local orientation, and decreases anything oriented differently. Hence, the filter increases the contrast 4 between the foreground ridges and the background, whilst effectively reducing noise. An alternative approach to enhancing the features in a fingerprint image is the technique employed by Sherlock [21] called directional Fourier filtering. The previous approach was a spatial domain technique that involves spatial convolution of the image with filters, which can be computationally expensive.

4) Alternatively, operating in the frequency domain allows one to efficiently convolve the

fingerprint image with filters of full image size. The image enhancement process begins by firstly computing the orientation image. In contrast to the previous method, which estimates the ridge orientation using a continuous range of directions, this method uses a set of only 16 directions to calculate the orientation. An image window is centred at a point in the raw image, which is used to obtain a projection of the local ridge information.

5) The image window is then rotated in each of the 16 equally spaced directions, and in each direction a projection along the window's y axis is formed. The projection with the maximum variance is used as the dominant orientation for that point in the image. This process is then repeated for each pixel to form the orientation image. Similar to the filtering stage applied by Hong et al.: after the orientation image has been computed, the raw image is then filtered using a set of bandpass filters tuned to match the ridge orientation. The image is firstly converted from the spatial domain into the frequency domain by application of the two-dimensional discrete Fourier transform. The Fourier image is then filtered using a set of 16 Butterworth filters with each filter tuned to a particular orientation. The number of directional filters corresponds to the set of directions used to calculate the orientation image.

6) After each directional filter has been independently applied to the Fourier image, the inverse Fourier transform is used to convert each image back to the spatial domain, thereby producing a set of directionally filtered images called prefiltered images. The next step in the enhancement process is to construct the final filtered image using the pixel values from the prefiltered images. This requires the value of the ridge orientation at each pixel in the raw image and the filtering direction of each prefiltered image. Each point in the final image is then computed by selecting, from the prefiltered images the pixel value whose filtering direction most closely matches the actual ridge orientation. The output of the filtering stage is an enhanced version of the image that has been smoothed in the direction of the ridges. Lastly, local adaptive thresholding is applied to the directionally filtered image, which produces the final enhanced binary image. This involves calculating 5 the average of the grey-level values within an image window at each pixel, and if the average is greater than the threshold, then the pixel value is set to a binary value of one; otherwise, it is set to zero. The grey-level image is converted to a binary image, as there are only two levels of interest, the foreground ridges and the

background valleys.

7) Overall, it can be seen that most techniques for fingerprint image enhancement are based on filters that are tuned according to the local characteristics of fingerprint images. Both of the examined techniques employ the ridge orientation information for tuning of the filter. However, only the approach by Hong et al. takes into account the ridge frequency information, as Sherlock's approach assumes the ridge frequency to be constant. By using both the orientation and ridge frequency information, it allows for accurate tuning of the Gabor filter parameters, which consequently leads to better enhancement results. Hence, we have chosen to employ the Gabor filtering approach by Hong et al. to perform fingerprint image enhancement.

2.6 Jain, Fellow, IEEE, Yi Chen et al.[2007] Proposed that Fingerprint friction ridge details are generally described in a hierarchical order at three different levels, namely,

Level 1 (pattern),

Level 2 (minutia points), and

Level 3 (pores and ridge contours).

Although latent print examiners frequently take advantage of Level 3 features to assist in identification, Automated Fingerprint Identification Systems (AFIS) currently rely only on Level 1 and Level 2 features. In fact, the Federal Bureau of Investigation's (FBI) standard of fingerprint resolution for AFIS is 500 pixels per inch (ppi), which is inadequate for capturing Level 3 features, such as pores.

With the advances in fingerprint sensing technology, many sensors are now equipped with dual resolution (500 ppi/1,000 ppi) scanning capability. However, increasing the scan resolution alone does not necessarily provide any performance improvement in fingerprint matching, unless an extended feature set is utilized. As a result, a systematic study to determine how much performance gain one can achieve by introducing Level 3 features in AFIS is highly desired. The paper propose a hierarchical matching system that utilizes features at all the three levels extracted from 1,000 ppi fingerprint scans. Level 3 features, including pores and ridge contours, are automatically extracted using Gabor filters and wavelet transform and are locally matched

using the Iterative Closest Point (ICP) algorithm.

ADVANTAGES:-

1) Experiments show that Level 3 features carry significant discriminatory information. There is a relative reduction of 20 percent in the equal error rate (EER) of the matching system when Level 3 features are employed in combination with Level 1 and 2 features. This significant performance gain is consistently observed across various quality fingerprint images.

2) It is suggested that friction ridges are composed of small “ridge units,” each with a pore, and the number of ridge units and their locations on the ridge are randomly established. As a result, the shape, size, alignment of ridge units, and their fusion with an adjacent ridge unit are unique for each person. Although there exist certain cases when ridge units fail to compose a ridge, also known as dysplasia, independent ridge units still exist on the skin .

3) Pores, on the other hand, penetrate into the dermis starting from the epidermis. They are defined as the openings of subcutaneous sweat glands that are placed on epidermis. The study in [16] showed that the first sweat gland formations are observed in the fifth month of gestation while the epidermal ridges are not constructed until the sixth month. This implies that the pores are stabilized on the ridges before the process of epidermis and dermis development is completed, and are immutable once the ridge formation is completed. Due to the fact that, each ridge unit contains one sweat gland, pores are often considered evenly distributed along ridges and the spatial distance between pores frequently appears to be in proportion to the breadth of the ridge, which, on an average, is approximately 0.48 mm [7]. A pore can be visualized as either open or closed in a fingerprint image based on its perspiration activity. A closed pore is entirely enclosed by a ridge, while an open pore intersects with the valley lying between two ridges (see Fig. 5). One should not expect to find two separate prints of the same pore to be exactly alike, as a pore may be open in one and closed in the other print.

CHAPTER 3

System Development

3.1 Approach

There are two approaches for fingerprint recognition. They are image based approach, texture based approach and minutiae based approach. In image based matching, the image itself is used as the template. It requires only low resolution images. Matching is done by optical correlation and is extremely fast. It is based on the global features of a whole fingerprint image. However it requires accurate alignment of the fingerprint samples and is not favorable for changes in scale, orientation and position. The second is the texture based approach. It uses texture information for matching and performs well with poor quality prints. However like image based matching it requires accurate alignment of the two prints and not invariant to translation, orientation and non-linear distortion. Minutiae-based approach is the last approach. Here the ridge features called minutiae are extracted and stored in a template for matching. It is invariant to translation, rotation and scale changes. It is however error prone in low quality images. The minutiae based approach is applied. Usually before minutiae extraction, image preprocessing is performed. In our project we have focused mainly on the preprocessing and extraction stage. Fingerprint enhancements techniques are used to reduce the noise and improve the clarity of ridges against valleys. The image preprocessing consists of the following stages. They are field orientation, ridge frequency estimation, image segmentation and image enhancement thinning. It is followed by a minutiae extraction algorithm which extracts the main minutiae features required for matching of two samples.

3.2 Algorithm Level Design

To implement a minutia extractor, a three-stage approach is widely used by researchers. They are preprocessing, minutia extraction and post-processing stage.[6]

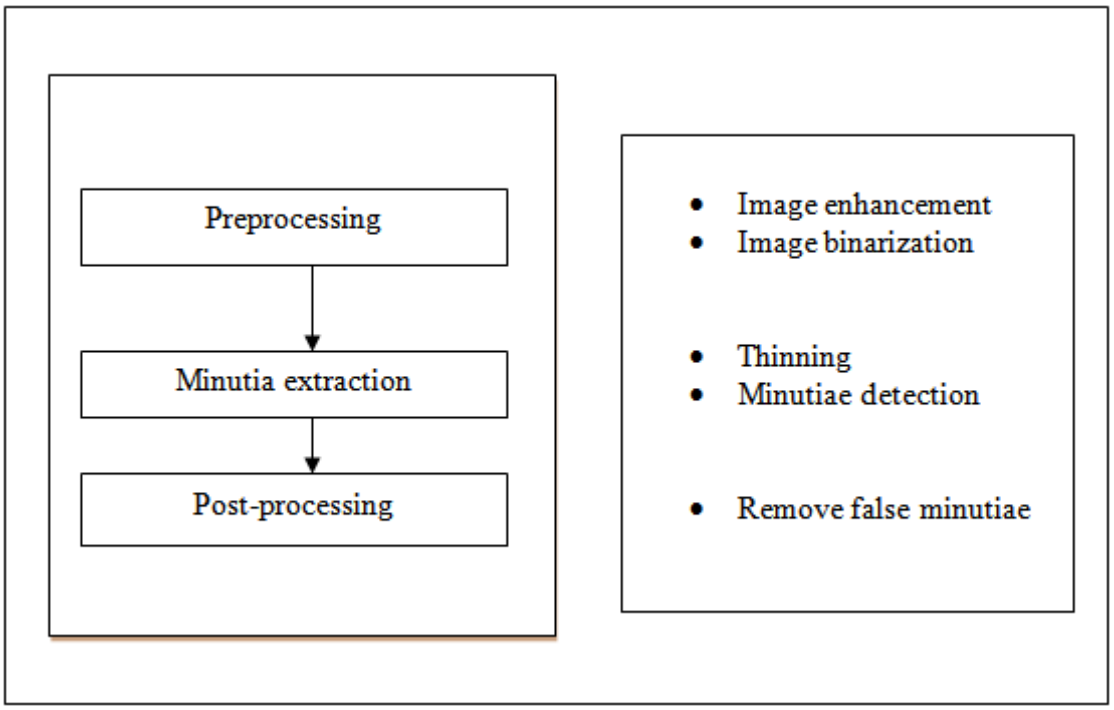


Figure 10: Minutia Extractor

For the fingerprint image preprocessing stage, Histogram Equalization and Fourier Transform are used to do image enhancement. And then the fingerprint image is binarized using the locally adaptive threshold method. The minutia matcher chooses any two minutiae as a reference minutia pair and then matches their associated ridges first. If the ridges match well, the two fingerprint images are aligned and matching is conducted for all the remaining minutiae.

3.2.1 Image Enhancement

The performance of minutiae extraction algorithms and other fingerprint recognition techniques relies heavily on the quality of the input fingerprint images. In an ideal fingerprint image, ridges and valleys alternate and flow in a locally constant direction. However the fingerprint images obtained are usually poor due to elements that corrode the clarity of the ridge elements. This leads to problems in minutiae extraction. Thus, image enhancement techniques are employed to reduce the noise and enhance the definition of ridges against valleys. In order to ensure good performance of the ridge and minutiae extraction algorithms in poor quality fingerprint images, an enhancement algorithm to improve the clarity of the ridge structure is necessary. A fingerprint image contains regions of different quality. They are a) well-defined region b) recoverable region c) unrecoverable region. Well-defined regions, recoverable regions and unrecoverable regions may be identified according to image contrast, orientation consistency, ridge frequency, and other local features. The goal of an enhancement algorithm is to improve the clarity of the ridge structures in the recoverable regions and mark the unrecoverable regions as too noisy for further processing. The input of the enhancement algorithm is a gray-scale image. The output may either be a grayscale or a binary image. There are three stages in image enhancement:

a) Segmentation

b) Normalization

c) Noise Removal

3.2.2 Segmentation

Fingerprint Image enhancement is used to make the image clearer for easy further operations. Since the fingerprint images acquired from scanner or any other media are not assured with perfect quality, those enhancement methods, for increasing the contrast between ridges and valleys 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.

Originally, the enhancement step was supposed to be done using the canny edge detector. But after trial, it turns out that the result of an edge detector is an image with the borders of the ridges highlighted. Using edge detection would require the use of an extra step to fill out the shapes

which would consume more processing time and would increase the complexity of the code, as shown in Figure .

Figure 11(A): Segmentation

In segmentation first if the depth of the image is greater than or equal to 3 we convert the image is converted into gray scale and then its precision is doubled and then whitening of the image is done.

Let V(k) be the variance for a block of size W×W. Then V(k) =1/W2 $W-1$ i=0 $W-1$ j=0 (I(i,j) – M(k))2 (3.1) Where I(i.j) is the grey scale value at pixel(i,j) and M(k) is the mean gray value. The variance threshold separates the foreground regions from the background regions. The foreground regions that are segmented are the areas having the ridge structures. The remaining regions are untouched.

However the threshold must be given properly. If the threshold value is too large, foreground regions may be incorrectly assigned as background regions.

Conversely, if the threshold value is too small, background regions may be assigned as part of the fingerprint foreground area.

A variance threshold of around 100 has been found to give optimal results in terms of differentiating the foreground and background regions.

Figure 11(B): Segmentation

3.2.3 Binarization

The binarization step is basically stating the obvious, which is that the true information that could be extracted from a print is simply binary; ridges vs. valleys. But it is a really important step in the process of ridge extracting, since the prints are taken as grayscale images, so ridges, knowing that they're in fact ridges, still vary in intensity. So, binarization transforms the image from a 256-level image to a 2-level image that gives the same information.

Typically, an object pixel is given a value of "1" while a background pixel is given a value of "0." Finally, a binary image is created by coloring each pixel white or black, depending on a pixel's label (black for 0, white for 1).

The difficulty in performing binarization is that not all the fingerprint images have the same contrast characteristics, so a single intensity threshold (global thresholding) cannot be chosen.
A locally adaptive binarization method is performed to binarize the fingerprint image. In this method, the image is divided into blocks (16x16), and the mean intensity value is calculated for each block, then each pixel is turned into 1 if its intensity value is larger than the mean intensity value of the current block to which the pixel belongs.

Figure 12: Binarization

3.2.4 Normalization

It is the next step in the enhancement algorithm. Normalization is done so that the gray level values lies within a given set of values. The fingerprint image is normalized to have a predefined mean and variance. This is required as the image usually has distorted levels of gray values among the ridges and the valleys. Normalization allows to standardize the distorted levels of variation in the gray scale values. Normalization involves pixel-wise operations and does not change the ridge and valley structures

Normalization is a linear process. Suppose the intensity range of the image is 50 to 180 and the desired range is 0 to 255 the process entails subtracting 50 from each of pixel intensity, making the range 0 to 130. Each pixel intensity is multiplied by 255/130, making the range 0 to 255. The normalized image is given by N(i,j) = M0 + √V0(I(i,j) - M)2 /V if I(i,j) > M M0 - √V0(I(i,j) - M)2 /V otherwise (3.2) Where for a pixel I(i,j) the estimated mean and variances are M and V respectively. M0 and V0 denote the desired mean and variance values.

3.2.5 Minutaie Extraction

It includes various steps required to finally extract a fingerprint and to be sent for matching.

3.2.5.1 Fingerprint Ridge Thinning

Ridge Thinning is to eliminate the redundant pixels of ridges till the ridges are just one pixel wide. An iterative, parallel thinning algorithm is used. In each scan of the full fingerprint image, the algorithm marks down redundant pixels in each small image window (3x3) and finally removes all those marked pixels after several scans. The thinned ridge map is then filtered by other Morphological operations to remove some H breaks, isolated points and spikes. In this step, any single points, whether they are single-point ridges or single-point breaks in a ridge are eliminated and considered processing noise.

Thinning algorithm –

Thinning process is done with applying 4 box of matrices, Each box include three 3×3 matrices and one 4×4 matrix. First each box will be introduced as follow:

1) Diagonal matrices This box consists of four 3×3 matrices, which are built for thinning diagonal lines. If any two pixels lie adjacent to each other at an angle of 45° and a third pixel connects both these pixels at 90°, then the third pixel should be eliminated and continuity of image will be preserved.

2) Horizontal matrices For thinning horizontal lines, we use horizontal matrices. Lines should be thin from down to up.

3) Vertical matrices Thinning of vertical lines could be done with applying another four matrices which thin lines from right to left.

4) Final matrices The thinned image should not have noise and spurious, so after applying all the above matrices, thinning should be continued.

Figure 13: After Thinning

3.2.5.2 Minutia Marking

After the enhancement of the fingerprint image the next step is minutiae extraction. The method extracts the minutiae from the enhanced image. 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 method used for minutiae extraction is the crossing number (CN) method. 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 3×3 window. CN is defined as half the sum of the differences between the pairs of adjacent pixel. The ridge pixel can be divided into bifurcation, ridge ending and non-minutiae point based on it. A ridge ending point has only one neighbor, a bifurcation point possesses more than two neighbors, and a normal ridge pixel has two neighbors. A CN value of zero refers to an isolated point, value of one to a ridge ending, two to a continuing ridge point, three to a bifurcation point and a CN of four means a crossing point. Minutiae detection in a fingerprint skeleton is implemented by scanning thinned fingerprint and counting the crossing

number. Thus the minutiae points can be extracted. A 3×3 window is used. The CN is given by CN=0.5 i=1∑ 8 (Pi- Pi+1)

For a pixel q, the eight pixels are scanned in an anti-clockwise direction. The pixel can be classified after obtaining its pixel value. The coordinates, orientation of the ridge segment and type of minutiae of each minutiae point is recorded for each minutiae. After a successful extraction of minutiae, they are stored in a template, which may contain the minutia position (x,y), minutia direction (angle), minutia type (bifurcation or termination), and in some case the minutia quality may be considered. During the enrollment the extracted template are stored in the database and will be used in the matching process as reference template or database template. During the verification or identification, the extracted minutiae are also stored in a template and are used as query template during the matching.

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

Table 3 : Property Of Cn Number.

3.2.6 Model Development

Standardized fingerprint model From the given images of fingerprint, which are low quality or scaled or rotated together, we propose a model to create a new fingerprint image, which contains features (ridge line and minutia) of the original ones. The model includes the following steps:

(1) Pre-processing fingerprint image: for each image, we recognize fingerprint area, thinned ridge lines and extract minutiae.

(2) Finding and adjusting parameter sets: at first, choose a fingerprint which has largest fingerprint area as mean image. Then, we use Genetic Algorithms in \ to find the transformation between mean image and others.

(3) Synthesizing fingerprint: with the transformations in previous step, we re-calculate parameters' value (to get exact value for parameters), add supplement ridge lines and minutiae to

mean fingerprint.

(4) Post-processing: this step will help removing the noise of step 3.

Figure 14: Pre And Post Processing

Pre-processing fingerprint:

For each input image, we find fingerprint area and thin ridge line whose width is 1 pixel. P is a point on processed fingerprint image and pixel(P) is value of pixel at P:

☐Pixel(P) = 1 if P belong to ridge

☐Pixel(P) = 0 if P belong to valley Each minutia,

we get in this step, contains the x- and y coordinates, the type (which is termination or bifurcation) and the angle between the tangent to the ridge line at the minutia position and the horizontal axis. Result of this step is a processed fingerprint called Flist.

Original Image Image after Removing Noise

Figure 15: Noise Removal

Finding and adjusting parameter set: Base on the result of pre-processing step, we use the Genetic Algorithm which is proposed by Tan and Bhanu in [9] to find the transformation between meanF (a fingerprint which has the largest fingerprint area as mean fingerprint) and others in FList. And then, we re-calculate the exact value of these parameters. Step 1: Find parameter set: In [9], Tan and Bhanu proposed a transformation: Yi =F(Xi) = s.R.Xi + T (1) Where s is the scale factor : angle of rotation between two fingerprints T= [tx,ty] is the vector of translation.[12]

Parameter set contains several parameters. Each parameter has form of . To build parameter set, we perform: Input: fingerprint template FList Output: parameter set ParamList.

1. meanF = fingerprint which has the largest fingerprint area
2. remove meanF from FList 3. For each fData in FList:
a. param = Find the transformation between meanF and fData
b. add param to ParamList

After finishing step 1, we perform the following tasks to re-calculate exact value of parameter in step 2: re-calculate exact value of parameter:

Input: FList, ParamList Output: ParamList with real value of parameters[12]

For each fData in FList: 1. Find 2 minutiae A, B in fData and 2 minutiae C, D in meanF in which A is corresponding to C and B is corresponding D .

2. Calculate the real value for parameters: a. s = sign(old value of s)[(distance between C and D) /(distance between A and B)] b. = sign(old value of)[the angle between AB □□□□ and CD □□□□] c. tx = sign(old value of tx) |(xA-xC)| d. ty = sign(old value of ty) |(yA-yC)|

3. Update new value for corresponding parameter of fData.[12]

CHAPTER 4

Performance Analysis

4.1 EXPERIMENTAL RESULT

Now we will show you the results of this experiment in the next pictures.

4.1.1 Database

Database used for experiment is DB4 FVC2004. Several fingerprint images in this database are low quality. Size of each fingerprint images is 288x384 pixels, and its resolution is 500 dpi. FVC2004 DB4 has 800 fingerprints of 100 fingers (8 images for each finger). Fingerprint images are numbered from 1 to 100 followed by a another number (from 1 to 8) which mean that the image fingerprint is first to 8th impression of certain finger .

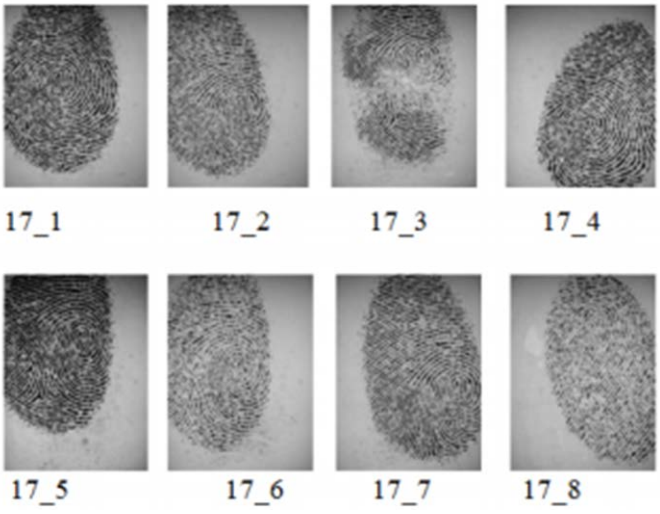


Figure 16: Sample Images From Database

4.2 Evaluation of the system

As we can see in the graph shown below, when eliminating a step from the whole process or changing some of the parameters, the matching process is affected.

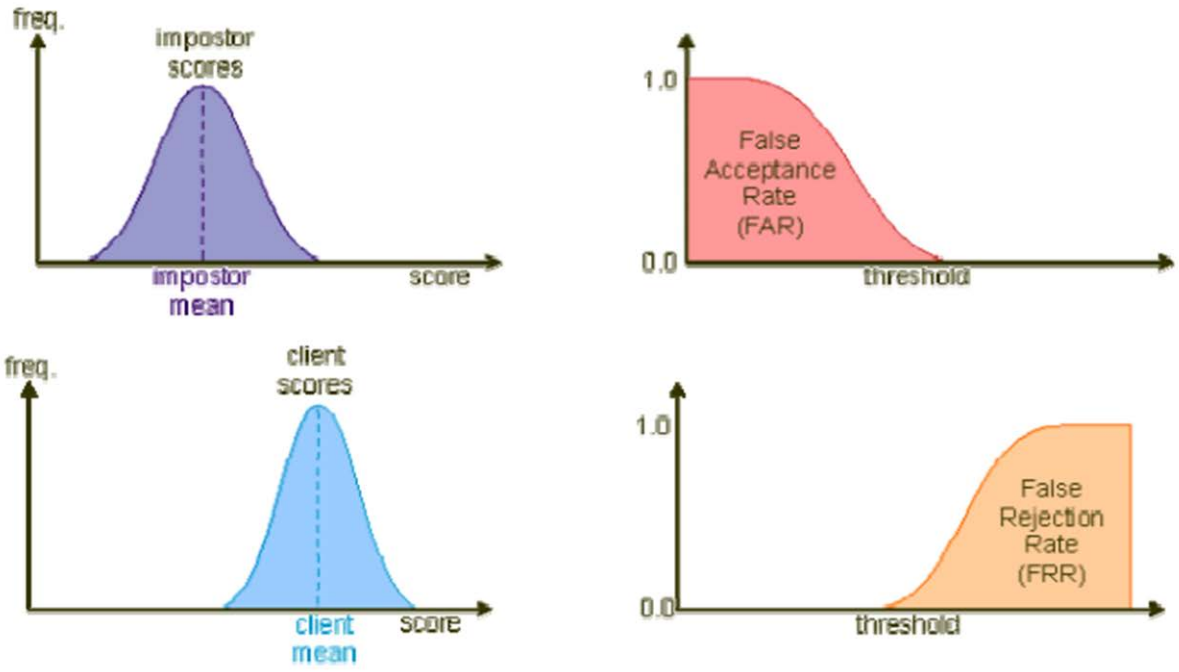
Observations:

1. When altering in such an important step such as the image enhancement part, the performance quality of the system drops rapidly as the noise in the image is increased. Because when working with a biometric identification system, obtaining clear and noise free images is a really hard thing, so this step is usually needed.

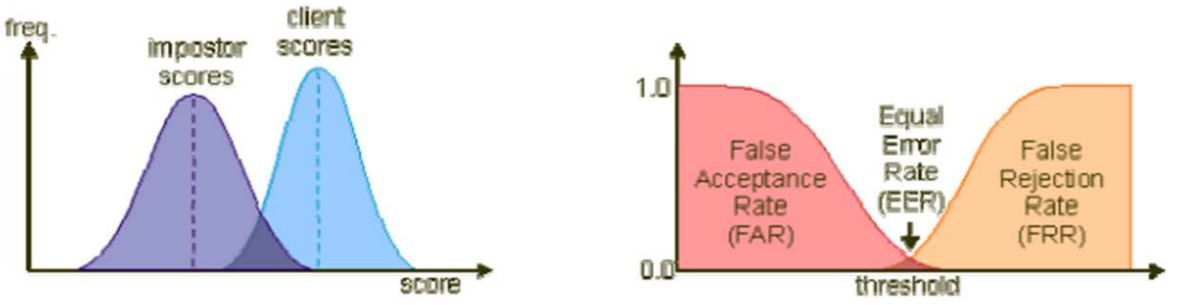
2. For the binarization step, as explained earlier, using global thresholding may introduce a few problems and may lead to the elimination of significant details by mistake. Here, I tried using global thresholding, with 2 different thresholds, once using an intensity threshold of 120 and the second time using a value of 80. As we can see from the graph, setting the threshold at 120 (although it’s almost the average value for a gray-scale image) affected the system performance a lot and led to false non-match results, while setting a fixed threshold as low as 80 gave better results. Still, it remains better to use the adaptive threshold method because, although it consumes more processing time, it still guarantees the quality of the results.

3. If we try to remove the H-breaks step, the system wouldn’t be greatly affected and the matching process wouldn’t become harder, but it’s considered a preprocessing step and it doesn’t add much complexity to the system, so no harm in keeping the accuracy higher.
Depending on the choice of the classification threshold, between all and none of the impostor patterns are falsely accepted by the system. The threshold depending fraction of the falsely accepted patterns divided by the number of all impostor patterns is called False Acceptance Rate (FAR). Its value is one, if all impostor patterns are falsely accepted and zero, if none of the impostor patterns is accepted. Look on the graphic on the right to see the values of the FAR for the score distribution of the left image for varying threshold. Now let's change to the client patterns. Similar to the impostor scores, the client pattern's scores vary around a certain mean value. The mean score of the client patterns is higher than the mean value of the impostor patterns, as shown in the left of the following two images. If a classification threshold that is too high is applied to the classification scores, some of the client patterns are falsely rejected. Depending on the value of the threshold, between none and all of the client patterns will be falsely rejected. The fraction of the number of rejected client patterns divided by the total

number of client patterns is called False Rejection Rate (FRR). According to the FAR, its value lies in between zero and one. The image on the right shows the FAR for a varying threshold for the score distribution shown in the image on the left. The choice of the threshold value becomes a problem if the distributions of the client and the impostor scores overlap, as shown in the next image on the left. On the right, the corresponding false acceptance and false rejection rates are displayed.



The choice of the threshold value becomes a problem if the distributions of the client and the impostor scores overlap, as shown in the next image on the left. On the right, the corresponding false acceptance and false rejection rates are displayed.



Note that if the score distributions overlap, the FAR and FRR intersect at a certain point. The value of the FAR and the FRR at this point, which is of course the same for both of them, is

called the Equal Error Rate (EER).

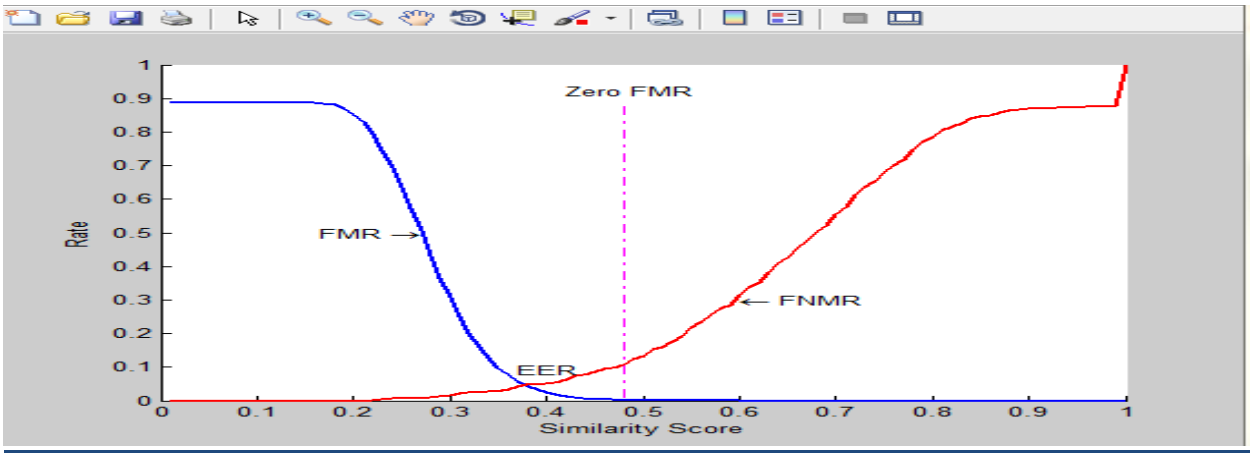


Figure 17: Similarity score,FMR,FNMR,EER

CHAPTER 5

Matching Module

The distinctiveness of a fingerprint can be determined by the overall pattern of ridges and valleys as well as the local ridge anomalies (minutiae points). Although the ridges possess the discriminatory information, designing a reliable automatic fingerprint matching algorithm is very challenging due to the nonlinear deformation and noise in fingerprint images. The existing popular fingerprint matching techniques can be broadly classified into two categories: (a) minutiae-based and (b) correlation-based. The minutiae based techniques typically match the two minutiae sets from two fingerprints by first aligning the two sets and then counting the number of minutiae that match. A typical minutiae extraction technique performs the following sequential operations on the fingerprint image: (*i*) fingerprint image enhancement, (*ii*) binarization (segmentation into ridges and valleys), (*iii*) thinning, and (*iv*) minutiae detection. Several commercial and academic algorithms follow these sequential steps for minutiae detection. Alternative techniques for minutiae detection directly operate on the gray scale fingerprint image itself and detect minutiae by adaptively tracing the gray scale ridges in the fingerprint images. The alignment between the input and the template fingerprints can be obtained using one or more of the fingerprint features. For example, an alignment can be achieved based on the orientation field of the fingerprints, the location of singular points such as the core and the delta, ridges, inexact graph-matching on the minutiae graphs, Hough transform , point patterns , etc.

The number of matched minutiae in certain tolerances is typically normalized by the total number of minutiae in the two sets to account for the falsely detected and missed minutiae during the feature extraction. One of the main difficulties in the minutiae-based approach is that it is very difficult to reliably extract minutiae in a poor quality fingerprint image. A number of image enhancement techniques can be used to improve the quality of the fingerprint image prior to minutiae extraction. Correlation-based techniques match the global pattern of ridges and furrows to see if the ridges align. The simplest technique is to align the two fingerprint images and subtract the input from the template to see if the ridges correspond. However, such a simplistic approach suffers from many problems including the errors in estimation of alignment, non-linear deformation in fingerprint images, and noise. An auto-correlation technique has been proposed

by Sibbald, which computes the correlation between the input and the template at fixed translation and rotation increments. If the correlation exceeds a certain threshold, the two fingerprints are declared to originate from the same finger. A variant of the correlation technique is to perform the correlation in the frequency domain instead of the spatial domain by performing a two-dimensional fast Fourier transform (FFT) on both the input and the template fingerprints. The sum of the pixel-to-pixel multiplication of the two frequency domain representations of the fingerprint images is then compared to a threshold to make a decision. One of the advantages of performing correlation in the frequency domain is that the frequency representations of the fingerprints are translation invariant. One of the major disadvantages, however, is the extra computation time required to convert the spatial image to a frequency representation.

The frequency domain correlation matching can also be performed optically. The input and the template fingerprints are projected via laser light through a lens to produce their Fourier transform and their superposition leads to a correlation peak whose magnitude is high for the matching pair and low otherwise. The main advantage of performing optical correlation is the speed; the main disadvantage is that optical processors have very limited versatility (programmability). A modification of the spatial correlation-based techniques is to divide the fingerprint images into grids and determine the correlation in each sector instead of the whole image. The correlation-based technique overcomes some of the limitations of minutiae-based approach. For example, the minutiae extraction algorithm detects a large number of spurious minutiae and misses genuine minutiae in very noisy fingerprint images. Correlation-based techniques are less sensitive to the noise in fingerprint images but have problems of their own. For example, correlation-based techniques are more sensitive to an error in estimation of the alignment between the two fingerprints. Also, the correlation-based techniques cannot easily deal with the non-linear deformation present in the fingerprint images. Additionally, the correlation-based techniques typically have larger template size.

It is desirable to explore representation schemes which combine global and local information in a fingerprint. Our fixed length code representation for the fingerprints, called Finger Code is suitable for matching as well as storage on a smartcard. The matching reduces to finding the Euclidean distance between these Finger Codes and hence the matching is very fast and the representation is amenable to indexing.

Given two set of minutia of two fingerprint images, the minutia match algorithm determines whether the two minutia sets are from the same finger or not.

- **Alignment stage** :First calculate the similarity of the two ridges associated with the two referenced minutia points. If similarity > threshold, transform each set of minutia to a new coordination system whose origin is at the referenced point and whose x-axis is coincident with the direction of the referenced point.

- **Match stage**: After we get two set of transformed minutia points, we count the matched minutia pairs by assuming two minutia having nearly the same position and direction are identical.

Initially the algo computes pairwise similarity between minutiae of two fingerprints Next, it aligns two fingerprints according to the most similar minutiae pair. The algorithm then finds minutiae that are close enough both in location and direction are deemed to be corresponding (mated) minutiae. Finally, the algorithm computes a similarity score to reflect the degree of match between two fingerprints based on factors such as the number of matching minutiae and the consistency of ridge count between matching minutiae.

Function used:

- A = imread(‘D:Disha\project \t1.jpg'); reads images from location.

- Pic1=rgb2gray(img1)converts truecolor image RGB to gray scale by eliminating hue and saturation info while retaining Luminance.

- edge_det_pic1 = edge(pic1,'prewitt'); Prewitt performs edge detection by calculating gradient vector of each point on the original image. The higher gray level intensity shows border between object and background..

- Percent_match=validtest(pic1,pic2) stores match score.

- Function[Percent_match]=validtest(pic1,pic2) calculates percentage of match score b/w 2 images.

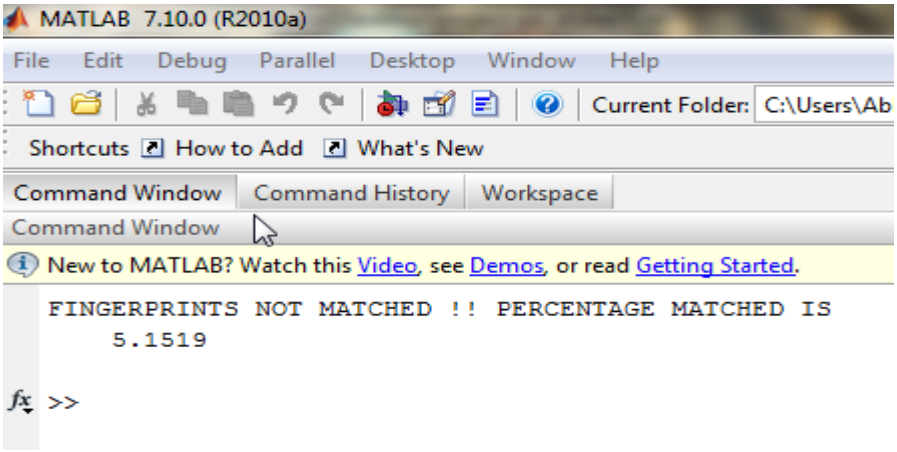
- [x,y,z]=size(pic1) gives two element row vector containing number of rows and columns

b/w objects.

Similarity of correlating the two ridges is derived from:

where (xi~xn) and (Xi~XN) are the set of minutia for each fingerprint image respectively. And m is minimal one of the n and N value.

If the similarity score is larger than 0.8, then go to step 2, otherwise continue.



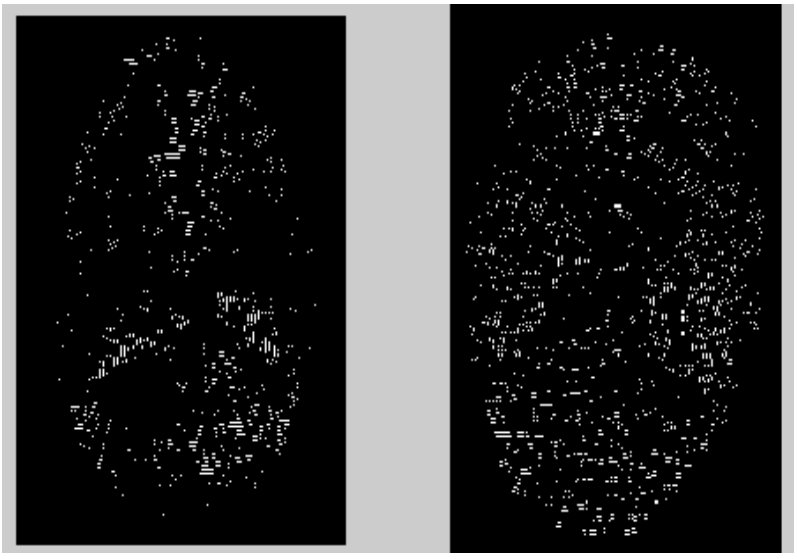


Figure 18: Fingerprint not matched !

Figure 19: Fingerprints matched !

CHAPTER 6

Conclusions And Future Work

The primary focus of the work in this project is on the enhancement of fingerprint images, and the subsequent extraction of minutiae. Firstly, I have implemented a series of techniques for fingerprint image enhancement to facilitate the extraction of minutiae. Experiments were then conducted using a combination of both synthetic test images and real fingerprint images in order to provide a wellbalanced evaluation on the performance of the implemented algorithm. The use of synthetic images has provided a more quantitative and accurate measure of the performance. Whereas real images rely on qualitative measures of inspection, but can provide a more realistic evaluation as they provide a natural representation of fingerprint imperfections such as noise and corrupted elements. The experimental results have shown that combined with an accurate estimation of the orientation and ridge frequency, the Gabor filter is able to effectively enhance the clarity of the ridge structures while reducing noise. In contrast, for low quality images that exhibit high intensities of noise, the filter is less effective in enhancing the image due to inaccurate estimation of the orientation and ridge frequency parameters. However, in practice, this does not pose a significant limitation as fingerprint matching techniques generally place more emphasis on the well-defined regions, and will disregard an image if it is severely corrupted. Overall, the results have shown that the implemented enhancement algorithm is a useful step to employ prior to minutiae extraction. The Crossing Number method was then implemented to perform extraction of minutiae. Experiments conducted have shown that this method is able to accurately detect all valid bifurcations and ridge endings from the thinned image. However, there are cases where the extracted minutiae do not correspond to true minutia points. Hence, an image postprocessing stage is implemented to validate the minutiae. The experimental results from the minutiae validation algorithm indicate that this additional postprocessing stage is effective in eliminating various types of false minutiae structures.

In combination with the implemented techniques for image enhancement and minutiae extraction, preliminary experiments on the statistics of fingerprints were conducted on a sample set of fingerprint images. Three types of statistical data were collected, which include minutiae density, distance between neighbouring minutiae, and ridge wavelength.

Although full analysis of the statistical data was not conducted, the results presented in this dissertation can be used as a basis for future work. Overall, I have implemented a set of reliable techniques for fingerprint image enhancement and minutiae extraction. These techniques can then be used to facilitate the further study of the statistics of fingerprints. In addition, these techniques can be also employed in other fingerprinting applications such as fingerprint matching and classification. Further work which can be carried out include the following:

- An investigation into a filter whose primary aim is to specifically enhance the minutia points. This project has followed the approach adopted by most previous work where the emphasis is on enhancing the ridge structures using Gabor, or Gabor-like filters. However, while the ridge structures are enhanced, this approach has shown to be less effective in enhancing areas containing minutiae points, which are the points of main interest.
- To perform the statistical experiments used in this project on a larger sample size, and to conduct a full analysis of the observed results.
- Further study into the statistical theory of fingerprint minutiae. In particular, the Tu and Hartley [12] approach can be investigated to determine the number of degrees of freedom within a fingerprint population. These results can then be used to help us better understand the statistical uniqueness of fingerprint minutiae.

REFERENCES

[1] Amengual, J. C., Juan, A., Prez, J. C., Prat, F., Sez, S., and Vilar, J. M. Real-time minutiae extraction in fingerprint images. In Proc. of the 6th Int. Conf. on Image Processing and its Applications (July 1997), pp. 871–875.

[2] Daly, F., Hand, D. J., Jones, M. C., Lunn, A. P., and McConway, K. J. Elements of Statistics. Addison-Wesley, 1999, pp. 349–352.

[3] Dankmeijer, J., Waltman, J. M., and Wilde, A. G. D. Biological foundations for forensic identification based on fingerprints. Acta Morphologica Neerlando-scandivanica 18, 1 (1980), 67–83.

[4] Daugman, J. G. Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters. Journal of the Optical Society of America (A) 2, 7 (July 1985), 1160–1169.

[5] Galton, F. Fingerprints. Mcmillan, 1982.

[6] Garris, M. D., Watson, C. I., McCabe, R. M., and Wilson, C. L. National Institute of Standards and Technology fingerprint database, November 2001.

[7] Guo, Z., and Hall, R. W. Parallel thinning with two-subiteration algorithms. Communications of the ACM 32, 3 (March 1989), 359–373.

[8] 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), 777–789.

[9] Jain, A., Hong, L., Pankanti, S., and Bolle, R. An identity authentication system using fingerprints. In Proceedings of the IEEE (September 1997), vol. 85, pp. 1365–1388.

[10] Jain, A. K., and Farrokhnia, F. Unsupervised texture segmentation using Gabor filters. Pattern Recognition 24, 12 (1991), 167–186.

[11] Jain, A. K., Hong, L., and Bolle, R. M. On-line fingerprint verification. IEEE Transactions on Pattern Analysis and Machine Intelligence 19, 4 (1997), 302–314.

[12] Jain, A. K., Prabhakar, S., and Hong, L. A multichannel approach to fingerprint classification. IEEE Transactions on Pattern Analysis and Machine Intelligence 21, 4 (1999), 348–359.

[13] Kingston, C. R. Probabilistic Analysis of Partial Fingerprint Patterns. PhD thesis, University of California, Berkeley, 1964.

[14] Kovesi, P. MATLAB functions for computer vision and image analysis. School of

Computer Science and Software Engineering, The University of Western Australia. http://www.cs.uwa.edu.au/~pk/Research/MatlabFns/index.html Accessed: 20 August 2003.
[15] Maltoni, D., Maio, D., Jain, A. K., and Prabhakar, S. Handbook of Fingerprint Recognition. Springer, 2003.
[16] Mehtre, B. M. Fingerprint image analysis for automatic identification. Machine Vision and Applications 6, 2 (1993), 124–139.
[17] Prabhakar, S., Wang, J., Jain, A. K., Pankanti, S., and Bolle, R. Minutiae verification and classification for fingerprint matching. In Proc. 15th International Conference Pattern Recognition (ICPR) (September 2000), vol. 1, pp. 25–29.

APPENDICES

Program No. 1

Image Enhancement :

SEGMENTATION

```
function segImage=Segmentation(originalImg)


[H W D] = size(originalImg);
if( D==3)
grayImg=im2double(rgb2gray(originalImg));
else
grayImg=im2double(originalImg);
end
Figure
imshow(grayImg),title('Image after rgb2gray');
cannyImg=edge(grayImg,'canny');
Figure
imshow(cannyImg),title('Image after canny');
lowResImg=reduceReselution(cannyImg);
globalThreshold = 0.10;
windowSize=15;
[W H]=size(lowResImg);
padding=8;
paddedImg=padarray(lowResImg,[padding padding],0,'both') ;
count=1;
for i=padding+1: W+padding
for j = padding+1 : 1 : H+padding


block=paddedImg(i-floor(windowSize/2):i+floor(windowSize/2),j-
floor(windowSize/2):j+floor(windowSize/2));
Vecblock=reshape(block,1,windowSize*windowSize);
```

```
Variance = var(Vecblock) ;
if (Variance<globalThreshold)
grayImg((i-padding)*2-1:(i-padding)*2,(j-padding)*2-1:(j-padding)*2)=1;
end
end
end
Figure;
imshow(grayImg),title('Image after whiting boundry');
segImage=deleteMargin(grayImg);
```

NORMALIZATION

```
function normalImg=Normalization(segImage)
[H W]=size(segImage);
vectorImg=reshape(segImage,1,H*W);
meanM=mean(vectorImg);
varV=cov(vectorImg);
estimatedM=0.8;
estimatedV=1;
for i=1:H*W
if(vectorImg(i)>meanM)
normalImg(i)=estimatedM+sqrt(estimatedV/varV*(vectorImg(i)-meanM)^2);
else
normalImg(i)=estimatedM-sqrt(estimatedV/varV*(vectorImg(i)-meanM)^2);
end
end
normalImg=reshape(normalImg,H,W);
Figure
imshow(normalImg),title('Image after normalization');
```

REMOVE NOISE

```
function enhancedImg=removeNoise(img)
[H,W]=size(img) ;
tempImg = double( zeros(H+6,W+6)) ;
count2=1;
tempImg(4:H+3,4:W+3)=img;
for i = 1:3
tempImg(i,(4:W+3))= img(5-i,:);
tempImg(H+3+i,(4:W+3))= img(H-i,:);
tempImg(4:H+3,i)= img(:,5-i);
tempImg(4:H+3,W+3+i)= img(:,W-i);
end
for i=4:H+3
for j=4:W+3
block=tempImg( (i-1:i+1),(j-1:j+1) );
blockVector=reshape(block,1,9) ;
blockVector=sort(blockVector);
mean = floor((sum(blockVector))/(9))  ;
median = blockVector(5) ;
if (mean>=median )
enhancedImg(i-3,j-3)=blockVector(4) ;
else
enhancedImg(i-3,j-3)=blockVector(6) ;
end
end
end
```

Program No. 2

BINARIZATION

```
% This function take an image and return a binary one. it does so
% by using two thresholds', one for the background and one for the
% fingerprint.
%
% The input: I1 - input picture
%
% The output: I3 - the binary picture
%-------------------------------------------------------------

function [ Iout ] = binnary ( I1 , Squere ,  Scale1 , Scale2 )
[m,n] = size (I1) ;

% padding the image with "Padding" number of pixels from every side.
% each of this padded pixels will be "painted" in black.
Pad = floor( Squere / 2 ) ;
I2 = padarray( I1 , [Pad Pad] , 0 , 'both' ) ;
temp2 = reshape( I1 , 1 , m*n ) ;
temp3 = sum(temp2) / size(temp2,2);
Treshold1 = temp3 * Scale1 ; % for background

%-------------------------------------------------------------
for i= 1:m
for j = 1:n
temp1 = I2(i:Squere-1+i,j:Squere-1+j) ;
temp2 = reshape( temp1 , 1 , Squere^2 ) ;
temp3 = sum(temp2) / size(temp2,2);
Treshold2 = temp3 * Scale2 ; % for image

tt = I2(Pad+i,Pad+j) ;
```

```
if ( (tt>=Treshold1) | (tt<=Treshold2) )
Table(i,j) = 0 ;
else
Table(i,j) = 1 ;
end ;
end ;
end ;

for i= 1:m
for j = 1:n
if ( Table(i,j) == 1 )
Iout(i,j) = 1 ;
else
Iout(i,j) = 0 ;
end ;
end ;
end ;
```

Program No. 3

MINUTAE EXTRACTION

```
function [menutiaeImg,menutiae]=menutieaExtraction(thinningImg)
count=1;
crossingNo=0;
menutiae(count,:)=[0,0,0];
[W H]=size(thinningImg)
for x=2:W-1
for y=2:H-1
if(thinningImg(x,y)==1)
crossingNo=getCN(thinningImg,x,y);
```

```
if((crossingNo==1)|(crossingNo==3))
menutiae(count,:)=[x,y,crossingNo];
count=count+1;
end
end
end
end
menutiaeImg = uint8(zeros(W,H,3));
for i=1:count - 1
x=menutiae(i, 1);
y=menutiae(i, 2);
if (menutiae(i, 3) == 1)
menutiaeImg(x,y,:) = [255,255,255];
else
menutiaeImg(x,y,:) = [0,0,255];
end
end
```

function – margeimg

```
function []= margeImg(img, menutiaeTable,lable)
[ImgH ImgW] = size(img);
ModifyImg = uint8(ones(ImgW,ImgH,3));
ModifyImg = img;
tableLength = length(menutiaeTable);

for i=1:ImgH
for j=1:ImgW
for x=1:tableLength
if( menutiaeTable(x,1)==i & menutiaeTable(x,2)==j )
if( menutiaeTable(x,3)==1)
ModifyImg(i,j,1) = 255;
```

```
ModifyImg(i,j,2) = 255;
ModifyImg(i,j,3) = 255;
end
if( menutiaeTable(x,3)==3)
ModifyImg(i,j,1) = 0;
ModifyImg(i,j,2) = 0;
ModifyImg(i,j,3) = 255;
end
end
end
end
end
```

Figure

```
imshow(ModifyImg),title(lable);
```

Program No. 4

MATCHING MODULE

```
clc; clear all; close all;

A = imread('E:\project pic disha\t1.jpg');
B = imread('E:\project pic disha\t6.jpg');

pic1 =rgb2gray(A);
pic2 =rgb2gray(B);
figure
subplot(1,2,1);
imshow(pic1)
subplot(1,2,2);
imshow(pic2)
```

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



edge_det_pic2 = edge(pic2,'prewitt');
figure
subplot(1,2,1);
imshow(edge_det_pic1)
subplot(1,2,2);
imshow(edge_det_pic2)
matched_data = 0;
white_points = 0;
black_points = 0;
x=0;
y=0;
l=0;
m=0;


for a = 1:1:255
   for b = 1:1:127
      if(edge_det_pic1(a,b)==1)
         white_points = white_points+1;
      else
         black_points = black_points+1;
      end
   end
end


for i = 1:1:255
   for j = 1:1:127
      if(edge_det_pic1(i,j)==1)&&(edge_det_pic2(i,j)==1)
         matched_data = matched_data+1;

      end
   end
end




total_data = white_points;
total_matched_percentage = (matched_data/total_data)*100;
```

```
OUTPUT_MESSAGE = 'FINGERPRINT MATCHED!! PERCENTAGE OF MATCH IS ';
OUTPUT_MESSAGE2 = 'FINGERPRINTS NOT MATCHED !! PERCENTAGE MATCHED
IS ';


if(total_matched_percentage >= 90)

  total_matched_percentage;
  disp(OUTPUT_MESSAGE);
  disp(total_matched_percentage);
else

    disp(OUTPUT_MESSAGE2);
    disp(total_matched_percentage);
end


addpath(genpath(pwd));

load('fmr.mat'); load('fnmr.mat');
figure, hold on;

afmr=mean(fmr,2);
plot(0.01:.01:1,afmr,'LineWidth',2);
text(0.60,0.3,'\leftarrow FNMR','HorizontalAlignment','left');

afnmr=mean(fnmr,2);
plot(0.01:.01:1,afnmr,'r','LineWidth',2);
text(0.27,0.5,'FMR \rightarrow','HorizontalAlignment','right');

plot([0.48 0.48],[0.01 0.88],'-.m');
text(0.42,0.92,'Zero FMR');

text(0.37,0.09,'EER');
xlabel('Similarity Score'); ylabel('Rate');
```