FINGERPRINT BASED PERSON IDENTIFICATION AND VERIFICATION

FAYYAZ UL AMIR AFSAR MINHAS

(Thesis submitted in partial fulfillment of requirements for the BS Degree in Computer and Information Sciences)

Pakistan Institute of Engineering and Applied Sciences
Nilore-45650, Islamabad
(August, 2005)

CERTIFICATE OF APPROVAL

Certified that the work contained in this thesis entitled

Fingerprint Based Person Identification and Verification

was carried out by Fayyaz ul Amir Afsar Minhas under my supervision and that in my opinion, it is fully adequate, in scope and quality, for the Degree of BS in Computer and Information Sciences.

Approved By:

Signatures: _____

Supervisor's Name: Dr. Muhammad Arif

_____ August, 2005

Dedicated to My Beloved Parents

جنھوں نے اپنے زخمی ھاتھوں سے

مجھے بھت اونچا تعمیر کیا!

ACKNOWLEDGEMENTS

Praise be to Allah Almighty for He bestowed upon us intelligence, gave us the mental strength to demystify the secrets of the universe, conferred to us the will and the strength to explore and the power to aim for and achieve our goals.

I would like to especially thank Dr. Muhammad Arif without whom this thesis would not exist. During the course of this project, he acted more than just a project supervisor for me. In the downtimes during the project, it was always his patience; encouraging attitude and intelligence that made me go through all the difficulties.

My appreciation goes to my project co-supervisor Dr. Mutawarra Hussain who always supported me with complete dedication and instilled into me the zeal and the fervor to work on.

I am also very grateful To Dr. Sikander Majid Mirza and Mr. Abdul Jalil, who were my project examiners, for their keen interest and extremely fruitful suggestions.

I must also thank Mr. Irfan Hameed, both as my project coordinator and as a teacher, especially for the help that he rendered to me in the development of the database design and implementation for the project.

I owe a great deal to Dr. Masroor Ikram for his technical and moral support in the development of a Laser based fingerprint scanner for the project.

I would also like to thank Mr. Sharat Chikkerur and Mr. Salil Prabhakar for their technical advice in the design and development of the fingerprint enhancement and matching modules and their help in the acquisition of the fingerprint scanner used with this project.

Lastly, many thanks to my roommate and dearest friend, Mr. Mudassar Abbas for helping me in the development of this thesis.

_____

Fayyaz ul Amir Afsar Minhas

PIEAS, Nilore, Islamabad

Table Of Contents

LIST OF FIGURES

LIST OF TABLES

1 Introduction

With the advancements in the field of science and technology through ages and the transformation of the world into a virtual global village by the advent of the Internet, various new commercial applications such as E-banking, electronic fund transfer, online shopping etc. have evolved. Such applications provide the ease of use that had up to a time, been only a speculation. The use of such electronic applications is increasing at an exponential rate with new technologies and features being added every day. However the prevalent use of such services has also pushed the issue of access control and personal identification into the limelight. Without proper person identification and verification, it is easy for any person to pose as someone else in electronic transactions, which would make the practicability of these transactions questionable. Here electronic security solutions come to the rescue.

Traditionally, two major approaches for person identification have been in use [JRP01]: (i) Token Based and (ii) Knowledge based. Token-based approaches establish the identity of a person on the basis of ownership of a token, such as an ID card or a driver's license whereas knowledge-based approaches rely on the possession of certain information e.g. a password by the subject in making an identification decision. The practicability of these approaches lies in their simplicity, ease of use and low system integration cost. However, both methodologies have one major flaw: Neither can accurately determine if the individual that possesses a token or knows some secret information is actually the individual it represents. Tokens can be stolen, and secret information can be guessed or fraudulently obtained. Once a person has the token or secret password, it is easy to pose as the original owner. Therefore these methodologies are unable to satisfy the security requirements of our electronically inter-connected information society.

1.1 Biometrics

Biometrics is the science of identifying a person on the basis of his physical and behavioral features [PIS05]. Since biometrics enable the identification of a person on the basis of his own characteristics which are considered as unique to him, biometrics are therefore more reliable and robust than both of the person recognition approaches

mentioned earlier. Biometrics, provide a solution to the security requirements associated with the currently existing information society and have the tendency to become the future of person identification and verification.

1.1.1 Requirements for a Biometric Identifier

For a biometric feature to be used in a biometric system, it must satisfy the following requirements [JPP00]:

a. **Universality**

   Which means that all persons should possess the characteristic.

b. **Distinctiveness**

   Which means that no two persons should be the same in the characteristic.

c. **Permanence**

   Which means that the characteristic should be permanent over a large period of time.

d. **Collectability**

   Which means that the characteristics can be measured thus enabling automatic verification.

Apart from that, there are many practical issues associated with a biometric feature, like:

a. **Performance**

   Which relates to the identification accuracy, speed and robustness of a biometric system utilizing the biometric feature as the identification measure.

b. **Acceptability**

   Which relates to the extent to which people are willing to accept the biometric feature in a practical biometric system.

c. **Circumvention**

   Which is a measure of the ease with which the biometric measure can be subjected to frauds and forgeries.

1.1.2 Biometric Technologies

A large number of biometric features are available for use in person identification and verification and can be categorized broadly as:

a. Physiological Biometrics

These biometrics are congenital and are associated physically with a subject e.g. face, fingerprints, iris etc.

b. Behavioral Biometrics

A behavioral biometric is a unique feature of a person's behavior on the basis of which he can be identified e.g. signatures, voice prints etc.

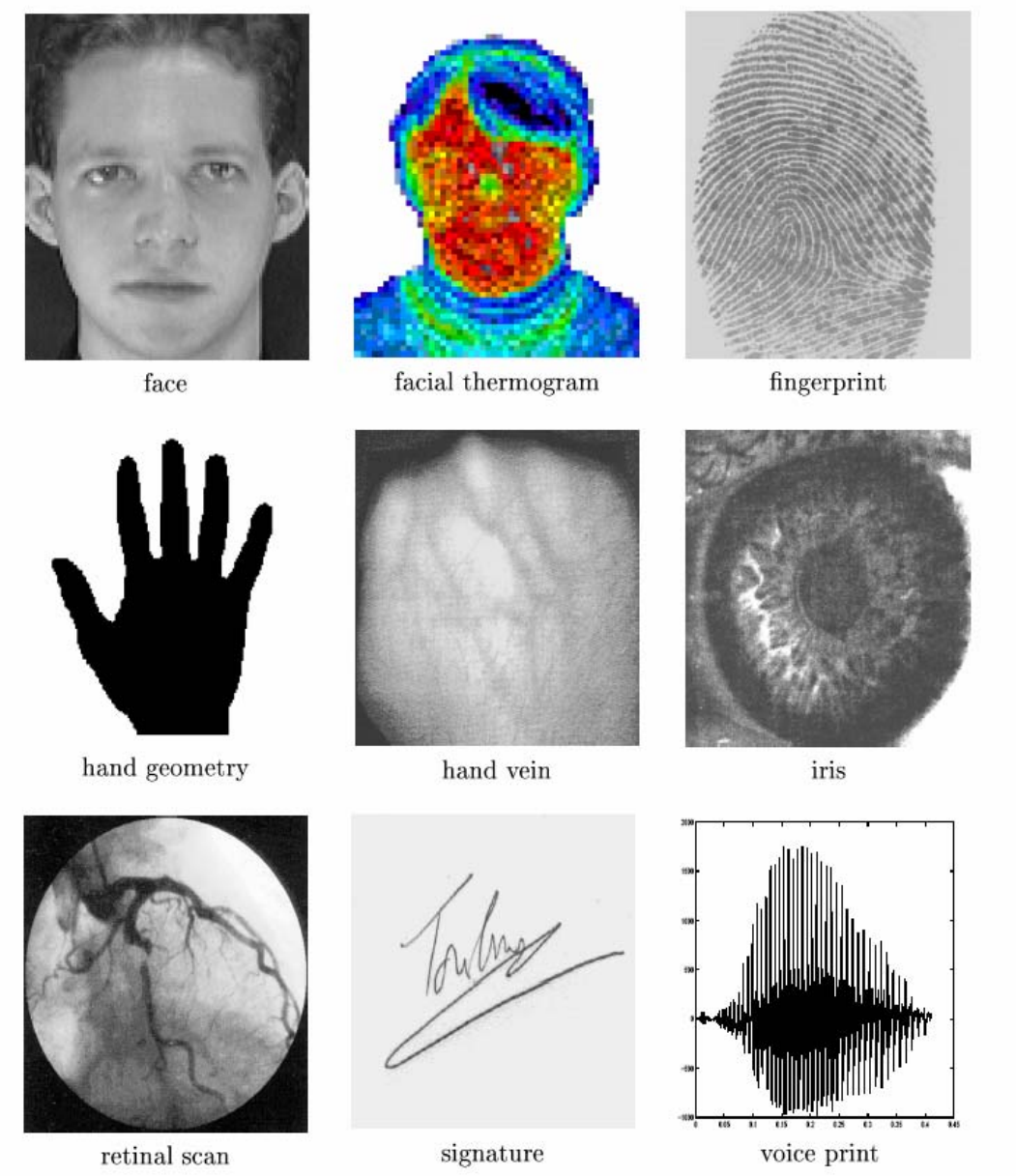

Fig. 1-1 Different Biometric Technologies

Generally it is speculated that physiological biometrics are more reliable in establishing the identity of a person in comparison to behavioral biometrics that can

easily be mimicked and falsified. Apart from that features extracted from physiological biometrics possess better clustering in comparison to behavioral features, which further demonstrates the usability of these features in practical biometric identification systems. A variety of biometric features are available for use in biometric systems, these include: face, fingerprints, facial thermograms, iris, retinal patterns etc. (see Fig.1-1.)

We now present a brief overview of these biometric technologies:

i. Face

Face recognition is the most common biometric used by human beings for person identification and it is one of the most active areas of research in the field of biometrics. Face recognition is an attractive field for automatic person identification system developers because of its high acceptability, non-intrusive nature and fast matching performance. This statement is further supported by the fact that the projected revenue for facial identification systems in 2007 is $429.1 Million. [FCW03].

Fig. 1-2 Problems in face recognition: All the images are of the same person.

However many practical problems manifest themselves in a face recognition system e.g. reliable face segmentation, handling effects of lighting and pose,

expression invariance, occlusion and long range identification etc. Moreover the performance of face recognition systems in outdoor environments is still very low. Considerable research is going on in this field, which is aimed at providing robust and efficient solutions to these problems. Approaches to automatic face recognition include the use of Principal Component Analysis (PCA), Fisher Discriminant Analysis (FDA), Singular Value Decomposition (SVD); Neural Network based Approaches and a variety of Local Feature Analysis techniques.

A large number of commercial products for face recognition currently exist in the market. Despite the great amount of research in this field, the applicability of facial recognition without the use of any contextual information in the process is still doubtful. This fact is illustrated in Fig.1-2.

ii. Facial Thermograms

Identification on the basis of facial thermograms is based on the hypothesis that the heat signature of the face of a person is unique. This heat signature (called facial thermogram) is captured by the use of a heat sensitive IR camera. Facial thermogram is supposed to be more reliable than visual face based identification because the former is invariant to ambient light and skin level alterations such as plastic surgery. Like face recognition, thermogram based identification is also a non-contact biometric feature and is not pose invariant. However the uniqueness and permanency of the facial thermograms is still unproven. The facial thermograms are affected by the emotions and the body temperature of the subject and the ongoing research aims at finding suitable solution to these issues. Moreover because of the high cost associated with a thermogram based person identification system, such systems have not been able to find widespread commercial appreciation.

iii. Hand Geometry

Biometric identification on the basis of hand geometry relies on the hypothesis that the shape of the hand, length of the fingers etc. can uniquely identify a subject. Such systems are operational at over 4000 different locations including the Ministry of Religious Affairs, Pakistan, Colombian legislature, and various airports and are among the most widely developed and time tested biometric systems with the first one developed in the 1980s. This biometric feature finds its use because of its high acceptability, reasonable accuracy and primarily due to its low cost. However the distinctiveness and permanence of this feature are sub optimal which results in a low performance of hand geometry based identification systems in terms of recognition

accuracy. This problem inhibits its use in one-to-many searches using hand geometry. The use of finger geometry for person identification is also being studied.

iv. Hand Vein

Hand vein based person identification approaches count on the uniqueness of the geometrical structure of hand veins of a subject. This structure is captured by the use of an IR camera. It is conjectured that hand vein based systems would provide a very reasonable level of accuracy for biometric identification and such systems have the potential of widespread use because of its high acceptability. However the permanence of these features over time is low which make the use of such a system over a prolonged period difficult.

v. Fingerprints

Fingerprint based biometric identification systems have been in use for over a century, which makes fingerprints as the most widely used biometric feature currently in operation. Fingerprints exhibit high permanence and extremely low variability with age, which makes them ideal for use in access control and forensic applications. Another advantage of using fingerprints is their regenerative nature. Considerable research has been carried out in the field of automatic fingerprint identification systems (AFIS) which makes fingerprint based identification systems highly reliable and robust. The practicability of fingerprints as a means of identifying a person can be judged by revenue for fingerprint based systems in the year 2002: a mighty $323 Million which is expected to rise to a tremendous $1.25 Billion in 2007 [FCW03]. One of the reasons for the extensive use of fingerprints is that fingerprints are one of the few biometric features judicially acceptable. Issues such as the processability of extremely low quality fingerprints and effective fingerprint classification are still being studied as research interests. Additionally palm prints are being used along with fingerprints to further improve the accuracy of fingerprint based authentication systems.

vi. Ear Shape

Ear shape has been used by people for person identification for over a century. However the uniqueness of ear shapes is still unproven. Currently, no commercial biometric systems exist which use this biometric feature but research is ongoing in this field since this biometric feature can be of great assistance in forensics. Along with visual ear images, ear thermograms are also being investigated as biometric features.

vii. Iris

The iris texture in the eye of an individual is unique and is permanent through out life. It is very difficult to change this pattern surgically. All these properties make iris one of the most accurate biometric technologies with a large number of systems in operation. However because of a somewhat complicated and costly acquisition process, iris has a lower acceptability than some of the other biometric technologies.

viii. Retinal Patters

Retinal patterns are considered to be the most secure biometric feature with currently operational systems providing a 100% accuracy. However because of the irksome and expensive acquisition process the acceptability of the retinal patterns in very low, and their use in commercial biometric applications is rather limited.

ix. Signatures

Signatures are a form of behavioral biometric which offers a highly acceptable approach to person identification and verification. Two types of signature verification systems exist: Online and Offline. The online signature verification systems operate by acquiring signatures using a special pressure sensitive tablet, which captures the shape of the signature along with pen dynamics. Offline signatures are scanned images of handwritten signatures on paper. Online signature verification systems provide a higher level of accuracy in comparison to their offline counterparts because the former utilize dynamic information such as pen tilt, pen pressure, signing speed etc. along with the analysis of the shaper of the signature. Different approaches to signature verification have been proposed in the literature. However because of the absence of a standard testing database, it is very difficult to compare these approaches. The error rate of online signature verification systems varies in the range of 1-10% whereas the error rate of offline signatures is considerably higher. However the detection of skilled forgeries is still a problem in signature verification.

x. Voiceprint

Voiceprint based person identification systems exploit the difference in the voice patterns of different persons for identification purposes. A large number of commercial products such as Tespar, BHS-1024 etc. exist which are based on voiceprint. Voiceprints possess high acceptability but the uniqueness of these patterns is questionable which prevents Voiceprint systems to achieve an accuracy required for high security applications. Moreover, voiceprints can easily be mimicked which lessens its potential to obtain widespread use.

xi. Gait

Identification of a person by using his walking style is of great use in outdoor biometrics. It is non-obtrusive and finds its implementation in automated surveillance systems. Gait recognition and stability analysis is an active area of research in the field of biometrics.

xii. Other Technologies

Other technologies such as body odor, lip shape, DNA etc. are also under investigation for person identification. However because of lower performance than fingerprints and iris, the practicability of such biometrics is limited.

Comparison of Biometric Technologies

Table 1-1 presents a comparison of the different biometric techniques in relation to the earlier mentioned requirements of a biometric identifier. Different biometric identifiers have different applications that vary enormously in nature; therefore the practical usage of a biometric technique at a particular location depends on a large number of factors with performance and cost being the highly weighted ones.

Table 1-1 Comparison of Different Biometric Technologies

Biometrics	Universality	Uniqueness	Permanence	Collectability	Performance	Acceptability	Circumvention
Face	high	low	medium	high	low	high	low
Fingerprint	medium	high	high	medium	high	medium	high
Hand Geometry	medium	medium	medium	high	medium	medium	medium
Hand Vein	medium	medium	medium	medium	medium	medium	high
Iris	high	high	high	medium	high	low	high
Retinal Scan	high	high	medium	low	high	low	high
Signature	low	low	low	high	low	high	low
Voice Print	medium	low	low	medium	low	high	low
F.Thermograms	high	high	low	high	medium	high	high

Fingerprints being the only biometric technology accepted in a court of law as indisputable evidence of identity and because of their high accuracy, claim about 48% of the current biometric market share as show in Fig. 1-3, which convincingly demonstrates the usefulness and effectiveness of fingerprints as a biometric identifier.

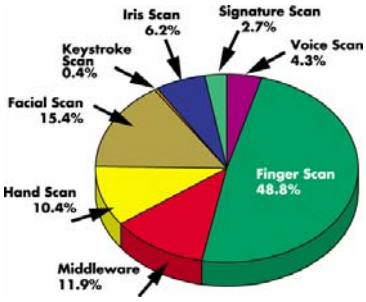

Fig. 1-3 Market Shares of Different Biometric Technologies

1.1.3 Applications

The applications of biometric based security systems can be categorized broadly as:

a. Access Control

Access control applications grant or deny the access to a location depending upon the verification decision generated by examining one or more biometrics of the subject. Access control applications can further be divided into logical and physical access control applications. Logical access control measures are used to protect vital information assets e.g. databases, computer data etc. and moderate the process of electronic transactions e.g. electronic fund transfer in terms of security. Physical access control applications are used to check the access to a physical location of vital strategic or tactical importance. Different applications of biometrics in terms of access control include:

b. Network / PC Login Security

Biometric Logon for PC and Networks is now well developed with a number of players in the market. Best of these in our view is the Identix technology, which is integrated into Windows 2000 Active Directory and provides rapid verification at logon. The second player in this market is the Digital Persona U.R.U system that is slightly behind the Identix system in its development but still provides a robust and affordable solution.

c. Web Page Security

Biometrics also provides measures for implementing security for web pages.

d. Customs and Immigration

Biometrics is providing a very efficient approach to the speedy and much secure implementation of the irrefutable security needs at customs and immigration.

Biometric systems such as INSPASS are becoming more and more common at Airports and Railway stations.

e. Employee Recognition

There are many employee recognition systems available but Biometrics provides a cheaper alternative to most, very few people lose their fingers or eyes when compared with those who lose smart cards or forget passwords.

f. Time and Attendance Systems

Time and attendance has always been a problem in some industries. Biometrics can effectively eliminate problems with buddy clocking by ensuring that the employee in question is present.

g. Voting Solutions

The management of voters to ensure no one votes twice has been a notoriously difficult application, however recent developments in the technology have allowed governments to adopt a high degree of security to prevent such a problem

h. Data Storage

Biometrics provides efficient and effective measures for secure data storage.

i. Encryption

Current applications in biometrics are also being used to generate person specific secure encryption and authentication keys based on some biometric feature that can be used for secure data transfer and storage.

j. Forensics

Some of the biometrics such as fingerprints, voice prints, DNA etc. are used in forensics. This use of the biometric technologies is increasing in the face of increasing threats to national security through out the world.

1.2 System Architecture and Operations

A biometric system includes the hardware, associated software and interconnecting infrastructure to enable the end-to-end biometric process. Technically, a biometric system is a pattern matching system, which makes an identification or verification decision by analyzing one or more biometric characteristic of a person. There are three major logical modules in a biometric system as shown in Fig. 1-4:

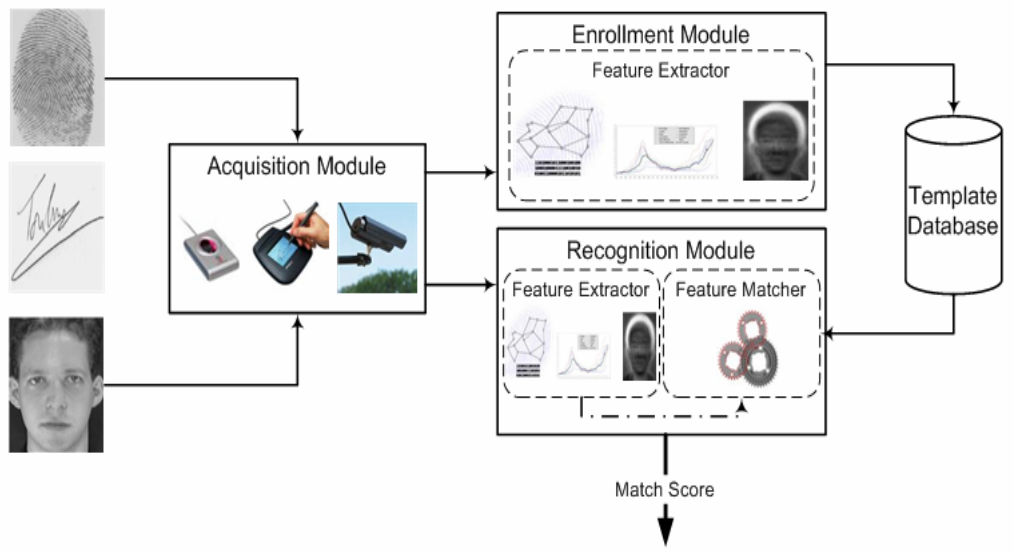

Fig. 1-4 Architecture of a typical Biometric System

a. Acquisition Module

The acquisition module is responsible for capturing a biometric feature from a subject by using a sensor technology suited for operation with the particular type of biometric characteristic being used. Examples include fingerprint scanners, signature tablets, cameras etc.

b. Enrollment Module

Enrollment is defined as the process of storing the features extracted from the raw biometric captured using the acquisition module in a database in the form of templates. A template is a compact representation of a person's biometric features, which are used during the matching process. Depending upon the particular application, the template may be stored in a central biometric system database or be recorded on magnetic strips or cards issued to the individual.

c. Recognition Module

The recognition module is responsible for performing the matching process at a point-of-access. This module first captures the biometric using an acquisition module and extracts certain features, which are then compared with the stored templates to find a similarity score, which is used to generate an authentication decision.

Depending upon its operational needs, a biometric system can either be an identification system or a verification system. A verification system authenticates the identity claim of a person by performing a 1-1 comparison of

the captured biometric feature with the subject's own pre-stored biometric template. Verification systems are usually used in access control applications such as point-of-sales terminals etc. In verification systems an individual to be identified submits a claim to an identity to the system usually through a magnetic strip, smart card, login name etc. and the system either rejects or accepts the identity claim of the person. An identification system conducts one-to-many comparison in order to search for the given biometric characteristic in the template database. Such a systems establishes the identity of a person or fails if the subject is not enrolled in the database without the subject to claim an identity. Such systems are mostly used in forensics where a one-to-many search is usually required.

1.3 Performance Analysis

The performance of a biometric system is usually analyzed in terms of accuracy and speed. A biometric system cannot produce a Boolean decision because of the intra-class variations of the biometric features; therefore, the result of a biometric system is a confidence level. Generally the identity decision produced by a biometric system is genuine type or imposter type, which is represented by two statistical distributions called the genuine distribution and the imposter distribution, respectively. The error produced by a biometric system is due to an overlap between these distributions. If the distributions do not overlap then the biometric system would always produce 0% error, but such ideal conditions do not exist in a practical system and there is always some overlap between the two distributions caused by the intra-class variations in the biometric features. For each type of identity there are two possible outcomes, true or false, thus resulting in a total of four possible outcomes:

- A genuine individual is accepted
- An imposter individual is rejected
- A genuine individual is rejected
- An imposter individual is accepted

The first two outcomes are correct whereas the latter two are inaccurate and result from the overlap between the two distributions. The confidence level associated with the identity established by a biometric verification system is quantized in terms of two error rates: False Acceptance Rate (FAR) and False Rejection Rate (FRR).

FAR is defined as the probability that an imposter is accepted as a genuine individual whereas FRR is the probability that a genuine individual is rejected as an imposter. FAR and FRR are duals: a low FAR usually results in a high FRR and vice versa. Another accuracy measure called the Equal Error Rate (EER) is also used in analyzing the performance of biometric systems, which is defined as the point at which the probability of false acceptance equals the probability of false rejection. The FAR and FRR are graphically represented in terms of the Receiver Operating Characteristics (ROC) curve. The ROC, like the FRR, can only take on values between 0 and 1 and is limited to values between 0 and 1 on the x-axis (FAR). It has the following characteristics:

- The ideal ROC only has values that lie either on the x-axis (FAR) or the y-axis (FRR); i.e., when the FRR is not 0, the FAR is 1, or vice versa.
- The highest point (linear scale under the definitions used here) is for all systems given by FAR=0 and FRR=1.
- The ROC cannot increase

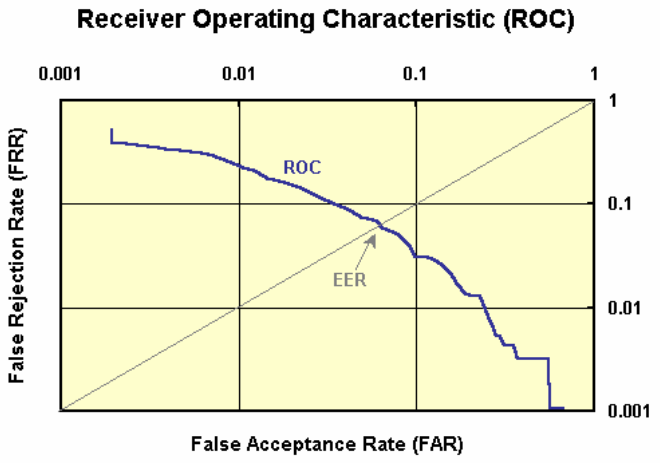

Fig. 1-5 An Example ROC Curve

Fig. 5 shows an example ROC curve. The area under the ROC curve is also significant in analyzing the accuracy of a biometric system. The area under the ROC curve of an ideal biometric system is unity and it decreases with the increase in the overlap between the genuine and imposter distributions. In order to optimize the performance of a biometric system the area under the ROC curve must be maximized. Another graphical representation is the FAR-FRR diagram, which also shows the

threshold values in the operation of the system. An example FAR-FRR diagram is shown in Fig.1-6.

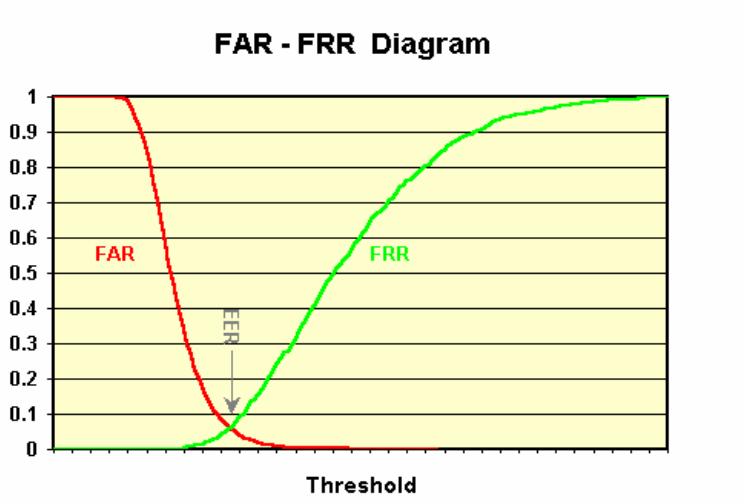

Fig. 1-6 An Example FAR-FRR Diagram

Another significant factor in evaluating the performance of a biometric system is the reject rate during enrollment and matching which is defined as the percentage of the biometric features rejected by the system because of any disagreement between the acquired feature and the mandatory requirements for that feature e.g. quality. This factor is important because by rejecting all the biometric features producing error in system operation one can always achieve an accuracy of 100%, which would be impractical. Other accuracy assessment measures can be found in [MMC+00].

An identification system is essentially a database retrieval system. In addition to the confidence level of the established identity, two other important accuracy measures, which characterize the retrieval accuracy of a database retrieval system are precision and recall. Precision is defined as the ratio of genuine records in the template database retrieved by the identification system and the total number of templates retrieved. Recall is defined as the ratio of the genuine records in the template database retrieved by the identification system and the total number of genuine records in the template database.

Apart from the accuracy measures, other factors such as the matching speed and the storage space requirements for the template database also affect the performance of a biometric system.

1.4 Project Objectives

The objective of the project is to develop a fingerprint based person identification and verification system. Fingerprints have been in use for person identification for over a century because of their high uniqueness and permanence. Fingerprints exhibit the following major advantages over other biometrics:

- Fingerprint verification is highly reliable and robust
- The validity and applicability of fingerprints has long been established
- It is the most commonly used biometric and has the potential to stay as the dominant biometric technique in the future

We have identified and addressed the following major issues in relation to fingerprint based person identification systems in this thesis:

a. Fingerprint Segmentation

Effective Segmentation of fingerprint images is an important step in the operation of a fingerprint based biometric authentication system. Segmentation refers to the separation of the fingerprint image foreground and background. The issues that affect the accuracy of a segmentation algorithm considerably include: The presence of oil and grease on the sensor surface, non-uniform contact and imprints left by a previous finger. The objective of conducting a study in this field is to analyze the problems in current segmentation algorithms and develop more robust algorithm, which are suited to the task of fingerprint segmentation.

b. Fingerprint Image Quality Evaluation

The matching performance of a fingerprint based person verification algorithm is deteriorated due to low quality fingerprint images. This degradation is caused by the extraction of false features in the noisy regions of fingerprints. The objective of developing an algorithm for the automatic evaluation of fingerprint image quality is to decrease the effect of these false features on the performance of the fingerprint matching process thus increasing the robustness of the matching process.

c. Fingerprint Enhancement

Fingerprint enhancement is an integral part of a fingerprint matching process and it is aimed at improving the visual quality for obtaining better matching performance. We have studied and implemented a number of fingerprint enhancement techniques and present a critical analysis of different approaches.

d. Fingerprint Matching

In fingerprint matching, the task is to compute a matching score between the given fingerprint and the stored template. The matching performance of a fingerprint-matching algorithm depends heavily on the feature-based representation of the fingerprints. It is desired that the features being used for the matching process should provide a compact and complete representation of the fingerprint print using which fingerprints belonging to different people can effectively and efficiently be separated. A study and implementation of different fingerprint matching techniques was carried out in the course of the project and results are described for analysis. An important step during fingerprint matching is the alignment or registration of the fingerprint images. In this thesis a novel approach to the coarse registration of fingerprint images is also presented.

e. Fingerprint Classification

Fingerprint classification is the task of assigning fingerprints into a number of classes or categories based on global ridge and furrow structure of the fingerprint. Fingerprint classification is used for database indexing in order to improve the matching speed during the process of fingerprint identification. Various algorithms for fingerprint classification were investigated during the project and a multi algorithm fusion based approach is presented along with its results.

1.5 Thesis Organization

The thesis is organized as follows: Chapter-2 gives the structural and functional design of a fingerprint based person identification system, after presenting a brief overview of the history and anatomy of fingerprints. It also describes the various processes involved in a fingerprint identification system with a special focus on fingerprint acquisition. Chapter-3 details about fingerprint classification and presents an in depth view of the classification schemes implemented during this project. Chapter-4 presents a novel approach to fingerprint segmentation. Chapter-5 presents various enhancement methodologies implemented in this project after giving a detailed review of the existing techniques. Chapter-6 focuses on fingerprint image quality evaluation, which can be used to, improve the matching performance of a fingerprint recognition system. Chapter-7 describes the various approaches to fingerprint matching. Chapter-8 is related to system development and the integration

of the algorithms described in previous chapters to make up fingerprint recognition system software. Chapter-9 presents the conclusions and future work.

Summary

Biometric identification is fast becoming the global standard for person identification and especially access control. Fingerprints are the most widely used biometric feature in the world because of their high permanence and uniqueness. A Typical biometric identification system consists of an acquisition module, an enrollment module and a recognition module. The performance of a biometric system is quantized in terms of False Acceptance Rate and False Rejection Rates, which are described graphically in the form of the ROC Curve. This project is aimed at exploring the techniques used for fingerprint based person identification and developing a fingerprint based person identification system software for use in access control applications.

2 Fingerprint based Person Recognition

A fingerprint is an impression of the curved lines of skin at the end of a finger that is left on a surface or made by pressing a finger against a surface. Fingerprints, because of their uniqueness, permanence and recoverability have been in use for fingerprint identification for a long time. In this chapter we present the history of fingerprints as biometrics, and study the anatomy, generation and uniqueness of fingerprints. We also point out the research possibilities in the field of fingerprint biometrics and present the general architecture of a fingerprint-based person identification and verification system along with its algorithm level design.

2.1 History of Fingerprints

Using unique characteristic traits for identification of an individual has been around since time immemorial. Tribe-members knew and recognized one another and that was the basis for deciding if someone belonged or not. The recognition was based on the characteristic traits that each of us is born with. The determination and codification of these unique characteristics has evolved into the science of biometrics.

Fig. 2-1 Some Prehistoric Fingerprint Images

Prehistoric

Picture writing of a hand with ridge patterns was discovered in Nova Scotia as shown in Fig. 2-1. In ancient Babylon, fingerprints were used on clay tablets for business

transactions. In ancient China, thumbprints were found on clay seals. In 14th century Persia, various official government papers had fingerprints (impressions), and one government official, a doctor, observed that no two fingerprints were exactly alike.

1686 - Malpighi

In 1686, Marcello Malpighi, a professor of anatomy at the University of Bologna, noted in his treatise; ridges, spirals and loops in fingerprints. However, he made no mention of their value as a tool for individual identification.

1823 - Purkinji

In 1823, John Evangelist Purkinji, a professor of anatomy at the University of Breslau, published his thesis discussing 9 fingerprint patterns, but he too made no mention of the value of fingerprints for personal identification.

1856 - Hershel

The English first began using fingerprints in July of 1858, when Sir William Herschel, Chief Magistrate of the Hooghly district in Jungipoor, India, first used fingerprints on native contracts and observed the uniqueness of fingerprints.

1880 - Faulds

During the 1870's, Dr. Henry Faulds, the British Surgeon-Superintendent of Tsukiji Hospital in Tokyo, Japan, took up the study of "skin-furrows" after noticing finger marks on specimens of "prehistoric" pottery. A learned and industrious man, Dr. Faulds not only recognized the importance of fingerprints as a means of identification, but devised a method of classification as well. He is also credited with the first fingerprint identification of a greasy fingerprint left on an alcohol bottle.

1882 - Thompson

In 1882, Gilbert Thompson of the U.S. Geological Survey in New Mexico, used his own fingerprints on a document to prevent forgery.

1888 - Galton

Sir Francis Galton, a British anthropologist began his observations of fingerprints as a means of identification in the 1880's.

1891 - Vucetich

Juan Vucetich, an Argentine Police Official, began the first fingerprint files based on Galton pattern types. At first, Vucetich included the Bertillon System with the files.

1892 - Vucetich & Galton

Juan Vucetich made the first criminal fingerprint identification in 1892. He was able to identify a woman by the name of Rojas, who had murdered her two sons,

and cut her own throat in an attempt to place blame on another. Her bloody print was left on a door post, proving her identity as the murderer.

Sir Francis Galton published his book, "Fingerprints", establishing the individuality and permanence of fingerprints. The book included the first classification system for fingerprints. Galton proved that fingerprints do not change over the course of an individual's lifetime, and that no two fingerprints are exactly the same. According to his calculations, the odds of two individual fingerprints being the same were 1 in 64 billion.

Galton identified the characteristics by which fingerprints can be identified. These same characteristics (minutia) are basically still in use today, and are often referred to as Galton's Details.

1897 - The Calcutta Anthropometric Bureau

On 12 June 1897, the Council of the Governor General of India approved a committee report that fingerprints should be used for classification of criminal records. Later that year, the Calcutta Anthropometric Bureau became the world's first Fingerprint Bureau.

1901 - Henry

Introduction of fingerprints for criminal identification in England and Wales, using Galton's observations and revised by Sir Edward Richard Henry.

1905-1918

Studies by the US Military on the individuality and permanence of fingerprint images indicated their effectiveness for use as biometric features for person identification.

1924 FBI Identification Division

In 1924, an act of congress established the Identification Division of the F.B.I. The National Bureau and Leavenworth consolidated to form the nucleus of the F.B.I. fingerprint files.

1946-1971

By 1946, the F.B.I. had processed 100 million fingerprint cards in manually maintained files; and by 1971, 200 million cards. With the introduction of AFIS technology, the files were split into computerized criminal files and manually maintained civil files. Many of the manual files were duplicates though, the records actually represented somewhere in the neighborhood of 25 to 30 million criminals, and an unknown number of individuals in the civil files. FBI, even today, remains to be the world’s largest keeper of fingerprints.

2004 FBI's IAFIS

The FBI's Integrated AFIS (IAFIS) in Clarksburg has more than 46 million individual computerized fingerprint records for known criminals. Old paper fingerprint cards for the civil files are still manually maintained in a warehouse facility, though most enlisted military service member fingerprint cards received after 1990, and all military-related fingerprint cards received after 19 May 2000, have now been computerized and can be searched internally by the FBI. In some future build of IAFIS, the FBI may make such civil file AFIS searches available to other federal crime laboratories. All US states and larger cities have their own AFIS databases, each with a subset of fingerprint records that is not stored in any other database. Thus, law enforcement fingerprint interface standards are very important to enable sharing records and mutual searches for identifying criminals.

2.2 Anatomy of Fingerprints

A fingerprint is an impression of the friction ridges found on the inner surface of a finger or a thumb. The most obvious structural characteristic of a fingerprint is the interleaved ridge-valley pattern as shown in Fig.2-2. Ridges and valleys can easily be discriminated in a medium to high quality fingerprint image by using their spectral intensity values; ridges are usually darker in intensity in comparison to the valleys. The width of a ridge varies from 100μm to 300μm. The ridge-valley cycle period is generally 500μm.

Fig. 2-2 Fingerprint Ridges and valleys

The uniqueness of fingerprints lies in the individuality of this pattern, which remains unaltered in case of injuries such as superficial burns, abrasions etc. since the original pattern is duplicated in any new skin that grows. However an injury that destroys the dermal papillae (which are peg like protuberances of the dermis or

connective tissue layer of the skin), will permanently obliterate the ridges. This damage can also be used in determining the uniqueness of individuals' fingerprints.

At a macroscopic level, a fingerprint is composed of a set of **ridgelines, which** often flow parallel and sometimes produce local macro-singularities called **whorl, loop** and **delta.** Sometimes a whorl type of singularity is not explicitly introduced because it can be described in terms of two facing loop singularities. The number of cores and deltas in a single fingerprint is regulated in nature by some stringent rules; fingerprints are usually partitioned into a number of classes (**arch, tented arch, left loop, right loop, whorl)** on the basis of their macro-singularities (for more details see Section 3.1). Such a classification is used to improve the search performance of an automatic fingerprint identification system. The center of the north most loop type singularity in a fingerprint image is called a **core**. Various fingerprint matching and classification algorithms are based on the core as a point of reference. For fingerprints that do not loop or whorl type of singularities (e.g. Arch type fingerprints), it is usually very difficult to locate the core. In such cases, the core is associated with the point of maximum ridgeline curvature. Unfortunately, due to the high variability in fingerprint patterns, it is difficult to reliably locate a reference point in all fingerprint images. Such reference points, as the core and the delta, play a vital role in various fingerprint analysis algorithms especially classification. The ridgeline flow can be effectively described by a structure called **directional map** (or directional image), which is a discrete matrix whose elements denote the orientation of the tangent to the ridgelines. Analogously, the ridgeline density can be synthesized by using a density map which gives a measure of the ridge frequency at a point that is defined as the inverse of the number of ridges per unit length along a hypothetical segment centered at that point and orthogonal to the local ridge orientation.

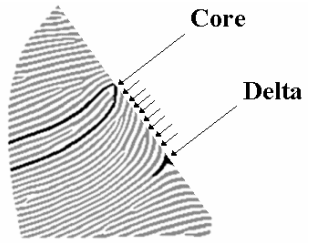

Fig. 2-3 Core and Delta in a Fingerprint Image

At a finer analysis other very important features can be discovered in the fingerprint patterns. These micro-singularities, called **minutiae** or Galton

characteristics, are essentially determined by the **termination** or the **bifurcation** of the ridgelines. Other types of such micro singularities include lake, spurs etc. which are shown in Fig. 2-4. Minutiae play a primary role for fingerprint matching, since most of the algorithms rely on the coincidence of minutiae to state whether two impressions are of the same finger or not. Minutiae matching, which is essentially a point pattern-matching problem, constitutes the basis of most of the automatic algorithms for fingerprint comparison.

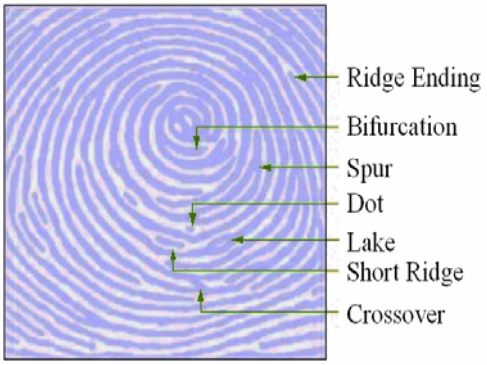

Fig. 2-4 Fingerprint Minutiae

In a negative image of a fingerprint, the corresponding minutiae take the same positions, but their type is exchanged: Ridge endings appear as bifurcations and vice versa. This property is known as the termination, bifurcation duality.

Each ridge of the epidermis is dotted with sweat pores along its entire length (see Fig. 2-5). These sweat pores range in size from 60-250μm. The arrangement of these sweat pores is highly distinctive and can be used for further improving the matching performance of an AFIS. However, the reliable detection of these sweat pores, requires very high-resolution (>1000dpi) scanners and good quality fingerprint images because of which very few matching techniques actually use these features.

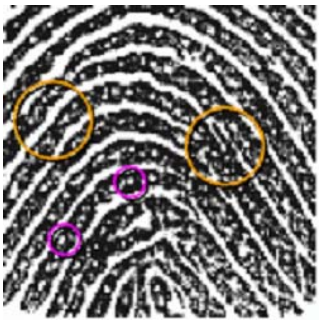

Fig. 2-5 Sweat Pores in Fingerprint Ridges (indicated by colored circles)

2.3 Uniqueness and Permanence of Fingerprints

The various characters of an organism are the consequence of the interaction of its genes and its environment. Physical appearance and fingerprints are, in general, a part of the individual's phenotype. In case of fingerprints, the general characteristics of the patterns are determined by the genes, however the cells on the fingertips grow in a microenvironment that is slightly different for different fingers. The finger details are determined by this changing microenvironment. Because of the large number of variations in the formation of fingerprints, it is virtually impossible for two fingerprints to be identical. However, as the fingerprint patterns are generated from the same genes, they are not totally random patterns. Thus it can be concluded that the fingerprint formation process is a chaotic system, rather than a random one, which results in the production of unique fingertip patterns for different fingers.

Dermatoglyphics studies indicate that the maximum difference between fingerprint types has been found among individuals of different races. Unrelated persons of the same race have very little generic similarity in their fingerprints, parent and child have some generic similarity as they share half the genes, siblings have more similarity and the maximum generic similarity is observed in identical (monozygotic) twins, which possess the closest genetic relationship. Various studies have shown that a significant correlation exists in the fingerprint class and other generic attributes of fingerprints such as ridge count, ridge width, ridge separation and ridge depth for identical twins. Identical twins are a consequence of the division of a single fertilized egg into two embryos. Thus, they have exactly identical DNA except for the generally undetectable micro mutations that begin as soon as the cell division starts. Fingerprints of identical twins start their development from the same DNA, so they show considerable similarity. However, identical twins are situated in different parts of the womb during fetus development, so each fetus encounters slightly different environment from their siblings. As a consequence, fingerprints of identical twins have different micro-details (e.g. minutiae), which can be used for identification purposes. It is claimed that a trained expert can usually differentiate between fingerprints of identical twins based on the minutiae dissimilarity measure [JPP00], which makes fingerprints more effective and efficient for person identification than DNA by the use of existing technologies.

However, the degree of uniqueness captured by fingerprint minutiae during the operation of an AFIS is still questionable. The theoretical probability that a fingerprint with 36 minutiae will share 12 minutiae with another fingerprint with 36 minutiae is 6.1x10-8 [PPJ02]. However the performance of today's minutiae based fingerprint matchers does not even come close to the theoretical maximum because of a number of issues relating to fingerprint quality, acquisition errors etc. However the combination of minutiae with other types of micro-details may improve the performance of a practical matching system. Various studies are being conducted for the systematic analysis of the uniqueness and individuality of fingerprints after a court ruling required that a scientific testimony for the uniqueness be established.

Fingerprints change very little with time; Fig. 2-6 shows the fingerprints after a person after a difference of 13 years, which exhibits very little change. However the permanence of fingerprints depends upon the nature of manual work being carried out by an individual. In general, the quality of the fingerprints degrades with age but the overall structure remains to the same.

Fig. 2-6 Permanence of Fingerprints Over Time

2.4 Tasks in an AFIS

A typical automatic fingerprint based person authentication system is associated with the following major tasks:

a. Fingerprint Classification

Fingerprint classification is the process of automatically assigning fingerprints into a number of classes or categories based on global ridge and furrow structure of the fingerprint. Fingerprint classification can be used during matching because two fingerprints belonging to different classes cannot be associated with the same finger.

Apart from its use in the matching algorithm, the main objective of fingerprint classification is to improve the performance of a fingerprint identification system by acting as a means of database indexing during the search process.

b. Fingerprint Matching

Fingerprint matching refers to the process of automatically establishing a matching score between two given fingerprints images. There are two problems in relation to fingerprint matching:

c. Fingerprint based Person Identification

Fingerprint based Identification is defined as the process of carrying out one-to-many comparison in order to search for the given fingerprint in the template database. The aim of this process is to establish the identity of the person without the subject having to claim an identity. For fingerprint-based identification both the speed and accuracy of the matching algorithm are important because the identification system needs to scan the entire database to establish an identity. The speed performance of the identification process can be improved by using fingerprint classification or other database indexing techniques.

d. Fingerprint based Person Verification

Verification refers to the process of authenticating the identity claim of a person by performing a 1-1 comparison of the given fingerprints with the subject's own pre-stored template. Unlike fingerprint based person identification, only the accuracy of the matching algorithm is of fundamental importance as the response time requirements are not at all difficult to meet because only 1-1 matching is conducted.

2.5 AFIS Architecture

Architecturally, A typical fingerprint based person identification and verification system comprises the following major modules:

- User Interface
- System (Template) Database
- Enrollment Module
- Classification Module
- Verification Module
- Identification Module

The objective of the user interface is to provide a means of communication for the user with the system. The user presents his or her fingerprints through the system interface after the desired operation (e.g. verification or identification) has been specified. An important part component of the user interface is fingerprint acquisition, which is discussed in detail in Section 2.6. The system (template) database is a repository, which contains the fingerprint template of a person along with his or her profile. The template of an individual is a compact representation of features extracted from fingerprint(s) of the individual after enrollment through the enrollment module. Depending upon the application, the system database may either be a physical database that resides in the system or a virtual database with the record of each individual being carried on a magnetic card issued to the individual. The feature matching algorithms present in the verification and identification modules produce a verification or authentication decision respectively, by comparing the features extracted from a given fingerprint image with the stored fingerprint templates. The fingerprint classification module can be used to provide the class information of a fingerprint to both the identification and verification modules, which may use it for database indexing or matching as has been explained earlier.

The specific design and architecture for a fingerprint based person identification and verification systems vary in a wide range depending upon the requirements such as security requirements, administrative and maintenance costs, performance requirements etc.

2.6 Fingerprint Acquisition

Fingerprints can be captured either online or offline depending upon the specific requirements of a fingerprint recognition system. Depending upon whether the acquisition process is online or offline, a fingerprint may be either (i) inked fingerprint or (ii) a live-scan fingerprint.

Inked fingerprint is a term, which is used to indicate that the fingerprint image is obtained from an impression of the finger on an intermediate media such as paper. Generally, inked fingerprints are obtained using the rolled method. A fingerprint captured using the rolled method is called a rolled fingerprint. An example of such a fingerprint is shown in Fig. 2-7.

Fig. 2-7 A Rolled Fingerprint Image

Rolled fingerprints are obtained by spreading a few drops of inks with a roller on a slab and then rolling the finger from one side of the nail to the other over the inked slab which inks the ridge patterns on top of the finger. After that the finger is rolled on a piece of white paper so that the inked impression of the ridge pattern of the finger appears on the paper. Rolled inked fingerprints impressed on paper can be electronically scanned into digital rolled fingerprints using optical scanners or video cameras. Rolled fingerprints have been a standard technique for the offline capture of fingerprints. Rolled inked fingerprints tend to have a large area of valid ridges and furrows, but have large deformations due to the inherent nature of the acquisition process. Direct feedback is not available to the subject to control the acquisition process, which, in turn, results in difficulties in quality control. Acquisition of rolled fingerprints is cumbersome and slow. In the context of an online automatic personal identification system, it is both infeasible and socially unacceptable to use the rolled inked method to acquire fingerprints in the operational phase.

In forensics, a special kind of inked fingerprints, called latent fingerprints, is of great interest. Constant perspiration exudation of sweat pores on fingerprint ridges and intermittent contact of a finger with other parts of the body and other objects leave a film of moisture and/or grease on the surface of fingers. In touching an object, the film of moisture and/or grease may be transferred to the object and leave an impression of the ridges thereon. This type of fingerprints is called latent fingerprint and such fingerprints are of fundamental importance in forensics. Now-a-days Special types of lights and recording hardware and software (as shown in Fig. 2-8) are used for the detection and capture of these latent fingerprints. An example of latent fingerprints is shown in Fig.2-9.

Fig. 2-8 Digital workstation for latent fingerprint recording (DCS-3)

Fig. 2-9 A latent Fingerprint (Left) and its Enrolled Match (Right)

The live-scan fingerprint is a collective term for a fingerprint image directly obtained from the finger without the intermediate step of getting an impression on a paper. A number of sensing mechanisms can be used to sense the ridge and furrows of the finger impressions, including (i) Optical Frustrated Total Internal Reflection (FTIR) [Har96, HS84, Kar89, Sto77], (ii) Ultrasound Total Internal Reflection [SW91], (iii) Optical total internal reflection of edge-lit holograms [SW91, Eas05, FSS92], (iv) Sensing of differential capacitance [MSW+96, HS98, Ver05] and non-contact 3D scanning [LKB+83]. These acquisition methods eliminate the intermediate digitization process of inked impressions and make it possible to build on-line systems. Depending on the clarity of ridge structures of scanned fingers and acquisition conditions, acquired live-scan fingerprints vary in quality. However, since there exists a direct feedback on such type of devices, it is relatively easier to control the quality of acquired fingerprints.

A live scan fingerprint is obtained using the dab method, in which a finger is impressed on the acquisition surface of a device without rolling. A dab live scan fingerprint only captures the ridges and furrows that are in contact with the

acquisition surface. Therefore, it tends to have a smaller area of valid ridges furrows and smaller deformation than a rolled fingerprint.

The most popular technology to obtain a live-scan fingerprint image is based on optical frustrated total internal reflection (FTIR) concept. When a finger is placed on one side of a glass prism, ridges of the finger are in contact with the prism, while the valleys of the finger are not in contact with the prism. The rest of the imaging system essentially consists of an assembly of an LED light sources and a CCD placed on the other side of the glass prism as shown in Fig. 2-10. The light source illuminates the glass at a certain angle and the camera is placed such that it can capture the laser light reflected from the glass. The light which is incident on the plate at the glass surface corresponding to valleys suffers total internal reflection, resulting in a corresponding fingerprint image on the imaging plane of the CCD. Multispectral Imaging techniques can be employed for spoof detection. The architecture of such a scanner is shown in Fig. 2-10 [SM05]. An example of live-scan fingerprint is shown in Fig. 2-14. Fig. shows some optical fingerprint scanners.

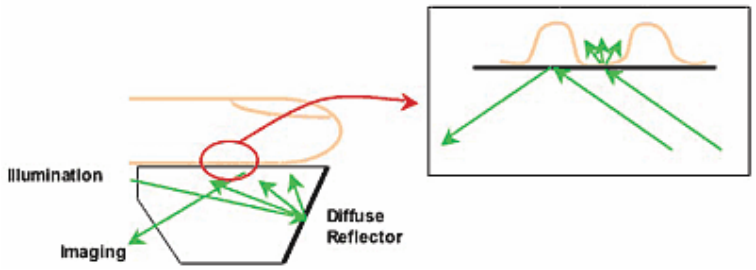

Fig. 2-10 Architecture of an Classical Optical Fingerprint Scanner

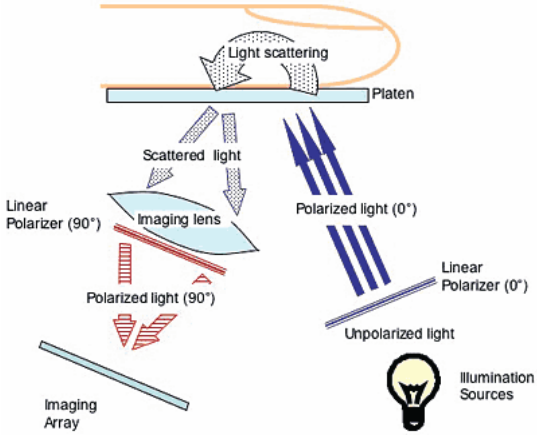

Fig. 2-11 Architecture of an Multispectral Optical Fingerprint Scanner

Fig. 2-12 Fingerprints Captured Using Classical and Multispectral Optical Fingerprint Scanners

Fig. 2-13 Digital Persona URU 500 (Left) and URU4000 (Right) Optical Scanners

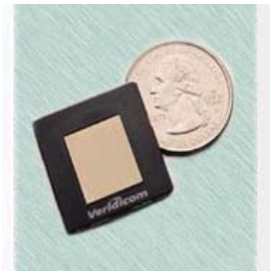

Fig. 2-14 Veridicom FPS 200 Fingerprint Sensor

Optical scanners are too large to be readily integrated in a number of applications such as laptop security, cellular phone security, and notebook security. Recently, a number of different types of compact solid-state fingerprint chips have become available. The quality of the images acquired using these solid-state chips is comparable to the quality of images acquired using optical scanners. These solid-state chips can be manufactured with a very low cost if manufactured in a large quantity. Fig. 2-14 shows the three different types of solid state fingerprint chips which are commercially available.

These solid-state chips utilize capacitative sensing of the pressure when a fingerprint is placed upon their surface and develop a digital fingerprint image. Like

optical scanners, capacitive fingerprint scanners generate an image of the ridges and valleys that make up a fingerprint. But instead of sensing the print using light, the capacitors use electrical current. Fig. 2-15 shows a simple capacitive sensor. The sensor is made up of one or more semiconductor chips containing an array of tiny cells. Each cell includes two conductor plates, covered with an insulating layer. The cells are tiny - smaller than the width of one ridge on a finger.

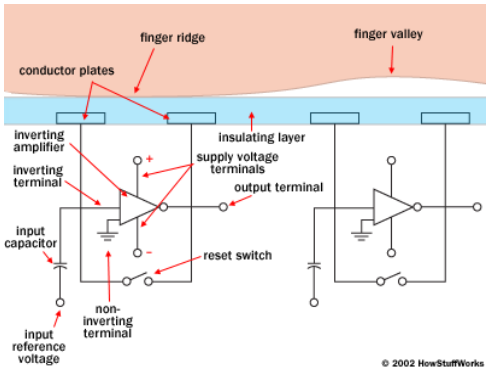

Fig. 2-15 A Capacitative Fingerprint Sensor

The two conductor plates form a basic capacitor. The surface of the finger acts as a third capacitor plate, separated by the insulating layers in the cell structure and, in the case of the fingerprint valleys, a pocket of air. Varying the distance between the capacitor plates (by moving the finger closer or farther away from the conducting plates) changes the total capacitance of the capacitor. Because of this quality, the capacitor in a cell under a ridge will have a greater capacitance than the capacitor in a cell under a valley, which is captured as variations in voltage levels after amplification by using operational amplifiers.

The scanner processor reads this voltage output and determines whether it is characteristic of a ridge or a valley. By reading every cell in the sensor array, the processor can put together an overall picture of the fingerprint, similar to the image captured by an optical scanner. The main advantage of a capacitive scanner is that it requires a real fingerprint-type shape, rather than the pattern of light and dark that makes up the visual impression of a fingerprint. This makes the system harder to be circumvented. Fig. 2-17 shows a fingerprint acquired from some capacitative scanners.

Fig. 2-16 Fingerprint Captured Using Capacitative Scanner

During the development of a fingerprint based person identification system, the various factors related to the fingerprint scanner include:

- Hardware Interface
- Frames per Second
- Automatic Finger Detection
- Encryption
- Supported Operating Systems
- Resolution
- Capture Area
- Price

A comparison of various fingerprint scanners available in the market is given below:Fig. 2-17 shows the images captured using different scanners.

Table 2-1 A Comparison of Different Fingerprint Scanners

	Technology	Company	Model	Dpi	Area (h×w)	Pixels
Optical	FTIR	Biometrika www.biometrika.it/eng/	FX2000	569	0.98"×0.52"	560×296 (165,760)
	FTIR	Digital Persona www.digitalpersona.com	UareU2000	440	0.67"×0.47"	316×228 (72,048)
	FTIR (sweep)	Kinetic Sciences www.kinetic.bc.ca	K-1000	up to 1000	0.002"×0.6"	2×900 (H×900)
	FTIR	Secugen www.secugen.com	Hamster	500	0.64"×0.54"	320×268 (85,760)
	Sheet prism	Identix www.identix.com	DFR 200	380	0.67"×0.67"	256×256 (65,535)
	Fiber optic	Delsy www.delsy.com	CMOS module	508	0.71"×0.47"	360×240 (86,400)
	Electro-optical	Ethentica www.ethentica.com	TactilSense T-FPM	403	0.76"×0.56"	306×226 (69,156)
Solid-state	Capacitive (sweep)	Fujitsu www.fme.fujitsu.com	MBF300	500	0.06"×0.51"	32×256 (H×256)
	Capacitive	Infineon www.infineon.com	FingerTip	513	0.56"×0.44"	288×224 (64,512)
	Capacitive	ST-Microelectronics us.st.com	TouchChip TCS1AD	508	0.71"×0.50"	360×256 (92,160)
	Capacitive	Veridicom www.veridicom.com	FPS110	500	0.60"×0.60"	300×300 (90,000)
	Thermal (sweep)	Atmel www.atmel.com	FingerChip AT77C101B	500	0.02"×0.55"	8×280 (H×280)
	Electric field	Authentec www.authentec.com	AES4000	250	0.38"×0.38"	96×96 (9,216)
	Piezoelectric	BMF www.bm-f.com	BLP-100	406	0.92"×"0.63	384×256 (98,304)

Fig. 2-17 Fingerprint Images of the same finger with ideal skin condition as acquired by different commercial scanners. a) Biometrika FX2000, b) Digital Persona URU 2000, c) Identix DFR200, d) Ethentica TctilSense T-FPM, e) ST-Microelectrnics TouchChip TCS1AD, f) Veridicom FPS110, g) Atmel FingerChip AT77C101B, h) Authentec AEF4000

2.7 Algorithm Level Design

The various steps involved in the operation of a fingerprint based person identification and verification system include:

- Fingerprint Acquisition
- Fingerprint Image Preprocessing
- Fingerprint Classification
- Feature Extraction
- Feature Matching

Fingerprint acquisition is the process of acquiring digital fingerprints and this step has already been detailed in Section 2.6.

Fingerprint image preprocessing includes operations such as fingerprint segmentation and enhancement. Fingerprint segmentation is defined as the process of separating the fingerprint image foreground and background. It is necessary because a large number of false features can be extracted in the background region of the

fingerprint if it is not removed. Moreover fingerprint segmentation decreases the temporal complexity of the feature extraction algorithm since the effective area of operation for the feature extraction algorithm is reduced. The fingerprint segmentation algorithm should have a minimum misclassification error in classifying a given region of the fingerprint image as foreground and background, which would ensure a compact and holistic representation of a fingerprint is being used in matching. Fingerprint image enhancement is used to improve the quality of the given fingerprint image. It's objective is to improve the contrast between the ridges and valleys of fingerprint and to increase ridge continuity. This operation results in more accurate feature extraction, which leads to better matching performance for an automatic fingerprint identification system. Fingerprint enhancement algorithms should be able to recover the maximum amount of information lost during the acquisition process while producing a minimum number of artifacts. Fig. 2-18 shows the results of the different steps in preprocessing.

Fingerprints have been traditionally classified into categories based in information in the global pattern of ridges. The generic shapes of the major fingerprint classes are shown in Fig. 2-19. In large scale fingerprint identification systems, elaborate methods of fingerprint classification systems are used to index individuals into bins based on classification of their fingerprints; these methods of binning eliminate the need to match an input fingerprint to the entire fingerprint database in identification applications and significantly reduce the computing requirements. A fingerprint classification algorithm should be invariant to rotation, translation and elastic distortion of the frictional information and should be robust to the loss of information during the acquisition process.

Fig. 2-18 Original, Segmented and Enhanced Images of a Fingerprint during Preprocessing

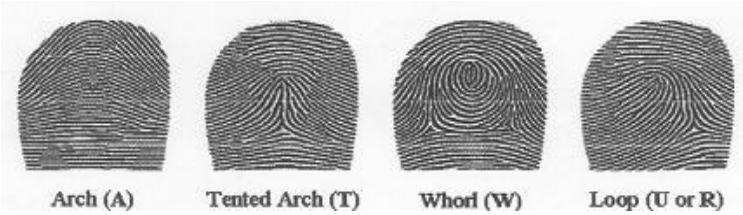

Fig. 2-19 Fingerprint Classes

The fingerprint representation (features) constitutes the essence of algorithmic level design and determines almost all aspects of the recognition mechanism. A representation should have the following two properties:

Saliency

Saliency means that a representation should contain enough class-specific information about the input data.

Suitability

Suitability means that the representation can be easily extracted, stored in a compact fashion and is useful for matching.

A matching algorithm is generally based on a similarity function to determine whether the two sets of features are from the same source. For a given representation deriving a similarity function is a complicated problem because of intraclass and interclass variations. The matching process/features must be invariant to translation, rotation and scaling and should be least affected by elastic distortions. Depending upon the specific application, the ability of the matching process to match partial fingerprints may also be desirable. The matching performance of an AFIS can be improved by the use of a fingerprint image quality evaluation algorithm which establishes the reliability and robustness of the features extracted in different regions of a given fingerprint image.

For automatic fingerprint identification, it is well known that he acquired image has redundancy and tends to have large intraclass variations which are caused by factors such as difference in placements of the finger on the sensor surface at different time, skin conditions, age, elastic distortions etc. Due to these variations, the fingerprint image itself is not a desirable representation. Currently, two major representation schemes for automatic fingerprint identification are:

Image based Representation

Such a representation assumes that the individuality of fingerprints may be exclusively determined in the spatial or frequency domain. For example, a fingerprint

can be represented by its Fourier spectrum or Wavelet coefficients. The image-based representations usually require that the input image be registered, which is in itself a complicated task and non-precise registration of fingerprints is a major source of error in the operation of such algorithms. Various image-based representations are detailed in [Bah96, JMT+75, JSW93, FHM91, FHM91]**.** Such global features are matched by using various types of classifiers which can be statistical, neural network based, or rule-based. Image based features are also referred to as global features.

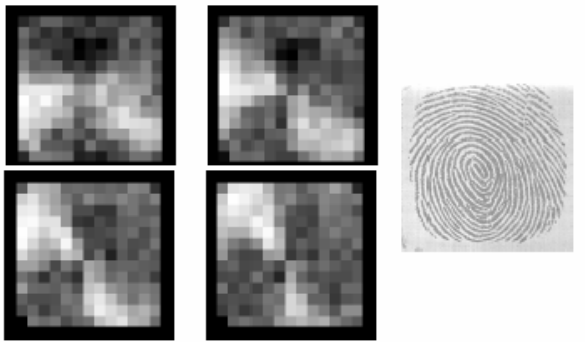

Fig. 2-20 Gabor Filter based Global Features for Fingerprint Identification (Ross, Jain and Reisman (2002))

Feature based Representation

Such a representation originates from the fact that if a pair of fingerprints belong to the same class and share a sufficiently large number of significant local ridge characteristics then it can be concluded that they are from the same finger. A large number of feature based methods which utilize different types of minute details, including minutiae, singular points, orientation field, ridge counts, ridge pores and ridges are present in the literature.

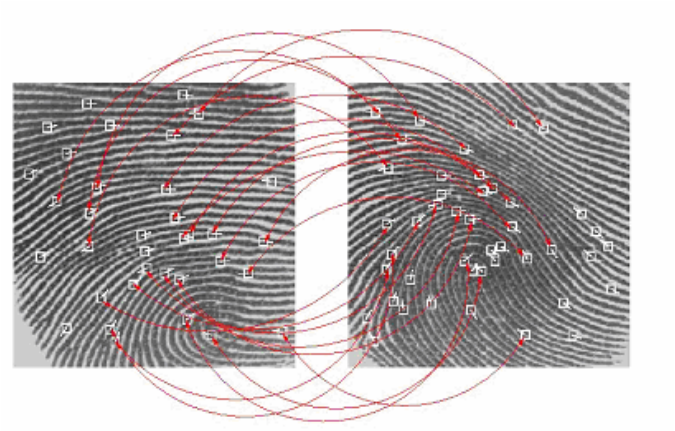

Fig. 2-21 Minutiae based Fingerprint Matching

Minutiae-based matching is the most popular and widely used technique, being the basis of fingerprint comparison made by fingerprint examiners. Minutiae are extracted from the two fingerprints and stored as set of points. Minutiae-based matching consists of finding the alignment between the template and the input minutiae sets that results in the maximum number of minutiae pairings. The number of matched minutiae sets is used to generate the matching decision, e.g. FBI guidelines dictate that for two fingerprints to be considered as matched, a minimum of 12 matched minutiae pairs should exist. Feature based representations are also called as local features.

2.8 Research Interests in Fingerprint Biometrics

While a significant progress has been made in automatic fingerprint identification, there are still a number of research issues, which need to be addressed to improve the system performance. Some of these problems are listed below:

a. Robust Live Scan Fingerprint Scanner

The quality of acquired fingerprint is critical to the performance of an AFIS. It is desired to have a more advanced live-scan fingerprint scanner that is able to tolerate different types of skin conditions and should be in responsive to elastic distortions of the skin.

b. Fingerprint Feature Extraction

In practice, a significant percentage of acquired fingerprint images are of poor quality. The performance of the feature extraction algorithms reported in the literature on different types of poor quality images is still far from desirable. To design a feature extraction algorithm that is robust to different types of image degradations is a challenge.

c. Fingerprint Segmentation

Despite the apparent simplicity of the task, segmentation of fingerprint images of low quality is a difficult problem, which leads to the degradation of the matching performance of an AFIS because of the extraction of false features in the background regions of the fingerprint. Moreover, in low quality fingerprint images, some of the image degradations are recovered during enhancement, but as the existing segmentation algorithms do not take into account this recovery effect inherent in the nature of the enhancement algorithm, the segmentation error is quite large. In this

thesis, we propose an efficient segmentation technique, which considers the nature of the enhancement algorithm and its recovery effect and results in a lower segmentation error.

d. Fingerprint Quality Evaluation

The processing of low quality fingerprint images requires an analysis of the image quality in different regions of the fingerprint, in order to establish the reliability of the feature extraction process in those regions. Despite its importance, in the face of the significant number of low quality images encountered by an AFIS, fingerprint image quality evaluation is still an open problem with very few algorithms existing in the literature for the process. In this thesis, we propose a neural network based quality evaluation algorithm, which also takes into consideration, the nature of the enhancement algorithm involved.

e. Fingerprint Enhancement

Fingerprint enhancement is aimed at the removal of image degradations for better feature extraction and matching. However, the design of a generic fingerprint enhancement algorithm, which is capable of processing all types of degradations, is a very challenging task.

f. Minutiae Matching

The performance of minutiae matching algorithms depends heavily on the reliability of minutiae and external alignment. To design a minutia-matching algorithm that is able to handle different situations such as a large percentage of spurious and missing minutiae and impression deformations is a demanding problem.

g. Fingerprint Classification

Although a number of automatic fingerprint classification methods have been proposed and some of them are used in operational AFIS, fingerprint classification still remains one of the most difficult problems for both humans and machines. Currently, the fingerprint classification framework is mainly intended for human experts, which may not be optimal for an automatic system. The current accuracy of classification algorithms is about 92% whereas it should be 99.9% for the scheme to be effective in database indexing.

h. Fingerprint Compression

Without a good fingerprint compression scheme, storing hundreds of millions of fingerprint is too expensive. A wavelet-based method (Wavelet Scalar Quantization), which has been proposed as the standard for fingerprint compression, can compress a

fingerprint image by a factor of 10 to 25 (See Fig. 2-22). An algorithm that can reach even higher compression ratio is an important research topic.

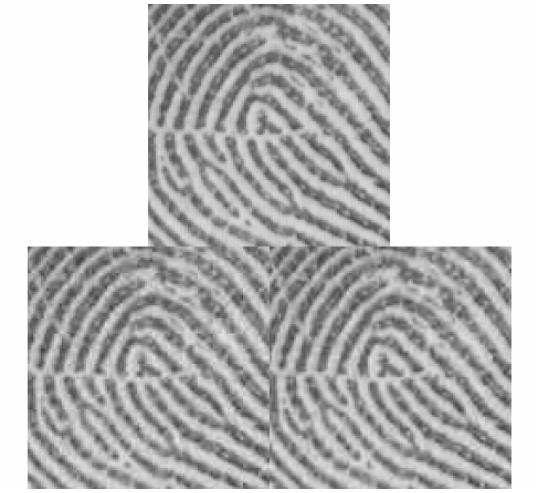

Fig. 2-22 Original Fingerprint (top), JPEG Compressed Fingerprint (Below, Left) and WSQ Compressed Fingerprint (Below, Right). The compression ratio is 12.9 for both the methods. The blocking in WSQ Compressed Images is Significantly Lower.

i. Computational Complexity of Matching

Computational complexity is a very important issue in AFIS. It is a practical requirement that all verifications should be performed in real time for all online applications. However, to achieve both high accuracy and high speed poses another difficulty.

j. Integration of Global and Local Features for fingerprint Matching

The existing matching algorithms are of two types, local and global feature based matching. The optimal combination of these features can lead to better matching performance. However, such a combination is still an open research problem in fingerprint-based biometrics.

All the above-mentioned issues affect each other greatly and the improvements in each can result in a much-improved system.

Summary

Fingerprints have been effectively in use for person identification for about 100 years. Fingerprints consist of ridge-valley patterns, which are characterized by the existence of both macro (loops and deltas) and micro singularities (minutiae). The general level

shape based classification of fingerprints is carried out by the use of the macro singularities where as the micro singularities are used for person identification. Fingerprints of any two different fingers are unique and they possess a high permanence which makes them ideal for use as a biometric identifier. A fingerprint identification system carries out the tasks of fingerprint classification, fingerprint verification and identification. Fingerprint Acquisition also an integral part for a fingerprint identification system. Fingerprint acquisition can be carried out both online and offline. In practical access control applications of fingerprints, online fingerprint scanners are employed instead of offline techniques. Fingerprint recognition involves the following major steps; Acquisition, preprocessing, classification, feature extraction and matching. The recognition of fingerprints and their classification are among the most challenging problems in pattern matching, which invokes a considerable research interest in the scientific arenas.

3 Fingerprint Classification

Fingerprint database indexing is an important step in the development of an automatic fingerprint identification system. Indexing is used to improve the matching performance of an AFIS by reducing the response time during the identification process. In this chapter we describe in detail various techniques for database indexing and fingerprint classification.

The identification of a person requires a comparison of his or her fingerprints with all the fingerprints in a database. The database may be very large (e.g. several million fingerprints) in many forensic and civilian applications. In such cases, the identification typically has an unacceptably high response time. The execution speed of the matching process can be improved by reducing the number of required comparisons. Sometimes, information about gender, race, age etc. is available for the individuals in the database, which can reduce the search space in identification significantly. However such information is not always available (e.g. in the forensic use of an AFIS) and in the general case, information intrinsic to the biometric samples has to be used for an efficient retrieval. A common strategy to achieve this is to divide the fingerprint database into a number of bins (based on some predefined classes). A fingerprint to be identified is then required to be compared only to the fingerprints in a single bin of the database based on its class. This process of improving the retrieval performance of an AFIS is called indexing. Fingerprint classification is one mechanism of carrying out database indexing.

Fingerprint classification refers to the problem of assigning a fingerprint to a class in a consistent and reliable way. Fingerprint classification uses global features such as ridge structure and singularities for assigning a fingerprint to a particular class.

3.1 Manual Fingerprint Classification

The first fingerprint classification rules were proposed by Purkinje in 1823, which classified fingerprints into nine categories i.e. traverse curve, central longitudinal stria, oblique loop, almond whorl, spiral whorl, ellipse, circle, and double whorl. However the most popular classification scheme currently in use worldwide was

originally proposed by Galton in 1892 and was extended by Edward Henry in 1902. The Galton-Henry classification scheme divides fingerprints into five most common classes i.e. arch, tented arch, left loop, right loop and whorl. These classes are shown pictorially in Fig. 3-1.

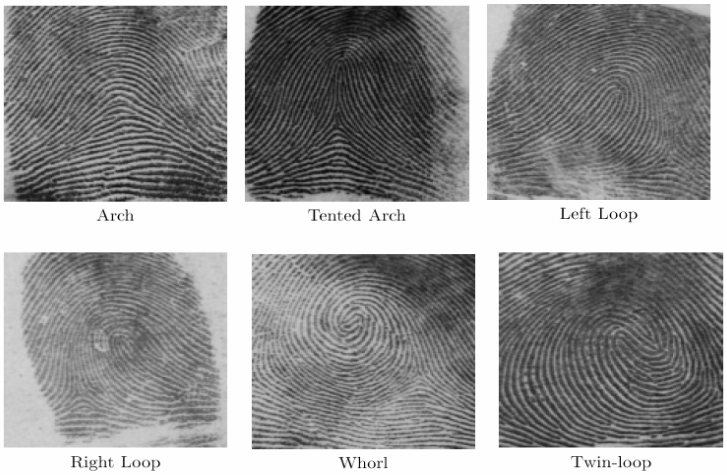

Fig. 3-1 Fingerprint Classes

The rules for classifying fingerprints according to the Galton-Henry classification system are given below:

Arch

An arch fingerprint has ridges that enter from one side, rise to a small bump, and go out the opposite side from which they entered. Arches do not have loops or deltas.

Tented Arch

The tented arch fingerprint is similar to the arch, except that at least one ridge exhibits a high curvature and a loop and a delta singularity are present in a tented arch.

Loop

A loop fingerprint has one or more ridges that enter from one side, curve back, and go out the same side they entered. A loop and a delta singularity are present; the delta is assumed to be south of the loop. Loops are further subdivided into left and right loops. Ridges in a left loop fingerprint enter and leave from the left side whereas in a right loop ridges enter and leave from the right side.

Whorl

A whorl type of fingerprint contains at least one ridge that makes a complete 360-degree path around the center of the fingerprint. Two loops (or a whorl) and two deltas can be found in whorl fingerprints. The whorl class is quite complex and in some classification schemes, it is further divided into two categories: twin loop (or

double loop) and plain whorl. In a plain whorl the line joining the two loops is parallel to the ridges between the two loops whereas the line is perpendicular to the ridges in a twin loop.

Arch, tented arch and the loop class of fingerprints are collectively called as lasso class whereas whorl and twin loop fingerprints are known as wirbel class.

Manual classification of fingerprints is carried out by using the above-mentioned rules. However accurate manual classification requires considerable experience of working with fingerprints, because of very high intra class variability among fingerprint classes. Efforts for automatic classification of fingerprint images are currently aimed at mimicking the manual process.

3.2 Issues in Automatic Fingerprint Classification

Automatic Classification of fingerprint images is a difficult problem because of the following reasons:

a. Low Inter-Class and High Intra-Class Variability

The variability within a fingerprint class in quite large as shown in Fig. 3-2 in which all the fingerprints belong to the same (whorl) class but are quite different in their apparent structure. Fingerprint classes in the Galton-Henry system have low variations between various classes as shown in Fig. 3-2 in which all fingerprints belong to different classes but apparently look alike. The small inter-class variation and the large intra-class variation in fingerprint classes, makes the task of automatic fingerprint classification intricate.

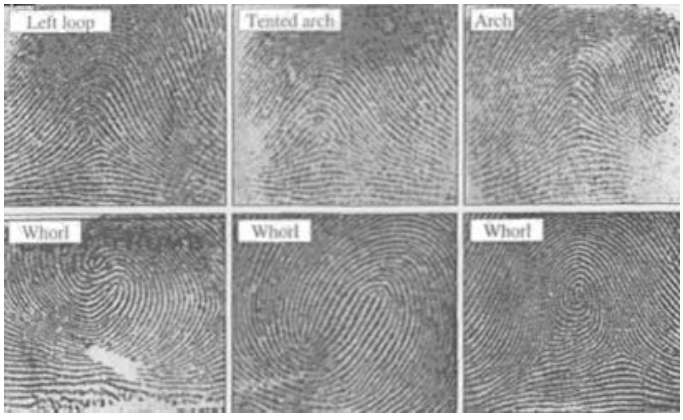

Fig. 3-2 Fingerprints belonging to different classes having a very similar appearance (top 3 images); Fingerprint belong to the same class having different appearance (bottom 3 images)

b. Low Quality of Fingerprint Images

Low quality is another issue in fingerprint classification because it gravely affects the performance of various classification algorithms. Fig. 3-3 shows some poor quality fingerprints for which accurate classification is very difficult.

Fig. 3-3 Low Quality Fingerprint Images

Because of the above-mentioned problems the accuracy of the best available fingerprint classification system is about 94%, which is far from the one required for reliable and effective database indexing.

3.3 Requirements for Fingerprint Classification

For automatic fingerprint classification, the following five issues are of great interest

a. Number of fingerprint categories

The number of categories in fingerprint classification should be sufficiently large as this would significantly narrow down the search and lead to better database indexing.

b. Distribution of fingerprints among the categories

Fingerprint should be uniformly distributed among the defined categories, the more uniformly the fingerprints are distributed among these categories; the more effective the resulting indexing mechanism.

c. Consistency of classification scheme

Consistency refers to the requirement for the classification scheme that the fingerprints in each category should be similar in terms of global pattern configuration.

d. Classification accuracy

For the classification scheme to be effective it must have a significantly high accuracy as incorrect classification would lead to false rejections which deteriorate the matching performance of an AFIS.

e. Computational requirements of the classification algorithm

The execution speed for a classification must be greater than the time for a linear search of the database. Otherwise the classification scheme would be rendered useless.

Unfortunately, in the Galton-Henry classification system there are only five classes, which can prove to be insufficient if the fingerprint database is very large. Moreover, fingerprints tend to be non-uniformly distributed among these classes as shown in Fig. 3-4. Many ambiguous fingerprints also exist whose exclusive membership in a fingerprint class cannot be reliably established, even by human experts, which leads to classification errors thus degrading the classification accuracy. There in applications where there is no need to comply with an existing classification schema, other classification and indexing schemes have been proposed, which are summarized in Section 3.10.

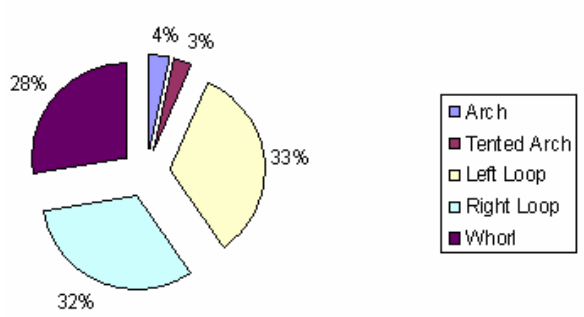

Fig. 3-4 Distribution of Fingerprint Classes

3.4 Features used for Fingerprint Classification

Fingerprint classification has attracted a significant amount of interest in the scientific community due to its importance and intrinsic difficulty, and a large number of research papers have been published on this topic in the last 30 years.

Although a wide variety of classification algorithms have been developed for this problem, a relatively small number of features extracted from fingerprint images have been used by most of the authors. In particular, almost all the methods are based on one or more of the following features:

a. Ridge Line Flow

Ridge flow is usually represented as a set of curves running parallel to the ridgelines; these curves do not necessarily coincide with fingerprint ridges and valleys, but they

exhibit the same local orientation. The ridgeline flow can be traced by drawing curves locally oriented according to the orientation images [CGW+95]. [CNJ+97] model the ridge flow lines in a fingerprint as B-Splines and adjacent ridges are merged to reduce the effects of noise during classification.

b. Orientation Image

Most of the existing fingerprint classification approaches make use of the orientation image because an accurate and detailed orientation image of a fingerprint contains all the information required for classification. The orientation image of a fingerprint is formed by calculating the local orientation of the fingerprint ridges at different pixels in a fingerprint, which is defined as the angle that the fingerprint ridges, crossing through an arbitrary small neighborhood centered at that pixel form with the horizontal. An in-depth review of different methods for orientation estimation is given in Section 5.3.1. Usually the orientation image is registered with respect to the core point before further processing. Fig. 3-5 shows the orientation fields for fingerprints belonging to different classes.

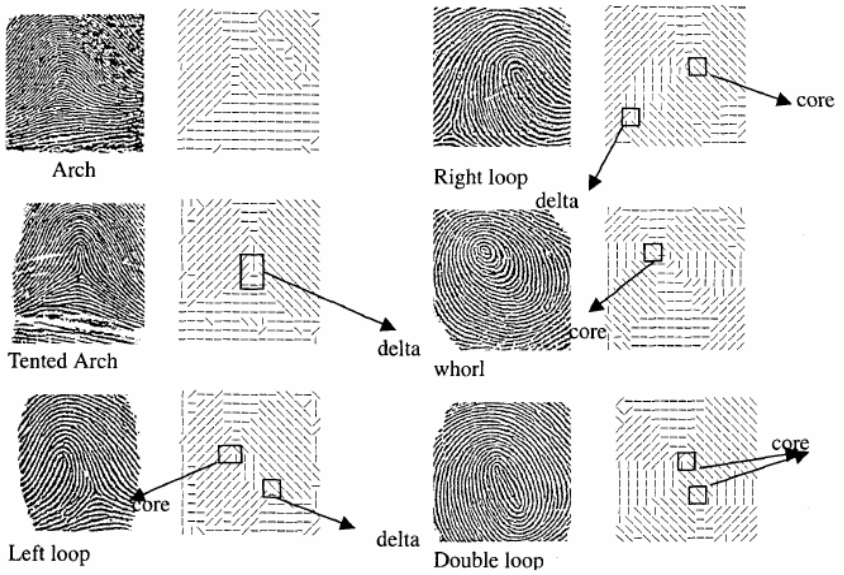

Fig. 3-5 Orientation Field for Different Classes

c. Singular points

The number and configuration of the singular points (see Section 2.2) in a fingerprint image can also be used for fingerprint classification. Fig. 3-6 shows the singular points in different classes of fingerprints.

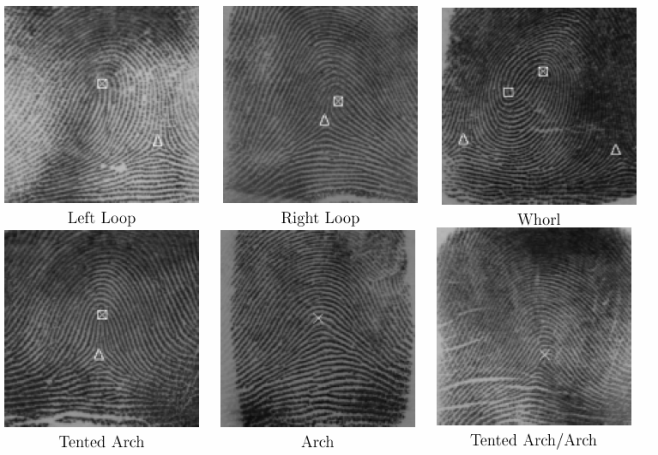

Fig. 3-6 Loops (squares) and Deltas (Triangles) in Different Fingerprint Classes

d. Filter Responses

Filter based approaches are used to capture the information present in the given fingerprint and especially in its orientation field which can be used for classification. Usually Gabor filters are employed for this purpose. Gabor filters exhibit both orientation specific and frequency specific behaviors, which can be used to analyze a fingerprint image therefore information inherent in the directional fields of fingerprints in relation to classification can be captured by the use of Gabor filters. In such methods, the given fingerprint is decomposed into a multiple component images by applying Gabor filters tuned to different orientations and frequencies. Typically four Gabor filters corresponding to four different orientations (0, 45, 90 and 135-degrees) are used. Various approaches utilizing Gabor filter for fingerprint classification include [JPH99, MRF01, YFP01].

3.5 A Survey of Fingerprint Classification Techniques

Most of the existing fingerprint classification methods can be coarsely assigned to one of following categories:

a. Syntactic Approaches

Syntactic approaches are traditional methods for the classification of fingerprints. In such approaches patterns are described by means of terminal symbols and production rules. Termination symbols are associated to a small group of directional elements within the fingerprint directional image. A grammar is defined for each class and a

parsing process is responsible for classifying each new pattern. Examples of syntactic approaches for fingerprint classification include [MF75, MF76 and RB80].

In general due to the great diversity of fingerprint patterns, syntactic approaches require very complex grammars whose inference requires complicated and unstable approaches; for this reason the use of syntactic approaches for fingerprint classification has been almost abandoned with a few exception [CF02].

b. Rule Based Approaches

Heuristic criterion based on the number and position of the singularities can be used to classify fingerprints. Other local features, such as ridgeline shape and local orientations, are exploited to improve performance. Singularities based approaches to fingerprint classification are commonly used by human experts; therefore several authors [KT84, MT93, CGW+95, KJ96, HJ99, Sen01, and JM02] adopt the same technique for fingerprint classification.

Since rule based methods relies heavily on the presence of singular points, some problems arise in the presence of noise or partial fingerprints, where singularity detection is complicated. We also use a rule-based approach for fingerprint classification, which is detailed later in the chapter. This approach is optimized for use with live-scan fingerprints in which some of the singular points especially deltas are frequently missing because of small capture areas of the fingerprint scanners.

c. Structural Approaches

These approaches are based on the mapping of low-level features, e.g. fingerprint ridge orientations, into high-level structures, which is represented by the use of data structures such as trees and graphs. A structural approach in [MM96] is based on the partitioning of the orientation image by minimizing a cost function associated with the variance of the element orientations in each region. A graph matching approach is used for obtaining the classification decision. Other similar approaches include [Sen97, CMM99b, LMM99, MRF01, and Sen01].

The main advantage of structural approaches is that, because of their reliance on global information only, these approaches are able to deal with partial and low quality fingerprints where singular points are either completely unavailable or their extraction is difficult. Moreover these approaches exhibit invariance to rotation and displacement. However it is not always convenient to partition an orientation field, especially for poor quality fingerprints.

d. Stochastic Approaches

These approaches are based on the development of probabilistic models for various features extracted in different regions of a fingerprint image. These models are then used for fingerprint classification. An example is the method proposed in [Sen97] in which two-dimensional Hidden Markov Models (HMM) are used for analyzing and classifying fingerprint images.

e. Statistical Approaches

In statistical approaches, a fixed size numerical feature vector is derived from each fingerprint and a general-purpose statistical classifier is used for the classification.

Examples of these approaches include Fitz et al. [FG96] who used a k-NN classifier for fingerprint classification based on wedge-ring features obtained from a hexagonal Fourier transform of a given fingerprint image, Capelli et al. [CMM99A] used the Karhunen-Loeve (KL) transform for capturing the significant components from the directional fields of fingerprints and used them for classification. Multi-space Karhunen-Loeve transform has also been used for classification [CMM00].

f. Neural Network based Approaches

Neural network approaches are mostly based on multi-layer perceptrons or Kohonen self-organizing networks and use the elements of the orientation image as input features. Various approaches using a neural network classifier for fingerprint classification include [CGW+95, WYJ98, JPH99, BBV+01, PPB+01, MRF01, Sen01]. Some researchers such as [CGW+95, WYJ98] have used probabilistic neural networks for performing fingerprint classification.

g. Multi-classifier Approaches

Multiple classifiers using different or similar features can be combined to improve the performance of an automatic fingerprint classification system. A very well known multi-classifier approach is PCASYS (Pattern-level Classification Automation SYStem), which combines a probabilistic neural network with a ridge-tracing module to perform fingerprint classification (see Fig. 3-7). PCASYS is primarily based on the orientation image of a fingerprint, which undergoes dimensional reduction through the KL transform. Various improvements have been proposed to PCASYS which include the combination of PCASYS with two other classifiers [Sen01], use of a feedback mechanism based on a genetic algorithm to automatically select the best parameters for PCASYS [WYJ98] and the adaptation of a feature extraction method for PCASYS [PPB+01].

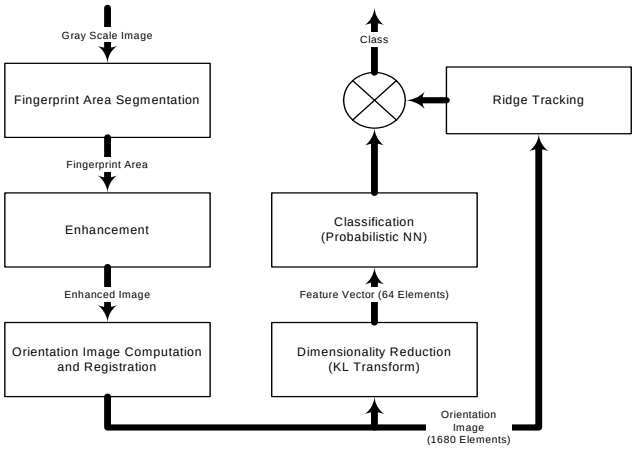

Fig. 3-7 PCASYS Architecture

Other examples of multi-classifier fusion include [JPH99, CMM00, MRF01 and YFP01.]

3.6 Performance Evaluation for Classification Systems

The performance of a fingerprint classification system in analyzed in terms of the error rate in the classification of fingerprints to different classes [MMJ+03]. The error rate is defined as the percentage ratio between the number of misclassified fingerprints and the total number of test samples. Another measure, i.e. classification accuracy is sometimes used which is the percentage of correctly classified fingerprints:

$$\text{error rate} = \frac{\text{number of misclassified fingerprints} \times 100}{\text{total number of fingerprints}}\% \qquad (3.1)$$

$$\text{accuracy} = 100\% - \text{error rate} \qquad (3.2)$$

The error rate of a classification system is generally specified as the function of the percentage of the database that the system has to search, which is known as penetration rate and is defined as:

$$\text{penetration rate} = \frac{\text{number of accessed fingerprints} \times 100}{\text{total number of fingerprints in the database}}\% \qquad (3.3)$$

As the distribution of fingerprint classes is not uniform, therefore a weighted scheme for accuracy assessment is sometimes adopted which uses the natural distributions of each class to weight the results.

An in depth view of the classification errors made by a fingerprint classification algorithm is provided by the use of a confusion matrix which is a two dimensional matrix whose rows represent true classes with the columns denoting the hypothesized classes. Each cell of the confusion matrix tells us how many fingerprints belonging to class r are assigned incorrectly to class c.

Sometimes a confidence level is also associated with the decision of a classifier, which improves the accuracy by assigning fingerprints with rejecting low confidence levels based on a certain threshold criterion. When such as scheme is employed, the accuracy of the fingerprint classification algorithm is analyzed by using a plot by taking the reject rate on one axis and the accuracy (or the error rate) on the other.

NIST has provided special databases for the performance evaluation of fingerprint classification algorithms, which include NIST DB4 and DB14 [WW92, WW93]. Both these databases contain 8-bit gray level images of rolled fingerprint impressions. DB4 contains about 2000 fingerprints whereas DB14 contains 27000 fingerprint pairs. These databases have become de facto standards for the performance evaluation of automatic fingerprint classification algorithms. However, these databases are not well suited for testing fingerprint classification on live-scan images, which usually do not contain all the singular points whereas these databases contain complete images of the rolled fingerprints. Table 3-1, 3-2 and 3-3 show the results of some of the earlier mentioned classification algorithms on the NIST databases.

Table 3-1 Error Rates on NIST DB4.

Method	Test Set	5 Classes		4 Classes	
		%	Weighted (%)	%	Weighted (%)
CGW+95	Second Half	-	-	11.4	6.1
KJ96	Whole DB	14.6	11.9	8.6	9.4
Sen97	Random 542	-	-	-	8.4
CMM99	Second Half	7.9	6.5	5.5	-
HJ99	Whole DB	12.5	10.6	7.7	-
JPH99	Second Half	10.0	7.0	5.2	-
MRF01	Second Half	12.1	9.6	-	-
Sen01	Second Half	-	-	-	5.1
YFP01	Second Half	10.7	9.0	6.9	-
JM02	Whole DB	-	-	8.8	9.3

Table 3-2 Confusion Matrix of the results on DB4 for the approach proposed in Cappelli, Maio and Maltoni (1999) [CMM99b]

True Class	Hypothesized Class				
	A	**L**	**R**	**W**	**T**
A	**420**	6	3	1	11
L	3	**376**	3	9	11
R	5	1	**392**	6	16
W	2	5	14	**377**	1
T	33	18	9	0	**278**

Table 3-3 Error Rates on NIST DB14

Method	Error Rate (%)
CGW+95	7.8
WYJ98	6.0
CMM00	5.6

3.7 Singular Point Extraction Techniques

We have used a rule-based approach to fingerprint classification, which relies heavily on the reliable detection of singular points in fingerprints therefore we present a detailed review of the different available schemes for singularity extraction in fingerprints in this Section.

3.7.1 Objectives of Singular Point Extraction

The objective a singular point extraction algorithm is to find out the spatial location of different singularities (loop and delta) in a given fingerprint image (see Section 2.2) as shown in Fig. 3-8.

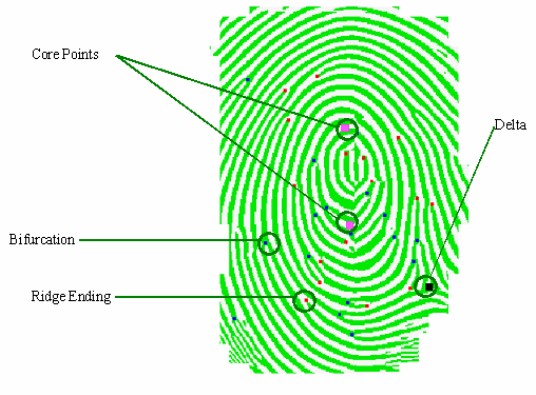

Fig. 3-8 Singular Points (Core and Delta) in a Fingerprint Image

Singular points are used not only for the purpose of rule-based fingerprint classification but also act as registration marks for a variety of classification, feature extraction and matching algorithms.

3.7.2 A Survey Of Different Techniques For Singular Point Extraction

Various approaches to singular point extraction in fingerprints can be broadly categorized as [MMJ+03]:

a. Poincare Index based Methods

The Poincare index [KT84] on the orientation field can be used to extract the core (loop) and delta points in fingerprints. A digital closed curve ψ about 25 pixels long, around each pixels is used to compute the Poincare index, which is defined as:

$$Poincare(i, j) = \frac{1}{2\pi}\sum_{k=0}^{N_\psi}\Delta(k) \tag{3.4}$$

where

$$\Delta(k) = \begin{cases} \delta(k) & \text{if } |\delta(k)| < \pi/2 \\ \pi + \delta(k) & \text{if } \delta(k) < \pi/2 \\ \pi - \delta(k) & \text{otherwise} \end{cases}$$

$$\delta(k) = O'(\psi_x(i'), \psi_y(i')) - O'(\psi_x(i), \psi_y(i))$$

$$i' = (i+1) \bmod N_\psi$$

O is the orientation field, and $\psi_x(i)$ and $\psi_y(i)$ denote coordinates of the i^{th} point on the arc length parameterized closed curve ψ. In case of singularities:

$$Poincare(i, j) = \begin{cases} 0 & \text{if [i,j] is not a singular point} \\ 1/2 & \text{if [i,j] is a loop singularity} \\ -1/2 & \text{if [i,j] is a delta singularity} \\ 1 & \text{if [i,j] is a whorl singularity} \end{cases}$$

Other Poincare index based approaches include [BG02b, KJ96]. Singularity detection in noisy or low quality fingerprints is difficult and the Poincare method may lead to the detection of false singularities. Regularizing the orientation image through a local averaging is quite effective in preventing the detection of false singularities.

b. Approaches based on the Local Characteristics of the Fingerprint Orientation Field

Singularities in a fingerprint image are characterized by the non-existence of a dominant ridge and the presence of a high curvature of the ridges. This characteristic can be used for the extraction of fingerprint singularities. Approaches based on these local characteristics of the ridge orientations include [SM92, CMM99b, KK01 and Nilsson and Bigun (2000a, b)]. [SM92] extract singularities according to the local histogram of the orientation image. Capelli et al. [CMM99b] propose an irregularity operator (given below), which analyzes the irregularities in the ridge orientations of the fingerprint image and is able to coarsely locate singular regions.

$$irregularity(i,j) = 1 - \frac{\left\| \sum_{h=-1\ldots1} \sum_{k=-1\ldots1} d_{i+h,j+k} \right\|}{\sum_{h=-1\ldots1} \sum_{k=-1\ldots1} \left\| d_{i+h,j+k} \right\|} \tag{3.5}$$

Here d_{ij} is the ridge direction at (i,j). If all the ridges in are parallel, then the irregularity operator exhibits a minimum irregularity of 0, and its value approaches 1 as irregularities in ridge orientation increase. [TK99, and KK01] propose multiresolution analysis approaches for the orientation field of a given fingerprint image to extract singularities. In [TK99], a directional matrix is used for the extraction of singular points.

c. Partitioning Based Methods

The singular points in fingerprints can be extracted by partitioning a fingerprint orientation image into regions characterized by homogenous orientations. This portioning can be achieved on the basis of discretization of a orientation image or a dynamic clustering algorithm. Some partitioning based approaches are [HH96, MM96 and CMM99b and RTO01].

3.8 Implemented Schemes for Classification

Our approach to fingerprint classification is primarily based on a combination of two rule-based schemes for fingerprint classification [KJ96 and BJJ+00.]. This approach is optimized for uses with live scan fingerprints in which core points (esp. the deltas) are frequently missing because of the small capture-area of the fingerprint acquisition device. Examples of such fingerprints are shown in Fig. 3-9.

Fig. 3-9 Fingerprints in which the delta singularities are missing

The various steps involved in fingerprint classification include:

a. Singular Point Extraction
b. Ridge Orientation Estimation
c. Quantization of the Orientation Image
d. Core Analysis
e. Rule Based Classification

These steps are explained in detail below:

3.8.1 Singular Point Extraction

Singular point extraction is aimed at the detection and extraction of core (loops) and delta within a fingerprint image. For this purpose we utilize the technique provided by RTO01. This method is based on an efficient directional image calculation method. It detects the singular points in an image by using signum change curves in the directional image. This algorithm consists of the following major steps:

3.8.1.1 Calculation of Pseudo Ridge Orientation Matrices

The first step in singular point detection is the calculation of pseudo-ridge orientation matrices denoted by PX_{ij} and PY_{ij} by using the equations given below:

$$PX_{ij} = \left|DY_{ij}\right| - \left|DX_{ij}\right| \tag{3.6}$$

$$PY_{ij} = DY_{ij} \cdot DX_{ij} \tag{3.7}$$

Where DX and DY are the matrices containing the horizontal and vertical partial derivatives of the image and are given by:

$$DX_{ij} = A_{i,j+1} - A_{i,j-1} \tag{3.8}$$

$$DX_{ij} = A_{i+1,j} - A_{i-1,j} \tag{3.9}$$

Where $A_{i,j}$ is the given fingerprint image.

These pseudo ridge orientations are reduced forms of the final ridge orientation in vector forms. These pseudo ridge orientations are smoothed by the use of a gaussian low pass-smoothing filter with size $[w_g \quad w_g]$ and variance σ_g. Fig. 3-10 shows these pseudo ridge orientations pictorially.

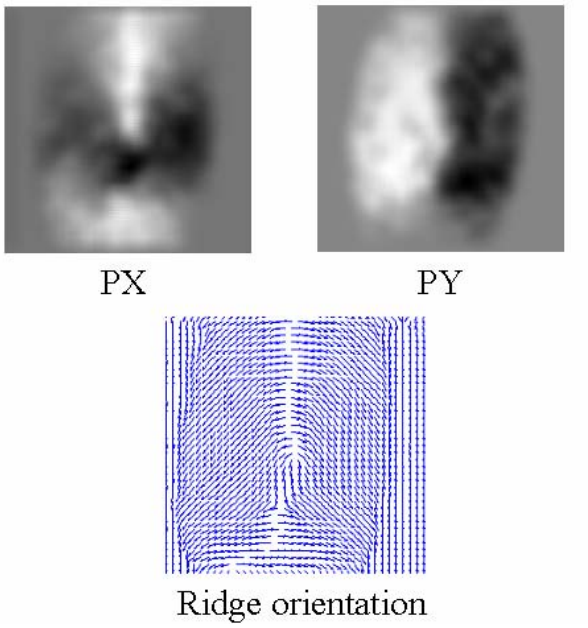

Fig. 3-10 Calculation of Pseudo Ridge Orientation Matrices

3.8.1.2 Determination of Congruencies in Pseudo Ridge Orientation Matrices

Singular points are detected by calculating the congruencies in PX and PY. PX_{ij} is zero when the average ridge orientation has rotated 45 or 135-degrees. Similarly PY_{ij} is zero when the orientation is horizontal or vertical. The non-zero values in PX and PY form distinct sign change curves in these matrices, which are known as transition lines. Singular points lie at the intersection of these lines as shown in Fig.3-11. These intersections can be found by extracting a 3x3-sized rectangle in every second point of both the matrices with the point under consideration at the center. If the minimum and maximum values of the block are of different sign, a transition line has been detected. If both of the matrices have a sign change at the same location, a

singular point has been found. Sometimes this technique yields improper singular points in adjacent blocks, which is avoided by discarding the singular point if another singular point has been found in the close neighborhood. The algorithm is also affected by noise in the border regions of the fingerprint. The effects of this noise are suppressed by applying a threshold T on the absolute values of PX and PY which tend to decrease in the border region. However, as these values also decrease near the core, therefore the threshold must be measured a few pixels (a>2) away from the point of interest.

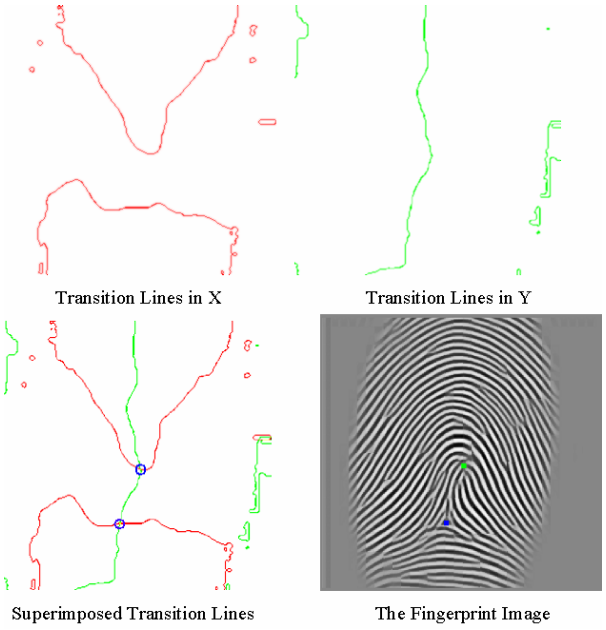

Fig. 3-11 Determination of Congruencies in Pseudo Ridge Orientation Matrices

3.8.1.3 Determination of the Type of a Singular Point

The type of the singular point at point (i, j) can be determined by using the information in which directions the sign changes are. The gradient PX at (i, j) is nearly orthogonal to the gradient PY at (i, j). If the angle between these two singularities is 90 degrees the found singularity is a delta, similarly if the angle is 270 degrees then the found singularity is a core (loop). The angle is calculated counter-clockwise from the PX gradient. As the precise value of the angle is not requires, therefore the calculations can be simplified by determining only whether the angle is over or below 180 degrees by calculating the determinant of the matrix formed by the two gradient vectors as follows:

$$d = \begin{vmatrix} \dfrac{\partial PX_{ij}}{\partial x} & \dfrac{\partial PY_{ij}}{\partial x} \\ \dfrac{\partial PX_{ij}}{\partial y} & \dfrac{\partial PY_{ij}}{\partial y} \end{vmatrix} \tag{3.10}$$

If the determinant is over zero the singular point is classified as a delta, otherwise it is classified as a core.

3.8.1.4 Regularization of Singular Point Extraction

The pseudo-orientation matrices are iteratively smoothed by increasing the size and variance of the gaussian filters at a rate of $[\Delta w_g \quad \Delta w_g]$ and $\Delta\sigma_g$ per iteration until a valid number of singularities are detected by using this method.

Fig. 3-12 shows the complete pseudo code for this algorithm. The performance of the singular point extraction process is improved both in terms of time (as the original image requires more smoothing for proper singular point extraction, which takes a large amount of time) and accuracy by using enhanced fingerprint images as input in place of raw fingerprint images, as shown in Fig. 3-13. The segmentation step is also used to remove any false singularities that are detected in the fingerprint background region. Fig. 3-14 shows the results of the singular point extraction algorithm.

```
n = image size (n x n), assumed that the image is a square
image = the original grayscale fingerprint image as a (n x n) matrix
a = treshold calculation point
w = window size (w x w) for the filterbox
σ = sigma parameter of the Gaussian filterbox
T = threshold value for the PY term
PT = Proximity threshold

for i , j := 2 to n-1 step 2
 dx := image(i,j+1)-image(i,j-1)
 dy := image(i+1,j)-image(i-1,j)
 PY(i/2,j/2) := dy*dx
 PX(i/2,j/2) := |dy|-|dx|
endfor

PY:=filter(PY,w, σ)
PX:=filter(PX,w, σ)

for i , j := 2 to n/2-a step 2
 if |PY(i+a,j+a)| > T
  A := PX(i-1:i+1,j-1:j+1)
  B := PY(i-1:i+1,j-1:j+1)
  if sign(min(A)*max(A)) = -1 AND sign(min(B)*max(B)) = -1
   if "distance to previous found singular point" > PT
    if [A(2,1)-A(2,3)] *[B(1,2)-B(3,2)]...
          ... - [B(2,1)-B(2,3)]*[A(1,2)-A(3,2)] > 0
     "delta found at (2i,2j)"
    else
     "core found at (2i,2j)"
    endif
   endif
  endif
 endif
endfor
```

Fig. 3-12 The pseudo code for the Singular point extraction process

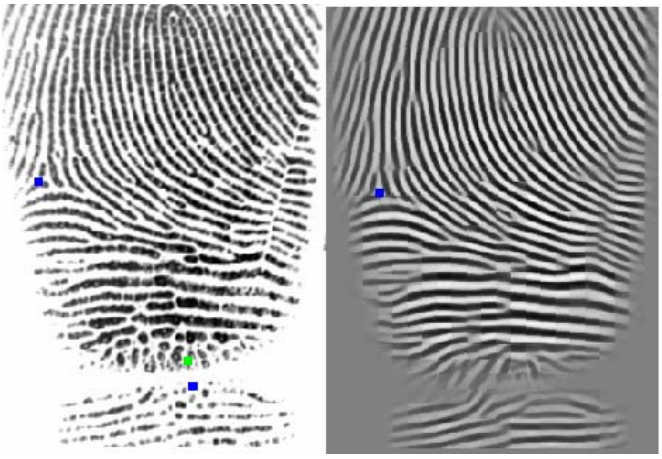

Fig. 3-13 Comparison of the results of the singular point detection algorithm for original (left) and enhanced (right) fingerprint images

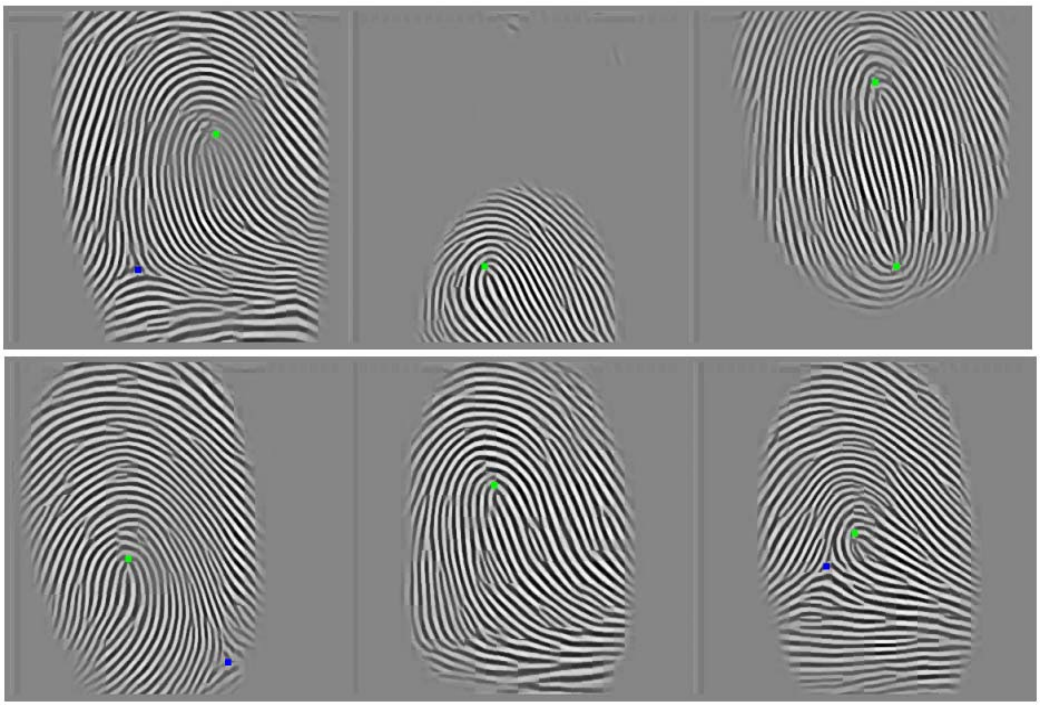

Fig. 3-14 Results of the Singular Point Extraction Algorithm for different Fingerprint Images

3.8.2 Ridge Orientation Estimation

Ridge Orientation Estimation has been carried out by using an approach proposed in [RCJ95]. The input to the ridge orientation estimation process is a fingerprint image and the output is a ridge orientation image, O, which contains the ridge orientations of wxw sized blocks of the fingerprint image as shown in Fig. 3-15.

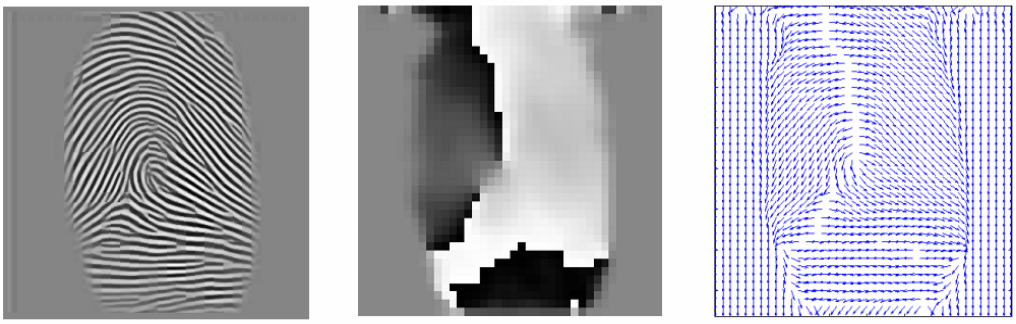

Fig. 3-15 Fingerprint Image (Left), Orientation Matrix (Center) and Orientation Vectors (Right)

3.8.3 Discretization of the Orientation Image

The ridge orientation image is discretized into 8 levels by using the following relation:

$$D(i, j) = \text{mod}(\text{round}(\frac{8}{\pi} O(i, j)), 8) \tag{3.11}$$

This discretization is carried out for use in core analysis. Fig. 3-16 shows the discretized orientation image of a fingerprint.

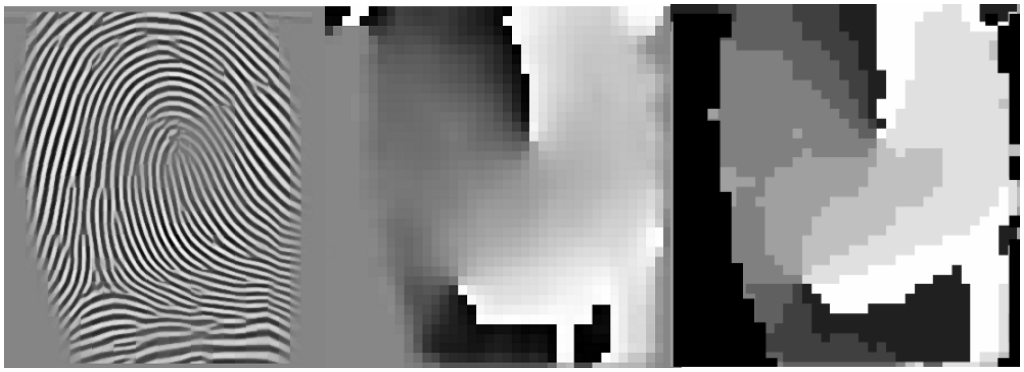

Fig. 3-16 The Discretized Orientation Image (Right) of a Fingerprint (Left)

3.8.4 Core Analysis

Core analysis is an integral part of the classification process. In this step, we analyze the core region of the fingerprint (if one or two cores have been extracted during the singular point extraction process) in order to extract features such as the curvature and orientation of a core, which would aid us in performing rule-based classification of fingerprint images.

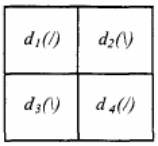

Fig. 3-17 Four regions around the core which are examined by the curvature detection algorithm

The curvature detection logic examines four areas around a core point. From the orientations of the blocks in each area, the curvature is defined as (see Fig. 3-17):

$$Curvature = \begin{cases} Convex & d_1 + d_2 < d_3 + d_4 \\ Concave & \text{else} \end{cases} \tag{3.12}$$

Here,

$$\begin{aligned} d_1 &= \sum_{i=1}^{3}\sum_{j=1}^{3}\left|2 - D(i,j)\right| \\ d_2 &= \sum_{i=1}^{3}\sum_{j=1}^{3}\left|6 - D(i,j)\right| \\ d_3 &= \sum_{i=1}^{3}\sum_{j=1}^{3}\left|2 - D(i,j)\right| \\ d_4 &= \sum_{i=1}^{3}\sum_{j=1}^{3}\left|6 - D(i,j)\right| \end{aligned} \tag{3.13}$$

If the curvature of the core is convex, the orientation is defined seeing the area beneath the core point. If the case is concave the area above the core point is examined. The orientation of a core point is given by:

$$Orientation = \begin{cases} left - skewed & d_l < d_r \\ right - skewed & else \end{cases} \tag{3.14}$$

Here,

$$d_l = \sum_{i=1}^{5}\sum_{j=1}^{5}\left|2 - D(i,j)\right| \tag{3.15}$$

$$d_r = \sum_{i=1}^{5}\sum_{j=1}^{5}\left|6 - D(i,j)\right| \tag{3.16}$$

Fig. 3-18 shows the results of core area curvature and orientation estimation.

Fig. 3-18 Results of the Core Analysis Step

In the case, where a single delta is available along with a single core, we analyze the orientations, $\alpha_1, \alpha_2, ..., \alpha_n$, of the ridges along a line segment joining the core to the delta and calculate a S-value which is given by:

$$S = \frac{1}{n}\sum_{i=1}^{n} \sin(\alpha_i - \beta) \tag{3.17}$$

where, β is the slope of the line-segment. The S-value is a measure of similarity between the ridge orientations, $\alpha_1, \alpha_2, ..., \alpha_n$, and the slope β, and is useful in providing a discrimination between a loop type of fingerprint and a tented arch. In a tented arch, the orientation of the line segment is along the local direction vectors, which leads to a smaller S-value, while in a loop image the line intersects local directions transversally resulting in a higher S-value (see Fig. 3-19). If the S-value is less than a certain threshold T_{SL}, then the fingerprint is classified as a tented arch, otherwise it is taken to be a loop. The same method can be used to distinguish between a whorl and a twin loop, in which case, the line segment joining the two core points is used with a threshold T_{SW}. In a whorl image, the two core points are connected along direction vectors, while in a twin loop image they cannot be connected (see Fig. 3-19).

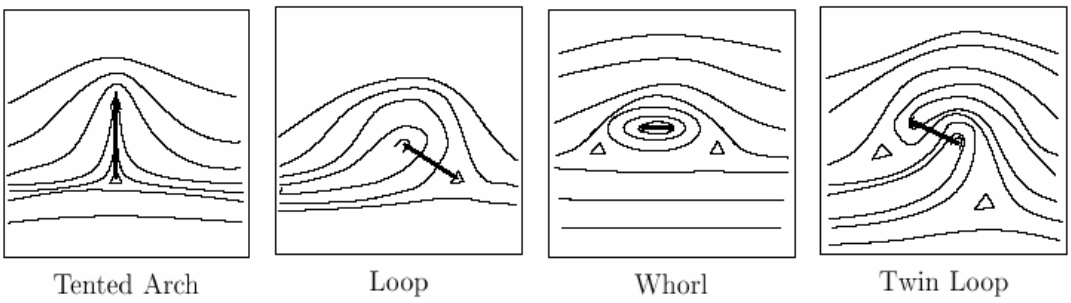

Fig. 3-19 Relationship between the ridge orientations and the line segment between singularities

3.8.5 Application of Decision Logic

The classification rules for different classes are described below:

Tented Arch

A fingerprint is classified as a tented arch if no singularities (core or delta) are found to exist in the input fingerprint image.

Arch

A fingerprint is classified as an arch if a single core exists in the image and either of the following conditions is true:

- A delta is present in the fingerprint image and the S-value is below a certain threshold, T_{SL} .
- In the absence of a delta, the curvature of the core must be lower shape or the left and right side area orientations of the core must be symmetric for the fingerprint to be an arch.

Left Loop

A fingerprint is classified as a left loop if a single core is present and $d_l < d_r$. In the presence of a delta the S-value must also be above a certain threshold, T_{SL} .

Right Loop

A fingerprint is classified as a left loop if a single core is present and $d_l > d_r$. In the presence of a delta the S-value must also be above a certain threshold, T_{SL} .

Whorl

A fingerprint is said to belong to the whorl category, if two cores are present in the fingerprint and the S-value is less than a certain threshold, T_{SW} .

Twin-Loop

A fingerprint is said to belong to the whorl category, if two cores are present in the fingerprint and the S-value is greater than a certain threshold, T_{SW} .

Unknown/Rejected

A fingerprint is rejected (is said to belong to the 'unknown' class) if the extracted core point(s) are very close to the borders of the image. In such cases, it is very difficult to obtain a reliable decision for an accurate fingerprint class. A fingerprint can also be rejected if the foreground area is very small or if certain image quality criterions are not met.

3.9 Results & Discussion

We have used the FVC 2000 and FVC 2002 databases [MMC+00, MMC+02] for the evaluation of the fingerprint classification algorithm. These databases are organized into 4 datasets each. Each dataset comprises of 800 fingerprint images with 8 images

belonging to a single subject. These databases contain fingerprints acquired by using a variety of sensing modalities and are intended for the evaluation of fingerprint matching problems and not for classification. However the use of these databases in assessing the performance of the fingerprint classification algorithm is aimed at analyzing the suitability and applicability of the algorithm for use with live-scan and partial fingerprints.

We tested our algorithm with the FVC 2002 DB1a dataset whose fingerprints were manually classified for comparison. The results obtained show a classification accuracy of 87% with ~5% rejection. Fig. 3-20 show some of the fingerprints correctly classified by the algorithm, which convincingly demonstrates the effectiveness of the algorithm for operation with live-scan and partial fingerprints. Fig. 3-21 shows some fingerprint images rejected by our algorithm. The accuracy can be further improved by increasing the reject rate.

The execution time for the algorithm is ~0.75s. Fig. 3-22 shows the distribution of the time among different constituent steps of the technique.

Fig. 3-20 Fingerprint correctly classified by the classification Algorithm. Note that the algorithm is capable of classifying correctly these images even in the absence of some singular points

Fig. 3-21 Some of the Fingerprint Images Rejected by the classification algorithm

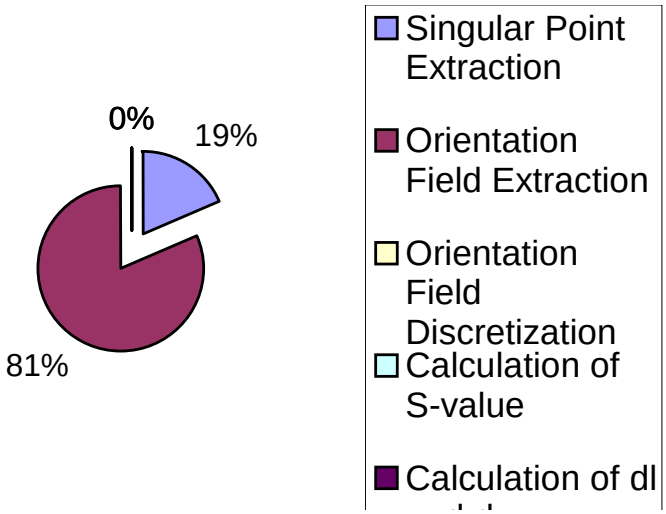

Fig. 3-22 Distribution of the Time spent in different steps of classification

The main causes of error in classification using our approach include:

- Missing Core

  In wirbel type of fingerprints, if a core point is missed in acquisition, then a misclassification error is produced, as the algorithm lacks the ability of classifying wirbel fingerprints using a single core. Examples of this type of error are shown in Fig.3-23. These fingerprints actually belong to Whorl class (as indicated by their neighboring Figures) but the algorithm fails to classify them because one of the core point is missing.

- Rotation of the Fingerprint

  Another factor in the misclassification of fingerprints using our classification technique is the rotation of fingerprints, which may cause left loops to be classified as right loops or vice versa. Some loop types of fingerprints are misclassified as arches because of rotation. Examples of this type of error are shown in Fig. 3-24.

Fig. 3-23 Fingerprints Misclassified because of Missing Loop Singularities

Fig. 3-24 A Left Loop being misclassified as a Right Loop because of rotation

3.10 Other Approaches to Fingerprint Database Indexing

The main problem of the classification schemes discussed in the previous sections is that the number of classes is small and The fingerprints are non-uniformly distributed among them: more than 90% of the fingerprints belong to only three classes (loops and whorls) which leads to poor performance in the retrieval process in terms of response time. Further more in automatic fingerprint classification errors and the rejected fingerprints are required to be handles gracefully. These problems can be addressed with two different approaches:

a. Sub Classification

Sub classification is aimed at the division of the fingerprint classes into further sub classes based on a certain criterion, e.g. FBI [FBI84] uses a manual sub classification criterion based on ridge counting. For right and left loop fingerprints, the number of ridges between the loop and delta singularities is determined and two subclasses are defined on the basis of the ridge count. For whorl fingerprints, the ridge just below the left most delta is traced until the position closest to the rightmost delta is encountered; then the number of ridges between that point and the rightmost delta is counted. Three sub classes are defined depending on the number of ridges and whether the traced ridge pattern passes over the right most deltas. Ridge counting is illustrated in Fig. 3-25.

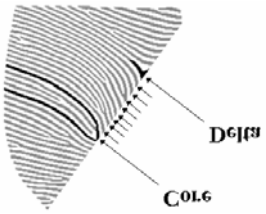

Fig. 3-25 Ridge Counting

Another approach for ridge counting in case of whorl fingerprints is shown in Fig. 3-26. An automated technique for processing a fingerprint in this way is given by [DL98].

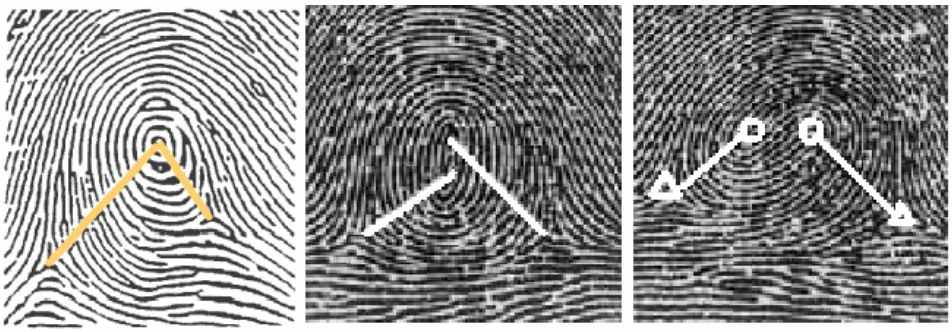

Fig. 3-26 Ridge Counting Methods for the Whorl Class of Fingerprints

b. Continuous Classification

The intrinsic difficulties in automating fingerprint classification and sub classification have led to the introduction of fingerprint classes which are not based on human defined classes. For the applications where there is no need to adhere to the Henry System, any technique able to characterize each fingerprint in a robust and stable manner can be used. Examples include: [LMM97, GCC97, SSL94, QJY+02 etc.]

Summary

Fingerprint classification is an important step in the operation of a fingerprint identification system. It reduces the search space during matching, which speeds up the identification process. Fingerprint classification; because of the low inter class variability among fingerprint class and high intra class variations within a class is a difficult problem to solve. A large number of approaches exist in the literature for fingerprint classification. We have implemented a combinational rule based approach to fingerprint classification, which is optimized for use with live-scan and partial

fingerprints. Apart from fingerprint classification, other schemes such as ridge-count etc. are also used for fingerprint database indexing.

4 Segmentation

4.1 Objectives of Fingerprint Segmentation

Fingerprint image segmentation is an important step in an automatic fingerprint identification and verification system. Segmentation refers to the decomposition of a given fingerprint image obtained through a sensor into the fingerprint foreground and the background. The fingerprint foreground is the region in which the fingertip is in contact with the sensor during acquisition. The noisy region at the borders of the fingerprint image is called the background. Most feature extraction algorithms extract a large number of spurious recognition features (e.g. minutiae) in the background region as shown in Fig. 4-1. Therefore the segmentation of fingerprint images into the fingerprint foreground and the background is an important step prior to minutiae extraction. Fig. 4-1c compares minutiae extraction before and after segmentation, which very clearly exhibits the importance of segmentation in terms of reducing false minutiae.

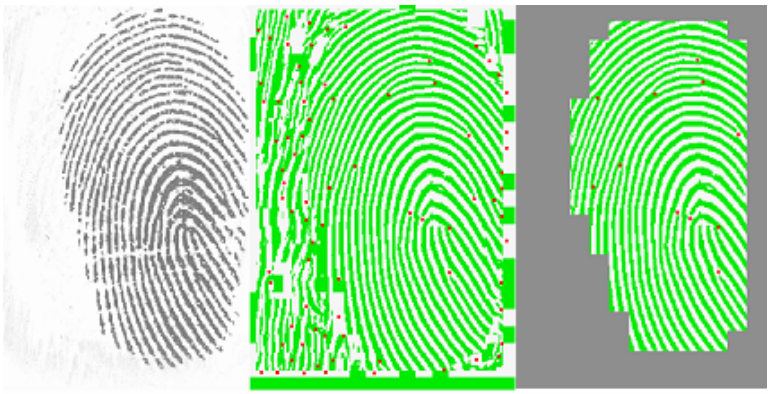

Fig. 4-1 (a) Original Fingerprint Image (b) Minutiae Extracted Before Segmentation (c) Minutiae Extracted After Segmentation

4.2 Literature Survey

The segmentation process is complicated by the striped and oriented nature of the fingerprint image that makes the use of global or local thresholding [GW92] ineffective. The real discrimination between the fingerprint foreground and its background is the

presence of an oriented pattern in the foreground and of an isotropic pattern in the fingerprint background. The detection of such patterns is made intricate by the presence of noise in the fingerprint image (such as dust and grease on the sensor surface). Other factors, such as the movement of the finger after its placement on the sensor surface, make the process of fingerprint segmentation further complex.

Several approaches to fingerprint segmentation exist in the literature. Mehtre et al. [MMK87] proposed a fingerprint image segmentation algorithm based on local histogram of ridge orientations computed for 16x16 blocks. The fingerprint foreground was discerned by the detection of a significant peak in the orientation histogram. However, this method proves to be ineffective in the presence of noise or when presented with a perfectly uniform block. These problems were addressed in [MC89] which uses block variance information along with the orientation histogram. This method gives only moderate segmentation accuracy. Ratha et al. [RCJ95] labeled a fingerprint image block as foreground or background according to the variance of the gray levels in the direction orthogonal to the ridge orientations. This method also provides quality indices for the different blocks of the fingerprint image. It behaves inadequately to low quality and dry fingerprint images in which the variance along the ridge direction can be quite high because of the presence of ridge breaks. Maio et al. [MM95] proposed a gradient-based approach for fingerprint segmentation. This method is based on the assumption that the gradient response in the fingerprint foreground area is higher in comparison to the background because of the presence of ridge valley structures in the foreground region. The performance of this method is degraded in the presence of grease on the sensor surface that keeps an imprint of the previous finger on the surface during the process of acquisition of a fingerprint image. Shen et al. [SKK01] used the variance of the Gabor filter responses obtained by the convolving a bank of 8 Gabor filters with each image block. This method has high noise tolerance but it is computationally intensive. Bazen et al. [BG01] proposed a pixel wise linear classifier based segmentation technique using the gradient coherence, intensity mean and intensity variance. The segmented image is post-processed morphologically to handle fragmented segmentation. The authors report a 6.8% segmentation error for the FVC 2002 fingerprint databases. The pixel wise application of this method results in high computational time, which is undesirable. A Radial Basis Function (RBF) Neural Network based scheme is proposed in [SWY02], which is based on intensity variance, image contrast, gradient coherence and main energy ratio but no

detailed experiment results and performance analysis could be found. In [QJX] fingerprint image segmentation is carried out by the use of feature dots but no information regarding the error rate of the algorithm could be established. In [WH98], Weldon et al. propose a multi-channel system for texture segmentation that can be extended and tuned for fingerprint image segmentation keeping in view the striped textural characteristics of the fingerprint image. A Hidden Markov Model (HMM) based approach for image segmentation and quality evaluation is proposed in [KBV02]. The transition probabilities for the HMM ensure an estimation consistent with the neighborhood which is effective in handling fragmented segmentation. The authors report a minimum error rate of 6.5%. In [CTR+04] a linear classifier is used for segmentation of fingerprint images. This method is based on the degree of clustering, the intensity mean and the intensity variance of the blocks in the image. Reported error rate of this algorithm is 2.45%. Bazen et al., [BG00] have carried out gradient coherence based fingerprint segmentation with a morphological post-processing stage but no error-rate statistics have been reported. Other segmentation methods include [Zho04], [Mos01], [JRL97], [JF91], [LS85] and [SU02].

4.3 A Novel Approach to Fingerprint Segmentation

In this Section an effective algorithm for fingerprint image segmentation is presented. This technique is based on the fusion of multiple features that are projected onto a one dimensional feature space using Fisher discriminant analysis. The classification of the fingerprint regions as foreground or background is carried out by the use of Learning Vector Quantization (LVQ) Neural Networks. The detail of the segmentation method is given henceforth.

4.3.1 Features Used for Fingerprint Segmentation

The segmentation and quality evaluation techniques discussed earlier in this chapter perform segmentation using features drawn only from the original fingerprint image. Sometimes it is possible that the enhancement algorithm can correct some errors present in the fingerprint image. Most of the existing segmentation algorithms do not take into account the capabilities of the particular fingerprint enhancement algorithm while carrying out segmentation or quality evaluation. In our approach, features extracted both from the original and the enhanced images are used for fingerprint image segmentation and quality

evaluation. In this way, we are able to detect irrecoverable regions in a much more effective manner incorporating the error correction capabilities of the fingerprint image enhancement algorithm.

For feature extraction, original and the enhanced images (for details about the enhancement process see chapter 5) of a fingerprint to be processed are first divided into $w \times w$-sized blocks. For each block the following features are extracted in the original and the enhanced images:

4.3.1.1 Intensity mean

The mean of a $w \times w$ sized block 'b' is given by

$$\mu_b^I = \frac{1}{w \bullet w} \sum_{I \in b} I \tag{4-1}$$

Where 'I' refers to the intensity values in the block. The mean is the measure of the average gray level in the block. Usually, we normalize the block mean by subtracting the global image mean from the means of the individual blocks.

$$\mu^b = \mu_b^I - \frac{1}{n \cdot n} \sum I \tag{4-2}$$

The intensity means of both the original and the enhanced image are used in the formation of the feature set.

The block-intensity-mean of the original image

Let $\mu_o^{b_1}, \mu_o^{b_2}, ..., \mu_o^{b_n}$ be the intensity means of the blocks $b_1, b_2, ..., b_n$ of the original image. Where n is the total number of blocks in the image each of size $w \times w$. Then the block intensity mean for the original image is expressed as the vector:

$$\mu_o = \left[\mu_o^{b_1} \quad \mu_o^{b_1} \quad \dots \quad \mu_o^{b_1} \right] \tag{4-3}$$

This feature can be used for segmentation because the foreground mean is generally lower than the background mean as shown in Fig. 4-2a.

Intensity mean is greatly affected by noise in the fingerprint image and by the change in the pressure applied by the person on the finger during acquisition. Fig. 4-2b shows the distributions of the mean for the background and the foreground of the fingerprint image blocks drawn at random from FVC 2000 DB2 [MMC+00] database.

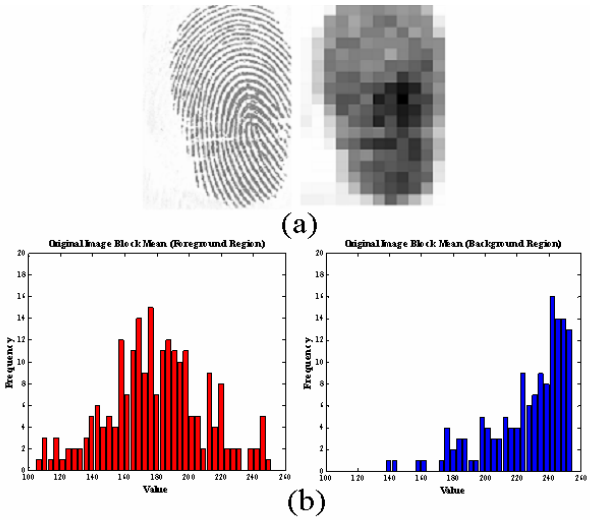

Fig. 4-2 (a) Block Mean of an Original Fingerprint Image, (b) Histogram of Enhanced Image Block Means for foreground and background image blocks

The block-intensity-mean of the enhanced image

The block-intensity-mean of the enhanced image is expressed as a vector of the colomized block image intensity means i.e.

$$\mu_e = \left[\mu_e^{b_1} \quad \mu_e^{b_2} \quad \dots \quad \mu_e^{b_n} \right] \tag{4-4}$$

where $\mu_e^{b_i}$ is the mean of the ith block of the enhanced image. The block mean of a image after enhancement is shown in Fig. 4-3a.

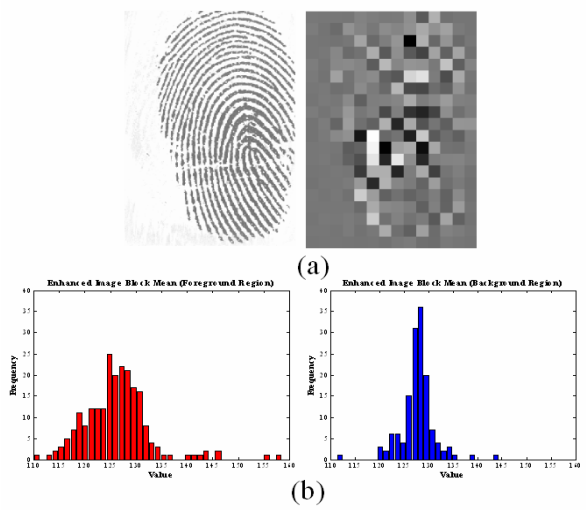

Fig. 4-3 (a) Block Mean of an Enhanced Fingerprint Image (b) Histogram of Enhanced Image Block Means for foreground and background image blocks

The histogram in Fig. 4-3b shows the distribution of this intensity mean for the fingerprint background and the foreground.

4.3.1.2 Intensity variance

The intensity variance of a $w \times w$-sized block 'b' is given by

$$v_b = \sigma_b^2 = \frac{1}{w \bullet w} \sum_{I \in b} (I - \mu_b)^2 \tag{4-5}$$

The block intensity variance is a measure of the amount of variation in the intensity values within a block. The intensity variances of both the original and the enhanced image are used in the formation of the feature set.

The block variance of the original image

The block-intensity-variance of the original image is expressed as a vector of the colomized block image intensity variances

$$v_o = \left[v_o^{b_1} \quad v_o^{b_2} \quad \ldots \quad v_o^{b_n} \right] \tag{4-6}$$

where $v_o^{b_i}$ is the variance of the i^{th} block of the original image. Fig. 4-4a shows the block variance for a particular image.

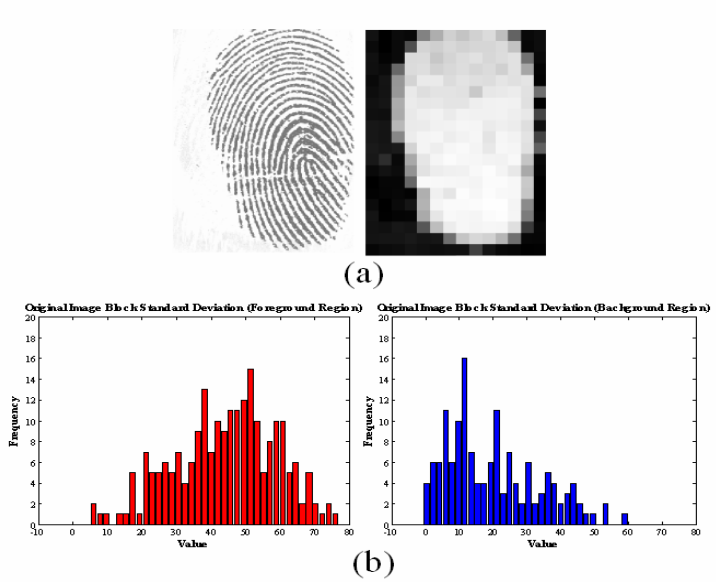

Fig. 4-4 (a) Block Standard Deviation of an Original Fingerprint Image (b) Histogram of Original Image Block Standard Deviations for foreground and background image blocks

The variance in the foreground of the fingerprint is higher than that in the background because of the presence of an alternating ridge valley structure in the foreground. This fact is shown in Fig. 4-4b.

The block variance of the enhanced image

The block-intensity-variance of the enhanced image is expressed as a vector of the colomized block image intensity variances

$$v_e = \left[v_e^{b_1} \quad v_e^{b_2} \quad \dots \quad v_e^{b_n} \right] \tag{4-7}$$

where $v_e^{b_i}$ is the variance of the i th block of the enhanced image. Fig. 4-5(a) shows the block variance for a particular enhanced image.

The variance of the background is lower than the variance of the foreground region. This phenomenon is illustrated in Fig. 4-5(b).

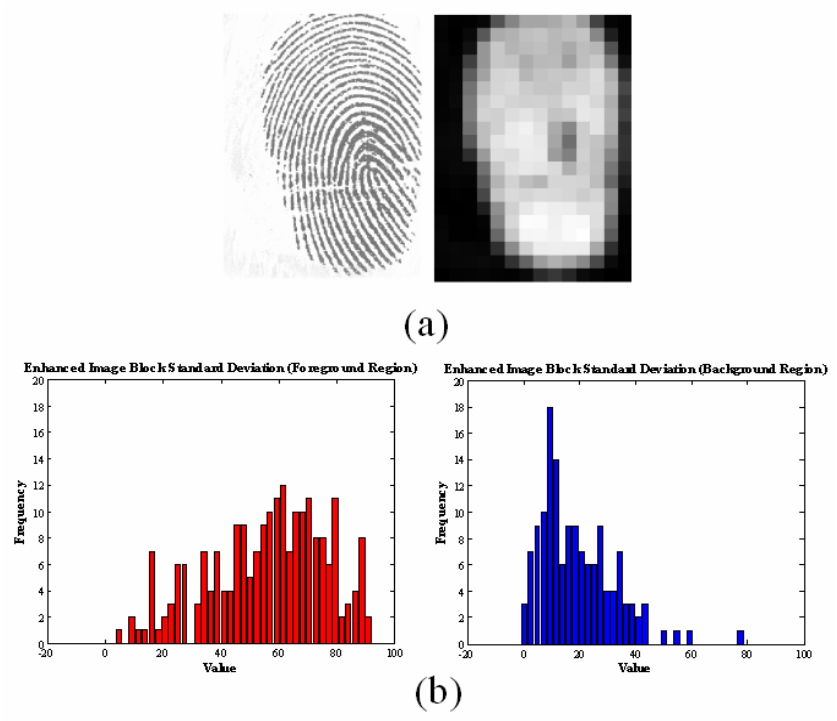

Fig. 4-5 (a) Block Standard Deviation of an Enhanced Fingerprint Image (b) Histogram of Enhanced Image Block Standard Deviations for foreground and background image blocks

4.3.1.3 Gradient Coherence

The coherence gives a measure how well the gradients are pointing in the same direction. Mathematically,

$$C_b = \frac{\left|\sum_W (G_{s,x}, G_{s,y})\right|}{\sum_W \left|(G_{s,x}, G_{s,y})\right|} = \frac{\sqrt{(G_{xx} - G_{yy})^2 + 4G_{xy}^2}}{G_{xx} + G_{yy}} \tag{4-8}$$

Where $(G_{s,x}, G_{s,y})$ is the squared gradient, $G_{xx} = \sum_W G_x^2$, $G_{yy} = \sum_W G_y^2$, $G_{xy} = \sum_W G_x G_y$ and (G_x, G_y) are the local gradients along X and Y-axes evaluated using the sobel masks. The gradient coherences of both the original and the enhanced image are used in the formation of the feature set.

The block coherence of the original image

The coherence of the original image is expressed as a vector of the colomized block image coherences:

$$C_o = \left[C_o^{b_1} \quad C_o^{b_2} \quad \ldots \quad C_o^{b_n} \right] \tag{4-9}$$

where $C_o^{b_i}$ is the coherence of the i th block of the original image.

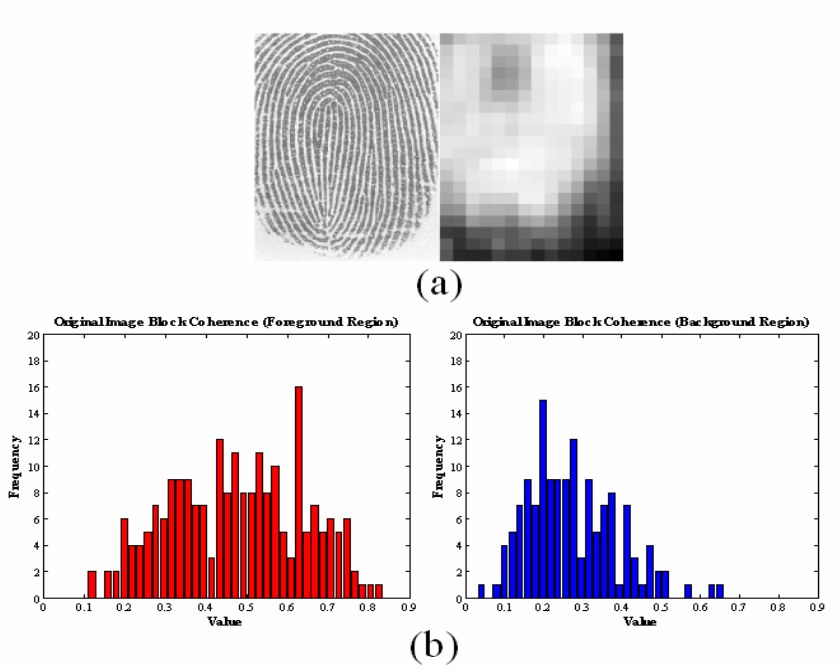

Fig. 4-6 (a) Block Coherence of an Original Fingerprint Image (b) Histogram of Original Image Block Coherence for foreground and background image blocks

Fig. 4-6a shows the coherence for a particular image. The background coherence is lower than the foreground coherence because of the presence of isotropic texture in the background of the image and the presence of smoothly changing ridge direction for the fingerprint foreground (see Fig. 4-6b). However the block coherence is low also at the singular points (core and delta) and special measures must be

adopted in order to improve this issue if only coherence is to be used for fingerprint segmentation.

The block coherence of the enhanced image

The coherence of the enhanced image is expressed as a vector of the colomized block image coherences:

$$C_e = \left[C_e^{b_1} \quad C_e^{b_2} \quad \ldots \quad C_e^{b_n} \right] \tag{4-10}$$

where $C_e^{b_i}$ is the coherence of the i^{th} block of the original image. Fig. 4-10a shows the coherence for a particular enhanced image. The background coherence is lower than the foreground coherence. (See Fig. 4-7b)

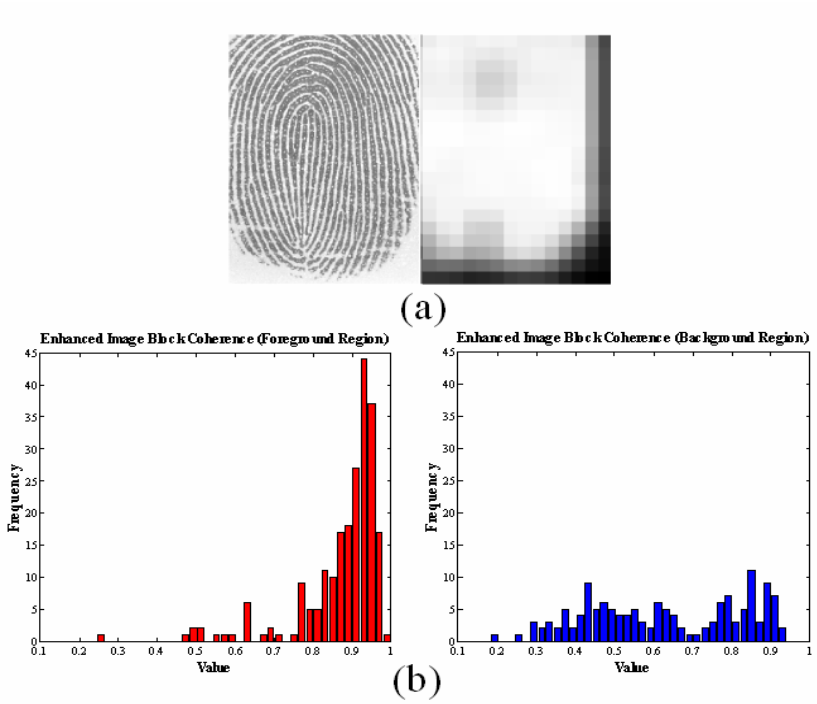

Fig. 4-7 (a) Block Coherence of an Enhanced Fingerprint Image (b) Histogram of Enhanced Image Block Coherence for foreground and background image blocks

4.3.1.4 Energy Map

Some enhancement methods use the ridge frequency and ridge orientation information extracted from a fingerprint image to create contextual filters, which are convolved with the fingerprint image. Examples of such algorithms include [HWJ98], [JHP+97], [YLJ+03] and [CWG04]. These algorithms utilize the orientation and ridge frequency information in the striped nature of the fingerprint image. The response of the

contextual filters used in these algorithms exhibit a high energy in the presence of a striped and oriented texture, which lies primarily in the fingerprint foreground. Therefore the energy of the filter responses of the features can be used for fingerprint segmentation. We have used the approach given in [CWG04] which uses a Fourier domain based block-wise contextual filter approach for enhancing fingerprint images. Here, the energy map is defined as:

$$E_b = \sum_{u,v \in b} |F_b(u,v)|^2 \tag{4-11}$$

Where $F_b(u,v)$ is the Fourier spectrum of the fingerprint image block 'b'. Fig. 4-8a shows the energy map of a fingerprint image.

The energy of the filter response is expressed as a vector of the colomized block image energies:

$$E = \left[E^{b_1} \quad E^{b_2} \quad \ldots \quad E^{b_n} \right] \tag{4-12}$$

where E^{b_i} is the energy of the ith block of the original image. The background energy is lower than the foreground energy (see Fig. 4-8b). However the energy response of the fingerprint can be distorted because of dust or grease on the sensor surface or in the presence of the imprint left over by a previous finger.

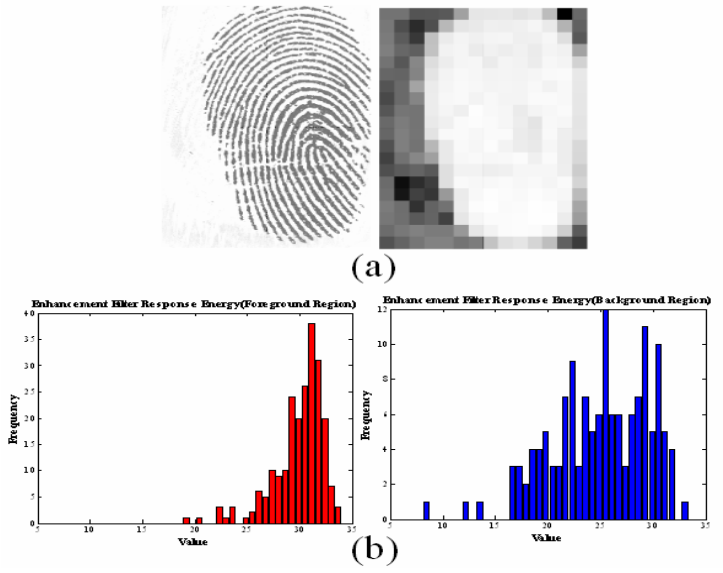

Fig. 4-8 (a) Energy map for a fingerprint image, (b) The histograms for filter response energy of the foreground and the background of fingerprint images

4.3.2 Discrimination Power of Different Features

We quantize the fitness of a particular feature for performing segmentation by analyzing the overlap between the distributions of the values of the feature for foreground and background blocks. This overlap is represented by a discrimination index, which is defined henceforth.

Let α_{b_f} be the histogram of the values of a particular feature 'f' evaluated for the background blocks. Let α_{f_f} be the histogram of the values of a particular feature 'f' evaluated for the foreground blocks. We define a normalization factor, η, as:

$$\eta_f = \min(\sum_{b=1}^{N_b} \alpha_{b_f}^k, \sum_{b=1}^{N_b} \alpha_{f_f}^k) \tag{4-13}$$

Here, N_b is the number of bins in both the histograms. The overlap, ς, is given by:

$$\varsigma_f = \sum_{k=1,(\alpha_{b_f}^k \cdot \alpha_{b_f}^k)>0}^{N_b} \min(\alpha_{b_f}^k, \alpha_{f_f}^k) \tag{4-14}$$

The discrimination index is defined as:

$$\beta_f = 1 - \frac{\varsigma_f}{\eta_f} \qquad \beta_f \in [0,1] \tag{4-15}$$

Table 4-1 Discrimination Indices for Different Features

Segmentation Feature	Discrimination Index of Background and Foreground Distributions
Original Image Block Mean	0.5634
Enhanced Image Block Mean	0.2028
Original Image Block Variance	0.5245
Enhanced Image Block Variance	0.6154
Original Image Block Coherence	0.4266
Enhanced Image Block Coherence	0.5245
Filter Response Energy	0.4755

The discrimination index is one if the two there is no overlap between the distributions of the two classes. A value of zero for the discrimination index indicates a complete overlap of the two class distributions. Table 4-1 presents the overlap indices of different features discussed earlier.

4.3.3 Fingerprint Segmentation

For fingerprint segmentation a given fingerprint image is divides into n blocks each of size $w \times w$. For fingerprint segmentation, the feature vector of ith block (i=1...n) of an image is described by the vector

$$\Phi_i = \left[(v_o^{b_i}) \quad (v_e^{b_i}) \quad (\mu_o^{b_i}) \quad (\mu_e^{b_i}) \quad (C_o^{b_i}) \quad (C_e^{b_i}) \quad (E^{b_i}) \right]^T \qquad (4\text{-}16)$$

Fingerprint image segmentation is carried out by the use of LVQ Neural Networks [Fau94]. LVQ Networks have been employed as classifiers in a variety of problems ranging in diversity from speech coding to writer-identification. Fig. 4-9 shows the architecture of a typical LVQ network.

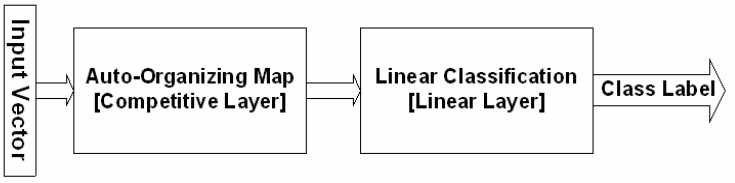

Fig. 4-9 Architecture of a LVQ Neural Network

LVQ networks consist of two layers: First is the competitive layer which finds the subclasses of the given input vectors. The second layer of the LVQ is a linear layer that transforms the competitive layer subclasses into target classifications provided by the user. We have used LVQ 2.0 as the training algorithm. LVQ 2.0 updates the two best matching units if one of the two belongs to the desired class and the distance ratio is within a defined window. LVQ Networks are best suited for the segmentation problem at hand because of the network's capability to minimize the classification error.

There are two phases of the segmentation algorithm viz. Training and Operation.

During the training phase, N (~300) blocks ($\Phi_1 \, \Phi_2 \, ... \, \Phi_N$) of both background and the foreground are collected (at random) from the images in the

database and their features are extracted. These blocks are represented as a vector X_{Train} given by:

$$X_{Train} = \begin{bmatrix} \Phi_1 & \Phi_2 & \cdots & \Phi_N \end{bmatrix} \tag{4-17}$$

The extracted features are then normalized to the range [0 1]. The mean of the training features is calculated as follows:

$$m = \frac{1}{N}\left[\sum_{i=1}^{N} v_o^{b_i} \quad \sum_{i=1}^{N} v_e^{b_i} \quad \sum_{i=1}^{N} \mu_o^{b_i} \quad \sum_{i=1}^{N} \mu_e^{b_i} \quad \sum_{i=1}^{N} C_o^{b_i} \quad \sum_{i=1}^{N} C_e^{b_i} \quad \sum_{i=1}^{N} E^{b_i}\right]^T \tag{4-18}$$

The blocks are classified into either of the two classes (foreground or background) manually to form the target class vector T as follows:

$$T_i = \begin{cases} 1 & \text{if } ith \text{ block belongs to background} \\ 2 & \text{if } ith \text{ block belongs to foreground} \end{cases} \tag{4-19}$$

Let C_f denote the foreground class and C_b denote the background class. Let N_B be the number of training blocks belonging to C_b and N_f be the number of training blocks belonging to C_f. X_{Train} is divided into X_1 and X_2 which contain the blocks belonging to the background and the foreground respectively. The mean for each of the $N_C = 2$ classes are:

$$m_1 = \frac{1}{N_B}\left[\sum_{b_i \in C_b} v_o^{b_i} \quad \sum_{b_i \in C_b} v_e^{b_i} \quad \sum_{b_i \in C_b} \mu_o^{b_i} \quad \sum_{b_i \in C_b} \mu_e^{b_i} \quad \sum_{b_i \in C_b} C_o^{b_i} \quad \sum_{b_i \in C_b} C_e^{b_i} \quad \sum_{b_i \in C_b} E^{b_i}\right]^T \tag{4-20}$$

$$m_2 = \frac{1}{N_F}\left[\sum_{b_i \in C_f} v_{o\,b_i} \quad \sum_{b_i \in C_f} v_{e\,b_i} \quad \sum_{b_i \in C_f} \mu_{o\,b_i} \quad \sum_{b_i \in C_f} \mu_{e\,b_i} \quad \sum_{b_i \in C_f} C_{o\,b_i} \quad \sum_{b_i \in C_f} C_{e\,b_i} \quad \sum_{b_i \in C_f} E_{b_i}\right]^T \tag{4-21}$$

Fisher discriminant analysis ([JDM00], [Yam00], [FH04) is used as a preprocessing stage to increase the discrimination between the segmentation classes. Fisher discriminant analysis, proposed by R.A. Fisher in the 1930s, groups features belonging to a single class and separates those belonging to different classes. In this form of discrimination analysis the features are projected from a N-dimensional space to a $N_c - 1$ dimensional space where N_c is the number of classes. Fisher Discriminant Analysis is based on the solution of the generalized Eigen value problem

$$S_B V = \Lambda S_W V \tag{4-22}$$

Here S_B is the between-class scatter matrix, which measures the amount of scatter between classes. It is calculated as the sum of the covariance matrices of the difference between the total mean and the mean of each class. Mathematically,

$$S_B = \sum_{i=1}^{N_C} n_i (m_i - m)(m_i - m)^T \tag{4-23}$$

where n_i is the number of feature sets (data points) in the class, m_i is the mean of the individual features in the class and m is the mean of individual features of all the data points. S_W is the within-class scatter matrix, which measures the amount of scatter within each class. Mathematically,

$$S_W = \sum_{i=1}^{N_C} S_i \tag{4-24}$$

Here S_i is the scatter matrix for the ith class given by,

$$S_i = \sum_{x \in X_i} (x - m_i)(x - m_i)^T \tag{4-25}$$

where x is a feature set (data point) and X_i is the denotes all feature sets in the class. V are the Eigen vectors corresponding to the eigen values Λ. The Eigen values are sorted in decreasing order and the first $N_c - 1$ Eigen vectors corresponding to the highest $N_c - 1$ Eigen values are selected. This Eigen vectors form the Fisher basis vector, V_b.

All the data points are then projected to the new $N_c - 1$ dimensional feature space by using the relation:

$$\tilde{x} = V_b^T x \tag{4-26}$$

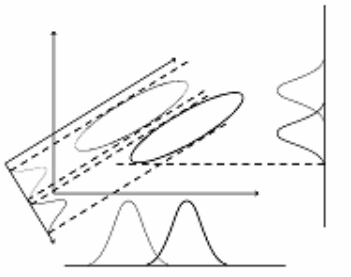

Fig. 4-10 Maximum Discrimination is provided by the projection on the Fisher Basis Vector

The basis vectors provide the best discrimination of the data for the given classes in the $N_c - 1$ dimensional feature space. Fig. 4-10 demonstrates the effectiveness of Fisher discriminant analysis in increasing the contrast between two classes.

The between-class and within-class scatter matrices are calculated for the segmentation features by using equation (4-23) & (4-24) respectively. The fisher basis vector,

$$V_B = [\omega_{v_o} \quad \omega_{v_e} \quad \omega_{\mu_o} \quad \omega_{\mu_e} \quad \omega_{C_o} \quad \omega_{C_e} \quad \omega_E]^T \tag{4-27}$$

where, ω_f is the weight of the feature 'f', is calculated. The data point, X, are projected onto the new subspace by using the equation (4-26) to give the projected vectors $\tilde{X}$.

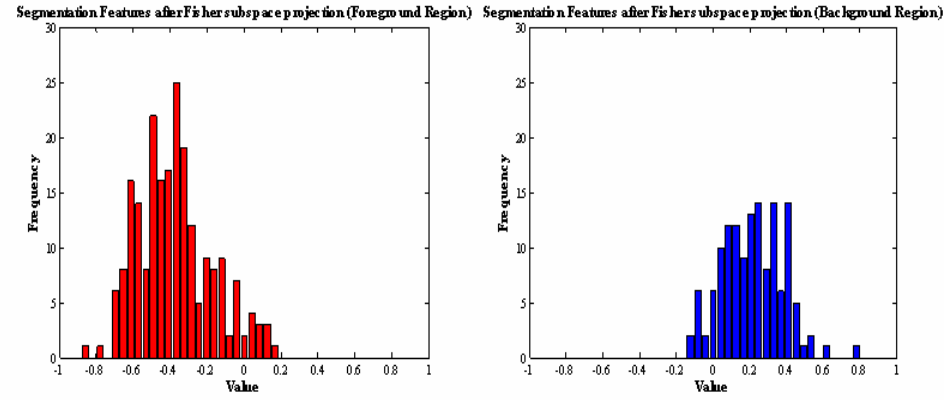

Fig. 4-11 The histograms of the segmentation features after projection on the Fisher subspace

Fig. 4-11 shows that the discrimination between the fingerprint foreground and the background provided by this new feature data is significantly improved in comparison to the original feature data. The value of the discrimination index for the new feature set is 0.8623, which proves its suitability for application to the segmentation problem. Fig. 4-12 shows a scatter plot for the background and foreground values of the data obtained after the application of the fisher discriminant analysis, which also shows a very small overlap.

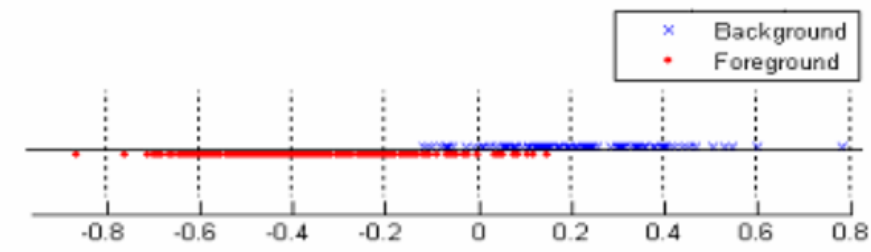

Fig. 4-12 Scatter plot of fisher subspace projection of segmentation features. The background and foreground points have been separated for improving clarity

This one-dimensional data is then presented to the LVQ NN consisting of 8 hidden neurons and able to discriminate between two classes (for fingerprint foreground and background) for training. Fig. 4-13 shows the block diagram for the training phase.

The classifier after training is used in the operation phase to segment the given fingerprint images. In this phase the input image is first divided into n-blocks and for each block the features mentioned earlier are extracted. These features are normalized to 0-1 range. Thus the image is described by,

$$X = \left[\Phi_1 \quad \Phi_2 \quad \cdots \quad \Phi_n\right] \tag{4-28}$$

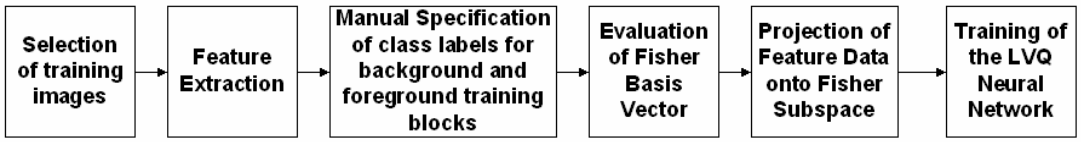

Fig. 4-13 Different steps in the training phase for fingerprint image segmentation

These features are then projected to a single dimension by using the basis vector obtained during the training phase, V_B. After the projection, the one dimension projected data, $\tilde{X}$ is presented to the trained LVQ NN for classification. The output of the neural network for each of the projected feature is used to form a segmentation vector, S' which, after conversion to a 2D matrix of the same size as the original image through resizing results in the segmentation mask, S. The segmentation mask is multiplied with the enhanced image in a pixel wise manner to yield the segmented image. Unlike other segmentation algorithms discussed earlier, no post-processing of any kind is required because fragmented segmentation is not an issue. If the total area of the foreground region, A_f, is smaller than a certain threshold, T_{A_f}, the image is rejected. Fig. 4-14 shows the block diagram for the segmentation process.

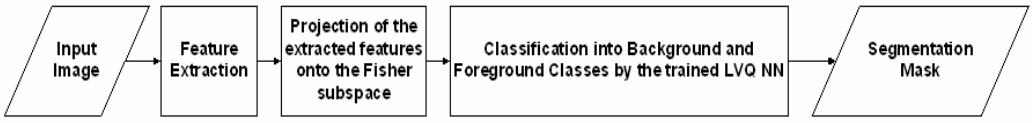

Fig. 4-14 Fingerprint segmentation by using FDA and LVQ

4.4 Results

Fig. 4-15 shows the results of fingerprint segmentation for very poor quality images, which clearly shows the effectiveness of the technique for low quality fingerprints. (All images are from FVC 2000 databases). The segmentation algorithm had been trained on only 300 blocks (120 background blocks and 180 foreground blocks) taken from 35 different (randomly chosen) images from the database.

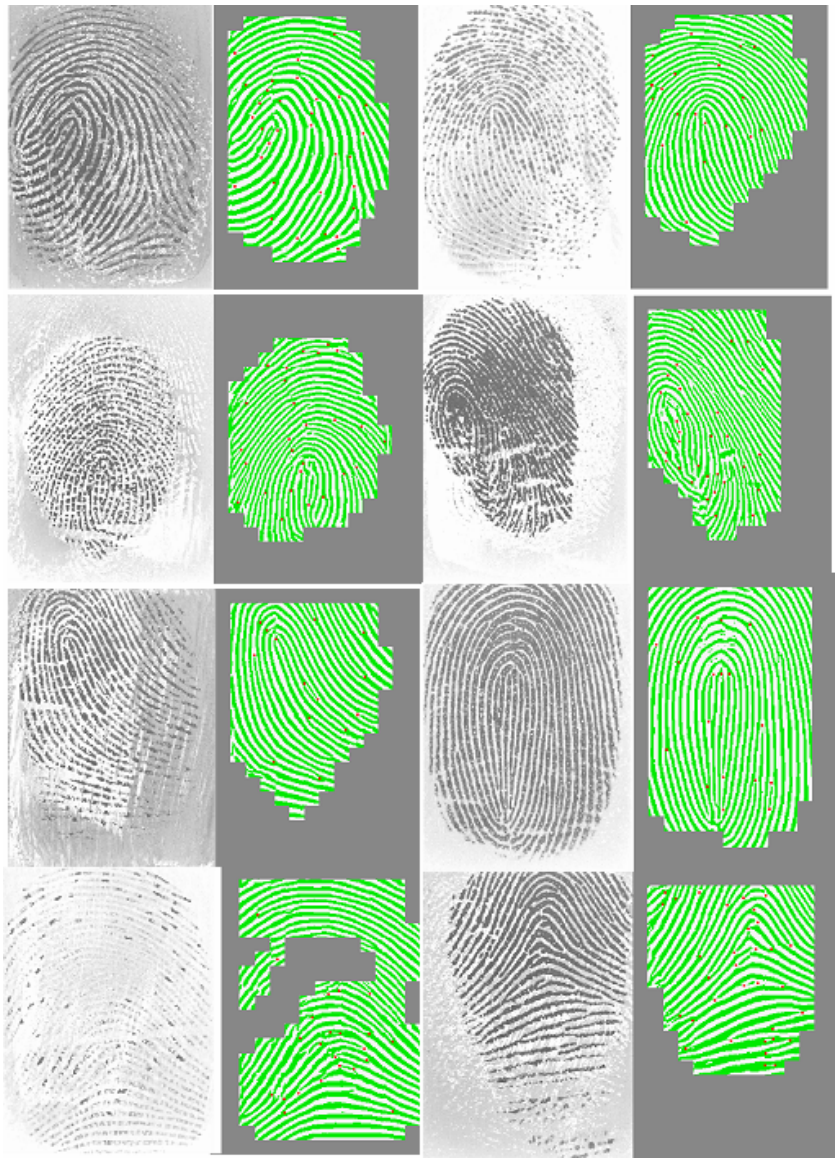

Fig. 4-15 Results of the Segmentation Algorithm

The algorithm can be trained for images from multiple databases, if simultaneous operation on a variety of databases is desired. The block size is taken to be 16 x 16. If high

resolution is desired then blocks of smaller sizes can be used. A block size of 16x16 is found to be effective for most practical purposes. Performance can be further improved by smoothing the input image by using a Gaussian kernel of a small size and low standard deviation.

The segmentation algorithm was tested for the FVC 2000 databases and the percentage of misclassified blocks turns out to be only 1.8%, which also is a testimony to the effectuality of the algorithm.

4.5 Other Approaches to Fingerprint Segmentation

Results of segmentation using a local gradient coherence based approach, which relies on the calculation of a segmentation mask by the binarization of a coherence images, are shown in Fig. 4-16. Here the coherence image obtained for a fingerprint is smoothed using a gaussian filter of size wxw and standard deviation σ. The binarization is carried out by using the equation given below:

$$C_B(i,j) = \begin{cases} 1, & C(i,j) > M_c - \lambda S_c \\ 0, & Otherwise \end{cases} \tag{4-29}$$

Fig. 4-16 Segmentation on the basis of Coherence

Where C_B is the binarized coherence image. M_c And S_c are the global mean and standard deviations of C respectively. C_B when multiplied pixel-wise with the original image results in the segmented image S. Here λ is optimized analytically for a given fingerprint database. This algorithm provides satisfactory performance in terms of segmentation accuracy. The main advantage of using this approach is it low computational complexity. This algorithm requires only 0.2s for a 364 x 256 sized fingerprint image on a P-IV with 512MB of RAM under Windows 2000 operating system using Matlab 7 [Mat05]. Moreover this method can be applied prior to enhancement, which makes it a universal approach to fingerprint segmentation independent of any fingerprint enhancement.

Results can be improved, if normalization is performed prior to segmentation. The binarized coherence image may contain 'noise' where spurious small areas of one class show inside another class. Morphological Operations such as that of curve fitting and active contour modeling [BG01] can be used to remove such noise. A simpler morphological post processing approach is to use dilation or region filling. Other improvements include the combinational use of block mean and variance in this algorithm.

Summary

Fingerprint Segmentation is a critical step in a fingerprint identification system. It is aimed at the separation of the fingerprint image foreground and background. We have developed a novel approach to fingerprint image segmentation that uses features such as block image mean, variance, coherence and enhancement filter energy of both the original and the enhanced images of a fingerprint. This gives the segmentation algorithm the ability to incorporate the recovery effect of the enhancement algorithm. A LVQ Neural Network is used as a classifier for the generation of the segmentation mask which when multiplied with the original fingerprint image results in the segmented fingerprint.

5 Fingerprint Image Enhancement

5.1 Objectives of Fingerprint 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. In such situations, the ridges can be easily detected and the minutiae and other recognition features can be precisely located in the image. However in practice, due to skin conditions (e.g. wet or dry, cuts and bruises), sensor noise, incorrect finger pressure, and inherently low quality fingers (e.g. elderly people, manual workers), a significant percentage of fingerprint images (app. 10%) are of poor quality where the ridge pattern is very noisy and corrupted. In general there are several types of degradation associated with the fingerprint images:

a. The ridges are not strictly continuous; i.e. the ridges have small breaks (gaps) as shown in Fig. 5-1.

Fig. 5-1 Ridge Breaks resulting in False Minutiae

b. Parallel ridges are not well separated. This is due to the presence of noise which links parallel ridges, resulting in their poor separation as shown in Fig. 5-2.

Fig. 5-2 Poor Separation among Parallel Ridges

c. Cuts, creases and bruises as shown in Fig. 5-3.

Fig. 5-3 A Cut in a Fingerprint Image

These three types of degradations make ridge extraction extremely difficult in the highly corrupted regions, which result in poor performance of the ridge and feature extraction and matching algorithms. In order to ensure good performance of these algorithms in poor quality fingerprint images, an enhancement algorithm to improve the clarity of the ridge structure is necessary.

A fingerprint expert is often able to correctly identify the minutiae by using various visual clues such as local ridge orientation, ridge continuity and so on. In theory, it is possible to develop an enhancement algorithm that exploits these visual clues to improve the quality if a fingerprint image.

Generally, for a given fingerprint image, the fingerprint areas resulting from the segmentation step may be divided into three categories (Fig. 5-4):

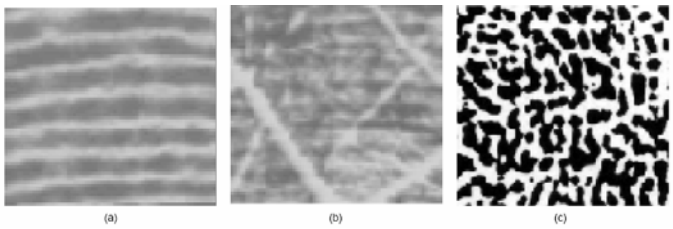

Fig. 5-4 Fingerprint Regions: (a) Well Defined Region, (b) Recoverable Region, (c) Unrecoverable Region

a. Well defined regions: where ridges are clearly differentiated from each other

b. Recoverable regions: where the ridges are corrupted by a small amount of gaps, creases, smudges, links and the like, but they are still visible and the neighboring regions provide sufficient information about their true structure.

c. Unrecoverable regions: where ridges are corrupted by such a severe amount of noise and distortion that no ridges are visible and the neighboring regions do not allow them to be reconstructed.

Good quality regions, recoverable regions and unrecoverable regions may be identified according to several criterions; In general, image contrast, orientation consistency, ridge frequency and other local features may be combined to define a quality index. The goal of an enhancement algorithm is to prove the clarity of the ridge structures in the recoverable regions and mark the unrecoverable regions as too noisy for further processing.

Usually, the input of the enhancement algorithm is a gray-scale image; the output may either be a gray-scale or a binary Image, depending upon the algorithm. General-purpose image enhancement techniques do not produce satisfactory results for fingerprint image enhancement. However, contrast stretching, histogram manipulation, normalization [HWJ98] and wiener filtering [GAK+00] have been shown to be effective as initial processing steps in a more sophisticated enhancement algorithm.

5.2 Literature Survey

The most widely used technique for fingerprint image enhancement is based on contextual filters. In conventional image filtering, only a single filter is used for convolution throughout the image. In contextual filtering, the filter characteristics change according to the local context. Usually a set of filter is pre-computed and one of them is selected for each image region. In fingerprint enhancement, the context is often defined by the local ridge orientation and local ridge frequency. In fact, the sinusoid-shaped wave of the ridge and valleys is mainly defined by a local orientation and frequency that varies slowly across the fingerprint area. An appropriate filter that is tuned to the local ridge frequency and orientation can efficiently remove the undesired noise and preserve the true ridge and valley structure.

Several types of contextual filters have been proposed in the literature for fingerprint enhancement. Although the have a different definition, the intended behavior is almost the same:

Provide a low pass (averaging) effect along the ridge direction with the aim of linking small gaps and filling impurities due to pores or noise

Performa band pass (differentiating) effect in the direction orthogonal to the ridges to increase the discrimination between the ridges and valleys and to separate the parallel linked ridges

The method proposed by O' Gorman and Nickerson [ON88, ON99] was one of the first to use contextual filtering for fingerprint enhancement; the authors defined a mother filter based on four parameters of a fingerprint image at a given resolution: minimum and maximum ridge width, and minimum and maximum valley width. The authors assume the ridge frequency to be constant through out the fingerprint input image and used convolution in the spatial domain for the purpose of filter application.

Sherlock et al., [SMM92, SMM94] performed contextual filtering in the Fourier domain.

Hong et al. [HWJ98] proposed an effective method based on Gabor filters. Gabor filters have both frequency selective and orientation-selective properties and have optimal joint resolution in both spatial and frequency domains. A Gabor filter is defined by a sinusoidal plane wave tapered by a Gaussian. The even symmetric 2D Gabor filter has the following form:

$$g(x, y : \theta, f) = \exp\left\{-\frac{1}{2}\left[\frac{x_\theta^2}{\sigma_x^2} + \frac{y_\theta^2}{\sigma_y^2}\right]\right\} \cdot \cos(2\pi f \cdot x_\theta) \qquad (5\text{-}1)$$

Where θ is the orientation of the filter and $[x_\theta, y_\theta]$ are the coordinates of $[x\ , y\ ]$ after a clockwise rotation of the Cartesian axes by $\frac{\pi}{2} - \theta$.

$$\left[x_\theta\ y_\theta\right]^T = \begin{pmatrix} \cos(\frac{\pi}{2} - \theta) & \sin(\frac{\pi}{2} - \theta) \\ -\sin(\frac{\pi}{2} - \theta) & \cos(\frac{\pi}{2} - \theta) \end{pmatrix} \left[x\, y\right]^T \qquad (5\text{-}2)$$

In this equation f is the frequency of a sinusoidal plane wave and σ_x and σ_y are the standard deviations along the x and y axis of the Gaussian envelope.

To apply Gabor filters to an image the four parameters $(\theta, f, \sigma_x, \sigma_y)$ must be specified. Obviously the frequency of the filter is completely determined by the local ridge frequency and orientation is determined by the local ridge orientation. The selection of the values σ_x and σ_y involves a tradeoff. The larger the values, the more robust the filters are to noise in the fingerprint image, but also the more likely to crate

spurious ridges and valleys. On the other hand, the smaller the values, the less likely the filters are to introduce spurious ridges and valleys but then they will become less effective in removing the noise. In fact, from the modulation transfer function (MTF) of the Gabor filters, it can be shown that increasing σ_x and σ_y decreases the bandwidth of the filter and vice-versa. Based on empirical data, [HWJ98] set σ_x and σ_y both equal to 4. To make the enhancement faster, instead of computing the best-suited contextual filter for each pixel at the run time, a set of filters for some predefined orientations and frequencies are a-priori created and stored. Filter application is done by the use of convolution in the spatial domain.

Greenberg et al. [GAK00] noted that by reducing the value of σ_x with respect to σ_y, the filtering creates fewer spurious ridges and is more robust against noise. In practice reducing σ_x results in increasing the filter bandwidth, independently of the angular bandwidth, which remains unchanged; this allows the filter to better tolerate errors in local frequency estimates. Analogously, one could decrease σ_y in order to increase the angular bandwidth near the singularities where the ridges are characterized by higher curvatures and the orientation changes rapidly.

Yang, [YLJ+03] provided an approach that models the sinusoidal shape of the fingerprint images in a much better way and produces better results in comparison to the algorithm proposed by Hong et al. [HWJ98]. They also proposed a method for the automatic detection and assignment of values for σ_x and σ_y.

Liu, [Liu00] proposed an enhancement algorithm that is based on Gabor filters but assumes that the ridge frequency for a fingerprint image is approximately the same. This dominant frequency is detected by the use of Hough transform applied on the Fourier spectrum of a fingerprint image. He also proposed mathematical relation for automatically finding the values of σ_x and σ_y, depending upon ridge orientations.

5.3 Feature Extraction for Enhancement

Most of the enhancement algorithms given above are based on the local ridge frequency and orientation of the fingerprint image. It is therefore necessary to provide an insight to ridge orientation and ridge frequency extraction algorithms.

5.3.1 Local Ridge Orientation Estimation

The ridge orientation at a pixel $[x, y]$ in a fingerprint image is defined as the angle θ_{xy}, that the fingerprint ridges, crossing through an arbitrary small neighborhood centered at $[x, y]$, form with the horizontal axis. Because fingerprint ridges are not directed, θ_{xy}, is an unoriented direction lying in $[0..180^{\circ}]$. The term direction is used to indicate an orientated direction in $[0..360^{\circ}]$. The fingerprint orientation image O is a matrix whose elements encode the local orientations of fingerprint ridges.

Local ridge orientation estimation is important because it gives us the orientation of the ridges at a given point that is then used to tune contextual filters for feature extraction and image enhancement. Apart from this ridge orientation estimation also gives us the angles for minutiae that occur in a fingerprint.

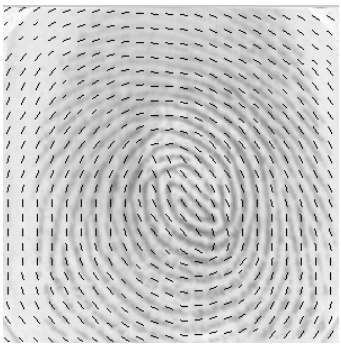

Fig. 5-5 Orientation Field Superimposed on an Image

5.3.1.1 A Survey of Different Orientation Estimation Techniques

The simplest and most natural approach for ridge orientation extraction is based on computation of gradients in the fingerprint image. The gradient $G(x_j, y_j)$ at a point $[x_j, y_j]$ of I, is a 2D vector $[G_x(x_j, y_j), G_y(x_j, y_j)]$, where G_x and G_y components are the derivatives of I in $[x_j, y_j]$ with respect to the x and y directions, respectively. It is known that the gradient phase angle denotes the direction of the maximum pixel intensity change. Therefore, the direction θ_{ij} of a hypothetical edge that crosses the region centered at $[x_j, y_j]$ is orthogonal to the gradient phase angle at $[x_j, y_j]$. This method, although simple and efficient, has some drawbacks. First using the classical Prewitt or Sobel convolution masks [GW92] to determine G_x and G_y components of

the gradient, and computing θ_{ij} as the arctangent of the $\frac{G_y}{G_x}$ ratio, presents problems due to the non-linearity and discontinuity around 90°. Second a single orientation estimate reflects the ridge-valley orientation at too fine a scale and is generally very sensitive to noise in the fingerprint image; on the other hand, simply averaging gradient estimates is not possible due to the circularity of angles: the average orientation between 5° and 175° is not 90° (as an arithmetic average would suggest) but 0°. Furthermore, the concept of average orientation is not always well defined; consider the two orthogonal orientation 0° and 90°; is the correct average orientation 45° or 135°?

Kass et al., [KW87] proposed a simple but elegant solution to the above problem, which allows local gradient estimates to be averaged. The basic idea is to double the angles and then use these angles in averaging calculations. This idea was utilized by Ratha et al. [RCJ95] who computed the dominant ridge orientation θ_{ij} by combining multiple gradient estimates within a 17x17 window centered at $[x_j, y_j]$:

$$\theta_{ij} = 90^\circ + \frac{1}{2}\tan^{-1}\left(\frac{2G_{xy}}{G_{xx} - G_{yy}}\right) \tag{5-3}$$

$$G_{xy} = \sum_{h=-8}^{h=+8}\sum_{k=-8}^{k=+8} G_x(x_i + h, y_j + k)\cdot G_y(x_i + h, y_j + k) \tag{5-4}$$

$$G_{xx} = \sum_{h=-8}^{h=+8}\sum_{k=-8}^{k=+8} G_x(x_i + h, y_j + k)^2 \tag{5-5}$$

$$G_{yy} = \sum_{h=-8}^{h=+8}\sum_{k=-8}^{k=+8} G_y(x_i + h, y_j + k)^2 \tag{5-6}$$

Where G_x and G_y are the x and y gradient components using 3x3 sobel masks.

Bazen et al., [BG02b] have shown that this method is mathematically equivalent to the PCA of the autocorrelation matrix of the gradient vectors. Another gradient based method, independently proposed by Donahue et al., [DR93], relies on least square minimization to perform the average orientation estimates and leads to the same expression.

Different approaches to the computation of the orientation image not directly based on the gradient computation have been proposed by [SS69, MMK87 and KT84]. Stock et al., [SS69] evaluated the local ridge orientations on the basis of pixel alignments relative to a fixed number of reference orientations. The total fluctuation

of the gray-scale is expected to smallest along the orientation of the ridges and largest in the orthogonal direction. Kawajoe et al., [KT84] made a straight comparison against four edge templates to extract a rough directional estimate in each 2x2 pixel neighborhood. These estimates were then arithmetically averaged over a large region to obtain a more accurate measure. These approaches do not provide very accurate estimates mainly because of the small fixed possible orientations.

Finally, other techniques have been proposed in [RB80, ON89, SM92, BLL93, Hun93, SBT+94, MBF+97 and AL00].

The reliability r_{ij} of the estimates θ_{ij} can be derived by the concordance (or coherence) of different orientation estimates in a neighbourhood of $[x_i, y_j]$ [KW87 and BG02b]. Infact, due to the continuity and smoothness of fingerprint ridges, sharp orientation changes often denotes unreliable estimation. JHP+97 computed the coherence of the orientations according to their variance in small 5x5 neighbourhoods whereas Donahue et al., [DR93] computed this according to the residual of the least-square minimization.

The orientation image D, computed from poor quality fingerprints may contain several unreliable elements due to local scratches or cluttered noise. In this situation, regularization or smoothing step is useful in enhancing D.

Other interesting regularization approaches have been proposed by [KT84], [ON89], [Pra97], and [Per98]. Sherlock et al., [SM93] proposed an effective mathematical model to synthesize a fingerprint orientation image from the position of loops and deltas along. Their work is quite different from those discussed until now: instead of starting from a fingerprint image the authors take as input only the position of the singularities. Obviously some simplifying assumptions have been made and the model does not cover all the variabilities of the fingerprint patterns. In nature, different ridge pattern fingerprints may present the same singularities at the same locations; on the other hand, this model can be very useful for several purposes such as orientation image restoration, fingerprint data compression, synthetic fingerprint generation and so on. Improvements of this method have been proposed in [VG96] and [ABC+02]. These new models introduce more degree of freedom to better cope with fingerprint pattern variability.

5.3.1.2 Development of the Ridge Orientation Estimation Algorithm

The ridge orientation estimation technique has the following major steps:

a. Normalization
b. Computation of Gradients
c. Estimation of local Orientations
d. Conversion to a Continuous Vector Field and LP Filtering

a. Normalization

In this step the given fingerprint image $I(x,y)$ is normalized so that it has a pre-specified mean and variance. This step is necessary for the extraction of the gradients. It also aids in minutiae extraction at later stages. The normalized image N is obtained as:

$$N[x,y] = \begin{cases} m_o + \sqrt{(I[x,y]-m)^2 \cdot v_o / v}, & I[x,y] > m \\ m_o - \sqrt{(I[x,y]-m)^2 \cdot v_o / v}, & Otherwise \end{cases} \quad (5\text{-}7)$$

Where m and v are the image mean and variance and m_o and v_o are the desired mean and variance after normalization.

b. Computation of Gradients

The normalized image is divided in to blocks of size wxw (16x16). The gradients G_x and G_y are computed at each pixel using a 3x3 Sobel operators given below.

-1	0	1
-2	0	2
-1	0	1

-1	2	-1
0	0	0
1	2	1

Fig. 5-6 Sobel Operators

c. Estimation of local Orientation

For each block centered at $[x_j, y_j]$ the local orientation is estimated by using the equations 3, 4, 5, and 6. (ows) is the orientation window size.

d. Conversion to a Continuous Vector Field and LP Filtering

Due to the presence of noise, corrupted ridge and furrow structures, minutiae, etc. in the input image, the estimated local ridge orientation may not always be a correct

estimate. Since local ridge orientation varies slowly in a local neighborhood when no singular points appear, a low pass filter can be used to modify the incorrect local ridge orientation. In order to perform the low pass filtering, the orientation image needs to be converted into a continuous vector field, which is defined as follows:

$$\phi_x(i, j) = \cos(2\theta(i, j)) \tag{5-8}$$

$$\phi_y(i, j) = \sin(2\theta(i, j)) \tag{5-9}$$

A $w_s x w_s$ averaging filter is convolved with $\phi_x(i, j)$ and $\phi_y(i, j)$ to yield $\phi_x'(i, j)$ and $\phi_y'(i, j)$ respectively. The local ridge orientation is estimated as:

$$O(i, j) = \frac{1}{2}\tan^{-1}\left(\frac{\phi_y'(i, j)}{\phi_x'(i, j)}\right) \tag{5-10}$$

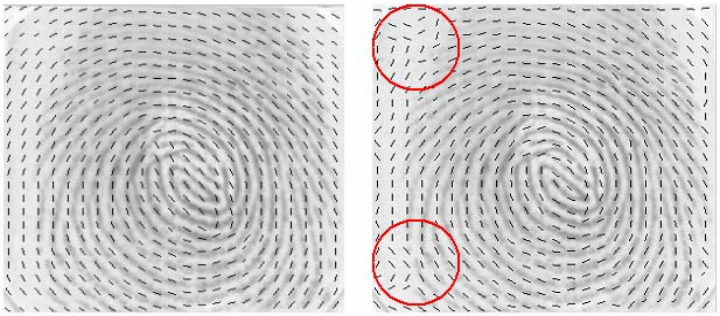

Fig. 5-7 Orientation Field without Low Pass Filtering and with Low pass Filtering

5.3.1.3 Results

The results of the fingerprint local ridge orientation estimation process are shown below. This algorithm takes about 0.4 seconds on a P4 2.0 GHz machine with 256MB RAM to execute on a 300x300 image.

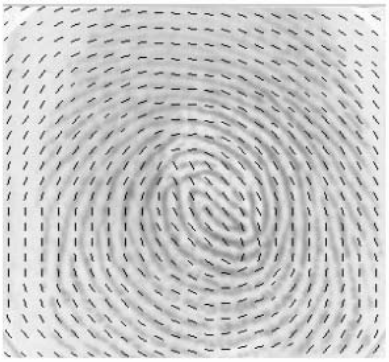

Fig. 5-8 Results of Ridge Orientation Estimation

5.3.2 Local Ridge Frequency Estimation

The local ridge frequency (or density) f_{xy} at point $[x, y]$ is the inverse of the number of ridges per unit length along a hypothetical segment centered at $[x, y]$ and orthogonal to the local ridge orientation θ_{xy}. A frequency image F, analogous to the orientation image can be defined if the frequency is estimated at discrete positions and arranged into a matrix.

Local ridge frequency estimation is useful since it provides us with a way to tune contextual filters that are used for feature extraction and enhancement.

5.3.2.1 A Survey of Different Ridge Frequency Extraction Techniques

The local ridge frequency varies across different fingers, and may also noticeably vary across different regions in the same fingerprint. Hong et al. [HWJ98] estimate local ridge frequency by counting the average number of pixels between to consecutive peaks of gray-level along a direction normal to the local ridge orientation. For this purpose, the surface S corresponding to the fingerprint is sectioned with a plane parallel to the z-axis and orthogonal to local ridge orientation. The frequency at $[x_i, y_j]$ is computed as follows.

A 32x16 oriented window cenetered at $[x_i, y_j]$ is defined in the ridge coordinate system i.e. rotated to align the y-axis with the local ridge orientation.

The X-signature of the gray level is obtained by accumulating, for each column X the gray levels of the corresponding pixels in the oriented window. This is a sort of averaging that makes the gray level profile smoother and prevents ridge peaks from being obscured due to small ridge breaks or pores.

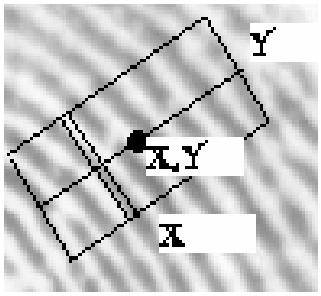

Fig. 5-9 A 32x16 Oriented Window

F_{ij} is determined as the inverse of the average distance between two consecutive peaks of the X-signature.

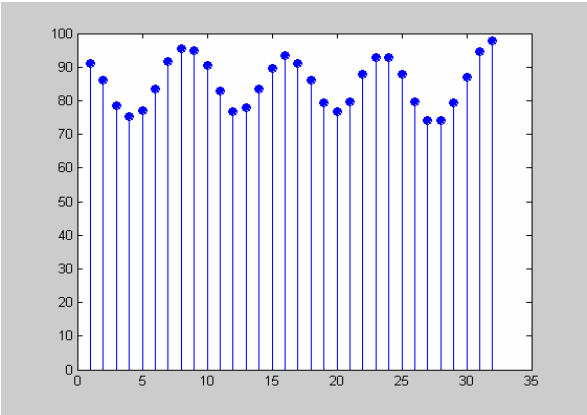

Fig. 5-10 The X-Signature

The method is simple and fast. However it is difficult to reliably detect consecutive peaks of gray levels in the spatial domain in noisy fingerprint images. In this case the authors suggest using interpolation and low pass filtering.

Jiang [Jia00] also computes the local ridge frequency starting from the X-signatures. However instead of measuring the distances in the spatial domain he makes use of a high order spectrum technique called Mix Spectrum. The ridge patterns in a fingerprint image are noisy periodic signals. When they deviate from a pure sinusoid shape, their energy is distributed to their fundamental frequency and harmonics. The mix spectrum technique enhances the fundamental frequency of the signal by exploiting the information contained in the second and third harmonics.

In the method proposed in [MM98a] the ridge pattern is locally modeled as a sinusoidal shaped surface and the variation theorem is exploited to estimate the unknown frequency.

Kovacs-Vajna et al., [KRF00] proposed a two step procedure: first the average ridge distance is estimated for each 64x64 sub-block of the image that is of sufficient quality and then this information is propagated, according to a diffusion equation, to the remaining regions.

Almansa et al., [AL97, AL2000] use scale space theory to locally estimate ridge width; their approach relies upon combinations of normalized derivatives computed point wise.

5.3.2.2 Development of the Local Ridge Frequency Extraction Algorithm

We use the method proposed in [HWJ98] for ridge frequency estimation because of its simplicity, fast execution and acceptable performance. The algorithm consists of the following steps:

a. Normalization (see Section 5.3.1)
b. Extraction of a wxw block X-Signature
c. Estimation of the X-Signature Frequency
d. Low pass filtering (post-processing)

The details of these steps are given henceforth.

b. Extraction of X-Signature

The normalized image is divided into wxw (16x16 by default) blocks. For each block centered at pixel (i, j) the oriented window of size (lxw) (32x16) is defined in the ridge coordinate system (Fig. 5-9). The X-signature, $X[0], X[1], ...X[l-1]$ of the ridges in the block are computed as:

$$X[k] = \frac{1}{w}\sum_{d=0}^{w-1} G(u,v), k = 0,1,...,l-1 \tag{5-11}$$

$$u = i + (d - \frac{w}{2})\cos O(i, j) + (k - \frac{l}{2})\sin O(i, j) \tag{5-12}$$

$$v = j + (d - \frac{w}{2})\sin O(i, j) + (\frac{l}{2} - k)\cos O(i, j) \tag{5-13}$$

Where O is the orientation image and G is the normalized image.

c. Estimation of the X-Signature Frequency

For each block, the peaks in the X-Signature are found and their positions are noted. The peaks can be found by utilizing the following algorithm:

```
For m =1 to l-2
     If(X[m]>X[m-1] and X[m]>X[m+1]);
          A peak lies at m
     End If
End For
```

However this process was replaced by the use of Matlab's 'imregionalmax' function that returns the positions of the peaks. The positions of the peaks of the X-Signature were stored in a vector $P_{positions}$. The first difference of this vector was taken

to give the differences between consecutive peaks that results in a vector P_{diff}. The mean of P_{diff} gives the average number of pixels, T between two consecutive peaks. The inverse of T gives the frequency estimate for the block centered at $[x_i, y_j]$. The ridge frequency image is obtained as

$$\Omega(i, j) = \frac{1}{T} \tag{5-14}$$

Alternate Method

Another method for the estimation of the ridge frequency is to subtract the mean of the X signature from the X signature to give the normalized X-signature itself and then find the roots of the normalized X-signature. The difference between the roots of the equation gives half of the time period 'T'. The frequency is given by equation (14).

d. Low Pass Filtering (Post-processing)

The ridge frequency image thus obtained is not very smooth; In order to smooth this image an 8x8 averaging filter is used.

5.3.2.3 Results

The results of the ridge frequency estimation process are shown in Fig. 5-11.

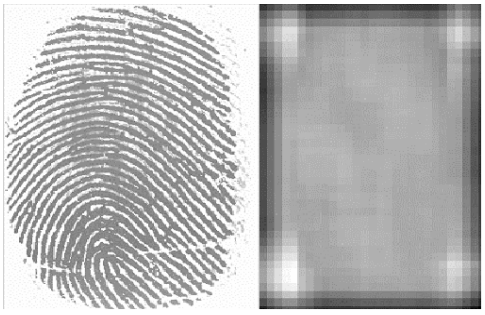

Fig. 5-11 The Ridge Frequency Image

5.3.3 Dominant Ridge Frequency Estimation

The dominant ridge frequency is the most frequently occurring frequency component for a fingerprint image. The objective of dominant ridge frequency estimation is to use the frequency value obtained in enhancement methods that utilize the dominant ridge frequency only.

5.3.3.1 Development of an Algorithm for Estimation of the Dominant Ridge Frequency

We have used the dominant ridge frequency finding technique proposed in [Liu00. The different steps involved in this process are:

a. Image Contrast Enhancement
b. Fast Fourier Transform of the image
c. Centralization of the zero frequency component
d. Extraction of Region of Interest
e. Application of Hough Transform

a. Image Contrast Enhancement

Image contrast is improved by the use of adaptive histogram equalization. [GW92].

b. Fast Fourier Transform of the Image

The contrast-stretched fingerprint image is converted to the frequency domain by the use of PxQ, 2D Fast Fourier Transform.

$$F = \Im(I) \tag{5-15}$$

c. Centralization of the zero frequency Component

The zero frequency components, by default, do not lie at the center of the image. In order to move the zero frequency components to the center, the image in the spatial domain is multiplied by $(-1)^{x+y}$ prior to taking the FFT of the image. Due to the properties of the exponentials:

$$\Im[I(x,y)(-1)^{x+y}] = F(u - M/2, v - N/2) \tag{5-16}$$

where [M N] is the size of the input image. The magnitude of the Fourier spectrum is given by:

$$|F(u,v)| = [\text{Real}^2(I) + \text{Imag}^2(I)]^{1/2} \tag{5-17}$$

The magnitude image of the spectrum is used for further processing.

d. Extraction of ROI

The range in which the dominant frequency can vary is limited to $\{1/3..1/25\}$. So we can discard the extra regions that are present in the Fourier spectrum. This is accomplished by cropping some area outside the $1/25$ threshold and by setting the values within the $1/3$ region to zero e.g. for a 364x256 Fourier image the central 21x21 square is set to zero and only the central 201x201 square area is used for further processing. The area outside the central 201x201 square is also cropped off.

This is also illustrated in the Fig. 5-12. The ROI Extraction process saves a considerable amount of computation time.

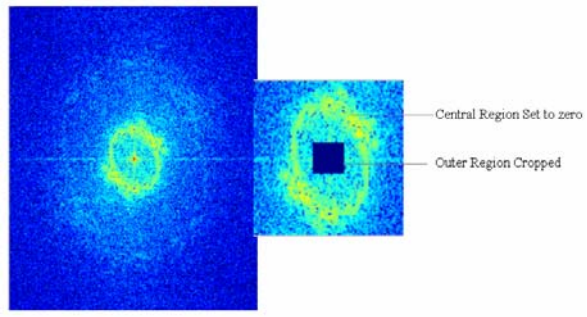

Fig. 5-12 Fourier Spectrum of a Fingerprint and the extracted ROI

e. Application of Hough Transform

Hough transform is used to find the circular ring by creating a circle one pixel thick within a square mask with size equal to the ROI extracted and multiplying it pixel wise with the ROI to give a radial response square. All the values in the radial response square are summed up. The process is repeated for differ radii and the radius 'R' for which the sum of the values in radial response square is the maximum is stored. The search for R is carried out within a limited region of the ROI. The dominant frequency is given by:

$$f = \frac{R}{\min(P,Q)} \tag{5-18}$$

5.3.3.2 Results

The various steps involved in this process are shown pictorially in Fig. 5-13. It takes 0.06 s on a P-IV, 2.2 GHz with 512MB RAM.

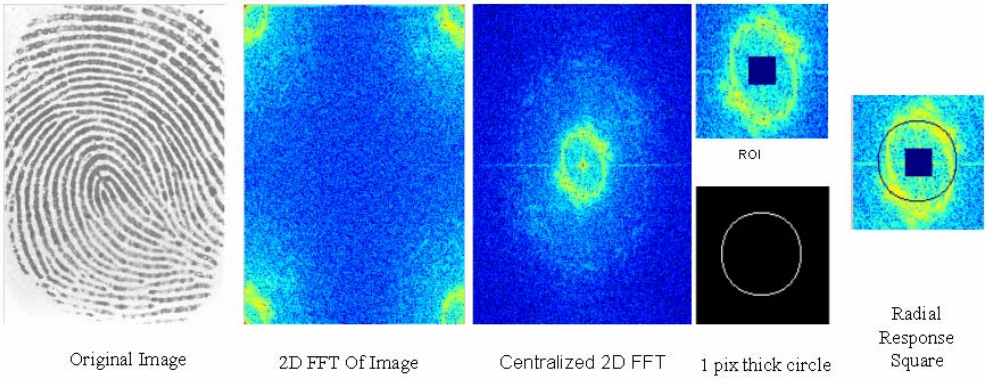

Fig. 5-13 Different Steps involved in Ridge Orientation Estimation

5.4 Implementation of Fingerprint Enhancement Algorithm

We have implemented four different approaches to fingerprint image enhancement, which are described henceforth.

5.4.1 Inverse Fourier Filtering for Enhancement

5.4.1.1 Proposed Methodology

This technique, proposed by Willis and Myers (2001) [WM01], performs contextual filtering based fingerprint image enhancement without requiring explicitly the computation of local ridge orientation and frequency. In this approach each 32x32 block in the image is enhanced separately; the Fourier transform of the block is multiplied by its power spectrum raised to a power k:

$$I_{enh}[x,y] = \Im^{-1}\left\{\Im(I[x,y]) \times \left|\Im(I[x,y])\right|^{k}\right\} \qquad (5\text{-}19)$$

The power spectrum contains information about the underlying dominant ridge orientation and frequency and the multiplication has the effect of enhancing the block accordingly.

The major steps involved in this process include (see Fig. 5-14)

Fig. 5-14 Enhancement by using the approach by Willis and Myers (2001)

a. Extraction of Small Windows from the Input Fingerprint

Directional information, which is of great interest because of the fingerprint's Ridge/valley structure, is contained in the magnitude of the frequency domain representation of the image. But if the whole image, including the whole fingerprint and thus all directions, is considered it naturally becomes hard to extract useful information from it. Instead of looking at the whole image, small windows are extracted from it and each window is processed separately.

By choosing the window size so that the image contains two or three roughly parallel ridges the dominant frequencies are presumed to correspond to the ridges in

the window. Moreover overlapping windows are used for removing any blocking effects that might appear if non-overlapping windows are used.

b. Application of the Weighted 2D FFT Method

This is achieved by at first taking the two-dimensional Fourier transform of the window, which can be computed according to

$$F(u,v) = \Im[f(x,y)] = \sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)\cdot e^{-2\pi j(\frac{ux}{M}+\frac{vy}{N})} \tag{5-20}$$

for u = 0, 1, 2, …,M and v = 0, 1, 2, …,N , where f(x, y) is the image, M is the width of the image and N is the height of the image.

Subsequently the above-mentioned Fourier transform is multiplied with its magnitude and finally the inverse two-dimensional Fourier-transform of that product is calculated.

The inverse two-dimensional Fourier transform may be calculated as follows,

$$f(x,y) = \Im^{-1}[F(u,v)] = \frac{1}{MN}\sum_{u=0}^{M-1}\sum_{v=0}^{N-1} F(u,v)\cdot e^{2\pi j(\frac{ux}{M}+\frac{vy}{N})} \tag{5-21}$$

for x = 0, 1, 2, …,M and y = 0, 1, 2, …,N. Now each window is indeed enhanced, but by using a power of the magnitude instead of just the magnitude itself as given below by equation 19.

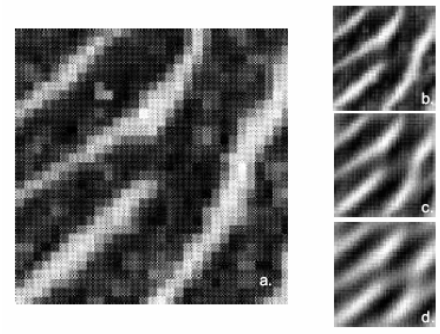

Fig. 5-15 (a) Original Image, (b) Enhanced Images with (b) k=1.2, (c) k=1.4, (d) k=1.7

The higher the power factor k is, the higher contrast and the better separation between ridges and valleys is achieved. On the other hand if a minutia is included in the window, it is highly likely that the branch that forms the minutia differs from the main direction in the window and thus may be obscured by a too high power factor. Concisely there is a tradeoff between high contrast and the risk of concealing important minutiae. Since the minutiae are vital for the final results, the latter is not an option. In this specific application power factors, k, between one and two were

considered (see Fig. 5-15) and k=1.4 was chosen as the optimal tradeoff between high contrast and possible obscuring of vital minutiae.

5.4.1.2 Results

Fig. 5-16 shows the results of this algorithm for a fingerprint image.

Fig. 5-16 Results of Fingerprint Image Enhacement using Willis and Myers (2001)

5.4.2 Block-wise Application of Gabor Filter for Enhancement

5.4.2.1 Proposed Methodology

This approach [HWJ98] is based on the use of Gabor filters for fingerprint enhancement. The following steps are involved in the development of this algorithm:

a. Normalization
b. Local Ridge Orientation Estimation
c. Local Ridge Frequency Estimation
d. Region Mask Generation
e. Filtering

Steps 1, 2 and 3 have already been detailed in the previous chapters. Let $I[x, y]$ be the input image then the normalized version is given by equation 7. The local-orientation field is obtained by the process described in Section 5.3.1. The local ridge frequency is calculated by the use of the techniques proposed in Section 5.3.2 and is denoted by $\Omega(i, j)$.

a. Region Mask Generation

As mentioned earlier, a pixel (or a block) in an input fingerprint image could be either in a recoverable region or an unrecoverable region. Classification of pixels into recoverable and unrecoverable categories can be performed based on the assessment of the shape of the wave formed by the local ridges and furrows. In this algorithm, three features are used to characterize the sinusoidal-shaped wave: amplitude (α), frequency (β) and variance (γ). Let X [1], X [2]... X[l] be the x-signature of a block centered at (i; j). The three features corresponding to pixel (block) (i, j) are computed as follows:

1. α = (avrage height of peaks - average depth of valleys)
2. $\Omega(i, j)$
3. $\gamma = \frac{1}{l}\sum_{i=1}^{l}(X[i] - (\frac{1}{l}\sum_{j=1}^{l}X[j]))^2$

The authors selected several typical fingerprint images with both labeled recoverable and unrecoverable regions and computed these three features. A total of 2,000 3-dimensional patterns were obtained by the authors. In order to find representative patterns for the two classes, the authors fed the 2,000 patterns to a squared-error clustering algorithm and identified six clusters. Four of these clusters correspond to recoverable regions and the remaining two correspond to unrecoverable regions. The six prototypes (corresponding to cluster centers) were used in a one-nearest neighbor (1NN) classifier to classify each w x w block in an input fingerprint image into a recoverable or unrecoverable block. If a block centered at (i; j) is recoverable, then

R (i, j) = 1, else R (i, j) = 0. After the image R is obtained, the percentage of recoverable regions is computed. If the percentage of recoverable regions is smaller than a threshold, $\Gamma_{\text{recoverable}}$ = 40, then the input fingerprint image is rejected. An accepted image is then passed through the filtering stage.

We did not implemented region filtering and the whole of the image is passed on to the filtering stage.

b. Filtering

An Introduction to Gabor Filters

A Gabor filter is given mathematically by:

$$g(x, y : \theta, f) = \exp\left\{-\frac{1}{2}\left[\frac{x_\theta^2}{\sigma_x^2} + \frac{y_\theta^2}{\sigma_y^2}\right]\right\} \cdot \cos(2\pi f \cdot x_\theta) \tag{5-22}$$

Where θ is the orientation of the filter and $[x_\theta, y_\theta]$ are the coordinates of $[x\ , y\ ]$ after a clockwise rotation of the Cartesian axes by $\frac{\pi}{2} - \theta$.

$$\left[x_\theta\ y_\theta\right]^T = \begin{pmatrix} \cos(\frac{\pi}{2} - \theta) & \sin(\frac{\pi}{2} - \theta) \\ -\sin(\frac{\pi}{2} - \theta) & \cos(\frac{\pi}{2} - \theta) \end{pmatrix} \left[x\ y\right]^T \tag{5-23}$$

In this equation f is the frequency of a sinusoidal plane wave and σ_x and σ_y are the standard deviations along the x and y axis of the Gaussian envelope. The Gabor filter has two components as given below:

$$g(x, y; f, \varphi) = h_x(x; f, \varphi) \cdot h_y(y; \varphi) \tag{5-24}$$

$$h_x(x; f, \varphi) = \mathrm{e}^{-\frac{1}{2}(\frac{x_\varphi^2}{\sigma_x^2})} \cdot \cos(2\pi x_\varphi f) \tag{5-25}$$

$$h_y(y; \varphi) = \mathrm{e}^{-\frac{1}{2}(\frac{y_\varphi^2}{\sigma_y^2})} \tag{5-26}$$

h_X behaves as a traditional Gabor function (the multiplication of a Gaussian with a sinusoid) which is a band-pass filter and h_Y represents a Gaussian function which is a low-pass filter. Therefore, TGF has a band-pass (differentiating) effect in the direction orthogonal to the ridges, which increases the discrimination between ridges & valleys. It has a low-pass (averaging) effect along the ridge direction fulfilling the aim of linking small gaps and filling impurities. These two properties of the Gabor filters make them ideal for use in fingerprint enhancement.

In the frequency domain the Gabor Filters are given as:

$$H(u, v) = \exp\left\{-\frac{1}{2}\left[\frac{(u' - u_o')^2}{\sigma_u^2} + \frac{(v' - v_o')^2}{\sigma_v^2}\right]\right\} \tag{5-27}$$

where,

$$\sigma_u = \frac{1}{2\pi\sigma_x} \tag{5-28}$$

$$\sigma_v = \frac{1}{2\pi\sigma_y} \tag{5-29}$$

$$\left(u', v'\right) = (u\cos\theta + v\sin\theta, -u\sin\theta + v\cos\theta) \tag{5-30}$$

The implementation of the Gabor filters can be understood in a much better way by the use of the Fourier spectrum interpretation of the fingerprint images and the Gabor filters themselves. Since Gabor filters possess both frequency related and orientation related parameters therefore we must study the effects of changes in the orientation and ridge frequency on the Fourier spectrum of the fingerprint image as application of the Gabor filters is carried out in the frequency domain by the use of multiplication rather than using convolution in the time domain in order to save computational time.

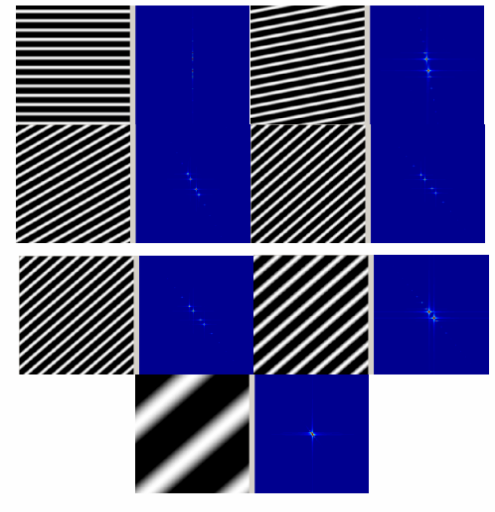

Fig. 5-17 Original images and their Fourier spectra for different orientations (above 4x2 images) and frequencies (below 3x2 images)

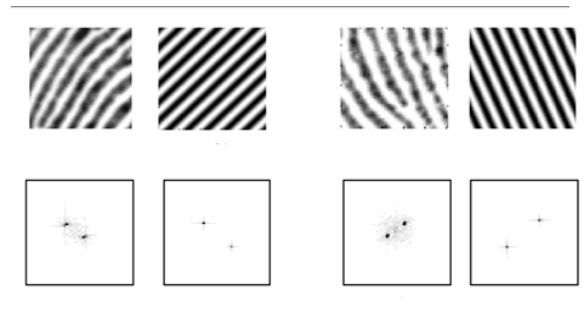

Fig. 5-18 Fingerprint Ridges Modeled as lines of Different Orientations

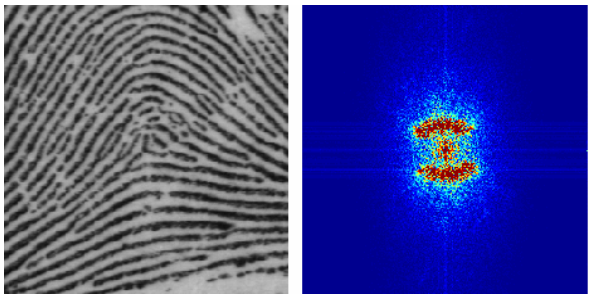

Fig. 5-19 Original Image and its Fourier Spectrum

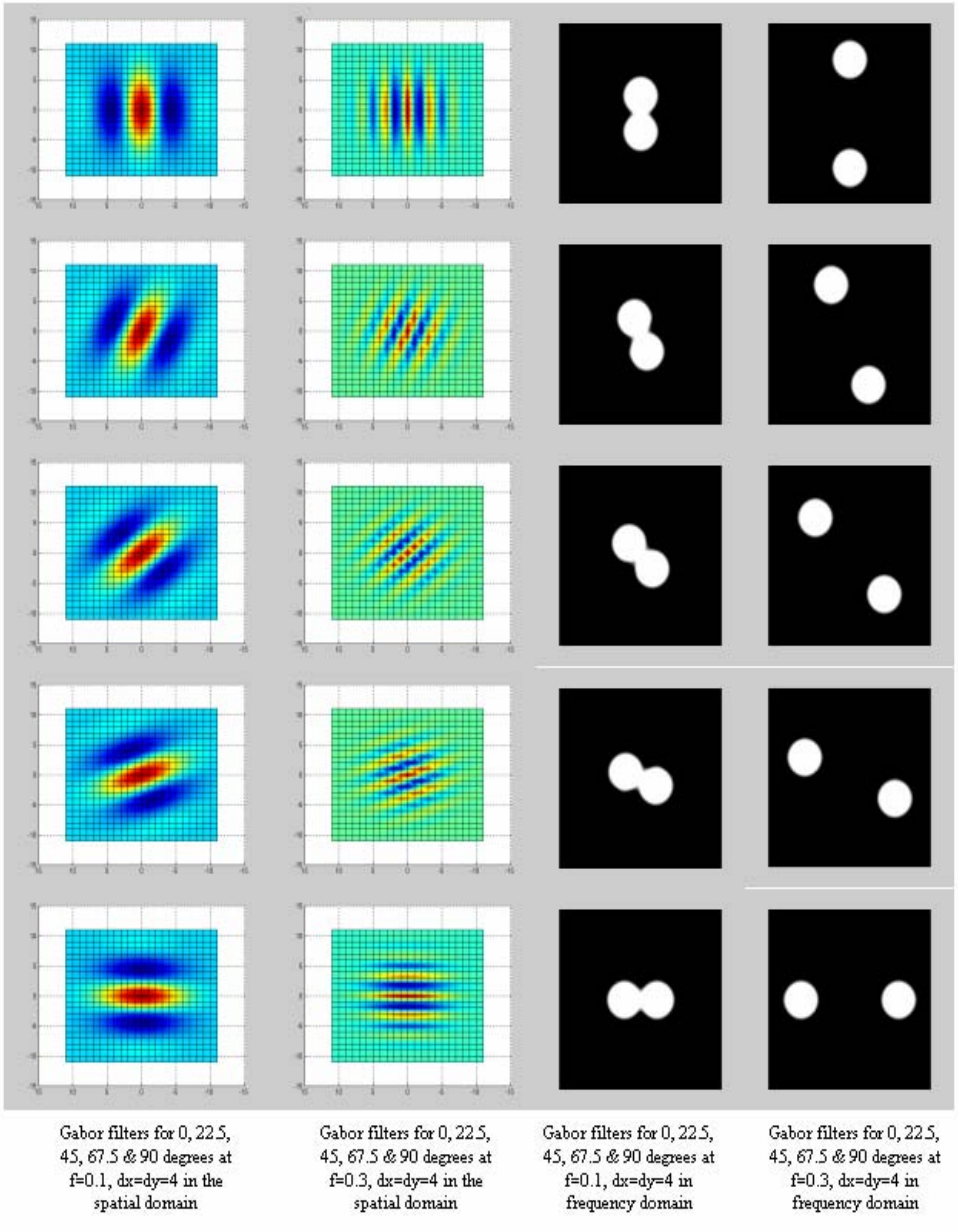

Fig. 5-20 Gabor Filters

Fig. 5-18 shows the Fourier spectra for different orientations and images. It can easily be observed that changes in the ridge frequency affect the radial positioning of the brighter dots in the Fourier spectrum whereas changes in orientation affect the angular positioning of the bright dots. A fingerprint image can be thought as a set of lines of different orientations and ridge frequencies. This model is further illustrated in the Figure below.

Fig. 5-19 shows the Fourier spectrum of a fingerprint image. A ring like pattern is quite prominent in the Fourier spectrum of the fingerprint image which tells us that a fingerprint possesses a dominant frequency and the local frequencies vary only by small amounts from the ridge frequency and that the fingerprint image contains different ridge orientations.

Fig. 5-20 shows Gabor filters tuned to different frequencies and orientations and their frequency spectrum counterparts.

It is clearly evident that a multiplication of a part of the fingerprint with a Gabor filter tuned to a frequency f and orientation θ in the frequency domain enhances the portion of the fingerprint where the orientation is equal to θ and the ridges have a frequency of f.

Application of the Gabor Filters

For the application of the filter we carry out the following steps:

1. Divide the original fingerprint image into windows of size wxw. These windows have an overlap of $\frac{1}{4}w$ with each other.

2. For each of the windows the local ridge orientation is obtained from the orientation field.

3. For each of the windows the local ridge frequency is calculated.

4. A Gabor filter tuned to the local ridge frequency and orientation is formed up.

5. The FFT of the Gabor filter is calculated.

6. The FFT of the fingerprint window is calculated.

7. The FFTs of the Gabor filter and the fingerprint window are multiplied

8. The inverse FFT of the product gives us the enhanced version of the fingerprint window.

5.4.2.2 Results

The time taken by this algorithm is 5.276s on a P4 1.8 GHz machine with 256MB RAM. If the time of all the processes i.e. Ridge orientation estimation and ridge frequency estimation then the time taken is 7.633s.

Fig. 5-21 Original Image and its enhanced version

5.4.3 Gabor Filter Bank Based Enhancement

5.4.3.1 Proposed Methodology

This approach [Liu00] provides a much efficient approach to the fingerprint enhancement problem using a Gabor Filter Bank. In this algorithm a set of K Gabor filters each having its central frequency equal to the dominant frequency of the fingerprint image is created. These Gabor filters have the orientation of $0, \Delta\theta$, $2\Delta\theta$, $3\Delta\theta$ … 180-degrees, where

$$\Delta\theta = 180 / K \tag{5-31}$$

Each of these filters is convolved with the whole of the fingerprint image by using multiplication in the frequency domain to produce K different convolution results as shown in Fig. 5-22.

$$f_i = I \otimes g_i \qquad i = 1, 2, ..., K \tag{5-32}$$

The energy of the filtered images is also calculated as:

$$e_i = f_i^2 \qquad i = 1, 2, ..., K \tag{5-33}$$

Fig. 5-23 shows these energy images.

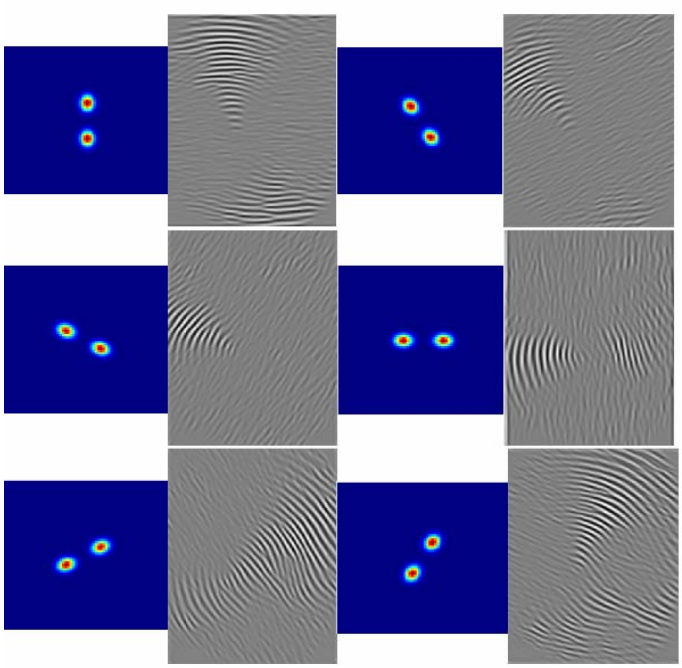

Fig. 5-22 Generation of Gabor Filter Response Images

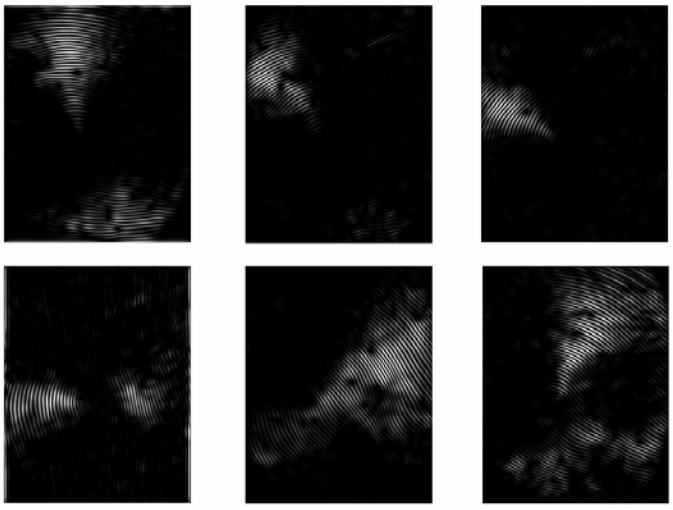

Fig. 5-23 Filter Energy Responses

These energy images are smoothed by using a low pass filter Γ as follows:

$$e_i^{'} = e_i \otimes \Gamma \qquad i = 1,...,K \tag{5-34}$$

The low pass filter employed is a gaussian low pass filter:

$$\Gamma = \frac{1}{2\pi\sigma}\exp\left[-\frac{(x^2+y^2)}{2\sigma^2}\right] \tag{5-35}$$

The smoothed energy images are shown in Fig. 5-24.

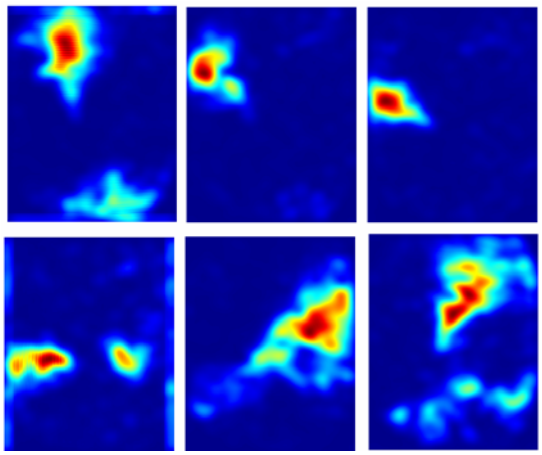

Fig. 5-24 Smoothed Energy Images

The resultant $e_i^{'}$ can be used as region masks. The bright areas show the regions of interest in different filtered images. The high response region images can be used to emphasize the components in different filtered images:

$$f_i^{'}(x,y) = f_i^{'}(x,y) \times e_i^{'}(x,y) \tag{5-36}$$

The results of this weighting operation are shown in Fig. 5-25.

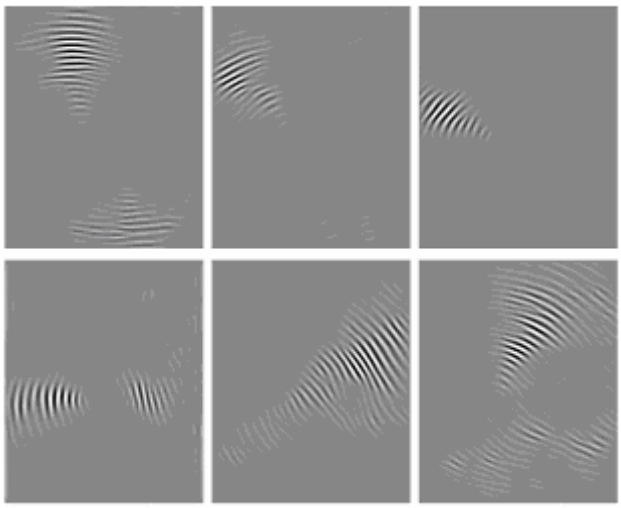

Fig. 5-25 After Weighting the Filter Responses using the Energy Images as Weighting Factors

The enhanced image is obtained by adding these images.

$$I_e(x,y) = \sum_{i=1}^{K} f_i^{'}(x,y) \tag{5-37}$$

The enhanced image is shown in Fig. 5-26, which shows significant noise removal when compared with the original image.

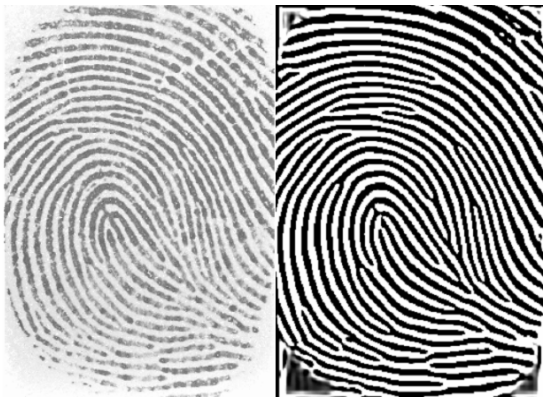

Fig. 5-26 Original and the Enhanced Image

This enhancement method also provides a segmentation mask, which is a binarized version of the sum of the smoothed filter energy sub images and indicates the region of interest in a fingerprint. It is given by:

$$S_m = \sum_{i=1}^{K} e_i' \tag{5-38}$$

Fig. 5-27 shows the segmentation mask for a fingerprint image. Small holes in the segmentation mask are removed by using a fill-algorithm.

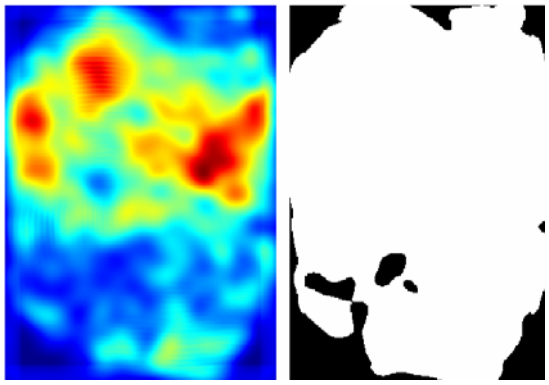

Fig. 5-27 The Sum of the Smoothed Energy Images and the Resulting Segmentation Mask Obtained through Mean and Standard Deviation based Thresholding

To describe the ridges in a fingerprint, only using the central frequency is not enough since the desired information about the ridges spreads around the central frequency in the frequency domain representation of a fingerprint. This phenomenon is caused by:

The frequency in different areas are not exactly the same throughout the fingerprint image as there are deviations among the ridges

Minutiae details change the ridge structure therefore frequency contents are changed in the presence of minutiae in a fingerprint region.

The Gabor filters need to be tuned to an appropriate bandwidth to extract the frequency components about the ridges and remove the unwanted elements. In equations 22 and 27 the bandwidth is determined by the parameters σ_x and σ_y. These parameters can be obtained by:

$$\sigma_u = \frac{(2^{B_u} - 1)W}{(2^{B_u} + 1)\sqrt{2\ln 2}} \tag{5-39}$$

$$\sigma_v = \tan\left(\frac{B_\theta}{2}\right)\frac{W}{\sqrt{2\ln 2}} \tag{5-40}$$

In these equations, W is the central frequency (dominant frequency), B_θ (the angular bandwidth) depends upon the number of orientations, K as:

$$B_\theta = \frac{\pi}{K} \tag{5-41}$$

The radial bandwidth, B_u, is the only undetermined parameter. A well-known fact is that humans can find the ridges in a fingerprint easily, and the frequency bandwidth of simple cell in the visual cortex is about 1 octave. Therefore the value of B_u, was taken to be 1 octave.

5.4.3.2 Results

Fig.5-28 shows the results of the enhancement algorithm. This algorithm is implemented in the file fpEnhanceEA2.m. This algorithm requires only 2.6s for enhancement on a P4 2.2 GHz machine with 512MB RAM.

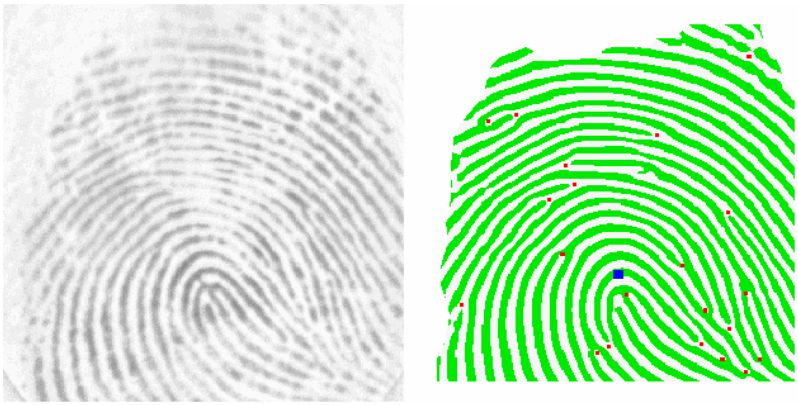

Fig. 5-28 The Enhanced, Segmented and Binarized version of a Fingerprint

5.4.4 Fourier Transform Based Contextual Filtering

5.4.4.1 Proposed Methodology

This method [CWG04] is based on directional Fourier domain filtering for fingerprint enhancement. This method uses novel approaches to fingerprint image orientation and ridge frequency estimation. The various steps involved in this approach include:

a. Fourier Domain Analysis
b. Directional Field Estimation
c. Ridge Frequency Estimation
d. Enhancement

These steps are described in detail henceforth.

a. **Fourier Domain Analysis**

This method models a fingerprint image block using a surface wave that is characterized completely by its orientation ϕ and frequency f. The parameters of the surface wave can be easily obtained from its Fourier transform that consists of two impulses whose distance from the origin indicates the frequency and its angular location indicates the orientation of the wave.

b. Directional Field Estimation

For the approximation of each block by a single orientation and frequency, a probabilistic approximation is used. We can represent the Fourier Spectrum in polar form as $F(r,\phi)$. Using this, a probabilistic density function $f(r,\phi)$ cane be defined along with the marginal density functions $f(\phi)$ and $f(r)$ as:

$$f(r,\phi) = \frac{|F(r,\phi)|^2}{\int_r \int_\phi |F(r,\phi)|^2 \, d\phi dr} \tag{5-42}$$

$$f(\phi) = \int_r f(r,\phi) dr \tag{5-43}$$

$$f(r) = \int_\phi f(r,\phi) d\phi \tag{5-44}$$

Here the assumption is that the orientation ϕ is a random variable that has the probability density function $f(\phi)$. The expected value of the orientation, by then be obtained by:

$$E\{\phi\} = \int_{\phi} \phi \cdot f(\phi) d\phi \tag{5-45}$$

c. Ridge Frequency Estimation

The average ridge frequency is estimated in a manner similar to the ridge orientation. Here the ridge frequency is taken to be a random variable with the probability density function $f(r)$. The expected value of the ridge frequency is given by:

$$E\{r\} = \int_{r} r \cdot f(r) dr \tag{5-46}$$

The frequency image obtained is smoothed by applying a 3x3 gaussian mask.

d. Enhancement

The authors have proposed a Fourier Domain based block-wise contextual filter approach for enhancement. The image is dived into 16x16 overlapping blocks that is filtered in the Fourier domain by a frequency and orientation selective filter whose parameters are based on the estimated local ridge orientation and frequency. The angular bandwidth of the filter adapts itself in the regions of high curvature (e.g. singular points) by using the directional histogram based on the assumption that $f(\phi)$ is unimodal and centered around $E\{\phi\}$ and defining the bandwidth as the angular extent where $P\left\{\left|\phi - E\{\phi\}\right| < \phi_{BW}\right\} = 0.5$. The filters are given by:

$$H(r,\phi) = H_{radial}(r) H_{angle}(\phi) \tag{5-47}$$

For the radial filter any classical 1-D band pass filter would be adequate; the butterworth filter is used because its implementation is simpler than such alternatives as the Chebyshev or Elliptic filter, especially if it is desired to vary the filter order.

$$H_{radial}(r) = \sqrt{\frac{(r \cdot r_{BW})^{2\pi}}{(r \cdot r_{BW})^{2\pi} + (r^2 - r_o^2)^{2n}}} \tag{5-48}$$

Here r_{BW} and r_o are the desired bandwidth and center frequency. A value of n=2 is used.

In designing the angular filter, one cannot be guided by the analogy to 1D filters because there is no meaningful 1D concept of orientation. KWG83 have used the following function:

$$H_{angle}(\phi) = \begin{cases} \cos^2 \dfrac{\pi}{2} \dfrac{(\phi - \phi_c)}{\phi_{bw}} & if\ |\phi| < \phi_{bw} \\ 0 & Otherwise \end{cases} \tag{5-49}$$

Here ϕ_{bw} is the angular bandwidth of the filter, i.e. the range of angles for which $\left|H_{angle}(\phi)\right| \geq 0.5$ and ϕ_c is its orientation at which $\left|H_{angle}(\phi)\right|$ would be a maximum (mean orientation).

5.4.4.2 Results

The results of the algorithm are shown in Fig. 5-29. This algorithm takes X seconds on a P-IV, 2.2 GHz processor with 512MB RAM.

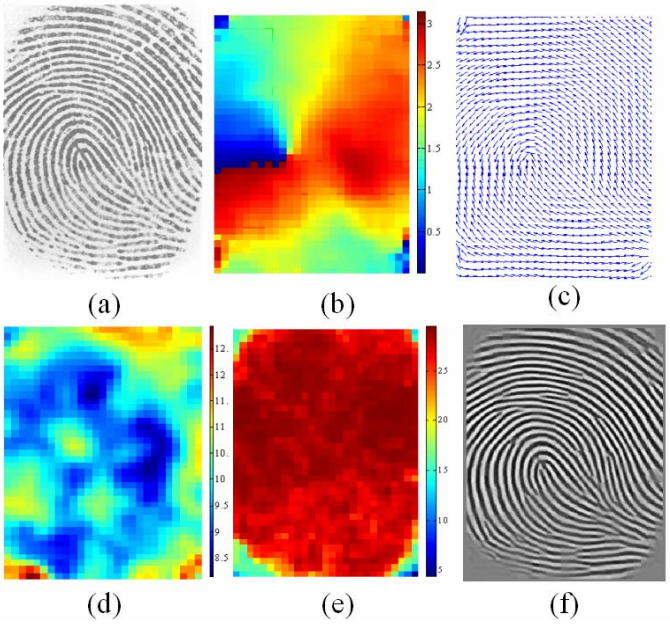

Fig. 5-29 Different Steps involved in the Enhancement Algortitm. (a) Original Image, (b) Orientation Matrix, (b) Orientation Field, (d) Ridge Frequency Map, (e) Filter Energy Response, (f) Enhanced Image

Summary

Fingerprint image enhancement is used to improve the visual quality of a fingerprint image. Contextual filters are used for fingerprint image enhancement, which utilize the local ridge orientation and frequency data of the fingerprint. In this project, we have implemented four different contextual filter based approaches to fingerprint image enhancement, which include Willis and Myers (2001), HWJ98, Liu00 and CWG04.

6 Fingerprint Image Quality Evaluation

In most of the segmentation algorithms only two classes are distinguished i.e. the foreground and the background. Extracting the features from the foreground image consisting of fingerprint pattern without considering the quality of the foreground image sometimes may lead to wrong identification and verification. It is due to false feature extraction from the areas of low quality in the fingerprint. The introduction of a quality evaluation algorithm that classifies the foreground of the fingerprint further into different regions based on their quality can prove to be very useful for determining the reliability of the recognition features extracted in a given region of the foreground. The poor quality of the fingerprint images is the major cause of the poor accuracy of a fingerprint identification system. Therefore, it is important to analyze the quality of a fingerprint image. Quality refers to the 'processability' of the image.

6.1 Objectives of Fingerprint Image Quality Evaluation

The performance of a feature extraction and matching algorithm is affected by the quality of the fingerprint image. The quality of a fingerprint image is degraded by a large number of factors such as inadequate contact of the fingertip with the sensor surface, sensor noise (e.g. dust, grease etc.), age and nature of work of the subject, skin conditions etc. Fig. 6-2 shows fingerprint images of different quality. Fig.6-1 (a) shows a fingerprint image of very good quality in which reliable minutiae extraction can be performed easily. Minutiae cannot be properly extracted from certain regions in Fig.6-1 (b) because of the smudgy nature of the fingerprint. Fig.6-1 (c) shows a fingerprint in which the contrast is extremely low because of dryness.

The quality of a region in a fingerprint image is a measure of the reliability and accuracy of the features extracted from that region. A fingerprint image quality evaluation algorithm can be used to form an image quality map and assign reliability indices to the local features (minutiae) being extracted in the region. This technique can be employed to decide whether a fingerprint can be used for the identification or

verification purposes or not. Moreover it can also improve the matching performance of the fingerprint identification system by producing a significant decrease in the equal error rate of the verification system.

Fig. 6-1 Fingerprint Images of Different Qualities

6.2 Literature Survey

Different algorithms are available in the literature for fingerprint image quality evaluation. Ratha et al. [RCJ95] labeled a fingerprint image block as foreground or background according to the variance of the gray levels in the direction orthogonal to the ridge orientations. This method also provides quality indices for the different blocks of the fingerprint image. It behaves inadequately to low quality and dry fingerprint images in which the variance along the ridge direction can be quite high because of the presence of ridge breaks. In [PHR+02] a method quantifying fingerprint image quality based on the directionality of the fingerprint texture has been presented. It also discusses some factors during fingerprint acquisition that lead to fingerprint images of low quality. Ratha et al. [NRB99] provide a subjective analysis of fingerprint image quality, which is helpful, in developing a general understanding of various factors affecting image quality. In [LJY02] image quality is evaluated by the verification of the striped pattern using the ratio of eigenvalues obtained from the correlation matrix of the gray level gradients of the image blocks. In [TWW04], a comprehensive discussion of fingerprint image quality estimation is provided. The authors in [MK] define fingerprint quality as a predictor of the matcher performance before the application of a matching algorithm. In [HWJ98], quality evaluation is carried out by using measures of the block contrast. A 1-Nearest Neighbor classifier is used to classify each $w \times w$-sized block in accordance with its

quality. In [NB92] a generic image quality assessment algorithm is presented which utilizes the information in the power spectrum of an image.

6.3 A New Approach to Quality Evaluation

All the algorithms mentioned above evaluate the quality of a fingerprint image without considering the recovery effect of the enhancement algorithm.

The low quality regions may or may not be used in the extraction of the recognition features depending upon the recognition method and the extent to which the fingerprint image enhancement algorithm is able to recover the information from the low quality regions. The proposed fingerprint segmentation and quality evaluation algorithms, unlike most of the techniques mentioned above, utilize the information (features) extracted from the original image as well as from the enhanced version of the fingerprint image. It ensures that the capabilities of the enhancement algorithm are also accounted for in the segmentation and quality evaluation processes.

We use fingerprint image quality evaluation to determine the good quality or recoverable regions (where recognition features can be reliably extracted after enhancement) and the unrecoverable regions. High proportion of the irrecoverable regions in an image can be used to reject a fingerprint image in order to prevent false identification of a person. Our method takes into account the capabilities of the enhancement and recognition feature extraction algorithms before deciding the quality of the fingerprint image.

6.3.1 Features Used

The various features available for fingerprint image quality evaluation include:

a. Block Intensity Mean
b. Block Intensity Variance
c. Block Gradient Coherence
d. Energy response of the filter used for fingerprint image enhancement

The block mean and block standard deviation (or variance) of both the original and enhanced images do not give a fine discrimination between the low and high quality regions of a fingerprint image. Therefore, they have not been used in fingerprint image quality evaluation. It was found experimentally that the reduced

feature set comprising of energy map and the coherences of both the original and enhanced images gives the best quality evaluation results.

The suitability of these features for use in fingerprint image quality evaluation is presented henceforth.

6.3.1.1 Gradient Coherence

The coherence of a region of a fingerprint image gives a measure how well the gradients of the fingerprint ridges are pointing in the same direction. Fingerprint images are characterized by the existence of a striped and oriented ridge and valley structure. The orientations of the ridges in a fingerprint image change slowly because of which the coherence of a fingerprint image foreground is comparatively higher than that in the background. In the presence of noise the orientation field of the fingerprint image is distorted which causes a decrease in the fingerprint image block gradient coherence. Therefore this feature can be used as means of quantizing the quality of a fingerprint image.

Mathematically Coherence is given by,

$$C_b = \frac{\left|\sum_W (G_{s,x}, G_{s,y})\right|}{\sum_W \left|(G_{s,x}, G_{s,y})\right|} = \frac{\sqrt{(G_{xx} - G_{yy})^2 + 4G_{xy}^2}}{G_{xx} + G_{yy}} \qquad (6\text{-}1)$$

Where $(G_{s,x}, G_{s,y})$ is the squared gradient, $G_{xx} = \sum_W G_x^2$, $G_{yy} = \sum_W G_y^2$, $G_{xy} = \sum_W G_x G_y$ and (G_x, G_y) are the local gradients along X and Y-axes evaluated using the sobel masks. The gradient coherences of both the original and the enhanced image are used in the formation of the feature set.

The block coherence of the original image

The coherence of the original image is expressed as a vector of the colomized block image coherences:

$$C_o = \left[C_o^{b_1} \quad C_o^{b_2} \quad \ldots \quad C_o^{b_n} \right] \qquad (6\text{-}2)$$

where $C_o^{b_i}$ is the coherence of the ith block of the original image. Fig.6-3a shows the coherence image for a fingerprint. Fig.6-2 b & c show the distributions of the block coherence for regions of different quality drawn manually at random from the FVC 2000 [MMC+00] DB2 database. It clearly shows that the block coherence in the good quality region of the fingerprint image is higher than that in a low quality region.

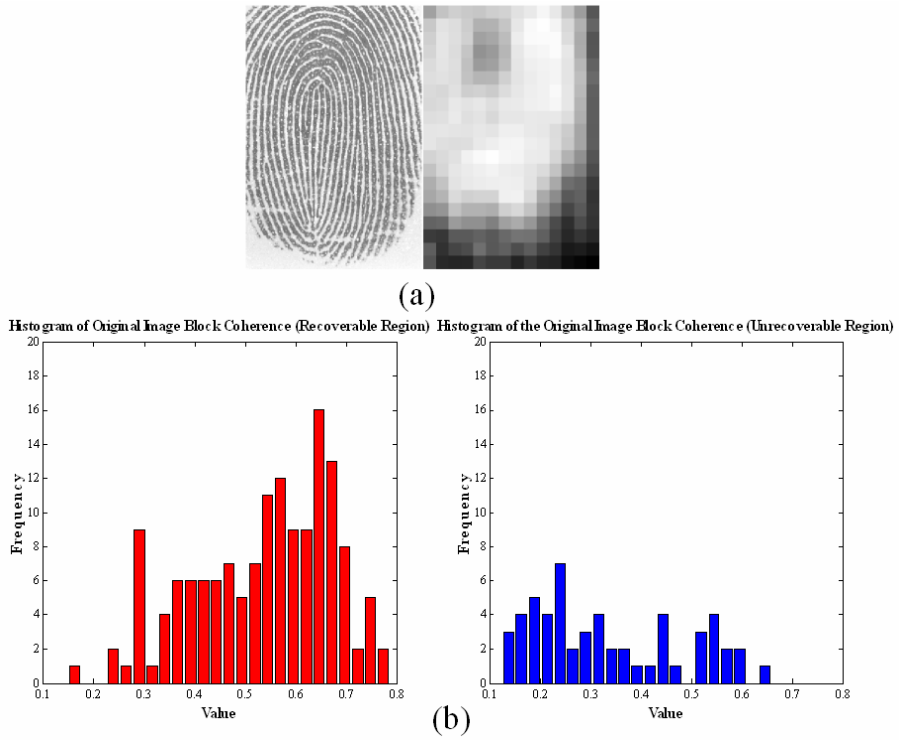

Fig. 6-2 (a) Block Coherence of an Original Image, (b) The histograms for block coherences of the recoverable (good quality) and the unrecoverable (bad quality) regions of original fingerprint images

The block coherence of the Enhanced image

A fingerprint image enhancement algorithm increases the discrimination between the ridges and the valleys and the continuity along a ridge. Therefore the net result of an enhancement algorithm is the increase in the fingerprint image block coherence. Here, we use the block coherence of the enhanced version of the fingerprint image in order to incorporate the recovery or the improvement in quality made by the enhancement algorithm. The coherence of the enhanced image is expressed as a vector of the colomized block image coherences:

$$C_e = \left[C_e^{b_1} \quad C_e^{b_2} \quad \ldots \quad C_e^{b_n} \right] \tag{6-3}$$

where $C_e^{b_i}$ is the coherence of the ith block of the original image. Fig.6-3a shows the coherence of a fingerprint image after enhancement. Fig.6-3 b & c very clearly demonstrate that the coherence in the good quality regions of the enhanced fingerprint image is higher than that in the low quality regions.

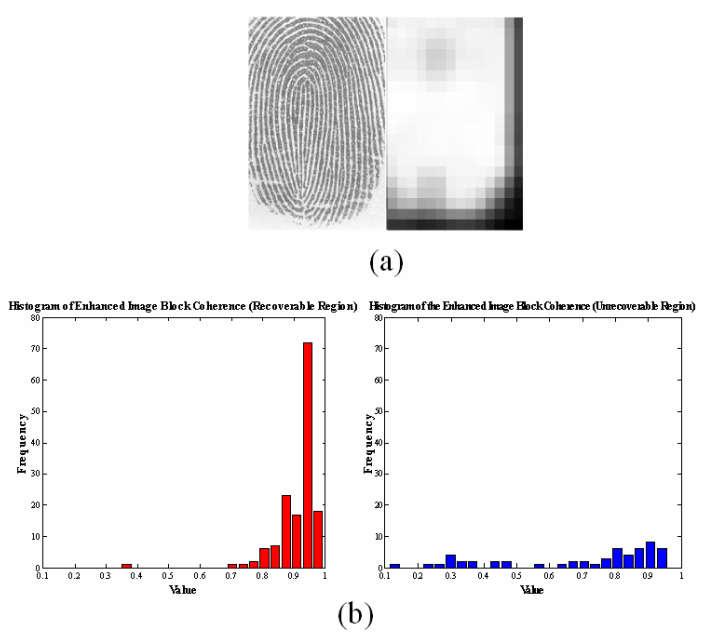

Fig. 6-3 (a) The block gradient coherence of an enhanced fingerprint image, (b) The histograms for block coherences of the recoverable (good quality) and the unrecoverable (bad quality) regions of enhanced fingerprint images

6.3.1.2 Enhancement Filter Energy Response

Some enhancement methods use the ridge frequency and ridge orientation information extracted from a fingerprint image to create contextual filters, which are convolved with the fingerprint image. Examples of such algorithms include [HWJ98], [JHP+97], [YLJ+03] and [CWG04]. These algorithms utilize the orientation and ridge frequency information in the striped nature of the fingerprint image. The response of the contextual filters used in these algorithms exhibit a high energy in the presence of a striped and oriented texture, which lies primarily in the good quality regions of the fingerprint image foreground. Therefore the energy of the filter responses of the features can be used for fingerprint image quality evaluation. We have used the approach given in [CWG04] which uses a Fourier domain based block-wise contextual filter approach for enhancing fingerprint images. Here, the energy map is defined as:

$$E_b = \sum_{u,v \in b} \left| F_b(u,v) \right|^2 \qquad (6\text{-}4)$$

where $F_b(u,v)$ is the Fourier spectrum of the fingerprint image block 'b'.

Fig.6- 4a shows the energy map of a fingerprint image. The energy of the filter response is expressed as a vector of the colomized block image energies:

$$E = \left[E^{b_1} \quad E^{b_2} \quad \ldots \quad E^{b_n} \right] \tag{6-5}$$

where E^{b_i} is the energy of the i th block of the original image. The energy of the response of the enhancement filter is higher in the good quality regions of the fingerprint image in comparison to that in the low quality areas as shown in Fig.6- 4 b.

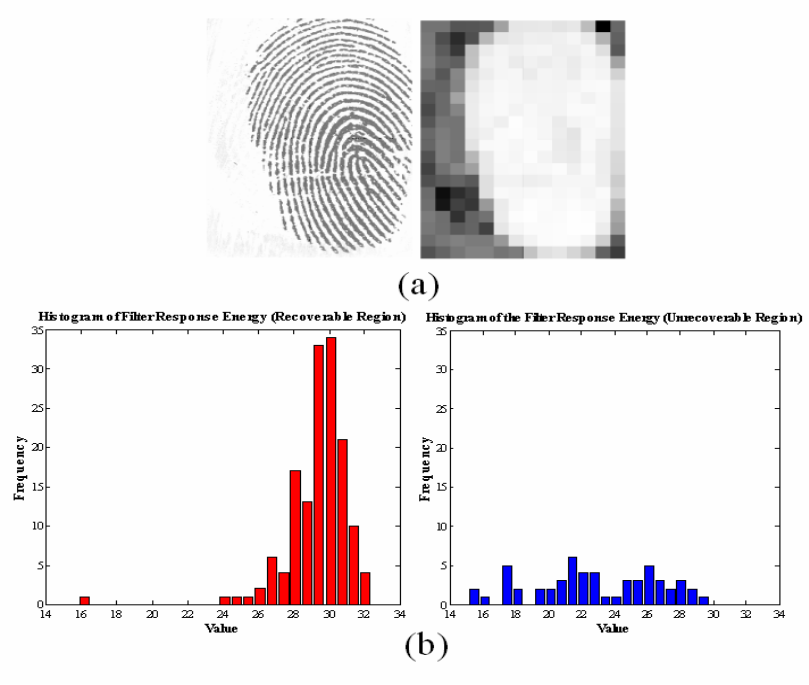

Fig. 6-4 Energy map for a fingerprint image, (b) The histograms for filter response energies of the recoverable (good quality) and the unrecoverable (bad quality) regions of fingerprint images

For fingerprint image quality evaluation, the i th block (i=1...n) of an image 'j' is described by the reduced feature vector

$$\Psi_i = \left[(C_o^{b_i}) \quad (C_e^{b_i}) \quad (E^{b_i}) \right]^T \tag{6-6}$$

6.3.2 Classifier used for Fingerprint image Quality Evaluation

Multilayer Feed forward back propagation Neural Networks [Fau94] are used for classification in fingerprint image quality evaluation. A back propagation network

operates by changing the weights at various layers using the error information backpropagated from the next layers. With the use of multiple layers the backpropagation neural network is capable of handling data that is not linearly separable. Fig.6-5 shows the general architecture of the backpropagation neural network.

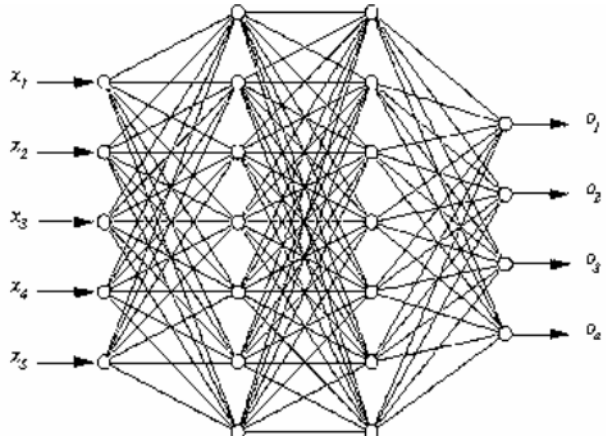

Fig. 6-5 A Backpropagation Neural Network with 2 Hidden Layers

6.3.3 Proposed Methodology

The fingerprint image quality evaluation algorithm is based on the use of multiplayer feed-forward back-propagation Neural Network. During the training phase about 200 blocks of various fingerprint images (chosen at random from the database) are classified manually as recoverable or irrecoverable blocks after considering the quality of the thinned images after enhancement (minutiae are considered to be the recognition features) thus forming the target vector as:

$$T_i = \begin{cases} -1 & \text{if } ith \text{ block belongs to unrecoverable region} \\ 1 & \text{if } ith \text{ block belongs to a recoverable region} \end{cases} \quad (6\text{-}7)$$

The features (mentioned earlier) are then extracted from these blocks and are normalized to a 0-1 range. The training set is described by:

$$X_{Train} = \begin{bmatrix} \Phi_1 & \Phi_2 & \cdots & \Phi_K \end{bmatrix} \quad (6\text{-}8)$$

The neural network consists of a single hidden layer with 4 neurons and a single output layer neuron. Bipolar sigmoid function is used as the activation function for all the layers. The output of the NN is taken to be bipolar since this provides much better quality resolution. The block diagram for the training phase is shown in Fig.6-6.

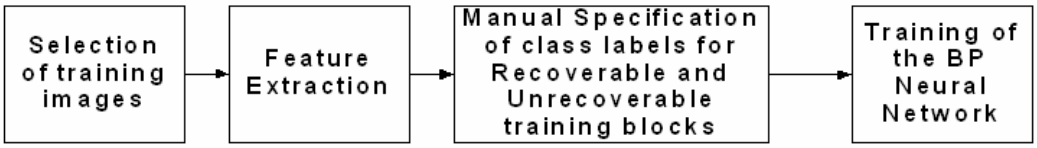

Fig. 6-6 Different Steps in the Training Phase of Fingerprint Image Quality Evaluation

During the application phase, the given segmented fingerprint image is divided into blocks and for each block the quality evaluation features are extracted. These features after normalization are given to the NN as input, which outputs a quality score. The quality scores of all the blocks in the image form the quality mask, which tells us about the reliability of feature extraction and matching for the different regions of the fingerprint image. If the number of blocks below a certain quality threshold is less then a specific threshold the image is rejected. Fig.6-7 shows the block diagram of the application process for quality evaluation.

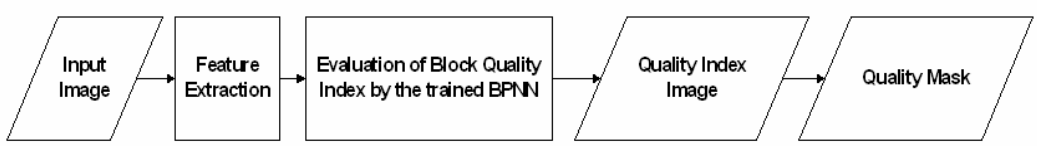

Fig. 6-7 Fingerprint segmentation by using Fisher Discriminant Analysis and LVQ NN

6.4 Results

The usefulness of the quality evaluation algorithm is illustrated by the improvement in the FAR (false accept rate) vs. FRR (false reject rate) curve for a particular fingerprint recognition system.

Fig.6-8 shows the quality maps (quality index images) of different fingerprints. Fig.6-9 shows the FAR (false accept rate) vs. FRR (false reject rate) curve for a particular fingerprint recognition system of a particular fingerprint recognition system before and after the application of our quality evaluation algorithm. This clearly demonstrates the usefulness of the quality evaluation algorithm. (Reject rate is taken to be 0% in the both the cases.)

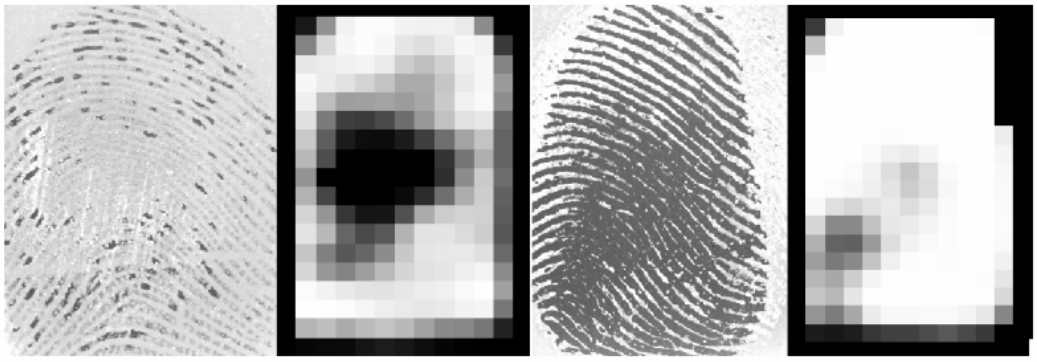

Fig. 6-8 The quality maps for different images

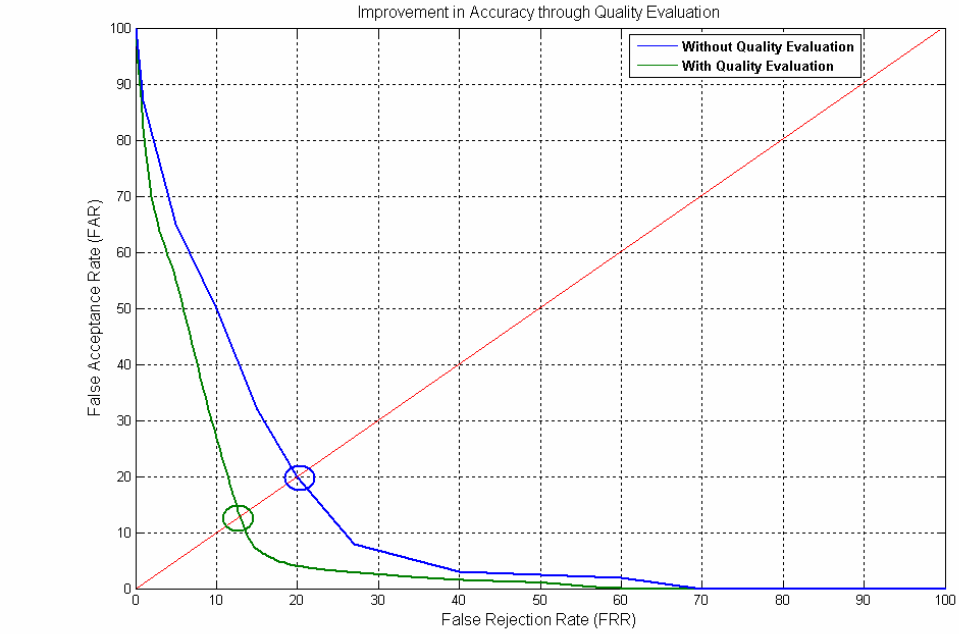

Fig. 6-9 Improvement in System Performance using Quality Evaluation

Summary

Fingerprint Image Quality Evaluation is used for quantizing the quality of a given fingerprint image which can be used to determine the reliability of the features extracted in various regions of the fingerprint, thus improving the matching performance. We have developed a Multilayer Perceptron based approach to the classification of the regions of fingerprint images on the basis of their quality by using features such as the block coherences of the original and enhanced images and the

enhancement filter response. The results obtained by the use of the quality evaluation algorithm show significant improvement in the matching performance of a feature-matching algorithm, which convincingly demonstrates the effectiveness of the approach.

7 Fingerprint Matching

In this chapter we describe various approaches to fingerprint matching, which is an integral part of an automatic fingerprint identification and verification system. We discuss the objectives and problems in matching; describe a literature survey of the existing matching techniques and then present the techniques that have been implemented during the course of this project along with their results.

7.1 Objectives of Fingerprint Matching

The objective of fingerprint matching algorithm is to compare two given fingerprints and return a degree of similarity (matching score) or a binary decision (mated/non-mated). Most of the fingerprints matching algorithms operate upon an intermediate representation of the fingerprint, which is obtained through feature extraction. The fingerprint image acquired during enrollment is called as the template (T) and the representation of the fingerprint to be matched is called as input (I). A matching algorithm computes a matching score between these fingerprint representations.

The fingerprint feature extraction and matching algorithms are usually quite similar for both fingerprint verification and identification problems. This is because the fingerprint identification problem can be implemented as a sequential execution of N one-to-one matches (verifications) between pairs of fingerprints. Fingerprint classification and indexing techniques are usually exploited to speed up the search in fingerprint identification.

7.2 Difficulties in Fingerprint Matching

The matching of fingerprint patterns is an extremely difficult problem [MMJ+03], mainly due to the large variability in different impressions of the same finger, which implies a very high intra-class variation. The main factors responsible for these variations include:

a. Displacement

The same finger can be placed at different locations on a fingerprint sensor during different acquisitions resulting in a (global) translation of the fingerprint area. Even minute displacements which might be imperceptible to the user may result in a

significant translation of the fingerprint image depending upon the resolution of the scanning process. Fig. 7-1 shows an example in which fingerprints from the same finger show considerable displacement.

Fig. 7-1 Effects of Displacement

b. Rotation

The same finger can be rotated at different angles with respect to the sensor surface during different acquisitions. Significant rotation of the fingerprint may be encountered in practical fingerprint acquisitions. Fig. 7-2 shows an example in which fingerprints from the same finger show considerable rotation with respect to each other.

Fig. 7-2 Effects of Rotation

c. Partial Overlap

Fingerprint displacement and rotation can cause a part of the fingerprint area to fall outside the sensor's capture area, resulting in a small overlap between the foregrounds of the template and input fingerprints. This problem is particularly serious for small sized fingerprint scanners. Fig. 7-3 shows an example in which fingerprints from the same finger shows only a partial overlap.

Fig. 7-3 Partial Overlap (indicated by red squares) between two fingerprints

d. Nonlinear Distortion

Fingerprint acquisition maps the 3D shape of fingerprints onto a 2D surface of the sensor thus producing nonlinear distortions in successive acquisitions of the same finger due to skin plasticity. The effects of non-linear distortion can be reduced by a correct fingerprint placement of the finger by the subject. A fingerprint placement is correct when the user:

- Approaches the finger to the sensor through a movement that is orthogonal to the sensor surface
- Once the finger touches the sensor surface, the user does not apply traction or torsion.

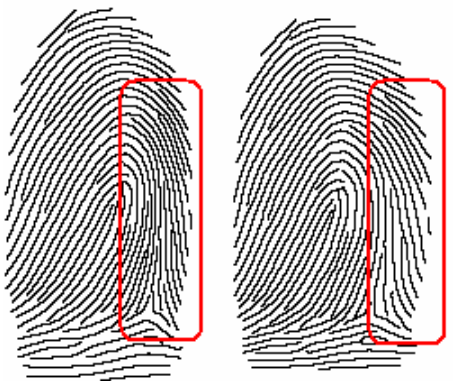

Fig. 7-4 Effects of Distortion: On the Left is a fingerprint images with distorted ridge patterns, on the right is its undistorted form

However, due to skin plasticity, the components of the force those are non-orthogonal to the sensor surface produce non-linear distortions (compression or stretching) in the acquired fingerprints which results in the inability to match fingerprints as rigid patterns. Non-contact 3D scanners are available in which the

effect of non-linear distortions is virtually removed, however, the cost of these scanners makes their use impractical in most commercial applications. Fig. 7-4 shows an example in which fingerprints from the same finger shows considerable difference because of these nonlinear distortions.

e. Pressure and Skin Conditions

The ridge structure of the fingerprint is not in uniform contact with the sensor surface because of non-uniform pressure on the finger, dryness of the skin, skin disease, sweat, dirt, grease, and moisture in the air etc. As a consequence fingerprint images are very noisy and the magnitude of the noise varies considerably among different acquisitions of the same finger. Fig. 7-5 shows an example in which fingerprints from the same finger shows considerable difference because of variations in skin contact.

Fig. 7-5 Effect of Pressure and Skin Conditions: A Wet Fingerprint (Left) and a Dry Fingerprint (Right)

f. Noise

Noise is mainly introduces because of the acquisition process involved e.g. the presence of residues left over the sensor surface from previous fingerprint captures. This problem is reduced in capacitative and non-contact 3D scanners. Fig. 7-6 shows an example, which illustrates the effect of noise in fingerprint acquisition.

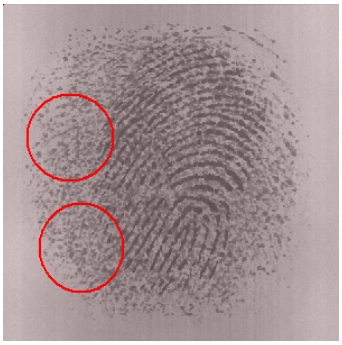

Fig. 7-6 Background Noise During Fingerprint Capture

g. Feature Extraction Errors

The feature extraction algorithms are imperfect and often introduce measurement errors: Errors can be made during any of the feature extraction stages e.g. ridge orientation and frequency estimation, singular point detection, segmentation etc. Fingerprint image enhancement may also introduce consistent biases that perturb the location and orientation of the reported minutiae from their gray scale counterparts. Another issues is the introduction of spurious minutiae in low quality fingerprints.

Fingerprints also exhibit low inter class variation as shown in Fig. 7-7.

Fig. 7-7 Class Variation in Fingerprints: (Left) These two fingerprints belong to two different fingers but look very similar (Low Inter Class Variation), (Right) The Fingerprints belong to the same finger but appear similar

Fingerprint matching remains to be a pattern matching problem to date due to the difficulty in matching low quality and partial fingerprints which are the major source of error or decrease in performance of an automatic fingerprint identification and verification system. An automated quality evaluation algorithm (such as the one proposed in chapter X.) can be used for improving the matching process accuracy.

7.3 A Survey of Fingerprint Matching Approaches

A large number of approaches to fingerprint matching have been proposed in literature, which can be broadly classified [MMJ+03] as:

1. Correlation Based Matching

These methods compute the matching score by using correlation between two fingerprints from different alignments.

2. Minutiae Based Matching

Minutiae based matching is the most popular and widely used technique for fingerprint matching and is the basis of fingerprint comparison by human experts.

Minutiae-matching is taken up as a point pattern-matching problem after the minutiae pairs obtained from the template and the input fingerprints have been aligned, to generate a matching score.

3. Ridge Feature Based Matching

These approaches make use of global features such as ridge orientation, frequency, shape and fingerprint textural information which can be extracted more reliably from low quality fingerprints for which minutiae extraction is not reliable. The distinctiveness of such global features is generally lower in comparison to minutiae based features.

7.3.1 Correlation based Techniques

Correlation based techniques are based on the computation of cross-correlation between two images which is a measure of image similarity. When the cross correlation or simply the correlation between two images is maximized then the diversity or difference between two images is minimized. The cross correlation between two fingerprints T and I is computed as follows:

$$CC(T,I) = T^T I \tag{7-1}$$

However because of displacement and rotation that unavoidably characterizes two impressions of a given finger, it is practically impossible to compute the similarity between two fingerprint images by using equation (1). Let $I^{(\Delta x,\Delta y,\theta)}$ represent a rotation of the input image I by an angle θ around the origin, which is usually taken to be the image center, and shifted by Δx, Δy pixels in directions X and Y, respectively; then the similarity between the two fingerprint images is measured as:

$$S(T,I) = \max_{\Delta x,\Delta y,\theta}[CC(T,I^{(\Delta x,\Delta y,\theta)})] \tag{7-2}$$

The direct application of this equation may not yield acceptable results because of the following problems:

- Non Linear Distortion makes impressions of the same finger significantly different in terms of global structure and the integration of the effects non linear deformations into the image space makes the use of correlation for matching prohibitive.

- Skin conditions and finger pressure cause significant variations in image brightness, contrast and ridge thickness due to which correlation based matching exhibits low performance in fingerprint matching.
- Correlation based techniques are computationally very expensive.

Various improvements that address these problems have been proposed in literature such as [CMC96, HTM+02, YIY90, KV00] etc.

A special issue in correlation-based matching [MJT+75, Gry95, Gry96, WGC00] is the use of optical correlation for fingerprint matching, which compute the similarity between two images by using joint transform correlator which operates on the Fourier transform of the two images obtained using optical lenses. The main incentive in the use of optical correlation based matchers is the high matching speed, which can be obtained by optical techniques. However because of the high cost of hardware and optical components, susceptibility of optical matchers to rotation and distortion variations, optical fingerprint matching technologies has not reached satisfactory maturity yet.

In this project we have not implemented any correlation based technique as current correlation based strategies do not provide a very high matching performance both in terms of accuracy and execution speed.

7.3.2 Ridge Feature based Matching

Minutiae features are difficult to extract in low quality fingerprint images. Moreover the extraction of fingerprints is computationally expensive. These problems have spurred researchers to seek alternatives to minutiae based fingerprint matching which are more robust and utilize features which can be extracted relatively easily with lesser computational resources being used. The combination of these features with minutiae can result in a significant improvement in the matching accuracy of a fingerprint based person authentication system. Examples of such features [MMJ+03] include:

a. Size and Shape of the External Fingerprint Silhouette

These features are very unstable as they are highly dependent upon the contact area of the finger with the sensor surface.

b. Number, type and Position of Singularities

These features are mainly used for classification that can be used in matching as well, since fingerprints belonging to different classes cannot belong to the same finger.

c. Spatial Relationship and geometrical attributes of the ridge lines

The Spatial Relationship and geometrical attributes of the ridge lines are captured by the use of tree grammars [MF86] and by defining the ridges and arches to be graph nodes dependent upon ridge adjacency and visibility and then using a graph matching based approach to compute the matching score between given fingerprints [IZ86].

d. Shape Features

These features capture the general shape of a fingerprint image, which can be encoded as a 1D shape signature used to augment minutiae based fingerprint-matching approaches [CK02].

e. Global and Local Texture Features

Global and local texture based features are important alternatives to minutiae. Textures are defined by spatial repetition of basic elements, and are characterized by properties such as scale, orientation, frequency, symmetry, isotropy and so on. Fingerprint ridgelines are mainly described by smooth ridge orientation and frequency except at singular points. The continuity of fingerprint ridges is also broken at the minutiae of the fingerprint. Various approaches exist in literature for global and local texture feature based matching for fingerprint images exist in literature and texture based fingerprint recognition is an active area of research. The texture information can be captured by the use of a variety of approaches such as the use of Fourier transform [CB93], Gabor Filters [JHP00, Ham99], Wavelet Transform [TKS01] etc. In this thesis we investigate two texture-based approaches, which are described in detail in a later Section.

f. Sweat Pores

Sweat pores are highly discriminative [SA94] but their capture requires the use of high resolution scanners whose cost is a major set back in the commercial use of recognition systems which use sweat pores for identification or verification.

g. Fractal Features

Fractal based features are also being investigated for fingerprint matching [Pol96].

7.3.3 Minutiae based Matching

Minutiae-based-matching is the most well known technique for fingerprint matching; thanks to its strict analogy with the way forensic experts match fingerprints and its acceptance as a proof of identity in the courts of law in almost all countries.

Minutiae are generally characterized by their position, orientation and type. This enables use to express a fingerprint as a triplet $m = \{x, y, \theta\}$ that indicates the minutiae location x, y and the minutiae angle θ. Moreover features such as the quality indices of the region in which a minutiae lie can also be used which can be used to mark the reliability of the minutiae for use in fingerprint verification. Let T and I be the minutiae set representations of the template and input fingerprints containing m and n minutiae respectively, given as:

$$T = \{m_i\} = \{(x_i, y_i, \theta_i)\} \qquad i = 1...m$$
$$I = \{m_j'\} = \{(x_j', y_j', \theta_j')\} \qquad j = 1...n$$

Minutiae m_j' in I and a minutiae m_i in T are considered as matching (mated) if the spatial distance (sd) between the two minutiae sets is less than a certain threshold r_o and the direction difference (dd) between them is smaller than an angular tolerance θ_o as shown in Fig. 7-8:

$$sd(m_j', m_i) = \sqrt{(x_j' - x_i)^2 + (y_j' - y_i)^2} \le r_o \tag{7-3}$$

$$sd(m_j', m_i) = \min(|\theta_j' - \theta_i|, 360^\circ - |\theta_j' - \theta_i|) \le \theta_o \tag{7-4}$$

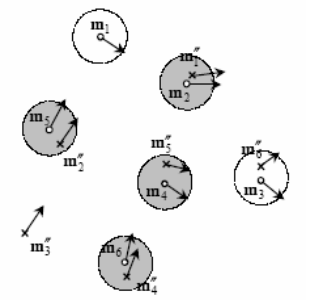

Fig. 7-8 Minutiae of I mapped into T coordinates for a given alignment. Minutiae of T are denoted by Os, whereas I minutiae are denoted by Xs. Note that I minutiae are erred to as m′′, because what is shown in the Figure is their mapping into T coordinates. Pairing is performed according to the minimum distance. The dashed circles indicate the maximum spatial distance. The gray circles denote successfully mated minutiae; minutia m_1 of T and minutia m''_3 of I have no mates, minutiae m_3 and m''_6 cannot be mated due to their large direction difference.

Tolerance boxes or hyper spheres characterize by r_o and θ_o are necessary to compensate for the unavoidable errors made by feature extraction algorithms and to account for the small plastic distortions that cause minutiae positions to change. The type of the minutiae (ridge ending or bifurcation) is usually not used as the type of the minutiae can be inverted during the enhancement step.

Registration of the minutiae sets is a mandatory step in minutiae based fingerprint matching systems. The alignment of two minutiae requires the displacement (in X and Y) and rotation (θ) in the placement of the finger to be recovered. Scale is considered when the resolution of the scanners used for fingerprint acquisition is different for the template and the input fingerprints.

A formulation of the minutiae based fingerprint-matching problem is described here [MMJ+03]. Let $map(.)$ be the function that maps a minutiae $m_j^{'}$ (from I) into $m_j^{''}$ according to a given geometrical transformation; for example by considering the displacement of $[\Delta x, \Delta y]$ and a counter-clockwise rotation θ around the origin. The origin in rotation is usually taken to be the minutiae centroid (the average point) or the core point. Here,

$$map_{\Delta x,\Delta y,\theta}(m_j^{'} = \{x_j^{'}, y_j^{'}\theta_j^{'}\}) = m_j^{''} = \left\{ x_j^{''}, y_j^{''}, \theta_j^{'} + \theta \right\} \tag{7-5}$$

where,

$$\begin{bmatrix} x_j^{''} \\ y_j^{''} \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} x_j^{'} \\ y_j^{'} \end{bmatrix} + \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} \tag{7-6}$$

Let $mm(.)$ be an indicator function that returns 1 when the two minutiae m_i and $m_j^{''}$ match according to equation (3) and (4).

$$mm\left(m_j^{''}, m_i \right) = \begin{cases} 1 & sd(m_j^{''}, m_i) < r_o \text{ and } dd(m_j^{''}, m_i) < \theta_o \\ 0 & Otherwise \end{cases} \tag{7-7}$$

Then the matching problem can be formulated to

$$\underset{\Delta x,\Delta y,\theta}{\text{maximize}} \sum_{i=1}^{m} mm(map_{\Delta x,\Delta y,\theta}(m_{P(i)}^{'}), m_i) \tag{7-8}$$

Where $P(i)$ is an unknown function that determines the pairing between I and T as:

1. $P(i) = j$ Indicates that the mate of the m_i in T is the minutiae $m_j^{'}$ in I. It does not necessarily mean that minutiae m_i and $m_j^{'}$ match in the sense of equation (3) and (4) but only that they are the most likely pair under the current transform.
2. $P(i) = null$ indicates that minutiae m_i in T has no mate in I.
3. A minutiae $m_j^{'}$ in I such that $\forall i = 1...m, P(i) \neq j$ has no mate in T
4. $\forall i = 1...m, k = 1...m, i \neq k \Rightarrow P(i) \neq P(k)$ or $P(i) = P(k) = null$ (This requires that each minutia in I is associated with a maximum of one minutia in T).

The solution to the minutiae-matching problem is trivial when the correct alignment is known. Here P is given as:

- $P(i) = j$ if $m_j^{''} = map_{\Delta x, \Delta y, \theta}(m_j^{'})$ is closest to m_i among the minutiae $\left\{ m_k^{''} = map_{\Delta x, \Delta y, \theta}(m_k^{''}) \mid k = 1...n,\ mm(m_k^{''}, m_i) = 1 \right\}$.
- $P(i) = null$ if $\forall k = 1..n,\ mm(map_{\Delta x, \Delta y, \theta}(m_k^{'}), m_i) = 0.$

To comply with constraint 4 above, each minutia $m_j^{''}$ already mated has to be marked, to avoid mating it again.

Fingerprint matching by the use of minutiae is a very challenging problem because the alignment parameters (transform for registration) are unknown along with the correspondence function P. Specialized point pattern matching approaches are used for matching fingerprints on the basis of minutiae. Some of these approaches include: [Rat96, UGS01, JRL+96].

7.4 Implemented Techniques

In this Section we presents the approaches to fingerprint matching that have been implemented in this project.

7.4.1 Ridge Feature Based Matching

Two ridge feature based fingerprint-matching techniques have been implemented in this project, which are described in detail henceforth:

7.4.1.1 Wedge-Ring Feature Extraction & Matching

We have implemented an approach to fingerprint matching which is based on the extraction of features vectors from the 2D FFT of the binarized image of a fingerprint using a wedge-ring feature extraction process [AAM97]. The various steps involved in this approach include:

i. Image Preprocessing

Image preprocessing involves contrast enhancement using histogram equalization or normalization.

ii. Image Segmentation

For feature extraction the fingerprint image foreground is segmented using morphological opening by approximating ridges as discs of radius equal to the average ridge width and subtracting the background region of the fingerprint.

iii. Weighted FFT Method for Fingerprint Enhancement

Here we utilize the weighted FFT enhancement method detailed in Section 5.4.1.

iv. Binarization

Binarization is performed on the enhanced image as an overlapped window operation by using local mean based thresholding.

v. Feature Extraction

The 2D FFT of the whole of the binarized image is then taken. This frequency domain image is then centralized so that the center of the image corresponds to lower frequencies with the frequencies increasing away from the center. Because of the symmetry of the FFT image only the upper half of the image is taken. This half is then subjected to wedge-ring classification. The wedge ring classification process can be viewed as a wedge and ring overlay (see Fig.7-9) that is placed on the image

Fig. 7-9 The Wedge-Ring Overlay

Feature extraction is then carried out by summing the spectral values in each ring and each wedge. This is done mathematically for a ring (i) as follows:

$$\lambda_i(r) = \sum_u \sum_v |F(u,v)| \tag{7-9}$$

Where $r \le \sqrt{u^2+v^2} \le r + \Delta r$ and $0 \le r \le N/2$. N is the dimension of the image. Δr is the width of each circle. For a wedge (i) we have:

$$\beta_i(\phi) = \sum_u \sum_v |F(u,v)| \tag{7-10}$$

Where $\phi \le \arctan(u/v) \le \phi + \Delta\phi$ and $0 \le \phi + \Delta\phi . \Delta\phi$ is the angular width of the wedge. The λ s and β s are then grouped to form the feature vectors.

$$\beta = [\beta_1, \beta_2, \beta_3, ..., \beta_L]$$

$$\lambda = [\lambda_1, \lambda_2, \lambda_3, ..., \lambda_K]$$

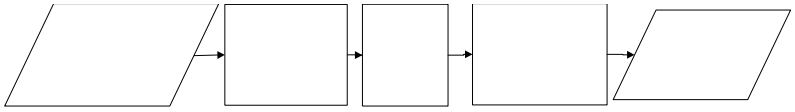

Fig. 7-10 Steps involved in Feature Extraction

vi. Template Generation

The feature vectors for M images of a single fingerprint are averaged to form the template for that fingerprint as follows.

$$\beta_{temp_j} = \frac{1}{M} \sum_{i=1}^{M} \beta_{j_i} \tag{7-11}$$

$$\lambda_{temp_j} = \frac{1}{M} \sum_{i=1}^{M} \lambda_{j_i} \tag{7-12}$$

Where β_{temp_j} and λ_{temp_j} are the j[th] element of the template vector of β and λ respectively. The feature vectors are obtained as:

$$\beta_{temp} = [\beta_{temp_1}, \beta_{temp_2}, \beta_{temp_3}, ..., \beta_{temp_L}]$$

$$\lambda_{temp} = [\lambda_{temp_1}, \lambda_{temp_2}, \lambda_{temp_3}, ..., \lambda_{temp_K}]$$

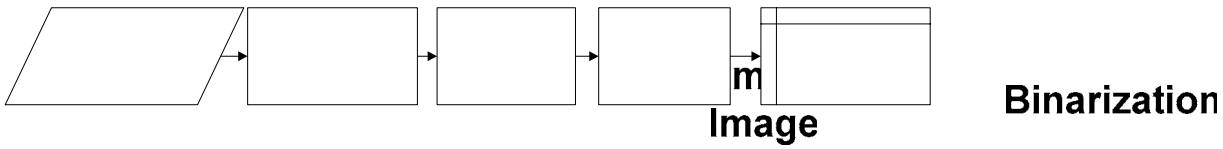

Fig. 7-11 Steps involved in Template Development

vii. Feature Matching and Verification

For verification of a given fingerprint image, the features acquired from the given fingerprint are compared with the template obtained for the person to be verified. The lambda values are compared using the following gaussian function

$$\lambda_s = e^{-(\lambda_{err}{}^T * \lambda_v^{-1} * \lambda_{err})/4} \tag{7-13}$$

Where λ_{err} is given by the absolute difference between λ_{temp} and λ vectors:

$$\lambda_{err} = \left|\lambda - \lambda_{temp}\right| \tag{7-14}$$

While λ_v is the vector obtained by evaluating the variances of each the λ s for the M enrolled images.

The beta values are compared using the following Gaussian function:

$$\beta_s = e^{-(\beta_{err}{}^T * \beta_v^{-1} * \beta_{err})/32} \tag{7-15}$$

Where $\beta_{err} = \left|\beta - \beta_{temp}\right|$ and β_v is the vector obtained by evaluating the variances of each the β s for the M enrolled images. λ_s and β_s represent the similarities between the lambda and beta vectors respectively of the given fingerprint with those of the template for the specified person. A person is verified if the values of both λ_s and β_s obtained for the provided image are greater or equal to 0.5. Changing this criterion changes the security level of the verification system. A low security level increases the FAR while decreasing the FRR whereas a higher security level increases the FRR with reduction in FAR.

7.4.1.2 Results and Discussion

For experimentation the FVC 2000 database was used. This database consists of 300x300 images. The power factor was taken to be 1.7 in the enhancement process. The results of each of the steps are shown in the Fig.7-12. The 2D FFT image obtained was then subjected to wedge-ring classification process. The image was overlaid with 6 wedges and 32 circles and the sum of spectral values was calculated for each of the circles and wedges thus forming λ and β feature vectors. Out of the 8 images of each finger in the database, 4 were used to develop the template for that person. The template for each person took about 400 bytes of storage, which is quite small. The FAR of the system is found to be 1% whereas the FRR of the system is

found to be about 5.5%. These ratios, though statistically insignificant because of the small size of the database, do illustrate the effectiveness of the approach. Using more images during the enrollment process can decrease these ratios. Image quality also affects the FAR and FRR of the system.

A limitation of this approach is that it does not show good results for fingerprints that contain a large number of cuts. This limitation can be removed by approximating the ridges through interpolation based on ridge orientation at the ends of the discontinuities.

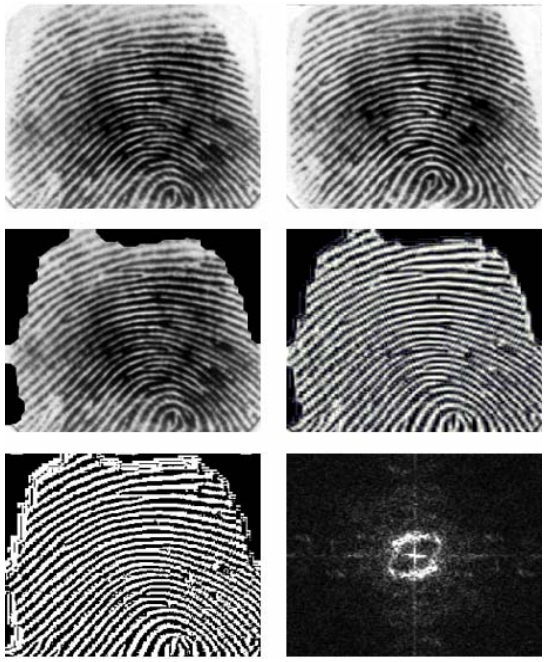

Fig. 7-12 Different Steps involved in Feature Extraction: Original Fingerprint, Histogram Equalization, Segmentation, Enhancement, Binarization and 2D FFT

7.4.1.3 Wavelet Transform Based Matching

The Fourier transform is a tool widely used for many scientific purposes, but it is well suited only to the study of stationary signals where all frequencies have an infinite coherence time. The Fourier analysis brings only global information, which is not sufficient to detect compact patterns. Gabor introduced a local Fourier analysis, taking into account a sliding window, leading to a time frequency-analysis. This method is only applicable to situations where the coherence time is independent of the frequency. This is the case for instance for singing signals which have their coherence time determined by the geometry of the oral cavity. Morlet introduced the Wavelet Transform in order to have a coherence time proportional to the period thus providing

optimum time frequency localization for the input signal. A detailed discussion of the wavelet transform is given in [Chui (1992)]. Fingerprints can be treated as non-stationary signals as the ridge frequency varies within the fingerprint image and is different at different spatial locations. Theore the use of the wavelet transform for the analysis of fingerprint patterns can yield better results by capturing more information about the patterns, which can be used for fingerprint matching. Various approaches to fingerprint feature extraction and matching using the wavelet transform are available in the literature [TIR01, LN99].

We have implemented a slight variation to the wavelet transform based approach to fingerprint identification proposed in [TIR01]. The various steps involved in this approach include:

a. Preprocessing

In this stage, the contrast of the input fingerprint images is enhanced using adaptive histogram equalization. If the quality of the input fingerprint is low then fingerprint image enhancement methods such as the one described in Section 5.4.4 can be used for better performance.

b. Core Point Extraction

In this approach, the core point is used as a reference in the extraction of a central subimage, which is a $N \times M$ sized rectangular region of the fingerprint centered at the core point. The core point is detected by using the approach given in Section 3.7. that takes as its input a segmented version of the fingerprint. Fingerprint segmentation was carried out by the use of a simple block-wise mean based approach.

c. Feature Extraction

In this step wavelet domain features are extracted from the central subimage by using the discrete wavelet transform decomposition. The 2D wavelet decomposition on J octaves of a discrete subimage $I[m,n]$ represents the image in terms of $3J+1$ subimages as:

$$\left[a_J, \left\{ d_j^1, d_j^2, d_j^3 \right\}_{j=1\ldots J} \right] \tag{7-16}$$

Where a_J is a low-resolution approximation of the original image and d_j^k are the detail subimages at different scales (2^j) and orientations (k). Wavelet coefficients of large magnitudes in d_j^1, d_j^2 and d_j^3 correspond respectively to vertical high frequencies (horizontal edges), horizontal high frequencies (vertical edges) and high

frequencies in both direction (diagonal edges). The scales capture the frequency specific information of the input image. A high coefficient at a particular frequency (scale) indicates the presence of an oscillatory pattern of that frequency in the input image. As fingerprints possess a frequency specific and an orientation specific behavior theore the energy distribution of the wavelet coefficients over different scales and orientations is a quite informative feature for fingerprint analysis. It is observed that these frequencies and orientations possess a degree of uniqueness among different fingers thus enabling the use of the wavelet transform coefficient energies in fingerprint based person identification. This energy is captured by the standard deviation of the wavelet subimages. A wavelet decomposition on J octaves of the discrete image is used to compute a feature vector of length $3J$ as:

$$\left[\left\{\sigma_j^1 \quad \sigma_j^2 \quad \sigma_j^3\right\}_{j=1\ldots J}\right] \tag{7-17}$$

Here σ_j^k is the standard deviation of the wavelet coefficients in the subimage d_j^k. In order to preserve the information concerning the spatial location of different details these wavelet features are extracted from small non-overlapping blocks of size $W \times W$ in the central subimage such that the total number of windows is $\frac{NM}{W^2}$. The global feature vector that includes the wavelet features extracted from all $W \times W$ blocks of the central subimage is given by:

$$\left[\left\{\left\{\sigma_j^1 \quad \sigma_j^2 \quad \sigma_j^3\right\}_{j=1\ldots J}^{w}\right\}_{w=1\ldots NM/W^2}\right] \tag{7-18}$$

Thus the length of the feature vector for a single fingerprint image is $3J\frac{NM}{W^2}$.

d. Enrollment

A number of k subimages of each individual were enrolled after they have been registered on the basis of the first image in the enrollment set of an individual using the registration technique described next. A k-Nearest Neighbor (k-NN) classifier [FH04 (2004)] is then trained on the basis of these subimages.

e. Registration

In order to account for the effect of rotation, fingerprints are registered by the use of a coarse registration technique. This technique is based on the use of the orientation field of the central subimage. The input to this process is the central subimages of two

fingerprints, which are to be registered in terms of rotation. The orientations at different points on a circle of diameter W centered at the core are calculated by the orientation estimation approach described in Section 5.3.1 to form radial orientation vectors for both the fingerprints. The histograms of the difference between these radial orientation values are then plotted for the selection of the top five angles. The input fingerprint subimage is rotated with respect to the erence and a criterion function is applied to determine the accuracy of the registration process. A typical criterion function is the standard deviation of the orientation fields of the two fingerprints. The rotation giving the minimum standard deviation is then selected.

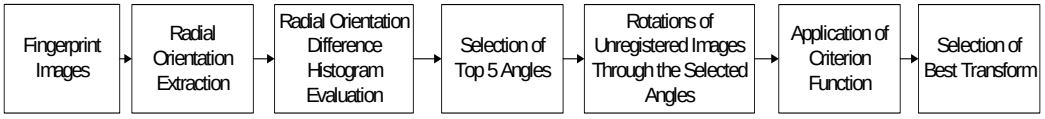

Fig. 7-13 Different Steps involved in Fingerprint Registration

The results of the registration step are shown in Fig. 7-14.

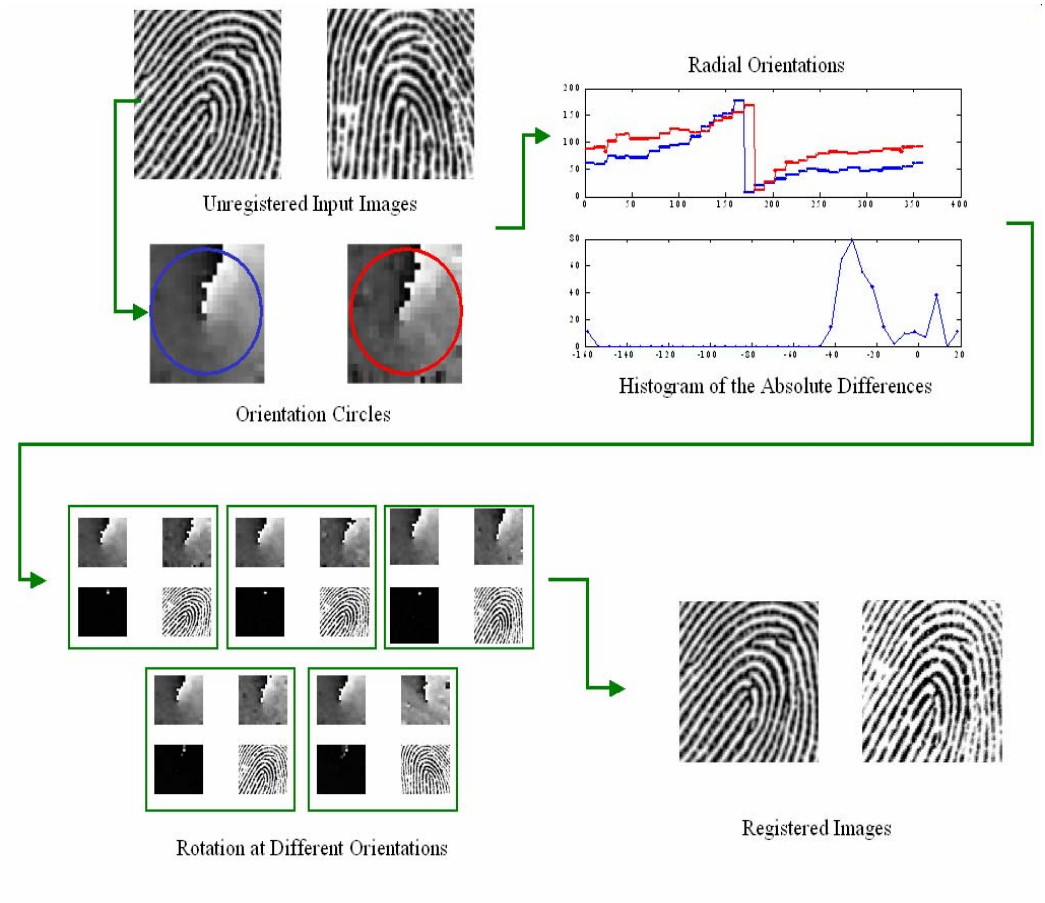

Fig. 7-14 Results of Different Steps involved in Registration

f. Matching

Matching is performed by using the trained k-NN classifier, which gives us the class label for a person when given the features from his fingerprints.

7.4.1.4 Results and Discussion

The various steps involved in the technique are shown pictorially in Fig. 7-15.

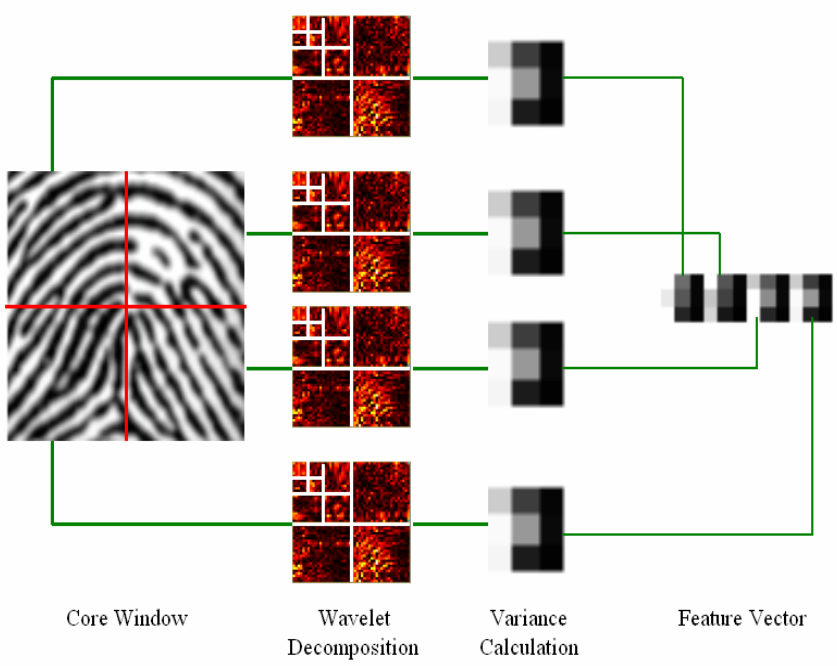

Fig. 7-15 Wavelet Domain Feature Extracion for Fingerprints

This system was tested with the University of Bologna database. The database used contains 21 subjects with 8 images per subject. Four of these images were used for training (enrollment) and the remaining for testing. The best results obtained exhibit a misclassification error of only 1.19%. In this case fingerprint enhancement and registration were not carried out as the input fingerprints are already of very good quality and are aligned in terms of rotation to a great extent. These results were obtained for $W = 48$, $N = M = 2W$ and a 1-NN classifier. The Daubechies wavelet filter with 10 vanishing moments was used for the decomposition. The increase in the number of subjects will further clutter the feature space thus making the classification error to increase.

This approach suffers from the following flaws:

a. The features extracted are not inherently rotation invariant. A rotation causes the energies of the wavelet transform coefficients to be changed which makes a priori registration mandatory.
b. The reliable detection of the core points can be a problem in low quality fingerprints, or arch like patterns.
c. This approach is suited for a closed-class operation. For a matching with a fingerprint of a finger that has not been enrolled a certain thresholding scheme will have to be adapted for the particular distance measure being used in the k-NN classifier.

This approach was tested with the FVC-2000 DB1a database aswell. This database contains images obtained from an optical fingerprint scanner and the quality of the fingerprints is much lower than the fingerprints in the University of Bologna Database. Moreover these fingerprints also exhibit considerable variations in terms of placement. 21 subjects were used (whose cores could be extracted visually) for testing purposes. The best results obtained for this dataset show an error of 16.67%. This error was reduced to about 9.5% by the use of fingerprint image enhancement and registration. The major contribution to the error, now, comes from the fingerprints in which a portion of the central subimage contains some background region. This error can further be reduced by the use of multiple classifiers for each window in the central subimage. The reliability of the decision being generated by a particular classifier depends upon the quality of the fingerprint in that region and the ratio of the background to foreground area in that region.

7.4.2 Minutiae Based Matching

In this thesis we have implemented two minutiae based approaches to fingerprint recognition, which are described henceforth.

7.4.2.1 Generalized Hough Transform Based Matching

The various steps involved in this technique [Rat96] are:

a. Image Preprocessing

The objective of fingerprint image preprocessing is to produce an intermediate representation of the image for which feature extraction can be carried out accurately and efficiently. Image preprocessing involves fingerprint segmentation and

enhancement. Fingerprint segmentation can be carried out by a variety of techniques described earlier in Chapter 4. For fingerprint enhancement we have used two different methods with this matching technique, which have been described in Sections 5.4.2 and 5.4.3 [HWJ98, Liu00].

b. Minutiae Extraction & Filtering

For minutiae extraction we carry out the following steps:

a. Binarization

The enhanced version of the fingerprint is then binarized by using an adaptive mean based block-wise binarization method. Fig. 7-16 shows the binarized version of a fingerprint.

Fig. 7-16 Enhanced and Binarized Fingerprint Image

b. Thinning

The binarized version of the fingerprint is then thinned using Matlab's `bwmorph` function. Fig. 7-17 shows the thinned version of a fingerprint.

Fig. 7-17 Thinned Fingerprint Image

c. Use of crossing number for minutiae detection

Minutiae Extraction is carried out by the use of the crossing numbers (CN) approach. CN of a pixel in the thinned version of a fingerprint image is defined as the number of 0 to 1 transition along a clockwise scan of the 8-neighborhood of that pixel. The mathematical relation for CN is given below:

$$cn(p) = \frac{1}{2}\sum_{i=1..8} \left| val(p_{i \bmod 8}) - val(p_{i-1}) \right| \quad (7\text{-}19)$$

Where p0 to p7 are the pixels belonging to an ordered sequence of pixels defining the 8-neighborhood of p and Val (p) is the pixel value. Crossing numbers 1 and 3 correspond to ridge endings and ridge bifurcations respectively. An intermediate ridge point has a crossing number of 2. This fact is further illustrated in Fig. 7-18.

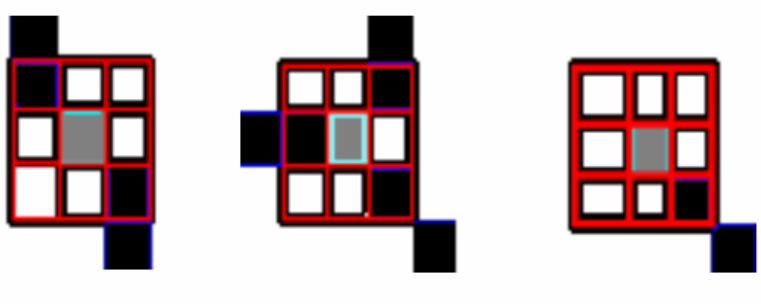

Fig. 7-18 cn (p)=2,cn (p)=3 and cn (p)=1 representing a non-minutiae region, a bifurcation and a ridge ending

Fig. 7-19 The Original FIngerprint Image and The Extracted Minutiae

The orientation of a fingerprint minutiae is taken to be the orientation of the corresponding block in which the minutiae lies. The minutiae that lie close to the border regions of the fingerprint (where the segmentation mask value is zero) are rejected during minutiae extraction.

d. Minutiae Filtering

The minutiae filtering stage extracts a large number of false minutiae such as the ones shown in Fig. 7-20.

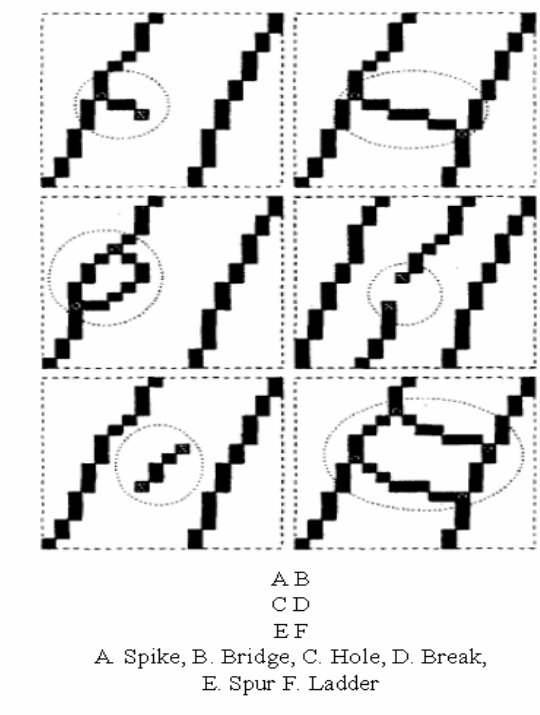

Fig. 7-20 False Minutiae

For reliable matching these minutiae must be removed for which we use an approach proposed by [TK00]. Moreover minutiae closer to each other than a certain threshold are also removed. Fig. 7-21 shows the filtered and unfiltered minutiae sets of a fingerprint image.

Fig. 7-21 False Minutiae Filtering: (Left) Fingerprint with False Minutiae and After the False Minutiae have been removed (Right)

c. Minutiae Matching

Minutiae matching comprises of the following steps:

i. Registration

The objective of registration is to recover an unknown transform between the erence and the input minutiae sets, which are given by:

$$P = \left\{ \left(p_x^1, p_y^1, \alpha^1 \right), ..., \left(p_x^{|P|}, p_y^{|P|}, \alpha^{|P|} \right) \right\} \tag{7-20}$$

$$Q = \left\{ \left(q_x^1, q_y^1, \beta^1 \right), ..., \left(q_x^{|Q|}, q_y^{|Q|}, \beta^{|Q|} \right) \right\} \tag{7-21}$$

Here the assumption is that the second fingerprint image can be obtained by applying a similarity transformation (rotation, scaling and translation) to the first image. The second point set Q is then a rotated, scaled and translated version of the set P, where the points may be shifted by a random noise, some points may be added and some may be deleted. As it is not known whether the two fingerprints belong to the same finger, we attempt to find the best transformation to the minutiae points of the set P, as many of these points as possible overlap with the minutiae points from the set Q. Two overlapping points are considered as a match only if hey have the same direction. There may be minutiae points in either set that do not match with any point in the other set.

The usual Hough transform for line detection can be generalized for point matching. The set of all allowed transformations are discretized and for each transformation, the matching score is computed. The transformation with the maximal matching score is believed to be the correct one. We consider transformations $F_{s,\theta,\Delta x,\Delta y}$ given by:

$$F_{s,\theta,\Delta x,\Delta y}\begin{pmatrix} x \\ y \end{pmatrix} = s\begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix}\begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix} \tag{7-22}$$

Where s, θ and $(\Delta x, \Delta y)$ are the scale, rotation and shift parameters, respectively. The space of transformations consists of quadruples $(s, \theta, \Delta x, \Delta y)$, Where each parameter is discretized into a finite set of values:

$$s \in \{s_1, ..., s_K\}, \theta \in \{\theta_1, ..., \theta_L\}, \Delta x \in \{\Delta x_1, ..., \Delta x_M\} \text{ and } \Delta y \in \{\Delta y_1, ..., \Delta y_M\}$$

Matching scores are collected in the accumulator array A, where the entry $A(k,l,m,n)$ counts the evidence for the transformation $F_{s_k,\theta_l,\Delta x_m,\Delta y_n}$, which is actually the number of minutiae that match between the erence minutiae set and the transformed input minutiae set. It is common in Hough Transform to

cast a vote not only in the correct bin $A(k,l,m,n)$ but also its neighbors, which makes registration more robust. The complete algorithm is shown below. Fig.7-22 shows the results of registration.

```
PROCEDURE Hough
  A(k, l, m, n) := 0,
    k = 1, ..., K;
    l = 1, ..., L;
    m = 1, ..., M;
    n = 1, ..., N
  FOR (p_x, p_y, α) ∈ P DO
    FOR (q_x, q_y, β) ∈ Q DO
      FOR θ ∈ {θ_1, ..., θ_L} DO
        IF α + θ = β THEN
          FOR s ∈ {s_1, ..., s_L} DO
(Δx, Δy) := (q_x, q_y) − s_k (cos θ_l  sin θ_l; −sin θ_l  cos θ_l) + (p_x, p_y)
            Add evidence for F_{s_k, θ_l, Δx, Δy}
          END FOR
        END IF
      END FOR
    END FOR
  END FOR
  Result := arg max_{k,l,m,n} A(k, l, m, n)
END PROCEDURE
```

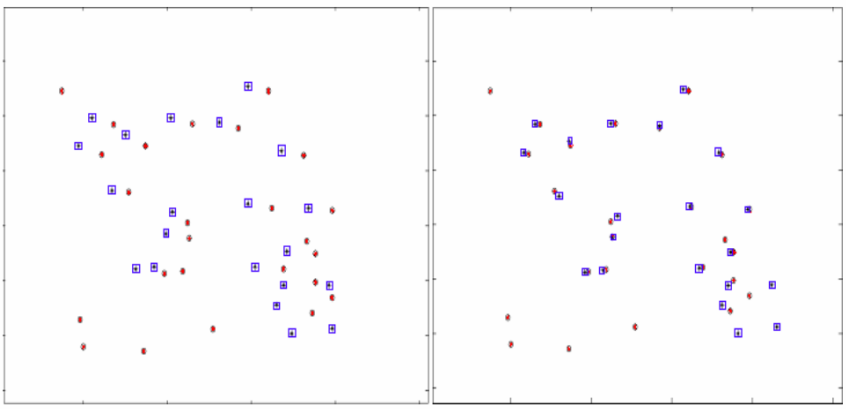

Fig. 7-22 Fingerprint Registration using Generalized Hough Transform

ii. Minutiae Pairing

After registering the two minutiae sets, the minutiae need to be paired. Two minutiae are said to be paired or matched if their components or features are equal (within some tolerance) after registration.

iii. Computation of the matching score

The matching algorithm is given below:

Input: A set of n_q minutiae points in the query fingerprint $\mathbf{f}^q$ and the rolled fingerprint database $\mathbf{f}^D = \left\{\mathbf{f}^r\right\}_{r=1}^{N}$. Let the rth database fingerprint have n_r minutiae points $\mathbf{f}^r\left(f_1^r, f_2^r, \ldots f_{n_r}^r\right)$.

Output: A list of top 10 records from the database with matching score greater than a threshold T.

Begin

FOR r = 1 to N DO

1. Register the database fingerprint with respect to the query fingerprint.
2. Compute the common bounding box for the query and reference fingerprints. Let the query print have n_q^b and reference print have n_r^b minutiae in this box.
3. Set the number of paired minutiae for the rth database fingerprint m^r to zero.

   FOR i = 1 to n_q^b DO

   Compute the tolerance vector for the ith minutiae points in the rth database fingerprint feature vector $\mathbf{f}_i^r$.

   If it can be paired with a query minutiae, then increment m^r and mark the query minutia paired. A paired query minutiae will not be paired again.

   END FOR
4. Compute the matching score (MS (q,r)):

   $$MS(q,r) = \frac{m^r * m^r}{n_r^b * n_q^b}.$$
5. Update a list of top 10 scoring database fingerprints.

END FOR

END

This algorithm computes the matching score between the two given minutiae sets after registering them. In order to reduce the amount of computation, the matching algorithm takes into account only those minutiae that fall within a common bounding box, which is the intersection of the bounding box for query and erence fingerprints. The shift in the minutiae features caused by skin plasticity or deformations, a tolerance box is created around each feature whose size depends upon ridge widths and the distance from the core point in the fingerprint as shown in Fig. 7-23.

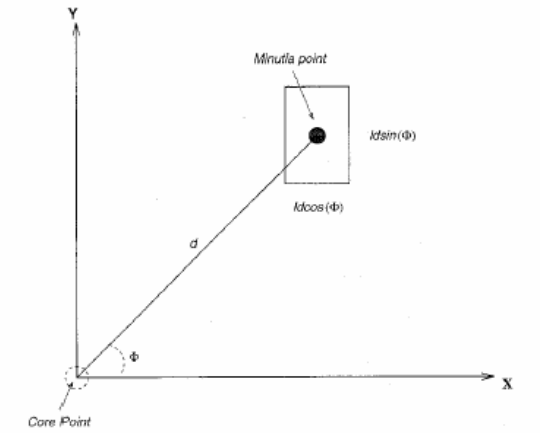

Fig. 7-23 The Tolerance Box around a Minutiae

7.4.2.2 Results:

Fig. 7-24 shows the ROC curves obtained on FVC 2000/2 databases. The total time taken by this algorithm is X seconds.

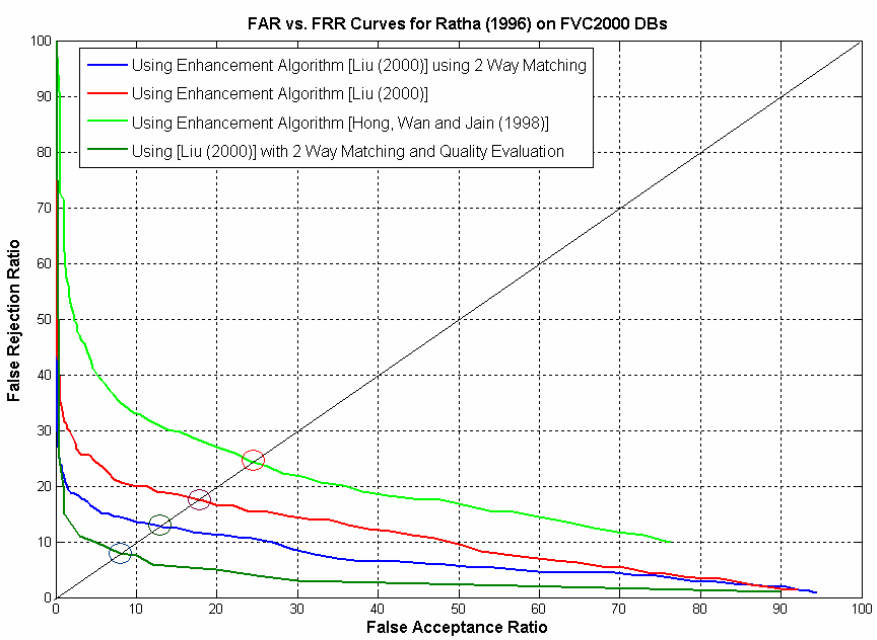

Fig. 7-24 FRR vs. FAR Curve for Rat96.

7.4.2.3 Minimum Cost Flow Optimization Based Matching

This method [JG05] is optimized for the recognition of partial fingerprints. Partial fingerprints are encountered not only in forensic investigations but in live-scan systems aswell because of the small capture area of the sensor. This approach is based on the use of localized secondary features from relative minutiae information. A

network flow optimization technique is used to obtain 1-1 correspondence of these secondary features by maximizing the number of matches and minimizing total feature distance between the secondary features of the query and erence fingerprints. The similarity score is generated on the basis of a heuristic technique. An alternative is to use a Neural Network, which can improve the matching performance. The various steps involved in this process include:

a. Preprocessing

Enhancement is carried out by the use of the technique proposed in Section 5.4.4. For fingerprint segmentation we apply the technique discussed in Section 4.3.

For minutiae extraction a chain code based minutiae detection technique is employed which has been proposed in [CWG04.]. Chain codes are a loss less representation of contours and yield a wide range of information about the contour such as curvature, direction and length etc. In case of fingerprints, if the ridge contours are traced consistently in a counter clockwise direction, the minutiae points are encountered as locations where the ridge contours exhibit a significant turn: Ridge endings occur as locations where the contour has a significant left turn and ridge bifurcations are characterized by the presence of a significant right turn in the ridge contours. Analytically the turning direction may be determined by considering the sign of the cross product of the incoming and outgoing vectors at each point. The product is right handed if the sign of equation X is positive and left handed if it is negative. The turn is considered to be significant only if $x_1 y_1 + x_2 y_2 \le T$. In practice a group of points along the turn satisfy this condition. We define the minutiae point as the center of this group.

$$\operatorname{sgn}\left(\vec{P}_{in} \times \vec{P}_{out}\right) = \operatorname{sgn}\left(x_1 y_1 - x_2 y_2\right)$$

Some false minutiae are also extracted which are removed by a heuristic based approach with the following rules:

i. The minutiae within a certain distance of each other and with similar angles are merged

ii. A minutiae whose direction is not consistent with the local ridge orientation is discarded

iii. The minutiae that are within a certain range of the border area of the fingerprint image foreground are also discarded

iv. Minutiae that lie within a certain spatial distance threshold and have opposite directions are also removed in order to handle ridge breaks which have not been recovered by the enhancement algorithm

Fig. 7-25 shows the minutiae extracted using this method.

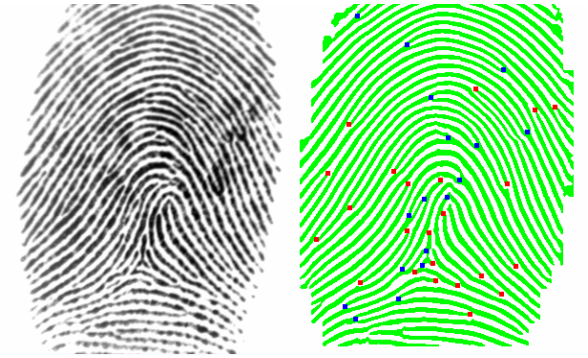

Fig. 7-25 Original Fingerprint Image and the Extracted and Conditioned Minutiae

b. Extraction of Secondary Features

A five-element secondary feature vector is generated for each minutiae $M_i(x_i, y_i, \theta_i)$ by using its two nearest neighbors $N_0(x_{n_0}, y_{n_0}, \theta_{n_0})$ and $N_1(x_{n_1}, y_{n_1}, \theta_{n_1})$. The secondary feature vector is represented by $S_i\left(r_{i_0}, r_{i_1}, \varphi_{i_0}, \varphi_{i_1}, \delta_i\right)$, where r_{i_0} and r_{i_1} are the Euclidean distances between the central minutia M_i and its neighbors N_0 and N_1 respectively. φ_{i_k} is the orientation difference between M_i and N_k, where k is 0 or 1. δ_i represents the acute angle between the line segments M_iN_0 and M_iN_1. N0 and N1 are the two nearest neighbors of the central minutia Mi and ordered not by their Euclidean distances but by satisfying the equation:

$$\left|\overrightarrow{N_0M_i} \times \overrightarrow{N_1M_i}\right| \geq 0$$

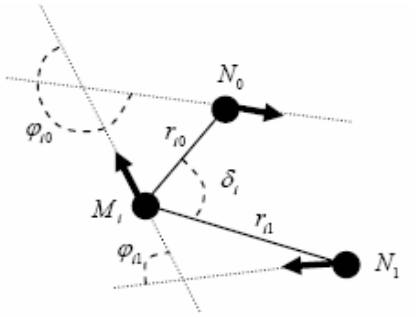

Fig. 7-26 Minutiae Triplet Features

N_0 is the first and N_1 is the second minutia that we encounter when we traverse the angle, $\angle N_0 M_i N_1$.

In order to accommodate the effects of distortions, threshold functions are associated with the features forming the secondary feature vector which define the tolerance areas in the calculation of these featuers. In this implementation, the threshold functions are based on normalized feature distances. The normalization factors depend on the distance r_{i_0} and r_{i_1} from the central minutia. The normalization factor for radial distances (r) increases with r_{i_0}, r_{i_1}. The normalization factors for angular distance (δ) and orientation difference (θ) decrease with r_{i_0}, r_{i_1}. The normalized feature distances not only give us the ability to handle different types of feature distances directly but also lect the dynamic tolerance areas as described above.

c. Minimum Cost Flow Optimization based Matching

If the number of minutiae in either of the input (I) and erence (R) minutiae is less than a predefined threshold (α) then finding the two nearest neighbors to construct a secondary feature makes it difficult to discover a match, as the fingerprint is small. In these conditions a brute-force matching of all the feature points directly by examining all the possible solutions and finding the most matches is used. For each minutia $p_i(x_i, y_i, \theta_i)$ on (I) and $q_j(x_j, y_j, \theta_j)$ on (R), p_i and q_i are considered as matched erence points and find all the other matched minutiae in the polar coordinate system by converting the matching into a minimum cost flow optimization problem which aims at finding the maximum number of matches while keeping the difference between the features to a minimum. In polar coordinates, a minutia $m_i\left(x_i, y_i, \theta_i\right)$ is represented as $\left(r_{i,k}, \Phi_{i,k}, \theta_{i,k}\right)$, with respect to the original point $m_k\left(x_k, y_k, \theta_k\right)$. Here $\left(r_{i,k}, \Phi_{i,k}\right)$ is in polar coordinates and $\theta_{i,k}$ is the orientation difference between θ_i and θ_k. The matching decision is generated on the basis of similarity between these features with certain threshold $T_r(\cdot)$, $T_\Phi(\cdot)$ and $T_\theta(\cdot)$. Two minutiae $p_i(r_{i,i'}, \Phi_{i,i'}, \theta_{i,i'})$ and $q_j(r_{j,j'}, \Phi_{j,j'}, \theta_{j,j'})$ are considered matched if:

$$\left|r_{i,i'} - r_{j,j'}\right| \le T_r(r_{i,i'})$$

$$\left|\Phi_{i,i'} - \Phi_{j,j'}\right| \le T_{\Phi}(\Phi_{i,i'})$$

$$\left|\theta_{i,i'} - \theta_{j,j'}\right| \le T_{\theta}(\theta_{i,i'})$$

These thresholds are dependent upon the values of $r_{i,i'}$ and $r_{j,j'}$ in order to handle distortion in various features.

When both (I) and (R) have a large number of minutiae, the brute-force matching technique becomes impractical because of the time complexity involved. In such cases secondary features are used for matching two minutiae sets.

The secondary feature representations $S_i\left(r_{i_0}, r_{i_1}, \varphi_{i_0}, \varphi_{i_1}, \delta_i\right)$ and $S_j\left(r_{j_0}, r_{j_1}, \varphi_{j_0}, \varphi_{j_1}, \delta_j\right)$ of two minutiae are matched by using minimum cost flow optimization with these features. Here two minutiae are considered matched if:

$$\left|r_{i_1} - r_{j_1}\right| \le T_r(r_{i_1})$$

$$\left|r_{i_0} - r_{j_0}\right| \le T_r(r_{i_0})$$

$$\left|\varphi_{i_0} - \varphi_{j_0}\right| \le T_{\varphi}(\varphi_{i_0})$$

$$\left|\varphi_{i_1} - \varphi_{j_1}\right| \le T_r(\varphi_{i_1})$$

$$\left|\delta_i - \delta_j\right| \le T_{\delta}(\delta_i)$$

The flow network representation of the MCF problem is given below.

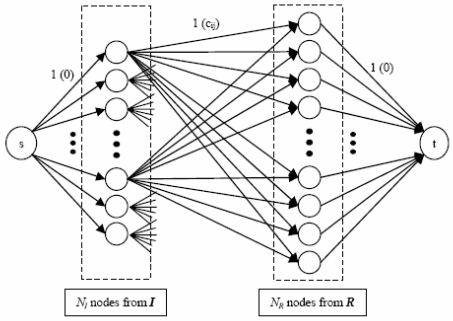

Fig. 7-27 Flow Network Representation for MCF based matching

It shows that:

- There is one and only one link that connects s to every point in the first set.
- There is one and only one link that connects t to every point in the second set.

- There is no link between the points within the same set.
- There is exactly one link between every point in first set and every point in the second set.
- Every link is associated with a capacity and a cost.

The cost matrix $c(i, j) = dist(m_i, m_j^{'})$ where $1 \le i \le N_I$ and $1 \le i \le N_R$, represents the costs of the edges between I and R. N_i and N_R are the numbers of feature points on I and R, respectively. The function, $dist(a,b)$, is the distance measure of two feature points, a and b, on I and T, respectively. For efficiency purposes, the edge between m_i and $m_j^{'}$ is removed if m_i is not in the tolerance area of $m_j^{'}$. Here , the capacity on every edge is set to 1 which ensures a 1-1 matching of feature points, and the costs associated with the edges that come from s and going to t are set to 0.

A global validation step is also used to verify a matching decision made for two minutiae sets by considering the orientations of the matched features in order to accommodate the importance of global features in matching. global structure validation is performed by plotting a histogram where each bin is about 36°. The dominating bin and its neighbors are identified. The matched feature pairs in the other bins (not in the dominating bin and its neighbors) are removed from the candidate list. Finally, according to the information derived from the matched secondary features, the minutiae from the coordinate system of the erence fingerprint R are converted into the coordinate system of query fingerprint I and the number of matched minutiae are obtained by the MCF Network discussed earlier.

d. Calculation of Matching Score

The convex hulls for a given erence point pair generates the overlapped areas of query and erence fingerprints. Let us denote the convex hull constructed from feature points on query fingerprint (I) as C_I. For every feature point on the erence fingerprint (R), if it falls inside C_I, we say it is in the overlapped area with I. Similarly, we would have a set of feature points on I that fall in the overlapped area with R.

Fig. 7-28 The Detected Overlapped Regions: a & b Belong to the same fingerprint whereas c & d belong to the same fingerprint

Thus, we can have the numbers (O_I and O_R) of feature points on overlapped areas of I and R. The following heuristic rule is used in the calculation of the matching score. Another approach for the calculation of the matching scores based on Neural Networks proposed in the original paper has not been applied.

LET $height_c$ as the height of the combined print
LET $width_c$ as the width of the combined print
LET max_h as the maximum possible height
LET max_w as the maximum possible width
LET T_m as a integer-valued threshold

IF $height_C > max_h$ OR $width_C > max_w$ THEN
  $Similarity\_score = 0$;
ELSE
  IF $O_I < 5$ THEN
    $O_I = 5$;
  END IF
  IF $O_R < 5$ THEN
    $O_R = 5$;
  END IF
  IF $n >= T_m$ AND $n > 3/5\ O_I$ AND $n > 3/5\ O_R$ THEN
    $Similarity\_score = S_{avg}$;
  ELSE
    $Similarity\_score = n^2 \times S_{avg}/(O_I \times O_R)$;
    IF $Similarity\_score > 1.0$ THEN
      $Similarity\_score = 1.0$;
    END IF
  END IF
END IF

7.4.2.4 Results

Fig. 7-29 shows the ROC curves obtained on FVC 2000/2 databases. The total time taken by this algorithm is X seconds.

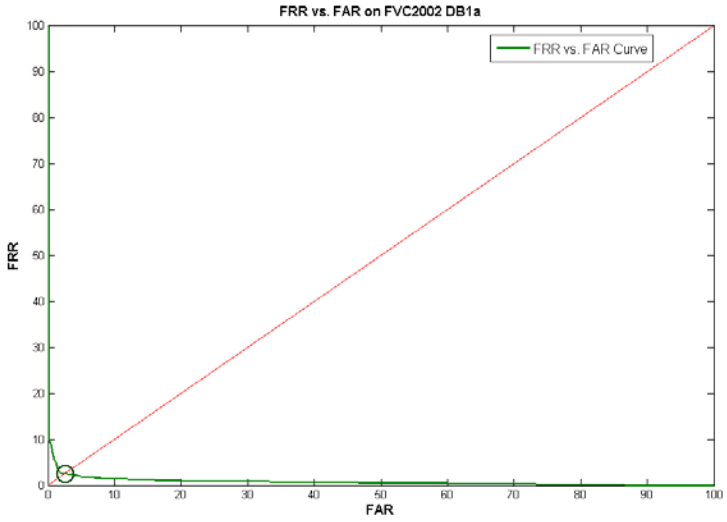

Fig. 7-29 FRR vs. FAR Curve for JG05. EER=~3%.

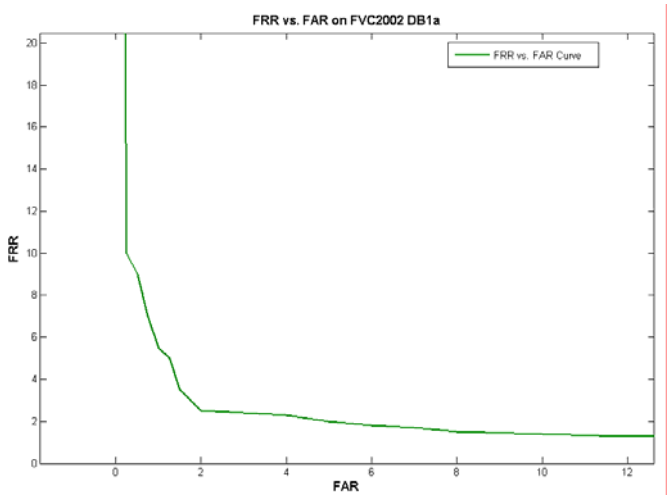

Fig. 7-30 Zoomed version of FAR-FRR Curve

Summary

Fingerprint image feature extraction and matching is the most important step in an automatic fingerprint identification system. The objective of this process is to extract features, which can be used for the computation of a matching score between two given fingerprint images in the matching stage. Fingerprint matching is made intricate by the low interclass variations between various fingerprint images of different

fingers and high intraclass variations among different images of the same finger. This variation is caused by the presence of displacement, rotation, nonlinear pressure distortions and sensor noise. There are three major approaches to the matching process, which include: a) Correlation based Matching, b) Ridge Feature based Matching and c) Minutiae based Matching. In this project we have implemented 2-ridge feature based matching, which rely on the use of the Fourier spectrum and the wavelet transform of the extraction of features used for recognition. We have also implemented 2 minutiae based approaches.

8 Techniques Implemented in AFIS

In this chapter a summary of results for different techniques is presented and the most optimal combination of techniques is discussed for use in an online AFIS. A Typical AFIS, as described in Chapter-2, comprises the following major steps:

a. Fingerprint Acquisition
b. Image Segmentation
c. Image Enhancement
d. Image Quality Evaluation
e. Feature Extraction
f. Feature Matching
g. Classification

The results of different schemes implemented for each of the steps are discussed henceforth.

8.1 Fingerprint Acquisition

Online Fingerprint Acquisition is carried out by the use of Digital Persona's URU4000 fingerprint scanner. The resolution of this gray scale fingerprint scanner is 512dpi with a capture area of 14.6mmx18.1mm. Offline fingerprints can be specified through an image file.

8.2 Image Segmentation

Image Segmentation is the process of separating the fingerprint foreground and the background. The Following Schemes for fingerprint segmentation were implemented:

i. Segmentation using Block Mean
ii. Segmentation using Block Variance
iii. Segmentation using Block Coherence
iv. Segmentation using a combination of Block Mean, Variance and Coherence
v. Segmentation using Fisher Basis [Section 4.3]

The Novel technique for fingerprint segmentation using Fisher Basis, described in detail, in Chapter 4, was chosen for fingerprint segmentation because of the low block segmentation error rate (1.8%) in comparison to Bazen et al.'s (2.45%). The other implemented techniques do not give good results in the presence of noise and are heavily dependent upon the quality (wet, dry, etc.,) of the given fingerprint image.

8.3 Image Enhancement

Fingerprint Image Enhancement is a critical step in the operation of an AFIS. For fingerprint image enhancement, the following approaches were implemented:

i. Inverse Fourier Filtering Based Enhancement [Section 5.4.1]
ii. Enhancement using Block Wise Application of Gabor Filters [Section 5.4.2]
iii. Enhancement using a Gabor Filter Bank [Section 5.4.3]
iv. Fourier Transform Based Contextual Filtering [Section 5.4.4]

Among these techniques, Fourier transform based Contextual Filtering was chosen, because of the low EER rate (~3%) obtained for fingerprint matching using this enhancement method.

8.4 Fingerprint Image Quality Evaluation

For Fingerprint Image Quality Evaluation, the technique proposed in Section 6.3 was used, which classifies the blocks in a fingerprint as low or high quality regions, by using a Back Propagation Neural Network. A rejection criterion was also implemented, which rejects a given fingerprint image, if the fingerprint foreground area is less than a certain threshold or if a significant region of the fingerprint foreground is of low quality.

8.5 Feature Extraction & Matching

The following types of features were extracted for use in fingerprint matching:

i. Wedge Ring Features extracted using the FFT [Section 7.4.1.1]
ii. Wavelet Domain Features [Section 7.4.1.2]

iii. Minutiae [Section 7.4.2.1]
iv. Minutiae Pairs [Section 7.4.2.2]

Among these, the use of minutiae pairs as features for fingerprint matching, was chosen because of their high uniqueness, which is evident from the low EER (~3%) obtained through their use in fingerprint matching, which is carried out by the use of the MCF Optimization based matching scheme proposed in Section 7.4.2.2.

8.6 Fingerprint Classification

For fingerprint classification, we use the combination approach given in section 3.8. because of it's ability to deal with partial fingerprints, which are a typical case, when fingerprint acquisition is carried out online. The classification accuracy is about 87% because of missing core points, which are required for accurate classification All the enrolled fingerprints have their class stored in the system database along with the template feature vectors. The classification scheme is used to improve the performance of the fingerprint identification process by searching for a given fingerprint only in the class to which it belongs. Because if the reduction in the size of the search space the identification speed is improved considerably. For example, for a given fingerprint database of 60 persons (small sized database), the average time to search for a single fingerprint is about 317.7411s without classification, whereas with classification, this search time is reduced to 151.0152s. However because of the errors in classification, the identification error increases. For instance, for the database used above, the identification error is 0% without classification, but due to misclassifications, the error is increaser to about 8%.

9 Conclusion and Future Work

In this project we have developed an online fingerprint identification and verification system, which is capable of carrying out fingerprint based person identification reliably and accurately and finds its applications in a number of commercial access control applications. The research work carried out under this project has resulted in the comprehension of the existing techniques for fingerprint verification and in the development of some novel methodologies involved in the processing of fingerprint images in various steps of an AFIS.

We had directed our efforts at the reduction of the error rate for fingerprint recognition and classification, therefore the optimization of the code in terms of execution time has not been performed. The performance of the system can be improved by porting the low level image processing components, currently implemented in Matlab to C++. This may enhance the effectiveness of the system in its operation as a fingerprint identification system. This performance level can also be achieved by the use of a distributed computing scheme for fingerprint matching.

Another issue that remains un-addressed is the optimization of the parameters for the image processing components, which is dependent upon the type of the sensor being used and the quality of the fingerprint images. In order to reduce the workload of the operator these parameters can be optimized automatically by using an optimization method such as genetic algorithms.

We have also not focused upon fingerprint compression, which is a fundamental requirement of a AFIS, as the number of fingerprint images to be stored is quite large. Moreover, fingerprint compression is one of the most active areas of research in this field.

Another interesting area of research is the study upon the uniqueness of fingerprint images, which is aimed at the evaluation of the probability of the similarity of two fingerprints of different fingers. Such a study is of crucial importance in defining the basis for the use of fingerprints in courts.

Fingerprint synthesis is also an emerging research area, aimed at gaining a much deeper insight to fingerprint recognition and analysis. Software such as SFinge are available in the market which can be used to generate artificial fingerprints. These

give a researcher the ability to generate an arbitrary sized fingerprint database very conveniently for performance evaluation purposes.

Studies aimed at finding the correlation between genetic diseases and fingerprints are also on going which are aimed at identifying risk factors of different diseases among individuals through the analysis of their fingerprints.

Fingerprint Classification, is still a very challenging problem and needs particular attention in order to achieve the desired 99.9% accuracy for effective use of this scheme in fingerprint identification.

References

[AAM97] F.A. Afsar, M. Arif and M. Hussain. "Fingerprint Based Person Identification & Verification for Commercial Applications", in proc. International Multitopic Conference (INMIC) 2004.

[ABC+02] Araque J., Baena M., Chalela B., Navarro D. and Vizcaya P., "Sysnthesis of Fingerprint Images" in proc. Int. Conf. On Pattern Recognition (16th), vol. 2, pp. 422-425, 2002.

[AL00] Almansa A. and Lindeberg T., "Fingerprint Enhancement by Shape Adaptation of Scale Space Operators with Automatic Scale Selection", IEEE Trans. On Image Processing, vol. 9, no. 12, pp. 2027-2042, 2000.

[AL97] Almansa A. and Lindeberg T., "Enhancement of Fingerprint Images Using Shape Adaptive Scale Space Operators," in Gaussian Scale Space Theory, J. Sporring, M. Nielsen, L. Florack, and P. Johannsen (Eds.), pp. 21-30, Klower, New York, 1997.

[Bah96] Bahuguna. Fingerprint Verification using Hologram Matched Filterings. In proc. Biometric Consortium Eighth Meeting, San Jose, California, June 1996.

[BBV+01] Bernard S., Boujemaa N., Vitale D., and Bricot C., "Fingerprint Classification using Kohonen Topologic Map", in proc. Int. conf. on Image Processing, vol. 3, pp. 230-233, 2001.

[BG00] Bazen A.M.and Gerez S.H., "Directional Field Computation for Fingerprints Based on the Principal Component Analysis of Local Gradients", Proc. ProRISC2000, 11th Ann. Workshop Circuits, Systems and Signal Processing, Nov. 2000.

[BG01] Bazen A.M., Gerez S.H., "Segmentation of Fingerprint Images", in 12th Annual Workshop on Circuits, Systems and Signal Processing, pp. 276-280, 2001

[BG02a] Bazen A.M. and Garez S.H., "Elastic Minutiae Matching By Means of Thin Plate Spine Models,", in proc. Int. Conf. On Pattern Recognition (16th), vol. 2, pp. 985 - 988, 2002.

[BG02b] Bazen A.M., and Gerez S.H. "Systematic Methods for the Computation of the Directional Fields and Singular Points of Fingerprints", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, no. 7, pp. 905-919, 2002.

[BJJ+00] Byoung H.C., Jeung S. K., Jae H.B. In-Gu B. Kee-Young Y., in Proc. 15th International Conference on Pattern Recognition, vol.2, pp. 859-862 2000.

[BLL93] Beyer J., Lake C. and Lougheed R., "Ridge Flow Determination in Fingerprint Images," in proc. Conf. Artificial Intelligence, Pattern Recognition, pp. 32-43, 1993.

[CB93] Coetzee L. and Botha E.C., "Fingerprint Recognition in Low Quality Images", Pattern Recognition, vol. 26, no. 10, pp. 1441-1460, 1993.

[CF02] Chang J.H., and Fan K.C., "A New Model for Fingerprint Classification by Ridge Distribution Sequences", Pattern Recognition, vol. 35, no. 6, pp. 1209-1223, 2002.

[CGP+05] S. Chikkerur, V. Govindaraju, S. Pankinti, R. Bolle, and N. Ratha. "Novel Approaches for Minutiae Verification in Fingerprint Images", WACV:111-116, 2005.

[CGW+95] Candela G.T., Grother P.J., Watson C.I., Wilkinson R.A., and Wilson C.L., "PCASYS – A Pattern Level Classification Automation System for Fingerprints", Tech Report: NIST TR 5647, Aug. 1995.

[CK02] Ceguerra A. and Koprinska I., "Integrating Local and Global Features in Automatic Fingerprint Verification", in proc. Int. Conf. On Pattern Recognition, (16th), vol. 3, pp. 347-350, 2002.

[CMC96] Crouzil A., Massip-Pailhes L., and Castan S., "A New Correlation Criterion Based On Gradient Fields Similarity", in proc. Int. Conf. On Pattern Recognition (13th), pp. 632-636, (1996).

[CMM00] Capelli R., Maio D. and Maltoni D., "Combining Fingerprint Classifiers" in proc. Int. Workshop on Multiple Classifier Systems, (1st), pp. 351-361, 2000.

[CMM99a] Capelli R., Maio D. and Maltoni D.,"Fingerprint Classification based on Multi-space KL", in proc. wksp. on Automatic Identification Advanced Technologies, pp. 117-120, 1999.

[CMM99b] Cappelli R., Lumini A., Maio D., and Maltoni D., "Fingerprint Classification by Directional Image Partitioning" IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 21, no. 5, pp. 402-421, 1999.

[CNJ+97] Chong M.M.S, Ngee T.H., Jun L. and Gay R.K.L., "Geometric Framework for Fingerprint Image Classification", Pattern Recognition, vol. 30, no. 9, pp. 1475-1488, 1997.

[CTR+04] Chen X., Tian J., Ren Q., Cheng J., Shang Z., Segmentation of Fingerprint Images Using Linear Classifier, EURASIP Journal on Applied Signal Processing, 2002.10. (SCIE), 2004.

[CWG04] Chikkerur, Wu, and Govindaraju (2004). Chikkerur S., Wu C., and Govindaraju V., A Systematic Approach for Feature Extraction in Fingerprint Images, ICBA, pp. 344-350, 2004.

[DL98] Drets G. and Liljenstrom H., "Fingerprint Sub-Classification and Singular Point Detection", International Journal of Pattern Recognition and Artificial Intelligence, vol. 12, no. 4, pp. 407-422, 1998.

[DR93] Donahue M.L. and Rokhlin S.I., "On The Use of Level Curves in Image Analysis," CVGIP: Image Understanding, vol. 57, no. 2, pp. 185-203, 1993.

[Eas05] Edge Lit Hologram for Live-Scan Fingerprintg. http://eastview.org/, 2005.

[Fau94] Fausett L., Fundamentals of neural networks: architectures, algorithms, and applications, Prentice-Hall, Inc., Upper Saddle River, NJ, 1994.

[FBI84] Federal Bureau of Investigation, "The Science of Fingerprints: Classification and Uses", U.S. Government Publication, Washigton D.C., 1984.

[FCW03] The Federal Computer Week Magazine, Comparison of biometric technologies, June 2003. http://www.fcw.com/fcw/articles/2003/0623/cov-side2-06-23-03.asp.

[FG96] Fitz A.P., and Green R.J., “Fingerprint Classification using Hexagonal Fast Fourier Transform”, Pattern Recognition, vol. 29, no. 10, pp. 1587-1597, 1996.

[FGG92] L. Frye, F. Gamble, and D. Grieser. Real-Time Fingerprint Verification System. Applied Optics, 31(5):652, 1992.

[FH04] Franc V. and Hlavac V., Statistical Pattern Recognition Toolbox for Matlab User’s guide, June 2004, http://cmp.felk.cvut.cz/~xfrancv/stprtool/stprtool.pdf

[FHM91] K. Fielding, J. Homer, and C. Makekau. Optical Fingerprint Identification by Binary Joint Transform Correlation. Optical Engineering, 30:1958, 1991.

[FSS92] Y. Fumio, I. Seigo, and E. Shin. Real-time Fingerprint Sensor using a Hologram. Applied Optics, 31(11):1794, 1992.

[GAK+00] Greenberg S., AladJem M., Kogan D. and Dimitrov I., "Fingerprintprint Image enhancement Using Filtering techniques" proc. Int. Conf. On Pattern Recognition (15th), vol. 3, pp. 326-329, 2000.

[GCC97] Germain R., Califano A. and Colville S., "Fingerprint Matching using Transformation Parameters", IEEE Computational Science and Engineering, vol. 4, no. 4, pp. 42-49, 1997.

[Gry95] Grycewicz T.J., "Fingerprint Identification with Joint Transform Correlator using Multiple Reference Fingerprints", proc. Of SPIE (Optical Pattern Recognition VI), vol. 2237, pp. 249-254, 1995.

[Gry96] Grycewicz T.J.,"Fingerprint Recognition using Binary Non-Linear Joint Transform Correlators", Optoelectric Devices and Systems for Processing, Critical Review, vol. CR65, 1996.

[GW92] Gonzalez R. C. and Woods R. E., Digital Image processing, Addison Wesley, 1992.

[Ham99] Hamamoto Y.,"A Gabor Filter based Method for Fingerprint Identification", in Intelligent Biometric Techniques in Fingerprint and Face Recognition, L.C. Jain, U. Halici, I. Hayashi, and S. B. Lee, (Eds.), CRC Press, Boca Raton, FL, 1999.

[Har96] M. Hartman. Compact Fingerprint Scanner Techniques. in proc. Biometric Consortium Eighth Meeting, San Jose, California, June 1996.

[HH96] Hung D.C.D and Huang C., "A Model for Detecting Singular Points of A Fingerprint", in proc. Florida Artificial Intelligence Research Symposium (9th), pp. 444-448, 1996.

[HJ99] Hong L., and Jain A.K., “Classification of Fingerprint Images” in proc. Scandinavian conf. On Image Analysis (11th), 1999.

[HS84] M. Hase and A. Shimisu. Entry Method of Fingerprint Image using a Prism. Trans. Inst. Electron. Commum. Eng. Japan, J67-D:627-628, 1984.

[HS98] Harris Semiconductor. The Harris Semiconductor Homepage. http://www.semi.harris.com/fingrloc/index.htm, 1998.

[HTM+02] Hatano T., Adachi T., Shigematsu S., Morimura H., Onishi S., Okazaki Y. and Kyuragi H., "A Fingerprint Verification Algorithm using the Differential Matching Rate" in proc. Int. Conf. On Pattern Recognition (16th), vol. 3, pp. 799-802, 2002.

[Hun93] Hung D.C.D., "Enhancement and Feature Purification of Fingerprint Images," Pattern Recognition , vol. 26, no. 11, 1661-1671, 1983.

[HWJ98] Hong L., Wan Y. and Jain A.K., "Fingerprint Image Enhancement: Algorithms and Performance Evaluation," IEEE trans. On Pattern Analysis and Machine Intelligence, vol. 20, no. 8, pp. 777-789, 1998.

[IZ86] Isenor D.K. and Zaky S.G., "Fingerprint Identification using Graph Matching", Pattern Recognition, vol. 19, pp. 113-122, 1986.

[JF91] Jain, A.K. and Farrokhnia, F., (1991) Unsupervised texture segmentation using Gabor filters, Pattern Recognition , vol. 23 , 1167-1186.

[JDM00] Jain A.K., Duin R.P.W., and Mao J., Statistical Pattern Recognition: A Review," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 22, no. 1, 2000.

[JG05] T. Jea and V. Govindaraju, "A Minutia-Based Partial Fingerprint Recognition System", Pattern Recognition, 2005.

[JHP+97] Jain A.K., Hong L., Pankanti S. and Bolle R., "An Identity Authentication System Using Fingerprints," Proc. Of IEEE, vol. 85, no. 9, pp. 1365-1388, 1997.

[JHP00] Jain A.K., Prabhakar S., Hong L., and Pankanti S., "Filterbank based Fingerprint Matching", IEEE Transactions on Image Processing, vol. 9, pp. 846-859, 2000.

[Jia00] Jiang X., "Fingerprint Image Ridge Frequency Estimation by Higher Order Spectrum," in proc. Int. on Image Processing, vol. 1, pp. 462-465, 2000.

[JM02] Jain A.K. and Minut S., “Hierarchical Kernel Fitting for Fingerprint Classification and Alignment” in proc. Int. Conf. On Pattern Recognition (16th), vol. 2, pp. 469-473, 2002.

[JMT+75] G. Johnson, D. McMahon, S. Teeter, and G. Whitney. A Hybrid Computer Processing Technique for Fingerprint Identification. IEEE Trans. Computers, 24:358-369, 1975.

[JPH99] Jain A.K., Prabhakar S., and Hong L., “A Multichannel Approach to Fingerprint Classification”, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 21, no. 4, pp. 348-359, 1999.

[JPP00] A.K. Jain, S. Prabhakar and S. Pankanti, On Quantifying Fingerprint Individuality, Technical Report MSU, October 2000.

[JRL+96] Johnnesen F.R., Raaschou S., Larsen O.V., and Jurgensen P., "Using Weighted Minutiae for Fingerprint Identification", in proc. Advances in Structral and Syntactical Pattern Recognition, pp. 289-299, 1996.

[JRL97] Jain A.K., Ratha N., and Lakshmanan S., Object detection using gabor filters, Pattern Recognition, vol. 30, pp. 295-309, 1997.

[JRP01] A. K. Jain, A. Ross, S. Prabhakar, Fingerprint matching using minutiae and texture features, in: Proc. International Conference on Image Processing (ICIP), Thessaloniki, Greece, 2001, pp. 282--285.

[JSW93] P. Jones, B. Santer, and T. Wigley. Correlation Methods in Fingerprint Detection Studies. Climate Dynamics, 8(6):265, 1993.

[Kar89] F. Karen. Encryption, Smart Cards and Fingerprint Readers, IEEE Spectrum, 26(8):22, 1989.

[KBV02] Klein S., Bazen A.M. and Veldhuis R.N.J., Fingerprint Image Segmentation Based on Hidden Markov Models, in proc. ProRISC 2002, 13th Annual Workshop on Circuits, Systems, and Signal Processing.

[KJ96] Karu K. and Jain A.K., “Fingerprint Classification,” Pattern Recognition, vol. 29, no. 3, pp. 389-404, 1996.

[KK01] Koo W.M. and Kot A., "Curvature Based Singular Points Detection", in proc. Int. Conf. On Audio and Video-Based Biometric Person Authentication (3rd), pp. 229-234, 2001.

[KRF00] Kovacs-Vajna Z. M., Rovatti R. and Frazzoni M., "Fingerprint Ridge Distance Computation Methodologies," Pattern recognition, vol. 33, no. 1, pp. 69-80, 2000.

[KT84] Kawagoe M. and Tojo A., "Fingerprint pattern Classification," Pattern Recognition, vol. 17, pp. 295-303, 1984.

[KV00] Kovacs-Vajna Z.M., "A Fingerprint Verification System based on Triangular Matching and Dynamic Time Warping", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 22, pp. 1266-1276, 2000.

[KW87] Kass M. and Witkin A., "Analyzing Oriented Patterns," Computer Vision, Graphics, and Image Processing, Vol. 37, No. 3, pp. 362-285, 1987.

[KWG83] Knutsson H., Wilson R., Granlund H.G., "Anisotropic Non-Stationary Image Estimation and its Applications- Part I: Restoration of Noisy Images", IEEE Trans on Communications, vol. COM-31, no. 3, pp. 388-397, March, 1983.

[Liu00] Liu J., "Fingerprint Processing Techniques for Biometric Applications," Research Report, School of Electrical and Electronic Engineering.

[LJY02] Lim E., Jiang X.D. and Yau W.Y., Fingerprint Quality and Validity Analysis, In proc. IEEE International Conference on Image Processing, vol. 1. pp. 469-472, Rochester, N. Y. USA, September 2002. (ICIP02).

[LKB+83] F.R. Livingstone, L. King, J. Beraldin, and M. Rioux. Development of a Real-Time Laser Scanning System for Object Recognition, Inspection and Robot Control. In proc. SPIE on Telemanipulator Technology and Space Telerobotics, vol. 2057, pp. 454-461, Boston, Massachusetts, September 1993.

[LMM97] Lumini A., Maio D. and Maltoni D., “Continuous vs. Exclusive Classification for Fingerprint Retreival”, Pattern Recognition Letters, vol. 18, no. 10, pp. 1027-1034, 1997.

[LMM99] Lumini A., Maio D. and Maltoni D., “Inexact Graph Matching for Fingerprint Classification”, Machine Graphics and Vision (Special Issue on Graph Transformations in Pattern Generation and CAD), vol. 8, no. 2, pp. 231-248, 1999.

[LN99] Lee SW, Nam BH. "Fingerprint recognition using wavelet transform and probabilistic neural network", in proc. International Joint Conference on Neural Networks, 1999.

[LS85] Love, P.L., and Simaan, M., Segmentation of a Seismic Section Using Image Processing and Artificial Intelligence Techniques, Pattern Recgnition vol. 18, no. 6, 1985, pp. 409-419.

[Mat05] Matlab, v 7.0, The Mathworks Inc., www.mathworks.com, (2005)

[MBF+97] Mardia K.V., Baczkowski A.J., Feng X. and Hainsworth T.J., "Statistical Methods for Automatic Interpretation of Digitally Scanned Fingerprints," Pattern Recognition Letters, vol. 18, no. 11-13, pp. 1197-1203, 1997.

[MC89] Mehtre B.M. and Chattergee B, "Segmentation of Fingerprint Images-A composite Method," Pattern Recognition , vol 22, no. 4, pp. 381-385, 1989.

[MF75] Moayer B., and Fu K., “A Syntactic Approach to Fingerprint Pattern Recognition, Pattern Recognition, vol. 7, pp. 1-23, 1975.

[MF76] Moayer B., and Fu K., “An Application of Stochastic Languages to Fingerprint Pattern Recognition”, Pattern Recognition, vol. 8, pp. 173-179, 1976.

[MF86] Moayer B., and Fu K., "A Tree System Approach for Fingerprint Pattern Recognition", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 8, no. 3, pp. 376-388, 1986.

[MJT+75] McMohan D., Johnson G.L., Teeter S.L. and Whitney C.G., "A Hybrid Optical Computer Processing Technique for Fingerprint Identification", IEEE Transaction on Computers, C-24, pp. 358-369, 1975.

[MK] Moon J. and Kim H., Study on Metrics for Fingerprint Image Quality, (Report) Department of Information and Communication Engineering, Inha University, Incheon, Republic of Korea.

[MM95] Maio D. and Maltoni D., An Efficient Approach to Online Fingerprint Verification, Proc. 8th. Int. Symposium on AI, Monterrey, Mexico, Oct., 1995.

[MM96] Maio D. and Maltoni D., “A Structural Approach to Fingerprint Classification” in proc. Int. conf. On Pattern Recognition (13th), 1996.

[MM98a] Maio D. and Maltoni D., "Ridge-line density Estimation in Digital Images," in proc. Int. conf. On Pattern Recognition, (14th), pp. 534-538, 1998.

[MMC+00] Maio D., Maltoni D., Cappelli R., Waynan J.L., and Jain A.K., "FVC2000: Fingerprint Verification Competition", tech. Report: DEIS, University of Bologna, Sept. 2000.

[MMC+02] Maio D., Maltoni D., Cappelli R., Waynan J.L., and Jain A.K., "FVC2002: Fingerprint Verification Competition",IEEE Transaction on Pattern Analysis and Machine Intelligence, vol. 24, no. 3, pp. 402-412, 2002.

[MMJ+03] D. Maltoni, D. Maio, A. K. Jain, and S. Prabhakar, Handbook of Fingerprint Recognition, Springer, 2003.

[MMK87] Mehtre B.M., Murthy N.N., Kapoor S., and Chatterjee B., Segmentation of Fingerprint Images Using the Directional Image, Pattern Recognition, vol. 20, no. 4, pp. 429-435, 1987.

[Mos01] Mosorov V., Using tophat transformation for fingerprint image segmentation, International Conference on Signals and Electronic Systems, 2001.

[MRF01] Marcialis G.L., Roli F., and Frasconi P., "Fingerprint Classification by Combination of Flat and Structural Approaches", in proc. Int. Conf. On Audio and Video-Based Biometric Person Authentication (3rd), pp. 241-246, 2001.

[MSW+96] K. McCalley, D. Setlak, S. Wilson, and J. Schmitt. A Direct Fingerprint Reader. In proc. CardTech/SecurTech, vol. I: Technology, pages 271-279, Atlanta, Georgia, May 1996.

[MT93] Moscinska K. and Tyma G., "Neural Network Based Fingerprint Classification", in proc. Int. Conf. On Artificial Neural Networks (3rd), 1993.

[NB92] Nill, N. B. and Bouzas B., Objective Image Quality Measure Derived From Digital Image Power Spectra, Optical Engineering, Vol. 31, No. 4, pp. 813-825, 1992.

[NRB99] Nalini K. Ratha, Ruud M. Bolle, Fingerprint Image Quality Estimation, IBM Computer Science Research Report RC 21622, 1999.

[ON89] O'Gorman L. and Nickerson J. V., "An approach to Fingerprint Filter Design," Pattern Recognition, vol. 22, no. 1,pp. 29-38, 1989.

[Per98] Perona P., "Orientation Diffusions," IEEE Trans. On Image Processing, vol. 7, no. 3, pp. 457-467, 1998.

[PHR+02] Pankanti S., Haas N., Ratha N.K., Bolle R.M., Quantifying Quality: A case study in fingerprints, Proc of IEEE Conference on AutoID 2002.

[PIS05] The Primode Information Security Glossary Website Glossary, July 2005. http://www.primode.com/glossary.html

[Pol96] Polikarpova N., "On the Fractal Features in Fingerprint Matching", in proc. ICPR96, 1996.

[PPB+01] attichis M.S., Panayi G., Bovik A.C. and Hsu S.P., "Fingerprint Classification using and AM-FM Model", IEEE Transactions on Image Processing, vol. 10, no. 6, pp. 951-954, 2001.

[PPJ02] . Pankanti, S. Prabhakar, and A. K. Jain, On the Individuality of Fingerprints, in Proc. Computer Vision and Pattern Recognition (CVPR) , pp. 805-812, Hawaii, Dec. 11-13, 2001. Also presented on invitation at CardTech/SecurTech 2002, California State Division - International Association for Identification (CSD-IAI) Seminar 2002, and Asilomar Microcomputer Workshop 2002.

[Pra97] radenas R., "Directional Enhancement in the frequency Domain of Fingerprint Images," proc. Of SPIE, vol. 2932, pp. 150-160, 1997.

[QJX] Qun R., Jie T., Xiaopeng Z., Automatic Segmentation of Fingerprint Images, Chinese Academy of Sciences, PR China

[QJY+02] Qun R., Jie T., Yuliang H. and Jiangang C. "Automatic Fingerprint Identification using Cluster Algorithm" in proc. Int. Conf. On Pattern Recognition (16th), vol. 2, pp. 398-401, 2002.

[Rat96] Ratha N.K, Karu K, Chen S. , Jain A.K., " Real-time matching system for large fingerprint databases," IEEE Transaction on Pattern Analysis and Machine Intelligence, vol 18, no. 8, pp. 799-813, 1996.

[RB80] Rao K. and Black K., "Type Classification of Fingerprints: A syntactic Approach," IEEE transactions on pattern analysis and machine Intelligence, vol. 2, No. 3, pp. 223-231, 1980.

[RCJ95] Ratha N. K., Chen S. Y., and Jain A.K., "Adaptive Flow Oriented Based Feature Extraction In Fingerprint Images," Patter Recognition, vol. 28, no. 11, pp. 1657-1672, 1995.

[RTO01] Ramo P., Tico M., Onnia V., and Saarinen J., "Optimized Singular Point Detection Algorithm for Fingerprint Images", in proc. Int. conf. On Image Processing, vol. 3, pp. 242-245, 2001.

[SA94] Stosz J.D. and Aleya L.A., "Automated System for Fingerprint Authentication using Pores and Ridge Structure", proc. Of SPIE (Automatic Systems for the Identification and Inspection of Humans), vol. 2277, pp. 210-223, 1994.

[SBT+94] Shumurun A., Bjorn V., Tam S. and Holler M., "Extraction of Fingerprint Orientation Maps Using a Radial Basis Function Recognition Accelarator," in proc. Int. Conf. On Neural Networks, vol. 2, pp. 1186-1190, 1994.

[Sen01] Senior A., “A Combination Fingerprint Classifier”, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 23, no. 10, pp. 1165-1174, 2001.

[Sen97] Senior A., “A Hidden Markov Model Fingerprint Classifier” in proc. Asilomar conf. on Signals Systems and Computers, (31st), pp. 306-310, 1997.

[SKK01] Shen L., Kot A.and Koh W.M., "Quality Measures of Fingerprint Images", in proc. Int. Conference on Audio and Video Based Biometric Person Authentication (3rd), pp. 266-271, 2001.

[SM05] Robert K. Rowe, Lumidigm, Inc. A Multispectral Sensor for Fingerprint Spoof Detection, at SensorMag Homepage, http://www.sensorsmag.com/articles/0105/25/, (2005)

[SM92] Srinivasan V.S. and Murthy N.N.,"Detection of Singular Points in Fingerprint Images", Pattern Recognition, vol. 25, and no. 2, pp. 139-153, 1992.

[SM93] Sherlock B.G. and Monro D.M., "A model for Interpreting Fingerprint Topology," Pattern Recognition, vol. 26, no.7, pp. 1047-1055, 1993.

[SMM92] Sherlock B.G., Monro D.M. and Millard K., "Algorithm for Enhancing Fingerprint Images," Electronic Letters, vol. 28, no. 18, pp. 1720, 1992.

[SMM94] Sherlock B.G., Monro D.M. and Millard K.,"Fingerprint Enhancement by Directional Fourier Filtering," IEEE proc. Vision Image and Signal Processing, vol. 141, no.2, pp. 87-94, 1994.

[SS69] Stock R. M. and Swonger C.W., "Development and Evaluation of a Reader of Fingerprint Minutiae," Technical Report: no. XM-2478-X-1:13-17, Cornell Aeronautical Lab, 1969.

[SSL94] Shan Y., Shi P. and Li J., "Fingerprint Preclassification using Key-Points", in proc. Int. Symp. On Speech Image Proc. And Neural Network, vol. 1, pp. 308-311 (1994).

[Sto77] R. Stock. Automatic Fingerprint Reading. in proc. Int. Carnahan Conf. on Electronic Crime Countermeasures, pp. 16-28, University of Kentucky, Lexington, Kentucky, 1977.

[SU02] Sato, S., and Umezaki, T., A fingerprint segmentation method using a recurrent neural network, Proceedings of the 2002 12th IEEE Workshop on Neural Networks for Signal Processing, 2002. pp. 345- 354

[SW91] J. Schneider and D. Wobschall. Live Scan Fingerprint Imagery using High Resolution C-Scan Ultrasonography. in proc. 25th Int. Carnahan Conf. on Security Technology, pp. 88-95. 1991.

[SWY02] Sen W., Weiwei Z., Yangsheng W., New Features Extraction and Application in Fingerprint Segmentation, (report) National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, 2002

[TIR01] Tico, M.; Immonen, E., Ramo, P., Kuosmanen, P., Saarinen, J. "Fingerprint Recognition Using Wavelet Features". In proc. Of The 2001 IEEE International Symposium on Circuits and Systems, pp. 21-24, vol.2, 2001.

[TK00] Tico, M. & Kuosmanen, P., “An Algorithm for Fingerprint Image Post processing” in Thirty-Fourth Asilomar Conference on Signals, Systems, and Computers, October 29, 2000 - November 1, 2000.

[TK99] M. Tico, P. Kuosmanen, "A Multiresolution Method for Singular Points Detection in Fingerprint Images", Proc. of IEEE International Symposium of Circuits and Systems (ISCAS '99), vol. IV, pp. 183-186, 1999.

[TKS01] Tico M., Kuosmonen P., and Saarinen J., "Wavelet Domain Features for Fingerprint Recognition", Electronics Letters, vol. 37, no. 1, pp. 21-22, 2001.

[TWW04] Tabassi E., Wilson C.L., Watson C.I., Fingerprint image quality, (NIST report) , August 2004.

[UGS01] Udupa R., Garg G. and Shrma P., "Fast and Accurate Fingerprint Verification", in proc. Int. Conf. On Audio- and Video-based Biometric Person Authentication (3rd), pp. 192-197, 2001.

[Ver05] Veridicom. Veridicom Homepage. http://www.veridicom.com/, 2005.

[VG96] Vizcaya P.R. and Gerhardt L.A., "A NonLinear Orientation Model For Global Description of Fingerprints," Pattern recognition, vol. 29, no. 7, pp. 1221-1231, 1996.

[WGC00] Watson C.L., Grother P.J. and Casasent D.P., "Distortion-Tolerant Filter for Elastic-Distorted Fingerprint Matching", tech report: NIST IR6489, National Institute of Standards and Technology, Gaithersburg, MD, 2000.

[WH98] Weldon T.P. and Higgins W.E., An Algorithm for Designing Multiple Gabor Filters for Segmenting Multi-Textured Images, 1998 IEEE Int. Conf. on Image Processing (ICIP98), Chicago, IL, 4-7 Oct. 1998.

[WM01] Willis A.J., and Myers L., “A Cost Effective Fingerprint Recognition System for Use with Low Quality Prints and Damaged Fingertips”, Pattern Recognition, vol. 34, no. 2, pp. 255-270, 2001.

[WW92] C.I.Watson and C.L.Wilson, Nist Special Database 4, Fingerprint Database. nology, 1992.

[WW93] C.I.Watson, Nist Special Database 14, Fingerprint Database, National Institute of Science and Technology, 1993.

[WYJ98] Wei D., Yuan Q., and Jie T., “Fingerprint Classification System with Feedback Mechanism based on Genetic Algorithm” in proc. Int. Conf. On Pattern Recognition (14th), vol 1., pp. 163-165, 1998.

[Yam00] Yambor W.S., Analysis of PCA-Based and Fisher Discriminant-Based Image Recognition Algorithms, M.S. Thesis, (Technical Report CS-00-103, Computer Science). July 2000.

[YFP01] Yao Y., Frasconi P., and Pontil M., “Fingerprint Classification with Combination of Support Vector Machines”, in proc. Int. Conf. On Audio and Video-Based Biometric Person Authentication (3rd), pp. 253-258, 2001.

[YIY90] Yahagi H., Igaki S. and Yamagishi F., "Moving-Window Algorithm for Fast Verification", in proc. Southeastcon conf., pp. 343-348, 1990.

[YLJ+03] Yang J.,Liu L., Jiang T. and Fan Y." A modified Gabor filter design method for fingerprint image enhancement", Pattern Recognition Letters, Volume 24 , Issue 12 (August 2003), pp.1805 - 1817, 2003.

[Zho04] Zhongchao S., A new fingerprint image segmentation algorithm based on ROI, Institute of Automation, Chinese Academy of Science, 2004.

Appendix-A: System Development

In this chapter we investigate different development aspects of a general-purpose fingerprint based person identification and verification system, which is the software development objective of this project. Section 8.1 gives an overview of the software requirements specifications for the system. Section 8.2 presents the architecture of the system with Section 8.3 detailing the system design. The various implementation issues are described in Section 8.4. Section 8.5 presents the software test report. Section 8.6 details the deployment process for the system with Section 8.7 giving the performance analysis of system developed. In Section 8.8 we look into future extensibilities for the software.

System Requirements

The objective of the software to be developed is to use fingerprints of a person for the purpose of person identification and verification. The system is to maintain a person database containing information about different persons along with the images of their fingerprints taken offline by some means. The system should be able to establish the identity of a person given his/her fingerprints. A large number of person identification systems based on fingerprints are now place but most of the economical systems face the following major problems:

- Low recognition rate (i.e. No of fingerprints recognized in a unit time)
- High false acceptance and false rejection ratios
- Incorrect system response for images of low quality
- High Computational Complexity

Our major emphasis is to develop a system that takes into account the issues mentioned above.

Product Functional Specifications

The objective of this project is to develop a fingerprint based person identification and verification system that would provide the functions listed below:

Person Enrollment

In this mode the operator (user) adds certain information about a person to a person-database. The operations in this mode are password protected.

Fingerprint Classification

In this mode a user may classify a given fingerprint image into any of the classes mentioned earlier.

Fingerprint based Person Verification

In this mode the user may carry out fingerprint verification in two different ways given below

- By providing two fingerprint images and verifying whether the two are from the same finger (referred as Identical fingerprint verification in future)
- By providing the ID of the person and his/her fingerprint and verifying whether the fingerprints belong to the person whose ID has been given (referred as automatic person verification in future)

Fingerprint based Person Identification

In this mode the user may identify a person whose information is present in the database using an image of the person's fingerprint.

Performance Requirements

Time Requirements

The verification/identification and enrollment times must be within 10 seconds on a P-IV 2.2 GHz processor with 512MB RAM running Windows 2000.

Space Requirements

Each stored feature vector of a fingerprint must take within 1KB of space

Accuracy

The Equal Error Rate for the system must be within 10% with an image reject rate of less than 5%.

System Database Specifications

The person database functions as a repository of information about different persons. It also serves as a container for finger print images and features. The system is to identify persons from this database when provided with the fingerprint of the person. In this Section we examine the data requirements for the system.

The information to be stored about a person in the person databases consist of two major parts

a. Token Identification Information

This part consist of the following fields

- A Unique Person ID
- Person Name. If this field is not known then this field may be left empty.

b. Fingerprints related information

For enrollment a person should have at least one record of a fingerprint image in the database. A person may have multiple records for his/her fingerprints each of which should always belong to a different finger. However fingerprints for which the information as to what hand or finger they belong does not exist can also be recorded. For each fingerprint the following information is to be stored

- Unique Fingerprint ID
- Person ID of the person to which this fingerprint belongs
- Hand Type Code i.e. the ID of the hand (Left or Right) to which this fingerprint belongs. If this field is not known then this field may be left empty.
- Finger Type Code i.e. to which type of finger (Index, Middle, Ring, Little or thumb) this fingerprint belongs. If this field is not known then it may be left empty
- Fingerprint Image.
- Fingerprint features. Whenever a fingerprint is added its corresponding features are stored in this field
- Fingerprint Class ID. This identifies the class to which a fingerprint belongs (e.g. arch, whorl etc.). Whenever a fingerprint is added to the database its corresponding class may either be entered manually or automatically determined by using the automatic fingerprint classification feature.

From the viewpoint of the person database the following fields are mandatory

- Person ID
- Unique Fingerprint ID
- Fingerprint image
- Fingerprint features
- Fingerprint class ID

The optional fields include

- Hand Type Code
- Finger Type Code
- Person Name

Thus the information to be provide by the user for the database consists of

- Person ID
- Person Name
- Fingerprint image
- Fingerprint class ID. This can also be determined automatically as mentioned earlier

This information is to be kept in unencrypted format in the database.

User Interface Specification

The user interface is to be a tab-based single Document Interface. This refers to the use of tabs to access the different modes of operations of the software. The different tabs will be placed on a single document interface. The system interface should enable the user to carry out the functional requirements specified earlier in an effective manner.

System Constraints

In this Section we consider the different constraints that were placed by project supervisors for system development.

Constraints on input image

- The input images must be of 500dpi and 300x300 pixels (Other sizes and resolutions may also be supported).

- The input images should consist of only 01 fingerprint without any overlapping with any other fingerprint
- The image formats to be supported include bitmaps (.bmp) and tiff files

Implementation Language Constraints

- For database Implementation MS SQL Server 2000 shall be used
- For programming the major processing portion of the application MATLAB 6.5/7.0 will be used
- For user interface design and the development of the system core and user interface, Visual Basic .NET (Framework 1.0 or above) will be used.

Enrollment Constraints

- An input image shall be enrolled if it contains a minimum of 12 minutiae points as specified by an FBI standard
- A person shall be enrolled in the database if and only if the following fields are provided
    - Person ID
    - One or more fingerprint image that satisfies the criterion mentioned above for input images

Use Case Diagrams

System development was carried out by using an iterative development strategy with functional requirement based task division. Fig. A-1 shows the use case diagram for the first version of the software, which is focused on person identification and verification. Fig. A-2 shows the use case diagram for version-2 of the system in which fingerprint classification will be added to the system.

Below a brief description of each of the use cases is presented:

a. Enroll Person in DB

Here the operator can enroll a person into the database by providing the information mentioned earlier if the enrollment constraints have been met. In response to this action the system would generate a complete report a verifying the fact that the person has been added to the database. While adding the fingerprints the corresponding features are also extracted which are then added to the database. The user may either

manually enter the class of the fingerprints or have it determined by utilizing the classification module.

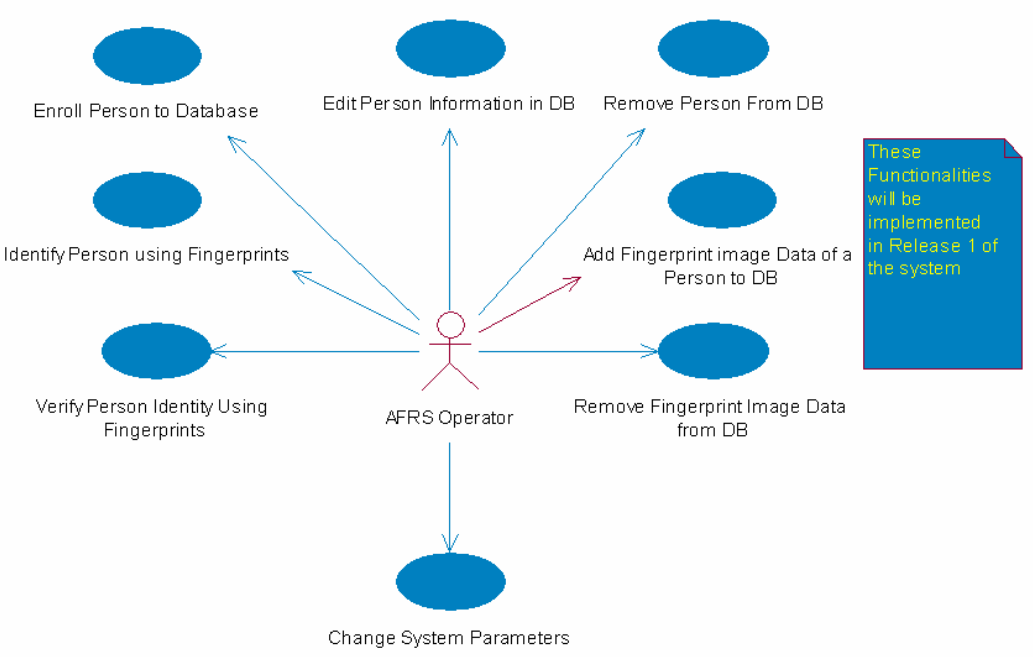

Fig. A-1 Use case Diagram for Version-1

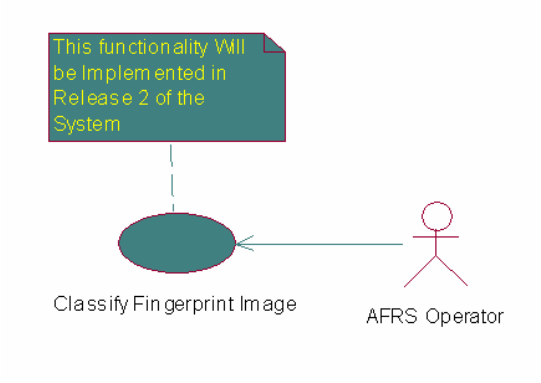

Fig. A-2 Use Case Diagram for Version-2

b. Edit/Remove Person from DB

If the user wishes to remove or edit the record of a particular person specified by the person ID then this use case will be utilized. Removal means the complete deletion of both the token and fingerprint information about the person from the database. Editing refers to the addition/removal of a particular field of information about a person from the database.

c. Add/Remove Fingerprint Image Data

A person must have at least one fingerprint in the database. If there is only one fingerprint present then that fingerprint can only be replaced by another fingerprint of the same person. New fingerprints can also be added by utilizing this functionality. In response to this action the system updates the database and reports the corresponding updates to the user. While adding the fingerprints the corresponding features are also extracted which are then added to the database. The user may either manually enter the class of the fingerprints or have it determined by utilizing the classification module. A template of the fingerprints is developed on the basis of the existing images and this template is used to identify/verify a person.

d. Classify Fingerprints

If fingerprint classification is desired then this functionality can be activated. The user input would consist of the name of a file containing the fingerprint image. The output by the system would be the class to which the input fingerprint belonged.

e. Verify fingerprints

The user can carry out

- Identical fingerprint verification
- Automatic person verification

by providing the required number of images and person information explained earlier.

f. Identify Person in DB using fingerprints

The input by the user would be the name of the file containing the fingerprint of the person to be identified. The user may optionally enhance the image. The output of the system would be the person ID, name, hand type code and the finger type code.

System Architecture

The conceptual architecture of the system is shown below.

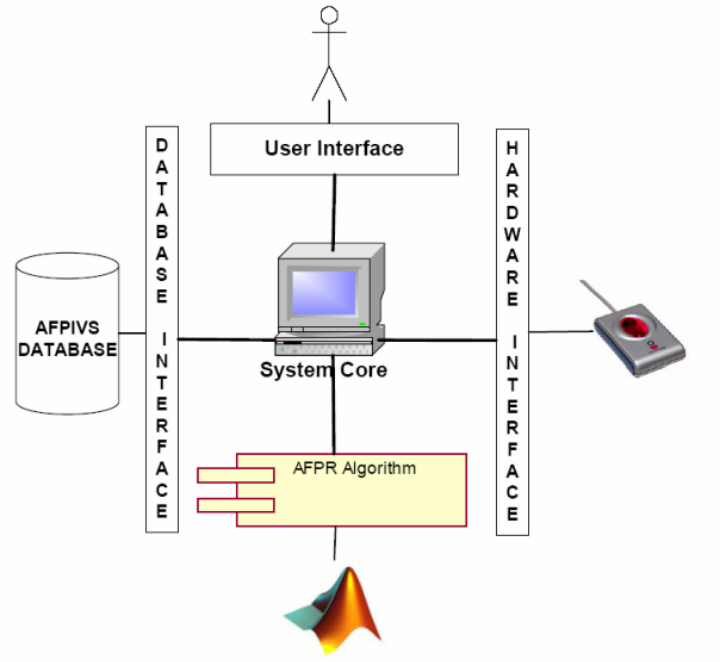

Fig. A-3 System Architecture

The major components of the system include:

a. AFPIVS Database

This is the personal database for the system. The complete specifications of the database have been given earlier. The system database is to be developed using SQL.

b. AFPR Algorithm

The AFPR algorithm component is responsible for carrying out the Image Processing and Pattern Matching tasks for the system when presented with fingerprint image(s). These tasks include:

i. Fingerprint Segmentation
ii. Fingerprint Enhancement
iii. Feature Extraction
iv. Quality Evaluation
v. Feature Matching
vi. Classification

It is a dll component originally implemented in Matlab.

c. System Core

The system core is responsible for integrating and synchronizing other system components. Its task is to coordinate low-level tasks such as acquisition, feature

extraction, matching etc. in order to fulfill a high level functionality such as identification, verification, enrollment etc.

d. User Interface

The task of the user interface is to provide the user with a means of interaction with the system through which the user is able to specify the task that he wishes to carry out and view its results.

e. Database Interface

This is an interoperability interface between the system core and the AFPIVS database which is aimed at providing a relatively higher level of abstraction to the system core during its interaction with the system database.

f. Hardware Interface

This is an interface between the system hardware (fingerprint scanners) and the system core, which is aimed at providing a high level of abstraction to the system core in its interaction with the sensor.

System Design

Based on the system architecture discussed in the previous Section, the system design has been developed at three levels, which are described below:

Software Design

The different classes involved in the software design of the system are shown in the class diagram in Fig. A-4. Individual class diagrams are shown in Fig. A-5 The objective of the major (high-level) classes is given below:

i. FrmAFPIVSMain

This class constitutes the user interface and acts as a container for objects of some of the other classes.

ii. cFPRS

This class constitutes the system core and is responsible for carrying out the functionalities as detailed by the user interface component and directs its worker objects such as Verification Mangers, Classification Managers etc. accordingly.

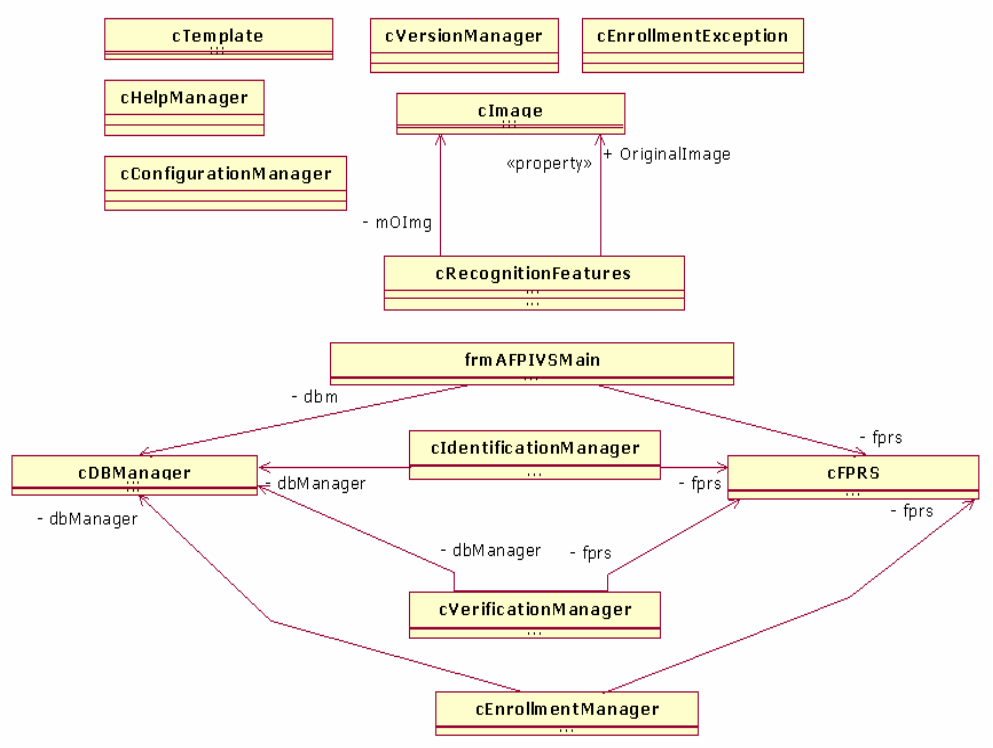

Fig. A-4 System Class Diagram

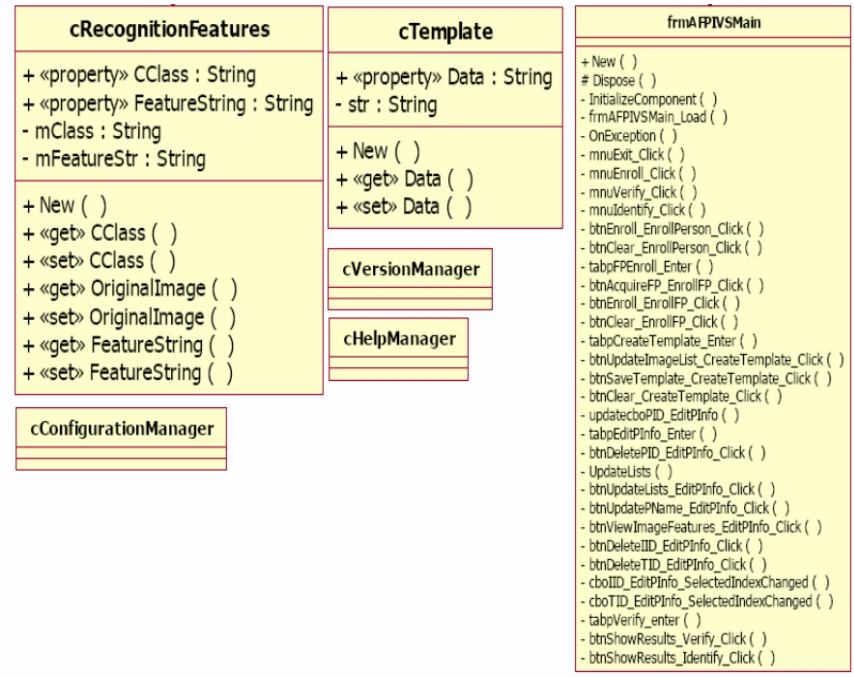

Fig. A-5 Individual Class Diagrams

iii. cDBManager

This provides the interface between the system core and the database and automatically generates queries depending upon the current functional status of the system core.

iv. cEnrollmentManager

This control class is responsible for person and fingerprint enrollment into the system and updating the personal or enrollment information for a subject in the database. It implements the rules for enrollment of a person into the system. This class uses the services provided by the AFPR Algorithm component to extract features from the fingerprint images and then enrolls them into the system database. It also has an ability to acquire fingerprints through a stored image file or a scanner by using the system hardware manager interface depending upon user options.

v. CVerificationManager

This control class is responsible for fingerprint based person verification. It uses the services provided by the AFPR Algorithm component to obtain features from the input fingerprint and carry out fingerprint matching to generate a matching score.

vi. CClassificationManager

This control class is responsible for fingerprint classification by utilizing the services provided by the AFPR Algorithm component.

vii. CIdentificationManager

Identification is carried out as a sequential search by using class, hand type and finger type information to reduce the search space. This control class is responsible for carrying out fingerprint based person identification by coordinating CVerificationManager and CClassificationManager objects.

Database Design

Database design was carried out using ERWIN data modeler. The database diagram for the system is shown in Fig. A-6.

A brief description of each of the tables is given below:

a. Person Table

This table stores the token information of a subject. It has the following fields:

- PID

  This is a unique ID for a person. Two persons in the database cannot have the same PID. This is the primary key for the person table.

- PName

  This is a string that stores person name. It cannot be null and cannot contain numerals.

- ETime

  ETime stores the time and date a person was enrolled into the system.

b. Finger Info Table

This table contains information about the finger type and hand type of a finger enrolled into the system for which fingerprints can be stored during enrollment. Multiple fingers of a person can be enrolled into the database. A single person can have a maximum of ten fingers, 5 belonging to the each of the two hands (Left, Right) with only one finger of each type (little, ring, middle, index, thumb). When a person is deleted/updated in the database, his records from this table are to be deleted/updated accordingly in order to maintain database integrity. This table has the following fields:

- PID

  This is the foreign key from the person table whose existence here implies that unless and until the person has been enrolled to the system, an entry for him/her cannot be created in this table.

- HID

  This is the hand ID, which can be Left, Right or Unknown.

- FID

  This is the Finger ID, which can be Little, Ring, Middle, index, Thumb, or Unknown. .

- Class

  This contains true class information for a finger of a subject in the database. It can be Arch, Tented Arch, Whorl, Twin Loop, Left Loop, Right Loop or Unknown depending upon the class specified or determined.

- ETime

  ETime stores the time and date when an entry was made into this table.

The primary key of this table comprises of {PID, HID, FID}.

c. Image Table

This table contains the fingerprint image in binary format along with the features (e.g. minutiae) extracted from the image. A fingerprint can belong to exactly one finger, however a finger may have many fingerprint images stored in the database. If the record of the finger of a person is deleted/updated, the corresponding entries of this table must also be deleted/updated to maintain database integrity. The various fields of this table are:

- {PID, HID, FID}

  This is actually the foreign key from the finger table whose existence here implies that unless and until the finger has been enrolled to the finger table, an entry for him/her cannot be created in this table.

- IID

  This along with {PID, HID, FID} uniquely identifies a fingerprint image.

- IDATA

  This is the binary format storage of a fingerprint.

- FVDATA

  It contains a serialized version of the features of a fingerprint.

- ETIME

  ETime stores the time and date when an entry was made into this table.

d. Template Table

This table holds the template for a person, which is used to identify a person. A person can have a single template and a template can belong to a single person. The template is based on multiple images, which must belong to the same finger. If the record of the fingerprint of a finger is deleted/updated, the corresponding template is deleted to maintain database integrity. A person, for which a template does not exist can be identified or verified. The constituent fields of this table are:

- {PID, HID, FID,IID}

  This is actually the foreign key from the finger table whose existence here implies that unless and until a fingerprint of the person is present, a template for that person cannot be created.

- TID

  This uniquely identifies the template. This has been kept for future enhancement in which a person can have multiple templates.

- TDATA

  This contains the template data (features) which when compared with the features of an input image, generate a matching score.

- ETime

ETime stores the time and date when an entry was made into this table.

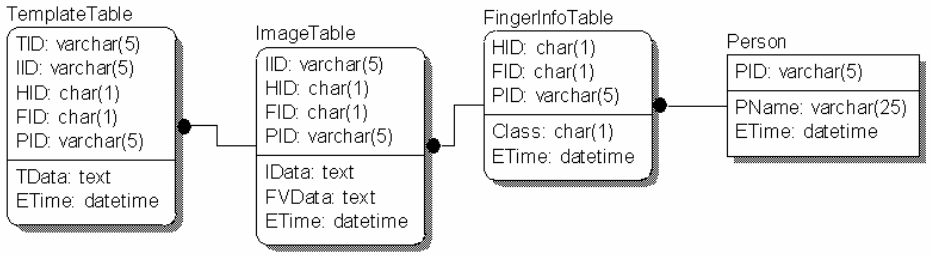

Fig. A-6 System Database Design

User Interface Design

The user interface was formulated by a prototyping approach. The user interface used in the project is tab-based on a single document interface as shown in Fig. A-7.

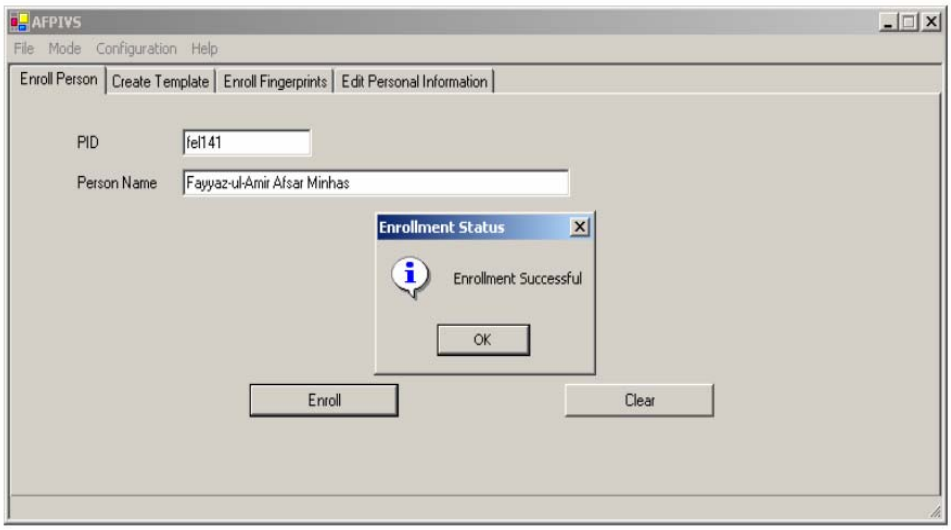

Fig. A-7 System User Interface

System Hardware Interface Design

The URU 4000 fingerprint scanner was used for the acquisition of fingerprints. This scanner was interfaced to the system core through the use of the URU 4000 Platinum SDK Extension Package.

Implementation

The implementation of the system is examined at three levels, which are given below.

Database Implementation

The system database was implemented primarily using the ERWIN modeling tool and its forward engineering wizard was used to generate a SQL listing of the database creation script. This script is used to setup the database on MS SQL Server 2000 or the Microsoft Data Engine (MSDE). MSDE is a limited version of the Microsoft SQL Server. In short, it is the Microsoft SQL Server 2000 database engine without any of the fancy UI tools, and with some limitations in the database size and the number of connections. The MSDE database is free, and can be distributed embedded in your own applications or as a small stand-alone SQL server. It is ideal for small websites and small businesses with less than 25 simultaneous users. The database is limited to 2 GB of data storage space, but you can easily upgrade it to a full Microsoft SQL Server without any limitations. Among the choices are, a standard edition or an enterprise edition. An MSDE database is a good and affordable starting point for any business, organization or even home solutions. The database does not have to be installed on the hard drive on the deployment computer, but it could be running from a CD-ROM. Because of these advantages we have used MSDE for the implementation of our database. Fig. A-8 shows some example entries in the database (by using the OSQL utility).

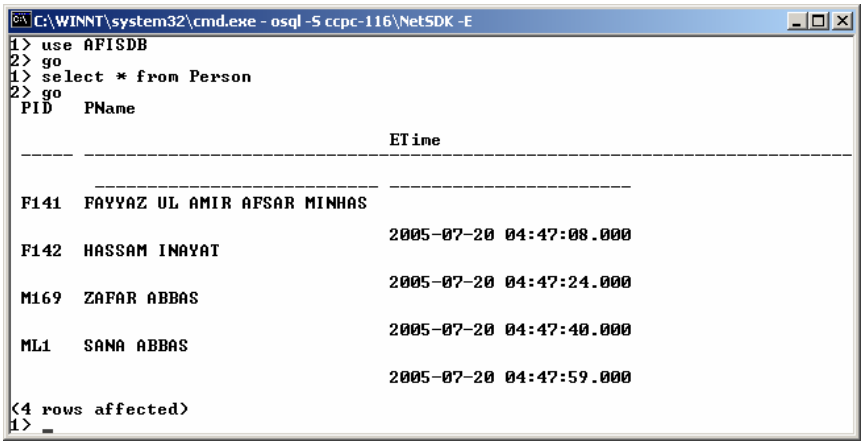

Fig. A-8 OSQL for Database Development

Implementation of the AFPR Algorithm Component

The fingerprint matching system proposed in Section 7.4.2.2 and the classification algorithm proposed in Section 3.8 were implemented using Matlab 7.0. The m-files of these algorithms were used to generate a COM based dll file component using Matlab's COMTOOL. This dll is then used in the development of the system in Visual Studio .NET. The implementation of the AFPR Algorithm component in Matlab also ensures its portability to Linux.

Implementation of the Classes

The classes mentioned earlier were implemented using Visual Basic .NET to generate class libraries. The main advantage of using class libraries is that they offer flexibility and easy update.

Complete System Implementation

The system was implemented in two major builds. In the first build, main focus was at person identification and verification whereas in the second build we added classification to the system. The system was integrated under the .NET environment, which offers the best code optimization, speed and security.

Software Testing

Software testing was carried at multiple levels through out the development of the system. These are:

a. Desk Checking

   Desk checking is aimed at the testing of a module or sub module as it is being coded.

b. Module Level Testing

   For module level testing of different system modules a proper test plan was developed for each module and rigorous testing was carried out at the completion of the development of a module.

c. Integrated Testing

This type of testing is carried out when multiple modules are to be integrated.

d. Stress Testing

Stress testing was carried out for the completed system. The system has been successfully tested for a maximal database size of 2GB (as specified by MSDE) with about 100000 personal records in the database with no significant decrease in response time.

e. Performance Analysis

Performance Analysis was carried out especially for the AFRR Algorithm component in order to ensure the requirements for response time, accuracy and storage capacity.

Deployment & System Requirements

The minimal system requirements for the software are given below:

a. P-III, 1.0GHz with 256 MB RAM (P-IV recommended)
b. Windows 2000/XP
c. .NET Frame work

The deployment process has been automated with the development of a special setup wizard built in VB .NET. This wizard carries out the following major tasks:

a. Setup MSDE the system database

The setup wizard automatically sets up MSDE on the target machine.

b. Install the Matlab Component Runtime (MCR)

The Matlab component runtime is required for operating the AFRR Algorithm component of the system. MCR is a redistributable component, which eliminates the need of the presence of Matlab on the target system in order to execute Matlab code present in the form of components (DLLs and executables).

c. Registering the AFRR Algorithm component assemblies

The AFPR Algorithm assemblies are COM components and must be registered. The installation wizard automatically registers the DLL components using REGSVC.

d. Setting up the System Database

The database is setup by first installing MSDE and then executing the database setup script. Here the user can specify the location where the

database is to be setup, which can be a remote computer. The user can also specify a system level database password for security purposes.

e. Copying the Executable Files

The executable files and assemblies are copied automatically to the location specified by the user for the system setup thus completing the installation process.

Analysis of Overall System Performance

a. Space Requirements

A single feature vector requires about 400Bytes of storage in the database. However if the image is also to be stored in the database then space requirements can go up, depending upon the resolution of the image and the type of compression being used.

b. Response time

For verifying a single fingerprint, the time taken by the system is 0.9s. The enrollment time is 0.4s whereas the time required for classification is 0.75s. All of these response times fulfill the requirement specifications.

c. Accuracy

As the ROC curve shown in Fig. 7-29, the Equal Error Rate for the system at 0% image rejection is ~3.15%. This error rate also fulfills the requirement specifications.

Extensibilities

In terms of software development, the following extensibilities are possible:

a. Integration of a Fingerprint Scanner

A fingerprint scanner can be easily accommodated because of the flexibility in the architecture and design of the software.

b. Customization of the software for a particular application

The software can be easily customized for a particular application (e.g. computer user login control etc.) because of the component-based nature of the system and the separation of the user interface from the application logic.

c. Improvement in storage space requirements

Storage space requirements can be decreased by using an effective compression technique such as WSQ compression.

d. Improvement in response time

   The response time of the system can further be decreased y shifting the low level implementation from Matlab to VC++ or VB.NET.

Summary

The system software has been developed using VB.NET with the image processing techniques implemented in Matlab 7.0. The Matlab code was integrated with VB.NET using Matlab Component Runtime (MCR). The database system used was MSDE. The URU 4000 Fingerprint scanner was also integrated into the system using the scanner SDK. The software fulfills all the functional and nonfunctional requirements mentioned in the SRS of the project.

Fayyaz ul Amir Afsar Minhas was born in Murid, District Chakwal, Pakistan on 28th August 1983. He did his matriculation from District Public School, Chakwal and his FSc. from P.A.F. Intermediate College, Kallar-Kahar. He joined the Pakistan Institute of Engineering and Applied Sciences (PIEAS), Islamabad in 2001 under the degree program for BS in Computer and Information Sciences. His research interests include Image Processing, Artificial Intelligence, Pattern Matching, Biometrics, Processor Scheduling and Distributed Computing.