

Ph.D Thesis

**DEVELOPMENT OF A BIOMETRIC
PERSONAL AUTHENTICATION SYSTEM
BASED ON FINGERPRINT AND SPEECH**

*Submitted to the
Cochin University of Science and Technology
in partial fulfilment of the
requirements for the degree of Doctor of Philosophy
under the faculty of Technology*

by

Praveen N.

under the supervision of
Prof. Tessamma Thomas



Department of Electronics
Cochin University of Science and Technology
Cochin
February, 2013

**DEVELOPMENT OF A BIOMETRIC PERSONAL
AUTHENTICATION SYSTEM BASED ON FINGERPRINT AND
SPEECH**

Author:

Praveen N.
Research Fellow
Audio and Image Research Laboratory
Department of Electronics
Cochin University of Science and Technology
Kochi, 682022, India
Email: Praveen.naniyat@gmail.com

Research Advisor:

Prof. Tessamma Thomas
Professor
Department of Electronics
Cochin University of Science and Technology
Kochi, 682022, India
Email: tess@cusat.ac.in

Department of Electronics
Cochin University of Science and Technology
Kochi, 682022, India
www.doe.cusat.edu
February, 2013

Cover Design: Suraj Kamal, CUCENTOL, CUSAT

CERTIFICATE

Certified that the research work presented in this thesis entitled **Development of a Biometric Personal Authentication System Based on Fingerprint and Speech** is a bonafide record of the research work carried out by Mr. Praveen N under my supervision in Department of Electronics, Cochin University of Science and Technology, Cochin-22. The results presented in this thesis or part of it has not been included in any other thesis submitted for the award of any other degree.

Cochin-22
27-02-2013

Prof. Tessamma Thomas
(Supervising Guide)
Department of Electronics
Cochin University of Science and Technology
Cochin 682022

DECLARATION

I declare that the work presented in the thesis entitled **Development of a Bometric Personal Authentication System Based on Fingerprint and Speech** is based on the original work done by me under the guidance and supervision of Prof. Tessamma Thomas, Professor, Department of Electronics, Cochin University of Science and Technology, Cochin-22, India and has not been included in any other thesis submitted previously for the award of any degree.

Praveen N.

Research Fellow

Department of Electronics

Cochin University of Science and Technology

Cochin 682022

Acknowledgments

There are many personalities I am obliged to complete this PhD work. Without their timely help and support this work would not have been materialised.

I would like thank my supervising guide, Prof. Tessamma Thomas for her guidance, creative ideas and insightful advice throughout my doctoral research. Without her constant support, this thesis would not be possible.

I would like to thank Prof. C K Aanandan, Professor and Head, Department of Electronics, for his encouragement and support rendered in submitting this work. I am grateful to Prof. K Vasudevan, Dean of Technology and former Head of the Department of Electronics, Prof. P R S Pillai, former Head of the Department of Electronics, Prof. P. Mohanan, Professor, Department of Electronics, Dr. James Kurian and Dr. Supriya M H, Associate professors, Department of Electronics for their support and motivation.

I take this opportunity to express my heartfelt thanks to my fellow researchers Dr. Deepa Sanker, Deepa J, Nobert Thomas Pallath, Reji A P, Anantharesmi S, Anu Sabareesh, Sethunadh and Tina P G for their valuable suggestions and support.

I am grateful to Dr. N. Gopukumar, Deputy Secretary, UGC SWRO for the timely allotment of FIP to complete my Ph.D. work. I am thankful to Principal and staff, N S S College, Rajakumari, Idukki for their help and support for completing this work. I am obliged to the support and motivation extended by my son Nikhil and my wife Dr. Saritha to complete this thesis. I am grateful to my father and mother, my father-in-law and mother-in law, my brother and in-laws for their timely help and support throughout the Ph.D. work. I take this opportunity to thank Dinesh R, Research Fellow, CREMA, Department of Electronics, Lindo Ouseph, Research Fellow, RCS Lab, and all the research scholars of Department of Electronics for their motivation and comments.

Abstract

Biometrics deals with the physiological and behavioral characteristics of an individual to establish identity. Fingerprint based authentication is the most advanced biometric authentication technology. The minutiae based fingerprint identification method offer reasonable identification rate. The feature minutiae map consists of about 70-100 minutia points and matching accuracy is dropping down while the size of database is growing up. Hence it is inevitable to make the size of the fingerprint feature code to be as smaller as possible so that identification may be much easier. In this research, a novel global singularity based fingerprint representation is proposed. Fingerprint baseline, which is the line between distal and intermediate phalangeal joint line in the fingerprint, is taken as the reference line. A polygon is formed with the singularities and the fingerprint baseline. The feature vectors are the polygonal angle, sides, area, type and the ridge counts in between the singularities. 100% recognition rate is achieved in this method. The method is compared with the conventional minutiae based recognition method in terms of computation time, receiver operator characteristics (ROC) and the feature vector length.

Speech is a behavioural biometric modality and can be used for identification of a speaker. In this work, MFCC of text dependant speeches are computed and clustered using k -means algorithm. A backpropagation based Artificial Neural Network is trained to identify the clustered speech code. The performance of the neural network classifier is compared with the VQ based Euclidean minimum classifier.

Biometric systems that use a single modality are usually affected by problems like noisy sensor data, non-universality and/or lack of distinctiveness of the biometric trait, unacceptable error rates, and spoof attacks. Multifinger feature level fusion based fingerprint recognition is developed and the performances are measured in terms of the ROC curve. Score level fusion of fingerprint and speech based recognition system is done and 100% accuracy is achieved for a considerable range of matching threshold.

Table of Contents

Chapter 1	13
Introduction	13
1.1 <i>Biometrics</i>	15
1.2 <i>Biometric characteristics</i>	16
1.2.1 Commonly used biometric characteristics	17
1.3 <i>Performance Evaluation of a biometric system</i>	19
1.4 <i>Fingerprint based personal authentication techniques</i>	22
1.5 <i>Speech based personal authentication techniques</i>	22
1.6 <i>Multibiometric fusion</i>	23
1.7 <i>Motivation and objectives of the work</i>	24
1.8 <i>Organisation of the thesis</i>	25
Chapter 2	29
Literature Survey	29
2.1 <i>Introduction</i>	31
2.2 <i>History of Biometrics</i>	31
2.3 <i>Fingerprint Recognition System</i>	33
2.3.1 Minutia based method	33
2.3.1.1 Preprocessing	33
2.3.1.2 Image Enhancement	33
2.3.1.3 Directional Field Estimation	36
2.3.1.4 Image Segmentation	36
2.3.1.5 Binarization and Thinning	38
2.3.1.6 Minutia Extraction	39
2.3.1.7 Fingerprint Matching	40
2.4 <i>Speaker recognition</i>	44
2.5 <i>Multimodal Biometrics</i>	47

Chapter 3	50
Fingerprint Recognition Techniques	50
3.1 <i>Introduction</i>	<i>52</i>
3.2 <i>Fingerprint Image Processing and Feature Extraction.....</i>	<i>52</i>
3.2.1 Ridge Orientation Estimation	54
3.2.1.1 Orientation Estimation by gradient based method:.....	54
3.2.2 Estimation of Local Ridge Frequency	57
3.2.3 Fingerprint Image Segmentation	59
3.2.4 Singularity and Core Detection	62
3.2.4.1 Singularity detection using Poincaré index:.....	63
3.2.4.2 Singularity detection based on local characteristics of orientation image	65
3.2.4.3 Core detection and fingerprint registration	66
3.2.5 Fingerprint Enhancement	68
3.2.5.1 Pixel-wise enhancement:	69
3.2.5.2 Contextual filtering:.....	70
3.2.5.2.1 Gabor filter based enhancement	72
3.2.5.2.2 FFT based enhancement.....	74
3.2.6 Minutiae Detection	74
3.2.6.1 Binarisation methods	75
3.2.6.2 Thinning of Binarised image	76
3.3 <i>Fingerprint Matching</i>	<i>78</i>
3.3.1 Correlation-Based Techniques	78
3.3.2 Minutiae-Based Matching Method.....	79
3.3.2.1 Problem formulation.....	79
3.3.2.2 Similarity score.....	81
3.4 <i>Conclusion.....</i>	<i>81</i>
Chapter 4	82
Development of a Novel Global Singularity Feature Based Fingerprint Recognition System	82
4.1 <i>Introduction</i>	<i>84</i>
4.2 <i>Minutiae based fingerprint recognition technique.....</i>	<i>84</i>

4.2.1	Fingerprint Normalization	85
4.2.2	Ridge orientation estimation.....	85
4.2.3	Estimation of Local Ridge Frequency	87
4.2.4	Fingerprint Image Segmentation	91
4.2.5	Fingerprint Enhancement	92
4.2.5.1	Gabor filter based fingerprint enhancement:	93
4.2.6	Fingerprint Thinning	94
4.2.7	Minutiae Detection	96
4.2.8	Core Detection and Feature Vector formation.....	96
4.2.8.1	Core Detection	96
4.2.8.2	Feature vector formation.....	97
4.2.9	Matching score determination	99
4.2.10	Performance estimation parameters for algorithm, FAR and FRR	100
4.2.11	Implementation of the Algorithm.....	100
4.2.11.1	Database used	100
4.2.11.2	Implementation.....	101
4.2.11.3	Results and discussion	102
4.3	<i>Development of a novel global singularity based fingerprint recognition</i>	<i>103</i>
4.3.1	Development of the method	104
4.3.1.1	Directional Field Estimation and strength computation	106
4.3.1.2	Singularity detection and fingerprint classification scheme	107
4.3.1.3	Baseline detection and feature vector formation	109
4.3.1.3.1	Baseline detection	109
4.3.1.3.2	Definition of the novel fingerprint structure and Feature vector formation	111
4.3.1.4	Fingerprint Matching Score	114
4.3.1.5	Implementation.....	115
4.3.1.5.1	Database used.....	115
4.3.1.5.2	Implementation	115
4.3.1.6	Results and discussions	116
4.4	<i>Conclusion.....</i>	<i>123</i>
Chapter 5		125
Speaker Recognition Techniques		125
5.1	<i>Introduction</i>	<i>127</i>

5.2	<i>Speaker Recognition</i>	127
5.2.1	Classification of speaker recognition	128
5.2.2	Basic Structure of Speaker Recognition System	130
5.2.2.1	Feature selection	130
5.2.2.2	Speech Features	131
5.2.3	MFCC based parametric representation	132
5.3	<i>Speaker Recognition Algorithms</i>	137
5.3.1	Minimum-Distance Classifier	138
5.3.2	Vector Quantization (VQ)	138
5.3.3	Gaussian Mixture Model (GMM)	141
5.4	<i>Data Clustering</i>	142
5.5	<i>Artificial Neural Network as a classifier</i>	145
5.5.1	Backpropagation ANNs	150
5.5.2	Some general issues in ANN development	153
5.6	<i>Summary</i>	159
Chapter 6		161
Development of a Text Dependant Speaker Recognition using MFCC Features and BPANN		161
6.1	<i>Introduction</i>	163
6.2	<i>Feature Extraction</i>	165
6.2.1	Speech recording	165
6.2.2	Speech Normalisation	166
6.2.3	Pre-processing	166
6.2.4	Voiced region extraction	167
6.2.5	MFCC Computation	168
6.2.6	Data clustering using k-means method	169
6.3	<i>Neural Network Design and Training</i>	171
6.3.1	Network Architecture	171
6.3.2	Data Normalisation	171
6.3.3	ANN training	172
6.4	<i>Implementation</i>	174
6.4.1	Database used	174

6.4.2	Implementation.....	174
6.4.3	Results and Discussion.....	175
6.5	<i>Conclusion</i>	179
Chapter 7	181
Multimodal biometric fusion	181
7.1	<i>Introduction</i>	183
7.2	<i>Information Fusion in Multimodal Biometrics</i>	184
7.2.1	Pre-mapping fusion: sensor data level.....	184
7.2.2	Pre-mapping fusion: feature level.....	185
7.2.3	Midst-mapping fusion.....	186
7.2.4	Post-mapping fusion: decision fusion.....	186
7.2.4.1	Majority Voting.....	186
7.2.4.2	Ranked list combination.....	187
7.2.4.3	AND fusion.....	187
7.2.4.4	OR fusion.....	187
7.2.4.5	Post mapping fusion.....	187
7.2.4.5.1	Weighted summation fusion.....	188
7.2.4.5.2	Weighted product fusion.....	188
7.2.4.5.3	Post-classifier.....	189
7.2.5	Hybrid fusion.....	190
7.3	<i>Development of a Multifinger Feature Level based fingerprint recognition system</i> 190	
7.3.1	Feature vector formation by fusion.....	192
7.3.2	Fingerprint matching score.....	193
7.3.3	Implementation of the Algorithm.....	193
7.3.3.1	Database used.....	194
7.3.3.2	Implementation.....	194
7.3.4	Results and Discussion.....	194
7.4	<i>Development of Multimodal fingerprint and speech Score level fusion based personal authentication using fingerprint and speech features</i>	199
7.4.1	Implementation.....	200
7.4.1.1	Fingerprint verification.....	200
7.4.1.2	Text-dependent Speaker verification.....	200

7.4.1.3	Combining the two modalities	201
7.4.1.4	Combining using Sum rule	201
7.4.1.5	Combining using Product rule	202
7.5	<i>Conclusion</i>	205
Chapter 8	207
Conclusion and Future Scope	207
8.1	<i>Conclusion</i>	207
8.2	<i>Future Scope</i>	209
References	211
Publications	232
Appendix A	233
A.1	<i>Flowchart for Fingerprint Feature Extraction</i>	233
A.2	<i>Fingerprint Matching</i>	241

Chapter 1

Introduction

1.1 Biometrics

Biometrics deals with the physiological and behavioral characteristics, including fingerprint, face, hand/finger geometry, iris, retina, signature, gait, voice pattern, ear, hand vein, DNA information of an individual to establish identity [Ross, 2006]. These characteristics are referred to as traits, indicators, identifiers or modalities of a human being to recognise or identify him. While biometric systems have their own limitations [Gorman, 2002] they have an advantage over traditional security methods in that they cannot be easily stolen or shared. Besides the advantage of security, biometric systems also enhance user convenience by alleviating the need to design and remember passwords. Biometrics has been used for solving crime related issues since 19th century. It was Alphonse Bertillon who first proposed and materialised the idea of using biometric traits for criminal investigations [Rhodes, 1956] During the 19th century, bureaucratic efforts were made to maintain organised criminal records that extended to the whole state/country [Torpey, 2001]. In the beginning of the biometric based authentication technology, body marks or birthmarks were used to index the identity of convicts [Troup, 1894]. Fingerprints have been identified as a personal identifier even in the pre-historic period as per several archeological evidences [Xiang, 1989; Laufer, 1912; Berry, 1991]. Human fingerprints have been discovered on a large number of archeological artifacts and historical items. It was used in India by British administrators as a workable identification system for combating fraud through impersonation, resolving disputed identities in civil legal disputes over land deeds or contract, monitoring the movement of criminal groups and maintaining criminal history of persons convicted of crimes [Sengoopta, 2002]. It was in 1858; Sir William Herschel formally used fingerprint deeds for signing contracts. This helped bring fingerprints to the attention of British scientists and bureaucrats [Simon, 2004]. Herschel was actually inspired by Bengali practice of signing documents with a fingertip dabbed in ink called *tip sahi* [Chatterjee, 1988; Herschel, 1916].

The fast improvements in the computer technology have led to the development of Automatic Fingerprint Identification systems (AFIS) over the past

decades. Most of the governments enforce biometric based authentication for their governmental & financial dealings, security access to national important areas, online transactions etc... The recent biometric based Aadhar project implemented by Unique Identification Authority of India is aimed as an initiative that would provide identification for each resident across the country and would be used primarily as the basis for efficient delivery of welfare services [UID, 2013].

Biometrics have wide range of application areas including sharing network login, electronics data security, e-commerce, internet access, ATM or credit card use, physical access control, personal phone, driving licence, passport, corpse identification, criminal investigation, parenthood determination etc. In some applications biometrics may be used to supplement ID card and passwords thereby imparting an additional level of security. Every biometric system is essentially a pattern recognition system that acquires biometric data from an individual, extracts a salient feature set from the data and matches this feature against the template feature set stored in the database. A decision is taken based on the match score obtained to determine whether the person is genuine or not.

1.2 Biometric characteristics

Jain et al. [Jain, 1999] have identified seven factors that determine the suitability of physical or behavioural trait to be used in a biometric application.

Universality: Every individual accessing the application should possess the trait.

Uniqueness: There should be sufficient difference across individuals for the trait.

Permanence: The biometric trait of an individual should be sufficiently invariant over a considerable period of time with respect to the matching algorithm.

Measurability: The trait should be sampled without much inconvenience to the individual.

Performance: The recognition accuracy and the resources required to achieve that accuracy should meet the constraints imposed by the application.

Acceptability: The extent to which people are willing to accept a particular biometric identifier in their daily lives.

Circumvention: This reflects how easy it is to fool the system by imitation by artefacts.

Every practical biometric system should have acceptable recognition accuracy and speed with reasonable resource requirements, wide range acceptability, minimal invasive to users and should be robust to fraudulent activities.

1.2.1 Commonly used biometric characteristics

- 1. Face:** Face features are the most common features used by humans to recognise one another. The main features include: i) location and shape of facial attributes such as the eyes, eyebrows, nose, lips, and chin and their spatial relationships ii) the global analysis of face image that represents a face as a weighted combination of a number of canonical faces. Often face recognition required a fixed and simple background with controlled illumination. Also the system need extra intelligence or computational power to match face images captured from two different views, under different illumination conditions and at different instance of time [Philips, 2003].
- 2. Fingerprint:** Fingerprints have the credits of a proven biometric authentication method over decades. The matching accuracy using fingerprints has been shown to be very high [Wilson, 2004]. A fingerprint is a well-defined oriented pattern on the surface of the fingertip whose formation is determined during the first seven months of foetal development [Jain, 2008]. Fingerprints have the advantages of high permanence over years. The accuracy of currently available fingerprint recognition system is adequate for an authentication system in several applications, but as the size of database increases, matching as well as

feature set size becomes a challenging problem for fingerprint researchers. Fingerprint based authentication holds a lion share in the biometric personal authentication technology market [Maltoni, 2005].

3. **Hand geometry:** The number of measurements taken from human hand including its shape, size of palm, and the length and width of fingers [Zunkel, 1999]. The techniques used in hand geometry based systems are very simple, relatively easy to use, and inexpensive. Hand geometry may not be invariant during the growth period of a child and also individuals' jewellery or limitations in dexterity may pose challenges to hand geometry based systems.
4. **Palmprint:** Similar to fingerprints, palmprints also contain pattern or ridges and valleys. The palm area is much larger than the fingerprint and hence has got more features than fingerprints. Human palmprint also contain additional distinctive features such as principal lines and wrinkles that can be captured even with a low resolution scanner.
5. **Iris:** The iris based personal authentication offers promising accuracy and speed and supports the feasibility of large-scale identification system. The iris is the annular region of the eye bounded by the pupil in the sclera on either side. The visual texture is formed during the foetal development and stabilizes during the first two years of life. The complex iris pattern carries distinctive information useful for personal recognition [Daugman, 2004]. Each iris is distinctive and even the irises of identical twins are different.
6. **Signature:** Signature can be taken as a characteristic of an individual. It requires an effort on the part of the user and is a behavioral biometric that changes over a period of time which is influenced by physical and emotional conditions of the signatories. Signatures of some people vary substantially: even successive impressions of their signature are significantly different. Also professional forgers may be able to reproduce signatures that fool the signature verification system [Harrison, 1981].
7. **Keystroke:** The way in which each person types on a keyboard is considered to be distinctive. This biometric is not expected to be unique

to each individual but it may be expected to offer sufficient discriminatory information to permit identity verification [Monrose, 1997]. Keystroke dynamics is a behavioral biometric which observe large intra-class variations in person's typing patterns due to changes in emotional state; position of the user with respect to the keyboard, type of keyboard used etc.

8. **Voice:** Voice is a combination of physical and behavioral biometric characteristics [Grother, 2004]. The physical characteristics of an individual's voice are based on the shape and size of the vocal tracts, mouth, nasal cavities and lips that are used in the synthesis of the sound. These physical characteristics of human are invariant for an individual, but the behavioral characteristics of human speech changes over time due to age, medical conditions, emotional state, etc.
9. **Gait:** The manner in which a person walks is taken as a biometric trait. This trait is appropriate in surveillance scenarios where the identity of an individual can be secretly established. Gait recognition algorithm extracts the human silhouette in order to derive the spatio-temporal attributes of a moving individual. Gait-based system also offers the possibility of tracking an individual over an extended period of time. There are several factors that affects the gait which include the choice of footwear, nature of clothing, walking surface, injuries on legs etc.

1.3 Performance Evaluation of a biometric system

In a biometric authentication system, the feature sets are matched with the stored feature template by means of distance or similarity scores. A biometric system rarely encounters two samples of a user's biometric trait that result in exactly the same feature set due to imperfect sensing conditions, physiological conditions of the users, changes in ambient conditions and variations in the user's interaction with the sensor. The variability observed in the biometric feature set of an individual is referred to as intra-class variation and the variability between features sets originating from two different individuals is known as inter-class

variation and usually a useful set exhibits small intra-class variation and large inter-class variation.

Similarity score is the metric used to measure the similarity between two biometric feature sets. The similarity score of two samples of the same biometric trait of a user is known as a genuine or authentic score. Similarly the score is known as an impostor score if it involves comparing two biometric samples originating from different users. An impostor score that exceeds the threshold η results in a false accept or false match, while a genuine score that falls below the threshold η results in a false reject or a false non-match. The *False Accept Rate (FAR)* or *False Match Rate (FMR)* of a biometric system can therefore be defined as the fraction of impostor scores exceeding the threshold, η . The *False Reject Rate (FRR)* or the *False Non-match Rate (FNMR)* of a system may be defined as the fraction of genuine scores falling below the threshold η . The *Genuine Accept Rate (GAR)* is the fraction of genuine scores exceeding the threshold, η . i.e.,

$$GAR=1-FRR \quad (1.1)$$

The FAR and FRR at various values of η can be plotted using a Detection Error Trade-off (DET) curve [Martin, 1997] at various thresholds. When a linear, logarithmic or semi logarithmic scale is used to plot these error rates then the resulting graph is known as the Receiver Operating Characteristic (ROC) curve [Egan, 1975]. The performance of a biometric system may also be summarised using other single-valued measures such as the *Equal Error Rate (EER)* and the d-prime value. The EER refers to that point in a DET curve where the FAR equals the FRR. Lower the EER value, better the system performance. ZeroFNMR is the lowest FMR at which no false non-matches occur and ZeroFMR is the lowest FNMR at which no false matches occur. FAR, FRR, ZeroFNMR, ZeroFMR and EER are shown in the fig. 1.1. The d-prime value (d') measures the separation between the mean values of the genuine and impostor probability distributions, in standard deviation units and is defined as,

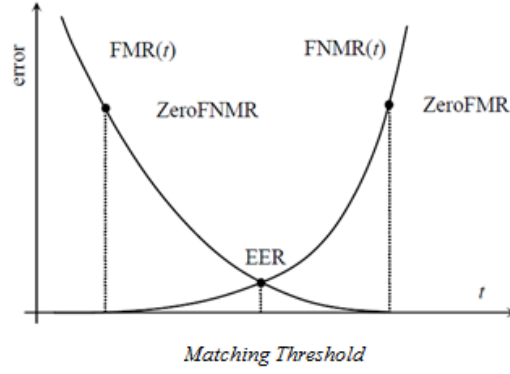


Fig. 1.1 FAR (FMR) and FRR(FNMR) curves, where the points corresponds to EER, ZeroFNMR and ZeroFMR are shown

$$d' = \frac{\sqrt{2}|\mu_{genuine} - \mu_{impostor}|}{\sqrt{\sigma_{genuine}^2 - \sigma_{impostor}^2}}, \quad (1.2)$$

where the μ 's and σ 's are the means and standard deviations, respectively, of the genuine and impostor distributions. Another single-valued measure known as F-Ratio [Poh, 2005] which is defined as ,

$$\text{F-ratio} = \frac{\mu_{genuine} - \mu_{impostor}}{\sigma_{genuine} + \sigma_{impostor}} \quad (1.3)$$

If the genuine and impostor distributions are Gausssian, then the EER and F-ratio are related according to the following expression:

$$EER = \frac{1}{2} - \frac{1}{2} \operatorname{erf}\left(\frac{\text{F-ratio}}{\sqrt{2}}\right) \quad (1.4)$$

where,

$$\operatorname{erf}(x) = \frac{2}{\pi} \int_0^x e^{-t^2} dt \quad (1.5)$$

1.4 Fingerprint based personal authentication techniques

A fingerprint is a well oriented pattern of interleaved ridges and valleys. A ridge is represented in a fingerprint image as a dark region and a valley is represented as a white region. Typically, ridges have a thickness of 100 μ m to 300 μ m and the ridge-valley has a thickness of about 500 μ m. Ridges and valleys often run in parallel, sometimes they bifurcate and sometimes they terminate. The various discontinuities and their small details are referred to as 'minutiae. A ridge can suddenly come to an end (termination) or can divide into two ridges (bifurcation). The minutia based fingerprint identification technique takes a major share in the fingerprint authentication technology. The steps carried out for feature extraction are: Ridge Orientation Estimation, Estimation of Local Ridge Frequency, Fingerprint Image Segmentation, Singularity and Core Detection, Fingerprint Enhancement, Minutiae Detection, Matching. The minutiae based identification method offer reasonable identification rate. Also this method is well accepted around the forensic and criminal investigation bodies of most of the countries. The feature minutiae map consists of about 70-100 minutia points and matching accuracy is dropping down while the size of database is growing up. Hence it is inevitable to make the size of the fingerprint feature code to be as small as possible, so that identification may be much easier.

1.5 Speech based personal authentication techniques

Speech is the most accessible biometric trait as no extra acquisition device or transmission system is needed due to the recent advancements in the mobile phone technology and VoIP (Voice over IP networks). This fact gives speech an advantage over other biometric traits, especially when remote users or systems are taken into account. Speech is a behavioural biometric modality and can be used for identification of a speaker.

Speaker recognition can be divided into text-dependant and text independent methods. In the text dependent method, the speaker requires to

provide utterances that are the same text, for training and recognition. The security level of password based systems can then be enhanced by requiring knowledge of the password, and also requiring the true owner of the password to utter it. To enhance the security, text-dependent systems can prompt to speak random texts, unexpected to the caller, which cannot be easily fabricated by an impostor.

In the text independent method, the speaker is not compelled to speak a specific text or sentence. The main applications of text-independent methods are speaker detection and forensic speaker recognition.

The basic step in the design of an automatic speaker recognition system is the reliable extraction of features and tokens that contain identifying information of interest. The short-term feature vectors include spectral information energy, pitch and mid-term and long-term tokens as phonemes, syllables and words. In text dependent system, the information present at different levels of the speech signal, like glottal excitation, spectral and supra-segmental features can be used to detect a speaker's spectral content of the speech signal, which is determined by the physical configuration and dynamics of the vocal tract. This is the most widely used information for speaker identity. Mel-Frequency Cepstral Coefficients (MFCC) is usually calculated for the temporal sequence for a window of 20-40 ms of speech. The speaker recognition is thus a problem of comparing a sequence of MFCC vectors to a model of the user. The matching is usually done using a template based method via Dynamic Time Warping (DTW) [Furui, 1981],[Subramanian, 2006] and by using statistical method via Hidden Markov Model (HMM) [Rabiner, 1989]. In the text independent system, short-term spectral analysis is used to model the speaker specificities, and using Vector Quantization (VQ) and ergodic HMM's (EHMM) speaker identification is possible.

1.6 Multibiometric fusion

Biometric systems that use a single modality are usually affected by problems like noisy sensor data, non-universality and/or lack of distinctiveness of the biometric trait, unacceptable error rates, and spoof attacks. A multibiometric

system improves the performance of the system by considering several traits from different scores. Various levels of fusion are possible in a multibiometric system that uses different biometric traits: fusion at the features extraction level, matching score level or decision level. Feature level fusion is quite difficult to consolidate as the feature sets used by different modalities may either be inaccessible or incompatible. Fusion at the decision level is too rigid since only a limited amount of information is available at this level. Therefore integration of the matching score level is generally preferred due to the ease of accessing and combining matching scores.

1.7 Motivation and objectives of the work

Fingerprint recognition systems have got impressive performance and also have the attraction as an inexpensive, minimum invasive, and simple biometric authentication technique. The advancement in the sensor technology, processor power, and memory capacity make the technology more acceptable in the person recognition arena. There are still a number of challenges in designing completely automatic and reliable fingerprint recognition systems, especially in the template size reduction, enhancement and representation of poor quality images. As the terrorist activities and identity frauds are alarmingly increasing, biometric based authentication systems are the most reliable technique for personal authentication. Current research in biometric authentication is directed towards biometric fusion, where multiple sources of information are used to improve the recognition accuracy. The modalities that are rich in discriminatory information, robust matching methods and cost-effective and computationally inexpensive technologies are the current research interest.

The objectives of the work presented in this thesis are:

1. To develop a minimum feature based fingerprint image representation and to use it to develop a fingerprint based personal authentication system.

2. To develop a speech based behavioral biometric modality based personal authentication system
3. To explore the possibility of combining the fingerprint biometric with other biometric parameters for enhancing the security of the system. To be precise, to develop a multimodal system with the combination of fingerprint and speech modalities is the final goal of this research.

In short, this research aims to develop a personal authentication system based on multimodalities to get a 100% recognition rate.

1.8 Organisation of the thesis

The thesis is organised as follows:

Chapter 1: Introduction

Chapter 1 gives the basic overall introduction of the thesis. In this chapter biometric technology is introduced. Various biometric modalities, biometric characteristics and performance evaluation factors are discussed. The motivation and objectives of the thesis are also discussed.

Chapter 2: Literature Survey:

A survey of the previous works done in the field of biometrics; especially in the field of fingerprint based authentication techniques and speaker recognition are discussed and presented here. Different methods relevant to the present work are also discussed.

Chapter 3: Fingerprint Recognition Techniques

This chapter explains various fingerprint authentication techniques. The steps carried out for feature extraction of fingerprints are discussed. The theory behind fingerprint orientation estimation, estimation of ridge frequency, fingerprint image segmentation, various methods for singularity and core detection,

fingerprint enhancement methods and minutiae detection are discussed. Various techniques for fingerprint matching are also mentioned.

Chapter 4: Development of a novel global singularity feature based fingerprint recognition technique

In this chapter, a novel global singularity feature based fingerprint recognition technique is explained. The methodologies developed for feature extraction and matching are discussed. To compare its performance with the state-of-art recognition system, the minutiae based fingerprint recognition system is implemented and presented first. A discussion on the performance of the system compared to the conventional state-of-the art minutiae based fingerprint recognition system is also presented.

Chapter 5: Speaker Recognition Techniques:

Various biological and signal processing aspects of speech are discussed in this chapter. Speech feature characteristics are explained and various spectral speech characteristics are also presented. Speaker recognition and various speaker recognition techniques are described in this chapter. Also data clustering, artificial neural networks and their design aspects are also presented in this chapter.

Chapter 6: Development of a Text Dependant speaker recognition system using MFCC features and BPANN

In this chapter a text independent speaker recognition method is developed. The steps in computation of spectral characteristic, the MFCC of speaker, are discussed. Clustering of the MFCCs is explained. The methodologies adopted for the designing and development of a successful artificial neural network based on backpropagation algorithm is explained. A discussion on the performance of the developed system, measured using FAR, FRR and EER parameters ends the chapter

Chapter 7: Multimodal Biometric Fusion

This chapter discusses various aspects of multimodal biometric fusion. Development of multifinger feature level fusion based authentication system is explained. The performance of the system measured is also given. Various fusion techniques in the score level fusion of fingerprint and speech modalities and the development of a score level fusion based personal authentication system are discussed. The performance of the system is given in terms of FAR, FRR and EER plots.

Chapter 8: Conclusions and future scope of work

Chapter 8 concludes the thesis with the future work directions.

Thesis also includes an appendix and the bibliography and list of publications by the author in the related field.

Chapter 2

Literature Survey

2.1 Introduction

In this chapter a review of the literature on fingerprint based and speech based authentication techniques are presented. The chapter discloses the pioneer work carried out in the field of fingerprint and speech as biometric. Also the recent trends in multimodal biometric techniques have also been included in this chapter.

2.2 History of Biometrics

Biometrics deals with the physiological and behavioural traits of a human being to recognise or identify him.

Biometric based personal authentication methods have been used for solving crime related issues since 19th century. It was Alphonse Bertillon, who first proposed and materialised body marks for criminal investigations [Rhodes, 1956]. Organised criminal records were maintained in UK based on body marks or birth marks to index the identity of convicts [Torpey , 2001]. Fingerprints have been identified as a personal identification even in the prehistoric period as per archaeological evidences [Berry J, 1991, Laufer B 1912, Xiang-Xin et.al. 1989]. In India British administrators used a workable identification system for combatting fraud through impersonation, land deeds or contracts, monitoring criminal groups and maintaining criminal history of persons convicted of crimes [Sengoopta, 1912]. In 1858, Sir William Herchel, an administrator of British India formally used fingerprinted deeds for signing contracts which attracted British scientists and bureaucrats. Signing documents with fingertip dabbed in ink was a Bengali practice called *tip sahi* [Chatterjee. 1988, Herschel, 1880].

In 1684 English plant morphologist, Nehemiah Grew published the first scientific paper reporting his systematic study on the ridge, furrow, and pore structure in fingerprints [Grew, 1684, Lee and Gaensslen, 2001]. The first literature on fingerprint based identification was by Henry Faulds, a Scottish physician, on 'Nature' describes the fingerprint patterns on ancient Japanese ceramics and proposed that their use be investigated for criminal identification [Faulds H, 1912].

Faulds also described some brief research establishing that Gibraltar monkeys and the various human races all shared the same basic pattern structure. In 1788 a detailed description of the anatomical formations of fingerprints was made by Meyer [Moenssens, 1971] in which a number of fingerprint ridge characteristics were identified and characterized. In 1809, Thomas Bewick, a British engraver and ornithologist, began to use his fingerprint as his trade mark and is considered as the one of the important milestones in the scientific study of fingerprint recognition. In his 1823 thesis titled “Commentary on the Physiological Examination of the Organs of Vision and the Cutaneous System”, Dr. Johannes E. Purkinje [1787–1869], professor at the University of Breslau in Germany, classified fingerprint patterns into nine categories and gave each a name

In a letter dated February 16, 1880, to the famed naturalist Charles Darwin, Faulds wrote that friction ridges were unique and classifiable, and alluded to their permanence [Lambourne, 1984]. In October 1880, Faulds submitted an article for publication to the journal *Nature* in order to inform other researchers of his findings [Faulds, 1880]. As the author of the first book on fingerprints [Finger Prints, 1892], Sir Francis Galton established that friction ridge skin was unique and persistent. Because Galton was the first to define and name specific print minutiae, the minutiae became known as Galton details. In 1899 Sir Edward Henry, established the well-known Henry System of fingerprint classification [Henry, 1934]

In the early twentieth century, fingerprint recognition was formally accepted as a valid personal identification method and become a standard routine in forensics [Lee and Gaensslen, 2001]. FBI fingerprint division was set up in 1924 with a database of 810,000 fingerprint cards [FBI, 1984]. In the early stage, automation to fingerprint identification involved the use of IBM punch-card sorters the card sorter would produce a set of cards fitting the given parameters, and operators could then retrieve the matching fingerprint cards. This would somewhat streamline the manual searching process, but not much [Bridges B C, 1942, Harling M, 1996, Stock R M, 1987]. John Fitzmaurice of Baird-Atomics and Joseph Wegstein and Raymond Moore of the U S National Bureau of Standards conducted experiments with optical recognition of fingerprint patterns during the year 1963 [Reed B, 1981, Stock R M, 1977]

FBI fingerprint database now stands well over 200 million and hence Automatic Fingerprint Identification System was inevitable. As database increases, efficient algorithms have to be formulated to reduce the template size as well as search time. AFIS technology has now rapidly grown beyond forensic applications into civilian applications [Maltoni, 2003]. Holographic imaging based research were carried out by General Electric [Horvath 1967], McDonnell Douglas, Sperry Rand and the KMS Technology Centre in the late 1960s and early 1970s. But the system expensive and exceedingly sensitive to inevitable noise contained even in inked not mention latent fingerprint impressions

In 1972, FBI installed a prototype AFIS using a fingerprint scanner. The FBI began testing automatic searching in 1979, and automated searches became routine by 1983 [Stock, 1987] AFIS stored individual fingerprints as relational data between minutiae, much in the way that Galton had proposed. Important work on topological coding was contributed by Sparrow [Sparrow, 1985]

2.3 Fingerprint Recognition System

2.3.1 Minutia based method

2.3.1.1 Preprocessing

Preprocessing is an inevitable stage in the fingerprint recognition systems which are based on minutiae features. Image enhancement has got the relevance in the latent fingerprint ridge extraction especially in the forensic and criminal scenarios. Also due to scanner conditions, ambient noises and fingerprint skin conditions pre-processing is required in all the fingerprint recognition systems.

2.3.1.2 Image Enhancement

Basically there are two types of fingerprint enhancement techniques: contextual and frequency based filters. O' Gorman and Nikerson [Gorman, 1989] applied the idea of contextual filtering for the enhancement of fingerprint images. Each point of the image is required to be convolved with one of the 16 predefined filters whose orientation best matches with the local ridge orientation. In the

Fourier domain, Sherlock et al. performed contextual filtering [Sherlock, 1994]. The image is convolved with pre-computed filters which are oriented in eight different directions at an interval of 45° of the same size as that of an image. Toshio Kamei [Toshio, 2004] proposed a fingerprint filter design for enhancing fingerprint images. Two distinct filters in the Fourier domain are designed, a frequency filter corresponding to ridge frequencies and a direction filter corresponding to ridge directions on the basis of fingerprint ridge characteristics. The false identification rate is reduced by about two thirds compared with other methods.

Hong et al. [Hong, 1998] proposed a fingerprint enhancement method based on the convolution of the image with Gabor filters. Gabor filter has frequency and orientation selective properties which are best suited for fingerprint like oriented patterns. A Gabor filter is defined by a sinusoidal plane wave modulated by Gaussian function they evaluated the performance of the image enhancement algorithm using the goodness index of the extracted minutiae and the accuracy of an online fingerprint verification system.

Greenberg et al. [2000] proposed that by reducing the value of σ_x with respect to σ_y , the filtering creates fewer spurious ridges and is more robust to noise. Reducing σ_x results in increasing the frequency bandwidth, independently of the angular bandwidth which remains unchanged; this allows the filter to better tolerate errors in local frequency estimates. Sherlock et al. [1994] pointed out that to increase the angular bandwidth one could decrease σ_y .

Keun et al. [Keun, 2000] proposed an algorithm based on directional filter bank. The algorithm decomposes a fingerprint image into directional sub-band images in the analysis stage, processes the sub-band images in the processing stage and reconstructs them as the enhanced image in the synthesis stage. The result shows a reduction in the noise influence on the ridges and valleys enhances the ridges' moving shape and preserve the spatial characteristics at minutiae and singular points.

Sen and Yangsheng [Sen, 2002] proposed a new method of filtering to enhance fingerprint in the singular point area. The singular point area is identified

fist and a new filter is designed to enhance this area. The enhancement algorithm is capable of improving both the quality of fingerprint image and the accuracy of the minutiae extraction.

An adaptive processing method that extracts five features from the fingerprint images, analyses image quality with Ward's clustering algorithm and enhances the images according to their characteristics is proposed by Yun et al. [Yun, 2004]. Khan et al. [Khan, 2005] proposed decimation-free directional filter structure, which provides output in the form of directional images as opposed to directional sub-band provided in previous directional filter banks.

Yang et al. [Yang, 2003] proposed a Modified Gabor Filter [MGF] by discarding the accurate prior sinusoidal plane wave assumption. Authors proposed an image independent adaptive parameter selection scheme, which leads to artefacts in some cases. MGF is more accurate in preserving the fingerprint image topography but like Gabor Filter approach the method fails when image regions are contaminated with heavy noises.

A Log-Gabor filter based fingerprint enhancement method was proposed by Wang et al. [Wang, 2008]. Log-Gabor filters can be constructed with arbitrary bandwidth and can be optimized to have minimal spatial extent. Jang et al. [Jang, 2006] developed an enhancement method based on Half Gabor Filter [HGF] to reduce the computational cost of Gabor based approach. The HGF is a modified filter that reduces the mask size of Gaussian filter while preserving the frequency property of the Gaussian filter. This is a faster method and saves memory space.

Another contextual filtering without requiring explicitly computing local ridge orientation and frequency was proposed by Watson et al. [Watson, 1994] and Willis and Myers [Willis, 2001]. The fingerprint is divided into 32×32 image blocks and each block is enhanced separately; the Fourier transform of the block is multiplied by its power spectrum raised to a power k . The value of k is found out experimentally as 0.6 by Watson et al. and 1.4 by Willis and Myers.

Chikkerur et al. [Chikkerur, 2007] proposed an image enhancement based on Short Time Fourier Transform [STFT] analysis and contextual/non-stationary filtering in Fourier domain. The ridge orientation, frequency, angular coherence

and region mask are probabilistically estimated simultaneously using STFT analysis. This algorithm reduces the space requirements in comparison and also uses the full contextual information like ridge orientation, frequency and angular coherence for enhancement.

2.3.1.3 Directional Field Estimation

The *directional field* [DF] describes the coarse structure, or basic shape, of a fingerprint. The DF is defined as the local orientation of the ridge-valley structures, and is determined using directional gradient. The DF is used for classification, enhancement and segmentation of fingerprint images.

Drets and Liljenström, [Drets, 1999], Wilson et al. [Wilson, 2000] used matched filter methods. Gorman and Nicherson, [Gorman, 1989] based on high frequency power in 3 dimensions, 2 dimensional spectral estimation method by Wilson et al. [Wilson, 2000] are the other methods for directional field estimation. The main drawbacks of these methods as these approaches do not provide as much accuracy as gradient based methods.

M. Kass and Witkin, 1987 introduced the gradient based method and was well cited and adopted by fingerprint researchers [Jain, 1997; Perona, 1998; Rao, 1992; Ratha, 1995]. Bazen and Gerez [Bazen, 2002] have shown that the above method is mathematically equivalent to the principal component analysis of the autocorrelation matrix of the gradient vectors.

2.3.1.4 Image Segmentation

The function of segmentation is to separate the foreground region and the background region and to eliminate background region and is very important for the reliable extraction of minutiae because without segmentation most feature extraction algorithms extract a lot of false features when applied to the noisy background area.

Mehetre et al. [Mehetre, 1987] proposed a method in which the fingerprint area is classified according to the local histogram of ridge orientations. A histogram is computed for each 16 X 16 block and ridge orientation is estimated at

each pixel. The orientation pattern is donated by the histogram peak while flat histogram donates the isotropic signal. This method fails when the input image has uniform blocks because no local ridge orientation can be found in those regions.

Mehetre and Chatterjee [Mehetre, 1989] modified the above method by assigning those blocks as backgrounds which have the gray scale variance lower than a particular threshold value. The above two methods have only moderate segmentation performance. Maio and Maltoni [Maio, 1997] segmented the fingerprint images as foreground or background by using the average magnitude of gradient in each block. The gradient response is high in the foreground area due to ridge valley alteration in the fingerprint area and low in the background region.

Bazen et al. [Bazen, 2001] proposed a segmentation technique in which coherence, mean and variance three pixel features are computed for each pixel. An optimal linear classifier is trained for the classification per pixel and morphology is applied to reduce the classification errors. This method has the disadvantage of low speed and moderate performance. Chen et al. [Chen, 2004] used clusters degree, mean and variance for segmentation by means of an optimized linear classifier. The error rate for misclassified blocks reported is 2.54%.

Akram et al. [Akram, 2008] proposed a modified gradient based method to extract the region of interest. In this method the local gradient values for fingerprint images which detect sharp change in the gray level value of background is computed. This technique segments the fingerprint images accurately especially very dry and wet fingerprint images are segmented in an accurate manner.

Jin Qi and Mei Xie [Jin, 2008] proposed an algorithm that uses the magnitude of gradient and the variance of gradient vector's direction. The method takes advantage of the two prominent properties of fingerprint, i.e., the clarity of ridge to valley and the slow change of fingerprint texture orientation. This technique can not only segment the fingerprint images but also measure the quality of sub-block or overall fingerprint segmented. This algorithm is found to be effective and robust.

2.3.1.5 Binarization and Thinning

The purpose of binarization is to generate image into a binary image for thinning process.

Stock et al. [Stock, 1969] proposed a composite approach based on a local threshold and a “slit comparison” formula that compares pixel alignment along eight discrete directions. Moayer et al. [Moayer, 1986] proposed a binarization technique based on iterative application of a Laplacian operator and a pair of dynamic thresholds. In each iteration, the image is convolved through a Laplacian operator and the pixels whose intensity lies outside the range bounded by the two thresholds are set to 0 and 1, respectively

A fuzzy approach is proposed by Verma et al. [Verma, 1987]. This approach uses an adaptive threshold to preserve the same number of 1 and 0 pixels for each neighborhood. Ratha et al. [Ratha, 1995] explained another binarization approach based on peak detection in the gray level profiles along sections orthogonal to the ridge orientation. This method is fast but most of the bifurcations and even some ridges are broken by the oriented line segments. Emiroglu et al. [Emiroglu, 1997] discussed a regional average thresholding algorithm which uses an 8×8 window but threshold only the 8×4 portion and slide the window for the next 4 points.

GA based method was propose by Abutaleb et al. [Abutaleb, 1999] to discriminate ridges and valleys along the gray-level profile of scanned lines. Dong et al. [Dong, 2004] used the inter ridge distance to calculate the size of the neighbourhood for each pixel. Zhang et al. [Zhang, 2006] proposed an algorithm in which fingerprint image is divided into regions, each containing ridges that have similar ridge gray scale value and in order to store these regions a data structure also has been proposed. The Binarisation is then obtained by local thresholding.

Thinning is the process of eroding the ridges of the fingerprint until they are one pixel wide. Thinning reduces the ridge pattern to a 1-pixel wide skeleton from which minutiae can easily be extracted.

Two types of algorithms are there for thinning:

1. Sequential
2. Parallel

In sequential, the pixels are examined for deletion in a fixed sequence in any iteration. The deletion of a pixel in any iteration [say n^{th}] depends upon the results of $[n-1]^{\text{th}}$ as well as on the pixels already processed in the n^{th} iteration. In parallel thinning algorithms a pixel satisfying a set of rules is immediately removed and deletion in n^{th} iteration depends only on the result that remains after the $[n-1]^{\text{th}}$ iteration.

Jang et al. [Jang, 1992] proposed a parallel thinning algorithm that requires only single pass iteration. The proposed algorithm requires a set of 20 elimination templates out of which 12 are 3×3 , four are 4×4 and four 5×5 . The algorithm consumes more time to execute. Datta et al. [Datta, 1994] proposed a thinning algorithm which ensures some basic properties of thinning but the proposed algorithm uses multipass iterations.

Zhang et al. [Zhang, 1996] used a 2-sub iteration thinning algorithm with template matching. The algorithm preserved the connectivity of the patterns well. Ahmed et al. [Ahmed, 2002] proposed a rule based rotation invariant thinning algorithm in which shape of the image is preserved because it thins the symbol to their central lines according to 20 rules which applied simultaneously to each pixel. Luping et al. [Luping, 2007] proposed a parallel template based pulse coupled neural network based thinning algorithm. The algorithm iteratively skeletonizes images using neuron pulses.

2.3.1.6 Minutia Extraction

Two types of minutiae extraction methods:

- a. Direct gray-scale extraction methods
- b. Binarisation based extraction methods

Leung et al. [Leung, 1990] introduced a neural network-based approach in which the image is first transformed into frequency domain where the filtering takes place and the resulting magnitude and phase signals constitute the input to neural network composed of six sub-networks, each of which is responsible for

detecting minutiae at a specific orientation. A classifier is finally employed to combine the intermediate responses

Maio et al. [Maio, 1997] proposed a direct gray-scale minutia extraction technique based on the ridge line following algorithm which extract a ridge line, given a starting point and direction. The algorithm runs until ridge ending or bifurcation occurs or any one of the three stopping criteria becomes true.

Jiang et al. [Jiang, 2001] proposed another minutiae extraction algorithm which tracks a central ridge and two surrounding valleys simultaneously. During the minutiae extraction process the relationship between central maximum of the ridge and two minima of the corresponding valleys are monitored. Any change in the relationship indicates the presence of a minutia.

Nilsson and Biguin [Nilsson, 2007] used a Linear Symmetry [LS] properties computed by spatial filtering via separable Gaussian filters and Gaussian derivative filters. Minutiae are identified in the gray-scale image as points characterised by the lack of symmetry. The most commonly employed method of minutiae extraction is crossing number [C_n] concept developed by Arcelli and Baja [Arcelli, 1984]. Other thinning techniques developed were by Ratha et al. [Ratha, 1995], Landy et al. [Landy, 1984], Mehtre [Mehtre, 1993], Coetzee and Botha [Coetzee, 1993]. In Ji et al. [Ji, 2007], the skeleton is computed through a Pulse Coupled Neural Network [PCNN] where the orientation image is used to constrain the thinning direction of PCNN thus allowing to reduce bothersome artefacts such as the short spikes that conventional thinning algorithms often produce.

2.3.1.7 Fingerprint Matching

Fingerprint matching deals with determination of whether two fingerprints are from the same finger or not. Due to displacement, rotation and other linear/nonlinear distortions minutiae extracted from different impressions of the same finger do not match with each other. Hence alignment of two fingerprints is required. After pre-alignment, process the matching is simply a pairing process. Approaches of fingerprint alignment have been divided into absolute and relative pre-alignment. In absolute pre-alignment every fingerprint image is aligned, independently of the others with respect to some reference point [core point].

Wegstein [Wegstein, 1982] proposed a method which pre-aligns fingerprints with respect to core positions and average orientations of regions around the core point. Bazen and Gerez [Bazen, 2002] described an absolute pre-alignment method based on the orientation of singularities. Accurate and reliable detection of reference points is a difficult task especially in poor quality images. Jain et al. [Jain, 1997] proposed a relative pre-alignment approach in which ridges associated with minutiae are used to estimate the alignment. Each minutiae in a fingerprint is associated with a ridge, during the feature extraction stage when a minutia is recorded, the ridge corresponding to that minutia is also recorded. Large memory is utilized to handle the template and efficiency is poor.

Luo et al. [Luo, 2000] proposed a slightly different method in which instead of correlating the y -coordinates of the sample points along the two ridges, the distances and relative angles of the sample points are matched. Zhang et al. [Zhang, 2002] explored the possibility of using core points for relative alignment. In the proposed algorithm the core points from the two fingerprints are detected using a multi-resolution algorithm and are used to calculate the translation parameter. This is a fast approach of fingerprint alignment but performance degrades for low quality fingerprints.

Feng et al. [Feng, 2006] proposed another approach which uses one minutia and several associated ridges to align two fingerprints. This is a computationally expensive method and has the drawback of slowness. Hu et al. [Hu, 2008] used eight types of special ridges to align two fingerprints. The ridge with maximum sampled curvature has been used as a reference ridge for initial alignment and other corresponding ridges have been aligned using different features. This method shows some improvement over the minutiae-based method but is dependent upon the pre-processing steps such as thinning, enhancement etc.

Zhao et al. [Zhao, 2010] proposed another technique for alignment of partial high-resolution fingerprints using pores. Expensive computational cost is caused by a large amount of pore features. Chen et al. [Chen, 1991] proposed a topology-based matching algorithm for fingerprint authentication in which the structure of the fingerprint is represented by a geometrical configuration of its minutiae, thus reflecting the general spatial structure in the neighbourhood of each minutia.

Hybrid matching algorithms using both local and global features are proposed in Jain et al. [Jain, 1999], Hunvandana et al. [Huvanandana, 2003], Jain et al. [Jain, 2001] etc. Jia et al. [Jia, 2004] proposed a minutia method based on the vector triangle this method incorporated the information of ridge lines into the process of determining the reference points in two fingerprint images and realizes the fingerprint matching in the polar coordinate system. Bhowmick et al. [Bhowmick, 2004] proposed a new technique of fingerprint matching using an efficient data structure, combining the minutiae representation with the individual usefulness of each minutia to make the matching more powerful. This method used the local topological structure of a valid minutiae and the global structure of the minutiae as a whole.

2.3.2 Image Based Fingerprint Authentication System

In this type of fingerprint authentication, image itself is used as a template or reference image and the intensity values at every point of this template are compared with the intensity values of the query image

Bazen and Gerez [Bazen, 2002] Proposed a three step correlation based fingerprint verification. In the first step small size templates have been selected in the primary reference. In the second step template matching has been used to find the position in the secondary image. Finally the third step aims to compare the template position in both the fingerprints to make the decision regarding the authenticity. But the method is computationally expensive.

Nandakumar et al. [Nandakumar, 2004] proposed a local correlation based fingerprint matching algorithm. A window of size 42 X 42 pixels around the minutia locations in the template image and 32 X 32 pixels size windows around the corresponding location in the query image are used. The normalized cross-correlation between the query window and template window is computed and peak is detected. If the peak lie outside 10 pixels from the centre, the correlation between template and query window is zero otherwise this is the absolute value of correlation between the query and template window. All the correlation values are computed and overall matching score is computed which decides a match or non-match. Minutia extraction is inevitable and the proposed system depends upon the

alignment algorithm used before matching and system is computationally expensive

Marana and Jain [Marana, 2005] proposed Ridge Based Fingerprint matching using the Hough transform. The major straight lines that match the fingerprint ridges are used to estimate rotation and translation parameters. Cavusoglu et al. [Cavusoglu, 2007] proposed a correlation based fingerprint matching algorithm.

The algorithm requires segmentation, ridge orientation, reference point detection, and normalized operations before the application of correlation algorithm. During the enrolment stage starting from the selected reference [core] point of the template image a set of features is obtained with different radius[r] and angle θ . For the authentication purpose the features of the input query image are obtained by rotating the query image with an incremental size of 1° in the given range of -15° to $+15^\circ$. For each rotation the normalized cross correlation value of cross correlation in the given range, determine the similarity of the query and template image. This method is efficient in terms of storage since instead of template the features of the template are stored. The algorithm performs better than the minutiae based approaches especially in the case of poor quality fingerprint image

Jain et al., [Jain 1999] described the use of logistic regression method to integrate multiple fingerprint matching algorithms. The integration of Hough transform based matching, string distance based matching and 2D dynamic programming based matching using the logistic regression has minimized the False Rejection Rate for a specified level of False Acceptance Ratio.

Ouyang et al. [Ouyang, 2006] proposed a correlation based matching method using local Fourier-Mellin descriptor and phase only correlation [POC] function. Texture correlation matching method for fingerprint verification using Fourier-Mellin descriptor and phase-only correlation function is proposed in this paper. Matching takes about 1 second in Celeron 2.0 GHz processor and the experimental results show that EER is 3.8%.

Ito et al. [Ito, 2005] proposed a band limited phase only correlation [BLPOC] based image matching algorithm. BLPOC method is more robust to the noise and provides a sharper peak to distinguish between the genuine and imposter matching than POC. The proposed technique is effective for verifying low quality images that cannot be identified accurately but the detection of the core point and larger number of computations involved for rotation restricts the use of this method and also the elastic deformation degrades the system performance.

Zhang et al. [Zhang, 2006] proposed another correlation based fingerprint matching method using both Fourier-Mellin transform and band limited phase only correlation. Here FMT is used to determine the rotational angle between the reference and query images. The experimental results on FVC2002 fingerprint databases show that the proposed algorithm is effective.

2.4 Speaker recognition

Speaker recognition is the process of automatically recognizing who is speaking by using speaker specific information included in speech waves [Furui, 1986, 1989; Rosenberg, 1991]. This method is used to identify and verify the identity of the people accessing systems. Speaker recognition is applied in services like banking over telephone, passenger information services, tele-shopping, voice mail, security control for confidential information etc. Speaker identity is related with the physiological and behavioural characteristics of the speech production system of each speaker which reflects in the spectral envelope and in the supra-segmental features [Furui, 1997].

Atal [Atal, 1972], Carey et al. [Carey, 1996] and Matsui and Furui [Masui, 1990] have used fundamental frequency to recognize speakers by extracting reliably.

Text dependent speaker recognition methods can be classified into DTW [dynamic time warping] or HMM [hidden Markov model] based methods.

Furui [Furui, 1981] proposed a text-dependent speaker recognition method by DTW method. The approach used is the spectral template matching approach.

Each utterance is represented by a sequence of feature vectors, generally, short-term spectral feature vectors, and the trial-to-trial variation of utterances of the same text is normalized by aligning the analysed feature vector sequence of a test utterance to the template feature vector using a DTW algorithm. The overall distance between the test utterance and the template is used for recognition decision.

HMM based methods have achieved significantly better recognition accuracies than DTW-based methods [Naik, 1989; Rosenberg, 1991; Zheng, 1988]. Rosenberg et al. [Rosenberg, 1990a, and b] introduced and tested a speaker verification system based on characterizing utterance's as sequences of sub-word units represented by HMMs. In this work, two types of sub-word units, phone-like units [PLIs] and acoustic segment units [ASUs] have been extensively studied. The result show the isolated digit utterances show only small differences in performance between PLU- and ASU-based representations.

Long-term sample statistics of various spectral features, such as the mean and variance of spectral features have been used by Furui et al. [Furui, 1972], Markel et al., [Markel, 1972], and Markel and Davi [Markel, 1979]. However, long-term spectral averages are extreme condensations of the spectral characteristics of speakers' utterances and, as such, lack the discriminating power included in the sequences of short-term spectral features used as models in text-dependent methods. Using the long-term averaged spectrum the effect of session-to-session variability was reduced by introducing a weighted cepstral distance measure [Furui, 1972].

Montacie et al. [Montacie, 1992] applied a multivariate auto-regression (MAR) model to the time series of cepstral vectors to characterize speakers using statistical dynamic features. The result show good improvement. Griffin et al. [Griffin, 1994] studied distance measures for the MAR-based method, and reported that when ten sentences were used for training and one sentence was used for testing, identification and verification was used for testing, identification and verification rates were almost the same as those obtained by an HMM-based method. The optimum order of the MAR model was 2 or 3, and that distance

normalization using a posteriori probability was essential to obtain good results in speaker verification.

Vector Quantization (VQ) based methods are the efficient ways of compressing the training data. VQ codebooks consisting of a small number of representative feature vectors are used to characterize speaker-specific features [Li, 1983; Matsui, 1990, 1991; Rosenberg, 1987; Shikano, 1985; Soon, 1987]. In the recognition stage, the input utterance is vector-quantized by using the codebook of each reference speaker.

Poritz [Poritz, 1982] proposed using a five-state ergodic HMM (E-HMM) to classify speech segments into one of the broad phonetic categories corresponding to HMM states. A linear predictive HMM was adopted to characterize the output probability function. In this method, the voices are categorized as strong voicing, silence, nasal/liquid, stop burst/post silence, and frication.

An extended work of Poritz was done by Tishby [Tishby, 1991] by using richer class of mixture autoregressive (AR) HMMs. In these models, the states are described as a linear combination (mixture) of AR sources. It can be shown that mixture models are equivalent to a larger HMM with simple states, together with additional constraints on the possible transitions between states.

Savic and Gupta [Savic, 1990] used a five-state ergodic linear predictive HMM for broad phonetic categorization. After the frames that belong to particular phonetic categories have been identified, feature selection is performed. In the training phase, reference templates are generated and verification thresholds are computed for each phonetic category. In the verification phase, after phonetic categorization, a comparison with the reference template for each particular category provides a verification score for that category. The final verification score is a weighted linear combination of the scores for each category.

Eatcock and Mason [Eatcock, 1990] has implemented a broad phonetic categorization by a speaker-specific hierarchical classifier instead of by an HMM, and is very effective.

Matsui and Furui [Matsui, 1990, 1991] tried a method using a VQ-codebook for long feature vectors consisting of instantaneous and transitional features calculated for both cepstral coefficients and fundamental frequency. Since the fundamental frequency cannot be extracted from unvoiced speech, there are two separate codebooks for voiced and unvoiced speech for each speaker. A new distance measure was introduced to take into account the intra- and inter- speaker variability and to deal with the outlier problem in the distribution of feature vectors.

Juang and Soong [Juang, 1990] have proposed a non-memory less source coding algorithm for VQ-based method using a segment quantization technique. The advantage of a segment quantization codebook over a VQ codebook representation is its characterization of the sequential nature of speech events.

Gauvain et al. [Gauvain, 1995] proposed a statistical modelling approach, where each speaker was viewed as a source of phonemes, modelled by a fully connected Markov chain. Maximum a posteriori (MAP) estimation was used to generate speaker-specific models from a set of speaker-independent seed models. Rosenberg et al. [Rosenberg, 1991] have tested a speaker verification system using 4-digit phrases under field conditions of a banking application. Newman et al. [Newman, 1996] used a large vocabulary speech recognition system for speaker verification.

2.5 Multimodal Biometrics

The term information fusion encompasses any area which deals with utilizing a combination of different sources of information, either to generate one representational format, or to reach a decision [Sanderson, 2004]. The information fusion research has been started in the early 1980s.

Chibelushi et al. [Chibelushi,] combined information from speech and still face profile images using a form or weighted summation of fusion. The EER achieved for speech classifier was about 3.4% while the face profile classifier produced an EER of 3.0%. The fused EER achieved was 1.5%.

Brunelli et al. [Brunelli, 1995a] combined the opinions from a face expert and a speech expert using the weighted product approach. The speech expert alone produced an identification rate of 51% and face expert produced an identification rate of 92%. The combined identification rate was increased to 95%.

Brunelli et al. [Brunelli, 1995b] used two speech experts and three face experts for person identification. The weighted approach was used to fuse the opinions, with the weights found automatically via a heuristic approach. All the five experts combined to achieve an identification rate of 91%.

Dieckmann et al. [Dieckmann, 1997] used three experts (frontal face expert, dynamic lip image expert and text-dependent speech expert). A hybrid fusion scheme involving majority voting and opinion fusion was utilized; two of the experts had to agree on the decision and the combined opinion had to exceed a pre-set threshold. The hybrid fusion scheme provided better performance than using the underlying experts alone.

Kittler et al. [Kittler, 1997] used one frontal face expert which provided one opinion for one face image. Multiple images of one person were used to generate multiple opinions, which were then fused by various means, including averaging. The error rates were shown to be reduced by 40% and that performance gains tended to saturate after using five images. In further work, Kittler et al. [Kittler,] provided theoretical foundations for common fusion approaches such as the summation and product methods. Experimental results showed that the summation approach outperformed the product approach.

Speech and visual lip information were combined using vector concatenation by Luetin [Luetin, 1997]. In text-dependent configuration, the fusion process resulted in a minor performance improvement, however, in text-independent configuration, the performance slightly decreased. This suggests that feature concatenation in this case is unreliable.

Jourlin et al. [Jourlin, 1997a, 1997b] used a form of weighted summation fusion to combine opinions of two experts a text dependent speech expert and a text dependent lip expert. Using an optimal weight, fusion led to better performance than using the underlying experts alone.

Abdeljaoued [Abdeljaoued, 1999] proposed to use a Bayesian post-classifier to reach the verification decision. Abdeljaoued used three experts and showed that use of the above classifier (with Beta distributions) provided lower error rates than when using the experts alone. Ben-Yacoub et al. [Ben, 1999] investigated the use of several binary classifiers for opinion fusion using a post-classifier and the investigated classifiers were: Support Vector Machine, Bayesian Classifier, Fishers Linear Discriminant, Decision Tree and Multi-Layer Perceptron. Three experts were used: a frontal face expert and two speech based expert. The SVM classifier and Bayesian classifier provided the best results.

Various binary classifiers for opinion fusion as well as the majority voting and AND & OR fusion methods were used by Verlinde [Verlinde, 1999]. The three experts used were: frontal face expert, face profile expert and a text independent speech expert. The Logistic Regression based classifier provided the lowest overall error rates as well as being the easiest to train.

Wark et al. [Wark, 1999] used the weighted summation approach to combine the opinions of a speech expert and a lip expert. Depending on the value of the weights, the performance in high noise levels was actually worse than using the lip expert alone. Multi-Stream Hidden Markov Models were evaluated for the task of text-dependent audio-visual person identification by Wark et al. [Wark, 2000].

Chapter 3

Fingerprint Recognition Techniques

In this chapter various steps involved in fingerprint recognition are discussed the steps associated with fingerprint orientation estimation, fingerprint segmentation, ridge frequency estimation, enhancement techniques and feature extraction methods are discussed. Also, fingerprint matching methods, one of the major challenges in the fingerprint authentication, are also discussed. The techniques explained are compared in terms of the accuracy and computational complexity.





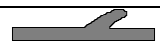


3.1 Introduction

A fingerprint is a well oriented pattern of interleaved ridges and valleys. A ridge is represented in a fingerprint image as a dark region and a valley is represented as a white region. Typically ridges have a thickness of $100\mu\text{m}$ to $300\mu\text{m}$ and the ridge-valley has a thickness of about $500\mu\text{m}$. ridges and valleys often run in parallel, sometimes they bifurcate and sometimes they terminate. The various discontinuities and their small details are referred to as 'minutiae'. Francis Galton observed that the minutiae remain unchanged over an individual's lifetime and categorized minutiae as per Table 3.1 [Galton, 1892]. According to American National Standards Institute finger minutiae has only four classes: terminations, bifurcations, trifurcations (cross overs) and undetermined. The Federal Bureau of Investigation (FBI) taxonomy is based on two classes: termination and bifurcation [Wegstein, 1982]. Each minutia is denoted by its calls, the x- and y-coordinates and the angle between the tangent to the ridge line at the minutia position and the horizontal axis (Fig. 3.1) [Maltoni, 2005]. In the figure, θ is the angle that the minutia tangent forms with the horizontal axis and for a bifurcation minutia, θ is now defined by means of the ridge ending minutia corresponding to the original bifurcation that exists in the negative image.

3.2 Fingerprint Image Processing and Feature Extraction

A direct comparison of two fingerprint images in an automated environment by means of correlation techniques is computationally expensive and unstable. Majority of the fingerprint recognition and classification algorithms require a feature extraction stage for identifying the main features of the fingerprint. Fingerprint features include singularities or minutia. Features may be used for matching of fingerprints or may be used for the derivation of other features.

Table 3.1 Galton's seven minutiae categories

	Termination
	Bifurcation
	Lake
	Independent ridge
	Spur
	Point or island
	Crossover

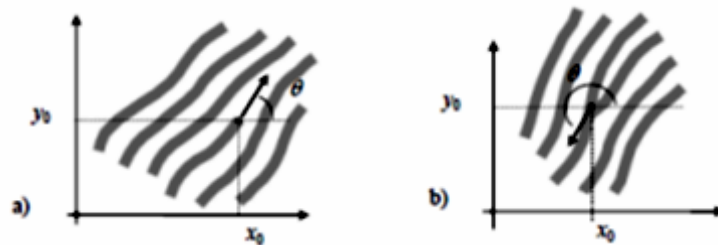


Fig.3.1 a) Ridge Ending Minutia b) Bifurcation Minutia

Fingerprint identification system will have to carry out several steps for feature extraction which include:

1. Ridge Orientation Estimation
2. Fingerprint Image Segmentation
3. Singularity and Core Detection

4. Fingerprint Enhancement
5. Minutiae Detection

The various techniques/methods available in this field are reviewed/explained in the following sections.

3.2.1 Ridge Orientation Estimation

The fingerprint orientation image (also called directional image) is first introduced by Grasselli [Grasselli, 1969]. The orientation image is a matrix \mathbf{O} whose elements encode the local orientations of fingerprint ridges. Orientation image is one of the key features used to filter, to find singularities in a fingerprint image and to classify fingerprint images. Let $[x, y]$ be a pixel in a fingerprint image. The local ridge orientation at $[x, y]$ is the angle θ_{xy} that the fingerprint ridges crossing through an arbitrary small neighbourhood centered at $[x, y]$ form with the horizontal axis. The fingerprint ridges are not directed, θ_{xy} is an un-oriented direction lying in $[0 \dots 180^\circ]$. Each element θ_{ij} , corresponding to the node $[i, j]$ of a square-meshed grid located over the pixel $[x_i, y_j]$ denotes the local orientation (Fig. 3.2). The magnitude of r_{ij} is associated with θ_{ij} to denote the reliability or consistency of the orientation. The value of r_{ij} is low for noisy and corrupted regions and high for good quality regions in the fingerprint image.

The most common and simplest method for extracting local ridge orientation is based on computation of gradients in the fingerprint image.

3.2.1.1 Orientation Estimation by gradient based method:

M. Kass and Witkin, [Kass, 1987] introduced the gradient based method in which the image gradient G_{xy} at point $[x_i, y_j]$ of image \mathbf{I} , is a two-dimensional vector $[G_x(x, y) \quad G_y(x, y)]^T$, where G_x and G_y components are the derivatives of \mathbf{I} at the point $[x_i, y_j]$ with respect to the x and y directions respectively and can be computed using the classical Prewitt or Sobel convolution masks [Gonzales, 1992]

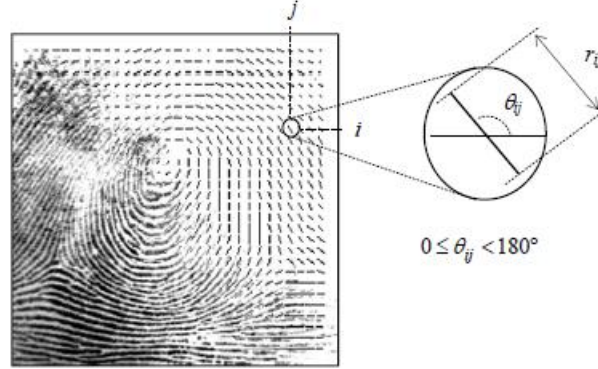


Fig.3.2 Fingerprint orientation field estimated over a square-meshed grid of size 16×16 .

The gradient vector $[G_x(x, y) \ G_y(x, y)]^T$ is defined as:

$$\begin{bmatrix} G_x(x, y) \\ G_y(x, y) \end{bmatrix} = \nabla I(x, y) = \begin{bmatrix} \frac{\partial I(x, y)}{\partial x} \\ \frac{\partial I(x, y)}{\partial y} \end{bmatrix} \quad (3.1)$$

where $I(x, y)$ represents the gray-scale image. The directional field is perpendicular to the gradients. Gradients are orientations at pixel-scale whereas directional field describes orientation of ridge-valley structure. An averaging operation is done on the gradients to obtain the directional field. Gradients cannot be averaged in the local neighbourhood as opposite gradients will cancel each other. To solve this problem Kass and Witkin doubled the angle of the gradient vectors before averaging. Doubling makes opposite vectors point in the same direction and will reinforce each other, while perpendicular gradients will cancel each other. After averaging, the gradient vectors have to be converted back to their single-angle representation.

The gradient vectors are estimated first in Cartesian co-ordinate system and is given by $[G_x \ G_y]$. For the purpose of doubling the angle and squaring the length, the gradient vector is converted to polar system, which is given by

$[\rho \ \phi]^T$ where $-\frac{1}{2}\pi < \phi \leq \frac{1}{2}\pi$

$$\begin{bmatrix} \rho \\ \phi \end{bmatrix} = \begin{bmatrix} \sqrt{G_x^2 + G_y^2} \\ \tan^{-1} \frac{G_y}{G_x} \end{bmatrix} \quad (3.2)$$

The gradient vector is converted back to its Cartesian as:

$$\begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \rho \cos \phi \\ \rho \sin \phi \end{bmatrix} \quad (3.3)$$

The average squared gradient $[\overline{G_{s,x}} \ \overline{G_{s,y}}]^T$ is given by

$$\begin{bmatrix} \overline{G_{s,x}} \\ \overline{G_{s,y}} \end{bmatrix} = \begin{bmatrix} \sum_W G_x^2 - G_y^2 \\ \sum_W 2G_x G_y \end{bmatrix} = \begin{bmatrix} G_{xx} - G_{yy} \\ 2G_{xy} \end{bmatrix} \quad (3.4)$$

where

$$\begin{aligned} G_{xx} &= \sum_W G_x^2 \\ G_{yy} &= \sum_W G_y^2 \\ G_{xy} &= \sum_W G_x G_y \end{aligned} \quad (3.5)$$

are estimates for the variances and crosscovariances of G_x and G_y , averaged over the window W . The average gradient direction ϕ is given by:

$$\phi = \frac{1}{2} \angle(G_{xx} - G_{yy}, 2G_{xy}) \quad (3.6)$$

where $\angle(x,y)$ is defined as:

$$\angle(x, y) = \begin{cases} \tan^{-1}(y/x) & x \geq 0 \\ \tan^{-1}(y/x) + \pi & \text{for } x < 0 \wedge y \geq 0 \\ \tan^{-1}(y/x) - \pi & x < 0 \wedge y < 0 \end{cases}$$

Fingerprint and orientation images are shown in fig. 3.3

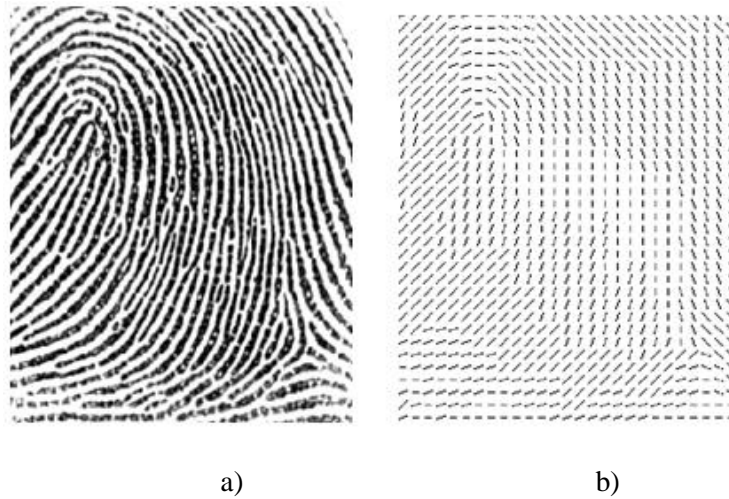


Fig.3.3 a) Fingerprint image b) Orientation field

Orientation field can be used to enhance the fingerprint by means of directional filters like Gabor filter. The major drawback of gradient-based orientation estimators is their failure in the near-zero gradient regions, i.e. in the ridge tops and valley bottoms.

3.2.2 Estimation of Local Ridge Frequency

Ridge frequency is one of the feature parameter used for enhancement using Gabor filter. The local ridge frequency (or density) $f(x,y)$ at $[x, y]$ 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} . The local ridge frequency has variations across different fingers; also vary across different regions in the same fingerprint.

Hong et al. [Hong, 1998] estimated local ridge frequency by counting the average number of pixels between two consecutive peaks of gray-levels along the direction normal to the local ridge orientation (Fig. 3.4). For this purpose a section of fingerprint orthogonal to local orientation is taken and the frequency f_{ij} is computed as follows:

1. A 32 X 16 oriented window centered at $[x, y]$ is defined in the ridge coordinate system.
2. The x -signature of the gray-levels is obtained by accumulating, for each column x , the gray-level values of the corresponding pixels in the oriented window.
3. f_{ij} is found out as the inverse of the average distance between two consecutive peaks of the x -signature.

Even though the method is simple and fast, it is difficult to reliably detect consecutive peaks of gray-levels in the spatial domain in noisy fingerprint images. Yang et al. [Yang, 2003] suggested making use of a fitting method based on first and second order derivatives as an alternative to extract ridge distances from the x -signature.

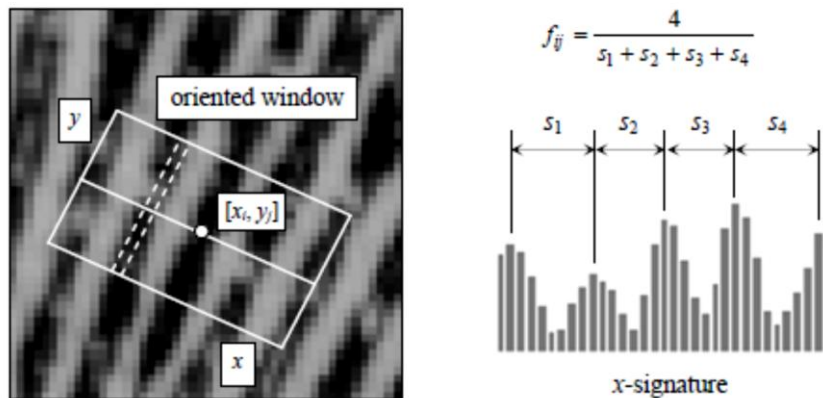


Fig.3.4 Scheme for ridge frequency estimation.

In the method proposed by Maio and Maltoni [Maio, 1998] the ridge pattern is modeled as a sinusoidal-shaped surface, and by using variation theorem, the frequency is estimated (Fig. 3.5). If the variation V of a function h in the interval $[x_1, x_2]$ are small, the variation may be expressed as a function of the average amplitude α_m and the average frequency f .

$$V(h) = (x_2 - x_1) \cdot 2\alpha_m \cdot f \quad (3.7)$$

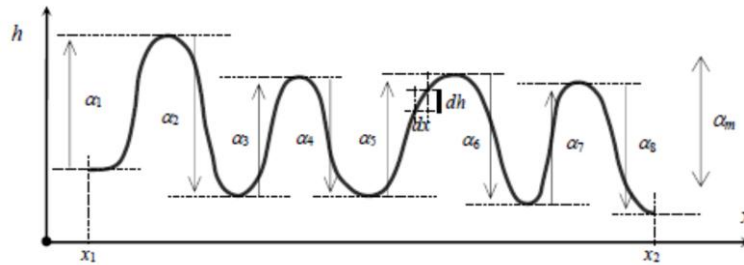


Fig.3.5 Variation of the function h in the interval $[x_1, x_2]$ is the sum of amplitudes $\alpha_1, \alpha_2, \dots, \alpha_8$. If the function is periodic or the function amplitude does not change significantly within the interval of interest, the average amplitude α_m can be used to approximate the individual α values.

$$f = \frac{V(h)}{2 \cdot (x_2 - x_1) \cdot \alpha_m} \quad (3.8)$$

Maio and Maltoni (1998a) proposed a practical method based on the above analysis. The variation and the average amplitude of a two-dimensional ridge pattern are estimated from the first and second-order partial derivatives and the local ridge frequency is computed from Equation (3.8).

3.2.3 Fingerprint Image Segmentation

Segmentation is the process of separating the foreground regions in the image from the background regions [Gonzalez, 2007]. The foreground region contains the ridges and valleys which is the area of interest in a fingerprint image. The background corresponds to the regions outside the borders of the fingerprint

area, which do not contain any valid fingerprint information. When minutiae extraction algorithms are applied to the background regions of an image, it results in the extraction of noisy and false minutiae. Thus, segmentation is employed to discard these background regions, which facilitates the reliable extraction of minutiae.

In a fingerprint image, the background regions generally exhibit a very low variance in gray value and the foreground regions have a very high variance. As the fingerprint images are striated patterns, using a global or local thresholding technique does not allow the fingerprint area to be effectively isolated. Variance thresholding method proposed by Mehtre [Mehtre, 1993] is one of the simplest methods available in the literature. In this method, fingerprint image is divided into blocks and the gray scale variance is calculated for each block in the image. If the variance is less than the global threshold, then the block is assigned to be a background region; otherwise it is assigned to be part of the foreground. Considering a block of size $w \times w$, the variance is given by:

$$V(k) = \frac{1}{W^2} \sum_{i=0}^{W-1} \sum_{j=0}^{W-1} [I(i,j) - M(k)]^2 \quad (3.9)$$

where $V(k)$ is the variance for the block k , $I(i,j)$ is the gray level value at pixel (i,j) , and $M(k)$ is the mean gray level value for the block k . Fig. 3.6 shows the variance thresholding based segmented image with $w = 16$ and $w = 8$



Fig.3.6 a) Fingerprint image b) Segmented image with $w = 16$ c) Segmented image with $w = 8$

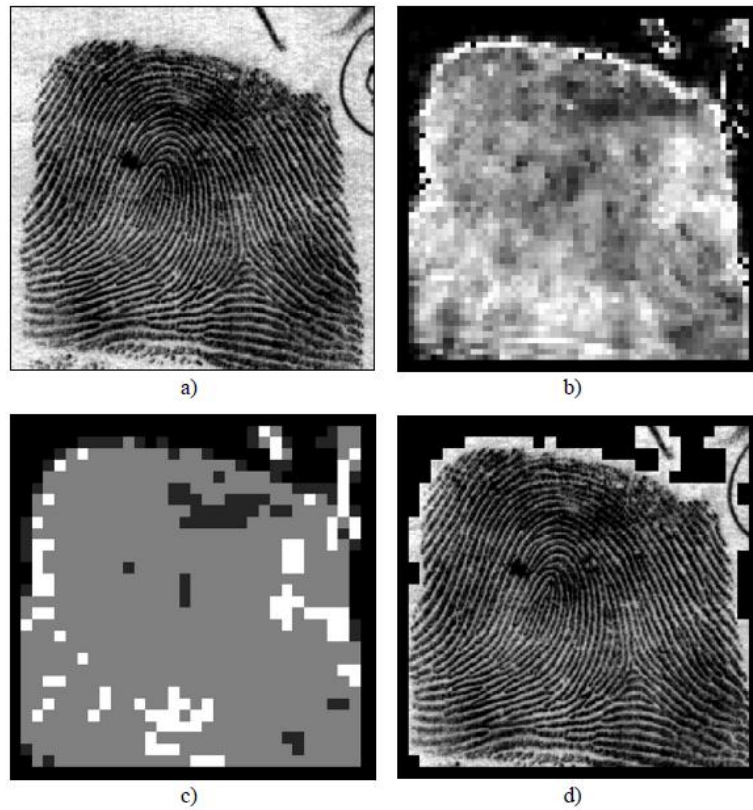


Fig 3.7 Segmentation of a fingerprint image proposed by Ratha et al. a) original image b) variance field c) quality image derived from the variance field d) Segmented Image

Ratha et al. [Ratha, 1995] assigned each 16×16 block to the foreground or the background, according to the variance of gray-levels in the orthogonal direction to the ridge orientation. Also a quality index from the block variance is derived; underlying assumption is that the noisy regions have no directional dependence, whereas regions of interest exhibit a very high variance in a direction orthogonal to the orientation of ridges and very low variance along ridges. Fig.3.7 shows some of the results obtain by the algorithm.

Maio and Maltoni [Maio, 1997] segmented foreground and background by using the average magnitude of the gradient in each image block because of the reason that the fingerprint area is rich in edges due to the ridge/valley alternation, the gradient response is high in the fingerprint area and small in the background. Shi et al. [Shi, 2004] proposed a method based on gray-scale statistics in which contrast-based segmentation is preceded by a non-linear histogram manipulation aimed at decreasing the contrast in ambiguous regions.

3.2.4 Singularity and Core Detection

Singularities are another set of important fingerprint structures that have both global and local properties [Neil, 2004]. Globally, a singularity is a region of a fingerprint where the ridge pattern makes it visually prominent. There are two types of fingerprint singularities: cores and deltas. Locally, a core is the turning point of an inner-most ridge and a delta is a place where two ridges running side-by-side diverge. Core and delta points are best illustrated by examples (Fig. 3.8). Singularities play important role in determining a fingerprint's class. For example, left loops (as in Fig. 3.8) have one core point near the centre of the print and one delta point to the lower right. Singularities also have other uses, such as fingerprint

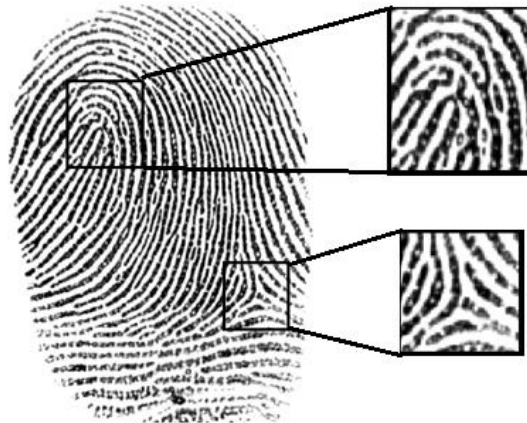


Fig. 3.8 Fingerprint Singularities: Core above and delta below

alignment and as a coarse discriminating feature. In this work, singularities are the key features used to identify fingerprints and hence a discussion of some of the techniques for singularity detection is followed. There are several approaches proposed in the literature, for singularity detection, which operate on the fingerprint orientation image. The most commonly used method was proposed by Kaswogoe and Tojo (Kawogoe, 1984) based on Poincaré index.

3.2.4.1 Singularity detection using Poincaré index:

Let \mathbf{G} be a vector field and C be a curve immersed in G ; then the Poincaré index $P_{G,C}$ is defined as the total rotation of the vectors of \mathbf{G} along C as shown in Fig. 3.9.

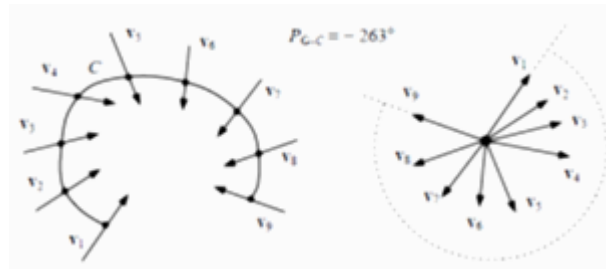


Fig. 3. 9 The Poincaré index computed over a curve C immersed in a vector field \mathbf{G} [Maltoni, 2005]

Let \mathbf{G} be the discrete vector field associated with a fingerprint orientation image \mathbf{D} and let $[i,j]$ be the position of the element θ_{ij} in the orientation image; then the Poincaré index $P_{G,C}$ at $[i,j]$ is computed as follows

- The curve C is a closed path defined as an ordered sequence of some elements of \mathbf{D} , such that $[i,j]$ is an internal point;
- $P_{G,C}$ is calculated by algebraically summing the orientation differences between adjacent elements of C . On close curves, Poincaré index assumes only one of the discrete values: 0° , $\pm 180^\circ$ and $\pm 360^\circ$. In the case of fingerprint singularities:

$$P_{G,C} = \begin{cases} 0, & \text{if } [i,j] \text{ does not belong to any singular region} \\ 360^\circ, & \text{if } [i,j] \text{ does not belong to a whorl type singular region} \\ 180^\circ, & \text{if } [i,j] \text{ does not belong to a loop type singular region} \\ -180^\circ & \text{if } [i,j] \text{ does not belong to a delta type singular region} \end{cases}$$

Fig. 3.10 shows three portions of orientation images. The path defining C is the ordered sequence of the eight elements \mathbf{d}_k ($k=0\dots7$) surrounding $[i, j]$. The direction of the elements \mathbf{d}_k is chosen as follows: \mathbf{d}_0 is directed upward; $\mathbf{d}_k(k=1\dots7)$ is directed so that the absolute value of the angle between \mathbf{d}_k and \mathbf{d}_{k-1} is less than or equal to 90° . $P_{G,C}$ is then computed as

$$P_{G,C}(i,j) = \sum_{k=0\dots7} \text{angle}(\mathbf{d}_k, \mathbf{d}_{(k+1)\text{mod}8}) \quad (3.10)$$

An example of singularities detected by the above method is shown in fig. 3.11.

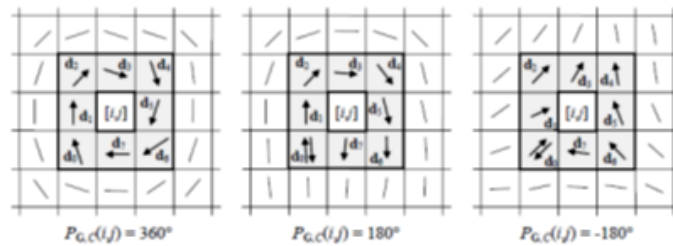


Fig. 3.10 Computation of the Poincaré index in the 8-neighborhood of points belonging to a whorl, loop, and delta singularity respectively [Maltoni, 2005]

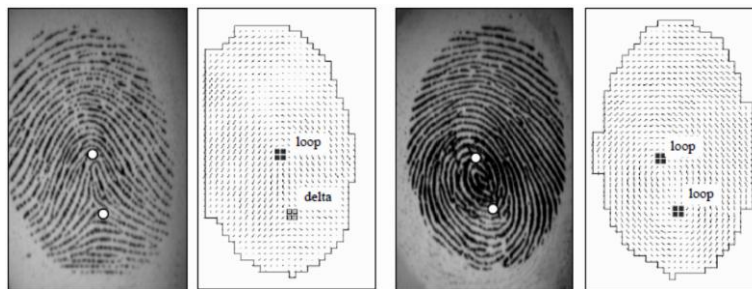


Fig.3.11. Singularity detection by using the Poincaré index method. [Maltoni, 2005]

An alternative method was proposed by Bazen and Gerez [Bazen, 2002] for locating singular points; according to Green's theorem, a closed line integral over a vector field can be calculated as a surface integral over the rotation of this vector field. Instead of summing angle differences along a closed path, the authors compute the "rotation" of the orientation image and then perform a local integration (sum) in a small neighborhood of each element.

Singularity detection in noisy or low-quality fingerprints is difficult and the Poincaré method may lead to the detection of false singularities [Maltoni, 2005] as given in fig. 3.12. By regularizing the orientation image through a local averaging is often quite effective in preventing the detection of false singularities. Karu and Jain [Karu, 1996] proposed smoothing of orientation image iteratively through averaging until a valid number of singularities is detected by the Poincaré index (fig. 3.12c).

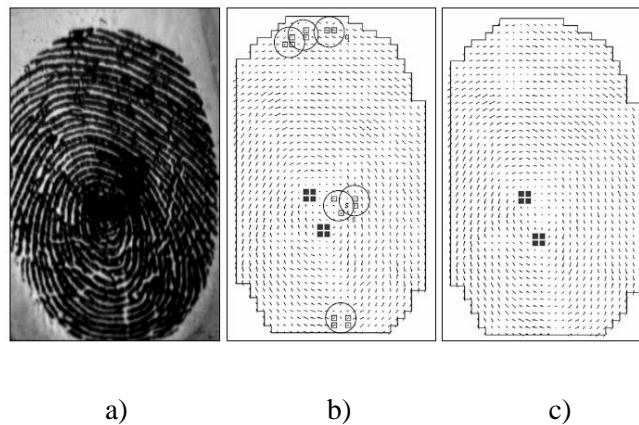


Fig 3.12. a) A poor quality fingerprint; b) the singularities of fingerprint in a) are extracted through Poincaré method (circles highlight the false singularities); c) the regularized orientation.

3.2.4.2 Singularity detection based on local characteristics of orientation image

The reliability r of the estimate can be derived by the concordance (or coherence) of the orientation vectors \mathbf{d} in the local window W (Kass and Witkin

(1987); Bazen and Gerez (2002)). In fact, due to the continuity and smoothness of fingerprint ridges, sharp orientation changes often denote unreliable estimation. Kass and Witkin (1987) defined the coherence as the norm of the sum of orientation vectors divided by the sum of their individual norms. This scalar always lies in $[0,1]$. Its value is 1 when all the orientations are parallel to each other (maximum coherence) and 0 if they point in opposite directions (minimum coherence):

$$r = coherence(\theta) = \frac{|\sum w \mathbf{d}|}{\sum w |\mathbf{d}|} \quad (3.11)$$

The above equation simplifies to:

$$r_{ij} = coherence(\theta_{ij}) = \sqrt{(G_{xx} - G_{yy})^2 + 4G_{xy}^2} \quad (3.12)$$

The coherence defined was used by Cappelli et al. [Cappelli, 1999] to coarsely locate singular regions. Fig. 3.13 shows the coherence map computed over 3×3 neighbourhoods.

3.2.4.3 Core detection and fingerprint registration

The core position may be defined as the location of the north most loop. When the core point is detected with the aim of registering fingerprint images, its location may be quite critical and an error at this stage often leads to a failure of subsequent processing like matching. On the other hand, if the core has to be used

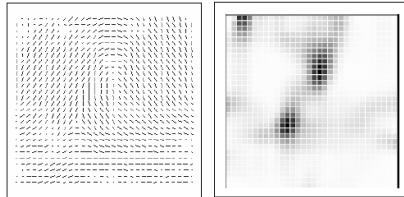


Fig. 3.13. Orientation image and the corresponding coherence map. Dark regions identifies the singularities

only for fingerprint registration, it is not important to find the north most loop exactly and any stable point in the fingerprint pattern is suitable.

Wegstein [Wegstein, 1982] proposed an automatic method, known as R92, searches a core point independently of the other singularities. The core is searched by scanning the orientation image to find *well-formed* arches; a well-formed arch is denoted by a sextet (set of six) of adjacent elements whose orientations comply with several rules controlled by many parameters. One sextet is chosen among the valid sextets by evaluating the orientation of the elements in adjacent rows. The exact core position is then located through interpolation (Fig. 3.14).

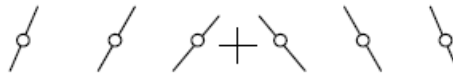


Fig.3.14. The core point “+” located on the chosen sextet

This algorithm was a fundamental component of the fingerprint identification systems used by the FBI and is still extensively used by other researchers like Candela et al. [Candela, 1995]. Other registration techniques include methods proposed by Novikov and Kot [Novikov, 1998] in which the core is defined as the crossing point of the line normal to the ridges as shown in fig.3.15 and used Hough transform to determine its coordinates. In the figure, the straight lines normal to the ridges identify a valid registration point that corresponds to the center of curvature



Fig.3.15. Fingerprint registration using Hough Transform method [Maltoni, 2005]

3.2.5 Fingerprint Enhancement

The feature extraction and minutia detection relies heavily on the quality of the input fingerprint image. A good quality fingerprint image is characterized by the ridges and valleys alternate and flow in a locally constant direction. The ridges can be easily detected in that case and minutiae can be precisely located in the image. But due to skin conditions, sensor noise, incorrect finger pressure, and inherently low-quality fingers, approximately 10% of the fingerprints are of poor quality and the ridge pattern is very noisy and corrupted. Fingerprint images have the following degradation due to several reasons:

- The ridges have small breaks or gaps
- Parallel ridges are not well separated due to the presence of noise which links parallel ridges, resulting in their poor separation.
- Cuts, ceases and bruises on the finger.

The results of the above degradation in fingerprint image are:

- A significant number of spurious minutiae are extracted
- A large number of genuine minutiae are missed and
- Large errors in the location of minutiae are introduced.

In order to overcome the degradation in quality of fingerprint image an enhancement step is carried out in all the fingerprint authentication systems.

Generally for a given fingerprint image, the fingerprint areas resulting from the segmentation step may be divided into three categories [Maltoni, 2003] (Fig. 3.16).

- *Well-defined region*: Well-defined region: ridges can be clearly differentiated from each another.
- *Recoverable region*: 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.

- *Unrecoverable region*: 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.

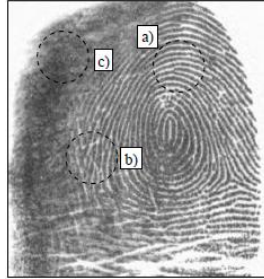


Fig. 3.16. Fingerprint image containing regions of different quality: a) well-defined region b) a recoverable region; c) an unrecoverable region. [Maltoni, 2005]

The aim of enhancement algorithm is to improve the clarity of the ridge structures in the recoverable regions and mark the unrecoverable regions as too noisy for further processing. Fingerprint images are well-defined pattern and general purpose image enhancement techniques do not produce satisfying and definitive results. In the pre-processing stage, histogram manipulation, contrast stretching, normalisation and Weiner filtering have been shown to be effective. There are two methodologies for fingerprint image enhancements: Pixel-wise enhancement and contextual-wise enhancement.

3.2.5.1 Pixel-wise enhancement:

In a pixel-wise image processing operation the new value of each pixel depends only on its previous value and some global parameters (but not on the value of the neighboring pixels). Normalization is one of the pixel-wise enhancement method carried out as pre-processing. The method proposed by Hong et al. [Hong, 1998] determines the new intensity value of each pixel in an image as

$$I'[x, y] = \begin{cases} m_0 + \sqrt{(I[x, y] - m)^2 \cdot \frac{v_0}{v}} & \text{if } I[x, y] > m \\ m_0 - \sqrt{(I[x, y] - m)^2 \cdot \frac{v_0}{v}} & \text{otherwise} \end{cases} \quad (3.13)$$

where m and v are the image mean and variance and m_0 and v_0 are the desired mean and variance after the normalization. The effect of normalization is shown in fig.3.17.



Fig.3.17. Normalization method using Hon et al. [Maltoni, 2005] ($m_0 = 100$ and $v_0 = 100$)

Kim and Park [Kim, 2002] introduced a block-wise implementation of Equation (9) where m and v are the block mean and variance, respectively, and m_0 and v_0 are adjusted for each block according to the block features. A similar adaptive normalization was proposed by Zhixin and Govindaraju [Zhixin, 2006]. However, this kind of normalization involves pixel-wise operations and does not change the ridge and valley structures. In particular, it is not able to fill small ridge breaks, fill intra-ridge holes, or separate parallel touching ridges

3.2.5.2 Contextual filtering:

Unlike in a conventional image filtering, where 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 filters 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 sinusoidal-shaped wave of ridges 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.

The first contextual filter was proposed by O’Gorman and Nickerson [Gorman, 1988, 1989]. A mother filter based on four main parameters of fingerprint images at a given resolution: minimum and maximum ridge width, and minimum and maximum valley width. The filter is a bell-shaped, elongated along the ridge direction, and cosine tapered in the direction normal to the ridges (Fig.3.18). The local ridge frequency is assumed constant and therefore, the selective parameter is only the local ridge orientation. Once the mother filter has been generated, a set of 16 rotated versions (in steps of 22.5°) is derived. The image enhancement is performed by convolving each point of the image with the filter in the set whose orientation best matches the local ridge orientation. Depending on some input parameters, the output image may be gray-scale or binary.

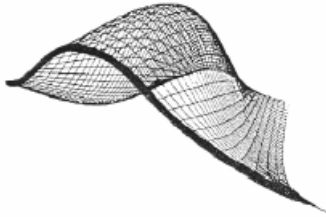


Fig.3.18. The mother filter response proposed by O’Gorman and Nickerson [Gorman, 1989]

Sherlock et al. [Sherlock, 1992, 1994] performed contextual filtering in the Fourier domain. The filter is defined in the frequency domain by the function:

$$H(\rho, \theta) = H_{radial}(\rho) \cdot H_{angle}(\theta) \quad (3.14)$$

where H_{radial} depends only on the local ridge spacing $\rho = \frac{1}{f}$ and H_{angle} depends only on the local ridge orientation θ . Both H_{radial} and H_{angle} are defined as bandpass filters and are characterized by a mean value and a bandwidth. A set of n discrete filters is derived by their analytical definition. The Fourier Transform \mathbf{P}_i , $i=1 \dots n$ of the filters is pre-computed and stored. Filtering of an input fingerprint image \mathbf{I} is performed as follows

- The FFT \mathbf{F} of \mathbf{I} is computed.
- Each filter \mathbf{P}_i is point-by-point multiplied by \mathbf{F} , thus obtaining in filtered image transforms \mathbf{PF}_i , $i = 1 \dots n$ (in the frequency domain).

- Inverse FFT is computed for each PF_i resulting in n filtered images **PI**_i, $i = 1 \dots n$ (in the spatial domain).

The enhanced image **I**_{enh} is obtained by setting, for each pixel $[x, y]$, **I**_{enh} $[x, y] = \mathbf{PI}_k[x, y]$, where k is the index of the filter whose orientation is the closest to θ_{xy} . The entire scheme is shown in fig. 3.19.

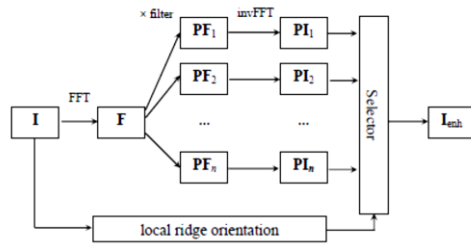


Fig.3.19. Fingerprint enhancement scheme proposed by Sherlock et al. [Sherlock, 1994].

3.2.5.2.1 Gabor filter based enhancement

Gabor filter based enhancement is the most effective method in this scenario and was proposed by Hong et al. [Hong, 1998]. A Gabor filter is defined by a sinusoidal plain wave modulated by a Gaussian. The even symmetric two-dimensional Gabor filter has the equation given by:

$$g(x, y; \theta, f) = \exp\left\{-\frac{1}{2}\left[\frac{x_\theta^2}{\sigma_x^2} + \frac{y_\theta^2}{\sigma_y^2}\right]\right\} \cdot \cos(2\pi f \cdot x_\theta) \quad (3.15)$$

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 an angle of $(90^\circ - \theta)$. i.e.,

$$\begin{bmatrix} x_\theta \\ y_\theta \end{bmatrix} = \begin{bmatrix} \cos(90^\circ - \theta) & \sin(90^\circ - \theta) \\ -\sin(90^\circ - \theta) & \cos(90^\circ - \theta) \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \sin\theta & \cos\theta \\ -\cos\theta & \sin\theta \end{bmatrix} \quad (3.16)$$

f is the frequency of the sinusoidal plane wave and σ_x and σ_y are the standard deviations of the Gaussian envelope along the x - and y - axes respectively. Fig.3.20 shows the graphical representation of the Gabor filter.

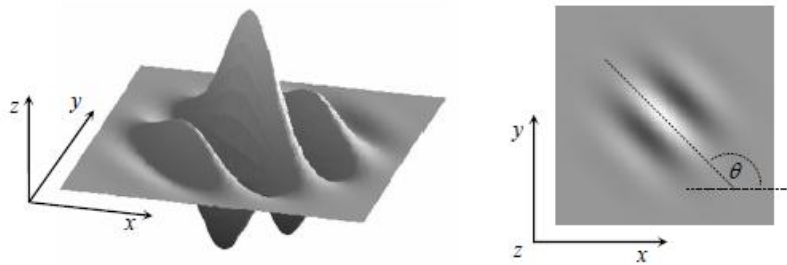


Fig. 3.20 Graphical representation of the Gabor filter defined by the parameters $\theta = 135^\circ$ $F=1/5$, and $\sigma_x=\sigma_y=3$ [Maltoni, 2005].

Gabor filters have the four parameters $(\theta, f, \sigma_x, \sigma_y)$ to be specified. The frequency of the filter is governed by the ridge frequency and the orientation is determined by the local ridge orientation. The values of σ_x and σ_y are to be selected in an empirical way. Large values make the filter robust, but are also more likely to create spurious ridges and valleys. Smaller the values, the filters are less likely to introduce spurious ridges but less effective in removing the noise. In practice, $\sigma_x = \sigma_y = 4$. Also, instead of computing the filter function for the given θ and f for each pixel, a set $\{g_{ij}(x,y): i=1\dots n_0, j=1\dots n_f\}$ of filters are a-priori created and stored, where n_0 is the number of discrete orientations $\{\theta_i: i= 1..n_0\}$ and n_f the number of discrete frequencies $\{f_j: j=1\dots n_f\}$. Then each pixel $[x,y]$ of the image is convolved, in the spatial domain, with the filter $g_{ij}(x,y)$ such that θ_i is the discretized orientation closest to θ_{xy} and f_j is the discretized frequency closest to f_{xy} . Fig. 3.21 shows an example of the filter set for $n_0= 8$ and $n_f= 3$.

There are several modifications for Gabor filters that can be found in the literature.

3.2.5.2.2 FFT based enhancement

Another contextual filtering without requiring explicitly computing local ridge orientation and frequency was proposed by Watson et al. [Watson, 1994] and Wills and Myers [Wills, 2001].

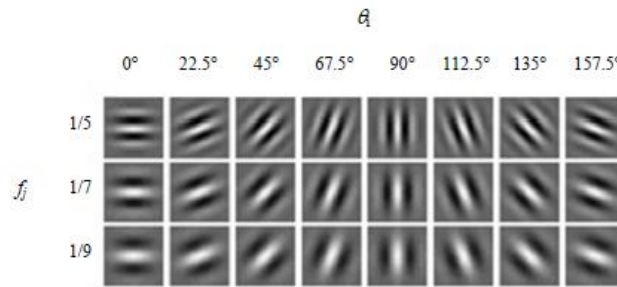


Fig. 3.21 A bank of 24 Gabor filters

The fingerprint is divided into 32×32 image blocks and each block is enhanced separately; the Fourier transform of the block is multiplied by its power spectrum raised to a power k :

$$I_{enh}[x, y] = F^{-1}\{F(I[x, y]) \times |F(I[x, y])|^k\} \quad (3.16)$$

The power spectrum contains information about the underlying dominant ridge orientation and frequency and the multiplication has the effect of enhancing the block accordingly. Watson et al. set $k=0.6$ and Willis and Myers proposed a value of $k=1.4$.

3.2.6 Minutiae Detection

Majority of fingerprint identification system rely on minutiae matching and detection of a reliable method for minutiae detection is the final and most important task in fingerprint feature extraction. The first step utilised in minutiae detection phase is to convert the gray-scale image into a binary image. Binarisation processes greatly benefit from a priori enhancement. The binary images are

subjected to a thinning stage which reduces the thickness of ridges to one pixel wide, resulting in a skeleton image. The process is shown in fig. 3.22.

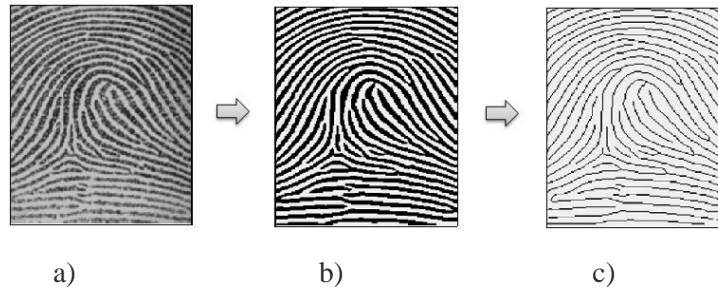


Fig. 3.22. a) Fingerprint Image b) Binarised Image c) Thinned Image

3.2.6.1 Binarisation methods

One of the easiest approaches in binarisation is to use a global threshold t and by setting the pixels whose gray-level is lower than t to 0 and the remaining pixels to 1. Contrary to that, a local threshold technique in which the t is changed locally by adapting its value to the average local intensity is also used for binarisation. For fingerprints or very poor quality, a local threshold method cannot always guarantee acceptable results and more effective fingerprint-specific solutions are necessary.

An iterative application of a Laplacian operator and a pair of dynamic thresholds was proposed by Moayer and Fu [Moayer, 1986]. The image is convolved through a Laplacian operator and the pixels whose intensity lies outside the range bounded by two thresholds are set to 0 and 1 respectively.

Ratha, Jain and Chen method: They used peak detection in the gray-level profiles along sections orthogonal to ridge orientation. A 16×16 oriented window centered around each pixel $[x,y]$ is considered. The gray-level profile is obtained by projection of the pixel intensities onto the central section. The profile is smoothed through a local averaging the peaks and the two neighboring pixels on either side of each peak constitute the foreground of the resulting binary images as in fig. 3.23.

Even though many approaches are proposed by researchers, contextual filtering based method proposed by Ó Gorman and Nickerson [Gorman, 1998] and Sherlock et al. [Sherlock , 1994] generates regular binary pattern. A good contextual based filter image can be locally thresholded to produce a regular binary ridge pattern.

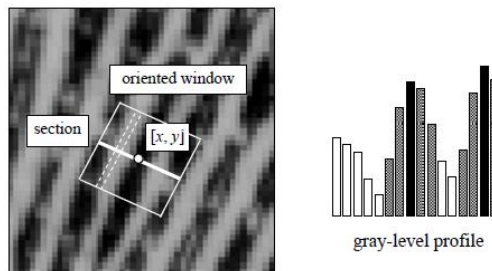


Fig.3.23. Gray-level profile obtained through projection of pixel intensities on the segment centered at $[x, y]$ and normal to the ridge orientation θ_{xy} .

3.2.6.2 *Thinning of Binarised image*

Minutiae detection is usually carried out by thinning the binarised image into a one pixel wide.

Morphology based operation: Method proposed by using successive deletion of pixels from all sides of the image. Each of the four sides are eroded away according to some set template (Fig.3.24) [Kasaei, 1997]. If the image matches with template, the middle pixel is removed. Initially the two north images are processed, and then the other compass points are overlaid. One all eight matrices have been sampled on the entire image; the process is repeated again on the newly formed image. Processing stops when no more pixels can be deleted.

Table 3.1. Thinning matrix

North	South	East	West
0 0 0	1 1 X	0 X 1	X X 0
X 1 X	X 1 X	0 1 1	1 1 0
X 1 1	0 0 0	0 X X	1 X 0
X 0 0	X 1 X	0 0 X	X 1 X
1 1 0	0 1 1	0 1 1	1 1 0
X 1 X	0 0 X	X 1 X	X 0 0

After achieving a binary skeleton, minutiae can be detected by computing a crossing number [Arcelli, 1984]. The crossing number $cn(\mathbf{p})$ of a pixel \mathbf{p} in a binary image is designed as the half of the sum of differences between pairs of adjacent pixels with 8-neighborhood of \mathbf{p} .

$$cn(\mathbf{p}) = \frac{1}{2} \sum_{i=1..8} |val(\mathbf{p}_{i \bmod 8}) - val(\mathbf{p}_{i-1})| \quad (3.17)$$

where $\mathbf{p}_0, \mathbf{p}_1, \dots, \mathbf{p}_7$ are the pixels belonging to an ordered sequence of pixels defining the 8-neighbourhood of \mathbf{p} and $val(\mathbf{p}) \in \{0,1\}$ is the pixel value. For an intermediate ridge point, $cn(\mathbf{p})=2$. $cn(\mathbf{p})=1$ for a termination minutia and $cn(\mathbf{p})=3$ for a bifurcation, crossover etc.

Other approaches in thinning are: Hung [Hung, 1993] used the algorithm by Arcelli and Baja [Arcelli, 1984]; Ratha et al. [Ratha, 1995] adopted a technique included in the HIPS library [Landy, 1984], Mehtre [Mehtre, 1993] employed the parallel algorithm described in Tamura [Tamura, 1978] and Coetzee and Botha [Coetzee, 1993] used the method by Baruch [Baruch, 1988]. In Ji et al. (2007) the skeleton is computed through a constrained PCNN (Pulse Coupled Neural Network) where the orientation image is used to constrain the thinning direction of PCNN thus allowing to reduce bothersome artifacts such as the short spikes that conventional thinning algorithms often produce.

3.3 Fingerprint Matching

Fingerprint matching involves the measurement of degree of similarity between two given fingerprints and to give a score between 0 and 1. Most of the fingerprint matching algorithms use feature matching rather than direct grayscale fingerprint image comparison. The fingerprint image acquired during enrollment is denoted as the template (\mathbf{T}) and the fingerprint to be matched as the input (\mathbf{I}). 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 (i.e., searching for an input fingerprint in a database of N fingerprints) can be implemented as a sequential execution of N one-to-one comparisons (verifications) between pairs of fingerprints. Matching can be simplified by indexing the fingerprint into many classes.

3.3.1 Correlation-Based Techniques

Let \mathbf{T} and \mathbf{I} be the two fingerprint images corresponding to the template and the input fingerprint, respectively. Then an intuitive measure of their diversity is the sum of squared differences (SSD) between the intensities of the corresponding pixels:

$$SSD(\mathbf{T}, \mathbf{I}) = \|\mathbf{T} - \mathbf{I}\|^2 = (\mathbf{T} - \mathbf{I})^T(\mathbf{T} - \mathbf{I}) = \|\mathbf{T}\|^2 + \|\mathbf{I}\|^2 - 2\mathbf{T}^T\mathbf{I} \quad (3.18)$$

where the superscript \mathbf{T} denotes the transpose of a vector. If the terms $\|\mathbf{T}\|^2$ and $\|\mathbf{I}\|^2$ are constant, the diversity between the two images is minimized when the cross-correlation (CC) between \mathbf{T} and \mathbf{I} is maximized:

$$CC(\mathbf{T}, \mathbf{I}) = \mathbf{T}^T\mathbf{I} \quad (3.19)$$

The cross-correlation is a measure of image similarity. The rotation and displacement of the image has to be considered and the similarity has to be calculated accordingly. Let $\mathbf{I}^{(\Delta x, \Delta y, \theta)}$ represent a rotation of the input image \mathbf{I} by an angle θ around the origin and shifted by Δx and Δy pixels in directions x and y respectively; then the similarity between the two fingerprint images \mathbf{T} and \mathbf{I} can be measured as :

$$S(\mathbf{T}, \mathbf{I}) = \max_{\Delta x, \Delta y, \theta} CC(\mathbf{T}, \mathbf{I}^{(\Delta x, \Delta y, \theta)}) \quad (3.20)$$

3.3.2 Minutiae-Based Matching Method

3.3.2.1 Problem formulation

Let \mathbf{T} and \mathbf{I} be the representation of the template and input fingerprints, respectively. The feature vector is of variable length and the elements are fingerprint minutiae. Each minutia is represented by a number of attributes, including its location in the fingerprint image, orientation, type etc... Each minutia is represented as a triplet $\mathbf{m}=\{x,y,\theta\}$ that indicates the x,y minutia location coordinates and the minutia angle θ

$$\mathbf{T} = \{\mathbf{T}_1, \mathbf{T}_2, \dots, \mathbf{T}_m\}, \quad \mathbf{m}_i = \{x_i, y_i, \theta_i\}, i = 1 \dots m$$

$$\mathbf{I} = \{\mathbf{I}'_1, \mathbf{I}'_2, \dots, \mathbf{I}'_n\}, \quad \mathbf{m}'_j = \{x'_j, y'_j, \theta'_j\}, j = 1 \dots n, \quad (3.21)$$

where m and n denote the number of minutiae in \mathbf{T} and \mathbf{I} , respectively.

The matching between minutia \mathbf{m}'_j in \mathbf{I} and a minutia \mathbf{m}_i in \mathbf{T} is maximum if the spatial distance (sd) between them is smaller than a given tolerance r_0 and the direction difference (dd) between them is smaller than an angular tolerance, θ_0 :

$$sd(\mathbf{m}'_j, \mathbf{m}_i) = \sqrt{(x'_j - x_i)^2 + (y'_j - y_i)^2} \leq r_0 \quad \text{and} \quad (3.22)$$

$$dd(\mathbf{m}'_j, \mathbf{m}_i) = \min(|\theta'_j - \theta_i|, 360^\circ - |\theta'_j - \theta_i|) \leq \theta_0 \quad (3.23)$$

Aligning of the two fingerprints is a mandatory step to maximize the number of matching minutiae. Aligning two fingerprints require displacement, rotation and other geometrical transformations:

- *Scale* has to be considered when the resolution of the two fingerprints varies (e.g., the two fingerprint images have been taken by scanners operating at different resolutions).
- Other *distortion-tolerant* geometrical transformations could be useful to match minutiae in case one or both of the fingerprints is affected by severe distortions.

Let $map(\cdot)$ be the function that maps a minutia \mathbf{m}'_j (from \mathbf{I}) into \mathbf{m}''_j according to a given geometrical transformation; for example, by considering a displacement of $[\Delta x, \Delta y]$ and a counterclockwise rotation θ around the origin:

$$map_{\Delta x, \Delta y, \theta}(\mathbf{m}'_j = \{x'_j, y'_j, \theta'_j\}) = \mathbf{m}''_j = \{x''_j, y''_j, \theta''_j + \theta\}$$

where

$$\begin{bmatrix} x''_j \\ y''_j \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x'_j \\ y'_j \end{bmatrix} + \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix}$$

Let $mm(\cdot)$ be an indicator function that returns 1 in the case where the minutiae \mathbf{m}''_j and \mathbf{m}_i match according to Equations (3.22) and (3.23):

$$mm(\mathbf{m}''_j, \mathbf{m}_i) = \begin{cases} 1, & sd(\mathbf{m}''_j, \mathbf{m}_i) \leq r_0 \quad \text{and} \quad dd(\mathbf{m}''_j, \mathbf{m}_i) \leq \theta_0 \\ 0, & \text{otherwise} \end{cases} \quad (3.24)$$

Then the matching problem can be formulated as

$$\underset{\Delta x, \Delta y, \theta, P}{\text{maximize}} \sum_{i=1}^m mm(map_{\Delta x, \Delta y, \theta}(\mathbf{m}'_{P(i)}), \mathbf{m}_i), \quad (3.25)$$

where $P(i)$ is an unknown function that determines the pairing between \mathbf{I} and \mathbf{T} minutiae; in particular, each minutia has either exactly one mate in the other fingerprint or has no mate at all:

1. $P(i) = j$ indicates that the mate of the \mathbf{m}_i in \mathbf{T} is the minutia \mathbf{m}'_j in \mathbf{I} .
2. $P(i) = \text{null}$ indicates that minutia \mathbf{m}_i in \mathbf{T} has no mate in \mathbf{I} .

3. A minutia \mathbf{m}'_j in \mathbf{I} , has no mate in \mathbf{T} if $P(i) \neq j \forall i = 1 \dots m$.
4. $\forall i=1 \dots m, k=1 \dots m, i \neq k \Rightarrow P(i) \neq P(k)$ or $P(i) = P(k) = \text{null}$

3.3.2.2 Similarity score

In the manual matching performed by forensic experts, the number of matching minutiae is the main output of the comparison. Automatic matching system converts this number into a similarity score by normalizing the number of matching minutiae by the average number of minutiae in \mathbf{T} and \mathbf{I} :

$$\text{score} = \frac{k}{(m+n)/2} \quad (3.26)$$

k is the number of matching minutiae.

The definition of optimal rules for combining various similarity contributions into a single score can be complex. Learning-based techniques where the rule and its parameters are optimized to best separated genuine and imposter scores have been proposed by Jea and Govindaraju [Jea, 2005], Srinivasan et al. [Srinivasan, 2006], Feng [Feng, 2008], Lumini and Nanni [Lumini, 2008].

3.4 Conclusion

In this chapter, various steps in fingerprint recognition techniques were discussed. The major ideas behind fingerprint image normalization, segmentation, contextual based fingerprint enhancement, FFT based fingerprint etc. were discussed. The idea behind fingerprint matching and matching score computation were explained. In the next chapter development of a novel fingerprint recognition system based on global singularity features is discussed. The state-of-the art minutiae based technique is used to compare the performance of the system developed.

Chapter 4

Development of a Novel Global Singularity Feature Based Fingerprint Recognition System

In this chapter a novel singularity based fingerprint image representation is developed. In the first phase of the work, conventional minutia based fingerprint recognition system is studied and a fingerprint recognition system is implemented. Minutiae based fingerprint recognition systems require accurate algorithms for minutiae extraction as well as in the matching phases. The drawbacks of minutiae based recognition systems are explored and based on the global singularity features; a new fingerprint recognition system is developed. The feature vector length is reduced to just 16 elements derived from a fingerprint polygon formed from the singular points and the fingerprint as against 70 to 100 elements required for the conventional minutiae method. This can reduce the feature database size considerably and hence search space during matching. The system achieved 100% recognition rate.

4.1 Introduction

Human fingerprint images are very rich in information content. The image information content in a fingerprint image can be classified into three: Level 1 features are called as global feature which deals with the singular points, their orientation and types. Level 2 features are referred to as local characteristics in which about 150 different local ridge characteristics are subjected. Two most prominent level 2 characteristics called minutiae are ridge endings and ridge bifurcations. In the level 3 features, fine level intra-ridge details can be detected. Width, shape, curvature, edge contours of ridges, and also positions and shapes of the sweat considered highly distinctive. Minutia feature based identification is the most common and the accurate method adopted by the most of the automatic fingerprint identification systems. Minutiae based system require fast and accurate algorithms to achieve high recognition rate. The main disadvantage of minutiae based system is the computational complexity in feature extraction and matching. Also the feature extracted will contain 70-100 minutia points with the position and orientation as the feature template elements.

In this research, a conventional minutia based system is developed and implemented. The computation complexity in feature extraction phase as well as in the matching phase is explored. A novel global singularity based system is developed and implemented in the next phase. In this method the fingerprint features are reduced to a 16 element vector consisting of polygonal angle, distance, area and ridge counts. Hence the feature vector size and the feature extraction time required is reduced significantly. The two methods are compared. The new method is recommended for access control applications.

4.2 Minutiae based fingerprint recognition technique

The different steps in the minutia based fingerprint recognition technique proposed in this work are as follows:

1. Fingerprint Normalization

2. Ridge Orientation Estimation
3. Estimation of Local Ridge Frequency
4. Fingerprint Image Segmentation
5. Fingerprint Enhancement
6. Fingerprint Thinning
7. Minutiae Detection
8. Core Detection and Feature vector formation
9. Fingerprint Matching

4.2.1 Fingerprint Normalization

To reduce the variations in the gray level values along ridges and valleys the fingerprint images should be first normalized. Hence an input fingerprint image is normalized to have a specified mean and variance.

Let $I(i, j)$ denote the gray level value at pixel (i, j) , M and VAR denote the estimated mean and variance of I , respectively, and $N(i, j)$ denote the normalized gray-level value at pixel (i, j) . The normalized image is found out as per the equation:

$$N(i, j) = \begin{cases} M_0 + \sqrt{\frac{VAR_0 \cdot (I(i, j) - M)^2}{VAR}} & \text{if } I(i, j) > M \\ M_0 - \sqrt{\frac{VAR_0 \cdot (I(i, j) - M)^2}{VAR}} & \text{otherwise} \end{cases} \quad (4.1)$$

Where M_0 and VAR_0 are the desired mean and variance values respectively. Fig. 4.1 shows the results of normalizing the fingerprint image. In this case, the image is normalized to have zero mean and unit standard deviation.

4.2.2 Ridge orientation estimation

Directional field or orientation field shows the coarse structure or basic shape of a fingerprint [Bazen, 2002] which gives the global information about a fingerprint image. It is defined as the local orientation of the ridge-valley structures. By computing directional field, singularities can be efficiently located.



Fig. 4.1. a) Fingerprint image b) Normalized image

M. Kass and Witkin, [Kass, 1987] introduced the gradient based method and was adopted by fingerprint researchers [Jain, 1997; Rao, 1992, Perona, 1998]. This method is used in this work. The steps in calculating the orientation of fingerprint image are as follows:

1. A block of size $w \times w$, centered at pixel $[i,j]$ is selected.
2. For each pixel in the block, compute the gradients G_x and G_y using the Sobel operator in the horizontal and vertical direction using the spatial mask

$$\begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}$$

and

$$\begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

3. The fingerprint ridge orientation is computed as given in chapter 3, section

3.2.1, as per equations 3.1-3.6. The average gradient direction ϕ of the block is given by equation 3.6 as:

$$\phi = \frac{1}{2} \angle(G_{xx} - G_{yy}, 2G_{xy}) \quad (4.2)$$

where,

$$\begin{aligned} G_{xx} &= \sum_w G_x^2 \\ G_{yy} &= \sum_w G_y^2 \\ G_{xy} &= \sum_w G_x G_y \end{aligned} \quad (4.3)$$

are estimates for the variances and crosscovariances of gradient, G_x and G_y , averaged over the window w . Fig. 4.2 shows the results of orientation field extraction of fingerprint images. Here fingerprint orientation images are calculated for $w=8, 16, 32$ and 64 . As the block size increases, orientation structure of the fingerprint image become more and more visible, even for poor quality images. But computational complexity is high for larger windows and hence an optimum window of size 16 to 32 is recommended.

4.2.3 Estimation of Local Ridge Frequency

The local ridge frequency (or density) $f(x,y)$ at $[x, y]$ 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} . The method used here was proposed by Hong et al. [Hong, 1998] in which local ridge frequency is estimated by counting the average number of pixels between two consecutive peaks of gray-levels along the direction normal to the local ridge orientation.

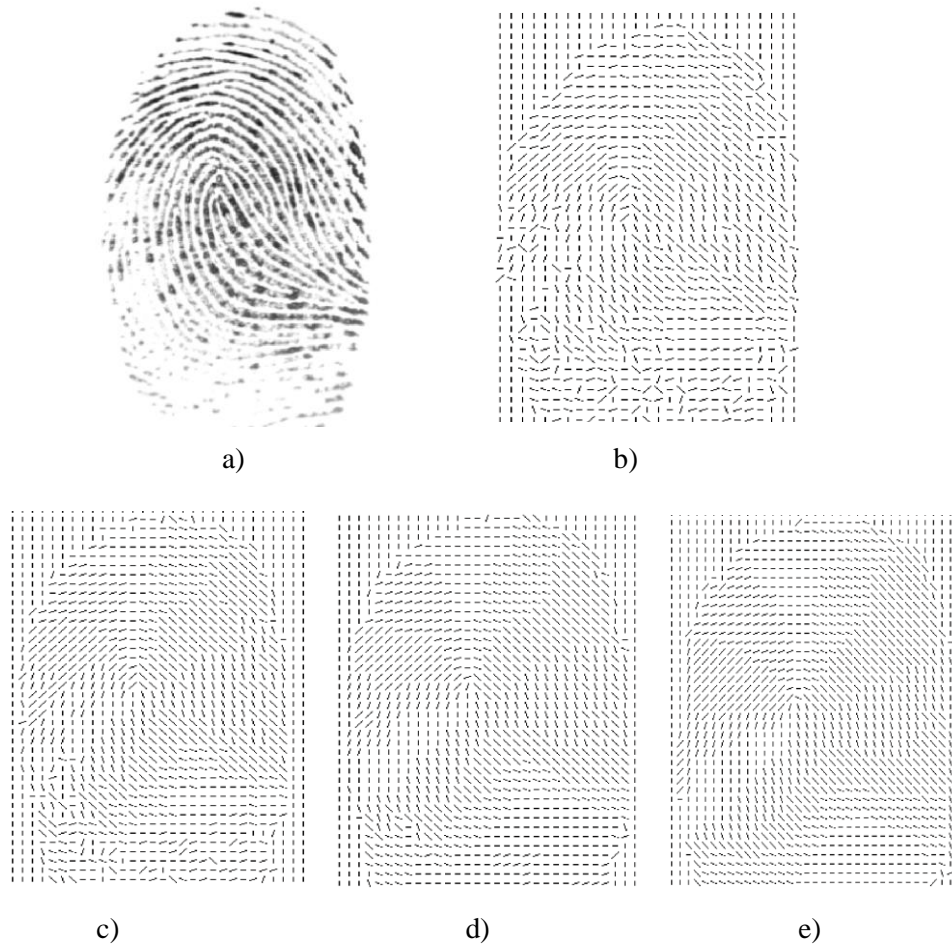


Fig. 4.2. a) Fingerprint image b) Orientation field with $w=8$ c) Orientation field with $w=16$ d) Orientation field with $w=32$ e) Orientation field with $w=64$

The steps in computing ridge frequency are as follows:

Let N be the normalized image and ϕ be the orientation image.

1. Divide N into blocks of size $w \times w$ (16×16)

2. For each block centered at pixel (i, j) , compute an oriented window of size $l \times w$ (32×16) that is defined in the ridge coordinate system as shown in fig.4.3.
3. For each block centered at pixel (i, j) , compute the x -signature, $X[0], X[1], \dots, X[l-1]$ of the ridges and valleys within the oriented window, where

$$X[k] = \frac{1}{w} \sum_{d=0}^{w-1} G(u, v) \quad k = 0, 1 \dots l - 1 \quad (4.4)$$

$$u = i + \left(d - \frac{w}{2}\right) \cos \phi(i, j) + \left(k - \frac{l}{2}\right) \sin \phi(i, j) \quad (4.5)$$

$$v = j + \left(d - \frac{w}{2}\right) \sin \phi(i, j) + \left(k - \frac{l}{2}\right) \cos \phi(i, j) \quad (4.6)$$

If no minutiae and singular points appear in the oriented window, the x -signature forms a discrete sinusoidal shape wave, which has the same frequency as that of the ridges and valleys in the oriented window. Therefore, the frequency of ridges and valleys can be estimated from the x -signature. Let $T(i, j)$ be the average number of pixel between two consecutive peaks in the x -signature, then the frequency, $\Omega(i, j)$, is given by:

$$\Omega(i, j) = \frac{1}{T(i, j)} \quad (4.7)$$

4. For a fingerprint image the frequency of the ridges and valleys in a local neighborhood lies in a certain range. For a 500 dpi image this range is in between $1/3$ to $1/25$. Therefore if the estimated value of the frequency is out of this range, then a value of -1 is assigned to indicate that a valid frequency is not obtained.
5. For the block in which minutiae and/or singular points appear and/or ridges and valleys are corrupted do not form a sinusoidal shaped wave, the frequency values for these blocks need to be interpolated from the

frequency of the neighboring blocks which have a well-defined frequency. The interpolation can be done as follows:

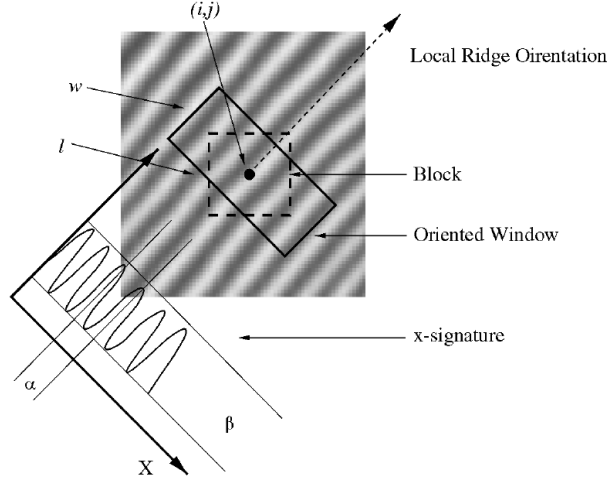


Fig 4.3. Oriented window and x-signature

$$i. \Omega'(i, j) = \begin{cases} \Omega(i, j) & \text{if } \Omega(i, j) \neq -1 \\ \frac{\sum_{u=-\frac{w\Omega}{2}}^{\frac{w\Omega}{2}} \sum_{v=-\frac{w\Omega}{2}}^{\frac{w\Omega}{2}} W_g(u, v) \mu(\Omega(i-uw, j-vw))}{\sum_{u=-\frac{w\Omega}{2}}^{\frac{w\Omega}{2}} \sum_{v=-\frac{w\Omega}{2}}^{\frac{w\Omega}{2}} W_g(u, v) \mu(\Omega(i-uw, j-vw)+1)} & \text{otherwise} \end{cases} \quad (4.8)$$

where

$$\mu(x) = \begin{cases} 0 & \text{if } x \leq 0 \\ x & \text{otherwise} \end{cases}$$

$$\delta(x) = \begin{cases} 0 & \text{if } x \leq 0 \\ 1 & \text{otherwise} \end{cases}$$

w_g is a discrete Gaussian kernel with mean and variance of zero and nine respectively, and $w\Omega = 7$ is the size of the kernel.

- ii. If there exist at least one block with the frequency value of -1, the swap Ω and Ω' and go to step (i).
6. Inter-ridge distances change slowly in a local neighborhood. A low-pass filter can be used to remove the outliers in f' :

$$F(i, j) = \sum_{u=-w\Omega/2}^{w_1/2} \sum_{v=-w\Omega/2}^{w_1/2} w_1(u, v)\Omega'(i - uw, j - vw) \quad (4.9)$$

where w_1 is a two dimensional low-pass filter with unit integral and $w_i=7$ is the size of the filter.

Fig. 4.4 shows the fingerprint frequency pattern image with $w=16$. The dark portion shows low ridge count area while white area indicates high ridge count area.

4.2.4 Fingerprint Image Segmentation

Variance thresholding method proposed by Mehtre [Mehtre, 1993] is used here. The equation for variance of a block $w \times w$ is given:

$$v(k) = \frac{1}{w^2} \sum_{i=0}^{w-1} \sum_{j=0}^{w-1} [p(i, j) - m(k)]^2 \quad (4.10)$$



Fig. 4.4 Fingerprint image and frequency image

where $v(k)$ is the variance for the block k , $p(i,j)$ is the gray level value at pixel (i,j) , and $m(k)$ is the mean gray level value for the block k . If the variance is less than the global threshold, then the block is assigned to be a background region; otherwise it is assigned to be part of the foreground. Fig. 4.5 shows the variance thresholding based segmented image with $w=16$ and $w=8$ for fingerprint images of poor quality and good quality. A block of size $w=8$ will segment the image of low quality without losing much ridge information.

4.2.5 Fingerprint Enhancement

As discussed in chapter 3, section 3.2.5, contextual filtering is the successful filtering method developed by researchers based on direction information of the subject filtered. Also, as fingerprints are well-defined highly oriented patterns, directional filtering is the best choice for fingerprint enhancement. In this work enhancement using Gabor filter [Hong, 1998], which is a directional filter is designed and implemented.

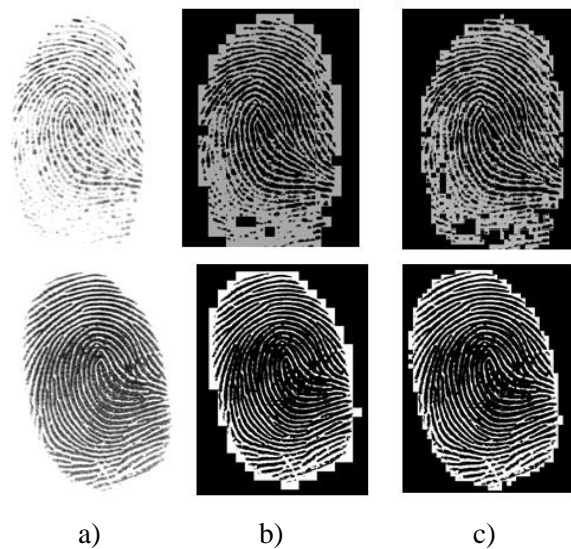


Fig.4.5 a) Fingerprint image b) Segmented normalized image for a window $w=16$ c) Segmented normalized image for a window $w=8$

4.2.5.1 Gabor filter based fingerprint enhancement:

A Gabor filter function is basically a 2D Gaussian function modulated by sinusoids. The filter function is given by:

$$g(x, y; \theta, f) = \exp\left\{-\frac{1}{2}\left[\frac{x_\theta^2}{\sigma_x^2} + \frac{y_\theta^2}{\sigma_y^2}\right]\right\} \cdot \cos(2\pi f \cdot x_\theta) \quad (4.9)$$

where

$$x_\theta = x \cos\theta + y \sin\theta \quad (4.10)$$

$$y_\theta = -x \sin\theta + y \cos\theta \quad (4.11)$$

and θ is the orientation of the Gabor filter, f is the frequency of the cosine wave, σ_x and σ_y are the standard deviations of the Gaussian envelope along the x and y axes, respectively. The visualization of the filter function is shown in fig. 4.6.

The filter is implemented and the fingerprint images are enhanced successfully. Fig. 4.7 and Fig 4.8 show some of the enhanced images with input images of various qualities. Fig. 4.7a is of moderate quality fingerprint image in which ridges are extracted very well, Fig. 4.7c and Fig.4.8a are poor quality images due to dry skin conditions and the ridges are retrieved reasonably well. Fig. 4.7d is a good quality image and the ridges are extracted well.

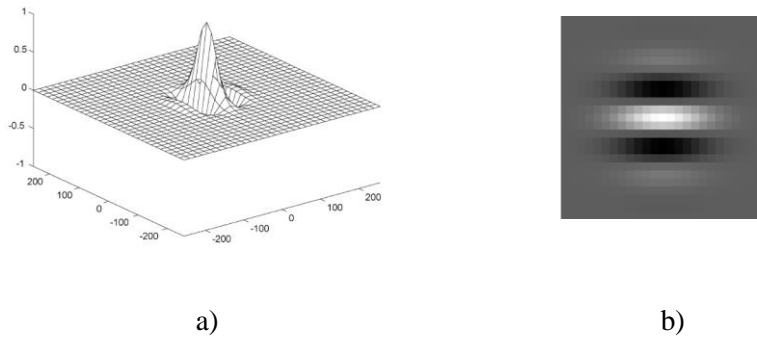


Fig. 4.6 a) An even-symmetric Gabor filter with $f=10$ b) the corresponding filter coefficient values



Fig. 4.7 a) and c) Fingerprint images and b) and d) Gabor filter based enhanced images

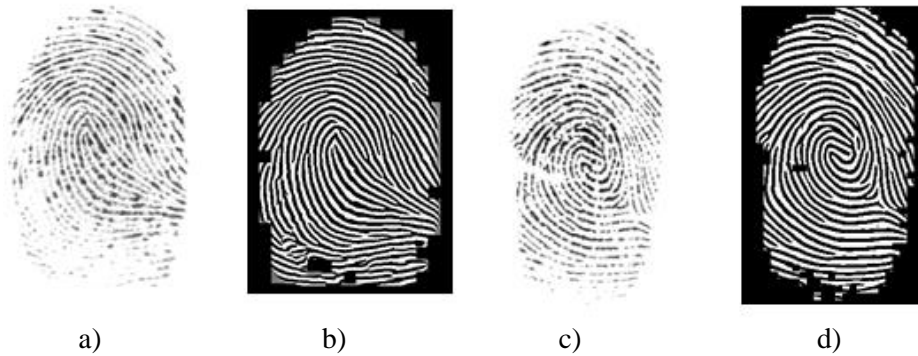


Fig. 4.8 a) and c) Fingerprint images and b) and d) Gabor filter based enhanced images

4.2.6 Fingerprint Thinning

The filtered image is thinned to extract minutiae features. Before thinning, the enhanced image is binarised using a global threshold value. The binarised image is subjected to morphological process of thinning. In this research, thinning is done by selective deletion of pixels using a set template [Table 4.1] proposed by Kasaei [Kasaei, 1997] as discussed in section 3.2.6 of chapter 3. Each of the masks of the template is overlaid on the binarised image and the nine image pixels are

compared with the template. If the pixels in the image match with the template, the middle pixel is removed. Next the second north mask and the other masks in the compass point order is applied on the same pixel. This is repeated for every pixel in the fingerprint binarised image to get a new fingerprint image. Once all the eight masks have been used on the entire image; the process is repeated on the newly formed image. Processing stops when no more pixels can be deleted. Fig. 4.9 gives the result of fingerprint thinning using this method.

Table 4.1. Template for Thinning

North	South	East	West
0 0 0	1 1 X	0 X 1	X X 0
X 1 X	X 1 X	0 1 1	1 1 0
X 1 1	0 0 0	0 X X	1 X 0
X 0 0	X 1 X	0 0 X	X 1 X
1 1 0	0 1 1	0 1 1	1 1 0
X 1 X	0 0 X	X 1 X	X 0 0

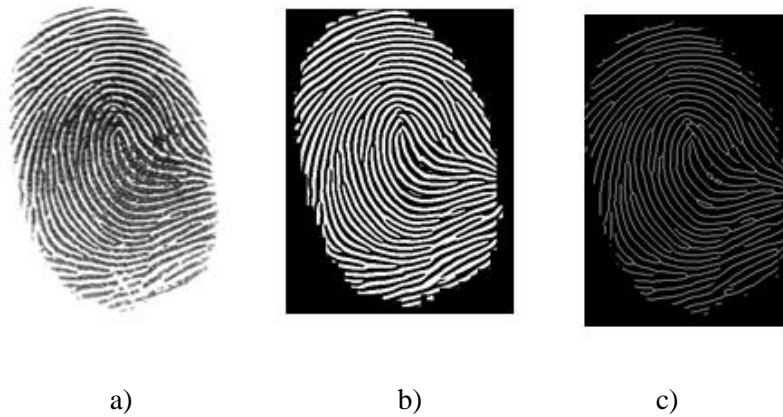


Fig. 4.9 a) Fingerprint image b) Binarised image c) Thinned image

4.2.7 Minutiae Detection

After obtaining a thinned image, minutiae is obtained using the crossing number computed from the thinned image as per the equation:

$$cn(p) = \frac{1}{2} \sum_{i=1..8} |val(p_{i \bmod 8}) - val(p_{i-1})| \quad (4.12)$$

where p_0, p_1, \dots, p_7 are the pixels belonging to an ordered sequence of pixels defining the 8-neighbourhood of p and $val(p) \in \{0,1\}$ is the pixel value. For an intermediate ridge point, $cn(p)=2$. $cn(p)=1$ for a termination minutia and $cn(p)=3$ for a bifurcation & crossover.

4.2.8 Core Detection and Feature Vector formation

4.2.8.1 Core Detection

Core point is defined as the reference point to define the minutiae feature map. Feature minutia vector is a triplet, consisting of the distance between the minutiae point and the core point, their orientation angle difference with respect to the core point, and the type of the minutia point. Hence core point has to be determined accurately to define the reference point. In this work, the core points are determined using the coherence computed from the orientation field. The coherence of the squared gradients is given by

$$Coh = \frac{|\sum_w G_{s,x}, G_{s,y}|}{\sum_w |(G_{s,x}, G_{s,y})|} \quad (4.13)$$

where $\begin{bmatrix} G_{s,x} \\ G_{s,y} \end{bmatrix} = \begin{bmatrix} G_x^2 - G_y^2 \\ 2G_x G_y \end{bmatrix}$; G_x and G_y are the squared gradients along the x and y direction.

$$Coh = \frac{\sqrt{(G_{xx} - G_{yy})^2 + 4G_{xy}^2}}{G_{xx} + G_{yy}} \quad (4.14)$$

The coherence is computed for the entire image and fig. 4.10 shows coherence images for various fingerprints with the block size taken as $w=16$. The point where the coherence shows a low value or the point where the orientation field has low strength is shown as dark points in the image. In the case of a fingerprint with delta(s), there exists more than one low coherence values as shown in Fig. 4.11, which corresponds to delta type singularity regions. Since core point is defined as the north most point of the inner loop, north most point of the low coherence pixel is identified as the core point. For a whorl type of fingerprint image, normally two core points exists and the topmost low value coherence point is taken as the core point as given in Fig. 4.10.

4.2.8.2 Feature vector formation

Feature minutiae vector is defined as the triplet given by:

$$m = [d_i, \theta_i, t_i]^T \quad (4.15)$$

where, d_i is the distance of the i^{th} minutia from the core point, θ_i is the angular difference between the i^{th} minutia and the core point and t_i is the type of the minutia point.

In this work the feature vector is taken from an area defined by an ellipse with the center as the core point detected and the major axis oriented along the direction of the core point. The feature images are shown in fig. 4.12 and fig. 4.13. In the figure the distance, d_i of i^{th} end point minutia is shown in red line and the distance d_i of the i^{th} bifurcation point minutia is shown in green line. The elliptical region of interest is also shown in the figure. The maximum elliptical distance from core point to the minutia point is the major axis of the ellipse, which is taken as 150 pixel units in a 500 dpi resolution scanner. The angular difference θ_i of the i^{th} minutia is the difference in orientation angle between the core point and the i^{th} minutia. The value of θ_i ranges between 0 and 180°. t_i is taken as 1 for an end point minutia and 2 for a bifurcation point minutia.

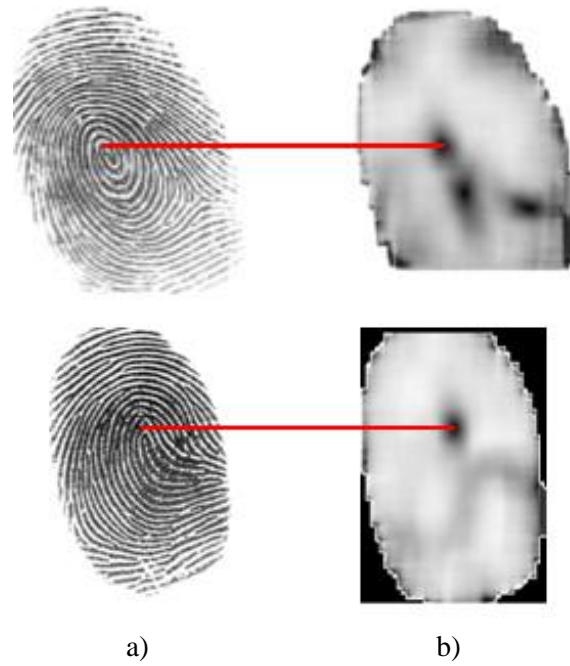


Fig. 4.10 a) Fingerprint and b) Corresponding Coherence image

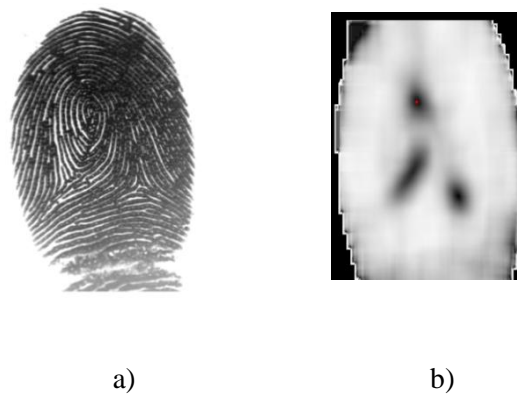


Fig. 4.11 a) Fingerprint and b) coherence image which shows core and deltas

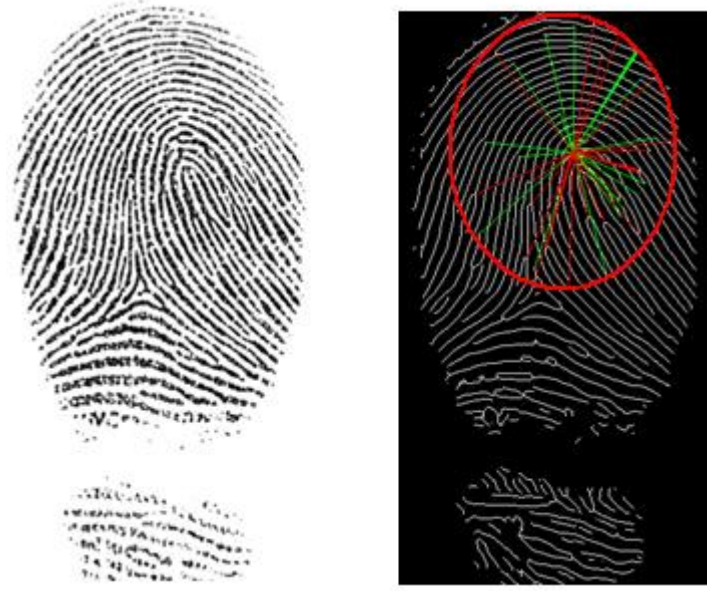


Fig. 4.12 a) Fingerprint image b) Feature image with elliptical ROI

4.2.9 Matching score determination

Let T and I be the fingerprint minutiae template and input image feature template such that $T = \{T_1, T_2, \dots, T_m\}$, $T_i = \{d_i, \theta_i, t_i\}$, $i = 1 \dots m$

and $I = \{I_1, I_2, \dots, I_n\}$, $I_j = \{d'_j, \theta'_j, t'_j\}$, $j = 1 \dots n$,

where m and n denote the number of minutiae in T and I .

Matching may be defined as computing the spatial distance $SD(\cdot)$ and directional distance $DD(\cdot)$ between the template and input image feature vector

$$SD(m'_j, m_i) = |d'_j - d_i| \leq r_0 \quad (4.16)$$

$$DD(m'_j, m_i) = \min(|\theta'_j - \theta_i|, 360^\circ - |\theta'_j - \theta_i|) \leq \theta_0 \quad (4.17)$$

The sum of spatial distance $SSD(.)$ and sum of directional distance $SDD(.)$ for each minutia is given by

$$SSD = \frac{1}{N} \sum_{i=1}^N SD \leq R_0 \quad (4.18)$$

$$SDD = \frac{1}{N} \sum_{i=1}^N DD \leq \vartheta_0 \quad (4.19)$$

If $R_0 + \vartheta_0 \leq Thr$, fingerprints matches, where Thr is the set threshold.

4.2.10 Performance estimation parameters for algorithm, FAR and FRR

The performance of the algorithm is estimated using the False Accept Rate (FAR) and False Reject Rate (FRR) and is defined as follows:

False Accept Rate (FAR) is defined as the number of false acceptances of impostor fingerprints to the number of identification attempts and is given as:

$$FAR = \frac{\text{No. of imposter test images accepted}}{\text{Total no. of identificatin attempts}} \quad (4.20)$$

False Reject Rate (FRR) is defined as the ratio of number of false rejections of a genuine fingerprint images to the number of identification attempts and is given as:

$$FRR = \frac{\text{No. of genuine fingerprint images rejected}}{\text{Total no. of identification attempts}} \quad (4.21)$$

4.2.11 Implementation of the Algorithm

4.2.11.1 Database used

In this research, minutia based fingerprint matching algorithm is tested with the FVC2004 DB1_A database. The database consists of 800 fingerprint images of 100 persons (100×8), of size 640×480 with a scanning resolution of 500dpi. 3 images of each person are considered to form the feature template

vector, by taking the average distance and angular difference of common minutia points. The other 500 images in the database are used to test the performance of the algorithm.

4.2.11.2 Implementation

The fingerprint image is first normalized using equation 4.1 to have a unit standard deviation and zero mean. Directional field estimation is carried out for the normalized image with the block size $w = 16$ which shows a consistency in the orientation. In the next step, ridge frequency is estimated for blocks of size $w = 16$, as ridge frequency is a parameter for contextual filtering using Gabor filter. Gabor filter function is generated with the filter parameter values $\sigma_x = \sigma_y = 4$, and the fingerprint image is filtered using the orientation and ridge frequency information. The results of filtering on images with different qualities are shown in fig. 4.7 and 4.8. As mentioned in section 4.5, a good fingerprint ridges extraction is obtained even for low quality images in the database. The enhanced image is then binarised using global threshold method. For thinning 8-masks shown in Table 4.1 is used iteratively to remove the central pixels. The number of iteration is set to 4 with which the image is skeletonized to one pixel wide ridges. In the skeletonized image false bifurcations and end points are removed by applying the heuristics that the bifurcations and end points may not occur closely. The closeness limit for bifurcations and endpoints is set to a 7 pixel circular distance within which the bifurcations and endpoints are considered as false minutiae. The minutiae are then extracted by calculating the crossing number for the entire image. Crossing number of 1 corresponds to an endpoint minutia and crossing number of 2 corresponds to a bifurcation minutia. The minutia position and the orientation of the minutia point are taken and the feature vector is formed.

The core point region of the fingerprint is identified using coherence computed from the orientation image. With respect to the core point, an elliptical region which is comparable with the fingerprint silhouette is defined. The major and minor axis of the ellipse is defined as 150 and 120 pixel distances respectively in a 500 dpi scanner resolution image. Also the major axis is aligned with the core point orientation in order to make the region of interest rotation invariant. The minutiae feature vector is extracted from the region of interest by taking the

relative ridge distance, orientation angular difference and type. The feature vector of each person's fingerprint is stored by taking the corresponding minutia average of distances and angles of 3 fingerprints. The feature vectors are stored in a database.

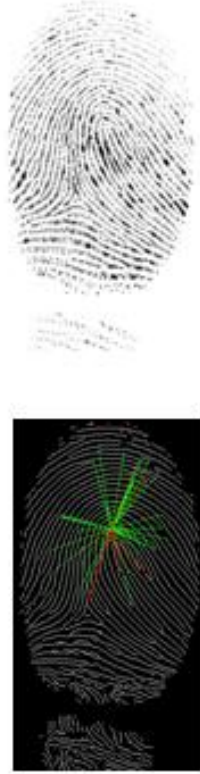
4.2.11.3 Results and discussion

The algorithm is developed and tested in a workstation based computer under MATLAB 7.10.0(R2010a) environment. The fingerprint images are downloaded from the FVC2004 website. The matching score is tested with the 500 images that are not taken for feature vector formation. Also the 300 images that are used for feature vector formation are also taken for validation. Table 4.2 shows portion of a typical feature template of size 50 to 100 of a fingerprint for the image in fig. 4.13. The minutia distance vector d_i will have a maximum value of 150 as the elliptical ROI has the major axis length of 150 pixel units and the angular vector θ_i will have a maximum value of 180° as the maximum orientation angle obtained is 180° . In this algorithm, the values of r_0 and θ_0 are taken as 10 pixel units and 10° respectively for matching case. The values of \mathbf{R}_0 and $\mathbf{\theta}_0$ will have a maximum value of 10 for $N=30$ number of matching points.

Hence for a threshold $Thr \leq 20$ can be taken as a perfect match of fingerprints. Based on this the genuine and imposter scores are computed and the values of FAR and FRR are calculated by taking the number of matching and non-matching cases. The FAR and FRR obtained are 0.75% and 0.875% respectively which shows a good result. The FAR and FRR can further be improved by taking more number of matching minutia points.

Minutiae based system are the conventional state-of-the-art method and gives accurate results. But the minutia extraction required several steps involving laborious computations. Fingerprint enhancement is an inevitable step to extract minutia without which false minutia points will result. Also matching module requires extra intelligence to obtain matching minutia pair. As the template size increases, searching for matching minutia in the entire feature template is time consuming. Hence a simple fingerprint feature representation as well as reduction in fingerprint template size is needed.

Table 4.2 Fingerprint feature vector



Sl.No.	d_i	θ_i	t_i
1	92.02717	3.898832	1
2	105.2093	12.95406	1
3	77.12976	7.640951	1
4	74.46476	9.307829	1
5	68.68042	10.3649	1
6	48.01042	7.380716	1
7	100.6479	46.82694	1
8	62.28965	26.7039	1
9	94.541	47.82218	1
10	100.1699	6.941953	1
11	117.3882	6.941953	1
12	13.92839	6.793244	1
13	47.43416	20.4502	2
14	114.4814	55.16297	2
15	86.97701	51.61639	2
16	40.49691	18.70201	2
17	13.0384	4.111522	2

Fig. 4.13 a) Fingerprint image and Feature Image

4.3 Development of a novel global singularity based fingerprint recognition

The minutia based fingerprint identification technique takes a major share in the fingerprint authentication technology. Also this method is well accepted around the forensic and criminal investigation bodies of most of the countries. The large numbers of fingerprint images, which are collected for criminal identification or in business for security purpose, continuously increase the importance of

automatic fingerprint identification systems. The purpose of Automatic Fingerprint identification System (AFIS) is to find out whether the individual represented by an incoming fingerprint image is the same as an individual represented by one of a large database of filed fingerprint images. Most of the AFIS can reach around 97% accuracy rates with a smaller database (less than 1000 fingerprint images) [Jain et al., 1997; Jain, et al. 2000]. However, the accuracy of identification is dropping down while the size of database is growing up. Nevertheless, the processing speed of AFIS also decreases if it involves a large fingerprint database. Hence it is inevitable to make the size of the fingerprint feature code to be as small as possible so that identification may be much easier.

In this research, a novel fingerprint representation based on level 1 fingerprint features is developed which reduce the fingerprint template size and hence the time for identification. A fingerprint is characterized by singularities-which are small number of regions where ridge lines form distinctive shapes: loop, delta or whorl (Fig.4.14). Singularities play a key role in classification of fingerprints [Henry, 1900; Cappelli, 2004] which sets fingerprints into a specific set. Classification eases the searching and in many of the AFIS, classification is the primary procedure adopted [Cappelli, 2000]. According to Galton-Henry classification scheme [Galton, 1892; Henry, 1900] there are four common classes of fingerprints: Arch, Left loop, Right Loop and Whorl (Fig. 4.15). Cappelli and Maltoni, [Cappelli, 2009] studied on the spatial distribution of fingerprint singularities and proposed a model for the four most common classes: Arch, Left loop, Right loop and Whorl. The model they proposed gives a clear indication of the fingerprint identity and is used here for identification with a sharp reference position.

4.3.1 Development of the method

In this research, fingerprint baseline, which is the line between the Distal and Intermediate phalangeal joint (first joint) line in the fingerprint, is taken as the reference line (Fig.4.16). This line can be clearly captured using optical based fingerprint scanners and using based a mask correlation technique.



Singularities are detected using directional field strength estimation and the ROI is formed with the singularities and the baseline.

Fig. 4.14 Types of singularities



Fig.4.15 Four common fingerprint classes

A polygon is formed as the ROI and feature vector is computed from the polygon. Since polygonal ROI is based only on the base line and main singular points – delta, core points, the template vector formed from the ROI with the distance and angle metrics will have only small dimension or length.



Fig.4.16 Fingerprint baseline

4.3.1.1 *Directional Field Estimation and strength computation*

As discussed in section 4.2.2, directional field can be computed using squared gradients as per equation 4.2. The window size taken for computation of directional strength is $W=16$ as discussed in section 4.2.2.

A sample image with directional field computed is shown in fig. 4.17.

Another metric, *Coherence*, which gives the strength of the orientation measures how well all squared gradient vectors share the same orientation. In this work singularities are located using Coherence computed using squared gradients as given in equation 4.14.



Fig.4.17. Fingerprint and its directional field image

4.3.1.2 Singularity detection and fingerprint classification scheme

From the orientation image, coherence image is calculated for various block sizes. For singularities, the strength of orientation is low and hence they have a low coherence value. It is found that as in the case of orientation image, as block size increases, the singularities are visible more and more clearly. To avoid computational overhead, a block size of size 16 is recommended. Fig.4.18 shows the coherence image formed for various block sizes for computing orientation with

singularities clearly indicated as the low coherence values (dark spots). Fig. 4.19 shows some of the examples of fingerprint classes and the singularities.

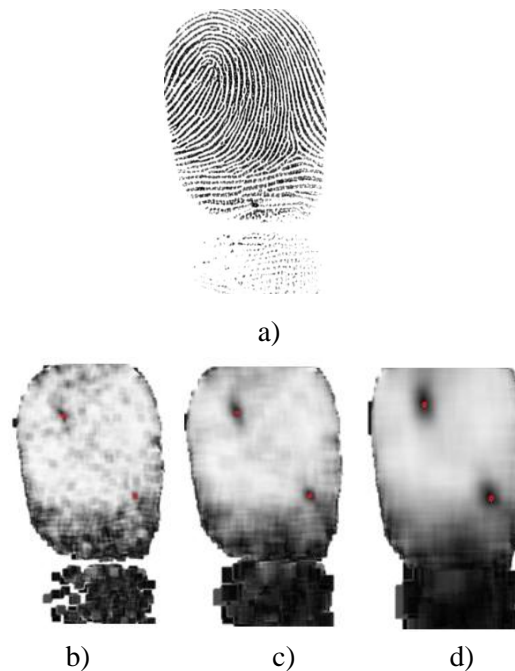


Fig.4.18. a) Fingerprint image b-d) coherence image with block size taken=8, 16 & 32

Based on fingerprint singularities and their relative location, fingerprints are classified into 7 types namely:

1. Left loop
2. Right Loop

3. Whorl without any Delta,
4. Whorl with one Delta
5. Whorl with two Deltas,
6. Arch
7. Tented Arch.

Fig. 4.19 shows fingerprint and coherence images of whorl with two deltas, right loop, left loop and arch.

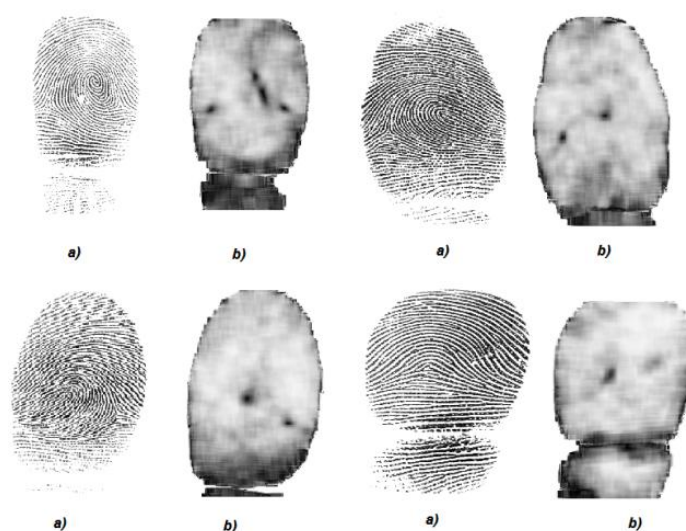


Fig.4.19 a) Fingerprint images of various classes and b) coherence images

4.3.1.3 Baseline detection and feature vector formation

4.3.1.3.1 Baseline detection

Hough Transform and other versions of Hough Transform based line identification are the most popular line identification technique used by image processing researchers [Atiquzzaman, 1992 ;Duda, 1972; Hough, 1962]. Guru et al. [Guru, 2003] have proposed a PCA based method for line detection. In all the cases

computational complexity is high. Also fingerprints are as such line patterns and hence identification of base line using Hough transform methods requires additional intelligence. In this work base line is detected using a correlation method as per the following steps:

1. Since baseline falls in the lower portion of the fingerprint image, computation for line identification needs to be done only in the lower portion of fingerprint image. To be exact, identification can be done below the centroid of the segmented fingerprint image
2. Binary masks of sizes from 200×50 to 200×3 are defined. Mask of 200×50 is to detect most slanted base line (about 23°) and 200×3 is to detect a horizontal line. A portion of the mask of size 200×50 is shown in Table 4.2.
3. Find the normalized cross-correlation peak between each masks and the fingerprint regions using

$$S = M \otimes F$$
 where S is the correlation peak, M and F are mask and fingerprint regions respectively.
4. If $S \geq T$, a threshold peak, presence of base line is identified and the baseline is drawn with reference to the mask direction

Table 4.3. Base line detection mask

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0
0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0
0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

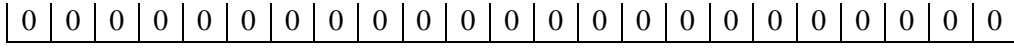


Fig. 4.20 shows the fingerprint base line detected for various image orientations.



Fig.4.20 Baseline detected for various fingerprint orientations

4.3.1.3.2 Definition of the novel fingerprint structure and Feature vector formation

From the singularity points identified and baseline detected, a polygon is formed with the baseline. The singular points are taken as the top vertices of the polygon. From the top vertices, perpendiculars are dropped to the baseline which completes a polygon. Based on the different types of fingerprints, the number of sides of polygon varies. For example, a whorl type fingerprint with two deltas will have a polygon of 6 sides. Fig.4.21 shows some of the polygons formed. The polygon thus formed is invariant to rotation. In general, feature vector describing the polygon is given by

$$F = (d_i, \theta_i, A_i, T_i, r_i)^T \quad (4.22)$$

where d is the distance metric, θ is the angle metric, A is the area of the polygon, T is the type of the fingerprint/polygon and r is the ridge counts.

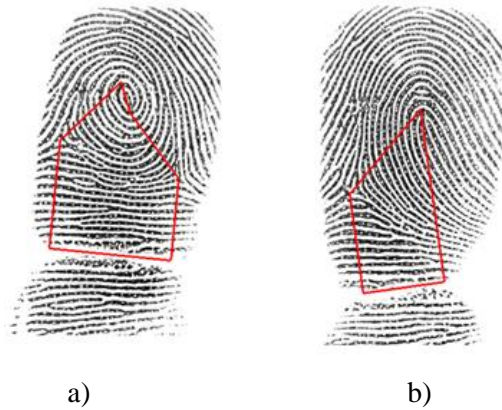


Fig. 4.21 Fingerprint polygons formed with a) whorl type and b) loop type fingerprints

The feature vector thus formed is a 16 element vector as shown below and has the interpretations in Fig.4.22 and Fig. 4.23.

$$[d_{cc}, d_{cb}, d_{cdr}, d_{dbr}, d_{bb}, d_{dbl}, d_{cdl}, \theta_C, \theta_{DR}, \theta_{DL}, \theta_{CC}, A, T, r_{CD}, r_{CB}, r_{DB}]^T$$

d_{cc} – Core to Core distance in a whorl type fingerprint found out using distance formula

d_{cb} – Core to base distance found out using distance formula

d_{cdr} – Core to delta distance along right found out using distance formula

d_{dbr} – Delta to base distance along right found out using distance formula

d_{bb} – Base to Base distance along the base line found out using distance formula

d_{dbl} – Delta to base distance along left found out using distance formula

d_{cdl} – Core to delta distance along left found out using distance formula

θ_C – Core angle found out using the angle between two lines formula

θ_{DR} – Core to delta angle along right found out using the angle between two lines formula

θ_{DL} – Core to delta angle along left found out using angle between two lines formula

θ_{CC} – Core to core convex angle found out using angle between two lines formula

A - Area of the polygon given by the standard polygon area formula knowing co-ordinates

T – Type of the fingerprint which depends on the relative position of singularities

r_{CD} – Ridge count between core and delta found out using the projection of the line between the two vertices, core and delta

r_{CB} – Ridge count between core and base found out using the projection of the line between the two vertices, core and base

r_{DB} – Ridge count between delta and base found out using the projection of the line between the two vertices, delta and base.

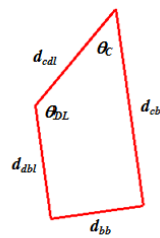


Fig.4.22. Fingerprint image polygon of right loop image



Fig.4.23. Fingerprint image polygon of whorl with two deltas

4.3.1.4 Fingerprint Matching Score

Matching is represented by a numerical score which shows how much the template obtained from the input image (I) matches with the existing fingerprint template (F). Any fingerprint algorithm compares two given fingerprint template elements and returns either a degree of similarity (difference being $<$ threshold) or a binary decision (match=1, mismatch=0). The matching score which is a number in the range from 0 to 1 is calculated as the ratio of the number of matched features to the total number of features.

In this work a simple matching score based on the Euclidean distance is formulated. The steps for computing the matching score are:

1. Compare the type T of the input fingerprint template with that of the feature template. If they are not matching, match score between the two templates is zero, else go to step 2
2. Find the Euclidean distances between the distance and angle features correspond to image I and the template T .

3. If $\sum_{n=1}^N |I_n - F_n| * w_n > Th$, it is an imposter. Th is the threshold value assigned empirically, typically a value of 200, based on the sum of the weighted difference, I_n and F_n are the input image and Template metrics, N is the number of features, which is 16, w_n is the weight factor for each feature.
4. If $\sum_{n=1}^N |I_n - F_n| * w_n \leq Th$, compute the match score as per the equation followed to get a match score of 1 and 0 for a genuine and imposter respectively

$$M = \frac{Th - \sum_{n=1}^N |I_n - F_n| * w_n}{Th} \quad (4.23)$$

5. If $M \leq t$ fingerprint matches, where $0 \leq t \leq 1$

4.3.1.5 Implementation

4.3.1.5.1 Database used

In our work, fingerprints are acquired using an optical fingerprint scanner with 500dpi resolution and 600×600 image sizes. The scanner used is enBioscanF of NITGEN & Company which has a scanning window area of $1.2'' \times 1.2''$ and hence the fingerprint baseline is captured clearly. The image has a size of 91 KB in the JPEG format. 600 fingerprint image patterns of 100 candidates are sampled and stored. 3 images of a candidate are taken for forming the feature template. The average of 3 fingerprint feature vectors are taken and stored as the feature vector of a user. The rest of the 3 images were used to validation of the algorithm.

4.3.1.5.2 Implementation

The flowchart for feature extraction and matching are given in Appendix A. The algorithm is developed and tested under MATLAB environment. Since this algorithm is based on the global fingerprint features, fingerprint preprocessing can be avoided. Also the image quality achieved by the scanner is satisfactory and hence the computational overhead of enhancement is eliminated in this method. The directional field estimation is carried out as the first step in the feature extraction process. Gradient based directional field estimation is done with a block

size of $w=16$, which yield a good approximation of the fingerprint orientation structure. The result of the fingerprint orientation image overlaid on the original image is shown in fig. 4.17. In the next step, the coherence image is found out from the fingerprint image. The singularities are identified from the coherence image by searching the low coherence pixel values. Based on the relative positions of the coherence values, fingerprint core(s) and deltas are identified and the fingerprints are classified accordingly as per the classification schemes given in the section 4.3.1.2. Fingerprint with Left Loop singularity is classified as type 1, Right loop is classified as type 2, Whorl without any delta is classified as type 3 whorl with one delta is classified as type 4, Whorl with two deltas is classified as type 5, an Arch type is classified as type 6 and a Tented Arch type fingerprint is classified as type 7. Base line of the fingerprint is extracted using the pre-defined masks. From the singular point co-ordinates, fingerprint polygon is formed by dropping the perpendiculars from the delta/core point to the base line. The distance and angular features are found out from the polygon co-ordinates. The area of the polygon is determined using the area formula, knowing the co-ordinates of the polygonal vertices. The feature vectors of 3 images each corresponding to one fingerprint are found and the average of the corresponding features are taken and stored as the template feature vector of a single user.

The algorithm is validated using the other 3 fingerprint data and matching score computed as in section 4.3.1.4. The FAR and FRR of the system is measured and plotted.

4.3.1.6 Results and discussions

The algorithm is tested for the 300 test images of 100 genuine (matching) users as well as for the 300 images of impostors (non-matching). Classification of fingerprints was done as per the classification scheme for 600 fingerprint patterns. Table 4.3 gives the template data obtained for 5 users which show the variations in the features due to scanner conditions, fingerprint conditions, feature extraction errors etc. and hence the average values for each feature metric are obtained and are shown in the table. Table 4.4 shows the averaged template data of 24 candidates of four different classes. In the template data, core to base distance d_{cb} is the maximum among the distance metrics and has a maximum value of about 300

pixel units. The angular metrics have a maximum value of 179° , since only convex angles of the polygon are considered. The area of the fingerprint polygon ranges between values of 5000 and 45000 in sq.pixel units. The ridge count depends on the ridge frequency of the fingerprint image and the distance and hence varies between a low value of 4 to a high value of 30.

Table 4.5 shows the matching score obtained for the validation data. The table contains the match score obtained for each finger with the template data. The first three columns of the match score corresponds to the match score obtained for the fingerprint with id 1 with that of its own template data. Similarly, matching score column 4, 5 and 6 gives the match score of finger 2 with that of its own template. Fig.4.24 shows the box plot for genuine and imposter score distributions which clearly discriminates the genuine and imposter fingerprints. The match score for the genuine varies from 0.88 to 0.99 with a median value of 0.92 and that of the imposter varies from 0.65 to 0.45 with a median value of 0.56. As can be seen, there is a difference (gap) of 0.36 between the genuine and impostor, with no overlap between the two groups. The outliers of genuine distribution ranges between 1 and 0.79 and that of imposter distribution ranges between 0.75 and 0.1 with whiskers distributed towards 0.

Fig.4.25 shows False Accept Rate (FAR) and False Reject Rate (FRR) of the system is computed by taking the probability of the genuine and impostor score distributions with the matching threshold, t . The FAR and FRR plots are zero at $t=0.79$. At this threshold Equal Error Rate (EER) is zero and thus 100% recognition rate can be achieved. Hence by fixing threshold of $t=0.79$ the fingerprints were efficiently identified with 100% recognition rate and also the feature points are restricted to a 16 element polygonal features.

Table 4.4 Template Obtained and averaged over each feature values

Finger id	d_{av}	d_{sh}	d_{sp}	d_{sv}	d_{st}	d_{sn}	d_{se}	θ_r	θ_{DR}	θ_{DL}	θ_{CC}	A	T	r_{CD}	r_{CS}	r_{DS}
101_2_1	0	234	0	0	112	181	123.9	58.5	0	121.5	0	23240	2	10	17.5	15.5
101_2_2	0	237.4	0	0	112	188.3	122.5	66.2	0	113.8	0	23875	2	11	19	15.5
101_2_3	0	232	0	0	108	176	121.7	55.6	0	124.4	0	22032	2	11	19.5	15
Average	0	234.5	0	0	111	181.8	122.7	60.1	0	119.9	0	23049	2	10.7	18.7	15.3
103_2_1	0	213.7	0	0	86	143.4	110	48.5	0	131.4	0	15500	2	11	20	13
103_2_2	0	235	0	0	90	145	114.6	41.5	0	138.5	0	17955	2	11	23	14
103_2_3	0	210	0	0	90	146	110.4	44.4	0	135.6	0	16020	2	11	20	13
Average	0	219.6	0	0	88.7	144.8	111.7	44.8	0	135.2	0	16492	2	11	21	13.3
106_2_1	0	260.2	0	0	140	143.1	176.8	48.1	0	131.7	0	26510	2	17	17	10.5
106_2_2	0	256.4	0	0	143	145.3	181.5	52.4	0	127.4	0	28832	2	18	16.5	10.5
106_2_3	0	256.4	0	0	143	145.3	181.5	52.4	0	127.4	0	28832	2	18	16.5	10.5
Average	0	258.3	0	0	142	144.2	179.1	50.2	0	129.6	0	27671	2	17.5	16.8	10.5
106_5_1	0	250.8	183	121.4	129	0	0	45.1	134.7	0	0	24120	1	13	15.5	8.5
106_5_2	0	267.8	189.8	122.4	122	0	0	40.3	139.5	0	0	23923	1	14	15	9
106_5_3	0	249.4	173.1	116.6	111	0	0	40	140.1	0	0	20375	1	14	15	7.5
Average	0	256	182	120.1	121	0	0	41.8	138.1	0	0	22806	1	13.7	15.2	8.33
107_3_1	0	263.3	0	0	96.4	144.7	152.3	39.1	0	141	0	19630	2	14	20	14.5
107_3_2	0	248.9	0	0	118	141.8	159.4	47.9	0	132	0	23110	2	13	23.5	15
107_3_3	0	244.4	0	0	115	140.3	154.8	47.8	0	132.5	0	22077	2	13	21	14
Average	0	252.2	0	0	110	142.3	155.5	44.9	0	135.2	0	21606	2	13.3	21.5	14.5

Table 4.5 Fingerprint template data

Finger id	d_{oc}	d_{sb}	d_{abr}	d_{bs}	d_{bsr}	d_{bsl}	d_{bsr}	d_{bsl}	d_{sb}	θ_C	θ_{DR}	θ_{DL}	θ_{OC}	A	T	r_{CD}	r_{CS}	r_{DS}
1	0	234.5	0	0	111	181.8	122.7	60.1	0	119.9	0	119.9	0	23049	2	10.7	18.7	15.3
2	0	219.6	0	0	88.7	144.8	111.7	44.8	0	135.2	0	135.2	0	16492	2	11	21	13.3
3	0	258.3	0	0	142	144.2	179.1	50.2	0	129.6	0	129.6	0	27671	2	17.5	16.8	10.5
4	0	256	182	120.1	121	0	0	41.8	138.1	0	0	22806	1	22806	1	13.7	15.2	8.33
5	0	252.2	0	0	110	142.3	155.5	44.9	0	135.2	0	135.2	0	21606	2	13.3	21.5	14.5
6	0	237.9	0	0	12	187.3	52.3	13.3	0	166.7	0	166.7	0	2567	2	3	15.5	14.8
6	0	253.2	0	0	44	218.9	56.31	52.1	0	127.8	0	127.8	0	10451	2	7	20	16.5
7	0	311.5	0	0	152	131.5	235.4	19.7	0	160.3	0	160.3	0	33557	2	17.5	18	6.25
8	0	253.5	0	0	165	154.1	192	59	0	121.2	0	121.2	0	33558	2	16.7	15.7	10.5
9	87.1	197.7	0	0	206	158.4	184.7	140	0	125.5	94.14	44695	4	44695	4	18	17	9.5
10	0	247.4	187.2	134.5	149	0	0	52.9	126.9	0	0	28485	1	28485	1	15	22.3	7
11	0	275.3	0	0	99.4	138.5	169	25.5	0	154.5	0	154.5	0	20553	2	15.7	21.8	9.67
12	46.9	317.1	82.85	171.5	200	171	307.8	175	133.8	94.29	156.4	27746	3	27746	3	21	18.7	15.7
13	0	224.1	49.43	178	175	0	0	21.3	158.6	0	0	3540	1	3540	1	2	18	14.8
14	0	191.7	0	0	73	136	91.91	43.1	0	136.9	0	136.9	0	11959	2	8.33	15.8	11.7
15	0	219	124.6	129.1	85.4	0	0	43.6	136.2	0	0	14905	1	14905	1	9.5	12	16
16	0	214.6	160.8	105.1	117	0	0	40.5	139.4	0	0	18810	1	18810	1	16.7	18.3	6.83
17	0	169.4	42.64	140.2	30.8	0	0	46.6	133.3	0	0	4778	1	4778	1	3	15.3	13
18	0	258.3	0	0	136	173	160.6	46.8	0	133.2	0	133.2	0	29344	2	10.5	17.8	12.5
19	0	239.2	54.71	207.2	39.8	0	0	45.9	134.1	0	0	9096	1	9096	1	3.33	18.2	16
20	64	182.8	0	0	180	116	198.1	154	0	129	103.3	33052	4	33052	4	15	15	8
21	0	274	213	96.17	117	0	0	26.8	153.1	0	0	21686	1	21686	1	12.3	17.2	5.67
22	0	274.1	0	0	118	162.8	162.7	42.1	0	138	0	138	0	25818	2	15.7	25.2	14.7
23	0	227.7	0	0	74.2	112.4	137	32.8	0	147.3	0	147.3	0	12611	2	14	18.5	9
24	0	240.8	125.3	121.9	39.9	0	0	15.6	164.7	0	0	7170	1	7170	1	4.33	14.8	9.33

Table 4.6 Match Score Obtained

Development of a Biometric Personal Authentication System Based on Fingerprint and Speech

Finger id	Match Score	Matching id	Match Score	Matching id	Match Score	Matching id	Match Score	Matching id	Match Score	Matching id
1	0.8565	1	0.87	1	0.991	1	0	1	0	1
2	0	2	0	2	0	2	0.87	2	0.87	2
3	0	3	0	3	0	3	0	3	0	3
4	0	4	0	4	0	4	0	4	0	4
5	0	5	0	5	0	5	0	5	0	5
6	0	6	0	6	0	6	0	6	0	6
7	0	7	0	7	0	7	0	7	0	7
8	0.382	8	0.347	8	0.274	8	0	8	0	8
9	0	9	0	9	0	9	0	9	0	9
10	0	10	0	10	0	10	0	10	0	10
11	0	11	0	11	0	11	0	11	0	11
12	0.4517	12	0.377	12	0.379	12	0	12	0	12
13	0	13	0	13	0	13	0	13	0	13
14	0.2534	14	0.1	14	0.117	14	0	14	0	14
15	0.0319	15	0	15	0	15	0	15	0	15
16	0	16	0	16	0	16	0	16	0	16
17	0	17	0	17	0	17	0	17	0	17
18	0	18	0	18	0	18	0	18	0	18
19	0	19	0	19	0	19	0	19	0	19
20	0	20	0	20	0	20	0	20	0	20
21	0	21	0	21	0	21	0	21	0	21
22	0	22	0	22	0	22	0	22	0	22
23	0	23	0	23	0	23	0	23	0	23
24	0.3963	24	0.485	24	0.501	24	0	24	0	24
25	0	25	0	25	0	25	0	25	0	25
26	0	26	0	26	0	26	0	26	0	26
27	0	27	0	27	0	27	0	27	0	27
28	0.2908	28	0.279	28	0.373	28	0	28	0	28
29	0	29	0	29	0	29	0	29	0	29
30	0	30	0	30	0	30	0	30	0	30
31	0	31	0	31	0	31	0	31	0	31
32	0	33	0	33	0	33	0	33	0	33
33	0.2576	34	0.121	34	0.192	34	0	34	0	34
34	0.2457	35	0.25	35	0.16	35	0	35	0	35

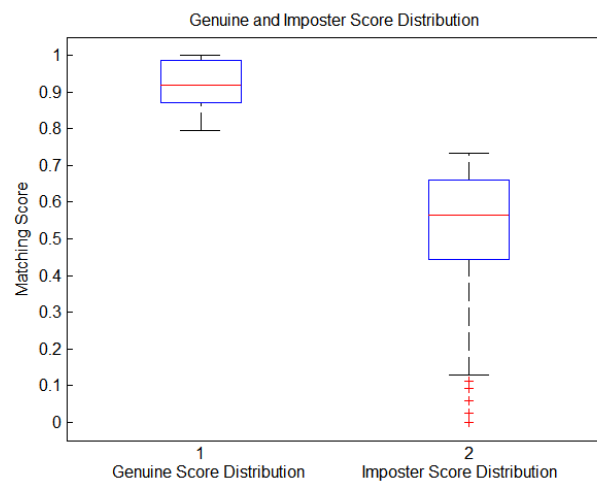


Fig. 4.24. Genuine and Imposter Score distribution

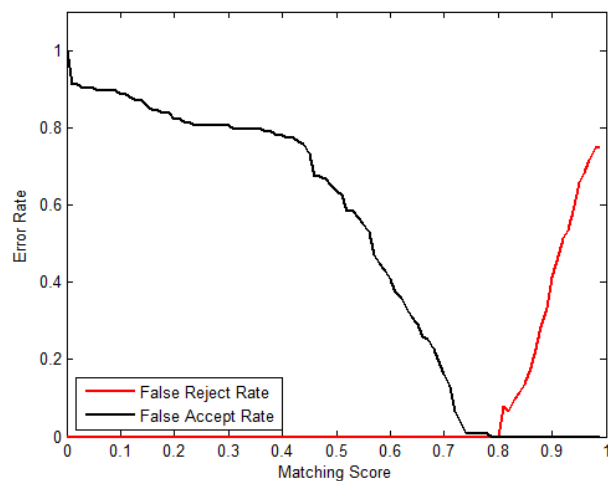


Fig. 4.25 FAR and FRR Characteristics

Table 4.6 gives a consolidation of the performance parameters of this novel method with that of the conventional minutia based method developed in the section 4.2; which shows the singularity based method outperforms the conventional minutia based method in terms number of features used for extraction as well as matching time. Since the conventional minutia based method requires

several processes like enhancement, segmentation, thinning etc., feature extraction stage requires about 3 times the computation time required for that of singularity based method. Also the feature matching requires less time only for singularity based method as the feature template size is just 16 element vector.

The method proposed here is recommended for access control applications.

Table 4.7 Comparison table of minutia based and singularity based fingerprint recognition system developed

Algorithm	Feature Vector Length	Feature Extraction Time Seconds	Feature Matching Time Seconds	FAR/FRR
Minutiae Based	50-100	592.670942	0.098625	FAR-0.75% FRR-0.875%
Singularity Based	16	182.549181	0.085623	0 at matching Threshold=0.8

4.4 Conclusion

This chapter presented a novel method for fingerprint identification based on the global singularity features. Singularities were identified using low value coherence computed using squared gradients. Fingerprint baseline forms the reference line for singularities and fingerprint polygon was formed. The fingerprint feature vector 16 element vector, derived from the singularity polygon the feature vector formed reduces the size of the template from conventional 70-100 minutiae points. Euclidean distance based simple matching was used and was tested with 100% accuracy. Conventional minutiae based fingerprint identification system was also developed and the performance of the two algorithms were compared with reference to feature extraction time, matching time and equal error rate.

Chapter 5

Speaker Recognition Techniques

Various signal processing aspects of speech are discussed in this chapter. Speech feature characteristics are explained and various spectral speech characteristics are also presented. Speaker recognition and speaker recognition techniques are explored with discussions on data clustering, Artificial Neural Networks, various types and their design issues are also presented in this chapter.

5.1 Introduction

Speech is the most fundamental method of communication between humans. The transformation of a speech signal to various domains is essential to enhance the communication ability. The speech pressure wave is primarily converted into electrical signals using a microphone transducer is the basic transduction involved in the digital signal processing of speech signal. The resulting analog signal is converted to digital domain using analog-to-digital (A/D) converter which can be processed using a digital computer. Discrete-time speech signal processing is defined as the modification of sampled speech signals using digital processor to obtain a new signal with desired properties. The modified speech signal is either stored in a digital computer or is converted back to analog signal which is amplified and played through a speaker.

At the physiological level, brain creates electric signals that move along the motor nerves which activate muscles in the vocal tract and vocal cords. This vocal tract and cord movement results in pressure changes within the vocal tract and at the lips initiating a sound wave that propagates in space. The sound wave creates a pressure change at the ear drum of a listener induces electric signals that move along the sensory nerves to the brain. At the linguistic level of the listener, brain performs speech recognition and understanding.

5.2 Speaker Recognition

Speaker recognition is the process of automatically recognising who is speaking by using specific information included in speech waves (Furui, 1997). Speaker recognition contains identification, verification, classification and segmentation, tracking and detection of speakers. It discovers the identity of a person based on the voice/speech recorded using a specific procedure. In speaker recognition system, the vocal tract characteristics of a person is modelled which is a mathematical model of the physiological system producing the human speech or simply a statistical model with similar output characteristics as the human vocal tract [Beigi, 1998; Flanagan, 1972; Miller, 1959]. New samples of speech may be

tested to determine the likelihood of them having been generated by the model of interest in contrast with other observed models. This is the underlying methodology for all speaker recognition applications [Beigi, 2011].

A speaker known to a speaker recognition system who is correctly claiming his/her identity is labelled a claimant and a speaker unknown to at the system who is posing as a known speaker is labelled an imposter [Quatieri, 2012]. A known speaker is also referred to as a target speaker while an imposter is called a background speaker.

5.2.1 Classification of speaker recognition

Speaker recognition can be classified into speaker identification and speaker verification [Furui, 1997] as in Fig. 5.1. Speaker verification task is to verify the claimed identity of person from his voice [Naik, 1990; Campbell, 1997]. The process involves a binary decision about the claimed identity. In speaker identification there is no identity claim and system decides the person identity about which the speaker is [Campbell, 1997]. The main difference between identification and verification is the number of decision alternatives. In speaker identification the number of decision alternatives is equal to the size of the population, whereas in verification there are only two cases: reject or accept, regardless of the population size. Therefore speaker identification performance decreases as the size of database increases [Furui, 1997]. In the case of speaker verification performance approaches a constant, independent of the size of population. Speaker identification methods can be classified into open set identification and closed set identification. In the closed set identification, the task of the system is to make a decision of who of the registered speakers is most likely the author of the unknown speech sample.

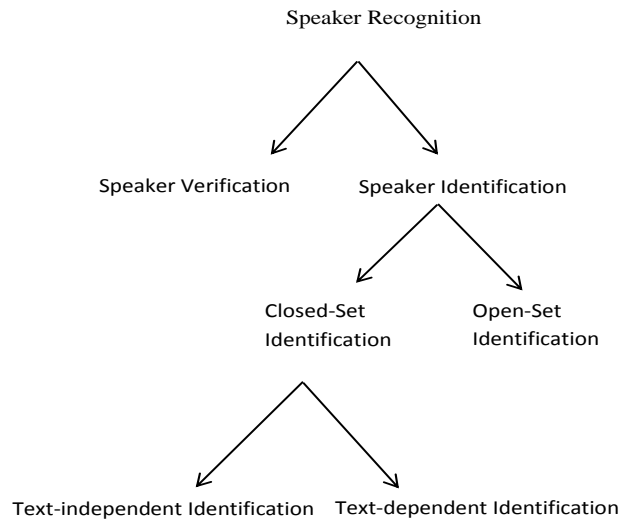


Fig 5.1 Speaker Identification Taxonomy

In the open set identification, a reference model for the unknown speaker may not exist and to make a conclusion that the speech sample is unknown.

Speaker recognition methods can also be divided into text-dependant and text independent methods. In text dependant method, the speaker requires to provide utterances that are the same text for training and recognition. In the text independent method, the speaker is not compelled to speak a specific text or sentence.

The most common method used in text dependant method are based on template matching techniques, where the time axes of an input speech sample and each reference template of the registered speakers are aligned, and the similarity between them is accumulated from the beginning to the end of the utterances [Naik, 1990; Furui, 1997; Campbell, 1997; Reynolds, 2002]. This method has got higher recognition rate than the text independent method as the method exploit voice individuality associated with each phoneme or syllable. Since both these

methods have serious weaknesses as the recorded voices of a registered speaker can be accepted as the registered speaker, a text prompted speaker recognition method has recently been proposed [Furui, 1997].

5.2.2 Basic Structure of Speaker Recognition System

In speaker recognition system, the task is to match a speech utterance from the unknown speaker with the speech models of the registered speaker. If the match is good enough, that is, above a threshold, the identity of speaker is established and authenticated. A high threshold makes it difficult for impostors to be accepted by the system as well as falsely rejecting rated users. Likewise, a low threshold enables a genuine or rated user to be accepted consistently, but the impostors are falsely accepted. The equal error rate (EER) is the common measure accepted as the over-all measure of system performance.

Speaker identification task can be divided into two phases. In the first phase, speech samples are collected from the speaker to derive a model. This is the enrolment or training phase. The collection of trained or enrolled models is called a speaker database. In the identification phase, a test sample from an unknown speaker is compared against the speaker database. In both phase, feature extraction is included, which is used to extract speaker dependant characteristics from speech. The features are extracted to reduce the amount of test data while retain speaker discriminative information, e.g. Linear Prediction Cepstral Coefficients (LPCC) or Mel-Frequency Cepstral Coefficients (MFCC) features.

The selection of the best parameter representation of acoustic data is an important task in the design of any speech recognition system [Melmerstein, 1980].

5.2.2.1 Feature selection

Speech is a behavioural characteristic which is a highly complex signal that carries several features mixed together [Rose P, 2002]. Hence feature extraction is necessary for representing speaker characteristics. Also to reduce the dimension of features and to overcome the curse of dimensionality, which implies that the number of needed training vectors increases exponentially with the dimensionality, feature extraction is inevitable in speaker recognition [Bishop,

1996; Jain, 1997; Jain, 2000;]. Furthermore, feature vectors decrease the storage saving and computational constraints. Wolf [Wolf, 1972] and Rose [Rose, 2002] listed several properties for an ideal feature for speaker recognition. They include:

- The feature should have large in-between speaker and small within-speaker variability.
- The features should have least influence on speakers' health or should be permanent.
- The features should be difficult to mimic.
- The features should occur frequently and naturally in speech.
- The features should be robust against noises and distortions.

Since speech consists of several features mixed together, several features can be extracted and combined to increase the robustness of feature set. For example: as formant frequencies are more robust, formant ratios along with short term spectral features can be thought of the fusion parameters to give robustness to feature vector.

5.2.2.2 *Speech Features*

According to Kinnunen [Kinnunen, 2005], a vast number of features have been proposed for speaker recognition, accordingly:

- Spectral features
- Dynamic features
- Source features
- Supra segmental features
- High level features

Spectral features involve short time speech spectrum description and have more or less the physical characteristics of the vocal tract. Hence spectral feature relates to the physical behaviour/characteristics of the vocal tract and hence has got its own importance due to simplicity.

Dynamic features relate to time evolution of spectral features. Source features are directly associated with glottal voice source. Supra segmental features

span over several segments and high level feature refer to symbolic type of information. Table 5.1 gives the consolidation of the features for speaker recognition [Kinnunen, 2005]. Since spectral features are the most common and accurate method available [Mermelstein, 1980], a discussion on MFCC based spectral features is followed.

Table 5.1 Features for speaker recognition

Feature Type	Examples
Spectral features	MFCC, LPCC, LSF, Long-term average spectrum (LTAS), Formant frequencies and bandwidths
Dynamic features	Delta features, Modulation frequencies, Vector autoregressive coefficients
Source features	F_0 mean, Glottal Pulse shape
Suprasegmental features	F_0 contours, Intensity contours, Micro prosody
High-level features	Idiosyncratic word usage, Pronunciation

5.2.3 MFCC based parametric representation

Mermelstein and Davis [Mermelstein, 1980] conducted experiments for the use of syllable-sized segments in the recognition of continuous speech. The aim of the test reported here was to select an acoustic representation of such segments. The methods used to evaluate the representation were both open and closed testing. In each case, same speaker produced both the reference and test data and also the same words in a variety of different syntactic contexts. In this study the speaker dependent representations is focussed to restrict the different sources of variations in the acoustic data.

White and Neely [White, 1976] derived twofold representations:

- 1) A 20 channel bandpass filtering approach using a Chebyshev norm on the logarithm of the filter energies as a closeness measure.
- 2) A linear prediction coding approach using a linear prediction residual proposed by Itakura (Itakura, 1975) as a similarity means.

White and Neely concluded that the above two methods were essentially equivalent when used with a dynamic programming time alignment method.

The use of filter banks with ear's frequency response was proposed by Feldtkeller and Zwicker [Feldtkeller, 1956; Schroeder, 1977]. The filters they proposed spaced linearly at low frequencies and logarithmically at high frequencies. Pals [Pals, 1977] showed that the first Eigen vectors of the covariance matrix for Dutch vowels of three speakers expressed in terms of 17 such filter energies, accounted for 91.8 % of total variance. Based on these studies, Davis and Mermelstein suggested a compact representation using a cosine transform of the real logarithm of the short-term energy spectrum on a mel-frequency scale. The block diagram of the structure of an MFCC processor is shown in Fig. 5.2. The primary goal of MFCC processor is to mimic the behaviour of human ears.

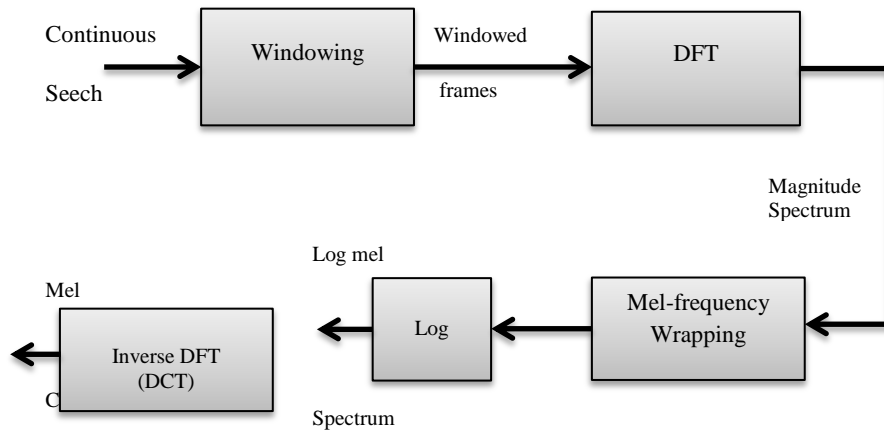


Fig. 5.2 MFCC computation block diagram

Cepstrum: The cepstrum is defined as the inverse DFT of the log magnitude of the DFT of a signal

$$c(n) = \mathcal{F}^{-1}\{\log|\mathcal{F}\{x[n]\}|\} \quad (5.1)$$

where \mathcal{F} is the DFT and \mathcal{F}^{-1} is the IDFT

Fig. 5.3 shows the block diagram of cepstrum computation

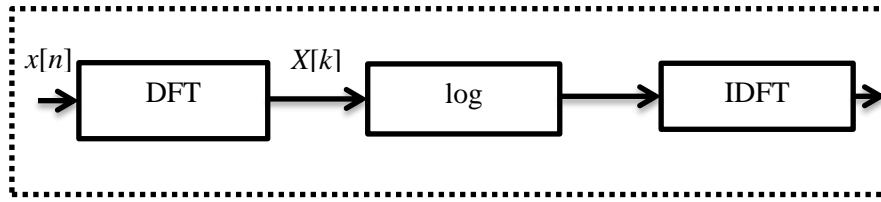


Fig. 5.3 Cepstrum computation

Framing: In the first step the speech signal is blocked into frames of N samples, with adjacent frames being separated by M samples ($N > M$) such that frames are overlapped by $N - M$ samples. Typical values for N is 256 which comes about 26 ms of period and $M = 100$.

Windowing: In order to minimise the signal discontinuities at the beginning and end of each frame and to minimise the spectral distortion, windows are used. If we define the windows as $w[n]$ $0 \leq n \leq N - 1$, where N is the number of samples in each frame, then the result of windowing is the signal

$$y[n] = x[n]w[n] \quad 0 \leq n \leq N - 1 \quad (5.2)$$

There most popular and commonly used window is Hamming window and is used in this work. Equation for a Hamming window is given by:

$$w(n) = 0.54 - 0.46 \cos\left(\frac{2\pi n}{N - 1}\right) \quad (5.3)$$

DFT: The next step in MFCC calculation is the FFT which is the windowed or Short-Time Fourier transform. Windowed FFT for N samples of the sequence $x[n]$ is calculated and speech spectrogram is obtained.

Mel Scale and Mel Frequency wrapping: Human perception of the frequency contents of sounds for speech signals does not follow a linear scale rather a logarithmic scale. This is particularly important to capture the phonetically important characteristics of speech. The scale *Mel* is given by

$$Mel(f) = 2595 \log_{10} \left(1 + \frac{f}{700} \right) \quad (5.3)$$

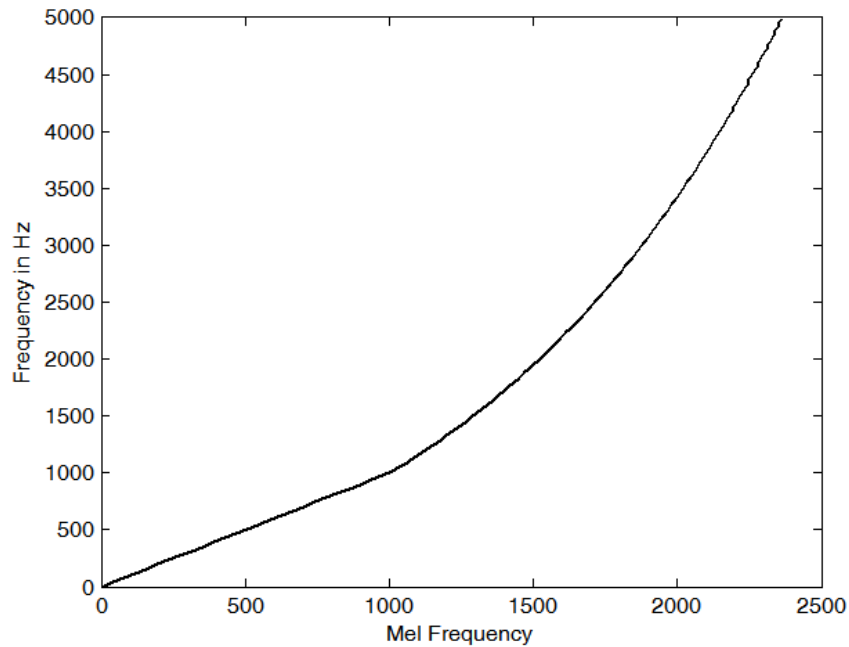


Fig 5.4 Linear Vs mel-frequency

The mel-frequency scale has linear frequency spacing below 1000 Hz and a logarithmic spacing above 1000 Hz. The relation between linear and mel-frequency is shown in Fig. 5.4. In order to simulate the subjective spectrum, a filter bank is used in which the frequency band is uniformly spaced on the mel- scale. The filter bank has a triangular bandpass frequency response and the spacing as well as the bandwidth is determined by a constant mel frequency interval [Mermelstein, 1980]. The number of mel filters is typically chosen as 20 over a frequency range of about 5000 Hz (Fig. 5.5). This mel wrapping filter bank can be viewed as a histogram bin in the frequency domain.

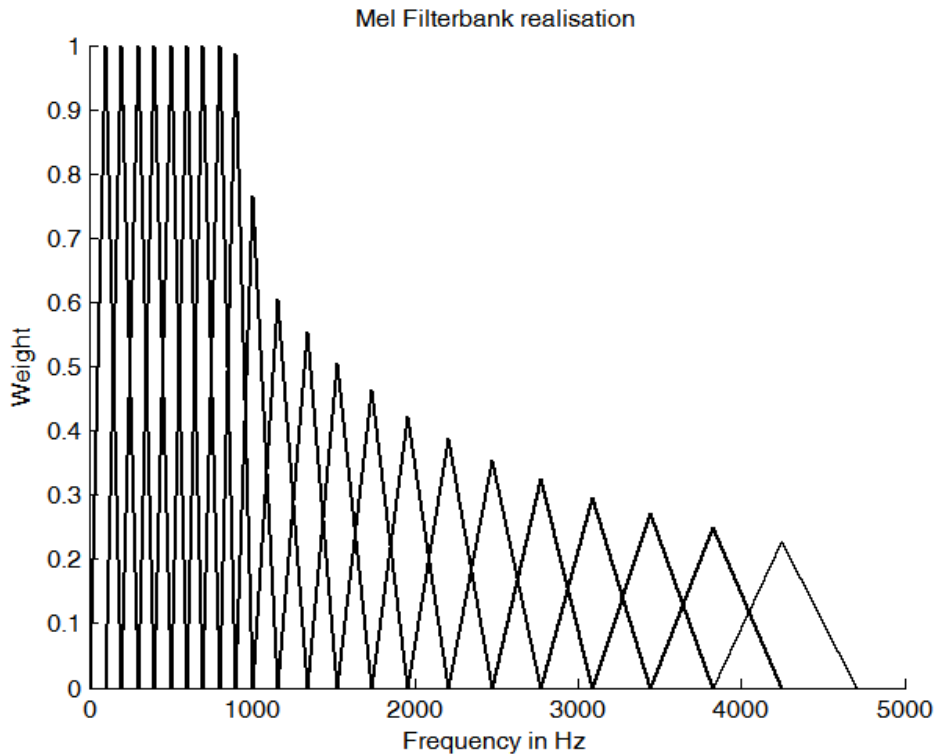


Fig. 5.5 Triangular weighting functions

The mel-frequency spectrum at analysis time \hat{n} is defined for $r=1, 2, 3 \dots R$ as

$$MF_{\hat{n}} = \frac{1}{A_r} \sum_{K=L_r}^{U_r} |V_r[K] X_{\hat{n}}[K]|^2 \quad (5.3)$$

where

$$A_r = \sum_{L_r}^{U_r} |V_r[K]|^2$$

is the normalising factor for the r^{th} mel-filter.

MFCC computation: For each frame, a discrete cosine transform of the log of the magnitude of the filter outputs is computed to form the function $mfcc_{\hat{n}}[m]$ i.e.,

$$mfcc_{\hat{n}}[m] = \frac{1}{R} \sum_{r=1}^R \log (MF_{\hat{n}}[r]) \cos \left[\frac{2\pi}{R} \left(r + \frac{1}{2} \right) m \right] \quad (5.4)$$

Typically, $mfcc_{\hat{n}}[m]$ is evaluated for a number of coefficients, N_{mfcc} , that is less than the number of mel-filter. For example, $N_{mfcc} = 13$ and $R=22$. By applying the method described above for each speech frame of around 30 msec with overlap a set of mel-frequency cepstrum coefficients is computed. This is the result of a DCT of the logarithm of the short-term power spectrum expressed on a mel-frequency scale. This set of coefficient is called the *acoustic vector* or *speech code vector*.

5.3 Speaker Recognition Algorithms

There are three most commonly used approaches to speaker recognition methods. They are:

1. Minimum –Distance Classifier
2. Vector Quatization (VQ)
3. Gaussian Mixture Model (GMM)

5.3.1 Minimum-Distance Classifier

This is one of the simplest approaches to speaker recognition where the average of feature vectors over multiple analysis frames for speakers from testing and training data are compared and then find the distance between these average test and training vectors [Campbell, 1997]. In speaker verification, the target speaker with the smallest distance from that of the test speaker is selected as the identified speaker.

Consider the average of mel-cepstral features for the test and training data:

$$\bar{C}_{mel}^{ts}[n] = \frac{1}{M} \sum_{m=1}^M C_{mel}^{ts}[mL, n] \quad (5.5)$$

$$\bar{C}_{mel}^{tr}[n] = \frac{1}{M} \sum_{m=1}^M C_{mel}^{tr}[mL, n] \quad (5.6)$$

where the subscripts *ts* and *tr* denote test and training data, respectively, *M* is the number of analysis frames which differs in training and testing, and *L* is the frame length. The mean-squared difference between the average testing and training feature vectors expressed as

$$D = \frac{1}{R-1} \sum_{n=1}^{R-1} (\bar{C}_{mel}^{ts}[n] - \bar{C}_{mel}^{tr}[n])^2 \quad (5.7)$$

where *R* is the number of mel-cepstral coefficients. If *D* exceeds some threshold, i.e., if $D > T$, the speaker present. T

5.3.2 Vector Quantization (VQ)

The minimum-distance classifier does not distinguish between acoustic speech classes. It uses average of feature vectors per speaker computed over all sound classes. If the average feature vectors over distinct sound classes, e.g., quasi-periodic, noise-like, and impulse-like sounds, are even finer phonetic

categorization within these sound classes, are taken and computed a distance with respect to each sound class, and then average the distances over classes is taken. This would reduce, for example, the phonetic differences in the feature vectors and help focus on speaker differences.

Suppose the speech segments corresponding to the sound classes in both the training and test data are known then the averages for the i^{th} class can be computed as

$$\bar{C}_i^{ts}[n] = \frac{1}{M} \sum_{m=1}^M C_i^{ts}[mL, n] \quad (5.8)$$

$$\bar{C}_i^{tr}[n] = \frac{1}{M} \sum_{m=1}^M C_i^{tr}[mL, n] \quad (5.9)$$

The Euclidean distance with respect to each class can be computed as

$$D(i) = \frac{1}{R-1} \sum_{n=1}^{R-1} (\bar{C}_i^{ts}[n] - \bar{C}_i^{tr}[n])^2 \quad (5.10)$$

Finally, the average over all classes is

$$D(I) = \frac{1}{I} \sum_{i=1}^I D_i \quad (5.11)$$

where I is the number of classes. To form a distance measure, the class distinctions for the training and test data have to be identified.

One approach to segmenting a speech signal in terms of sound classes is through speech recognition on a phoneme or word level. Hidden Markov Model(HMM) is one of the segmentation technique for yielding the speaker classes in terms of desired acoustic phonetic or word classes and can achieve good speaker recognition performance [Deller, 1993; Reynolds, 1995; Jelinek, 1998]. An

alternative approach is to obtain acoustic classes without segmentation of an utterance sound class to specific phoneme or word categories; there is no identifying or labelling of acoustic classes. Vector quantization (VQ) method is one of the techniques used to achieve sound categorisation using k -nearest neighbour clustering algorithm. Each centroid in the clustering is derived from training data and represents an acoustic class, but without identification or labelling. The distance measure used in the clustering is given by the Euclidean distance between a feature vector and a centroid. In testing phase, a class is picked for each feature vector by finding the minimum distance with respect to the various centroids from the training stage and the average of the minimum distances over all test feature vectors. Fig. 5.6 shows pictorial representation of a) training and b) testing in speaker recognition by VQ. In Testing the average minimum distance of test vectors to centroids is lower for speaker A than for speaker B

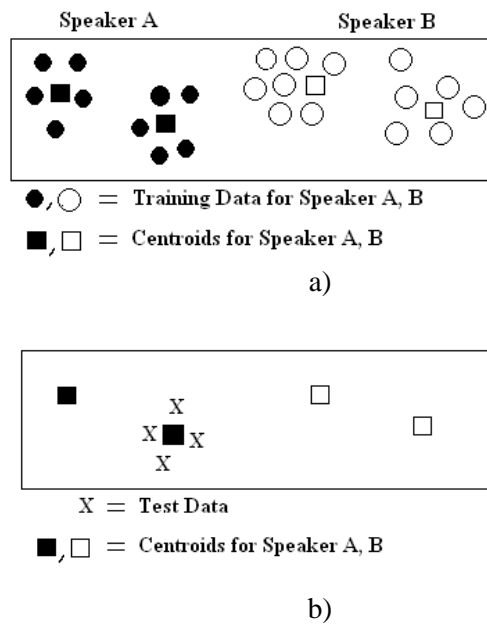


Fig. 5.6 VQ based pictorial representation of a) training and b) testing in speaker recognition by VQ

5.3.3 Gaussian Mixture Model (GMM)

Due to context, coarticulation, and anatomical and fluid dynamical variations, a particular sound is never produced by a speaker with exactly the same vocal tract shape and glottal flow [Quatterri, 2012]. Thus speech production is not deterministic. A probabilistic model through a multi-dimensional Gaussian probability distribution function (PDF) is an efficient way of representing this non-deterministic speech profile [Reynolds, 1995]. The Gaussian PDF of a feature vector \underline{x} for the i^{th} state is written as

$$b_i(\underline{x}) = \frac{1}{(2\pi)^{\frac{R}{2}} |\Sigma_i|^{\frac{1}{2}}} e^{-\frac{1}{2}(\underline{x}-\underline{\mu}_i)^T \Sigma_i^{-1}(\underline{x}-\underline{\mu}_i)} \quad (5.14)$$

where $\underline{\mu}_i$ is the state mean vector, Σ_i is the state covariance matrix, and R is the dimension of the feature vector. The vector $(\underline{x} - \underline{\mu}_i)^T$ denotes the matrix transpose of $(\underline{x} - \underline{\mu}_i)$ and Σ_i^{-1} indicate the determinant and inverse, respectively of matrix Σ_i . the mean vector $\underline{\mu}_i$ is the expected value of elements of the feature vector, \underline{x} , while the covariance matrix Σ_i represents the cross-correlations and the variance of the elements of the feature vector.

The probability of a feature vector being in any one of I stat, is (or acoustic classes), for a particular speaker mode, denoted by λ , is represented by the union or mixture of different Gaussian PDFs:

$$p(\underline{x}|\lambda) = \sum_{i=1}^I p_i b_i(\underline{x}) \quad (5.15)$$

where the $b_i(\underline{x})$ are the component mixture densities and p_i are the mixture weights as given in fig. 5.7. Given that each individual Gaussian PDF integrates to unity, then $\sum_{i=1}^I p_i = 1$ is taken to ensure that the mixture density represents a true PDF. The speaker model λ then represents the set of GMM mean, covariance, and weight parameters, i.e.,

$$\lambda = \{p_i, \underline{\mu}_i, \Sigma_i\} \quad (5.16)$$

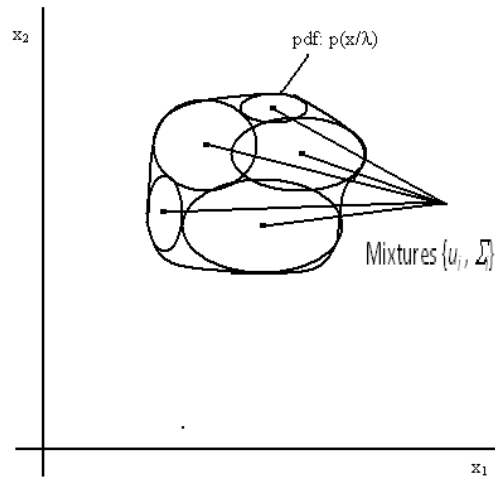


Fig. 5.7 Gaussian mixture model is a union of Gaussian pdfs assigned to each acoustic state.

Now a probabilistic model for the feature vectors is available. Then the parameters for each speaker models are estimated and classified the utterances. Clusters can be formed with the training data to represent each cluster with a Gaussian pdf, the union of Gaussian pdfs being the GMM. One approach to estimate the GMM model parameters is by maximum-likelihood estimation [Quatieri, 2010]. In the identification process, the probabilities of each speaker model with the given features are computed. i.e., $P(\lambda_j|x_{ji})$ and then choose the speaker with highest probability. This technique is known as Maximum *a posteriori* (MAP) probability classification.

5.4 Data Clustering

Data classification systems are either categorised as supervised or unsupervised based on whether they assign new inputs to one of a finite number of

discrete supervised classes or unsupervised categories [Bishop, 1996; Cherkassky, 1998; Duda, 2010]. In a supervised classification, the mapping from a set of input data vectors ($x \in \mathfrak{R}^d$; where d is the input space dimensionality), to a finite set of discrete class labels ($y \in 1, 2 \dots C$, where C is the total number of class types), is modelled in terms of some mathematical function $y = y(X, W)$, where W is a vector of adjustable parameters [Rui, 2005]. The values of these parameters are determined by an inductive algorithm [Bishop, 1996]. Unsupervised classification is called as clustering, or explanatory data analysis, no labelled data are available [Everitt, 2001; Jain, 1988]. In clustering, the aim is to separate a finite unlabelled data set into a finite and discrete set of natural hidden data structures, rather than provide an accurate characteristics of unobserved samples generated from the same probability distribution. Clustering partitions data into a certain number of clusters i.e., into groups, subsets or categories.

An illustration of cluster analysis is given in Fig. 5.8 [Rui, 2005]. The four basic steps are briefly explained below:

1. *Feature selection or extraction*: Feature selection chooses distinguishing features from a set of candidates and feature extraction utilises some transformations to generate useful and novel features from the original ones [Rui, 2005, Jain, 1988, Bishop, 1996]. Selection of features can decrease the workload and simplify the subsequent design process. An ideal feature should meet the criterion as discussed in section 5.2.2
2. *Clustering algorithm design or selection*: This step is combined with the selection of a corresponding proximity measure and the construction of a criterion function. Patterns are grouped according to whether they resemble each other. Almost all the clustering algorithms are explicitly or implicitly connected to some definition of proximity measure.

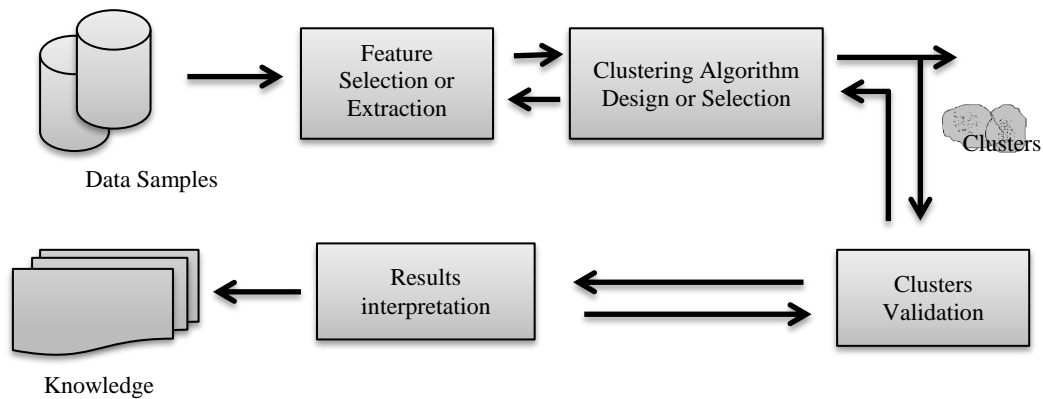


Fig. 5.8 Steps in cluster analysis

3. *Cluster validation:* Given a data set, each clustering algorithm can always generate a division, no matter whether the structure exists or not. Moreover, different approaches usually lead to different clusters; and even for the same algorithm, parameter identification or the presentation order of input patterns may affect the final results. Therefore, effective evaluation standards and criteria are important to provide the users with a degree of confidence for the clustering results derived from the used algorithms. These assessments should be objective and have no preferences to any algorithm. Also, they should be useful for answering questions like how many clusters are hidden in the data, whether the clusters obtained are meaningful or just an artifact of the algorithms, or why we choose some algorithm instead of another.
4. *Results interpretation:* The final goal of clustering is to provide users with meaningful insights from the original data, so that they can effectively solve the problems encountered. Experts in the relevant fields interpret the data partition. Further analyses, even experiments, may be required to guarantee the reliability of extracted knowledge.

5.5 Artificial Neural Network as a classifier

Artificial Neural Networks (ANNs) may be defined as structures comprising of densely interconnected adaptive simple processing elements that are capable of performing massively parallel computations for data processing and knowledge representations (Hecht, 1990; Schalkoff, 1997). ANNs have the remarkable information processing characteristics of biological systems such as non-linearity, high parallelism, robustness, fault and failure tolerance, learning, ability to handle imprecise and fuzzy information and their capability to generalise [Jain, 1996]. Artificial models possessing such characteristics are desirable because: (i) nonlinearity allows better fit to the data, (ii) noise-insensitivity provides accurate prediction in the presence of uncertain data and measurement errors, (iii) high parallelism implies fast processing and hardware failure-tolerance, (iv) learning and adaptively allow the system to update (modify) its internal structure in response to changing environment, and (v) generalization enables application of the model to unlearned data.

Biological Neural Network: A biological neuron is the basic building block of the nervous system (Fig. 5.9a). A human nervous system consists of billions of neurons of various types and lengths relevant to their location in the body (Schalkoff, 1997). A biological neuron consists of three major units: dendrites, cell body and axon. The cell body consists of a nucleus which holds the heredity information and plasma that holds the molecular equipment used for producing the maternal needed by the neuron [Jain, 1996]. The role of dendrites is to receive signals from other neurons and pass them to the cell body. Axon branches into collaterals collects signals from the cell body and carries away through the synapse to the dendrites of neighbouring neurons. An electric impulse signal travels within the dendrites and through the cell body towards the pre-synaptic membrane of the synapse. On arriving the impulse at the membrane, a chemical neurotransmitter is released from the vesicle [Fig. 5.9b]. The concentration of the neurotransmitters transmitted is proportional to the strength of the incoming signal. The neurotransmitter diffuses within the synaptic gap between the post-synaptic membrane and then into the dendrites of neighbouring neurons. New electrical signal is generated at the next neuron depending on the threshold of the receiving

neuron and is passed through the neuron as in the previous neuron. The amount of signal that passes through a receiving neuron is proportional to the intensity of the signal receiving from each neuron, strength and the threshold of the receiving neuron.

As the neuron has a large number of dendrites and synapses, it receives and transfer parallel many signals. These signals may either excite or inhibit the firing of the neuron. This is the basic operation of the building unit of ANNs (MCCulloch, 1943). Fig. 5.10 gives an analogy between artificial neuron and biological neuron. In the figure, the nodes represent the synapses and the connection weights represent the synapses and the threshold approximates the activity in the soma [Jain, 1996]. Here n biological neurons with various signals of intensity x and synaptic strength w feeding into a neuron with a threshold of b and the equivalent artificial neuron system is given. Both the biological and ANN learn by incrementally adjusting the magnitude of the synaptic strengths [Zupan, 1993]. It was Rosenblatt who introduced the working of a single artificial neuron and named '**Perceptron**' for solving the issues in character recognition [Hecht, 1990]. As shown in Fig. 5.10, an artificial neuron receives inputs as stimuli from the surroundings, combine them in a special way to form an effective or net input (ξ). This is passed over through a linear threshold gate, and transmitted the output, y signal forward to another neuron or to the environment. The neuron fires or activates only when ξ exceeds the neurons threshold limit which is also called the bias, b (Haykin, 1994). For n signals, the perceptron neuron operation is expressed as

$$y = \begin{cases} 1, & \text{if } \sum_{i=1}^n w_i x_i \geq b, \\ 0, & \text{if } \sum_{i=1}^n w_i x_i < b \end{cases} \quad (5.17)$$

A positive weight ($w_i > 0$) enhance the net signal, ξ and excite the neuron, whereas a negative weight reduce ξ and inhibit the neuron activity and link is called inhibitory.

The perceptron can be trained on a set of example using a special learning rule [Basheer, 2000]. The perceptron weights are changed in proportion to the

difference(error) between the target (Correct) output, Y , and the perceptron solution, y , Rosenblatt proposed (Rosenblatt, 1962) the perceptron rule that will yield an optimal weight vector in a finite number of iterations, regardless of the initial values of the weights. In this case, the linearly separable classes can be accurately classified in which a linear hyper plane can place one class of objects on one side of the plane and the other class on the other side (Hecht, 1990). Multilayer perceptron architecture was proposed in which additional layers of neurons placed between the input layer and the output neuron.

This is to handle the non-linearly separable problem. The layers introduced in between the input and output layer are called hidden layers and their nodes called hidden nodes. Rumelhart et al, (Rumelhart, 1986) rediscovered the backpropagation network, which is one of the multilayer perceptron trained by delta learning rule. Learning procedure is an extension of the simple perceptron algorithm so as to handle the weights connected to hidden nodes [Hecht, 1990].

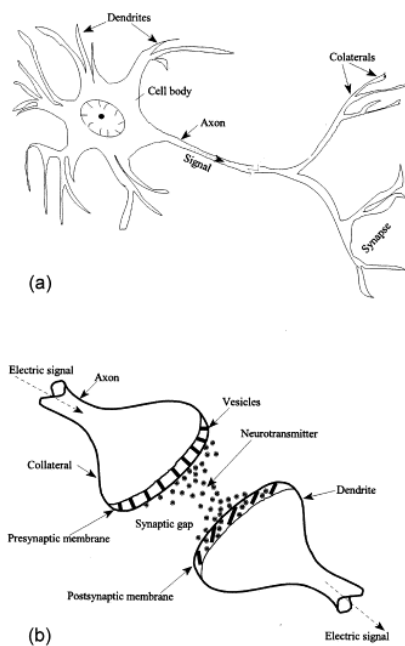


Fig. 5.9 a) Biological neuron b) Signal transfer between two biological neurons [Basheer, 2000]

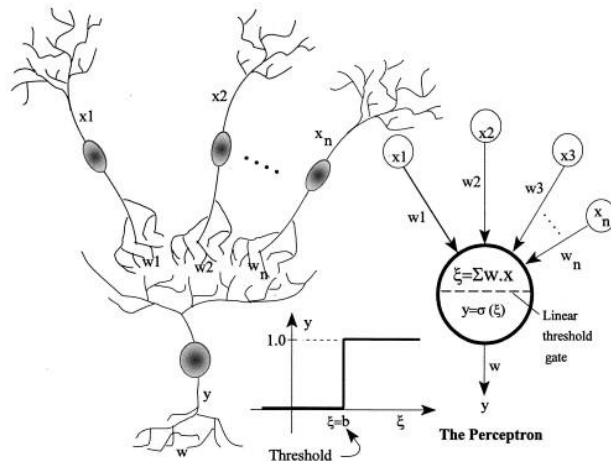


Fig. 5.10 Biological neuron Vs Artificial single layer perceptron model

Learning in ANN and Learning rules

An intelligent system is one which has the ability to learn. In the case of artificial systems, learning is viewed as the process of updating the internal representation of the system in response to external stimuli so that it can perform a specific task [Basheer, 2000]. In the ANN scenario, learning involves adjusting the weights of the links, pruning or creating some connection links, and/or changing the firing rules of the individual neurons. Learning in ANN is performed with training examples using an iterative process. A learnt ANN can handle imprecise, fuzzy, noisy, and probabilistic information without noticeable adverse effect on response quality, and can generalise from the tasks it has learned to unknown ones.

Learning rules propose how the network weights should be adjusted between successive training cycles (epochs). Basically there are three types of rules:

- Error-correction learning(ECL) [Haykin, 1994] rule is used in supervised learning in which the arithmetic difference between the ANN solution at any stage during training and the corresponding correct answer is used to modify the connection weights so as the reduce the overall network error.

- Boltzmann learning (BL) [Jain, 1996] rule is similar to ECL, however each neuron generates an output based on a Boltzmann statistical distribution.
- Hebbian learning (HL) rule [Hebb, 1949] is the oldest learning rule, developed based on neurobiological experiments, which postulates that “if neurons on both sides of a synapse are activated synchronously and repeatedly, the synapse’s strength is selectively increased.”

In the competitive learning (CL) rule [Haykin, 1994], all neurons are forced to compete among themselves such that only one neuron will be activated in a given iteration with all the weights attached to it adjusted.

Popular ANNs

ANN has emerged as one the most accurate and robust patter classifier and a tool for solving non-linear problems. Hence new or modification of existing ones, are being constantly developed. Pham [Pham, 1994] estimated that over 50 different ANN types exist.

- Hopfield networks*: This is a fully connected two-layer recurrent network that acts as a nonlinear associative memory and especially efficient in solving optimization problems [Hopfield, 1984].
- Adaptive resonance theory (ART) networks*: These networks are trained by unsupervised learning where the network adapts to the information environment without intervention. This is a fully interconnected network [van Rooij, 1996].
- Kohonen networks*: These networks are two-layer networks that transform n-dimensional input patterns into lower-ordered data where similar patterns project onto points in close proximity to one another [Kohonen, 1989]. These networks are also called self-organising feature maps.
- Backpropagation networks*: This is the most popular ANN and is considered as the workhorse of ANNs [Rumelhart, 1986]. A backpropagation (BP) is an MLP consisting of an input layer which accepts the input variables, an output layer with nodes representing the dependent variable, hidden layer(s) containing the

nodes to help to solve the nonlinearity in the data. This network uses supervised learning with ECL rule. The error computed at the output side is propagated backward from the output layer to hidden layer, and finally to the input layer. The neurons in this network can be fully or partially interconnected.

- e. *Recurrent networks:* In a recurrent network, the outputs of some neurons are fed back to the same neurons or to neurons in preceding layers. Thus the information flow is possible in both forward and backward directions and hence the ANN is associated with a dynamic memory [Pham, 1994]
- f. *Counterpropagation networks:* Developed by Hecht-Nielsen [Hecht, 1988, 1990], this network are trained by hybrid learning to create a self-organizing look-up table useful for function approximation and classification. Unsupervised learning is carried out to create a Kohonen map of the input data.
- g. *Radial basis function (RBF) networks:* This is a multilayer feedforward error-backpropagation network with three layers. this network uses a variety of learning algorithms including a two-step hybrid learning [Haykin, 1994]

In the present work, BPANN is used to train the clustered speech patterns of speakers. A description BPANN and the BP Algorithm is given below.

5.5.1 Backpropagation ANNs

BPANNs are popular due to their flexibility and adaptability in modelling a wide spectrum of problems in many application areas [Basheer, 2000]. The feedforward error-backpropagation is the state-of-art method for training ANNs. BP is based on searching an error surface using gradient descent for points with minimum error. For each iteration, during the forward activation produce a solution, and a backward propagation of the computed error to modify the weights. Fig. 5.11a shows a fully connected, layered, feedforward network. In the figure, weights on connections between the input and hidden layers are denoted by w_i , while weights on connections between the hidden and output layers are denoted by w_h . This network has three layers, although it is possible and sometimes useful to

have more. Each unit in one layer is connected in the forward direction to every unit in the next layer. Activations flow from the input layer through the hidden layer and then on to the output layer. The knowledge of the network is encoded in the weights on connection between units. The existence of hidden units allows the network to develop complex feature detectors, or internal representations. A backpropagation unit will sums up its weighted inputs produces a real value between 0 and 1 as output based on a sigmoid function, (Fig. 5.11b) which is continuous and differentiable, as required by the backpropagation algorithm. A backpropagation network typically starts out with a random set of weights (Rich and Knight, 1994).

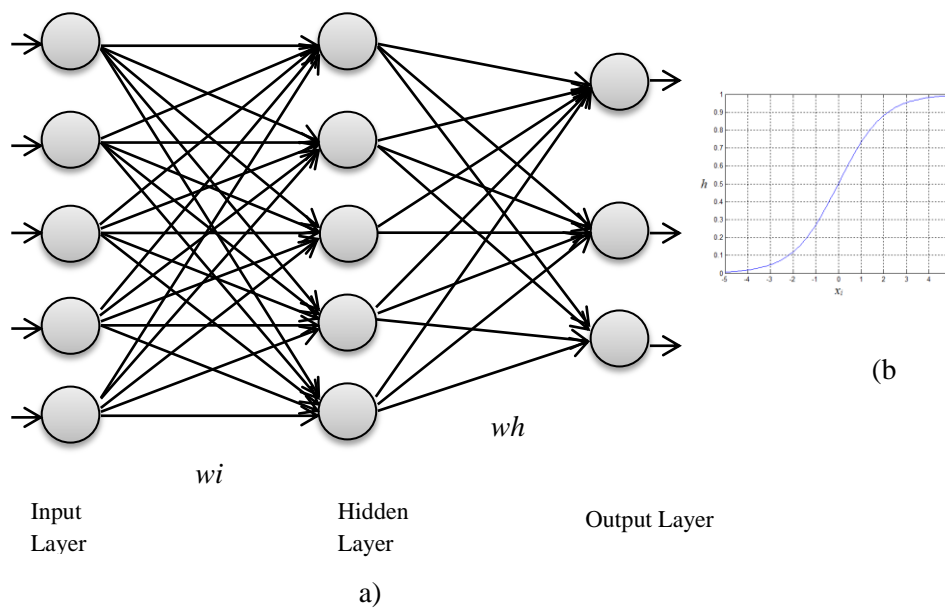


Fig. 5.11 a) A MLP Backpropagation network b) Sigmoid activation function

Given: A set of input-output (x:y) vector pairs.

Compute: A set of weights for a three layer network that maps inputs onto corresponding outputs. (w_i is the weight of the input layer to the hidden layer, w_h is the weight of the hidden layer to the output layer)

Step 1 Let A be the number of units in the input layer, B be the number of units in the hidden layer, C be the number of units in the output layer.

Step 2 $w_{ii,j} = \text{random}(-0.1, 0.1)$ for all $i = 0 \dots \dots \dots A, j = 1 \dots \dots \dots B$

$wh_{i,j} = \text{random}(-0.1, 0.1)$ for all $i = 0 \dots \dots \dots B, j = 1 \dots \dots \dots C$

Step 3 Initialize the activation of the threshold units. The values of the threshold units should never change. Set the learning rate, η .

Step 4 Choose an input-output pair. Assign activation levels to the input units.

Step 5 Propagate the activations from the units in the input layer to the units in the hidden layer using the activation function

$$h_j = \frac{1}{1 + e^{-\sum_{i=0}^A w_{i,j}x_i}} \text{ for all } j = 1 \dots \dots B \quad (5.18)$$

Step 6 Propagate the activations from the units in the hidden layer to the units in the output layer

$$o_j = \frac{1}{1 + e^{-\sum_{i=0}^B wh_{i,j}h_i}} \text{ for all } j = 1 \dots \dots C \quad (5.19)$$

Step 7 Compute the errors of the units in the output layer, δ_{2j}

$$\delta_{2j} = o_j(1 - o_j)(y_j - o_j) \text{ for all } j = 1 \dots \dots C \quad (5.20)$$

Step 8 Compute the errors of the units in the hidden layer, δ_{1j}

$$\delta_{1j} = h_j(1 - h_j) \sum_{i=1}^C \delta_{2j} wh_{j,i} \text{ for all } j = 1 \dots \dots \dots B \quad (5.21)$$

Step 9 Adjust the weights w_i and wh

$$\Delta wh_{i,j} = \eta \delta_{2j} h_i \quad (5.22)$$

for all $i = 0 \dots \dots \dots B, j = 1 \dots \dots \dots C$

$$\Delta w_{i,j} = \eta \delta_j x_i \quad (5.23)$$

for all $i = 0 \dots \dots A, j = 1 \dots \dots B$

Step 10 Go to step 4 and repeat. When all the input-output pairs have been presented to the network, one epoch has been completed. Repeat steps 4 to 10 for as many epochs as desired.

The algorithm generalizes straightforwardly to networks of more than three layers. For each extra hidden layer, insert a forward propagation step between steps 6 and 7, an error computation step between steps 8 and 9, and a weight adjustment step between steps 10 and 11. Error computation for hidden units should use the equation in step 8, but with i ranging over the units in the next layer, not necessarily the output layer (Rich and Knight, 1994).

5.5.2 Some general issues in ANN development

A number of issues should be addressed before initiation of any network training. Some of the following issues are only relevant to BP ANNs while others are applicable to the design of all ANN types [Basheer, 2000].

a. Database size and partitioning

The main issue in ANN development project is to specify the size of database. Models developed from data generally depend on database size. ANNs, like other empirical models, may be obtained from databases of any size; however generalization of these models to data from outside the model development domain will be adversely affected. Since ANNs are required to generalize for unseen cases, they must be used as interpolators. Data to be used for training should be sufficiently large to cover the possible known variation in the problem domain. The development of an ANN requires partitioning of the parent database into three subsets: *training, test, and validation*. The training subset should include all the data belonging to the problem domain and is used in the training phase to update the weights of the network. The test subset is used during the learning process to check the network response for untrained data. The data used in the test subset should be distinct from those used in the training; however they should lie within

the training data boundaries. Based on the performance of the ANN on the test subset, the architecture may be changed and/or more training cycles be applied.

The third portion of the data is the validation subset which should include examples different from those in the other two subsets. This subset is used after selecting the best network to further examine the network or confirm its accuracy before being implemented in the neural system and/or delivered to the end user. The size of the database is not governed by any of the mathematical rules. Usually experienced base rules of thumb and analogy between ANNs and statistical regression exist for limiting the size of various data subsets. According to Baum and Haussler [Baum, 1989], the minimum size of training subset to be equal to the number of weights in the network times the inverse of the minimum target error. Dowla and Rogers [Dowla, 1995] and Haykin [Haykin, 1994] suggest an example-to-weight ratio (EWR) > 10 , while Masters [Masters, 1994] suggests $EWR > 4$. Looney [Looney, 1996] recommends 65% of the parent database to be used for training, 25% for testing, and 10% for validation, whereas Swingler [Swingler, 1996] proposes 20% for testing and Nelson and Illingworth [Nelson, 1990] suggest 20–30%.

b. Data pre-processing, balancing and enrichment

In order to accelerate convergence during training, many pre-processing techniques are usually applied before the data can be used for training. Among these are noise removal, reducing input dimensionality, and data transformation [Basheer, 2000], treatment of non-normally distributed data, data inspection, and deletion of outliers. The training data has to be distributed nearly evenly between the various classes to prevent the network from being biased to the over-represented classes [Swingler, 1996]. Adding random noise to their input data, keeping output class unchanged, duplicating the under-represented input/output examples are some of the methods to balance the data. Small database size poses another problem in ANN development because of the inability to partition the database into fairly-sized subsets for training, test, and validation. To expand the size of the database, the trivial way is to get new data (if possible) or introduce random noise in the available examples to generate new ones. Noise addition normally enhances the ANN robustness against measurement error

c. Data normalization

Normalization (scaling) of data within a uniform range (e.g. 0–1) is essential (i) to prevent larger numbers from overriding smaller ones, and (ii) to prevent premature saturation of hidden nodes, which impedes the learning process. This is especially true when actual input data take large values. There is no unique standard procedure for normalizing inputs and outputs. One way is to scale input and output variables (z_i) in interval $[\lambda_1, \lambda_2]$ corresponding to the range of the transfer function:

$$x_i = \lambda_1 + (\lambda_2 - \lambda_1) \left(\frac{z_i - z_i^{min}}{z_i^{max} - z_i^{min}} \right) \quad (5.24)$$

where x_i is the normalized value of z_i , and z_i^{max} and z_i^{min} are the maximum and minimum values of z_i in the database. It is recommended that the data be normalized between slightly offset values such as 0.1 and 0.9 rather than between 0 and 1 to avoid saturation of the sigmoid function leading to slow or no learning [Haussoun, 1995; Masters, 1994].

d. Input /output representation

Proper data representation also plays a role in the design of a successful ANN [Masters, 1994]. The data inputs and outputs can be continuous, discrete, or a mixture of both. For example, in a classification problem where each of the input variables belongs to one of several classes and the output is also a class, all the inputs and outputs may be represented by binary numbers such as 0 and 1 (or 0.1 and 0.9 to prevent saturation). If two inputs (A and B) are to be assigned to four levels of activation (e.g., low, medium, high, and very high), then each input may be represented by two binary numbers such as 00, 01, 10, and 11 to indicate the four levels. Binary inputs and outputs are very useful in extracting rules from a trained network. For this purpose, a continuous variable may be replaced by binary numbers by partitioning its range into a number of intervals, each assigned to a unique class. Specialized algorithms for discretizing variables based on their distribution also exist [Kerber, 1992].

e. Network weight initialization

Initialization of a network involves assigning initial values for the weights (and thresholds) of all connections links. Some researchers ,e.g., Li et. al., [Li, 1993; Schmidt, 1993] indicate that weights initialization can have an effect on network convergence. Weight initialization can also be performed on a neuron-by-neuron basis by assigning values uniformly sampled from the range $(-r/N_j + r/N_j)$, where r is a real number depending on the neuron activation function, and N_j is the number of connections feeding into neuron j .

f. BP learning rate (η)

A high learning rate, η , will accelerate training (because of the large step) by changing the weight vector, W , significantly from one cycle to another. However, this may cause the search to oscillate on the error surface and never converge, thus increasing the risk of overshooting a near-optimal W . A small learning rate drives the search steadily in the direction of the global minimum, though slowly. Wythoff (1993) suggests $\eta = 0.1- 1.0$, Zupan and Gasteiger [Zupan, 1993] recommend $\eta=0.3-0.6$, and Fu [Fu, 1995] recommends $\eta=0.0-1.0$. The adaptive learning rate, $\eta(t)$, which varies along the course of training, could also be used and can be effective in achieving an optimal weight vector for some problems.

g. Activation function

The transfer (activation) function is necessary to transform the weighted sum of all signals impinging onto a neuron so as to determine its firing intensity. Most applications utilizing BPANNs employ a sigmoid or a log sigmoid function, which possesses the distinctive properties of continuity and differentiability on $(-\infty, \infty)$ interval, essential requirements in BP learning. The advantage of choosing a particular transfer function over another is not yet theoretically understood.

h. Convergence criteria

Three different criteria may be used to stop training: (i) training error ($\rho \leq \epsilon$), (ii) gradient of error ($\nabla \rho \leq \delta$), and (iii) cross validation, where ρ is the

arbitrary error function, while ε and δ are small real numbers. The third criterion is more reliable; however it is computationally more demanding and often requires abundant data. Convergence is usually based on the error function, ρ , exhibiting deviation of the predictions from the corresponding target output values such as the sum of squares of deviations. Training proceeds until ρ reduces to a desired minimum. The most commonly used stopping criteria in neural network training is the sum-of-squared-errors (SSE) calculated for the training or test subsets as

$$SSE = \frac{1}{N} \sum_{p=1}^N \sum_{i=1}^M (t_{pi} - O_{pi})^2 \quad (5.25)$$

where O_{pi} and t_{pi} are, the actual and target solution of the i th output node on the p th example, N is the number of training example, and M is the number of output nodes. Generally, the error on training data decreases indefinitely with increasing number of hidden nodes or training cycles, as shown in Fig. 5.12.

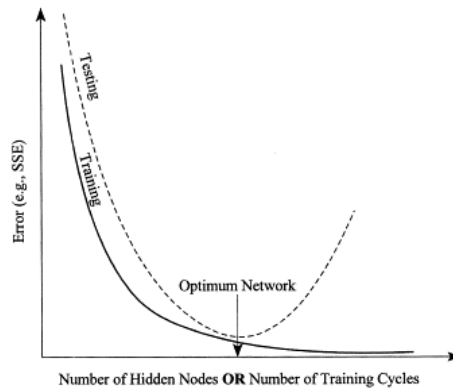


Fig. 5.12 Criteria of termination of training and selection of optimum network Architecture [Basheer, 2000]

i. Number of training cycles

The number of training cycles required for proper generalization may be determined by trial and error. For a given ANN architecture, the error in both training and test data is monitored for each training cycle. Training for so long can result in a network that can only serve as a look-up table, a phenomenon called overtraining or memorization [Zupan, 1993; Wythoff, 1993].

j. Training modes

Training examples are presented to the network in either one or a combination of two modes: (i) example-by-example training (EET), and (ii) batch training (BT) [Zupan, 1993; Wythoff, 1993]. In EET mode, the weights are updated immediately after the presentation of each training example. BT requires that weight updating be performed after all training examples have been presented to the network. That is, the first learning cycle will include the presentation of all the training examples, the error is averaged over all the training examples, and then backpropagated according to the BP learning law. Once done, the second cycle includes another presentation of all examples, and so on. The effectiveness of the two training modes can be problem specific [Hertz., 1991; Haykin, 2003; Swingler, 1996].

k. Hidden layer size

In most function approximation problems, one hidden layer is sufficient to approximate continuous functions [Basheer et al., 2000; Hecht-Nielsen, 1990]. Generally, two hidden layers may be necessary for learning functions with discontinuities. The determination of the appropriate number of hidden layers and number of hidden nodes (NHN) in each layer is one of the most critical tasks in ANN design. Unlike the input and output layers, one starts with no prior knowledge as to the number and size of hidden layers. As shown in Fig. 5.23, a network with too few hidden nodes would be incapable of differentiating between complex patterns leading to only a linear estimate of the actual trend. In contrast, if the network has too many hidden nodes it will follow the noise in the data due to over parameterization leading to poor generalization for untrained data (Fig. 5.23).

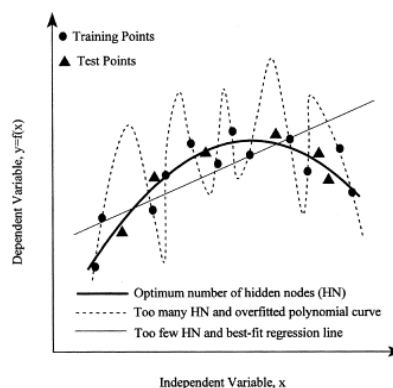


Fig. 5.13 Effect of hidden layer size on network generalization [Basheer, 2000]

Several rules of thumb are available in the literature which relate hidden layer size to the number of nodes in input ($NINP$) and output ($NOUT$) layers. Jadid and Fairbairn (1996) called for an upper bound on NHN equal to $NTRN / [R + (NINP + NOUT)]$, where $NTRN$ is the number of training patterns and $R = 5 - 10$. Lachtermacher and Fuller [Lacht, 1995] suggest that NHN for a one-output ANN with no biases be determined from $0.11NTRN \leq NHN(NINP + 1) \leq 0.30NTRN$. Masters, [Masters, 1994] suggests that the ANN architecture should resemble a pyramid with $NHN \approx (NINP \cdot NOUT)^{1/2}$.

1. Parameter optimization

As can be seen, BP training requires a good selection of values of several parameters, commonly through trial and error. Six parameters should not be set too high (large) or too low (small), and thus should be optimized or carefully selected.

5.6 Summary

This chapter discussed basics of speaker characteristics. Speaker recognition and speaker spectral characteristic, MFCC, have been introduced. Data clustering of MFCC which is required to classify speakers is explained. A discussion on Artificial Neural Network ends the chapter. Based on the discussions

in this chapter, implementation of a text dependant speaker recognition system is discussed in the next chapter.

Chapter 6

Development of a Text Dependant Speaker Recognition using MFCC Features and BPANN

In this chapter a text dependent speaker recognition method is developed. The spectral characteristics, Mel-frequency Cepstral Coefficients of a speaker are computed for a given sentence. The first 13 MFCCs are considered and each coefficient is clustered to form a 65 clustered in each coefficient to form the speech code vector. The speech code is trained using a multi-layer perceptron backpropagation gradient descent network and the network is tested for various test patterns. The performance is measured using FAR, FRR and EER parameters. The recognition rate achieved is about 96.18% for a cluster size of 5 in each coefficient.

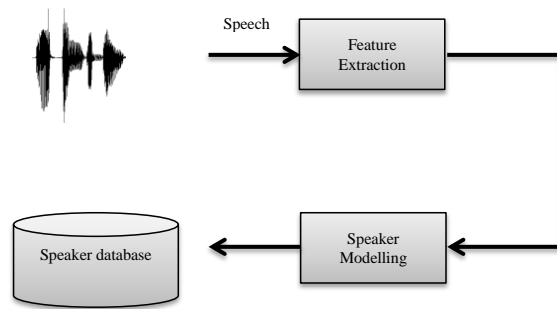
6.1 Introduction

Speaker verification/identification tasks are a typical pattern recognition problem. The important step in the speaker recognition is the extraction of relevant features from the speech data that is used to characterise the speakers. There are two speaker dependent voice characteristics: high level and low level attributes [Quatieri, 2010; Campbell, 2003]. Low level attributes related to the physical structure of the vocal tract are categorised as spectral characteristics where as high level attributes are like prosody (pitch, duration, energy) or behavioural cues like dialect, word usage, conversation patterns, topics of conversation etc. There are two types of speaker recognition methods: text dependant and text independent. Text dependent speaker recognition method uses phoneme context information and hence high recognition accuracy is easily achieved. The text independent speaker recognition method does not require specially designed utterances and hence is user friendly. [See chapter 5 section 5.4.1]. A general speaker recognition system consists of an enrolment phase and recognition phase as given in Fig. 6.1.

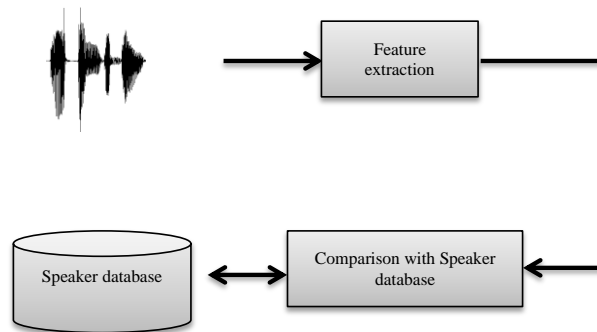
In the speaker enrolment phase, speech samples are collected from the speakers to train their models. The collection of enrolled models is also called a speaker database. In the identification phase, a test sample from an unknown speaker is compared against the speaker database. Both phases include the feature extraction step to extract the speaker dependent characteristics from speech.

In this research, speech samples of a given text are recorded for certain number of speakers. The acoustic analysis based on MFCC has proved good results in speaker recognition. Also MFCC has proved to be good in confrontation to different variation such as noise, prosody, intonation. Fig. 6.2 shows the text dependent speaker recognition system developed in this research. In the feature extraction or training phase, speech samples are recorded, normalized and pre-processed. The samples are then framed with a fixed frame size and an overlap. MFCCs are computed for all the frames of voiced speech samples. Vector Quantisation (VQ) based k -means clustering is done for the entire MFCC taking the cluster index. The code vector is trained using a discriminative classifier, multilayer perceptron with gradient descent backpropagation ANN. In the testing

phase, system is tested with another data set of test speech patterns. The performance of the system is measured using false acceptance rate (FAR), false rejection rate (FRR) and equal error rate (EER). The result is compared with minimum distance based classifier.

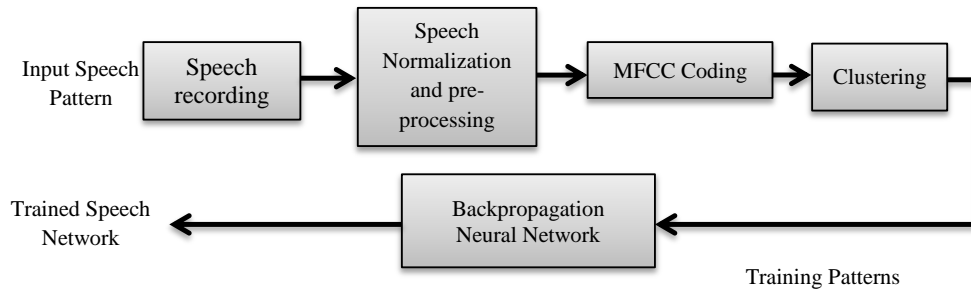


a) Speaker enrolment

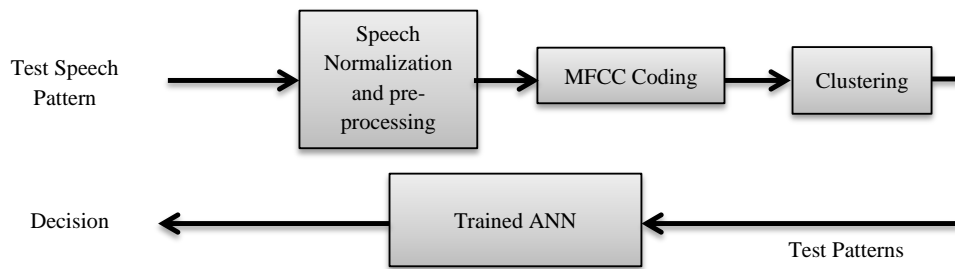


b) Speaker Verification

Fig. 6.1 Speaker Recognition System



a) Training phase



b) Testing phase

Fig. 6.2 Speaker Recognition steps

6.2 Feature Extraction

6.2.1 Speech recording

Since this is text dependent based speaker recognition, speech samples of the same text/sentence are to be recorded for all the speakers. The speakers are prompted to speak a sentence “open the door”. Speech samples of 40 speakers are recorded in a laboratory environment using a high quality microphone with a

DELL T7400 Intel Xeon based workstation computer under MATLAB 7.10.0.1 environment.

6.2.2 Speech Normalisation

In order to keep the signal level in between a set interval, each speech sample is normalised before pre-processing. Based on the maxima and minima of the signal, normalisation has been carried out according to the equation:

$$x_i = \lambda_1 + (\lambda_2 - \lambda_1) \left(\frac{z_i - z_i^{min}}{z_i^{max} - z_i^{min}} \right) \quad (6.1)$$

where x_i is the normalized value of z_i , and z_i^{max} and z_i^{min} are the maximum and minimum values of z_i in a speech file. λ_1 and λ_2 can be set to 0.1 and 0.9. The speech signal and normalized speech signal are shown in fig. 6.3

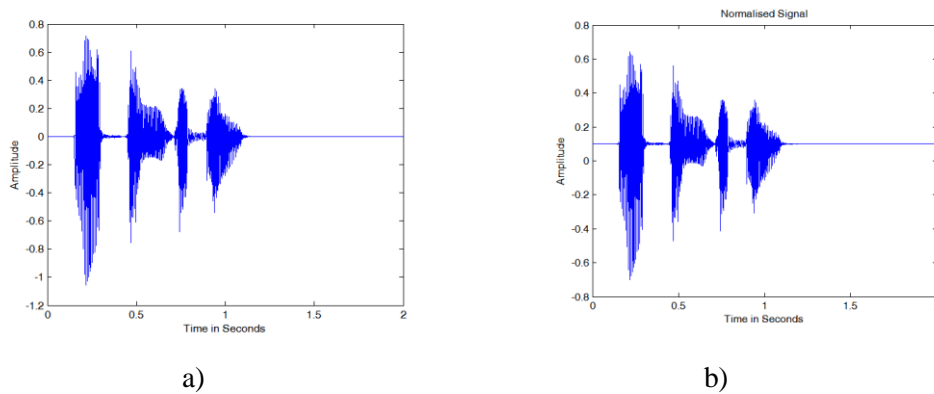


Fig. 6.3 a) Speech Signal b) Normalised speech signal

6.2.3 Pre-processing

As a pre-processing step, pre-emphasis filtering is done for the speech signals. The speech is processed by a high-emphasis filter before being subjected to the cepstrum analysis. This is required as the higher frequencies contain more

speaker-dependent information than the lower frequencies. A high pass filter with a transformation function given as:

$$H(z) = 1 - az^{-1} \quad (6.2)$$

Here the value of a is 0.95.

6.2.4 Voiced region extraction

MFCCs of the voiced regions are only computed in this work. Hence voiced regions are isolated from the speech signal by considering the energy of the signal. The signal energy, E_n is computed at each amplitude as per equation 6.3 and the voiced regions are extracted using the signal energy threshold. The voiced regions extracted for the normalized signal is shown in Fig. 6.4.

$$E_n = S_n^2 \quad (6.3)$$

where S_n is the n^{th} sample signal amplitude.

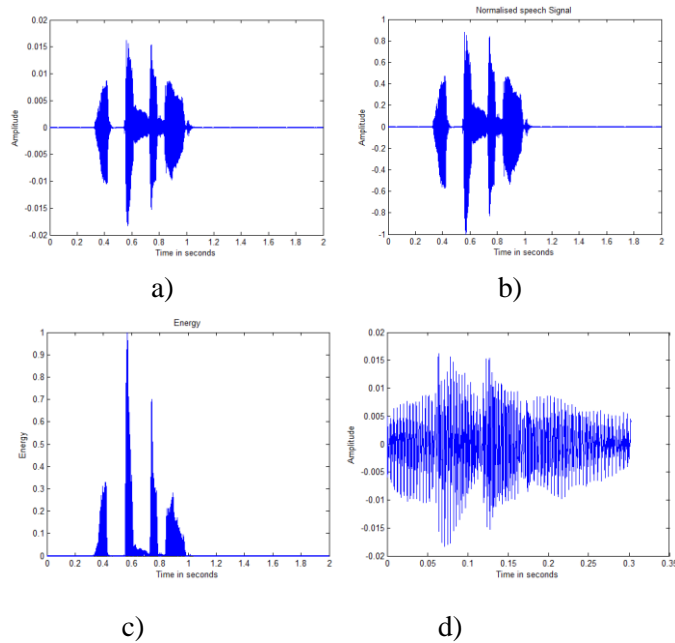


Fig. 6.4 a) Speech signal b) Normalized speech signal c) Energy of the signal computed d) Voiced regions extracted

6.2.5 MFCC Computation

This method is a short-term spectral analysis method and so speech signals are divided into short frames of fixed length. As discussed in chapter 5, section 5.3, a Hamming window, which is the most popular and common window is used with a window length of $N=256$. An N -point DFT is taken for each frame. The plots for a typical frame are shown in fig. 6.5.

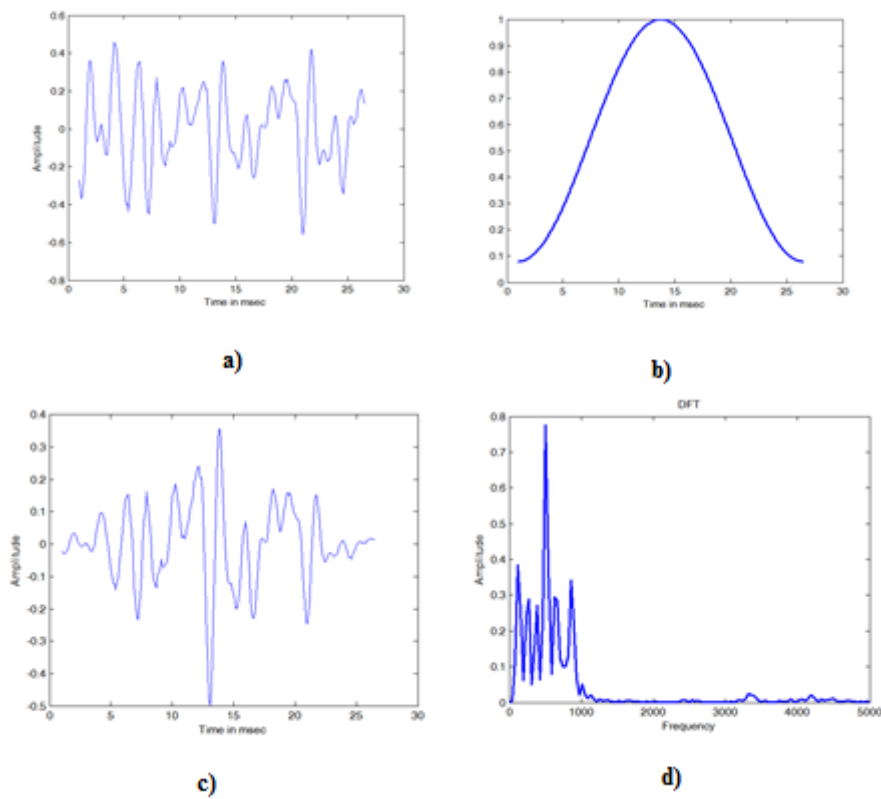


Fig 6.5 a) A Speech Frame b) Hamming window c) Windowed Frame d) DFT of windowed frame

The DFT of frames is weighted by triangular filter banks and the mel-frequency spectrum at any analysis time \hat{n} is computed as per equation 5.5. The next step is to compute the DCT of the log of the magnitude of the filter outputs of

each frame, which is the mel-frequency Cepstral coefficient of each frame, as given in equation 5.7 of chapter 5.

Table 6.1 shows the MFCC of speech samples computed with number of MFC coefficients, $N_{mfcc} = 13$ and the number of filter banks, $R=22$.

Table 6.1 MFCC of a single speech samples computed with $N_{mfcc} = 13$ and $R=22$. (Only 20 frames are shown)

	1	2	3	4	5	6	7	8	9	10	11	12	13
Frame 1	2.3	-4.03	-6.24	-0.86	0.21	-1.28	-2.17	-1.05	-0.27	-1.28	-1	-0.37	-0.13
2	4.15	-3.75	-6.13	-0.06	1.13	-1.4	-1.57	0.11	-0.35	-0.93	-0.88	0	0.16
3	3.39	-4.4	-6.95	-0.9	0.35	-2.1	-1.57	-0.67	-0.67	-1.11	-1.1	-0.19	0.02
4	3.07	-4.3	-5.71	-1.05	-0.23	-1.49	-1.32	-0.32	-0.4	-1.26	-0.97	-0.06	0.23
5	3.86	-4.06	-6.84	-1.47	-0.42	-2.69	-1.61	-0.15	-1.11	-1.39	-1	-0.55	-0.13
6	3.09	-3.46	-5.52	-1.99	-1.29	-1.93	-1.72	-0.35	-0.43	-1.04	-1.42	-0.99	-0.41
7	1.57	-0.36	-3.65	-1.15	0.47	-0.73	-1.18	-0.03	0.06	-0.41	-1.47	-1.2	-0.38
8	-0.82	0.01	-2.03	0.18	0.85	-0.18	-0.06	0.28	-0.1	-0.09	-1.11	-1.05	-0.33
9	-3.68	0.25	-0.77	0.08	1.44	0.07	-0.2	0.15	0.13	0.1	-0.16	-0.17	0.23
10	-4	0.09	-0.87	-0.74	0.41	-0.21	-0.42	0.09	0.39	-0.39	-0.48	-0.34	-0.21
11	-4.84	-0.04	-1.64	-1.47	-0.28	-0.02	0.61	0.55	0.44	0.06	-0.87	-0.66	-0.48
12	-5.23	-0.6	-1.82	-1.65	-1.4	-0.61	0.47	0.58	0.08	-0.81	-0.39	-0.56	-0.47
13	-6.51	-1.09	-1.91	-1.38	-1.07	0.02	0.14	0.25	0.03	-0.92	-0.55	-0.51	-0.4
14	-7.09	-0.96	-1.03	-0.58	-0.37	-0.12	-0.06	-0.35	0.26	0.34	0.24	-0.51	-0.25
15	-1.8	-2.74	-1.77	-0.52	-2.18	-0.28	0.42	-0.62	0.12	0.69	-0.3	-0.85	0.15
16	2.79	-5.47	-2.16	-0.19	-1.91	-0.68	0.69	0.18	-0.72	-0.29	0.2	-0.14	0.54
17	1.59	-6.71	0.1	-0.84	-0.94	0.43	0.55	-0.14	-1.06	-0.24	0.19	-0.49	0.57
18	-2.33	-6.5	0.27	-2.61	-1.54	-0.35	0.56	-1.8	-1.15	-0.52	0.09	-0.44	0.08
19	-2.79	-6.27	0.51	-3.24	-0.93	-0.79	0.83	-2.56	-0.25	-0.33	-0.61	-0.13	0.1
20	-2.74	-5.79	-0.35	-3.38	-1.96	-1.21	0.34	-2.53	-0.66	0.15	-1.57	-0.32	0.03

6.2.6 Data clustering using k-means method

The MFCC are to be calculated for about 40-45 frames. The number of coefficients (feature vector dimension) becomes 45×13 coefficients. In order to reduce the size, clustering of coefficients over each corresponding coefficient, in the different frames is done. This will limit the size of speech vectors of speakers to a (5×13) element speech code book.

k -means clustering, which is a squared mean clustering is done as per the algorithm given below:

k-Means Algorithm (LBG Algorithm):

1. Design a 1-vector codebook which represents the centroid of the entire set of training vectors.
2. Double the size of the codebook by splitting each current codebook \mathbf{y}_n according to rule

$$\begin{aligned}\mathbf{y}_n^+ &= \mathbf{y}_n(1 + \varepsilon) \\ \mathbf{y}_n^- &= \mathbf{y}_n(1 - \varepsilon)\end{aligned}$$

where n varies from 1 to the current size of the codebook, and ε is a splitting parameter (eg. $\varepsilon=0.01$)

3. Nearest-Neighbor Search: for each training vector, find the codeword in the current codebook that is closest in terms of similarity measurement and assign that vector to the corresponding cell.
4. Centroid update: update the codeword in each cell using the centroid of the training vectors assigned to that cell.
5. Iteration 1: repeat steps 3 and 4 until the average distance falls below a preset threshold
6. Iteration 2: repeat steps 2, 3 and 4 until a codebook size of M is designed.

The LBG algorithm designs an M -vector codebook in stages. It starts first by designing 1-vector codebook, then uses a splitting technique on the codewords to initialize the search for a 2-vector codebook, and continues the splitting process until the desired M -vector codebook is obtained. In this work, clustering is done for various cluster sizes ranging from 4 to 10 to find an optimum cluster size which will give the maximum performance during speaker recognition. Clustered speech codebook for cluster size 5 is given in table 6.2.

Table 6.2 *k*-means clustered MFCCs of the speech sample computed with 5 cluster centres along each column

1	2	3	4	5	6	7	8	9	10	11	12	13
3.39	1.53	0.2	-1.7	-4.87	1.72	0.09	-1.22	-3.52	-5.63	0.27	-0.82	-2.07
-3.63	-6.23	0.17	-0.52	-1.13	-1.99	-2.99	1.14	0.02	-1.2	-2.5	-3.65	0.15
-0.3	-0.74	-1.23	-1.86	0.64	0.26	-0.19	-1.09	-1.73	0.41	0	-0.5	-1.21
-2.13	0.17	-0.27	-0.76	-1.22	-2.24	0.98	0.48	0.06	-0.39	-1.1	0.18	-0.45
-1.04	-1.43	-2.03	0.7	0.46	-0.22	-0.58	-1.07	0.85	0.56	0.1	-0.17	-0.4

6.3 Neural Network Design and Training

6.3.1 Network Architecture

Multilayer perceptrons are feedforward neural networks trained using the backpropagation algorithm [Haykin, 2004]. They can be used as binary classifiers for speaker verification to separate the speaker and non-speaker classes. In this research a multilayer perceptron artificial neural network is trained with backpropagation algorithm.

Design of a neural network involves the determination of number of neurons in the input layer, number of hidden layers and their size and determination of a proper activation function in the hidden and the output layer. In this work, the input layer size depends on the number of clustering coefficient. For example, a 5×13 codebook requires 65 numbers of input neurons. The number of hidden layer is set to 1, which is optimum as per the criterion given in chapter 5, section 5.6.2. The activation function taken for both the hidden and output layer is log-sigmoid. Fig. 6.6 shows the basic architecture of multilayer perceptron ANN for identifying 40 numbers of speakers. It contains 65 input neurons, 1 hidden layer with 40 neurons and 40 output layer neurons.

6.3.2 Data Normalisation

Before giving the data for training to the network, normalization is done as per equation 6.1 with $\lambda_1 = 0.1$ and $\lambda_2 = 0.9$ to avoid saturation of the sigmoid

function leading to slow or no learning. For each speaker 5 speech samples of the same sentence have been recorded in the laboratory environment and stored. The clustered MFCCs (65 nos) are computed is arranged as a column vector for each speech data.

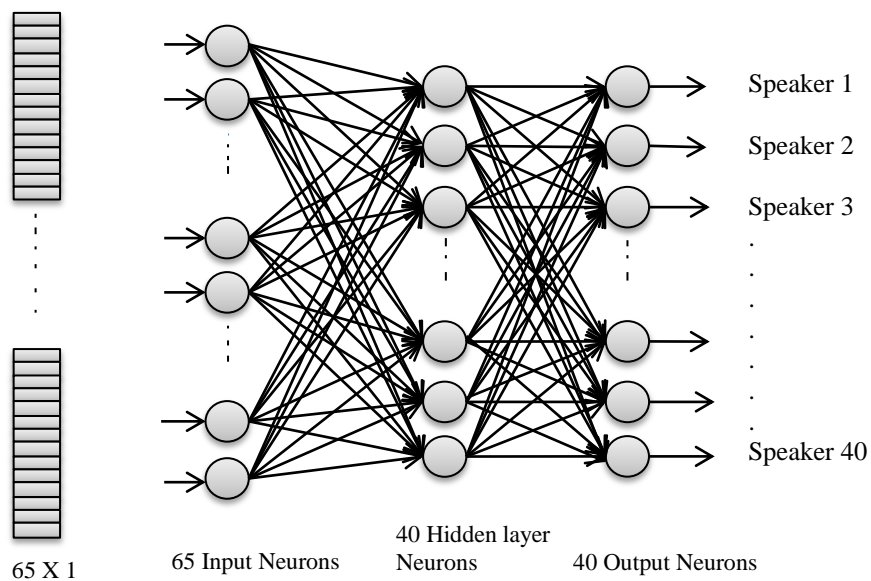


Fig. 6.6 Neural Network Architecture for Speaker Recognition

Speech feature vectors of all the 40 speakers are normalised and arranged as a matrix of size 65×200 (for a cluster size of 5). Column 1-5 correspond to speaker 1, column 6-10 correspond to speaker 2 and likewise (Table. 6.3).

6.3.3 ANN training

The BPANN is trained as per the algorithm given in chapter 5, section 5.5.1, with the input neuron size as a function of cluster size. The network used is gradient decent backpropagation with adaptive learning rate for 1000 epochs. The performance and gradient is set as $1e-08$ and $1e-10$.

Table 6.3 MFCC database for ANN training. (Column 1-5 belongs to speaker 1, column 6-10 belongs to speaker 2. Only 31 of the 65 elements of speech vector are shown)

1	2	3	4	5	6	7	8	9	10
0.677366	0.69571	0.666782	0.671767	0.677375	0.651942	0.703869	0.687714	0.828368	0.747015
0.589055	0.607622	0.635558	0.605292	0.608384	0.586128	0.650438	0.632041	0.688933	0.61133
0.526539	0.53164	0.563182	0.532357	0.493563	0.504812	0.580154	0.553229	0.567629	0.519409
0.436378	0.452035	0.480784	0.435261	0.385987	0.392028	0.491674	0.449511	0.47193	0.446029
0.286355	0.315605	0.259942	0.229054	0.220231	0.237902	0.382183	0.282617	0.341171	0.316973
0.598294	0.593041	0.605607	0.587208	0.603059	0.620087	0.64016	0.659477	0.673637	0.642869
0.521052	0.525355	0.531718	0.561671	0.573043	0.545372	0.589077	0.592349	0.629917	0.600196
0.458964	0.437222	0.425084	0.449068	0.478077	0.442349	0.511742	0.492301	0.555	0.501173
0.35039	0.350232	0.2879	0.340548	0.360719	0.351373	0.403896	0.356772	0.40624	0.35888
0.250397	0.280944	0.222563	0.207611	0.232921	0.232184	0.320286	0.259776	0.320562	0.260759
0.529552	0.520557	0.538685	0.499524	0.531228	0.529274	0.532228	0.563811	0.549058	0.55614
0.477912	0.478085	0.486235	0.45308	0.485729	0.486333	0.490121	0.516544	0.510788	0.514339
0.418958	0.429396	0.430718	0.411574	0.426883	0.429421	0.437533	0.465666	0.456084	0.480118
0.345161	0.347587	0.358282	0.348515	0.353405	0.351802	0.254681	0.406455	0.368749	0.431295
0.221855	0.246464	0.242309	0.247141	0.240287	0.214791	0.154711	0.275601	0.235812	0.332396
0.524812	0.525598	0.546341	0.50432	0.534814	0.538772	0.558015	0.547344	0.563795	0.554481
0.492256	0.490166	0.509588	0.470696	0.515018	0.488018	0.50651	0.506461	0.507314	0.515812
0.463437	0.451821	0.467941	0.44475	0.462587	0.444056	0.450904	0.459959	0.437704	0.447221
0.422476	0.407779	0.413176	0.417841	0.416612	0.404124	0.389344	0.377918	0.371963	0.374828
0.375529	0.368829	0.350898	0.390422	0.358487	0.364196	0.316958	0.281941	0.285738	0.311483
0.570846	0.563893	0.540591	0.511628	0.547387	0.557002	0.550561	0.542272	0.561836	0.575761
0.517727	0.52193	0.525534	0.485923	0.496588	0.51698	0.518485	0.496628	0.520605	0.537023
0.460183	0.465838	0.488075	0.446554	0.44739	0.453719	0.470627	0.447433	0.48739	0.465252
0.398646	0.404838	0.40497	0.39607	0.392947	0.395625	0.406539	0.40345	0.406514	0.424269
0.344015	0.345258	0.356977	0.344974	0.328306	0.328458	0.335907	0.367443	0.368679	0.377804
0.523742	0.551213	0.575207	0.525384	0.50996	0.535547	0.548958	0.545884	0.559638	0.60555
0.502492	0.527364	0.531835	0.507019	0.493514	0.514476	0.526756	0.52691	0.528108	0.585972
0.48179	0.507354	0.487558	0.48624	0.478789	0.488714	0.495215	0.494604	0.498668	0.508225
0.458466	0.47241	0.444778	0.453729	0.454741	0.462159	0.457312	0.448512	0.472369	0.463861
0.428783	0.431293	0.413515	0.418286	0.424391	0.424395	0.431875	0.413928	0.432995	0.43289
0.54704	0.578124	0.581729	0.606258	0.571779	0.575728	0.57557	0.602832	0.576848	0.586913

For each input cluster sizes, neural networks are trained and the corresponding net is stored and prepared for testing with a test database. The test database consists of

known and unknown speakers which are classified as genuine and impostor speakers. 200 speech samples in the test database are tested and the false acceptance rate (FAR) and false rejection rate (FRR) of the system are estimated.

6.4 Implementation

6.4.1 Database used

In this work, speech samples of 40 persons for a given prompted text of duration 2 sec are recorded using a high quality microphone under MATLAB 7.10.0.1 environment. The sampling frequency used to record is 10 K Hz. For each speaker, 10 samples are stored of which 5 samples are used for training phase and other 5 samples are taken for testing/validating the algorithm.

6.4.2 Implementation

Before extracting the features from the speech signal, each speech sample is normalised using the equation 6.1 to set the signal amplitude within a desired range i.e., between 0.1 and 0.9. Each speech sample is then subjected to a pre-emphasis filtering as per equation 6.2 with $a=0.95$. In the next step, the voiced regions in the filtered waveform are extracted using the energy computed from the signal amplitude using equation 6.3. For energy $E_n \geq 0.01$, the signal regions are sampled as voiced regions. The voiced regions extracted for a speech signal are shown in fig. 6.4d.

MFCCs of speech signals are extracted using the equation 5.6 (chapter 5) for the voiced regions. For this, speech signals are framed for about 256 samples (about 26 msec) with a frame overlap of 100 samples. FFT is computed for each speech signals and the spectrum is weighted with 22 Triangular filter banks. MFCC of each frame is calculated and stored.

In this study, MFCCs of speech samples are computed and clustered with clusters of different sizes, i.e., clusters of size 10, 8, 7, 5 and 4 using k -means clustering algorithm. Backpropagation based ANN is designed with the input neuron size dependant on the cluster size i.e., for cluster size of 5, the number of

input neurons is 65. In this case a gradient descent based backpropagation with adaptive learning rate is selected for training. The number of hidden layer is set as 1 and the hidden and output neuron size is set as 40, irrespective of the input neuron dimension. 5 samples each of the 40 speakers are used to train the network. The trained network is stored for testing/validation.

6.4.3 Results and Discussion

Out of 10 speech samples recorded for each speaker, 5 samples are taken for each of the 40 speeches for feature extraction and training. i.e., the input data size for training the data is of size $200 \times$ clustered MFCC data size as mentioned in section 6.3.1. ANN is designed and trained for various input cluster sizes of 10, 8, 5 and 4. The False Acceptance Rate (FAR) and False Rejection Rate (FRR) of the system for various clusters are shown in Figs. 6.7-6.11. The Equal Error Rates (EERs) corresponding to each is shown in Table 6.4 and has a minimum EER for the cluster of size 5. Hence cluster of size 5 can be considered as the best option for the recognition, with the EER is approximately equal to 0.0382 within a matching threshold of 0.152-0.167. Thus the maximum genuine acceptance rate or recognition rate achieved for this system is 96.18%.

To compare the performance of the ANN based classifier, VQ based minimum distance classifier (mean-squared difference) is also designed as per the discussion given in section 5.3.1 of chapter 5. As in the case of the ANN based classifier, here also, 5 samples each were considered for a speaker. For a given speaker the train database consists of average of 5 the corresponding clustered MFCC for a single speaker. The test database consists of the other 5 clustered MFCC of the speakers. Euclidean distance between the average training vectors and test vectors are calculated using equation 5.23 and 5.24 and the EER of the system is found out for various cluster size. The EER observed for the cluster size of 5 is 0.1662 which is the minimum EER obtained among all the clusters. That is only 83.38% of recognition rate is achieved using VQ based minimum distance classifier method.

A comparison of ANN and minimum distance based classifiers are shown in Table 6.5 from which the neural network method is shown to be superior for speaker identification with minimum error.

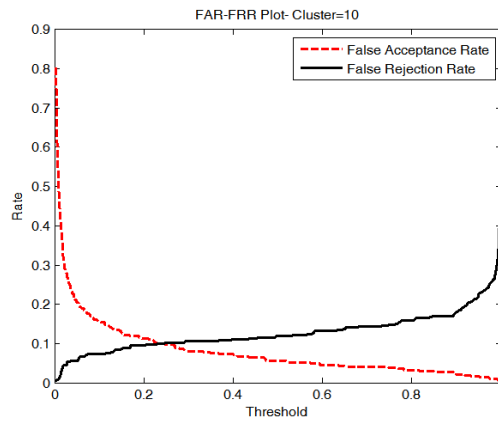


Fig. 6.7 FAR-FRR plot for cluster size=10. EER=0.0997 at a Threshold =0.232 to 0.247

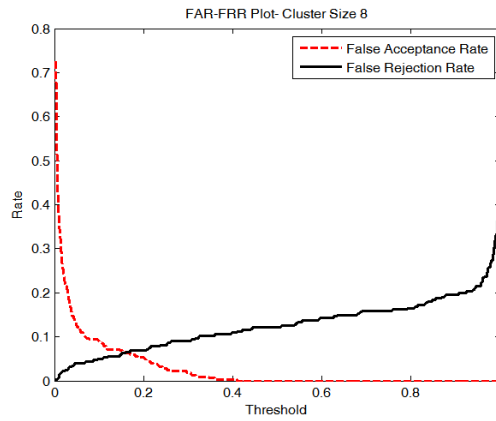


Fig. 6.8 FAR-FRR plot for cluster size=8. EER=0.0639 at a Threshold of 0.1625

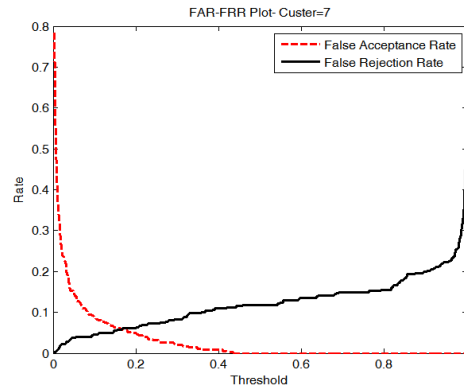


Fig. 6.9 FAR-

cluster size=7. EER=0.0576 at a Threshold of 0.163-0.166

FRR plot for

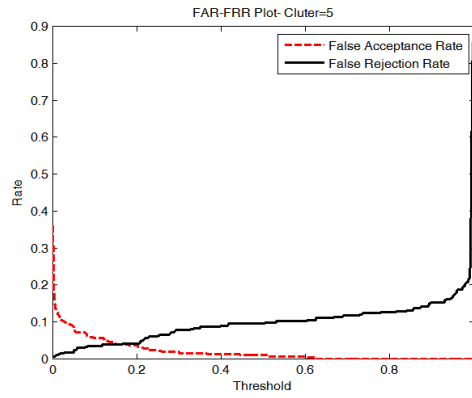


Fig. 6.10 FAR-FRR plot for cluster size=5. EER=0.0382 at a Threshold of 0.152-0.167

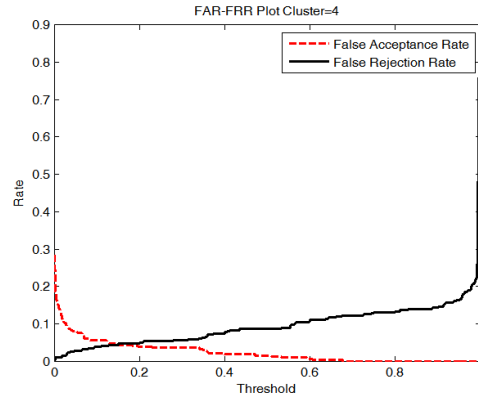


Fig. 6.11 FAR-FRR plot for cluster size=4. EER=0.0452 at a Threshold of 0.1505

Table 6.4 Cluster Size and EER for ANN based classifier

Cluster Size	EER	Threshold Range
10	0.0997	0.2320
8	0.0639	0.1625
7	0.0576	0.163-0.166
5	0.0382	0.152-0.167
4	0.0452	0.1505

Table 6.5 Comparison of EER for ANN and Euclidean distance classifier

Cluster Size	EER	
	Euclidean distance	Neural Network
10	0.2456	0.0997
8	0.2093	0.0639
7	0.1812	0.0576
5	0.1662	0.0382
4	0.1753	0.0452

6.5 Conclusion

In this chapter speaker recognition technique based on the spectral characteristics, mel-frequency Cepstral coefficient, was developed. The speech signals were normalized and the voice regions were extracted to compute the MFCC. MFCC of each speech signals were computed. *k*-means clustering was done for various cluster sizes. A discriminative classifier, multilayer perceptron ANNs, were designed and trained to recognize 40 speakers for various input neuron dimensions. The maximum recognition rate achieved was 96.18% for a cluster size of 5.

VQ based minimum Euclidean distance classifiers were designed and the recognition rate achieved was 83.34% for a cluster size of 5. Hence the neural network based speaker recognition system was found to be superior. In the next chapter, multibiometric fusion of fingerprint and speech modalities are discussed and multibiometric personal authentication systems are developed based on fingerprint and speech modalities under various levels of fusion.

Chapter 7

Multimodal biometric fusion

In this chapter multimodal based biometric fusion of traits are introduced with various levels of fusion. A multifinger feature level fusion based authentication technique is implemented first and the performance is compared with unimodal based fingerprint recognition system. Also a score level fusion based recognition system is developed using fingerprint and speech, and the equal error rate (EER) measured shows a good improvement with 100% recognition rate over a large span of match score thresholds.

7.1 Introduction

Biometric systems that use a single modality are usually affected by problems like noisy sensor data, non-universality and/or lack of distinctiveness of the biometric trait, unacceptable error rates, and spoof attacks [Jain, 2004]. A multibiometric system improves the performance of the system by considering several traits from different scores [Jain, 2005]. Multibiometric systems deal with two or more evidences taken from different sources like multiple fingers of the same person, multiple samples of the same instances, multiple sensors for the same biometric, multiple algorithms for representation and matching or multiple traits. A multibiometric system that uses different biometric traits is expected to be more robust to noise, address the problem of non-universality, improve the matching accuracy and provide reasonable protection against spoof attacks and thus multibiometric system has received considerable attention among researchers [Jain, 2005].

Various levels of fusion are possible in a multibiometric system that uses different biometric traits: fusion at the feature extraction level, matching score level or decision level. Feature level fusion is quite difficult to consolidate as the feature sets used by different modalities may either be inaccessible or incompatible. Fusion at the decision level is too rigid since only a limited amount of information is available at this level. Therefore integration of the matching score level is generally preferred due to the ease of accessing and combining matching scores.

In the case of verification, fusion at the matching score level can be approached in two distinct ways: in the first approach the fusion is viewed as a classification problem while in the second approach the fusion is viewed as a combination problem. In the classification approach, a feature vector is constructed using the matching scores output by the individual matchers; this feature is then classified into one of the two classes: “Accept” for a genuine user or “Reject” for an imposter [Jain, 2005]. In the other case the individual matching scores are combined to generate a single scalar score which is then used to make the final decision. The combination approaches have been shown to be the better

approach by Ross and Jain [Ross, 2003]. Combination approach has been implemented in this work and modalities taken are fingerprint and speech.

7.2 Information Fusion in Multimodal Biometrics

Sanderson and Paliwal [Sanderson, 2004] defines information fusion as “.. any area which deals with utilizing a combination of different sources of information, either to generate one representational format or to reach a decision”. According to them the information fusion can be classified into three main categories: Pre-mapping fusion, midst-mapping fusion and post-mapping fusion (Fig. 7.1). In pre-mapping fusion, information is combined before any use of classifiers or experts; in midst-mapping fusion, information is combined during mapping from sensor-data/feature space into opinion/decision space, while in post-mapping fusion, information is combined after mapping from sensor-data/feature space into opinion/decision space.

Pre-mapping fusion is categorized as sensor data level fusion and feature level fusion. In post-mapping fusion, there are two main sub-categories: decision fusion and opinion fusion. Ross and Jain refer to opinion fusion as score fusion. A brief discussion on the various fusion methods is presented in the next section.

7.2.1 Pre-mapping fusion: sensor data level

In this level the raw data from sensors is combined [Hall, 2001]. There are two methods to combine the sensor level data: weighted summation and mosaic construction. Weighted summation can be employed to combine visual and infrared images into one image, or, in the form of an average operation, to combine the data from two microphones. Mosaic construction can be employed to create one image out of images provided by several cameras, where each camera is observing a different part of the same object [Iyengar, 1995].

7.2.2 Pre-mapping fusion: feature level

In feature level fusion, features extracted from data provided by several sensors are combined. If the features are proportionate, the combination can be accomplished by a weighted summation (e.g., features extracted from data provided by two microphones).

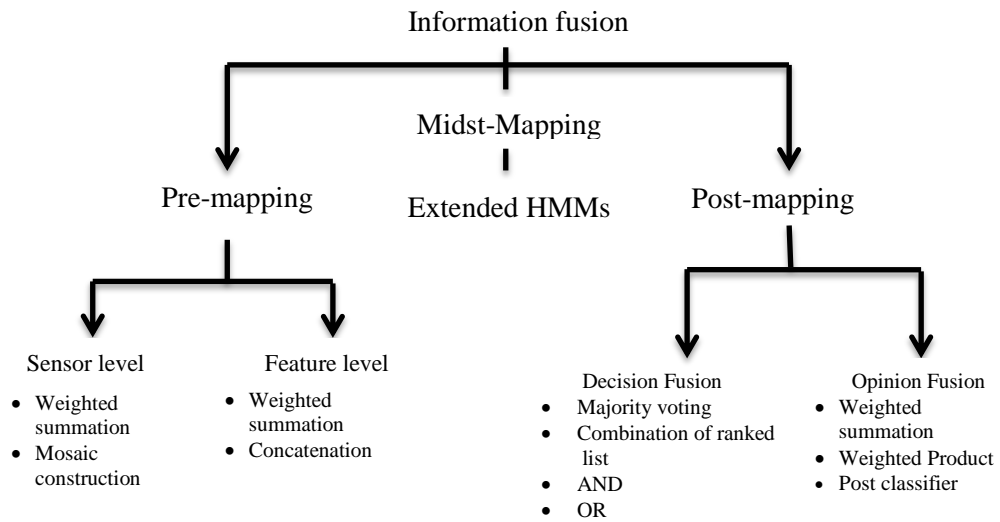


Fig. 7.1 Levels of Fusion

If the features are disproportionate, feature concatenation can be employed [Adjoudani, 1995; Hall, 2001; Leutiin, 1997; Ross, 2003], where a new feature vector can be constructed by concatenating two or more feature vectors. Mathematically, a feature-level fusion function $f_R(\cdot)$ converts a collection of M biometric feature sets $X=\{x_1, x_2, \dots, x_M\}$ into a single fused feature set x_f , i.e.,

$$\mathbf{x}_f = \mathbf{f}_R(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_M). \quad (7.1)$$

Feature-level fusion schemes can be categorized into two broad classes, namely homogenous and heterogeneous. A homogeneous feature fusion scheme is

used when the feature sets to be combined are obtained by applying the same feature extraction algorithm to multiple samples of the same biometric trait (e.g., minutia sets from two impressions of the same finger). This approach is applicable to multi-sample and multi-sensor systems.

There are three disadvantages to the feature vector concatenation approach:

1. There is no explicit control over how much each vector contributes to the final decision.
2. The separate feature vectors must be available at the same frame rate, which is a problem when combining speech and visual feature vectors.
3. The dimensionality of the resulting feature vector, which can lead to the “curse of dimensionality” problem as defined by Duda, et al. [Duda, 2010].

7.2.3 Midst-mapping fusion

Midst-mapping fusion is a relatively new and a more complex concept. Here several information streams are processed concurrently while mapping from feature space into opinion/decision space. Midst-mapping fusion can be used for exploitation of temporal synergies between the streams, with the ability to avoid problems present in vector concatenation. In this type of fusion extended Hidden Markov models have been shown useful for text-dependent person verification.

7.2.4 Post-mapping fusion: decision fusion

Each classifier in an ensemble of classifiers provides a hard decision [Hall, 2001; Iyengar, 1995]. The classifiers can be of the same type but working with different features (e.g. audio and video data), non-homogeneous classifiers working with the same features, or a hybrid of the previous two types. The decisions can be combined by majority voting, combination of ranked lists, or using AND&OR operators.

7.2.4.1 Majority Voting

In this fusion, a consensus is reached on the decision by having a majority of the classifiers declaring the same decision [Ho, 1994; Iyengar, 1995,

Radová, 1997]. For a two class classification task, the number of classifiers must be odd and greater than two to prevent ties.

7.2.4.2 *Ranked list combination*

In this method, each classifier provides a ranked list of class labels, with the top entry indicating the most preferred class and the bottom entry indicating the least preferred class [Achermann, 1996; Ho, 1994]. By combining using various means, the decision is reached by selecting the top entry in the combined ranked list [Ho, 1994].

7.2.4.3 *AND fusion*

In AND fusion [Luo, 1995; Varshney, 1997], a decision is reached only when all the classifiers agree. This type of fusion is quite restrictive. For multi-class problems no decision may be reached, thus it is mainly useful in situations where one would like to detect the presence of an event/object, with a low false acceptance bias.

7.2.4.4 *OR fusion*

In OR fusion [Luo, 1995; Varshney, 1997], a decision is made as soon as one of the classifiers makes a decision. This type of fusion is very relaxed providing multiple possible decisions in multi-class problems. OR fusion is mainly useful where one would like to detect the presence of an event/object with a low false rejection bias.

7.2.4.5 *Post mapping fusion*

This type of fusion is also referred to as score level fusion in which, an ensemble of experts provides an opinion on each possible decision [Hall, 2001; Iyengar, 1995; Ross, 2003]. In the case of matching scores output by the various modalities are heterogeneous, score normalization is needed to transform these scores into common domain prior to combining them [Jain, 2005]. This is accomplished by mapping the output to the [0, 1] interval, where 0 indicates the lowest opinion and 1 the highest opinion. The normalization techniques that are

commonly used are: min-max, decimal scaling, z -score, median and MAD, double sigmoid, tanh-estimators and biweight estimators [Jain, 2005].

Opinions can be combined using weighted summation or weighted product approaches before using a classification criteria such as the MAX operator which selects the class with the highest opinion to reach a decision. A post-classifier can be used to directly reach a decision.

The natural benefit of weighted summation and product fusion over feature vector concatenation and decision fusion is that the opinions from each expert can be weighted. The weights can be set to reflect the reliability and discrimination ability of each expert; thus when fusing opinions from a speech and a face expert, it is possible to decrease the contribution of the speech expert when working in low audio SNR conditions.

7.2.4.5.1 Weighted summation fusion

In this method, the opinions regarding class j from N_E experts are combined using the expression:

$$f_j = \sum_{i=1}^{N_E} w_i o_{i,j} \quad (7.2)$$

where $o_{i,j}$ is the opinion from the i^{th} expert and w_i is the corresponding weight in the $[0, 1]$ interval, with the constraints $\sum_{i=1}^{N_E} w_i = 1$ and $\forall i: w_i \geq 0$. This approach is also known as linear opinion pool [Altiçay, 2000] and sum rule [Alexandre, 2001; Kittler, 1998]

7.2.4.5.2 Weighted product fusion

Assuming the experts are independent, the opinions regarding class j from N_E experts can be combined using a product rule

$$f_j = \prod_{i=1}^{N_E} o_{i,j} \quad (7.3)$$

By introducing weighting to account for varying discrimination ability and reliability of each expert,

$$f_j = \prod_{i=1}^{N_E} (o_{i,j})^{w_i} . \quad (7.4)$$

The weighted product approach is also known as logarithmic opinion pool [Altiçay, 2000] and product rule [Alexandre, 2001; Kitler, 1998]. There are two downsides to weighted product fusion [Sanderson, 2004]:

1. The first one is that one expert can have a large influence over the fused opinion.
2. The second downside is that the independence assumption is only strictly valid when each expert is using independent features.

7.2.4.5.3 Post-classifier

All experts indicate the opinions as a likelihood of a particular class, the opinions from N_E experts regarding N_C classes from $N_E N_C$ -dimensional opinion vector, which is used by a classifier to make the final decision. This type of classifier is referred to as a post-classifier. The main disadvantage of this approach is that the resultant dimensionality of the opinion vector is dependent on the number of experts as well as the number of classes, which can be quite large in some applications. However, in a verification application, the dimensionality of the opinion vector is usually only dependent on the number of experts [Yacoub, 1999]. Each expert provides only one opinion, indicating the likelihood that a given claimant is the true claimant. The post-classifier then provides a decision boundary in N_E -dimensional space, separating the impostor and true claimant classes.

7.2.5 Hybrid fusion

It may be necessary to combine various fusion techniques due to practical consideration. E.g. Hong and Jain [Hong, 1998] used a fingerprint expert and a frontal face expert; a hybrid fusion scheme involving a ranked list and opinion fusion was used: opinions of the face expert for the top n identities were combined with the opinions of the fingerprint expert for the corresponding identities using a form of the product approach. This hybrid approach was used to take into account the relative computational complexity of the fingerprint expert [Sanderson, 2004].

In this work, feature level fusion of fingerprints is done first using the global singularity based feature representation and the FAR and FRR are plotted for various thresholds. The result is compared with single finger based recognition. In the next work, a multimodal recognition system is developed based on fingerprint and speech modalities. The fingerprint recognition score is combined with speech based recognition score using sum and product rule. The FAR and FRR are plotted for various thresholds and EER is measured. The results are compared with the unimodal systems.

7.3 Development of a Multifinger Feature Level based fingerprint recognition system

In this work, consecutive fingerprints of a person are acquired using an optical based fingerprint sensor. The individual images are processed separately to obtain the fingerprint feature vectors. The feature extractions of the fingerprints are based on the global singularity features as explained in chapter 4 and can be summed as given as:

1. Orientation field estimation and strength
2. Singularity detection
3. Baseline detection and polygon formation
4. Feature vector formation by fusion

Orientation field is estimated using squared gradients proposed by Kass and Witkin [Kass, 1987] as done in section 4.2.2. The block size taken is 16×16 and Sobel operator is used to find the gradients in the x & y -directions. The orientation field is computed using equation 3.6 of chapter 3. From the orientation field, strength of orientation, *coherence*, is found out as per equation 3.11 of chapter 3. From the coherence image, singularities can be detected using the lowest value coherence in the image. Fig 7.2 shows the successive fingerprint images with low value coherence which clearly defines the singularities – core and delta. The fingerprint base line is detected using correlation method by pre-stored masks as discussed in chapter 4, section, 4.2.3. The most slanted line that can be detected is about $\pm 23^\circ$. From the fingerprint singularities perpendiculars are drawn to the baseline to form a fingerprint polygon (Fig. 7.3.). The angle and distance features are evaluated from this polygon. The ridge counts are obtained by counting the number of intensity minima between the desired singularities in the polygon. The above steps are done for each of the two fingers.

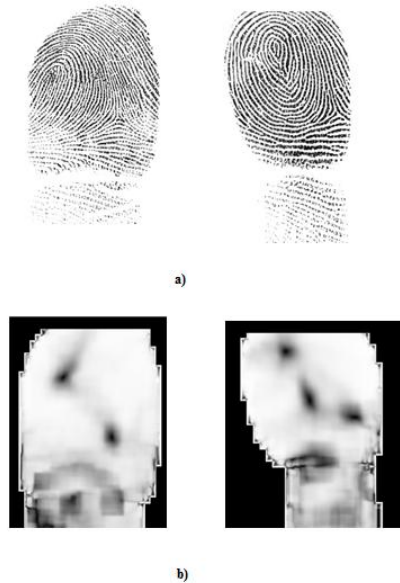


Fig. 7.2. a) Multifinger Fingerprint images of successive fingers b) coherence images showing singularities



Fig. 7.3. Fingerprint and Polygon formed

7.3.1 Feature vector formation by fusion

For each fingerprint, a set of 16 element polygonal metrics is found out. Let $X = \{x_1, x_2, \dots, x_m\}$ and $Y = \{y_1, y_2, \dots, y_n\}$ denote the 16 element feature vectors ($x \in R^m$) and ($y \in R^n$) of the two fingers. The feature vector of the combined fingers is defined as $Z = X \cup Y$. In this case, feature normalization is not required as the features are from the same modality and are homogeneous. Thus a 32 element feature vector is formed. If the fingerprint feature polygon vectors of first finger are denoted by $(d_{1i}, \theta_{1i}, t_{1i})$ and that of second finger is $(d_{2i}, \theta_{2i}, t_{2i})$ then the feature level fused template is given by the vector $(d_{1i}, \theta_{1i}, t_{1i}, d_{2i}, \theta_{2i}, t_{2i})$.

7.3.2 Fingerprint matching score

Matching gives a numerical score which shows how much the input image (I) matches with the existing fingerprint template (T). Any fingerprint algorithm compares above two given fingerprints and returns either a degree of similarity or a binary decision. The matching score which is a number in the range 0 to 1 is calculated as the ratio of the number of matched features to the total number of features. In this work we have formulated a simple matching score based on the Euclidean distance which is an extension of equation 4.23. The steps for computing the matching score are:

1. Compare the types t_{1i} and t_{2i} of the input fingerprint template with that of the feature templates. If they are not matching, match score between the two template is zero, else go to step 2
2. Find the Euclidean distances between the corresponding distance features and angles of the input image I and query template T .
3. If $\sum_{n=1}^N |I_n - F_n| * w_n > Th$, it is an imposter. Th is the threshold value assigned empirically, typically a value of 400 based on the sum of the weighted difference, I_n and F_n are the input image and Template metrics, N is the number of features, which is 32, w_n is the weight factor for each feature.
4. $\sum_{n=1}^N |I_n - F_n| * w_n \leq Th$, compute the match score as per the equation followed to get a match score of 1 and 0 for a genuine and imposter

$$M = M = \frac{Th - \sum_{n=1}^N |I_n - F_n| * w_n}{Th} \quad (7.5)$$

Th is the threshold value assigned which is empirically found out as 400, w_n is the weight assigned for each metric, I_n and F_n are the input image and template metrics, N is the feature vector length, which is 32 in this case.

5. If $M \leq t$, fingerprint matches, where $0 \leq t \leq 1$ is the matching threshold.

7.3.3 Implementation of the Algorithm

Implementation is done in a manner similar to that of the single fingerprint implementation.

7.3.3.1 Database used

In this work, consecutive fingerprint images (forefinger and middle finger) have been acquired using a fingerprint scanner, enBioscan-F with 500 dpi resolution and image size of 600×600 pixels. Fingerprint samples of about 300 persons are collected with 4 samples per finger and features are extracted for 2 samples each for creating the database. The rest of the 2 samples of each finger are used for validation of the algorithm.

7.3.3.2 Implementation

Individual fingerprints are processed for finding the orientation field with a 16×16 block size, from which the coherence values are also calculated. The singularities are detected and identified from the coherence image. Baseline is captured clearly for individual fingerprints and the polygon is drawn for whorl, left loop, right loop and arch classes of the input images and the features from the polygon are evaluated. Classification of fingerprints is done for each fingerprint as per the classification scheme discussed in chapter 4. The templates are formed from 2 sets of the corresponding fingerprints of each of the users. The averages of the corresponding features are taken and the 32 element vector template database is created. Table 7.1 shows the 32 element feature template of 2 sets of fingerprint of 15 candidates with their averaged feature values. Table 7.2 gives the final feature template of candidates stored in the database. About 300 fingerprint pairs were taken and the features were extracted and stored as template data. Another sample set of 300 fingerprint pairs of same persons were taken as test set.

7.3.4 Results and Discussion

Matching score has been calculated for each fingerprint in the test set with the template. Box plot, which is a statistical plot of the score distribution, is shown in fig.7.4. A genuine fingerprint is one which is supposed to match with the same fingerprint template in the template data. An imposter is one whose fingerprint does not match with the template data. The genuine fingerprint distribution shows a matching score range from 0.77 to 0.99 with a median value of 0.91, whereas the imposter fingerprint distribution match score ranges from 0.45 to 0.63 with a median value of 0.56. The whiskers of genuine distribution range in

between 0.72 and 1 and that of imposter distribution range between 0 and 0.71. Hence there is no overlap between the match score of genuine and imposter cases. Thus fixing a matching score threshold between 0.72 and 0.71 can identify all the fingerprints with 100% accuracy. The Receiver Operator Characteristics (ROC) graph is shown in fig.7.4. The False Acceptance Rate(FAR) is a measure of how many imposter users are falsely accepted into the system as "genuine" users is plotted by varying the threshold from 0 to 1. The False Rejection Rate (FRR) is a measure of how many genuine users are falsely rejected by the system as "imposters". It is also calculated for various thresholds and is plotted in fig. 7.4. The Equal Error Rate (EER) is defined as the condition at which $FAR=FRR$ as in fig.7.4, it is approximately equal to zero at a threshold 0.715. For this threshold, fingerprints are identified with 100% accuracy.

From chapter 4, fig.7.5 shows the ROC correspond to single finger based identification system. A higher matching threshold of 0.79 is needed to identify the persons with 100% accuracy. Hence in the multifinger feature level based fusion provides 100% recognition of persons at a lower threshold and also matching is efficient than the single finger based method as the types of the two fingers are matched first.

Table 7.2 Fingerprint Fusion Template Data

Finger Id	d_{cc}	d_{la}	d_{su}	d_{sw}	d_{sh}	d_{sl}	d_{td}	θ_{cc}	θ_{la}	θ_{su}	θ_{sw}	θ_{sh}	θ_{sl}	A_1	T_1	F_1	10^6	10^6	d_{sc}	d_{sb}	d_{sd}	d_{sw}	d_{sh}	d_{sl}	θ_{cc}	θ_{la}	θ_{su}	θ_{sw}	θ_{sh}	θ_{sl}	A_2	T_2	F_2	20^6	20^6
1	0	236	0	0	112.1	184.7	123.2	62.37	0	117.65	0	233.58	2	10.5	18.3	15.5	0	251.1	0	101.2	170	123.7	50.1	0	131.4	0	203.92	2	11.5	20.8	12.5				
2	0	224	0	0	88	144.2	112.3	44.98	0	134.94	0	167.28	2	11.215	13.5	0	231.6	0	43.1	172.9	72.83	36.4	0	143.6	0	872.2	2	4.5	21.8	16.8					
3	0	258	0	0	144.6	144.2	179.1	50.25	0	129.55	0	270.71	2	17.5	16.8	10.5	0	271.2	0	124.8	103.7	197.7	29.9	0	150.3	0	242.86	2	16	17.5	9.25				
4	0	259	186	122	123.9	0	0	42.71	137.1	0	0	240.21	1	13.5	15.3	8.75	0	297.7	182.7	129.6	71.56	0	0	23.1	157	0	153.89	1	15	20.5	8				
5	0	256	0	0	107.3	143.2	155.8	43.54	0	136.5	0	213.70	2	13.5	21.8	14.8	0	232.5	158.1	119.2	110.6	0	0	44.3	133.9	0	139.79	1	12	16.5	8.5				
6	0	238	0	0	120.2	187.3	52.3	13.33	0	166.66	0	256.7	2	3	15.5	14.8	0	247.4	102.7	170.8	68.13	0	0	36.5	143.3	0	142.52	1	9	18	12.3				
7	0	253	0	0	44.01	218.9	56.31	52.07	0	127.82	0	104.51	2	7	20	16.5	0	233.2	91.71	184.4	76.87	0	0	57.7	122.3	0	160.62	1	10.5	23.5	14.3				
8	0	312	0	0	151.5	131.5	253.4	19.79	0	160.27	0	339.57	2	17.5	18	6.25	153.9	123.7	212.8	109.1	106.1	0	0	170	141.2	0	151.2	201.21	5	11	14.5	16.5			
9	0	254	0	0	161.6	154.4	189.6	58.26	0	121.88	0	329.95	2	17.5	16	10.3	0	308.5	0	118	145	201.7	18.7	0	161.3	0	267.47	2	14	19	11.5				
10	86	200	0	0	205.4	157.1	185.8	140.4	0	126.25	93.5	445.64	4	18	17	9.5	0	259.2	0	122.1	99.08	201	37.3	0	142.8	0	218.50	2	15	19	8.5				
11	0	247	186	134	147.9	0	0	52.8	127	0	0	282.58	1	15	22.5	7	0	295.8	230.2	121.6	150.2	0	0	40.8	139.2	0	287.74	1	19	27.3	7.75				
12	0	272	0	0	100.7	136.7	168.4	38.68	0	151.39	0	205.48	2	15	22.3	10	84.48	270.9	109.9	178.1	104	169.3	273.6	152	136.7	107.4	106.7	175.55	3	14	13.5	18			
13	48	320	84.3	170	206.6	169	311	175.5	134.3	94.441	157	279.88	3	20.5	18.5	15.5	0	288.5	0	65.61	195.3	114.1	30.3	0	149.7	0	159.94	2	11.5	20.3	13.5				
14	0	226	50.1	179	17.7	0	0	21.22	158.7	0	0	339.2	1	2	17.5	14.8	0	282	128.1	194.7	93.36	0	0	41.4	138.7	0	223.96	1	12.5	19.5	13.5				
15	0	196	0	0	72.5	136.5	93.49	40.63	0	139.37	0	120.97	2	8	15.8	11.3	0	226.5	0	76.5	143.4	113.1	37	0	146.8	0	145.18	2	11	18.3	10.5				

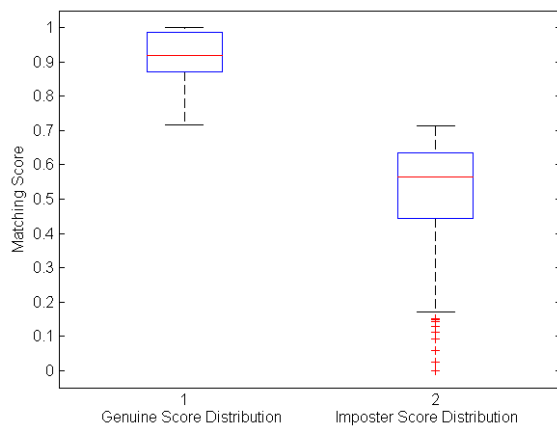


Fig. 7.4. Box plot of score distribution

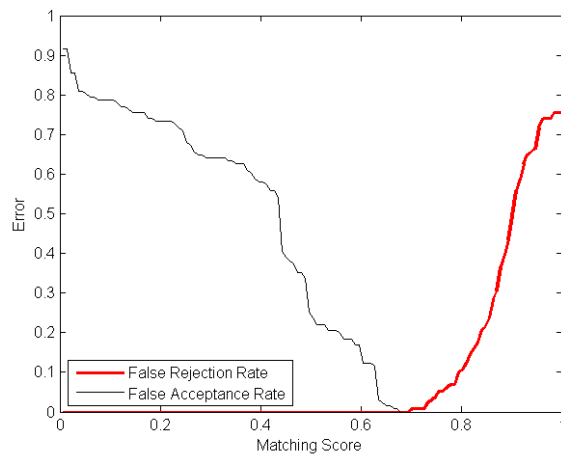


Fig.7.5 FAR-FRR Graph- Multifinger

7.4 Development of Multimodal fingerprint and speech Score level fusion based personal authentication using fingerprint and speech features

Multimodal features are those from different biometric modalities like fingerprint, speech, face, iris etc. In the earlier discussion, feature level fusion of multiple fingerprint features have been carried out. In this research, the effect of fusing the score levels by combining the classifiers has been studied and a system is developed to authenticate person using the physiological biometric trait, fingerprint, and the behavioral biometric trait, speech.

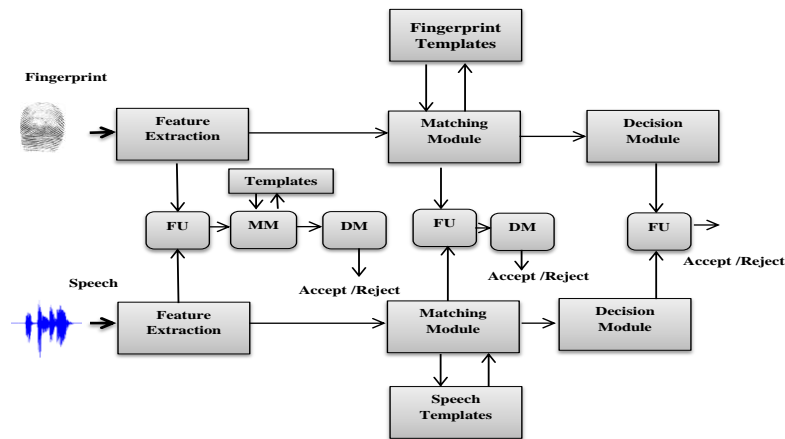


Fig. 7.6 Bimodal biometric system showing three levels of fusion.

Fig. 7.6 shows a bimodal biometric system showing the three levels of fusion: fusion at the feature extraction level, fusion at the matching score level and fusion at the decision level. In the figure, FU represents the fusion module, MM represents the matching module and DM represents the decision module. In the first level, feature level fusion is shown with the fusion module FU fuses the features. The matching module MM compares the feature level fused templates and

based on the score obtained, a decision is taken at decision module, DM. In second level of fusion, matching module level fusion is carried out at the fusion module using the majority voting, combination of ranked list, AND and OR operations. A decision is taken at the decision module for matching. In the last phase of fusion, score level fusion is carried out at the decision module, DM using weighted summation, weighted product and by a post classifier.

7.4.1 Implementation

7.4.1.1 Fingerprint verification

Fingerprint verification involves extracting a feature set from a fingerprint image acquired by means of a good quality optical based fingerprint scanner and matching it with its template stored in the data. The feature set consists of 16 polygonal feature metrics and the methods used for feature extraction were that described in section 4.3 of chapter 4. The matching process involves the computation of a similarity measure using the distance between corresponding polygonal features in the input and feature template. A score of 1 gives a 100% matching and a score of 0 gives no matching. The threshold for a match was fixed at 0.79 as per the discussion given in section 4.2.5 of chapter 4.

7.4.1.2 Text-dependent Speaker verification

In text dependent speaker verification, the speech samples of a known sentences or words were recorded using a good quality microphone in the laboratory environment. Feature extraction involves the computation of 13 MFC coefficients of the speech frames which were clustered to give 13×5 element matrix for each speeches. These coefficients were input to a trained backpropagation based artificial neural network for classification/identification. The neural network output ranges between 0 to 1 in which a 0 output indicates the absence of speaker template in database and a score of 1 indicates the presence of the corresponding template in the speaker database. Match threshold can be fixed between 0.152 and 0.167 as per the experiments described in section 6.4 of chapter 6. The speaker is identified by the output neuron number which is triggered to a high output.

7.4.1.3 Combining the two modalities

The database for combining the two modalities consisted of matching scores of two different modalities- fingerprint and speech. The fingerprint and speech data were obtained from a user set consisting of 40 users. For each user, 6 fingerprints were acquired and 10 voice prints were recorded. Out of the 6 fingerprints, 3 fingerprint image features were extracted and the features were averaged to form the feature templates. Other 3 images of 40 users were used to generate the match scores using the distance based match score generation as per equation 4.23. Hence a 120 genuine scores were obtained for fingerprint modality and 4680 (120×39) imposter scores were obtained. Similarly, 10 speech samples of 40 users were recorded, out of which 7 were used for ANN training and 3 were used for verifying the neural network. Here also 120 genuine scores were obtained for speech modality and 4680 imposter scores were obtained.

As the scores of the two classifiers ranges between 0 and 1, normalization of scores is not required in this fusion method.

7.4.1.4 Combining using Sum rule

In this combining method, weighted averages of the scores from the two modalities were taken. In this research the effect of weights on each modality combined together to give a score were investigated. Starting from an initial weight, $w=0.1$, the scores were calculated according to the sum rule:

$$M = m_f w + m_s (1 - w) \quad (7.6)$$

where M is the combined match score, m_f is the match score for the fingerprint based recognition for the database taken as explained in the previous section, m_s is the match score for the speech based identification system and w is the weight taken for fingerprint match score which ranges from 0.1 to 0.9. For each weight, the False Accept Rate and False Reject Rate are calculated for various matching threshold. The FAR and FRR were calculated for each weights and plotted as in fig.7.7. From the plot, equal error rates (EERs) are found out by taking the range of match threshold at which the error is zero. From the graph, for $w = 0.9$, equal error rate found out is 6.24×10^{-4} and equal error rate is 0 for $w < 0.9$. Table 7.3 gives the EERs corresponding to each of the above plots. The EER is zero for weights in the range 0.1-0.8 and for weights ≥ 0.9 the EER is greater than 0.

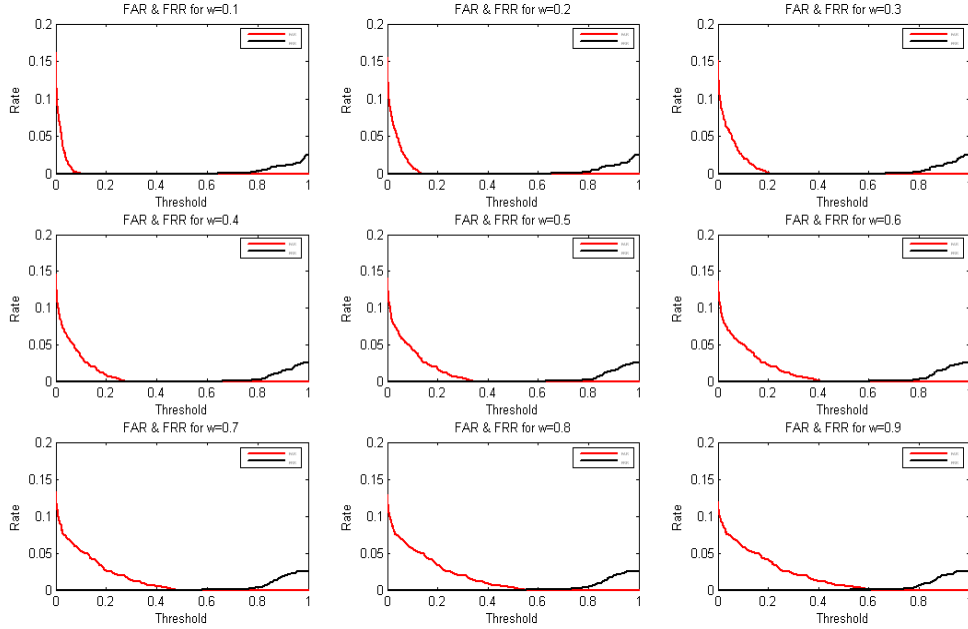


Fig. 7.7 FAR and FRR Plots correspond to various weights taken with sum rule

At lower weights, the threshold range difference is large. The discriminative power of neural network classifier is predominant here even at lower weights. Hence a lower weight $w=0.1$ for fingerprint score which yields 100% recognition with a threshold range between 0.1 and 0.639 is recommended for the system.

7.4.1.5 Combining using Product rule

The match scores were combined using the product rule as per the rule:

$$M = (m_f)^w \cdot (m_s)^{1-w} \quad (7.7)$$

where M is the fused match score, m_f is the match score for the fingerprint based recognition, m_s is the match score for the speech based identification system and w is the weighting factor for fingerprint match score which ranges from 0.1 to 0.9.

Table 7.3 Weights and Threshold range for EER

Weight	Threshold range for EER	Difference	EER
0.1	0.1-0.639	0.539	0
0.2	0.139-0.651	0.512	0
0.3	0.205-0.664	0.459	0
0.4	0.274-0.656	0.382	0
0.5	0.342-0.629	0.287	0
0.6	0.41-0.603	0.193	0
0.7	0.478-0.577	0.099	0
0.8	0.546-0.551	0.005	0
0.9	0.596-0.612	0.016	6.24×10^{-4}

Fig.7.8 shows the ROC curves obtained using product rule and Table 7.4 show the corresponding weights and threshold range for EER. From the graph the FAR and FRR falls to zero for a reasonable range of matching threshold. The EER is found to zero for all the weight values with maximum threshold difference of 0.331 for the weight $w=0.1$.

From the two experiments on fusion using sum and product rule, EER was shown to be zero for a considerable range of threshold using sum rule with weight, $w=0.1$. The discriminative power of neural network classifier was quite evident from the plot as even for lower weights the EER was shown to be zero. the product rule and by taking the weight, $w=0.1-0.4$ and by fixing the match score threshold around 0.4 a person can be effectively recognized as a genuine or imposter.

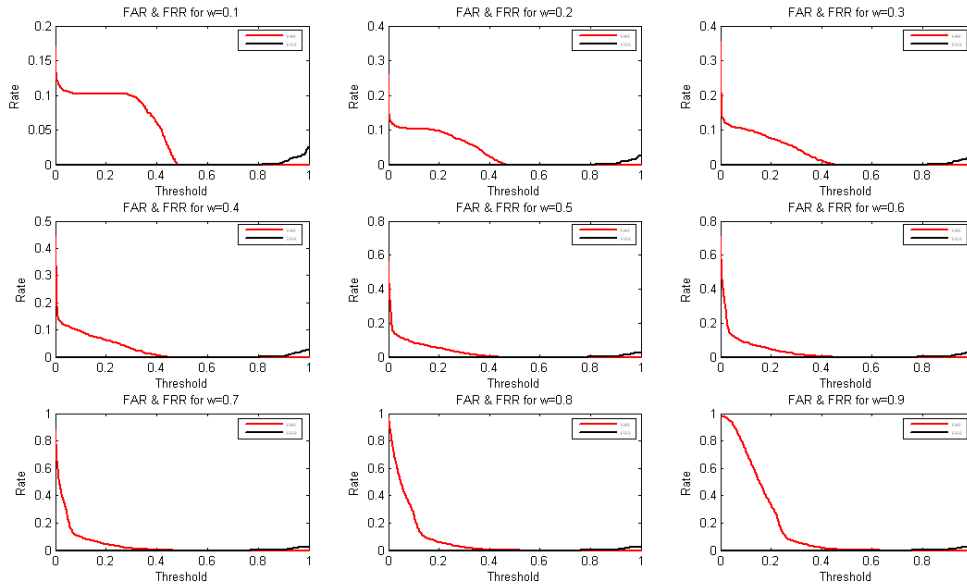


Fig. 7.8 FAR and FRR Plots correspond to various weights taken with product rule

Table 7.4 Weights and Threshold range for EER. EER=0 for all the shown weights

Weight	Threshold range for EER	Difference
0.1	0.484-0.815	0.331
0.2	0.468-0.816	0.348
0.3	0.454-0.818	0.364
0.4	0.444-0.803	0.399
0.5	0.439-0.789	0.350
0.6	0.444-0.777	0.333
0.7	0.467-0.768	0.301
0.8	0.521-0.760	0.239
0.9	0.631-0.754	0.123

7.5 Conclusion

In this chapter multimodal biometric fusion techniques were discussed. A global singularity based feature level fusion of fingerprints was developed and 100% recognition rate was achieved for a matching threshold of 0.715. Fusion of fingerprint and speech match score, which is one of the successful fusion techniques for personal authentication, was discussed using the sum and product rule. The system was able to classify the genuine and imposter with an EER of zero for a considerable matching threshold range of 0.539. The sum rule was found to be more effective for the score level fusion used in this system due to large span of matching threshold.

Chapter 8

Conclusion and Future Scope

8.1 Conclusion

The work presented in this thesis is a study on biometric personal authentication based on fingerprint and speech. In this chapter, a compilation of the work carried out for the development of biometric based personal authentication has been given, along with the future scope of the work.

Biometrics has the advantage that the fear of theft of identity of a person is eliminated. Unlike passwords and tokens, the identity of a person is himself or herself and may not be able to manipulate easily. In this work, a discussion on fingerprint based personal authentication techniques were explained in detail in chapter 3. Fingerprint based personal identification has been practiced since 19th century and the state-of-the-art minutiae based fingerprint identification was the most common identification technique used in all the applications. In chapter 4, novel global singularity based fingerprint identification has been developed. Singularities were identified using the directional field strength, coherence computed using squared gradients. The low coherence values were identified as the singularities using the coherence computed with a window size of $w=16$. The fingerprints were classified into seven types based on the relative positions of the singularities. Base line of the fingerprint which is defined as the first phalangeal joint was extracted using pre-defined masks and fingerprint polygons were formed

with the base line and the singular points. The global singularity feature points were defined as the polygonal convex angles, polygonal side lengths, ridge counts between the core and delta, delta and base, core and base, and the area of the polygon. The feature points were a 16 element length polygonal metrics. The feature points were stored in different feature database based on fingerprint types.

An Euclidean distance matching score was formulated. The False Accept Rate and False Reject Rate were computed using the probabilities measured with matching threshold. In this research, the FAR and FRR measured for the system was zero at a matching threshold of 0.79. At this threshold, fingerprints were identified with 100% accuracy. This authentication system is recommended for security based access control applications. To compare the performance of the system, conventional minutiae based fingerprint recognition system has been developed. The global singularity based system outperforms the minutiae based system in terms of feature extraction time and matching time.

In the next phase of the research, behavioral biometric modality based personal authentication system has been discussed. Speech, which is one of the simplest biometric modality, has also physiological characteristic dependence on vocal tract length. Speech production, speaker recognition and spectral characteristics were explained in chapter 5. In Chapter 6, development of MFCC based text dependent speaker recognition was discussed. The MFCCs were computed for 2 seconds duration speech. The MFCCs were clustered using k -means clustering technique to reduce the speech code book size. A backpropagation artificial neural network was designed with adaptive learning and the speech code book was trained to identify speakers. The system was tested with sufficient number of test speeches and the FAR and FRR were computed. The equal error rate (EER) measured was about 3.82% with a cluster size of 5. The system can identify speakers with an accuracy of 96.18% which was relatively good result compared with the VQ based Euclidean distance minimum classifier.

Recent biometric research advances towards the implementation of multibiometric based identification systems. Multibiometric based systems were discussed and developed in chapter 7. A Multifinger feature level fusion based personal identification system was developed first. The consecutive fingers were

captured and the feature template was a 32 element vector by the concatenation of the individual fingerprint features derived from global singularity features. The FAR and FRR were measured and the EER observed for the system was zero percent at a lower matching threshold of 0.72. Hence the system was able to identify persons with 100% accuracy at this matching threshold.

The final goal of this research was to develop a multibiometric personal authentication system based on bimodal traits. Fingerprints and speech modalities have been subjected to fusion at score level. The fingerprint matching score was computed using global singularity based identification system and MFCC based speech matching score was computed using the artificial neural network. The score level fusion was carried out using sum and product rule. The recognition rate in both cases was 100% with a matching threshold between 0.1 and 0.639 for sum based fusion and a matching threshold between 0.454 and 0.818 for the product rule based score level fusion. The sum rule was found to be more effective for score level fusion.

To sum up, in this research the objectives set were completely carried out with 100% success.

8.2 Future Scope

High security access control areas require accurate personal authentication systems. Multibiometric systems offer 100% accuracy and also are not vulnerable to mimic and attack by imposters. As the database of the genuine users increase, accuracy declines. This may be overcome by combining more modalities under different levels of fusion. The key issue in any of the personal authentication system is the matching time and feature extraction time. Hence efficient algorithms with multimodalities are still a challenging issue in biometric based personal authentication systems.

References

- Abdeljaoued, 1999; Y. Abdeljaoued, Fusion of person authentication probabilities by Bayesian statistics, In: Proc. 2nd Int. Conf. Audio and Video-based Biometric Person Authentication, Washington D.C., 1999, pp. 172- 175.
- Abutaleb, 1999; A. S. Abutaleb and M. Kamel, A Genetic Algorithm for the Estimation of Ridges in Fingerprints , IEEE Transactions on Image Processing, vol. 8, no. 8, pp. 1134-1138, 1999.
- Achermann, 1996; B. Achermann, H. Bunke, Combination of classifiers on the decision level for face recognition, Technical report IAM-96-002, Institut für Informatik und angewandte Mathematik, Universität Bern, 1996.
- Adjoudani, 1995; A. Adjoudani, C. Benoît, Audio-visual speech recognition compared across two architectures, In: Proc. 4th European Conf. Speech Communication and Technology, Madrid, Spain, 1995, Vol. 2, pp. 1563-1567.
- Ahmed, 2002; M. Ahmed and R. Ward, A Rotation Invariant Rule based Thinning Algorithm for Character Recognition , IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, no. 12, pp. 1672-1678, 2002.
- Akram, 2008; Akram, M. Usman, et al. Improved fingerprint image segmentation using new modified gradient based technique. Electrical and Computer Engineering, 2008. CCECE 2008. Canadian Conference on. IEEE, 2008.
- Alexandre, 2001; L.A. Alexandre, A.C. Campilho, M. Kamel, On combining classifiers using sum and product rules, Pattern Recogn. Lett. 22, 2001, 1283–1289.

- Altıçay, 2000; H. Altıçay, M. Demirekler, An information theoretic framework for weight estimation in the combination of probabilistic classifiers for speaker identification, *Speech Commun.* 30, 2000, 255–272.
- Arcelli, 1984; Arcelli C. and Baja G.S.D., A width independent fast thinning algorithm, *IEEE Transactions on Pattern Analysis Machine Intelligence*, vol. 4, no. 7, pp. 463–474, 1984.
- Ballard, 1981; Ballard D.H., “Generalizing the Hough transform to detect arbitrary shapes,” *Pattern Recognition*, vol. 13, no. 2, pp. 110–122, 1981.
- Basheer, 2000; Basheer, I A, and M Hajmeer. *Artificial Neural Networks : Fundamentals , Computing , Design , and Application.* *Journal of Microbiological Methods*, 43, 2000, 3–31.
- Baum, 1989; Baum, E., Haussler, D., What size net gives valid generalization? *Neural Computation* 1, 151–160, 1989.
- Bazen, 2001; A. M. Bazen and S. H. Gerez, Segmentation of Fingerprint Images , *Proceedings of Workshop on Circuits, Systems and Signal Processing*, Veldhoven, Netherlands, pp. 276- 280, Nov., 2001.
- Bazen, 2002; A.M. Bazen, S.H. Gerez, Systematic methods for the computation of the directional fields and singular points of fingerprints, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24 (7) (2002) 905–919.
- Ben, 1999; S. Ben-Yacoub, Y. Abdeljaoued, E. Mayoraz, Fusion of face and speech data for person identity verification, *IEEE Trans. on Neural Networks* 10 (1999) 1065-1074.
- Berry, 1991; Berry J., The history and development of fingerprint, *Advances in Fingerprinting Technology*, H. C. Lee and R. E. Gaensslen, eds., NY: Elsevier, 1991, pp1-38.
- Berry, 2001; Berry and Stoney (2001). Berry J. and Stoney D.A., The history and development of fingerprinting, in *Advances in Fingerprint Technology*, 2nd edition, H.C. Lee and R. Gaensslen (Eds.), pp. 1–40, CRC Press, Boca Raton, FL, 2001.

- Bhowmick, 2004; P. Bhowmick and B. B. Bhattacharya, "Approximate Fingerprint Matching using Kd Tree", Proceedings of the IEEE International Conference on Pattern Recognition, Cambridge, United Kingdom, pp. 544-547, 2004.
- Bishnu, 2006; Bishnu A., Das S., Nandy S.C. and Bhattacharya B.B., "Simple algorithms for partial point set pattern matching under rigid motion," Pattern Recognition, vol. 39, no. 9, pp. 1662–1671, 2006.
- Bishop, 1996; Bishop, C. Neural Networks for Pattern Recognition. Oxford University Press, New York, 1996.
- Brass, 2002; Brass P. and Knauer C., "Testing the congruence of d-dimensional point sets," International Journal of Computational Geometry and Applications, vol. 12, pp. 115–124, 2002.
- Brunelli, 1995; R. Brunelli, D. Falavigna, Person identification using multiple cues, IEEE Trans. Pattern Anal. Machine Intell. 10, 1995, 955–965.
- Campbell, 1997; J Campbell, Speaker Recognition: A Tutorial, Proceedings of IEEE, Vol. 85, no.9, pp 1437-1462, Sept. 1997.
- Campbell, 2003; J. Campbell, D. Reynolds and R. Dunn. Fusing high- and low level features for speaker recognition. In Proc. Eurospeech, 2003.
- Candela, 1995; 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.
- Cappelli, 1999; Cappelli R., Maio D. and Maltoni D., Fingerprint Classification Based on Multi-space KL, in Proc. Workshop on Automatic Identification Advances Technologies, pp. 117–120, 1999.
- Cappelli, 2004; R. Cappelli and D. Maio, The state of the Art in Fingerprint Classification, Automatic Fingerprint Recognition Systems, N. Ratha and R. Bolle, eds., Springer, 2004.
- Cappelli, 2009; RaffaeleCappelli and DavideMaltoni, On the Spatial Distribution of Fingerprint Singularities, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol 31, No. 4, April 2009 pp- 742-748

- Cavusoglu, 2007; A. Cavusoglu and S. Gorgunoglu, A Robust Correlation based Fingerprint Matching Algorithm for Verification, *Journal of Applied Sciences*, vol. 7, no. 21, pp. 3286-3291, 2007.
- Chatterjee, 1988; Chatterjee S. K. and R. V. Hague, *Fingerprint or Dactyloscopy and Ridgeoscopy*, Calcutta, Srijib Chatterjee, 1988.
- Chen, 2004; X. J. Chen, J. Tian, J. G. Cheng, X. Yang, Segmentation of Fingerprint Images using Linear Classifier, *EURASIP Journal on Applied Signal Processing*, pp. 480-494, 2004.
- Chen, 2004; Xinjian Chen, Jie Tian, Jiangang Cheng, Xin Yang, Segmentation of Fingerprint Images Using Linear Classifier, *EURASIP Journal on Applied Signal Processing* 2004:4, 480–494.
- Chikkerur 2005; Chikkerur S., Govindaraju V., Pankanti S., Bolle R. and Ratha N., Novel Approaches for Minutiae Verification in Fingerprint Images, in *Proc. Workshops on Application of Computer Vision*, vol. 1, pp. 111–116, 2005.
- Chikkerur, 2007; S. Chikkerur, A. N. Cartright and V. Govindaraju, Fingerprint Enhancement using STFT Analysis, *Pattern Recognition*, vol. 40, no. 1, pp. 198-211, 2007.
- Coetzee, 1993; Coetzee L. and Botha E.C., Fingerprint recognition in low quality images, *Pattern Recognition*, vol. 26, no. 10, pp. 1441–1460, 1993.
- Daugman, 2004; J. Daugman. How Iris Recognition Works? *IEEE Transactions on Circuits and Systems for Video Technology*, 14(1):21–30, 2004.
- Deller, 1993; J. R. Deller. I. G. Proakis, J. H. L. Hansen, *Discrete-Time Processing of Speech Signals*, New York: Macmillan, 1993.
- Dieckmann, 1997; U. Dieckmann, P. Plankensteiner, T. Wagner, SESAM: A biometric person identification system using sensor fusion, *Pattern Recognition Letters* 18 (1997) 827-833.
- Dowla and RoDowla, F.U., Rogers, L.L., 1995. *Solving Problems in Environmental Engineering and Geosciences With Artificial Neural Networks*. MIT Press, Cambridge, MA

- Drets, 1999; G. A. Drets and G. Liljenström, Fingerprint Sub classification: A Neural Network Fingerprint Classification, Intelligent Biometric Techniques in Fingerprint and Face Recognition, L C. Jain, U. Halici, I. Hayashi,, S. B. Lee, and S. Tsutsui, eds., pp. 109-134, Boca Raton, Fla.: CRC Press, 1999.
- Dretsand, 1999; G. A. Dretsand G. G. Liljenström, Fingerprint Sub classification: A Neural Network Fingerprint Classification, Intelligent Biometric Techniques in Fingerprint and Face Recognition, L C. Jain, U. Halici, I. Hayashi,, S. B. Lee, and S. Tsutsui, eds., pp. 109-134, Boca Raton, Fla.: CRC Press, 1999.
- Duda, 2010; R.O. Duda, P.E. Hart, D.G. Stork, Pattern Classification. Wiley India, 2010.
- Egan, 1975; J. Egan. Signal Detection Theory and ROC Analysis. Academic Press, New York, 1975.
- Emiroglu, 1997; I. Emiroglu and M. B. A. Khan, Preprocessing of Fingerprint Images , Proceedings of European Conference on Security and Detection, London, UK, pp. 147-151, April, 1997.
- Everitt, 1993; EVERITT, B. S., Cluster Analysis. Edward Arnold, Ltd., London, UK., 1993.
- Faulds, 1880; Faulds H, On the skin furrows of the hand, *Nature*, 22(574): 605, 1880.
- Feldtkeller , 1956; Feldtkeller, R., & Zwicker, E. Das Ohr als chrichtenempfänger. Stuttgart: S. Hi r zel, 1956.
- Feng, 2006; J. Feng, Z. Ouyang and A. Cai, Fingerprint Matching using Ridges, *Pattern Recognition*, vol. 39, no. 11, pp. 2131-2140, 2006.
- Feng, 2008; Feng J., Combining minutiae descriptors for fingerprint matching, *Pattern Recognition*, vol. 41, no. 1, pp. 342–352, 2008.
- Fu, 1995, Fu, L., Neural Networks in Computer Intelligence. McGraw-Hill, New York, 1995.

- Furui, 1981; S. Furui. Cepstral analysis technique for automatic speaker verification. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 29:254–272, 1981.
- Furui, 1997; Furui, S. “Recent Advances in Speaker Recognition”, In Springer, editor, *Audio- and Video-based Biometric Person Authentication*, Crans-Montana, Switzerland, March 12-14, pp. 237-251. 1997.
- Galton, 1892; Galton F, *Fingerprints* McMillan, London, 1892.
- Gorman, 1988 O’Gorman L. and Nickerson J., Matched Filter Design for Fingerprint Image Enhancement, in *Proc. Int. Conf. on Acoustic Speech and Signal Processing*, pp. 916–919, 1988.
- Gorman, 1989; L. O. Gorman and J. V. Nicherson, An Approach to Fingerprint Filter Design, *Pattern Recognition*, Vol. 22, No. 1, pp.-29-38, 1989
- Grasselli, 1969; Grasselli A., On the automatic classification of fingerprints, in *Methodologies of Pattern Recognition*, S. Watanabe (Ed.), Academic, New York, 1969.
- Greenberg, 2000; Greenberg S., Aladjem M., Kogan D. and Dimitrov I., Fingerprint Image Enhancement Using Filtering Techniques, in *Proc. Int. Conf. on Pattern Recognition (15th)*, vol. 3, pp. 326–329, 2000.
- Grother, 2004; P. Grother and P. J. Phillips. Models of Large Population Recognition Performance. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 68–75, Washington D.C., USA, June/July 2004.
- Hall, 2001; D.L. Hall, J. Llinas, Multisensor data fusion, in: D. L. Hall and J. Llinas (Eds.), *Handbook of Multisensor Data Fusion*, CRC Press, USA, 2001, pp. 1-1 - 1-10.
- Harrison, 1981; W. R. Harrison. *Suspect Documents, their Scientific Examination*. Nelson-Hall Publishers, 1981.
- Hassoun, 1995; Hassoun, M.H., *Fundamentals of Artificial Neural Networks*. MIT Press, Cambridge, MA, 1995.

- Haykin, 1994; Haykin, S., *Neural Networks: A Comprehensive Foundation*. Macmillan, New York, 1994.
- Hebb, 1949; Hebb, D.O., *The Organization of Behavior*. Wiley, New York, 1949.
- Hecht, 1988; Hecht-Nielsen, R.,. Applications of counterpropagation networks. *Neural Networks* 1, 131–139, 1988
- Hecht, 1990; Hecht-Nielsen, R., *Neurocomputing*. Addison-Wesley, Reading, MA, 1990.
- Henry, 1900; Henry E., *Classification and Uses of Finger Prints*, Routledge, London, 1900.
- Herschel, 1880; W. J. Herschel, Skin Furrows of the Hand , *Nature*, vol. 23, pp. 76-76, 1880.
- Herschel, 1916; Herschel, W J., *The Origin of Finger-Printing*, London: Oxford University Press, 1916.
- Ho, 1994; T.K. Ho, J.J. Hull, S.N. Srihari, Decision combination in multiple classifier systems, *IEEE Trans. Pattern Anal. Machine Intell.* 16, 1994, 66–75.
- Hong, 1998; Hong L, Wan Y, Jain , A Fingerprint image enhancement: algorithm and performance evaluation. *IEEE Trans Pattern Anal Mach Intell* 20(8):777–789, 1998.
- Hong, 1998; L. Hong, A. Jain, Integrating faces and fingerprints for personal identification, *IEEE Trans. Pattern Anal. Machine Intelligence*, 20, 1998, 1295–1306.
- Hong, 1998; Lin Hong, Yifei Wan, and Anil K. Jain. Fingerprint image enhancement: Algorithm and performance algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 8, pp. 777–789, May 1998.
- Horvath, 1967; Horvath et al, Holographic Technique recognizes Fingerprint, *Laser Focus*, June, 1967, pp 18-23.

- Hu, 2008; C. Hu, J. Yin, E. Zhu, H. Chen and Y. Li, Fingerprint Alignment using Special Ridges, Proceedings of 19th IEEE international conference on Pattern Recognition, Tampa, FL, pp. 1- 4, 2008.
- Huang. 2007. Huang C.Y., Liu L.M. and Hung D.C.D., Fingerprint analysis and singular point detection, Pattern Recognition Letters, vol. 28, no. 15, pp. 1937–1945, 2007.
- Hung, 1993; Hung D.C.D., Enhancement and feature purification of fingerprint images, Pattern Recognition, vol. 26, no. 11, 1661–1671, 1993.
- Hung, 1996; 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.
- Huvanandana, 2003; S. Huvanandana, S. Malisuwan and J. N. Hwang, A Hybrid System for Automatic Fingerprint Identification, Proceedings of the IEEE International Symposium on Circuits and Systems, Bangkok, Thailand, vol. 2, pp. 952-955, 2003.
- Itakura, 1975; Itakura F, Minimum prediction residual principle applied to speech recognition, IEEE Tran. on ASSp, 23, 1975.
- Iyengar, 1995; S.S. Iyengar, L. Prasad, H. Min, Advances in Distributed Sensor Technology, Prentice Hall PTR, New Jersey, 1995.
- Jain , 2004; A.K. Jain, A. Ross, Multibiometric systems, Commun. ACM 47 (1) (2004) 34–40 (special issue on multimodal interfaces)
- Jain, 1996; Jain, A.K., Mao, J., Mohiuddin, K.M., Artificial neural networks: a tutorial. Comput. IEEE March, 31–44, 1996.
- Jain, 1997; A. K Jain, L. Hong, S. Pankanti, and R. Bolle, ‘An Identity Authentication System Using Fingerprints, Proc. IEEE, Vol. 85, no. 9, pp1365-1388, Sept. 1997.
- Jain, 1999; A. K. Jain, L. Hong and Y. Kulkarni, A Multimodal Biometrics System using Fingerprint, Face and Speech, Proceedings of 2nd International Conference on Audio-and Video-based Biometric Person Authentication, Washington D. C., U. S. A., pp. 182-187, March, 1999.

- Jain, 1999; A. K. Jain, R. Bolle, and S. Pankanti, editors. *Biometrics: Personal Identification in Networked Society*. Kluwer Academic Publishers, 1999.
- Jain, 2000; A.K. Jain, S. Prabhakar and S. Pankanti, Filterbank-based fingerprint matching. *IEEE Trans. Image Process.* 9 5 (2000), pp. 846–859.
- Jain, 2001; A. K. Jain, A. Ross and S. Prabhakar, Fingerprint Matching using Minutiae and Texture Features, *Proceedings of the IEEE International Conference on Image Processing*, Florianopolis, Brazil, pp. 287-289, 2001.
- Jain, 2001; Jain A.K., Pankanti S., Prabhakar S. and Ross A., Recent Advances in Fingerprint Verification, in *Proc. Int. Conf. on Audio- and Video-Based Biometric Person Authentication (3rd)*, pp. 182–191, 2001.
- Jain, 2005; Score normalization in multimodal biometric systems Jain, Anil, Karthik Nandakumar, and Arun Ross. *Score Normalization in Multimodal Biometric Systems*, 38, 2270–2285, 2003.
- Jain, 2008; Anil K Jain, Arun Ross, *Introduction to Biometrics*. In Anil K Jain, Patrick Flynn, Arun A Ross, Editors, *Handbook of Biometrics*, pp 1-22, Springer, 2008.
- Jang, 1992; B. K Jang and R. T. Chin, One Pass Parallel Thinning: Analysis, Properties and Quantitative Evaluation , *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 11, pp. 1129-1140, 1992.
- Jang, 2006; W. Jang, D. Park, D. Lee and S. Kim, Fingerprint Image Enhancement Based on Half Gabor Filter , *International Conference on Biometrics*, Hong Kong, pp. 258-264, 2006.
- Jea 2005; Jea T.Y. and Govindaraju V., A minutia-based partial fingerprint recognition system, *Pattern Recognition*, vol. 38, no. 10, pp. 1672–1684, 2005.
- Jelinek, 1998: F. Jelinek, *Statistical Methods for Speech Recognition*, The MIT Press, Cambridge, MA, 1998.
- Jia, 2004; C. Jia, M. Xei and Q. Li, A Fingerprint Minutiae Matching Approach based on Vector Triangle and Ridge Structure, *Proceedings of IEEE*

- International Conference on Communications, Circuits and Systems, Chendu, China, vol. 2, pp. 871-875, 2004.
- Jia, 2007; Jia J., Cai L., Lu P. and Lu X., Fingerprint matching based on weighting method and the SVM, *Neurocomputing*, vol. 70, no. 4–6, pp. 849–858, 2007b.
- Jiang, 2001; X. Jiang, W. Y. Yau and W. Ser, Detecting the Fingerprint Minutiae by Adaptive Tracking the Gray Level Ridge, *Pattern Recognition*, vol. 34, no. 5, pp. 999-1013, 2001.
- Jin, 2008; Jin Qi, Mei Xie, Segmentation of fingerprint images using the gradient vector field, *Cybernetics and Intelligent Systems*, 2008 IEEE Conference on , vol., no., pp.543-545, 21-24 Sept. 2008.
- Jourlin, 1997a; P. Jourlin, J. Luetin, D. Genoud, H. Wassner, Acoustic-labial speaker verification, *Pattern Recognition Letters* 18 (1997) 853-858.
- Jourlin, 1997b; P. Jourlin, J. Luetin, D. Genoud, H. Wassner, Integrating acoustic and labial information for speaker identification and verification, In: *Proc. 5th European Conf. Speech Communication and Technology*, Rhodes, Greece, 1997, Vol. 3, pp. 1603-1606.
- Kasaei, 1997; S. Kasaei, M. D., and Boashash, B. Fingerprint feature extraction using block-direction on reconstructed images. In *IEEE region TEN Conf., digital signal Processing applications, TENCON* (December 1997), pp. 303–306.
- Kass, 1987; M. Kass and A. Witkin, Analyzing Oriented Patterns, *Computer Vision, Graphics, and Image Processing*, Vol. 37, No. 3, pp.-362-385, March 1987.
- Kerber, 1992; Kerber, R., ChiMerge: discretization of numeric attributes. In: *AAAI-92, Proceedings of the 9th National Conference on AI*. MIT Press, Cambridge, MA, pp. 123–128, 1992.
- Keun, 2000; Keun S, Lee J, Park C, Kim B, Park K, New fingerprint image enhancement using directional filter bank. School of Electrical Engineering Kyungpook National University, SEOUL 702-701, Daegu, Korea, 2000.

- Khan, 2005; Khan M, Khan K, Khan A, Fingerprint image enhancement using decimation- free directional filter bank. Department of Electrical Engineering. Inform Technol J, Islamabad, Pakistan 4(1):16–20, 2005.
- Kim, 2002; Kim B.G., Kim H.J. and Park D.J., New Enhancement Algorithm for Fingerprint Images, in Proc. Int. Conf. on Pattern Recognition (16th), vol. 3, pp. 879–882, 2002.
- Kinnunen, 2005; Tomi Kinnunen and Pasi Franti. Speaker Discriminative Weighting Method for VQ-based Speaker identification, Macmillan Publishing Company, New York, 2005
- Kittler, 1997; J. Kittler, J.Matas, K. Johnsson, M. U. Ramos-S´anchez, Combining evidence in personal identity verification systems, Pattern Recognition Letters 18 (1997) 845-852.
- Kittler, 1998; J. Kittler, M. Hatef, R.P.W.Duin, J.Matas, On combining classifiers, IEEE Trans. Pattern Analysis and Machine Intelligence 20 (1998) 226-239.
- L, 1993; Li, G., Alnuweiri, H., Wu, W., Acceleration of backpropagation through initial weight pre-training with Delta rule. In: Proceedings of an International Joint Conference on Neural Networks, San Francisco, CA, pp. 580–585 1993.
- Lacht, 1995; Lachtermacher, G., Fuller, J.D., Backpropagation in time-series forecasting. J. Forecasting 14, 381–393, 1995.
- Landy, 1984; Landy M.S., Cohen Y. and Sperling G., Hips: A Unix-based image processing system, Computer Vision, Graphics and Image Processing, vol. 25, no. 3, pp. 331–347, 1984.
- Laufer, 1912; Laufer B, *History of Finger-Print System*, Washington: Government Printing Office, 1912.
- Lee and Gaensslen, 2001; Lee H.C. and Gaensslen R.E., *Advances in Fingerprint Technology*, 2nd edition, Elsevier, New York, 2001.
- Leung, 1990; M. Leung, W. Engeler and P. Frank, Fingerprint Image Processing using Neural Network, Proceedings of the IEEE Region 10th Conference on

- Computer and Communication Systems, Hong Kong, pp. 582-586, September, 1990.
- Leuttin, 1997; J. Luettin, Visual Speech and Speaker Recognition, PhD Thesis, Department of Computer Science, University of Sheffield, 1997.
- Looney, 1996; Looney, C.G., Advances in feedforward neural networks: demystifying knowledge acquiring black boxes. *IEEE Trans. Knowledge Data Eng.* 8 (2), 211–226, 1996.
- Lumini, 2008; Lumini A. and Nanni L., Advanced methods for two-class pattern recognition problem formulation for minutiae-based fingerprint verification, *Pattern Recognition Letters*, vol. 29, no. 2, pp. 142–148, 2008.
- Luo, 1995; R.C. Luo, M.G. Kay, Introduction, in: R.C. Luo, M.G. Kay (Eds.), *Multisensor Integration and Fusion for Intelligent Machines and Systems*, Ablex Publishing Corporation, Norwood, NJ, 1995, pp. 1–26.
- Luo, 2000; X. Luo, and J. Tian, Knowledge Based Fingerprint Image Enhancement, *Proceedings of 15th International Conference on Pattern Recognition*, Barcelona, Spain, vol. 4, pp. 783- 786, Aug., 2000.
- Luping, 2007; J. Luping, Y. Zhang, S. Lifeng and P. Xiaorong, Binary Fingerprint Image Thinning using Template-based PCNNs, *IEEE Transactions on Systems, Man and Cybernetics*, vol. 37, no. 5, pp. 1407-1413, 2007.
- Maio 1999; Maio D. and Maltoni D., A Secure Protocol for Electronic Commerce Based on Fingerprints and Encryption, in *Proc. World Conf. on Systems Cybernetics and Informatics*, vol. 4, pp. 519–525, 1999.
- Maio, 1997; D. Maio and D. Maltoni, Direct Gray-Scale Minutiae Detection in Fingerprints, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.19, no.1, pp. 27-40, 1997
- Maio, 1998; Maio D. and Maltoni D., Neural Network Based Minutiae Filtering in Fingerprints, in *Proc. Int. Conf. on Pattern Recognition (14th)*, pp. 1654–1658, 1998b.

- Maltoni, 2005; D. Maltoni, D. Maio, A. K. Jain, and S. Prabhakar. Handbook of Fingerprint Recognition. Springer-Verlag, 2005
- Marana, 2005; A N Marana and A K Jain, Ridge-Based Fingerprint Matching using Hough Transform, Proceedings of the IEEE Brazilliab Symposium on Computer Graphica and Image Processing, pp. 112- 119, 2005.
- Martin, 1997; A. Martin, G. Doddington, T. Kam, M. Ordowski, and M. Przybocki. The DET Curve in Assessment of Detection Task Performance. In Proceedings of the Fifth European Conference on Speech Communication and Technology, volume 4, pages 1895–1898, Rhodes, Greece, September 1997.
- Masahiro Kawagoe and Akio Tojo, Fingerprint pattern classification, Pattern Recognition Volume 17, Issue 3, 1984, Pages 295-303 .
- Masters, 1994; Masters, T., Practical Neural Network Recipes in C11. Academic Press, Boston, MA. 1994.
- McCulloh,, 1943; McCulloh, W.S., Pitts, W., A logical calculus of the ideas immanent in nervous activity. Bull. Math. Biophys. 5, 115– 133, 1943.
- Mehetre, 1987; B.M. Mehtre, N.N. Murthy, S. Kapoor and B. Chatterjee, Segmentation of Fingerprint Images using the Directional Image , Pattern Recognition, vol. 20, no. 4, pp. 429-435, 1987.
- Mehetre, 1993. Mehtre B.M., Fingerprint image analysis for automatic identification, Machine Vision and Applications, vol. 6, no. 2–3, pp. 124–139, 1993.
- Mermelstein, 1980; Davis, S. and Mermelstein, P. Comparison of parametric representation for monosyllabic word recognition in continuous spoken sentence, IEEE Trans. Acoustic and Speech Signal Processing 28, 357–366, 1980.
- Moayer, 1986; B. Moayer and K. Fu, A Tree System Approach for Fingerprint Pattern Recognition, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 8, no. 3, 376-388, 1986.

- Monrose, 1997; F. Monrose and A. Rubin. Authentication Via Keystroke Dynamics. In Proceedings of Fourth ACM Conference on Computer and Communications Security, pages 48–56, Zurich, Switzerland, April 1997.
- Murty, 1992; Murty K.G., Network programming, Prentice-Hall, Englewood Cliffs, NJ, 1992.
- Nandakumar, 2004; K. Nandakumar and A. K. Jain, Local Correlation-based Fingerprint Matching, Proceedings of Indian Conference on Computer Vision, Graphics & Image Processing, pp. 1-6, Kolkata, December, 2004.
- Neil, 2004; Neil, Yager, N., & Amin, A. (2004). Fingerprint verification based on minutiae features: a review. *Pattern Analysis & Applications*, 7(1), 94–113.
- Nelson, 1990; Nelson, M., Illingworth, W.T., A Practical Guide To Neural Nets. Addison-Wesley, Reading, MA, 1990.
- Nilsson, 2001; K. Nilsson and J. Bigun, Using Linear Symmetry Features as a Preprocessing Step for Fingerprint Images, Proceedings of the 3rd International Conference on Audio and Video based Biometric Person Authentication, Halmstad, Sweden, pp. 247-252, June, 2001.
- Nilsson, 2002a. Nilsson K. and Bigun J., Complex Filters Applied to Fingerprint Images Detecting Prominent Points Used Alignment, in Proc. Workshop on Biometric Authentication (in ECCV 2002), LNCS 2359, Springer, pp. 39–47, 2002a.
- Nilsson, 2002b. Nilsson K. and Bigun J., Prominent Symmetry Points as Landmarks in Fingerprint Images for Alignment, in Proc. Int. Conf. on Pattern Recognition (16th), vol. 3, pp. 395–398, 2002b.
- Nilsson, 2003. Nilsson K. and Bigun J., Localization of corresponding points in fingerprints by complex filtering, *Pattern Recognition Letters*, vol. 24, no. 13, pp. 2135–2144, 2003.
- Novikov, 1998; Novikov S.O. and Kot V.S., Singular Feature Detection and Classification of Fingerprints Using Hough Transform, in Proc. of SPIE (Int. Workshop on Digital Image Processing and Computer Graphics (6th):

- Applications in Humanities and Natural Sciences), vol. 3346, pp. 259–269, 1998.
- Perona, 1998; P. Perona, Orientation Diffusions, *IEEE Trans. Image Processing*, vol.7, no.3, pp.457-467, Mar.1998.
- Pham, 1994; Pham, D.T., Neural networks in engineering. In: Rzevski, G. et al. (Eds.), *Applications of Artificial Intelligence in Engineering IX, AIENG/94, Proceedings of the 9th International Conference*. Computational Mechanics Publications, Southampton, pp. 3–36, 1994.
- Philips, 2003; P. J. Phillips, P. Grother, R. J. Micheals, D. M. Blackburn, E. Tabassi, and J. M. Bone. *FRVT2002: Overview and Summary*. Available at <http://www.frvt.org/FRVT2002>, March 2003.
- Poh, 2005; Poh, N.; Bengio, S.; , F-ratio Client-Dependent Normalisation for Biometric Authentication Tasks, *Acoustics, Speech, and Signal Processing, 2005. Proceedings. (ICASSP '05). IEEE International Conference on* , vol.1, no., pp. 721- 724, March 18-23, 2005
- Pols, 1977; Pols, L. C. W. Spectral analysis and identification of Dutch vowels in monosyllabic words. Unpublished doctoral dissertation, Free University, Amsterdam, 1977.
- Prabhakar, 2003; Prabhakar S., Jain A.K. and Pankanti S., Learning fingerprint minutiae location and type, *Pattern Recognition*, vol. 36, no. 8, pp. 1847–1857, 2003.
- Quatieri, 2012; Thomas F. Quatieri, *Discrete Time Signal Processing Principles and Practice*, Pearson Education Inc. India.
- Rabiner, 1989: L. R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77:257–286, 1989.
- Radová, 1997; V. Radová, J. Psutka, An approach to speaker identification using multiple classifiers, in: *Proc. IEEE Conf. Acoustics, Speech and Signal Processing, Munich*, vol. 2, 1997, pp. 1135–1138.

- Ranade, 1993; Ranade A. and Rosenfeld A., Point pattern matching by relaxation, *Pattern Recognition*, vol. 12, no. 4 pp. 269–275, 1993.
- Rao, 1992; A. R. Rao and R. C. Jain, Computerized Flow Field Analysis: Oriented Texture Fields, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 14, no. 7., pp. 693-709, July, 1992.
- Ratha, 1995; N. K. Ratha, S. Y. Chen, and A. K. Jain, Adaptive Flow Orientation-based Feature Extraction in Fingerprint Images, *Pattern Recognition*, vol. 28, no. 11, pp. 1657-1672, 1995.
- Reynolds, 1995; D A Reynolds, Speaker Identification Using Gaussian Mixture Speaker Model, *Speech Communication*, vol.17, pp 91-108, Aug. 1995.
- Reynolds, 2002; Reynolds, D. A., An Overview of Automatic Speaker Recognition Technology. In *Proc. International Conference on Acoustics, Speech, and Signal Processing in Orlando, FL, IEEE*, pp. IV: 4072-4075, 13-17 May 2002.
- Rhodes, 1956; Rhodes H. T. F., *Alphonse Bertillon: Father of Scientific Detection*, Abelard-Schuman, New York, 1956.
- Rich and Knight, 1994; Elaine Rich, Kevin Knight, *Artificial Intelligence*, Tata McGraw Hill, 1994.
- Rose P, 2002; *Forensic Speaker Identification*, T & F, London, 2002.
- Rosenblatt, 1962, Rosenblatt, R., *Principles of Neurodynamics*. Spartan Books, New York, 1962.
- Rosenfeld, 1976; Rosenfeld A. and Kak A., *Digital Picture Processing*, Academic, New York, 1976.
- Ross, 2003; A Ross, A Jain, Information fusion in biometrics, *Pattern Recogn. Lett.* 24, , 2003, 2115-2125.
- Ross, 2006; Arun Ross, Karthik Nandakumar, Anil K Jain, *Handbook of Multibiometrics*, Springer, 2006.
- Rumelhart, 1986; Rumelhart, D.E., Hinton, G.E., Williams, R.J., Learning internal representation by error propagation. In: Rumelhart, D.E., McClelland, J.L.

- (Eds.). *Parallel Distributed Processing: Exploration in the Microstructure of Cognition*, Vol. 1. MIT Press, Cambridge, MA, Chapter 8, 1986.
- Sanderson, 2004; C. Sanderson, K.K. Paliwal, Identity Verification Using Speech and Face Information, *Digital Signal Processing*, Vol. 14, No. 5, 2004, pp. 449-480.
- Schalkoff, 1997; Schalkoff, R.J., *Artificial Neural Networks*. McGraw-Hill, New York, 1997
- Schmidt, 1993; Schmidt, W., Raudys, S., Kraaijveld, M., Skurikhina, M., Duin, R., Initialization, backpropagation and generalization of feed-forward classifiers. In: *Proceeding of the IEEE International Conference on Neural Networks*, pp. 598–604, 1993.
- Schroeder, 1977; Schroeder M. R, Recognition of complex acoustic signals, *Life Sciences*, Research Report, T. H. Bullock ed., 1977, 55, 323-328.
- Sen, 2002; Sen W, Yangsheng W, Fingerprint enhancement in the singular point area. *IEEE Signal Process Lett* 11(1), 2002.
- Sengoopta, 2002; Sengoopta C., *Imprint of the Rang: the Emergence of Fingerprinting in India and its voyage to Britain*, London, Macmillan, 2002.
- Sherlock, 1992; Sherlock B.G., Monro D.M. and Millard K., Algorithm for enhancing fingerprint images, *Electronics Letters*, vol. 28, no. 18, pp. 1720, 1992.
- Sherlock, 1994; Sherlock B.G., Monro D.M. and Millard K., Fingerprint enhancement by directional Fourier filtering, *IEE Proceedings Vision Image and Signal Processing*, vol. 141, no. 2, pp. 87–94, 1994.
- Simon, 2004; Simon A. Cole, History of Fingerprint Pattern Recognition, Nalini Ratha and Ruud Bolle, Eds., *Automatic Fingerprint Recognition Systems*, Springer, NY, 2004.
- Sparrow, 1985; Sparrow M. and Sparrow P., A Topological Approach to the Matching of Single Fingerprints: Development of Algorithms for Use on Latent Fingermarks, U.S. Government Publication/U.S. Department of

- Commerce, National Bureau of Standards, Gaithersburg, MD/Washington, DC, 1985.
- Srinivasan 2006; Srinivasan H., Srihari S.N., Beal M.J., Phatak P. and Fang G., Comparison of ROC-Based and Likelihood Methods for Fingerprint Verification, in Proc. SPIE Conf. on Biometric Technology for Human Identification III, 2006.
- Srinivasan, 1992; Srinivasan V.S. and Murthy N.N., Detection of singular points in fingerprint images, *Pattern Recognition*, vol. 25, no. 2, pp. 139–153, 1992.
- Stock, 1969; R. M. Stock and C.W. Swonger, Development and Evaluation of a Reader of Fingerprint Minutiae, Tech. Report: No. XM-2478-X-1:13-17, Cornell Aeronautical Laboratory, 1969.
- Stock, 1977; Stock R.M., Automatic Fingerprint Reading, in Proc. Int. Carnahan Conf. on Electronic Crime Countermeasures, pp. 16–28, 1977.
- Stockman, 1982; Stockman G., Kopstein S. and Bennett S., Matching images to models for registration of and object detection via clustering, *IEEE Transactions on Pattern Analysis Machine Intelligence*, vol. 4, no. 3, pp. 229–241, 1982.
- Subramanian, 2006; V. Ramasubramanian et. al. Text-dependent speaker-recognition systems based on one-pass dynamic programming algorithm. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pages 901–904, 2006.
- Swingle, 1996; Swingler, K., *Applying Neural Networks: A Practical Guide*. Academic Press, New York, 1996.
- Tamura, 1978; Tamura H., A Comparison of Line Thinning Algorithms from Digital Topology Viewpoint, in Proc. Int. Conf. on Pattern Recognition (4th), pp. 715–719, 1978.
- Torpey, 2001; Torpey, J, J. Caplan Eds., *Documenting Individual identity: The Development of State Practices since the French revolution*, Princeton: Princeton University Press, 2001.

- Toshio, 2004; Kamei T., Image filter design for fingerprint enhancement, in Automatic Fingerprint Recognition Systems, N. Ratha and R. Bolle (Eds.), Springer, New York, pp. 113–126, 2004.
- Troup, 1894; Troup C. E., A. Griffiths and M. L. Macnaghten, Report of the committee appointed by the Secretary of the State to inquire in the best means available for identifying habitual criminals in *British Sessional Papers, House of Commons*, London, : Eyre and Spittiswoode, 1894.
- UID, 2013: www.uidai.gov.in
- Varhney, 1997; P.K. Varshney, Distributed Detection and Data Fusion, Springer-Verlag, New York, 1997.
- Verlinde, 1999; P. Verlinde, A contribution to multi-modal identity verification using decision fusion, PhD Thesis, Department of Signal and Image Processing, Telecom Paris, France, 1999.
- Verma, 1987; M. R. Verma, A. K. Majumdar and B. Chatterjee, Edge Detection in Fingerprints , Pattern Recognition, vol. 20, pp. 513-523, 1987.
- Wang, 2008; W. Wang , L. Jianwei, F. Huang and H. Feng, Design and Implementation of Log-Gabor Filter in Fingerprint Image Enhancement , Pattern Recognition Letters, vol. 29, no. 3, pp. 301-308, 2008.
- Wark, 1999; T.Wark, S. Sridharan, V. Chandran, Robust speaker verification via fusion of speech and lip modalities, In: Proc. International Conf. Acoustics, Speech and Signal Processing, Phoenix, 1999, Vol. 6, pp. 3061-3064.
- Wark, 2000; T. Wark, Multi-modal speech processing for automatic speaker recognition, PhD Thesis, School of Electrical & Electronic Systems Engineering, Queensland University of Technology, Brisbane, 2000.
- Watson, 1994; Watson C.I., Candela G.I. and Grother P.J., Comparison of FFT Fingerprint Filtering Methods for Neural Network Classification, Tech. Report: NIST TR 5493, Sept. 1994.
- Wegstein, 1972; Wegstein J.H., The M40 Fingerprint Matcher, U.S. Government Publication, National Bureau of Standars, Technical Note 878, U.S Government Printing Office, Washington, DC, 1972.

- Wegstein, 1982; Wegstein J.H., An automated fingerprint identification system, U.S. Government Publication, U.S. Department of Commerce, National Bureau of Standards, Washington, DC, 1982.
- Wegstien, 1978; . Wegstein J.H. and Rafferty J.F., The LX39 latent fingerprint matcher, U.S. Government Publication, National Bureau of Standards, Institute for Computer Sciences and Technology, 1978.
- Willis, 2001; 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.
- Wilson, 1994; C. L. Wilson, G. T. Candela, and C. I. Watson, Neural Network Fingerprint Classification, *J. Artificial Neural Networks*, Vol. 1, No. 2, pp.
- Wilson, 1997; Wilson C.L., Watson C.I. and Paek E.G., Combined optical and neural network fingerprint matching, *Proc. of SPIE (Optical Pattern Recognition VIII)*, vol. 3073, pp. 373–382, 1997.
- Wilson, 2004; C. Wilson, A. R. Hicklin, M. Bone, H. Korves, P. Grother, B. Ulery, R. Micheals, M. Zoepfl, S. Otto, and C. Watson. Fingerprint Vendor Technology Evaluation 2003: Summary of Results and Analysis Report. NIST Technical Report NISTIR 7123, National Institute of Standards and Technology, June 2004.
- Wythoff, 1993; Wythoff, B.J., Backpropagation neural networks: a tutorial *Chemometr. Intell. Lab. Syst.* 18, 115–155. 31, 1993.
- Xiang, 1989; Xiang-Xin Z and L. Chun-Ge, The historical application of hand prints in Chinese litigation, *Fingerprint Whorld*, 14 (55), 84-88, 1989.
- Yacoub, 1999; S. Ben-Yacoub, Y. Abdeljaoued, E. Mayoraz, Fusion of face and speech data for person identity verification, *IEEE Trans. Neural Networks*, 10, 1999, 1065–1074.
- Yang, 2003; J. Yang, L. Liu, T. Jiang and Y. Fan, A Modified Gabor Filter Design Method for Fingerprint Image Enhancement , *Pattern Recognition Letters*, vol. 24, no. 12, pp. 1805- 1817, 2003.

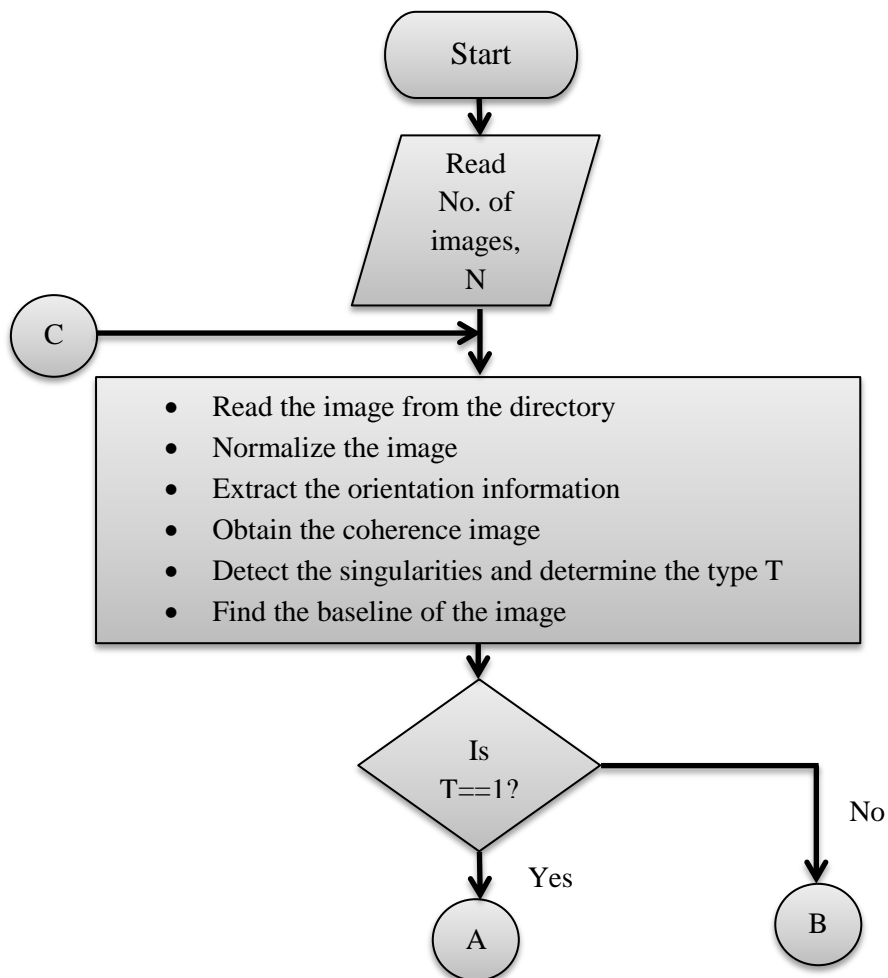
- Yun, 2004; Yun E, Hong J, Cho S Adaptive enhancing of fingerprint image with image characteristics analysis. AI, LNAI 3339. Springer, Berlin, pp 120–13, 2004.
- Zhang, 1996; Y. Y. Zhang and P. S. P. Wang, A Parallel Thinning Algorithm with Two-Sub Iteration that Generates One Pixel Wide Skeletons , Proceedings of the IEEE Conference on Pattern Recognition, Vienna , Austria, vol. 4, pp. 457- 461, Aug., 1996.
- Zhang, 2002; W. Zhang and Y. Wang, Core-Based Structure Matching Algorithm of Fingerprint Verification, Proceedings of the 16th International Conference on Pattern Recognition, Quebec City, Canada , pp. 70-74, August, 2002.
- Zhang, 2006; Y. Zhang and Q. Xiao, An Optimized Approach for Fingerprint Binarization, Proceedings of the International Joint Conference on Neural Networks, Vancouver, BC, Canada, pp. 391-395, July, 2006.
- Zhao, 2010; Q. Zhao, D. Zhang, L. Zhang and N. Luo, High Resolution Partial Fingerprint Alignment using Pore–Valley Descriptors, Pattern Recognition, vol. 43, pp. 1050-1061, 2010.
- Zhixin, 2006; Zhixin S. and Govindaraju V., Fingerprint Image Enhancement Based on Skin Profile Approximation, in Proc. Int. Conf. on Pattern Recognition (18th), vol. 3, pp. 714– 717, 2006.
- Zunkel, 1999; R. Zunkel. Hand Geometry Based Authentication. In A. K. Jain, R. Bolle, and S. Pankanti, editors, Biometrics: Personal Identification in Networked Society, pages 87–102. Kluwer Academic Publishers, London, UK, 1999.
- Zupan, 1993; Zupan, J., Gasteiger, J., Neural Networks For Chemists: An Introduction. VCH, New York, 1993.

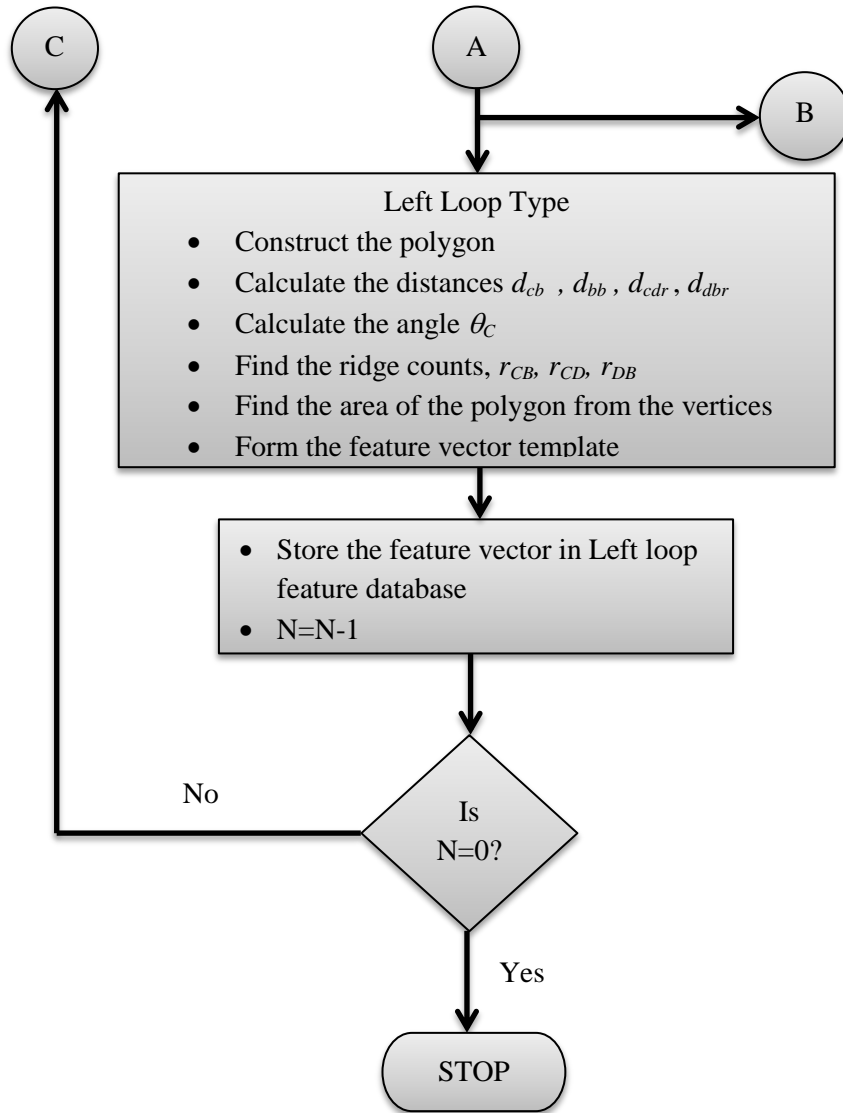
Publications

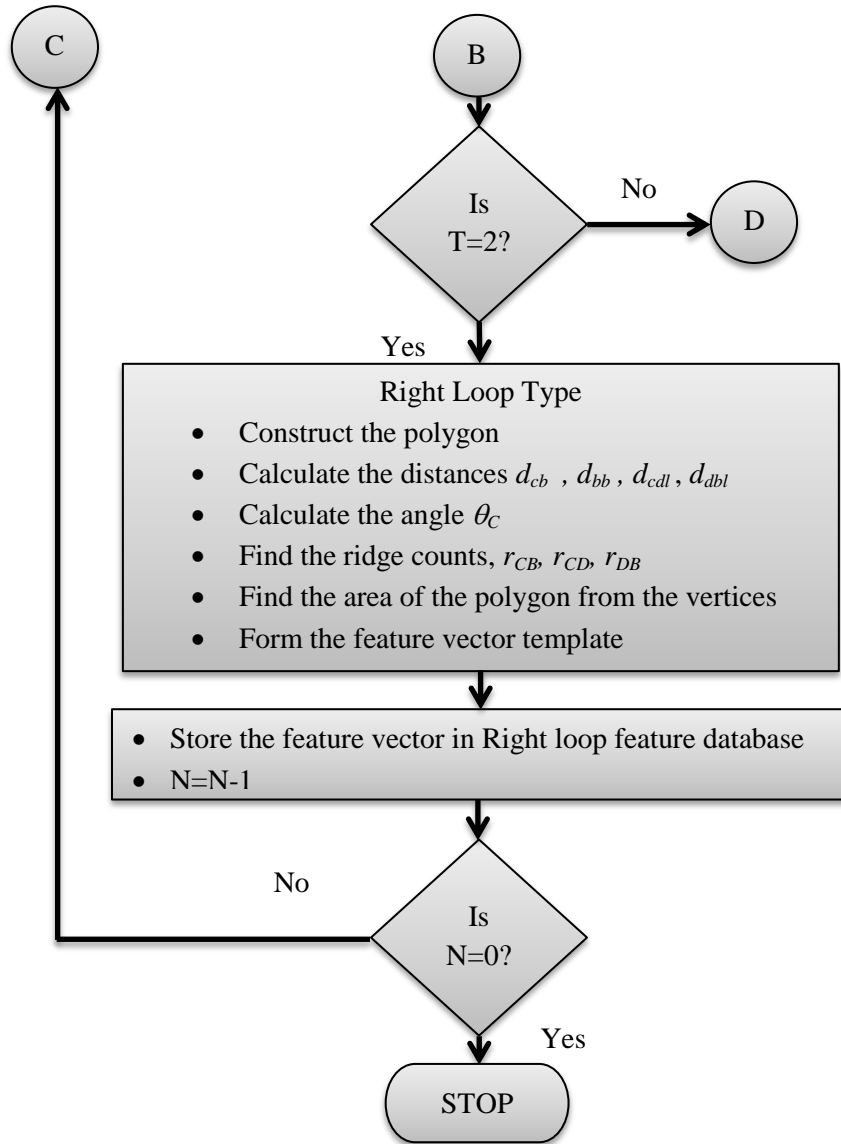
1. Praveen N , Tessamma Thomas, *A novel Approach for Fingerprint Identification Based non Polygonisation of Singularities*”, Proceedings of the Second International Conference on Signal and Image Processing ICSIP-2009 pp 536-529, August, 2009.
2. Praveen N and Tessamma Thomas, “*Singularity Based Fingerprint Identification*”, International Journal of Research and Reviews in Computer Science (IJRRCS) Vol. 2, No. 5, October 2011, ISSN: 2079-2557 Science Academy Publisher, United Kingdom, pp. 1155-1159
3. Praveen N, Tessamma Thomas, *Multifinger Feature Level Fusion Based Fingerprint Identification(IJACSA)* International Journal of Advanced Computer Science and Applications, Vol. 3, No. 11, 2012.
4. Praveen N, Tessamma Thomas, Text-dependent Speaker recognition by MFCC and Backpropagation Neural Network. Submitted to Computer Engineering and Intelligent Systems Journal, IISTE.
5. Praveen N, Tessamma Thomas, Score level fusion based personal authentication using fingerprint and speech. Submitted to IETE Journal of Research.

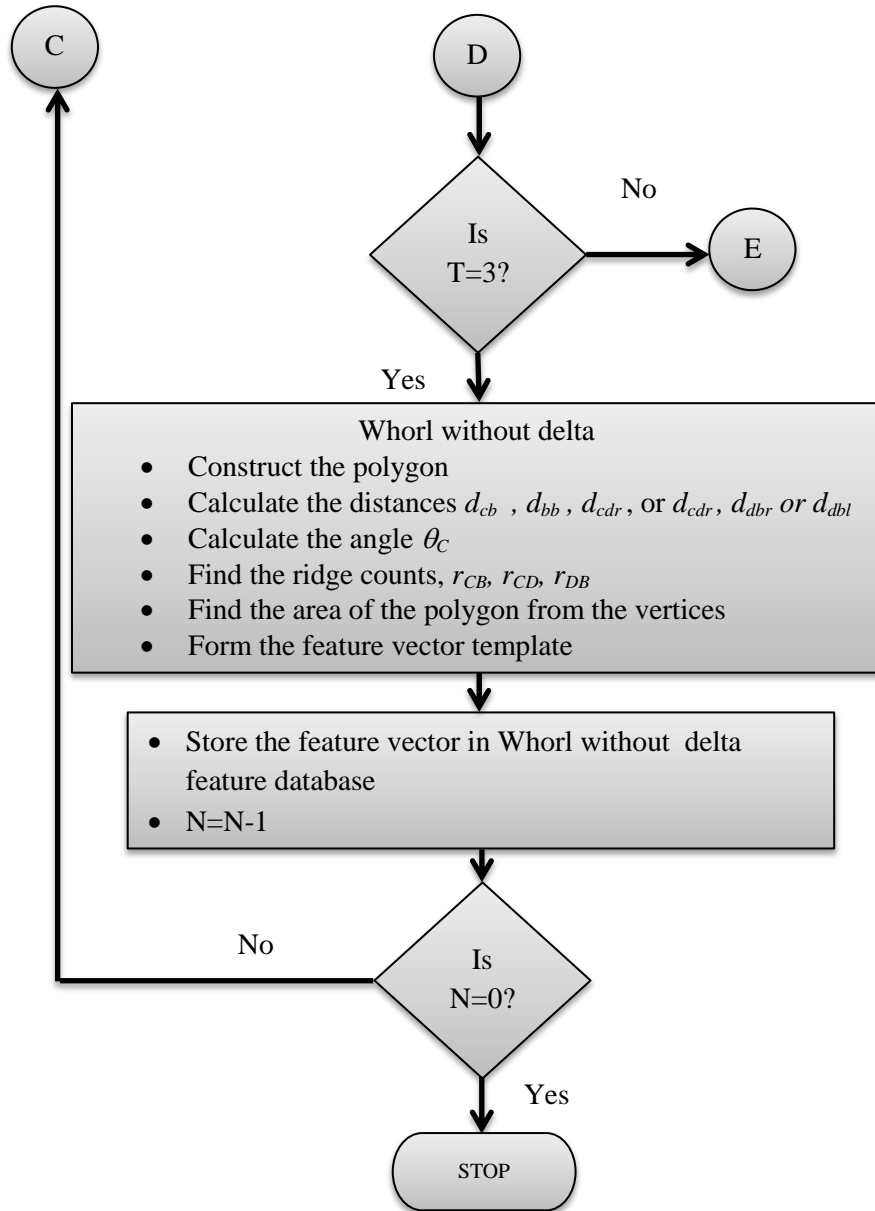
Appendix A

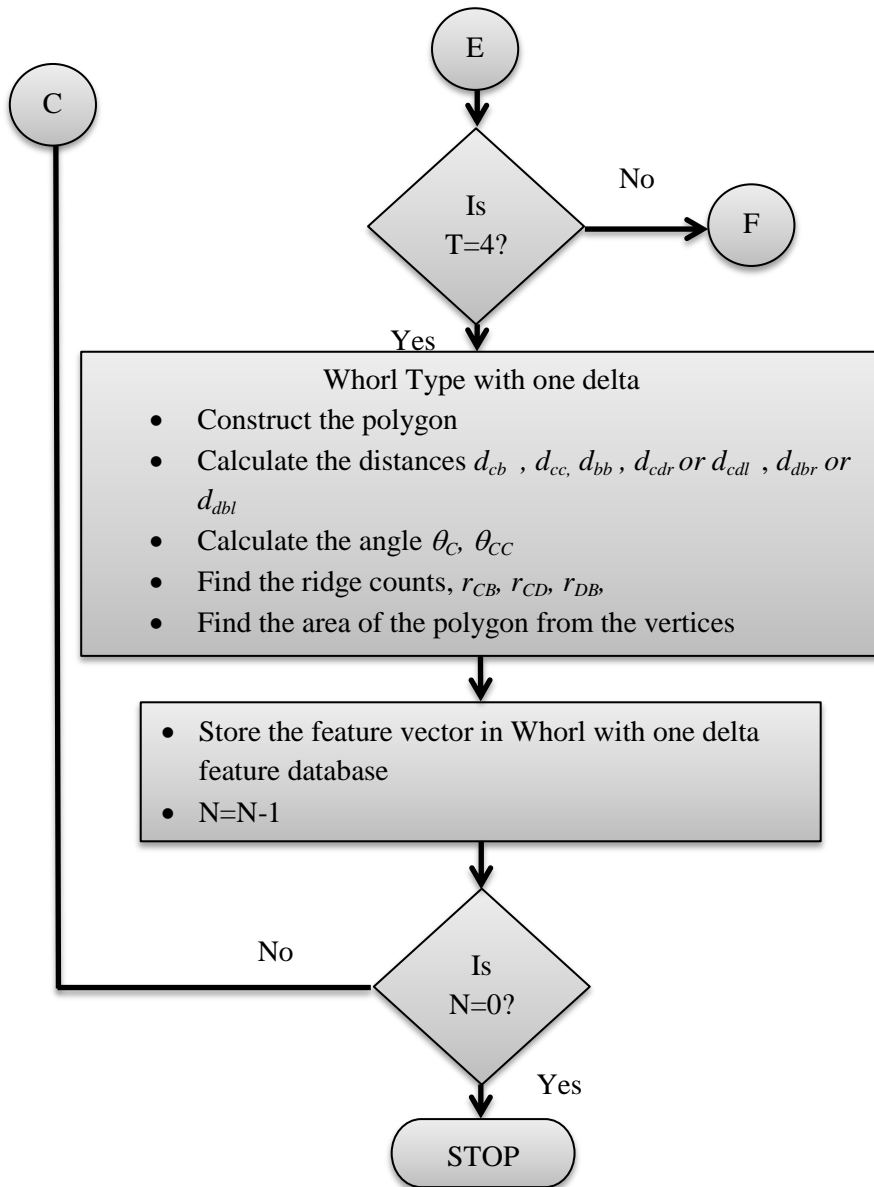
A.1 Flowchart for Fingerprint Feature Extraction

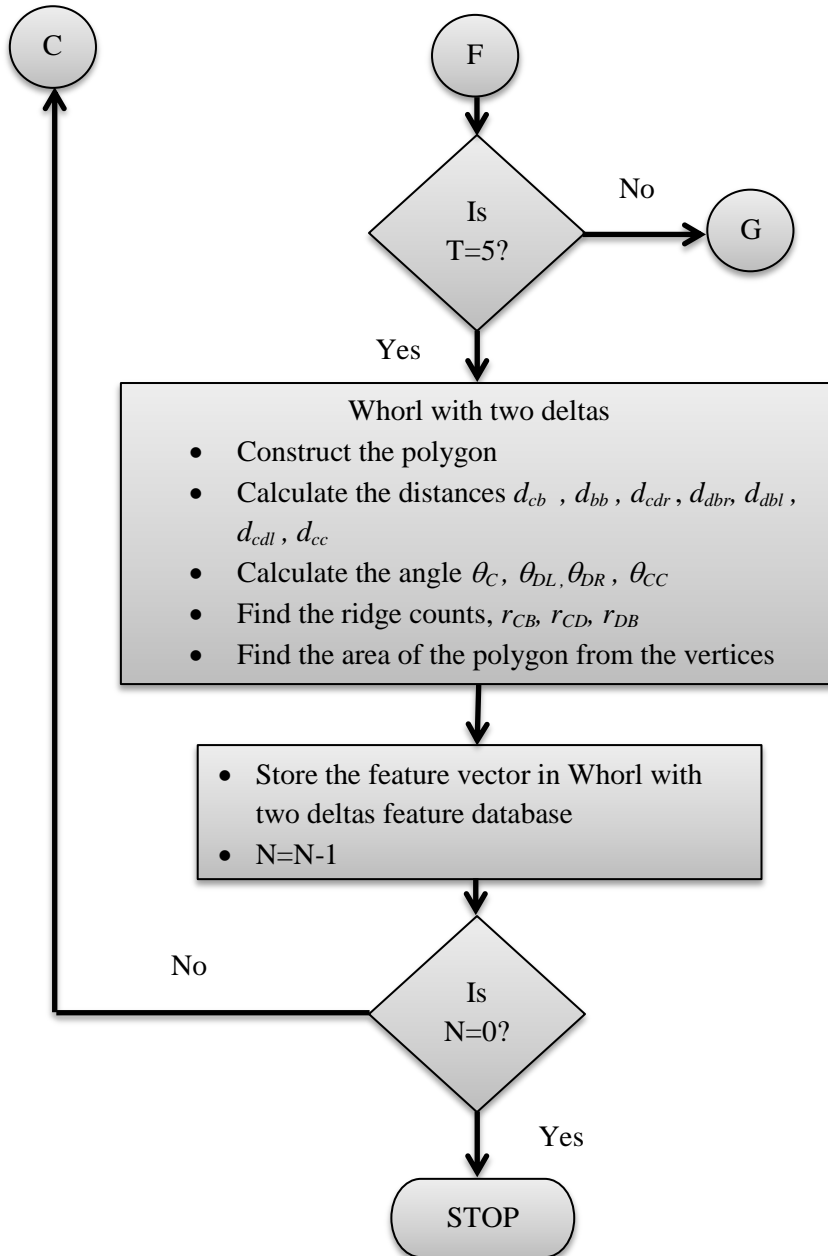


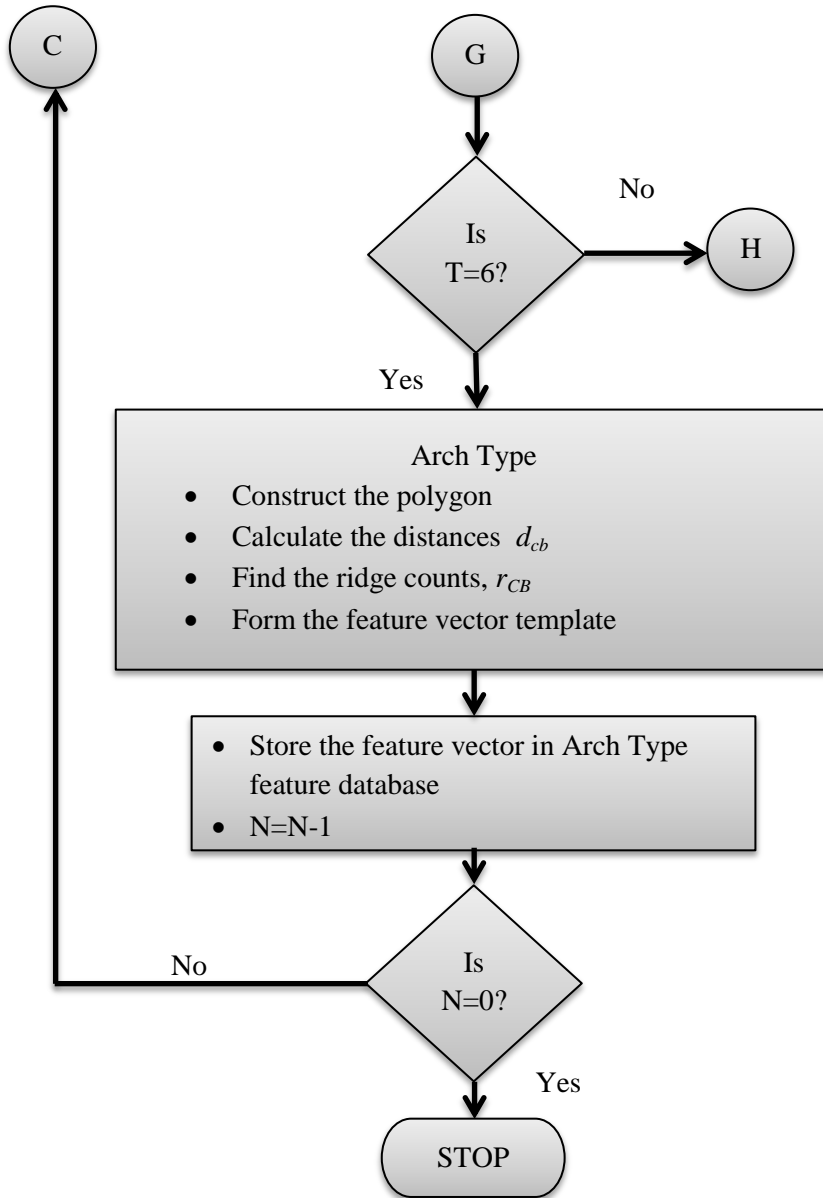


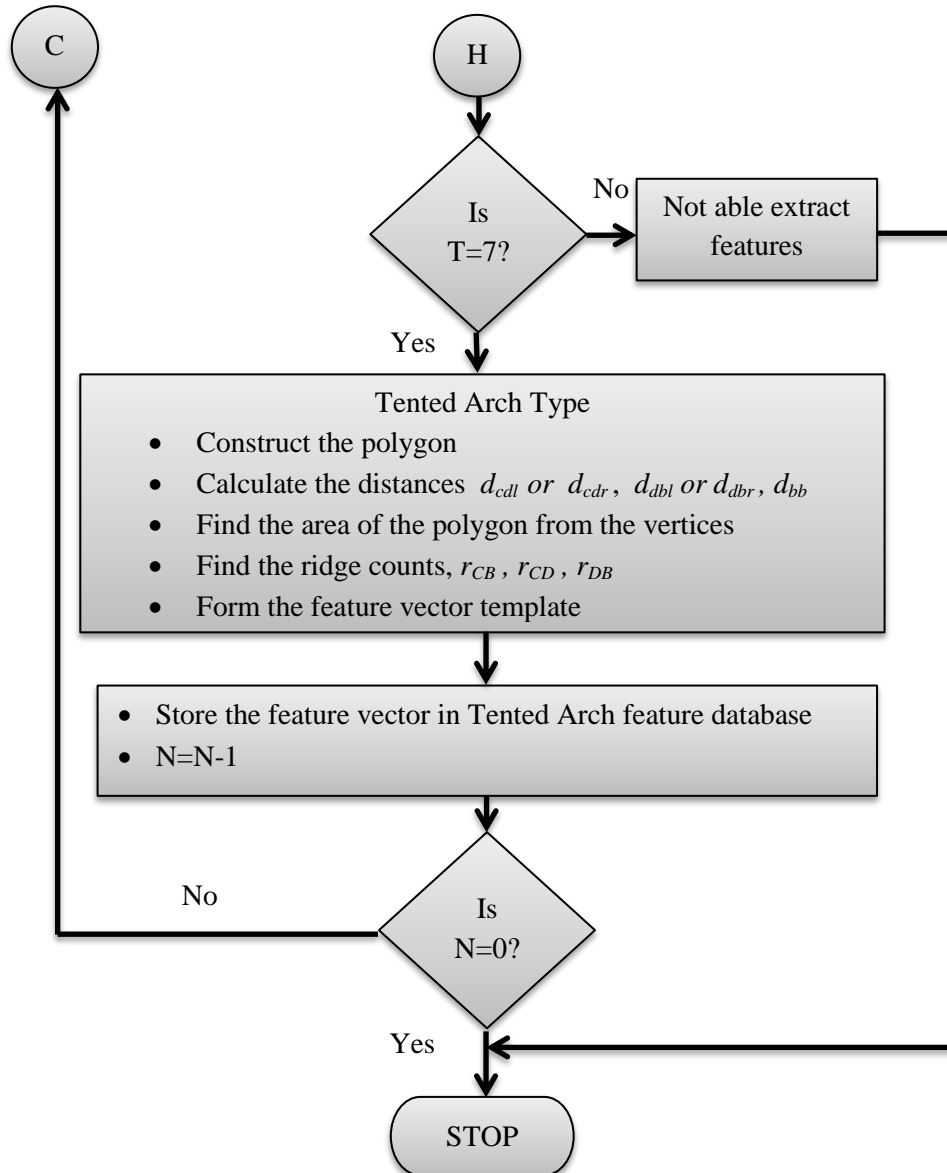












A.2 Fingerprint Matching

