Sheffield Hallam University

Fingerprint-based biometric recognition allied to fuzzy-neural feature classification.

MOHAMED, Suliman M.

Available from the Sheffield Hallam University Research Archive (SHURA) at:

http://shura.shu.ac.uk/20071/

A Sheffield Hallam University thesis

This thesis is protected by copyright which belongs to the author.

The content must not be changed in any way or sold commercially in any format or medium without the formal permission of the author.

When referring to this work, full bibliographic details including the author, title, awarding institution and date of the thesis must be given.

Please visit http://shura.shu.ac.uk/20071/ and http://shura.shu.ac.uk/information.html for further details about copyright and re-use permissions.

CITY CAMPUS, HOWARD STREET
SHEFFIELD S1 1WB

101 698 958 X

Fines are charged at 50p per hour

19 JAN 2006 9.00 p.m

20 JAN 2006
6pm

26 JAN 2006
9pm

12/11/07
4:10pm

ProQuest Number: 10697378

All rights reserved

INFORMATION TO ALL USERS
The quality of this reproduction is dependent upon the quality of the copy submitted.

In the unlikely event that the author did not send a complete manuscript
and there are missing pages, these will be noted. Also, if material had to be removed,
a note will indicate the deletion.

ProQuest 10697378

Published by ProQuest LLC (2017). Copyright of the Dissertation is held by the Author.

All rights reserved.
This work is protected against unauthorized copying under Title 17, United States Code
Microform Edition © ProQuest LLC.

ProQuest LLC.
789 East Eisenhower Parkway
P.O. Box 1346
Ann Arbor, MI 48106 – 1346

Fingerprint-Based Biometric Recognition Allied to Fuzzy-Neural Feature Classification

Suliman M Mohamed

A Thesis Submitted in Partial Fulfilment of the
Requirements of the Sheffield Hallam University
for the Degree of Doctor of Philosophy

April 2002

SHEFFIELD HALLAM UNIVERSITY
TH
★ ★
ADSETTS CENTRE

ABSTRACT

The research investigates fingerprint recognition as one of the most reliable biometrics identification methods. An automatic identification process of humans-based on fingerprints requires the input fingerprint to be matched with a large number of fingerprints in a database. To reduce the search time and computational complexity, it is desirable to classify the database of fingerprints into an accurate and consistent manner so that the input fingerprint is matched only with a subset of the fingerprints in the database. In this regard, the research addressed fingerprint classification. The goal is to improve the accuracy and speed up of existing automatic fingerprint identification algorithms. The investigation is based on analysis of fingerprint characteristics and feature classification using neural network and fuzzy-neural classifiers.

The methodology developed, is comprised of image processing, computation of a directional field image, singular-point detection, and feature vector encoding. The statistical distribution of feature vectors was analysed using SPSS. Three types of classifiers, namely, multi-layered perceptrons, radial basis function and fuzzy-neural methods were implemented. The developed classification systems were tested and evaluated on *4,000* fingerprint images on the NIST-4 database. For the five-class problem, classification accuracy of 96.2% for FNN, 96.07% for MLP and 84.54% for RBF was achieved, without any rejection. FNN and MLP classification results are significant in comparison with existing studies, which have been reviewed.

DEDICATION

This thesis is dedicated to my parents, my wife, Naglla; my son, Mustafa, my daughters Ma-ab and Mehad, and to my family members, friends...This achievement is ours.

ACKNOWLEDGEMENT

First and foremost thanks to God who gave me the power, health, and motivations that gave me a chance to explore this topic and allowed me to complete this work.

I would like to express my gratitude and thanks to my Director of Studies Dr Henry O. Nyongesa for his excellent supervision co-operation and encouragement during this work and for help that he provided to me at the most difficult times.

I would like to thank the Government of the Sudan and the Sudanese General Administration of Human Resource Development, who funded this research. And extend my thanks to the rest of my supervisory team Prof. J Siddiqi and Dr Marcos Rodrigues, as well to the members of CRC and the staff of CMS at Sheffield Hallam University, to Dr B. Khamis, Ms Jo Laughton and Dr A. Babalkhar, to all my colleagues in the Harmer Building room 2416, and to my parents, relatives, friends all for their encouragement and support. I would to express my appreciation for all their supports directly and indirectly.

NOTE

The author hereby declares that, except where duly acknowledge, this thesis is entirely his own work. This point is particularly emphasised in relation to all figures and tables unless otherwise mentioned. Where these have been reproduced, the original sources are acknowledged. Otherwise, they represent the author's own work.

OVERVIEW

I. Thesis Objectives and Research Questions

The objective of the thesis is to investigate human identification based on fingerprint biometric identification systems, design a fingerprint classification system, and study the performance of fingerprint classification in automatic identification. And the aim of this research is to examine the issue of fingerprint recognition related to the following Research Questions:

RQ1. Is fingerprint-based biometric ID reliable?

RQ2. How can fingerprint feature extraction and encoding be efficiently achieved?

RQ3. Is fingerprint classification an important step in fingerprint identification?

RQ4. How fingerprint classification, with reasonable error rate, can be achieved?

II- Overview of Research

These crucial questions have been answered in the content of this thesis. An automatic fingerprint recognition system is concerned with some or all of the following issues: fingerprint acquisition, fingerprint enhancement, fingerprint feature extraction, fingerprint classification, fingerprint matching (verification/identification). A particular emphasis has been given to fingerprint classification problems, which have been investigated using computational intelligence methodologies. The seven Chapters of the thesis are summarised as follows:

Chapter 1. Introduces the issue of human-ID problem and discusses the basis for formal identification, including, conventional-based techniques, such as, names, codes, knowledge-based, token-based identification, and biometric-based techniques. An overview of biometric approaches, such as, fingerprint, face, ear, iris, voice, DNA, retina, hand-geometry are discussed. A brief historical review of the fingerprint recognition system and its advantages is also given.

Chapter 2. Presents analysis and discussion of fingerprint characteristics and features. An overview of a fingerprint recognition system is discussed, leading to general steps for implementation of a complete automatic fingerprint recognition system.

Chapter 3. Presents fingerprint image processing techniques by reviewing current approaches, in general, and a discussion of classification techniques.

Chapter 4. Discusses automatic fingerprint feature extraction techniques, and presents the proposed fingerprint feature extraction and feature encoding algorithm.

Chapter 5. Discusses statistical properties of the extracted features and implementation of fingerprint classification using neural and fuzzy-neural networks classifiers.

Chapter 6. Presents an analysis and discussion of the experimental results obtained from the implemented classifiers and provides a comparison with previous studies.

Chapter 7. Presents conclusions and remarks on research methodology. A summary of contributions and further research is also provided.

Appendix 1. Presents overview of NIST-4 fingerprints database and fingerprint images file format.

Appendix 2. Discusses steps for implementation of fuzzy-neural automatic network construction in Neuframe^{TM}.

Appendix 3. Copies of some significant papers, which have been published in Journals and Proceedings arising from this research.

LIST OF PUBLICATIONS

The following publications (Journals + Proceedings) have been published arising from this research:

1. Suliman M Mohamed & Henry O Nyongesa, Automatic Fingerprint Classification System Using Fuzzy Neural, Proceeding of the 2001 International Conference on AI, Vol. 1, PP. 395-401, Las Vegas, Nevada, USA, June, 2001, ISBN: 1-892512-78-5.

2. Suliman Mohamed and Henry Nyongesa, Fingerprint Feature Extraction for Classification, Proceedings of the 27th International Conference in Computing and Industrial Engineering, Beijing, August 28-30, 2000.

3. Suliman Mohamed, Sayed Horbaty, and Aboul-Ella Hassanien, An Image Metamorphosis Algorithm Based on Navier Spline Interpolation, Egyptian Computer Society Journal vol. 22, no. 1, pp. 9-12, Jan. 2000

4. Suliman M Mohamed, Henry O Nyongesa and Jawed Siddiqi, Automatic Fingerprint Identification System Using Fuzzy Neural, Proceeding of the 2000 International Conference on AI, Vol. 2, PP. 859-865, Las Vegas, Nevada, USA, June, 2000, ISBN: 1-892512-57-2.

5. Suliman M Mohamed and Henry O Nyongesa, Biometrics Based ID Security In 21st Century Technologies, Proceeding of 4th Conference Association of Egyptian-American Scholars, PP. 36-45, Toronto, Ontario, Canada, June 2-4, 2000,.

6. Suliman M Mohamed and Henry O Nyongesa, Fingerprint Recognition System Using Fuzzy Neural Techniques, Proceeding of 8th International Conference in AI Applications, PP. 295-305, Egypt, Cairo, February 3-6, 2000.

7. Suliman M Mohamed and Henry O Nyongesa, Image Pattern Recognition Using Fuzzy/Self-Organising Network, Proceeding of the 6th UK Workshop on Fuzzy Systems, PP. 115-120, Brunel University, Uxbridge, UK, September, 1999.

8. Suliman M Mohamed, Image Pattern Processing Using AI Applications, Proceeding of 9th UK Ph.D. Consortium, PP. 26-27, Birmingham, UK, June, 1999.

9. Suliman Mohamed and Henry Nyongesa, Automatic Fingerprint-based Biometric Recognition using Fuzzy/Neural Networks Techniques; paper accepted by Computers and Industrial Engineering, April, 2002.

10. Suliman Mohamed and Henry Nyongesa, Fingerprint Classification using Fuzzy Neural Networks accepted for 2002 Fuzz-IEEE Hawaii, USA 12-17 May 2002.

11. Suliman M Mohamed, Computer Crime and Resolutions, The Arab Journal of Crime Attack, Vol. 11, PP 54-57, Feb. 2001.

Tables of Contents

Chapter 1. Introduction to Human Identification

Chapter 2. Fingerprint Characteristics, Analysis and Recognition

Chapter 3. Review of Fingerprint Image Processing and Classification Techniques

Chapter 4. Fingerprint Feature Extraction

Chapter 5. Fingerprint Classification by using Fuzzy-Neural Network Classifiers

Chapter 6. Experimental Results and Analysis

Chapter 7. Conclusions and Further Works

Appendix 1. NIST Special Database 4 Fingerprint Image File Format

Appendix 2. Implementation Steps of Fuzzy-Neural Classifier

List of Tables

Chapter 1

Chapter 2

Chapter 4

Chapter 5

Chapter 6

List of Figures

Chapter 1

Chapter 2

Chapter 3

Chapter 4

Chapter 5

Chapter 6

Chapter 7

Appendix 1

Appendix 3

Chapter 1

Introduction to Human Identification

1.1 The Issue of Human Identification

The issue of the Human Identification (H-ID) is an assertion about an individual's identity. In the simplest case, this assertion could be a claim that the individual makes. H-ID means both Identification and Authentication (Verification). Identification is the process for establishing the identity of the individual. Authentication refers to the process by which a system establishes that an identification assertion is valid. A number of authentication and identification mechanisms are commonly used in practice; each has advantages and disadvantages. This chapter undertakes a survey of H-ID techniques.

H-ID is a delicate notion and a practical matter that requires consideration at the levels of philosophy and psychology. In a variety of contexts, each of us needs to identify other individuals, in order to conduct a conversation or transact business. Organisations also seek to identify the individuals with whom they deal, variously to provide better service to them, and to protect their own interests. In earlier civilisations, branding and even maiming were used to mark the criminal for what he was. The thief was deprived of the hand, which committed the thievery. The Romans employed the tattoo needle to identify and prevent desertion of mercenary soldiers [Clarke94a]. More recently, law enforcement officers with extraordinary visual memories, so-called "camera eyes" identified past offenders by sight. Photography lessened the burden on memory but was not the answer to the criminal identification problem, because personal appearances change.

Historically, H-ID mechanisms have been characterized into three approaches: something you know, something you have, and something you are. The key challenge to H-ID systems management is identified as being able to devise a scheme, which is practicable, secure and economic, and of sufficiently high integrity to address the risks the organisation or individuals confront in dealing with people. It is suggested that much greater use be made of schemes which are designed to afford people anonymity, or enable them to use multiple identities or pseudonyms, while at the same time protecting the organisation's own interests [Davies94].

Currently, human identification methods fall into five board categories; name-based (e.g. surname, other-names, religion-name), codes-based (e.g. digit code, alpha-numeric codes), tokens-based (e.g. identity badges, ID cards, passport, smart cards, credit cards, keys), knowledge-based (e.g. Personal Identification Numbers (PINs), passwords, user-names), and biometric-based (e.g. fingerprint, face, hand geometry, signature, iris, ear, voice recognition). The problems with the first four methods are that names and codes suffer from pseudonyms and multiple identities; the tokens-based are easily lost or stolen; and the knowledge-based are all too likely to be forgotten or stolen. Two or more of these methods may be combined. However, individuals may still inadvertently weaken security by adopting bad practices, such as, writing down PINs and passwords or re-using them for multiple applications. From the user's perspective, none of these validation methods is ideal [Jain01]. In today's interconnected information society an accurate automatic personal identification is becoming very important to a wide range of application domains (e.g. intranet, e-commerce, distance learning, welfare benefits disbursement and automatic access control). The need to identify and authenticate ourselves to machines is a crucial issue in today's electronic society, and its necessary to close the gap between human and machine understanding to secure our transactions and networks in order protect the interest of organisations and individuals.

Many organisations store data about people in their systems. In order, to reliably associate data with particular individuals, it is necessary that an effective and efficient identification scheme be established and maintained. As information technology becomes the key to wealth in the 21st century, biometric-based security will play a central role in providing a high level of security to existing and future products. Biometrics Identification (Bio-ID) technology is the science of automatically identifying individuals based on their physiological or behavioural characteristics, or so-called positive personal identification. Bio-ID technology has advanced tremendously over the last few years and has moved from research laboratories and Hollywood to real-world applications. Like any technology with commercial applications, it has been difficult, until now, to assess the state-of-art in Bio-ID in the open literature. Biometrics offer new solutions that are user friendly and accessible. This Chapter attempts to disseminate the technological aspects and implications of biometrics methodologies in comparison with the conventional methods. In particular, the Chapter surveys the biometric techniques in commercial use and research stages, assesses the capabilities and limitations of different biometrics items, disseminates the principles of design of biometric systems; compares the performance of the current biometric items, and investigates personal privacy and security implications of biometric-based identification technology.

Currently, the performance of biometric systems is gauged by error rates, namely, False Accepts Rate (FAR), and False Reject Rate (FRR) [Hong97]. FAR occurs when an unauthorised user is identified as an authorised user and, therefore, accepted by the system (i.e. the acceptance of impostors and mistaken matching). FRR occurs when an authorised user is not recognized as such, and is rejected by the system (i.e. rejecting or failing to recognise correct matches). In order to describe the performance of a system, both the FAR and FRR must be determined. These FAR and FRR are accepted as the metrics by which biometric system performance is judged today, although the final judgement is dependent on many other issues, such as ergonomics (ease of use), universality, uniqueness, permanence (unchangability), acceptability, speed, hardware simplicity, plus the area of the application. The investigation in this research addresses fingerprint-based identification technology as the most practical biomteric for its accuracy, uniqueness, permanence, hardware simplicity and applicability.

Fingerprint-based identification is one of the most significant and reliable biometric-based identification methods [Jain99a]. Every individual has a unique fingerprint and it offers an infallible means of personal identification. Other personal characteristics may change but fingerprints do not. It is virtually impossible that two people have the same fingerprint, (probability 1.9E-15) [Hong97]. Therefore, fingerprints have been used in many applications since the 18-century, including law-enforcement, civilian and commercial applications. The nature and security of each application requires a different degree of accuracy. For example, a criminal case may require a higher degree of matching than an access control system. Furthermore, in terms of performance, applications such as banking would like to have a system with higher FAR than FRR, while security organisations would like to have a system with higher FRR than FAR.

The rest of this Chapter is organised as follows. In section 1.2, an overview of H-ID is presented, defining human identity and human identification. Section 1.3 discusses the needs of organisations for formal identification and H-ID management challenges.

Section 1.4 presents conventional approaches for formal identification. Section 1.5 presents biometric-based technologies, while Section 1.6 reviews fingerprint identification. Section 1.7 is a summary of the Chapter.

1.2 Overview of Human Identification

In order to understand the H-ID we need to define what is meant by both the human identity and human identification, as well some other related terms like authentication (verification) and matching. In the context under discussion, identity is used to mean to recognise correctly someone or something (e.g. "I agreed to identify the body") [L-Dict95], "the condition of being a specified person" [O-Dict76], or "the condition of being oneself and not another" [M-Dict81]. It implies the existence for each person of private space or personal *lebensraum*, in which one's attitudes and actions can define one's self. The dictionary definitions, however, miss a vital aspect. The origin of the term implies equality or 'one-ness', but identities are no longer rationed to one per physiological specimen. A person may adopt different identities at various times during a life-span, and some individuals maintain several at once. Nor are such multiple roles illegal or even used primarily for illegal purposes.

The term 'identification' means, "the act or process of establishing the identity of a person, or recognising him", or "the treating of a thing as identical with another" [O-Dict76], or "the act or the process of recognising or establishing as being a particular person", or "the act or the process of making, representing to be, or regarding or treating as the same or identical" [M-Dict81]. Within the context of information communication and technology, and e-business the purpose of identification is more concrete: it is used to link a stream of data with a person.

Identification is applicable to data stored in structured, tangible and manageable form, such as in corporate databases and document filing schemes; to data stored in less formal ways, as in private notes; and in incorporeal form, such as, human knowledge and memory, behaviour or characteristics.

The original needs for identification were social rather than economic. The social dimension of human culture is reflected by the idea of a person 'identifying' with a group. Indeed, group-membership ('One of Us' or 'One of Them') was probably a far more important matter than individual identity ('Who am I' or 'Who is he/she'), or even ('I', or 'You') throughout pre-historic times and most of the historic era [Clarke92].

Relatives, friends and acquaintances recognise a person on a contextual basis, in which physical appearance, voice characteristics, knowledge of private information, location and espoused name, or even colour all play a part. These features are not individually reliable, they only work when the people involved are in close proximity, and they depend upon human memory, with all its vagaries. They are, however, sufficient for most social purposes.

As the complexity of economic transactions developed, the need arose for parties to know with whom they were dealing. It became normal for parties to provide one another with information about themselves, appropriate to the nature of the transaction. This may have been an explicit identifier (e.g. of the property the person owned, such as 'the Henry's Bookshop', or perhaps of the person himself, e.g. 'Mohamed, King Sudan'). Alternatively, a number of pieces of information might together identify the person ('meet me at the bookshop on the corner in ten minutes time; I'll be wearing a white coat with a black lapel'; or 'you'll always find me in this corner of the market on tuesdays'). The purposes of the interchange of identification include providing a gesture of goodwill, to develop mutual confidence, and to reduce the scope for dishonesty; to enable either person to initiate the next round of communications; and to enable either person to associate transactions and information with the other person. If identification is to be more than casual and unreliable, it must have an adequate basis. There are several types of evidence that could be used. Some depend on physical or physiological characteristics of the person, others on more abstract information. In practice a person is accepted as being the person to whom a record relates because they represent themselves as being that person. They know things that in the normal course of events only that person would be expected to know. They do things that would only be in the interests of that person to do. Or they are in possession of a code or token, which it is reasonable to expect that, and preferably only that, person to have. In practice, it is common to use various techniques in combination.

1.3 Organisations Needs for Formal Identification

Most identification mechanisms are fraught with difficulties, and hence the vast majority of transactions involves risk, suffer from drawbacks, and they also cost money. A primary purpose of an identification system is to provide a basis whereby individuals and organisations on the one hand, and system-designers and policy-makers on the other, can use rational processes to implement schemes, which balance high security, costs, benefits, reliability and the risks involved. Both identity and identification are vague and ambiguous. They continue to be treated with considerable looseness by most legal systems, particularly those who are dealing with electronic systems. There are also many circumstances in which informal identification or even none at all, suffices for economic transactions. In some situations, however, organisations have a need for reliable identification of the individuals they deal with. The reason may be to protect the individual; for example, a record of any allergies the individual may have to drugs, and of drugs, which the person is currently taking. More commonly, the purpose is to protect the organisation, for example, to ensure that the person can be contacted and located in the future in the event that he/she does not fulfil an obligation such as payment of a debt, or to guard against a person misrepresenting their status, e.g. their educational qualifications, occupation, income or medical condition.

The 21-century, through development in IT and computer mediated communication, has seen a vast growth in organisational size, and in distance between organisations and people. Organisations have increasingly assumed that there is a need for large quantities of data about people. This 'information-richness' has assumed the dimension of an imperative; to the extent those individuals who demur when asked for evidence of identity are frequently presumed to have something to hide.

Organisations often assume that a one-to-one relationship exists between persons and identities, no matter how many different roles he or she may play, or choose to adopt [Hong97]. There are exceptions, however, in a variety of contexts; for example, many banks and insurance companies have only recently adopted 'client-oriented' approaches (whereby all accounts or policies which each individual has with the company are recorded against a single identifier, rather than being scattered around the company's divisions and branches); and employees who are also customers of their employer frequently have distinct employee and customer codes. Implicit within many of the issues discussed in this Chapter are the questions as to when anonymity is unacceptable and identification necessary; and in what circumstances the restriction of a person to a single identity is appropriate. In establishing the organisation ID schemes, they apply variants and combinations of the techniques described in the section 1.4, in manners appropriate to the circumstances. It is vital that in doing so, they appreciate the difficulties involved. Of special importance is the need to achieve an appropriate balance between the harm arising from FAR and FRR rates (see section 1.1).

1.3.1 Management Challenges

Many organisations no longer rely on their employees to recognise individual clients. They seek a means whereby individual humans can be recognised reliably over a period of time, without reliance on human memory, and (in some cases) despite the preference by the person not to be recognised. The problem of linking the client with his number, and hence with his data, remains a very difficult one [Clarke94a]. Information systems have tended to use codes rather than names as the primary identification mechanism [Hong97]. As information technology develops, there are signs that artificial codes may be beginning to give way to natural names again. This is because processing capabilities for textual data are improving; so too is the handling of non-unique identifiers, of partial data and of partially incorrect data like misspellings. Code-based data management is being supplemented by, and may progressively be supplanted by, text-string-based and contextual techniques.

1.3.2 Inherent Objections to Identification

In considering the design of identification schemes, many interests need to be balanced. Powerful institutions and individuals seek to sustain their influence over individuals, variously as customers, suppliers, employees, patients, critics and others. There is also a collective interest in social control, primarily in order to sustain law and order and thereby protect life, health and property, but also to ensure that equity is achieved or, more realistically, that the inequities are planned, rather than accidental. Against these motivations for tight social control, and an efficient identification scheme to support it, it is necessary to balance the interests of individuals in the various aspects of civil liberty. Private spaces in which people can behave free from intrusions are being rapidly invaded by data surveillance technologies.

The need to identify one-self may be intrinsically distasteful to some people. For example, they may regard it as demeaning, or implicit recognition that the organisation with whom they are dealing exercises power over them [Clarke92]. Many people accept that, at least in particular contexts, an organisation with which they are dealing needs to have their name. Some, however, feel it is an insult to human dignity to require them to use a number or code instead of a name. Some feel demeaned by demands, as part of the ID process, that they reveal information about themselves or their family, or embarrassed at having to memorise a password or PIN. Some people are unwilling to submit to the regimen of carrying tokens, or unprepared to produce them, on the grounds that this reeks of a totalitarian regime, reflects and perpetuates a power relationship that they despise (such as the South African pass laws during the period of apartheid), or carries with it the seeds of discrimination (as reflected by the content of the token) [Blume89]. Another factor which forces compromise between the interests of accountability and law and order on the one hand, and civil liberties on the other, is the importance of multiple identities as a means of avoiding physical harm and death at the hands of violent opponents. This is particularly indicated by debate in United States of America and United Kingdom about the need for smart ID cards following the bombing of the World Trade Centre in New York and Pentagon in Washington on September 11, 2001.

1.3.3 Identification Schemes in Organisations

Organisations vary enormously in the care with which they apply H-ID techniques to their needs. At one extreme, some organisations have no interest whatsoever, as in retail cash sales and public advisory bodies like tourist offices. At the other, a small number of organisations, mainly in criminal investigation, national security, and financial departments depend upon the collection of conventional and physical characteristics. Many organisations depend upon documentary evidence when they establish a relationship with an individual. Thereafter, they usually depend on the person's knowledge (e.g. of his name, his birth-date or his client-code), or on his ability to present a token issued by the organisation itself (e.g. an ID-card, or an Automatic Teller Machine ATM-card), or on a combination of both. Many administrators treat tokens-based evidence as though it were the, or even the only, authoritative basis for identification. In fact, as discussed in sections 1.4, all tokens-based are dependent on a seed document and the integrity of an identification scheme also depends on a provable relationship between the person and the document. A common approach taken by many organisations is to seek a variety of information about a person, from a variety of sources, and, in the absence of inconsistencies or 'bad' references, accept the person as being identified by that loose set of data. For the majority of the population, and the majority of purposes, token-based identification schemes, supported by limited cross checking, provides sufficiently reliable evidence of identity. Most people have no inclination to establish multiple identities, and for many of those who are interested in doing so, the effort, difficulty and cost outweigh the potential benefits. However, for those with an axe to grind, a serious prank to play, or criminal intent, the effort is frequently perceived to be worthwhile.

The lack of integrity of most identification schemes is mirrored by the lack of regard paid to identification cards [Chaum85]. No matter what care and expense is invested in the design and issue of cards, their potential for ensuring accurate identification of individuals is dependent on the assiduousness of gatemen and door attendants. Anecdotes abound of the swapping of cards between bearded black-haired giants and petite blonde women; and of cards carrying the bearer's dog's photograph going undetected for long periods [Clarke94c].

Universality of Coverage	**Every relevant person should have an identifier**
Uniqueness	Each relevant person should have only one identifier no two people should have the same identifier
Permanence	The identifier should not change, nor be changeable
Indispensability	The identifier should be one or more natural characteristics, which each person has and retains. If artificial, the identifier should be enforcedly available at all times
Collectibility	The identifier should be collectible by anyone on any occasion
Storability	The identifier should be storable manually or automatically
Exclusivity	No other form of identification should be necessary or used
Precision	Every identifier should be sufficiently different from every other identifier that mistakes are unlikely
Simplicity	Recording and transmission should be easy and not error-prone
Cost	Measuring and storing the identifier should not be costly
Convenience	Measuring and storing the identifier should not be unduly inconvenient or time-consuming
Acceptability	Its use should conform to contemporary social standards

Table 1.1 Desirable Characteristics of an Identifier

In order to address criteria for assessing the quality of a H-ID system we need to discuss its desirable characteristics [Clarke92]. In Table 1.1, a set of criteria for desirable characteristics has been proposed, whereby an organisation can assess alternative means of identifying people with whom it deals. Inevitably, these objectives exhibit internal conflict.

1.4 Bases for Formal Identification

A variety of means are available for identifying a person, in order to associate data with them. Table 1.2 explains some of these means. This section examines identification using five main identification tools: names-based, codes-based, knowledge-based, token-based, and biometric-based techniques.

Means of Identifying	**Definition**
Names	What the person is called by other people
Code	What the person is called by an organisation
Knowledge	What the person knows
Tokens	What the person has
Bio-dynamics and imposed physical characteristics	What the person is and what the person does

Table 1.2 Means for Identifying a Person.

1.4.1 Names-Based Identification

The nature of names is tightly bound up within the cultural and legal environment. The discussion in this sub-section is oriented towards those countries whose traditions originate in Great Britain. In English-language cultures, names generally comprise one or more Christian, first, given or fore-names, and a one-word (sometimes two-word or hyphenated) surname. In Britain, a 'Christian name' is that provided at the time of baptism. If the child was not baptised, or no name was given at baptism, then a name is gained by 'repute', i.e. by usage, [Clarke94b]. Whether or not a person has a Christian name, and whether or not he or she uses it, alternative and additional names may be used, without restraint by the law. In short, there is nothing illegal about the use of an alias, 'claim-to-be' or 'also-known-as' (aka), although acts which involve the use of an alias may be.

The contemporary concern with gender equality is not the only demonstration of the flexibility of names, and of the many-to-one relationship between names and individuals. People in security-sensitive positions (such as staff in high security buildings, weapons stores or military in war areas, prisons, and operatives and managers for espionage and counter-espionage agencies) use multiple identities as a means of protecting themselves and their families. Actors, authors and playwrights use multiple identities in order to more readily take advantage of artistic licence.

The effect is that, with minor exceptions, there may be such a thing as 'the legal name of a person', but there is no compulsion to use it, and no prohibition on using any other name or names instead, or as well. Examples of this are seen when the press reports that a person has been arrested and charged with a variety of offences under a variety of names. A person may, in general, use any name he or she wishes to, provide he or she does not thereby attempt to defraud.

As a basis for identification, names lack constancy and reliability. Moreover, many other difficulties arise. The legal and administrative systems of many countries in which anglo-saxon-celtic traditions dominate continue to have difficulty with non-European names, which may, for example:

- have different sequences (e.g. family name may appear first);
- have additional components in the name (e.g. a name of religious rather than identificatory significance);
- be incomplete (e.g. there may be no family name);
- be assigned in unfamiliar ways (e.g. the surname may come from the matriarchal rather than patriarchal line;
- leap-frogging generations (e.g. as in many in African and Arab countries);
- change in ways or at times foreign to local traditions (e.g. at puberty); and
- is variable depending on the context (e.g. by omitting the religious component)?

In some cultures, a relatively small number of first names are prevalent (e.g. John and James in Anglo-Saxon communities), and in others a small number of surnames may dominate (e.g. Kim in Korea, Ng and Nguyen in Vietnam, and Singh universally among Sikhs) [Clarke94a].

A further challenge arises from misspellings and variations. In some cases these are initiated or enforced by organisations with whom the person deals, which may have difficulties with transliteration from a non-Roman script, or from a Roman script which uses diacritics such as ç, ü and ø, or in handling long series of consonants (such as "wrysczwicz"). Various techniques are used by organisations to cope with these difficulties, such as 'phoenix' algorithms to deal with homophone (like-sounding) names. Specialised software packages combine these techniques to assist large organisations in matching new transactions with existing data.

In order to cope with these uncertainties, it is common to use further data as additional elements of the identification, or as confirmatory data. Most common among these are date-of-birth (which suffers from being, for some people, particularly sensitive), and address. Address is also sensitive for some people, and is volatile. In developed countries, it is not uncommon for a number of the population to be at a different address from where they were one year earlier. In short, names-based IDs are a challenging, weak, and risky foundation on which to build a reliable H-ID for an organisation's or individuals identification system.

1.4.2 Codes-Based Identification

To cope with the weakness of the name-based ID, it is common for organisations to create coding schemes. Code-based IDs are commonly based on a set of digits, but may incorporate alphabetic characters. A major reason why code-based IDs are of value to organisations is that their issue can be controlled, and hence the uniqueness of the code assured. It is common to not only assign a code to each person with whom the organisation deals, but also to request him or her to remember it. Code-based ID is much related to both knowledge-based and token-based IDs (see sections 1.4.3 and 1.4.4, respectively). Code-based IDs are issued with a tokens-based ID bearing the code, and request them to have the token with them when they conduct transactions with the organisation. It is possible to build some degree of internal and external validity checking into a code-based ID. For example, it can carry a check-digit, such that, at the point of data capture, a simple computation will generally detect an invalid code. Or it may include, in clear or in hidden form, some characteristic of the person, such as their year of birth, gender, location of residence, or pension status, enabling alert users to detect possible errors, or fraudulent or criminal behaviour.

Code-based ID may be devised in such a manner as to be readily human-readable, machine-readable, or both. Bar-coding is a currently popular means of recording codes on products and packages in such a way that devices can read the code, and card-borne magnetic stripes and memory-chips can also contain identification codes [Harry83].

The "European Article Numbering Association" (EAN) was created in 1977 as a non-profit body to develop a Unified Product Code (UPC) system [Muenz99]. The system originated in USA and was established by the Uniform Code Council (UCC) in 1973. Over the last 20 years the EAN.UCC system has been established as a world-wide system for identification of products, services, transport units, assets and locations [Chaum85]. The EAN.UCC application identifier is an element string represented in bar code symbols endorsed by EAN International and UCC. Identification numbers are printed using a 12-digit code. The system provides a means by which a common language is used to accurately and speedily communicate trading information across the supply chain to any industry in any part of the world. Basically the EAN.UCC System provides the following standards:

- Unambiguous identification of products, services, transport units and locations,
- Representation of supplementary information,
- Methods of data capture by which information can be represented in a format that can be automatically captured into the computerised system.

1.4.3 Knowledge-Based Identification

People may be recognised by demonstrating that they are in possession of information which only that person would be expected to know. Examples of data used for knowledge-based IDs are one's family and given names, prior names, father's name, mother's name, mother's and grandmothers' maiden names, date and place of birth, address, marital status, religion, occupation, and others. Identification on the basis of such information suffers from the problems that some people just do not know (e.g. orphans and refugees), some forget, and any such information is or can be known by other people. It results in many error of commission and omission, and of high False-Accept-Rates (FAR) and False-Reject-Rates (FRR).

Passwords are a very common application of knowledge-based identification. Another is the Personal Identification Numbers (PINs) used in conjunction with Automatic Teller Machines (ATM) and Computer Mediated Communications (CMC). Imposed passwords and PINs are not easily memorised and are easily forgotten. As a result, a significant proportion of the population records them on or near the card whose unauthorised use it is supposed to prevent, reducing the effectiveness of the identification scheme. User-nominated passwords and PINs may be more readily guessed by an impostor, but less likely to be stored adjacent to the card or terminal. Knowledge-based approaches to personal identification seldom provide organisations with an adequate basis for the operation of their information systems. They can have some value as a means of secondary, correlative confirmation of identity, particularly in relatively low-security contexts, but still with high risks and a lot of frauds have been reported.

1.4.4 Tokens-Based Identification

A token is something, which a person has in his or her possession, in particular documentary evidence. Commonly used documents are birth and ID cards, passport, driver's licence (or in most states of the U.S., non-driver's licence), employer-issued building security card, credit card, club membership card, statutory declaration, keys, affidavit, or letter of introduction. The passport is a particular token, which is popularly regarded as being especially reliable.

Generally, a high-integrity identity is constructed by accumulating a variety of low-integrity evidence [Eaton86]. People who are responsible for nominally high-integrity schemes accept accumulations of data as being sufficient evidence, especially for people whose (apparent) background suggests that they will have difficulty producing 'harder' evidence. Token-based schemes are of value in tightly controlled environments, as a variant on the 'turnaround document' approach [Clarke94a]: the person first presents at a counter, then must wait in an anonymous area prior to visiting the counter a second time. If an identifier is issued on the first occasion, and interchange or theft of the identifier is unlikely, then its presentation on the second occasion will be fairly reliable 'proof of identity' within that limited context. Generally, however, tokens-based schemes alone are of limited integrity. To overcome the deficiencies, it is necessary to have adequate means of associating the token with the person to whom it relates, and of detecting attempts by others to use fraudulently.

1.4.5 Biometrics-Based Identification

The term 'biometrics' is used to refer to any of a variety of identification techniques, which are based on some physical and difficult-to-alienate characteristic. They are sometimes referred to as 'positive personal identification', because they are claimed to provide greater confidence that the identification is accurate. Among many other instances in the animal world, mice and penguins are capable of using their olfactory senses, aided to some extent by other cues, to very reliably recognise their parents and their progeny, even among large populations packed into a small space. Humans with such capabilities appear to be limited to those whose other senses are severely impaired, such as the blind and deaf [Chaum85]. Hence biometric techniques involve 'metrics' or measurements of some kind, rather than depending merely on informal or subliminal methods.

Class Scheme	Relative Examples of Biometric Techniques
Appearance	The familiar passport descriptions of height, weight, colour of skin, hair and eyes, visible physical markings; gender; race; facial mark and hair style, wearing of glasses; supported by photographs
Social behaviour	Habituated body-signals; general voice characteristics; style of speech and gait; supported by video-film
Bio-dynamics	The manner in which one's signature is written; statistically-analysed voice characteristics; keystroke dynamics, particularly in relation to login-id and password
Natural physiography	Skull measurements; teeth and skeletal structures; thumbprint, fingerprint sets and handprints; retinal and iris scans; earlobe capillary patterns; hand geometry; DNA-patterns
Imposed physical characteristics	Dog-tags, collars, bracelets and anklets; brands and bar-codes; embedded micro-chips and transponders

Table 1.3. Taxonomy of Biometric Techniques.

Table 1.3 presents a classification scheme and examples of biometric techniques. Many of these features change naturally over time, such as hair colour, gait, speech, height and weight. Natural changes can be enhanced or retarded by such means as tinting, platform shoes and dieting. Some are fairly readily changed, whether for personal whim, or as an explicit attempt to change appearance; for example, contact lenses change eye-colour, the shape of glass-frames has the effect of changing the shape of the face. And changed facial hairstyles (including beard, moustache, sideburns, eyebrows and eyelashes) make physical identification quite difficult. Gross devices like wigs, body-padding and cosmetic surgery can be of assistance in the endeavour to deceive, but frequently the pattern adopted by the scores of facial muscles is more effective. Most such attempts to change appearance are an endeavour to communicate, perhaps even to achieve, change in some aspects of identity. Although intentionally 'deceptive', only a very small proportion would be regarded as 'acts of deceit', or are undertaken in order to affect criminal fraud.

Photographs, although not originally intended for the purpose, can be used as evidence of identity. Photographs provide a gross representation of part of a person's physiognomy or facial features, at a particular point in time, and under particular lighting conditions. Depending on their size, the fineness of grain, and the precision of the reproduction media and on whether it is in monochrome or colour, a photograph may be a more or less faithful representation, in two dimensions, of a historical three dimensional reality. It may therefore be a more or less fair guide to the past appearance of a person from a particular perspective, under particular conditions. For the reasons discussed in the previous paragraphs, a photograph is a highly unreliable means of recognising a person at a later time, particularly if the person is seeking to avoid being recognised. The internal passport scheme in the F.B.I. attempted to surmount this obstacle by containing three photographs taken at three different ages or with at least the right ear clearly visible in the photograph so as to be integrated with ear shape [FBI84]. All such relatively informal appearance, bio-dynamics, and behaviour-based schemes can assist in detecting an impostor, but are not reliable for that purpose. They are of limited use in confirming that the person presenting is the right one.

In addition to gross features, others of a far finer nature enable more precise identification. Thumbprints, individual fingerprints and sets of fingerprints have been used since the end of the nineteenth century in matters of a criminal nature [Clarke94a]. In relatively free countries, the situation generally is that there is no authority for the compulsory provision of fingerprints, unless they are being charged with a criminal offence; and there is no authority for the prints to be retained unless the charge is pursued and the offence proven. A few countries, however, apply fingerprinting in other areas such as immigration matters.

1.5 An Overview of Biometrics

Theoretically, any human physiological, dynamic, physical or behavioural characteristic can be treated as a biometric feature to make a personal identification as long as it satisfies the following requirements:

(i) Universality, which means that every person should have the characteristic;

(ii) Uniqueness, which indicates that no two persons should be the same in terms of the characteristic;

(iii) Permanence, which means the characteristic should be invariant with time; and

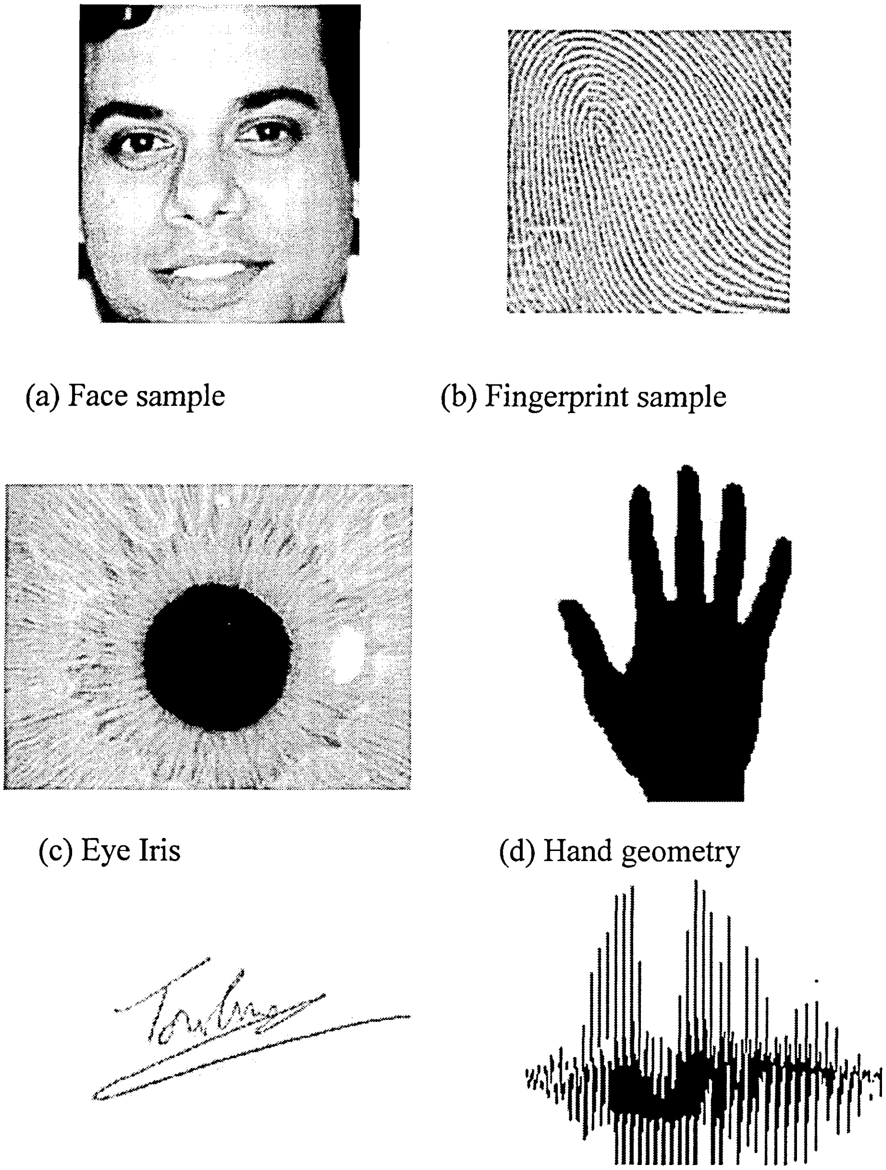
(iv) Collectability, which indicates that the characteristic can be measured quantitatively.

In practice, there are some other important requirements such as:

(i) Performance, which refers to the achievable identification accuracy, the resource requirements to achieve acceptable identification accuracy, and the working environmental factors that affect the identification accuracy.

(ii) Acceptability, which indicates to what extent people are willing to accept the biometric system.

(iii) Circumvention, which refers to how easy it is to overcome the system by fraudulent techniques.



(a) Face sample

(b) Fingerprint sample

(c) Eye Iris

(d) Hand geometry

(e) Signature sample

(f) Voiceprint sample

Figure: 1.1. Samples of Biometrics.

In this section, we review some of the most common biometrics including fingerprint, hand-geometry, face, iris, retina, signature, ear, speech, gait, DNA, and keystrokes. Most of the biometrics are visible especially the physical characteristics. However, some of them are not easy to present, such as a gait and body odour. Figure 1.1 shows examples of the most common biometric samples.

1.5.1 Fingerprint Identification

The fingerprint image is made of foreground ridges, which are separated by background valleys. Ridge flow direction forms different patterns like arches, loops, whorls, and also gives rise to various minutiae like ridge endings, ridge bifurcation's, cores, and deltas. Both foreground and background consist of a similar set of minutiae, the tiny patterns used for fingerprint identification. Each individual has a unique fingerprint and the uniqueness of a fingerprint, is exclusively determined by the local ridge characteristics and their relationship. These local ridge characteristics are not evenly distributed. Most of them depend heavily on the impression conditions and quality of fingerprints and are possibly observed in fingerprints.

Fingerprint identification (F-ID) is a highly specialised biometric system that compares a single finger image with a database of finger images. F-ID is predominantly used for law enforcement, but is also being put to use in civil applications [Jain99a]. For law enforcement, finger images are collected from crime scenes, known as latent, or are taken from criminal suspects when they are arrested. In civilian applications finger images may be captured by placing a finger on a scanner or by electronically scanning inked impressions on paper. Fingerprint recognition technology is already a key player in information security devices, and Police departments have long been interested in the improvement of F-ID methods as an important factor in making more effective the administration of criminal justice and security. Although, fingerprint-matching systems are usually associated with criminal and forensic identification, they have become more popular in civilian and commercial applications, such as, access control, high-security areas in organisations and financial security. These applications have been conceived due to the fingerprint's favourable characteristics such as, unchangeability and uniqueness in an individual's lifetime. Additionally, the most noteworthy characteristic of fingerprint biometrics is the template size, which is typically under 500 bytes [Baldi93]. The adaptability of fingerprint recognition hardware to the computer keyboard and mouse make it a viable alternative to the workstation's password. New optical components like moulded lens and single chip fingerprint imaging devices have lowered hardware costs from the thousands dollars per unit to fewer than fifty dollars [Ammar96].

1.5.2 Face Recognition

The face is one of the most acceptable biometric because it is one of the most common methods of identification which humans use in their visual interactions. In addition, the method of acquiring face images is non-intrusive. Two primary approaches to the identification based on face recognition are the following [Jain99a]:

(i) Transformation approach: the universe of the face image domain is represented using a set of orthonormal basis vectors.

(ii) Attribute-based approach: facial attributes like nose, eyes, mouth, etc. are extracted from the face image and the invariance of geometric properties among the face landmark features is used for recognising features.

In summary, facial recognition is a very interesting biometric technique, which will no doubt continue to be developed in coming years as it does indeed offer potential for certain applications that could not be easily matched by other means. Its application does, however, requires a little more attention than with other biometrics and would probably be well suited to bespoke situations, which can be carefully designed to accommodate the special requirements of this technique [Hong98].

1.5.3 Iris Recognition

The iris is composed of elastic connective tissue, known as the trabecular meshwork. It consists of pectinate ligaments adhering into a tangled mesh revealing striations, ciliary processes, crypts, rings, furrows, a corona, sometimes freckles, vasculature, and other features. During the first year of life, a blanket of chromatophore cells usually changes the colour of the iris, but the available clinical evidence indicates that the trabecular pattern itself is stable throughout a lifespan [Adler97]. This protected internal organ, which can be imaged adequately at a distance up to a meter, reveals a number of independent degrees-of-freedom of textural variation across individuals. The iris has in excess of 250 characteristics that are unique to each person, which is more than ten times the number of identifiers carried by a fingerprint [Jain97]. Being an internal organ of the eye, the iris is immune (unlike fingerprint) to environmental influences, except for its papillary response to light.

1.5.4 Voice Recognition

Voice recognition is most often deployed in environments where the voice is already captured, such as telephony and call centres. If users become accustomed to speaking to their PC, especially in speech-to-text applications, voice-scan may also become a solution for PC and web access. The voice is a characteristic of an individual however, it is not expected to be sufficiently unique to permit identification of an individual from a large database of identities. Moreover, a voice signal available for authentication is typically degraded in quality by the microphone, communication channel, and digitizer characteristics. Voice capture is unobtrusive and voiceprint is an acceptable biometric in most societies [Kasabov00]. However, age, health, and emotions affect voice characteristics. In addition, the enrolment procedure has often been more complicated than with other biometrics leading to the perception of voice verification as unfriendly in some quarters. Hence voice-based ID is risky and easy to forge.

1.5.5 Signature Verification

Signature has long been accepted as a legitimate means of authentication. The way a person signs his/her name is known to be a characteristic of that individual. Although, signatures require contact and effort with the writing instrument, they seem to be acceptable in many government, legal, and commercial transactions as a method of personal authentication. Signatures are a behavioural biometric, evolve over a period of time and are influenced by physical and emotional conditions of the signatories. There are two approaches to signature recognition verification namely static and dynamic [Julian00]. In static signature verification, only shape (geometric) features of the signature are used for authentication and identity. Typically, the signature impressions are normalised to a known size and decomposed into simple components (strokes). The shapes and relationships of strokes are used as features. In dynamic signature verification, not only the shape features are used for authenticating the signature but also dynamic features like acceleration, velocity, and trajectory profiles of the signature are employed.

1.5.6 Other Biometrics

A number of other biometrics-based technologies are available and being developed in the educational and commercial research laboratories world-wide. These include the following:

1. DNA: (DeoxyriboNucleic Acid) is the one-dimensional unique code for one's individuality, except for the fact that identical twins have identical DNA pattern. It is, however, currently used mostly in the context of forensic applications of identification [Jain00]. Three issues limit the utility of this biometric for other applications:

- Contamination and sensitivity: it is easy to steal a piece of DNA from an unsuspecting subject to be subsequently abused for an ulterior purpose;
- Real-time applicability: the present technology for genetic code matching is not geared for online unobtrusive identifications.
- Privacy issues: information about susceptibilities of a person to certain diseases could be gained from the DNA pattern and there is a concern that the unintended abuse of genetic code information may result in discrimination.

2. Body Odour: It is known that each object exudes an odour that is characteristic of its chemical composition and could be used for distinguishing various objects. The feature vector consists of the odour comprising of the normalised measurements from each sensor. After each act of sensing, the sensors need to be initialised by a flux of clean air. Body odour serves several functions in the animal world including communication, attracting mates, assertion of territorial rights, and protection from a predator [Jain99a]. A component of the odour emitted by a human (or any animal) body is distinctive to a particular individual. It is not clear if the invariance in a body odour could be detected. Despite that chemical odour-based identity authentication systems exist [Howard01].

3. *Ear Shape:* In ear the structure, the cartilaginous tissues of the pinna are distinctive. The features of an ear are not expected to be unique to each individual [Jain99a]. The ear recognition approaches are based on matching vectors of distances of salient points, on the pinna from a landmark location on the ear. No commercial systems are available yet and authentication of individual identity based on ear recognition is still a research topic.

4. *Facial Thermogram:* The facial thermogram is an image taken with an infrared camera that shows the heat patterns of the face. These images are unique, and combined with highly sophisticated pattern matching algorithms that check for relative temperature differences across the face, means that this technique is unaffected by age, health, or even the temperature of the body. With 19,000 data points it's extremely accurate and will distinguish identical twins in the dark [Julian00]. The development of this technology continues in the direction of improving cost effectiveness in order to increase its applicability to wider range identification and verification applications. The facial thermogram offers the promise of providing accurate, effective and highly secure identification technology, once costs are reduced, and if there is a flexible method of acquisition.

5. *Gait:* Gait is the peculiar way one walks and is a complex spatio-temporal behavioural biometric. Gait is not supposed to be unique to each individual, but is

sufficiently characteristic to allow identity authentication. Gait is a behavioural biometric and may not stay invariant especially over a large period of time, due to large fluctuations of body weight, major shift in the body weight (e.g. waddling gait during pregnancy, major injuries involving joints or brain (e.g., cerebella lesions in Parkinson disease, or due to inebriety) [Jain99a]. Humans are quite adept at recognising a person at a distance from his gait. Although, the characteristic gait of a human walk has been well researched to detect abnormalities in lower extremity joints, the use of gait for identification purposes is very recent.

6. *Hand geometry:* Hand geometry systems work by taking three-dimensional views of the hand in order to determine the geometry and metrics around finger length, height and other details. There exists a wealth of information within the geometry of an individual hand and in fact, the leading hand geometry device measures around 90 such parameters [Julian00]. A successful trial with the hand geometry units was conducted at the Federal prison in Jesup, Georgia, and the US Federal Bureau of Prisons [Clarke94b] is using hand geometry units to monitor the movements of prisoners, staff, and visitors within certain prisons.

7. *Keystroke Dynamics:* the keystroke is a behavioural biometric for some individuals, and one may expect to observe large variations from typical typing patterns. Keystroke dynamic features are based on time duration between the keystrokes. Some variants of identity authentication use features based on inter-key delays as well as dwell times (how long a person holds down a key) [Armstrong02]. Some commercial systems are already appearing in the market.

8. *Retina:* The retina scan offers the promise of extremely accurate identification, but is invasive and currently requires the head to stabilise so that a light can be directed to the back of the retina. The principle behind retina scanning is that the blood vessels at the retina provide a unique pattern, which may be used for H-ID. The relative uniqueness of retinal vascular patterns was discovered in 1935 when doctors studying eye diseases found that the patterns were both intricate and stable [Davies94].

9. *Vein tree (Hand vein):* Vein pattern scanning is an idea whereby the veins in the back of the hand and wrist are scanned while the user grips a bar within the reading device. A related technology using near-infrared imaging is the facial thermogram technology [Julian00]. Identification based on hand veins is an infrared image of the back of a clenched human fist. The structure of the vasculature could be used for identification, but at the present there are many questions, which remain unanswered with this technique especially around relative uniqueness and capturing an individual's pattern.

1.5.7 Biometrics Selection

Whichever biometric technique will eventually gain performance in identification or authentication will depend on a number of factors, including technological improvements that provide for performance, cost, universality, and consumer acceptance. Mapping the application requirements with all the attributes of the specific biometric will determine the optimal biometric technology. No single biometric technology, or single security device for that matter, can deliver guaranteed 100 percent security. Most customers that purchase security identification devices want the best possible security solution that is affordable, easy to implement, yet is unilaterally accepted by the intended user population. The answer to these needs will most likely be combination of techniques to include multiple biometrics, which may affect the ease of implementation. Growth in the biometrics research and industry has led to an ever-increasing number of vendor products available to the prospective buyer [Jain00]. As with most new and emerging technologies, there are many small vendors, many product claims, providing an atmosphere that is difficult for customers to make strong differentiation amongst products.

The public may perceive biometric information as confidential, much like a social security number. The use of biometric information as a means of access control and/or non-repudiation purposes is likely to enter the privacy issues debate. To choose the right approach to biometric identification or authentication, implementers must understand the application, the user-base and the characteristics of the biometric device itself. One also must consider the conditions under which it will be used and how fallback authentication methods, such as knowledge or tokens-based, will be instituted when biometrics are not available. There are some factors to consider before choosing a biometric system, which fall into multiple levels of requirements that can be broken down into two classes, namely, verification and identification. An authentication/verification (i.e. matching the live fingerprint with the template stored on the token (smartcard-ID) typically requires a simpler algorithm and has better performance than identification (i.e. matching the given biometric with the database of stored information).

Table 1.4 presents a comparison of different biometrics from a previous study [Jain99a]. It is clear from the table that there is no single "best" biometric technology. Different identification and authentication environments require different biometric solutions.

	Characteristics						
Biometrics	Universality	Uniqueness	Permanence	Collectability	Performance	Acceptability	Circumvention
Face	High	Low	Medium	High	Low	High	Low
Fingerprint	Medium	High	High	Medium	High	Medium	High
Hand geometry	Medium	Medium	Medium	High	Medium	Medium	Medium
Keystrokes	Low	Low	Low	Medium	Low	Medium	Medium
HandVein	Medium	Medium	Medium	Medium	Low	Medium	High
Iris	High	High	High	Medium	High	Low	High
Retinal	High	High	Medium	Low	High	Low	High
Signature	Low	Low	Low	High	Low	High	Low
Voice	Medium	Low	Low	Medium	Low	High	Low
Facial thermogram	High	High	Low	High	Medium	High	High
Odour	High	High	High	Low	Low	Medium	Low
DNA	High	Medium	High	Low	High	Low	Low
Gait	Medium	Low	Low	High	Low	High	Medium
Ear	Medium	Medium	High	Medium	Medium	High	Medium

Table: 1.4 Comparisons of Biometric Technologies.

There are technologies, which provide high accuracy, but they may be expensive or difficult to use. There are technologies, which require almost no effort to use but they may be unable to provide a high enough level of accuracy. There are technologies, which are easily integrated into existing infrastructures but they may not perform quickly enough to deploy in many situations. There are technologies, which complement current authentication methods but they may incorrectly reject a large number of users. Hence, no single technology is ideally suited for all applications.

Luckily, there is a biometric technology (or combination of technologies) to suit nearly every authentication scenario, from point-of-sale to high-security installations, to e-commerce. There are five steps to selecting a biometric system.

1. The total time taken for enrolling a person in the application;
2. The total time taken by an individual to successfully use the application for a transaction;
3. The FRR of the application;
4. The FAR of the application;
5. The performance of the application across the population.

Vendors use FAR and FRR to rate biometric accuracy. Both measures focus on the system's ability to allow limited entry to authorised users. However, these measures can vary significantly, depending on how you adjust the sensitivity of the mechanism that matches the biometric. For example, a tighter match between the measurements and the user's templates increases the sensitivity, which will decrease the FAR, but at the same time can increase the FRR.

When assessed against the desirable characteristics, most natural physiological identifiers are universal and indispensable, but many are not unique, including facial appearance that can be affected by weight, colour of eyes and skin. Fingerprint techniques, however, have advantages over the other biometrics as shown in Table 1.4. However, some factors may affect fingerprint verification. Generally speaking, Caucasians have the easiest prints to read, with print becoming less defined in other races. Women also tend to have more fingerprint details than men, however, women typically have a smaller print area which makes alignment more critical. Occupation also does have a significant impact. For white-collar workers, the problem is not so acute. For manual workers, the system designer might consider using a fingerprint reader using ultrasound, which looks beneath the skin surface for fingerprint details. Age will affect the fingerprint recognition systems and will induce higher errors in the system. For example, ridge structure will change in later ages. A combination of robust software, a good classification and matching algorithm should minimise these problems.

1.5.8 Multi-Biometrics:

User verification systems that use a single biometric indicator often have to contend with noisy sensor data, restricted degrees of freedom and unacceptable error rates. Attempting to improve the performance of an individual matcher in such situations may not prove to be effective because of these inherent problems. Multimodal biometric systems seek to alleviate some of these drawbacks by providing multiple evidences of the same identity. These systems also help achieve an increase in performance that may not be possible by using a single biometric indicator. Cases of biometrics integration have been reported [Hong98], [Jain01]. In this work, the authors report a multimodal biometric system, which integrates face recognition, fingerprint verification, and speaker verification in making personal identification. The system takes advantage of the capabilities of each individual biometric, to overcome some of the limitations of a single biometric. Preliminary experimental results demonstrate that the identity established by such an integrated system is more reliable than the identity established by a face recognition system, a fingerprint verification system, and a speaker verification system. Identification based on multiple biometrics represents an emerging trend.

1.6 History of Fingerprints

Humans have discovered the uniqueness of the fingerprint and used fingerprints in different ways for a very long time. Modern fingerprint techniques were initiated in the late 18th century. Fingerprint evidence was mentioned as early as the 5th century from God's revelation in a verse in the Quran [Quran592]: "Yes, we are able to create (to put together) in perfect order the very tips of his fingers". The meaning as interpreted in the version of the Quran V.75: 4, in [Khan85], [Ibrahim01] is each human being having his or her own special fingerprints and not resembling anyone else. Modern scientific studies have shown that the fingerprints are completely unique and there's no similarity between any two fingerprints even among the same person's fingerprints and the same hand's fingerprints [Jain99b], and [Battley37].

In 1683, Marcello Malpighi, a professor of anatomy at University of Bologna noted in his treaties, ridges, spirals and loops in fingerprints [FBI01]. He made no mention of their value as a tool for individual identification. A layer of skin was named after him; "Malpighi" layers. In 1684, English plant morphologist Nehemiah Grew published a paper reporting his systematic study on the ridge, furrow, and pore structure in fingerprints, which is believed to be the first scientific paper on fingerprints [Battley37]. Since then, a number of researchers have invested huge amounts of efforts studying fingerprints. In 1788, Mayer Bill made a detailed description of the anatomical formations of fingerprints in which a number of fingerprint ridge characteristics were identified. In1809, Thomas Bewick began to use his fingerprint as trademark, which is believed to be one of the most important contributions in the early scientific study of fingerprint identification [Battley37].

There are a number of activities during 18th century including Henry and Galton's systems [Battley37]. In 1823, John Evangelist Purkinje, a professor of anatomy at the

University of Breslau proposed the first classification scheme, which classified fingerprints into nine categories according to the ridge configurations [USDJ74]. In 1856, Sir William Hershel, Chief Magistrate of the Hooghly districts in Jungipoor, India first used fingerprints on native contracts on a whim, and with no thought toward personal identification. As his fingerprint collection grew, however, Herschel began to note that the inked impressions could, indeed, prove or disprove identity. While his experience with fingerprinting was admittedly limited, Herschel's private conviction that all fingerprints were unique to the individual, as well as permanent throughout that individual's life inspired him to expand their use. The native was suitably impressed, and Herschel made a habit of requiring palm prints and later, simply the prints of the right Index and Middle fingers on every contract made with the locals. During the 1870's, Henry Fauld, first scientifically suggested the individuality and uniqueness of fingerprints, after noticing finger marks on specimens of "prehistoric" pottery.

In 1880, Fauld forwarded an explanation of his classification system and a sample of the forms he had designed for recorded inked impressions to Sir Charles Darwin. Darwin, in advanced age and ill health, informed Dr. Faulds that he could be of no assistance to him, but promised to pass the materials on to his cousin, Francis Galton. Dr. Faulds published an article in the journal, "Nature" [Battley37] in which he discussed fingerprint as a means of personal identification, and the use of printer's ink as a method for obtaining such fingerprints. He is also credited with the first fingerprint identification of a greasy fingerprint left on an alcohol bottle. In 1882, Gilbert Thompson of the United States Geological Survey in New Mexico used his own fingerprints on a document to prevent forgery. This is the first known use of fingerprints in the U.S., although at the same time, Herschel asserted that he had used fingerprints for about 20 years [FBI01]. In the late 19th century, Sir Francis Galton conducted an extensive study of fingerprints. He introduced the minutiae features for single fingerprint classification in 1888. Galton's primary interest in fingerprints was as an aid in determining heredity and racial background. While he soon discovered that fingerprints offered no firm clues to an individual's intelligence or genetic history, he was able to scientifically prove what Herschel and Faulds already suspected: that fingerprints do not change over the course of an individual's life time, and that no two fingerprints are exactly the same. According to his calculations, the odds of two individual fingerprints being the same were 1 in 64 billion. Galton identified the characteristics by which fingerprints can be identified. These same characteristics (minutiae) are basically, still in use today, which are often referred to as Galton's details, and established the individuality and permanence of fingerprints. And in 1892, Galton published his book, "Fingerprints", establishing the individuality and permanence of fingerprints. The book included the first classification system for fingerprints. In 1891, Juan Vucetich, an Argentine Police official, began the first fingerprint files based on Galton pattern types. At first, Vucetich included the Bertillon system with the files. In 1892, Vucetich made the first criminal fingerprint identification. He was able to identify a woman by the name of Rojas, who had murdered her two sons, and cut her own throat in an attempt to place blame on another. Her bloody print was left on a door-post, proving her identity as the murderer. An important advance in fingerprint identification was made in 1899 by Edward Henry, who (actually his two assistants from India) established the famous Henry's system of fingerprint classification [Battley37]. This was an elaborate method of indexing fingerprints very much tuned to facilitating the human experts performing fingerprint identification.

By early 20th century, the formations of fingerprints were well understood. The biological principles of fingerprints are summarised as follows: (i) Individual epidermal ridges and furrows (valleys) have different characteristics for different fingers; (ii) The configuration types are individually variable, but they vary within limits, which allows for systematic classification; (iii) The configurations and minutiae details of individual ridges and furrows are permanent and unchanging for a given finger [Jain01]. In the late 20th century, fingerprint identification was formally accepted as a valid personal identification method by law enforcement agencies and became a standard routine in forensics area [FBI01]. Fingerprint identification agencies were set-up world-wide and criminal fingerprint databases were established [Hong97]. In 1901, the use of fingerprints for criminal identification in England and Wales was introduced, using Galton's observations and revised by Edward Henry. Thus, began the Henry classification system used even today in many places all over the world, with some changes in some countries [Hong99]. During 1901-1903, the first systematic uses of fingerprint in U.S. were introduced by the New York Civil Service Commission for testing, and the New York State Prison System began the first use of fingerprints for criminals. In 1905, saw the use of fingerprints for the U.S. Army. During the next 25 years more and more law enforcement agencies joined in the use of fingerprints as means of personal ID. Many of these agencies began sending copies of their fingerprint cards to the National Bureau of Criminal Identification, established by the International Association of Police Chiefs. Edmond Locard wrote in 1918 that if 12 point's minutiae (Galton's details) were the same between two fingerprints, it would suffice as positive identification. Although, there is no required number of points necessary for identification most countries have set their own standards, which do include a minimum number of points. By 1946, the Federal Berea of Investigation (FBI) had processed 100 million fingerprint cards in manually maintained files; and by 1971, 200 million cards. Starting in early 1960's, FBI, Home Office in UK, and Paris Police Department invested a large amount of effort to develop Automatic Fingerprint Identification Systems (AFIS) [Hong97]. Their efforts were so successful that a large number of AFIS are currently installed and in operation at law enforcement agencies world-wide. With the introduction of AFIS technology, the manually maintained files were split into computerised criminal files and manually maintained civil files. These systems have greatly improved the operational productivity of these agencies and reduced the cost of hiring and training human fingerprint experts for manual fingerprint identification. A number of automatic fingerprint classification, identification, and recognition systems have been implemented since early 1990's including, [Baldi93], [Blue94], [Ammar96], [Maio96], [Lumini99b], [Jain99b], [Zsolt99], [NIST00], [Asker00], [Jain00] and [Prabhakar01].

1.7 Conclusion

This Chapter set out to give a presentation, from literature, of H-ID. It has discussed many conventional personal identification methods, which suffer from a number of drawbacks and are unable to positively identify a person. On the other hand, a typical biometric-based system is also not perfect. Biometric-based ID systems however, have two basic advantages over conventional-based ID methods. Firstly, the person to be identified does not have to use anything for verification except themselves, and secondly the critical variable for identification cannot be lost, copied or passed on. A comparison of the different biometric methods identified that the fingerprint-based is the most reliable biometric compared to the other biometrics using a number of desirable characteristics. Subsequent Chapters will address fingerprint characteristics analysis, feature extraction, and classification.

Chapter 2

Fingerprint Characteristics, Analysis and Recognition

2.1 Introduction

In the literature there are many different ways to describe the fingerprint. According to [Battley37] a fingerprint is the impression made by the papillary ridges and valleys (furrows) on the ends of the fingers and thumbs, or the fingerprints are graphical flow-like ridges present on a human finger [Jain99a]. In general, the formations of the fingerprints depend on the initial conditions of the embryonic mesoderm from which they develop. Fingerprints afford an accurate and reliable means of personal identification, because the ridge arrangement on every finger of every human being is unique and does not alter with growth or age. Fingerprints serve to reveal an individual's true identity despite changes in personal appearance resulting from age,. The practice of utilising fingerprints as a means of identification, referred to as dactyloscopy, is an indispensable aid to modern law enforcement. Any ridged area of the hand or foot may be used as identification. However, finger impressions are preferred to those from other parts of the body because they can be taken with a minimum of time and effort, and the ridges in such impressions form patterns (distinctive outlines or shapes) that can be readily sorted into groups for ease of filing. Early anatomists described the ridges of the fingers, but interest in modern fingerprint identification dates from 1880, describing the uniqueness and permanence of fingerprints [Battley37].

The submission of a fingerprint sample to a recognition system for identification or verification is a crucial issue. The recognition system may allow more than one attempt to identify or verify. Data extracted from a fingerprint sample is used either to build a reference template or to compare against a previously created reference template. Fingerprint recognition is one of the most reliable personal identification/verification methods. However, manual fingerprint recognition is so tedious, time-consuming, and expensive that it is incapable of meeting today's increasing performance requirements. Fingerprint recognition systems are widely used. Because of the large volume of fingerprints and recent advances in computer technology, there has been increasing interest in automatic processing of fingerprints. It plays a very important role in forensic, commercial, and civilian applications such, as criminal identification, access control and credit card verification. This Chapter describes fingerprint pattern types, characteristic features, and introduces a discussion on how to design and implement an automatic fingerprint recognition system.

The rest of this Chapter is organised as follows: Section 2.2 provides discussion and analysis of fingerprint pattern types, their interpretation, characteristics and features. Section 2.3 presents a discussion on the automatic fingerprint recognition process including sample acquisition, enhancement, feature extraction, classification, and matching. Section 2.4 discusses some fingerprint applications. Finally in section 2.5 we draw some conclusions and remarks on this Chapter.

2.2 Fingerprint Pattern Types, Characteristics and Interpretation

2.2.1 Pattern Types

Fingerprint patterns are classified in three ways: by the shapes and contours of individual patterns; by the finger positions of the pattern types; and by the ridges in loops and whorls [Battley37]. The information obtained is incorporated in a concise formula, which expresses the individuals fingerprint classification. Although this coarse classification is not enough to identify a fingerprint uniquely, it is useful in deciding when two fingerprints do not match and to decrease the size of the databases of fingerprints. Firstly, as shown in Figure 2.1 the fingerprint is mainly classified into three main groups. These three groups are subdivided into a number of small groups (about eighteen fingerprints pattern types), as shown in the Figure 2.2. No matter how definite fingerprint rules and pattern definitions are made, there will always be patterns about which there is doubt as to the class they should be given. The main reasons for this are the fact that no two fingerprints are exactly a-like, and differences in the degree of judgement and interpretation.

Generally, fingerprint classification can be defined as putting the given fingerprint images into groups with similar features. The classification can be with a smaller number of pattern types as in Figure 2.1, or with a larger number of pattern types as in Figure 2.2. The number of classes mainly depends on the features, and also the type of methodology used, for example, whether it is manual or automatic classification. Manual fingerprint classification algorithms are very time consuming, and usually not accurate. In manual algorithms, the class is determined after examining the whole sequence of 10 fingerprints, while in the automatic algorithms a single fingerprint is sufficient [Prabhakar01].

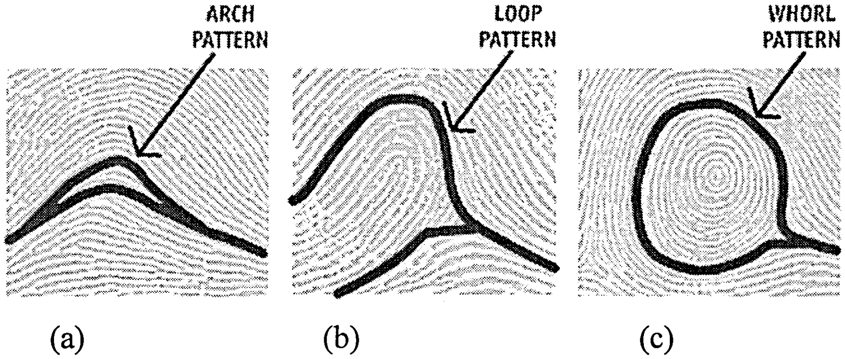


Figure 2.1. Henry system classes of fingerprints

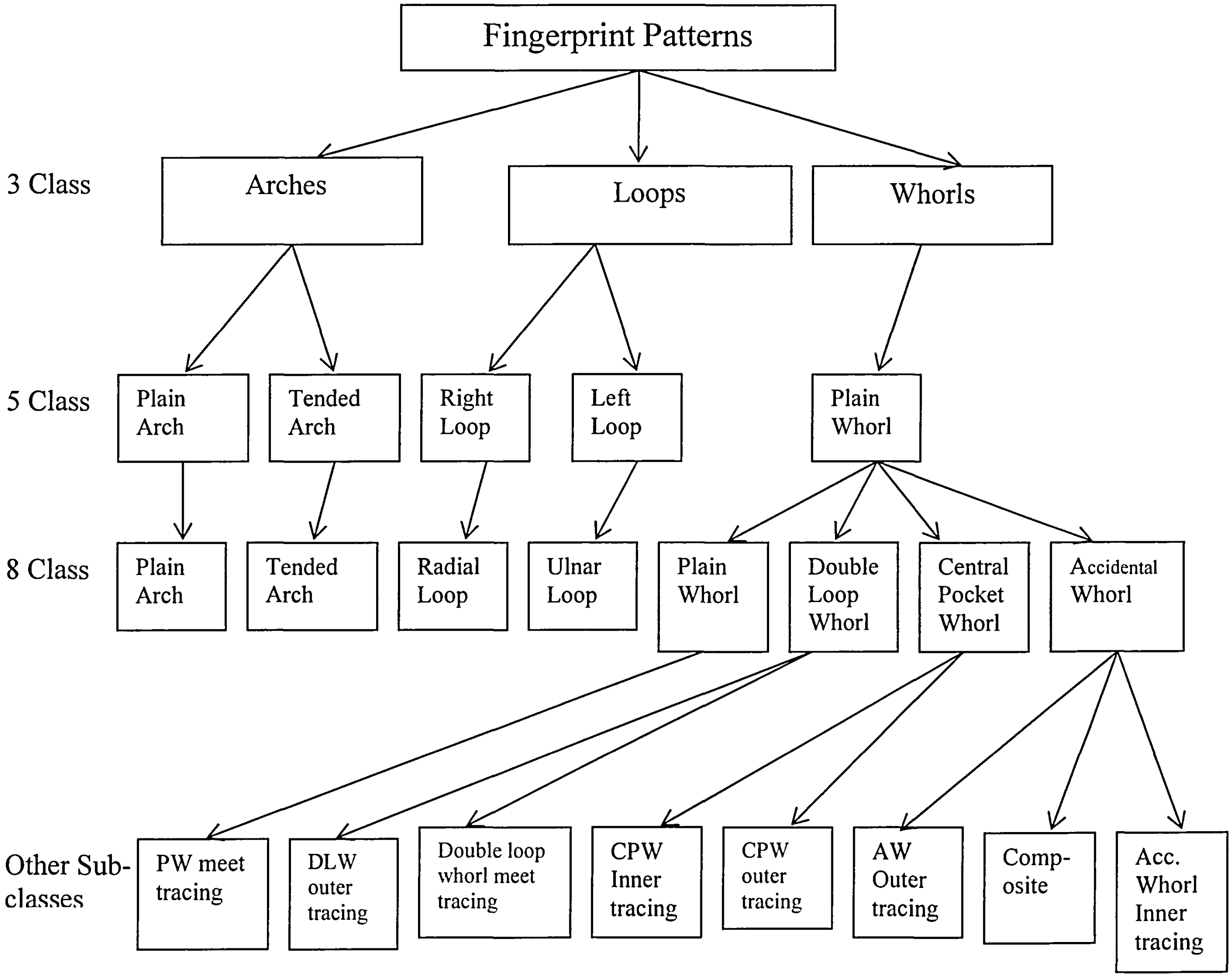


Figure 2.2. A tree-diagram of fingerprint patterns classes.

Moreover, manual systems provide present a smaller number of pattern types, as in case of Henry classification [Battley37], which classifies fingerprints into only three main categories (arch, loop and whorl), as shown in Figure 2.1. The FBI follows the Henry system of classification but recognizes eight different types of fingerprints: radial loop, double loop, ulnar loop, central pocket loop, plain arch, tended arch, plain whorl, and accidental. Due to the small interclass separability of the fingerprint pattern types, it is extremely difficult to design an eight-class classifier with high accuracy. As a result, most automatic systems reduce the number of fingerprint types to a subset of classes defined in the Henry system. This is illustrated in Figure 2.2.

2.2.2 Fingerprint Pattern Interpretation

The result of any kind of fingerprint image processing completely depends on the quality of the given image pattern. The pattern area is the only part of the fingerprint impression with which we are concerned in regard to interpretation, recognition, classification or identification. The following is a summary of fingerprint characteristics.

- **Pattern Area**

Pattern area is the main part of the finger impression and consists of the ridges and all their features. As shown in Figure 2.3 the pattern area presents most of the important features. In the pattern area, there are also so-called type lines, which enclose the loops and whorls. Type lines may be defined as the innermost ridges, which start parallel, diverge, and tend to surround the pattern area [Almansa00b]. There may not be continuous ridges, hence if there is a break in a type line the ridge which is nearest to the type line (on the outside) is considered as a continuation.

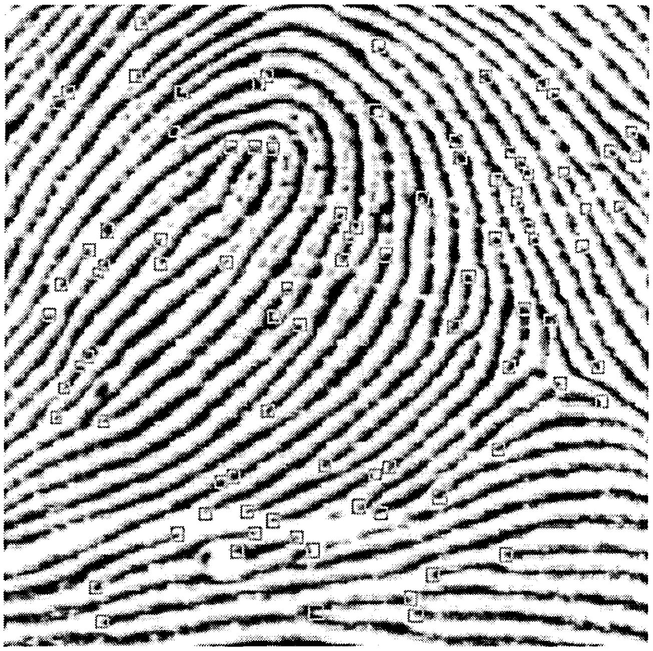


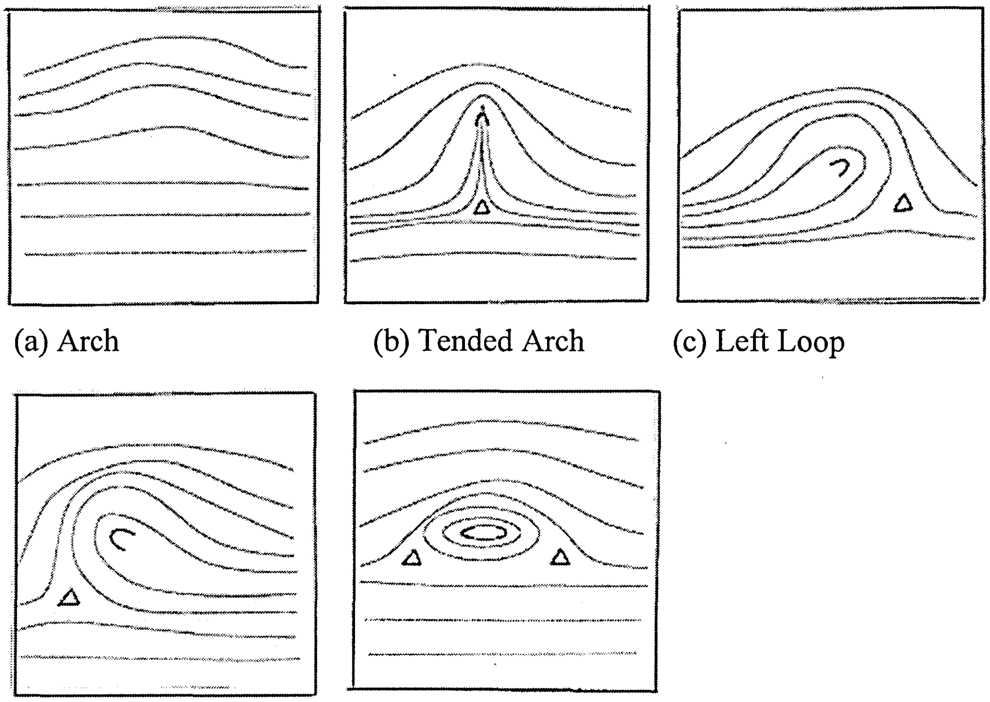
Figure 2.3. A pattern area with different characteristics and features.

- **Arch Patterns**

Plain arches have a mound-like contour, while tended arches have a spike-like or steeple-like appearance in the centre. The plain arch type is the simplest of the fingerprint patterns, and is easily distinguished. In plain arches the ridges enter on one side of the impression and flow or tend to flow with a rise or wave in the centre. While the tended arches most of the ridges enter one side of the impression and flow or tend to flow out upon the other side, as in the plain arch type, however, the ridges at the centre do not wave [Almansa00a]. Skeletons (a) and (b) in Figure 2.4 illustrate arch pattern types.

- **Loop Patterns Types**

Loops have concentric hairpin or staple-shaped ridges and are described as "radial (left)" or "ulnar (right)" to denote their slopes; ulnar loops slope toward the little finger side of the hand, radial loops toward the thumb. A loop pattern must possess several requisites before it may be properly classified as a loop. This type of pattern is the most numerous of all and constitutes about 65 percent of all prints. A loop is a type of fingerprint pattern in which one or more of the ridges enter on either side of the impression, recedes, touch or pass an imaginary line drawn from the delta to the core, and terminate or tend to terminate on or toward the same side of the impression from whence such ridge or ridges entered [Almansa00b]. Figure 2.4 illustrates loop pattern types.



(a) Arch (b) Tended Arch (c) Left Loop

(d) Right Loop (e) Whorl

Figure 2.4 Pattern level skeletons of the five samples

- **Whorl Patterns**

Whorls are usually circular or spiral in shape and occur in 30 percent of all fingerprints. The plain whorl has two deltas and at least one ridge making a complete circuit, which can be spiral, oval, circular or any variant of a circle. An imaginary line drawn between the two deltas must touch or cross at least one of the recurring ridges within the inner pattern area. Figure 2.4 illustrates whorl pattern type.

- **Ridges**

The epidermis (outer skin) is dotted with sweat pores for its entire length and is anchored to the dermas (inner skin) by a double row of peglike protuberances, or papillae. Injuries such as superficial burns, abrasions, or cuts do not affect the ridge structure or alter the dermal papillae, and the original pattern is duplicated in any new skin that grows. An injury that destroys the dermal papillae, however, will permanently obliterate the ridges.

- **Valleys**

Valleys are the white narrow lines between any two ridges, shown in Figure 2.1.

- **Minutiae Points**

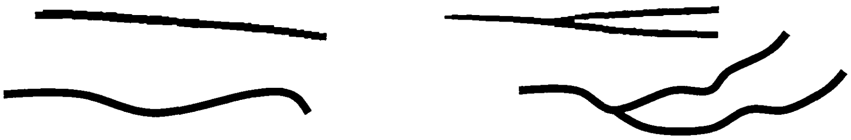
The minutiae points are the basic features of the fingerprint for matching (identification and authentication). The main characteristics of ridges are ridge-endings and bifurcations. There are other fingerprints features, such as lakes, purrs, crossovers, independent ridge points, divergences, and spurs. These features are mainly a combination of the basic features, namely, ridge-ends and bifurcations.

- **Ridge-Ends**

Ridge-Ends are points where a ridge begins or ends abruptly. A ridge may be quite long or very short. It needs to be followed to its end point to find a ridge ending. (See Figure 2.5 (a)).

- **Bifurcations**

Bifurcations are points where a ridge divides into two or more branches, or formed where two separate ridges join. (See Figure 2.5 (b)).



(a) Ridge Ends

(b) Bifurcations

Figure: 2.5. Minutiae points (Ridges Endings and Bifurcations)

- **Lake**

Lake is the joining of two bifurcations, where one forms the left side and the other forms the right side. Alternatively, it could be described as a ridge bifurcating making a small circular or an elliptical shape and then rejoining that ridge. It is also known as an Enclosure. (See Figure 2.6 (a)).

- **Divergence**

Divergence is the spreading apart of two lines, which have been running parallel or nearly parallel. (See Figure 2.6 (b)).

- **Independent Ridge**

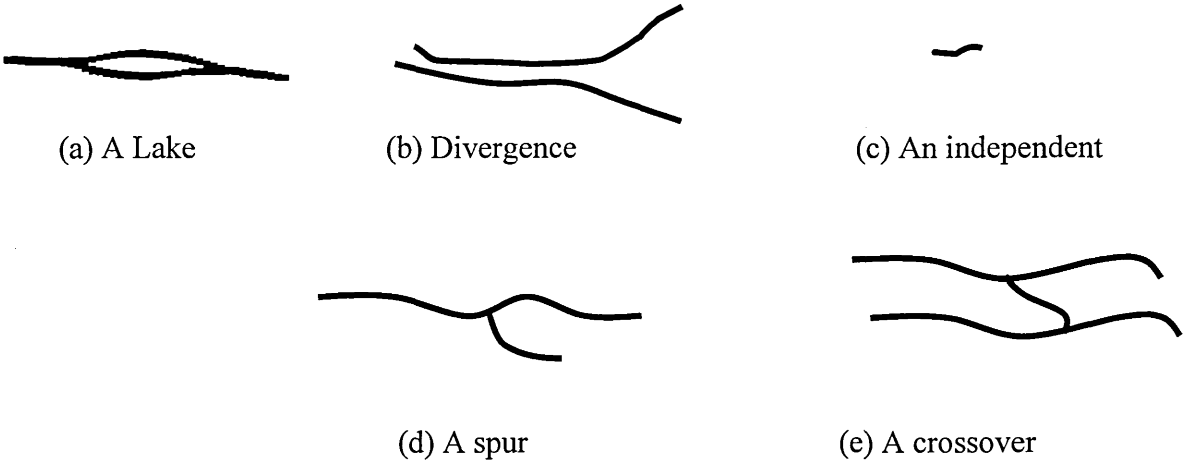
Independent ridge resembles a ridge ending but this is a small ridge, which is separated from the ridge on both sides. (See Figure 2.6 (c)).

- **Spur**

Spur is a combination of an independent ridge and a bifurcation. One of this bifurcation's lines is smaller than the other one. (See Figure 2.7 (d)).

- **Crossover**

Crossover appears as an independent ridge that crosses over a space between two parallel ridges. (See Figure 2.6 (e)).



(a) A Lake

(b) Divergence

(c) An independent

(d) A spur

(e) A crossover

Figure: 2.6. Other features of the fingerprint.

Most automatic fingerprint recognition systems rely on minutiae points, that is, either ridge endings or bifurcations. The other features are essentially a combination of the minutiae points. For instance, as it can been seen in Figure 2.6 (a) Lake and Figure

2.6 (e) Crossover are a combination of two bifurcations, and Figure 2.6 (d) Spur resembles a bifurcation too. However, one of the ridges is shorter than other.

- Singular points

Cores and deltas are known as the singular points. The singular points of fingerprint are very useful in fingerprint classification.

- **Delta points**

Delta points are the points that lie on a ridge nearest the centre of the divergence of type lines. In a fingerprint, the delta point may be one of features of fingerprints opening towards the core point. It is similar to a river delta as shown in the Figure 2.7.

- **Core points**

Core points are the approximate centre of the finger impression where the orientations of the ridges are turned to a semicircle as shown in the Figure 2.7.

In the core and delta areas of the fingerprint, the other features are clearly visible. More details of singular points are discussed in Chapter 5.

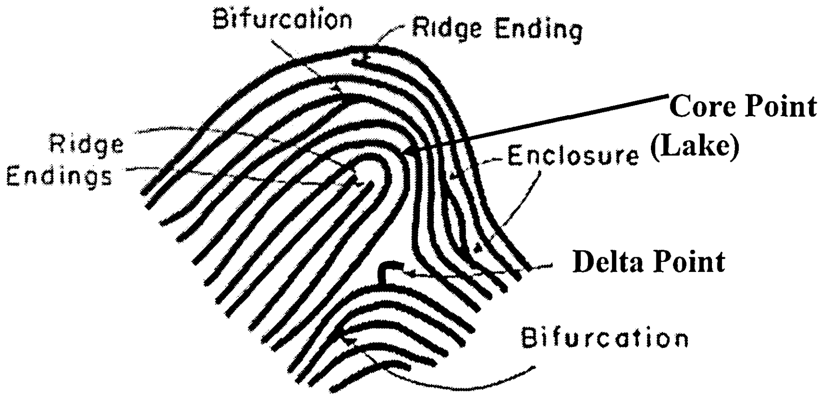


Figure: 2.7. Core and delta are with other different features.

2.3 Overview of Automatic Fingerprint Processing

An Automatic Fingerprint Recognition System (AFRS) is concerned with some or all of the following issues:

- Fingerprint Acquisition: How to capture fingerprint images and how to present them in a proper format,
- Fingerprint Enhancement: To improve the quality of fingerprint images,
- Fingerprint Feature Extraction, for classification or matching,
- Fingerprint Classification: To assign a given fingerprint to one of the pre-specified categories according to its geometric appearance,
- Fingerprint Matching: (verification/identification).

In order for fingerprint classification and matching algorithms to be implemented automatically, it is necessary to obtain the most important features (minutiae and singular points), to recover deformations, and detect spurious features. The process of an AFRS can be summarised as shown in Figure 2.8. Some parts of Figure 2.8 are not been addressed in this research namely, fingerprint acquisition and matching (identification and verification).

2.3.1 Fingerprint Acquisition

A number of methods are used to acquire fingerprints. Among them, the inked impression method remains the most popular. It has been essentially a standard technique for fingerprint acquisition for more than 100 years, [Battley37]. The first step in capturing an inked impression of a fingerprint is to place a few dabs of ink on a slab, then roll it out smoothly until the slab is covered with a thin, even layer of ink. The finger is then rolled from one side of the nail to the other side over the inked slab. After that the finger is rolled on a piece of paper so that the inked impression of the ridge pattern of the finger appears on the paper. This method is time-consuming and unsuitable for an on-line fingerprint verification system [Jain97].

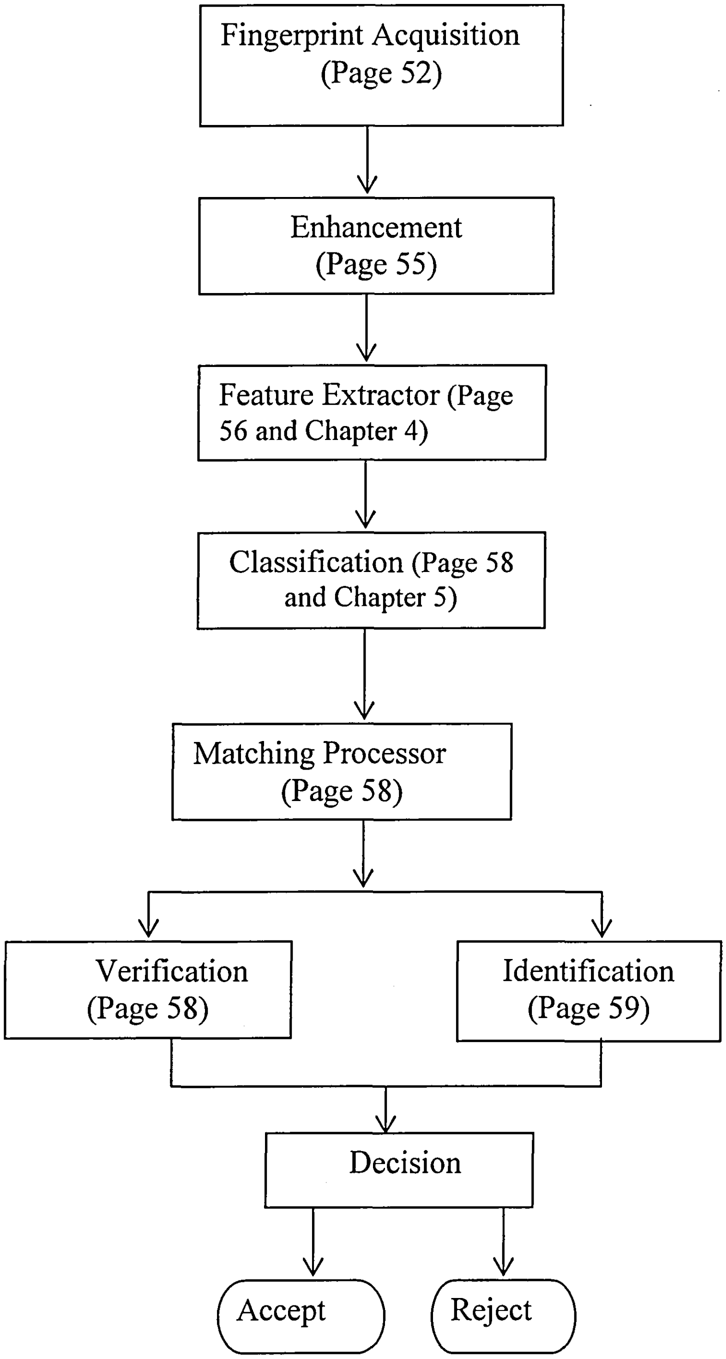

Fingerprint Acquisition (Page 52)
Enhancement (Page 55)
Feature Extractor (Page 56 and Chapter 4)
Classification (Page 58 and Chapter 5)
Matching Processor (Page 58)
Verification (Page 58)
Identification (Page 59)
Decision
Accept
Reject

Figure: 2.8. Steps taken by a general-purpose of fingerprint recognition

The second method, which is more efficient and reliable, is the optical data generation system. It consists of a prism and a uniform light beam that transforms the three-dimensional data into two-dimensional data, which can be photographed and quantized. This system makes use of the total internal reflection obtained in a 90° prism. The uniform light beam is shone into the prism. Normally, all this light is reflected out of by the 45° surface of the prism. However, when a finger is pressed on this surface, the reflection index of that surface is changed at the places where the ridges of the finger touch the prism's surface. This means that the ridges of the print appear dark in the image, while the background is of high intensity. This reflected image of the fingerprint is photographed by the video camera and quantized by the framegrabber [Coetzee93]. The optical method of fingerprint data generation is not perfect either because the contrast and focus of the image obtained are sometimes poor. However, the method is clean, fast, and most of the problems can be overcome by good pre-processing techniques.

The third method is the fingerprint scanner, which is capable of directly acquiring fingerprints in digital form. This method eliminates the intermediate digitisation process of inked fingerprint impressions and makes it possible to build an on-line system [Jain99c]. The fourth method is solid-state sensors. These are microchips containing a device that images the fingerprint via one of the several technologies, including electrical measurements and temperature sensitive sensors [John96]. Solid-state sensors, offer the capability to automatically adjust the sensitivity of a pixel, row or local area to provide added control of image quality.

One major problem in processing fingerprint image is the quality of the original image. Problems exist in feature extraction from poor quality images. Other problems exist in extracting feature from fingers of elderly people as well as manual labourers.

The problem with elderly people's prints is that prominence of the ridges diminishes, with the result that the fingerprint pattern is not clear. Manual workers have the problem that the skin on the hands is subject to severe punishment, with the result that false minutiae and singular-points are created by cuts in the skin and in some cases the ridges are worn away.

2.3.2 Fingerprint Enhancement

Fingerprint enhancement is a common step in several systems for automatic fingerprint identification [McCabe92]. In practice, due to variations in impression conditions, ridge configuration, skin conditions (aberrant formations of epidermal ridges, postnatal marks, and occupational marks), acquisition devices, and non-cooperative attitude of subjects, a significant percentage of acquired fingerprint images are of poor quality. The ridge structures in poor-quality fingerprint images are not always well defined. In order to ensure that the performance of the feature extraction algorithm will be robust with respect to the quality of input fingerprint images, an enhancement that can improve the clarity of the ridge structures is necessary. A fingerprint system is able to correctly identify the minutiae and singular point by using various visual clues such as local ridge orientation, ridge continuity, ridge tendency, as long as the ridge are not corrupted completely [Hong96]. It is possible to develop an enhancement algorithm that exploits these visual clues to improve the clarity of ridge structures in corrupted fingerprint images. Generally, for a given fingerprint image, the region of interest can be divided into the following three categories:

1. Well-defined regions, where ridges are clearly differentiated from one another,

2. Recoverable corrupted regions, where ridges are corrupted by a small amount of creases, smudges, but, still visible and neighbouring regions provide sufficient information about the true ridge,
3. Unrecoverable corrupted regions, where ridges are corrupted by such severe amount of noise and distortion that no ridges are visible and the neighbouring regions do not provide sufficient information about the true ridge structures either.

The goal of an enhancement algorithm is to improve the clarity of ridge structures of fingerprint images in recoverable regions and to remove the unrecoverable regions. A fingerprint enhancement algorithm should not result in any spurious ridge structures. This is very important because spurious ridge structure, may change the individuality of input fingerprints. Fingerprint enhancement can be conducted on either, binary ridge images or grey-level images.

2.3.3 Feature Extraction

The general fingerprint appearance is valleys separated by continuous ridges. Fingerprints also have a number of features, such as bifurcations, ridge ending, cores, deltas, ridges and lakes. A single rolled fingerprint may have as many as 100 or more different feature points that can be used for identification and classification purposes. There is no exact size requirement as the number of points found on a fingerprint impression depends on the location of the print. As an example, the area immediately surrounding a delta or core will probably contain more feature points per square millimetre than the area near the tip of the finger, which tends to have fewer points. In Figure 2.9, we see part of a fully rolled fingerprint with 10 labelled features. There are a total of 22 different features on this 1/4" square section of impression.

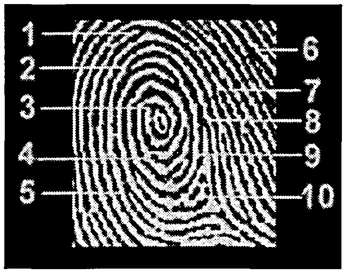


Figure: 2.9. Section (1/4" square) from the central part of a pattern with labelled features: *(1) independent ridge, (2) a spur point, (3) core point, (4) and (7) ridge end points, (5) and (8) bifurcation points, (6) a crossover point, (9) a lake point, and (10) delta point.*

In this research, we concentrate on features which give the uniqueness of fingerprints, namely, minutiae points (ridge ends and bifurcations) and the singular points (delta and core points). The former are useful in identification purposes, while the latter are useful for classification purposes. Using empirically determined thresholds can reduce extraneous features. For instance, a bifurcation having a branch that is much shorter than an empirically determined threshold length is eliminated because it is likely to be a spur. Two endings on a very short isolated line are eliminated because this line is likely to be due to noise. Two endings that are closely opposing are eliminated because these are likely to be on the same ridge, due to a scar or noise or a dry finger condition that results in discontinuous ridges. Endings at the boundary of the fingerprint are eliminated because they are not true endings, but rather the extent of the fingerprint in contact with the capture device [Asker01]. Feature extraction is discussed in Chapter 5.

2.3.4 Fingerprint Classification

Fingerprint classification plays a vital role in fingerprint recognition. The goal of fingerprint classification is to assign a given fingerprint to a specific category according to its geometric properties. Generally, manual fingerprint classification is performed within a specific framework, such as, the Henry system [Battley37]. Classification is based on ridge patterns, local ridge orientations and singular-points. Therefore, if these properties can be extracted and described quantitatively then fingerprint classification becomes an easier task. Fingerprint classification is the main contribution of this research. Chapter 3 will discuss current approaches of fingerprint classification; Chapter 4 will present the feature extraction for classification, while Chapter 5 will present the implementation of the feature analysis and classification system using neural network and fuzzy-neural classifiers.

2.3.5 Fingerprint Matching

Fingerprint matching determines whether two fingerprints are from the same finger (fingerprint verification), or search for a given fingerprint in database of templates (fingerprint identification). It is widely believed that if two fingerprints are from the same source, then their local ridge structures (minutiae details) match each other topologically [Coetzee92].

- **Verification**

Verification is the comparison of a claimant fingerprint against an enrolee fingerprint, where the intention is that the claimant fingerprint matches the enrolee fingerprint. To prepare for verification, a person initially enrols his/her fingerprint into the verification system. A representation of that fingerprint is stored in some compressed format along with the person's name or other identity. Verification is also known as, *one-to-one matching*.

- **Identification**

Identification is the process where a fingerprint of unknown ownership is matched against a database of known fingerprints to associate with an identity. Identification is also known as, *one-to-many matching*.

The two most prominent structures are ridge endings and ridge bifurcation's (minutiae). Based on this observation, and by representing the minutiae as a point pattern, an automatic fingerprint verification/identification problem may be reduced to a point pattern matching problem. In the ideal case, if the correspondences between the template and input fingerprint are known, there are no deformations such as translation, rotation and non-linear deformations between them, and each minutiae present in a fingerprint image is exactly localised, then fingerprint verification/identification consists of the trivial task of counting the number of spatially matching pairs between the two images. However, in practice no correspondence is known beforehand, there is relative translation, rotation and non-linear deformations between template minutiae and input minutiae, spurious minutiae are present in both templates and inputs, and some minutiae are missed.

2.4 Fingerprint Applications

Fingerprints have the potential to be widely adopted in a very broad range of civilian applications, such as, (i) banking security and as electronic fund transfers, ATM security, cheque cashing, and credit card transactions, (ii) physical access control, such as, airport access control, and high security areas, (iii) information system security, such as, access to databases via login privileges, (iv) customs and immigration to permit faster immigration procedures (v) national ID systems, which provide a unique ID to the citizens, and (vi) smart cards, desktop PCs, workstations, and computer networks. It could be used during transactions conducted via the and Internet (electronic commerce). In automobiles, biometrics can replace keys with key-less entry devices.

Forensic	**Civilian**	**Commercial**
Criminal Investigation Corpse Identification Determination	National ID Driver's license Border crossing	ATM Credit cards Cellular phones Access controls

Table: 2.1 Biometric Application Areas.

In general we can divide the applications into three main areas, which are forensic civilian and commercial applications as shown in Table 2.1. Public acceptance of fingerprint applications however, is very low, as most people still assume that fingerprints are related to law-enforcement and forensic application.

2.5 Conclusion

Fingerprint pattern types, their interpretation, characteristics and features analysis has been presented. Issues of fingerprint recognition have been reviewed including fingerprint acquisition, enhancement, feature extraction, classification, and matching. The particular emphasis of this research is the issue of fingerprint classification, which will be discussed in the subsequent chapters.

Chapter 3

Review of Fingerprint Image Processing and Classification Techniques

3.1 Introduction

Computer image processing and analysis is often required in many automatic visual inspection processes. While trained human operators are able to grade a product accurately, their performance fails remarkable when they have to deal with high speed and repetitive tasks. On the other hand, in spite of several years of research in pixel-based image processing techniques computerised image processing systems are often unable to recognise characteristics that would be obvious to human visual inspection [Nyongesa01]. In general, image-processing can be treated as a part of a pattern recognition problem. Usually the "input" to this activity is a scene or image source, and the "output" is some decision, description, action, or report that represents a source. Several pattern recognition algorithms have been proposed for fingerprint classification, including syntactic approaches [Moayer75], [Kameshwar80], [Xia091], methods based on detection of singular points [Rao80], [Asker01], connectionist algorithms such neural networks [Wilson94], and structural methods based on (dynamic) graph matching [Lumini99b].

Various techniques are used to interpret images. Most research is focused on the intermediate and high levels of abstraction. There are researchers who take clues from biological systems to develop theories, and there are those who focus on mathematical and physics theories regarding the imaging process. Eventually however, theory becomes practice in the specification of an algorithm embodied in an executable program with appropriate data representations. The technical problem is that of automatically deriving a sensible description from an image. Typically, in any domain there are named objects and characteristics that can be used to make a decision. Obviously, there is a wide gap between the basic nature of images (essentially arrays of numbers) and their descriptions [Mehtre89]. It is the bridging of this gap that is the essence of image analysis, image processing, and computer vision. One example is model matching where stored geometric descriptions of objects of the domain are matched with extracted features from the images [Karu96]. Techniques are called "bottom-up" when the primary direction of flow of processing is from lower abstraction levels (images) to higher levels (objects), and conversely "top-down" when the processing is guided by expectations from the domain.

In real-world image processing, the input dimensionality can be of a very high order and the discriminate functions for approximation are very non-linear and complex. A classifier based directly on the measured objects (i.e. image pixels) would require a large number of parameters in order to approximate, and generalise well all over the input domain. In this Chapter, we review current approaches in fingerprint image processing, which combines image pre-processing, feature extraction, classification, and matching.

The rest of this Chapter is organised as follows. Section 3.2 presents an overview of the stages of a pattern recognition system. Section 3.3 discusses image understanding, while Section 3.4 discusses current approaches for fingerprint image processing and classification. Finally, Section 3.5 draws some conclusions and remarks on this Chapter.

3.2 Overview of Pattern Recognition

Pattern Recognition (PR) is the science of giving names to the natural objects in the real world [Jouko98]. The basic setting of PR is, there is one unknown object presented as a set of signals or measurements in the input of black box called a PR system. At the output of the system, there is a set of predefined classes. The purpose of the system is to assign the object to one of the classes. The list of classes may also contain a special reject class for the objects the system is unable to classify. PR involves many stages such as making the data collection, pre-processing, segmentation, feature extraction, classification based on the extracted features, and finally post-processing.

- Data Collection: the first stage in any PR system is data collection. Before a pattern vector is made up a set of measurements need to be performed using some technical equipment and may be converted to numerical form. In the case of image analysis, such equipment includes video cameras and scanners. In any case, the data collection devices should record the objects with the highest fidelity available. Any additional noise will be disadvantageous to successful operation of the system. The data collection phase should also be designed in such a manner that the system will be robust to variations in operation of individual signal measurement devices.
- Pre-processing: real-world input data always contains some amount of noise and certain pre-processing is needed to reduce its effect. Pre-processing is normally

accomplished by some simple filtering method on the data. In case of image processing, the image may be median filtered to remove spurious point noise which may hamper segmentation process.

- Segmentation: the pre-processed input data has to be split in subparts which make meaningful entities for classification. Segmentation may either be a clearly separate process or tightly interwoven with previous or following processes [Jouko98]. In either case, after the PR system has completed processing a totality of data, the resulting segmentation of the data to its subparts can be revealed. Depending on how the application has been realised, the segmentation block may either add information regarding the segment boundaries to the data follow, or simply, copy all the segments to the following stage.
- Feature Extraction: feature extraction is to obtain from the input data the information which is most relevant for classification purposes, while minimising the intra-class pattern variability while enhancing the inter-class pattern variability. During the feature extraction process the dimensionality of data is reduced [Ratha95a]. This is almost always necessary, due to the limits in memory and computation time. A good feature extraction scheme should maintain and enhance those features of the input data, which make distinct pattern classes separate from each other.
- Classification: all the preceding stages should be designed and tuned to aim at success in the classification phase. The operation of classification step can be simplified to being that of a transform of quantitative input data to qualitative output information. The output of the classifier may either be a discrete selection of one of the predefined classes, or a real-valued vector expressing the likelihood values for assumptions that the pattern was originated from the corresponding class.

- Post-processing: in most PR systems, some data processing is performed after the classification stage. Post-processing brings some a priori information about the surrounding world into the system. This additional expertise can be utilised in improving the overall classification accuracy. The post-processing stage is generally possible if the individual objects or segments make up meaningful entities. The soundness of these high-level objects can be examined and if an error is indicated, further steps can be taken to correct the mis-classification.

The discussed PR stages can be summarised, as shown in Figure 3.1. Depending on the measurements and the classes, there are divergent areas of PR, including recognition of speech [Leondes98a], analysis of time signals [Moraitakis00], character recognition [Leondes98b], document classification [Rudy98], and image recognition [Tsao93].

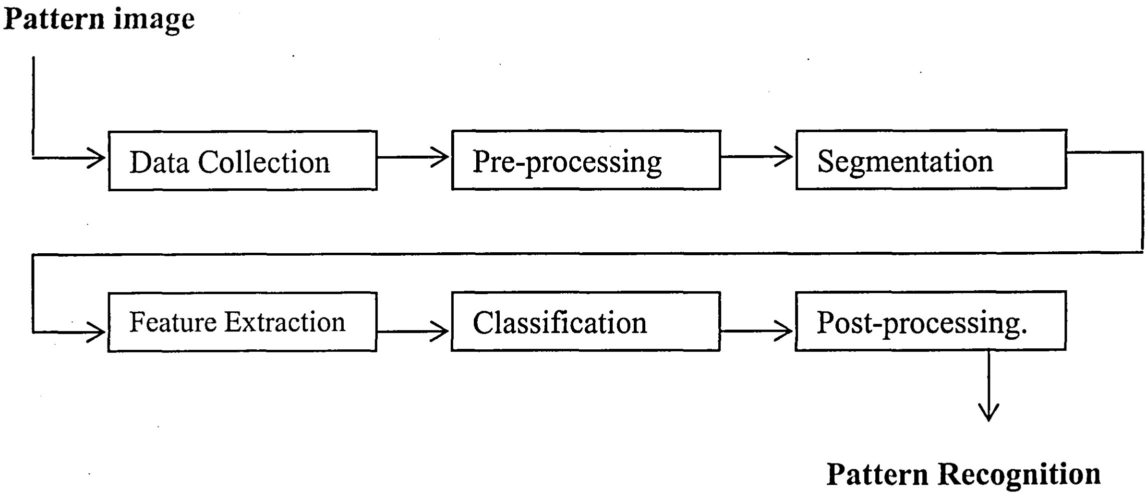


Figure 3.1 A block diagram of a generic pattern recognition system.

3.3 Image Understanding

Image understanding is the automation process of visual tasks by computer. In order to make the link between image data and domain descriptions, an intermediate level of description is introduced. Processing usually starts with some image processing, where noise and distortion are reduced and certain important aspects of the imagery are emphasised. Then, features are extracted from the image(s) that characterise the information needed for description. Typically, these features can be blobs, edges, lines, corners, regions, etc. They are stored at the intermediate level of abstraction. Such descriptions are free of domain information; they are not specifically objects or entities of the domain of understanding, but they contain spatial and other information to represent the sense of the image. It is the spatial/geometric information that can be analysed in terms of the domain in order to interpret the images. There are alternate views of vision, resulting in other paradigms for understanding. For instance, "connectionist" views will describe the process differently, and yet produce useful and practical results [Isenor86].

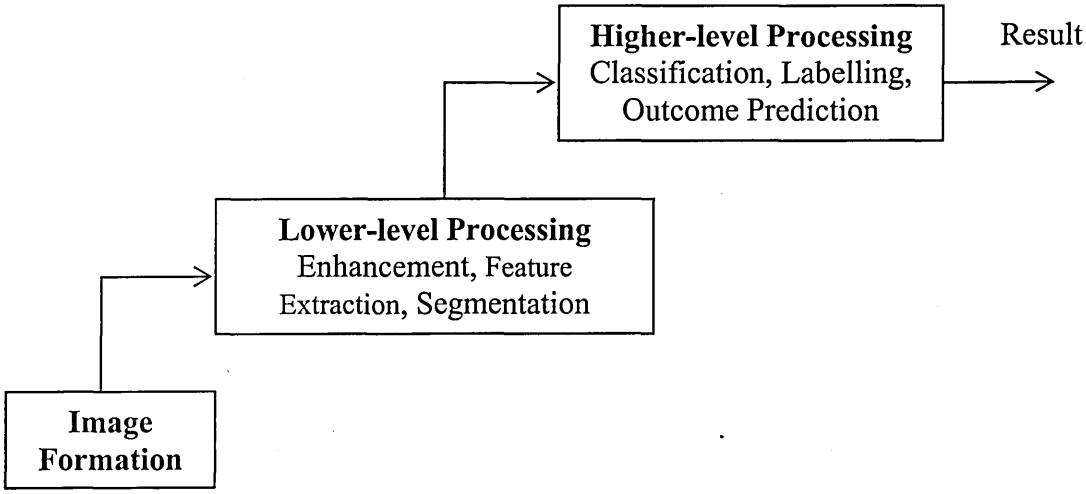

Higher-level Processing
Classification, Labelling, Outcome Prediction
Result
Lower-level Processing
Enhancement, Feature Extraction, Segmentation
Image Formation

Figure 3.2 Model for image understanding system in three-level.

To provide a common platform for studying the various problems of image understanding, we can employ a three-level model as shown in Figure 3.2. At the lowest level, images are formed (acquired in certain format). Then image processing is separated into two levels as: lower-level processing and high-level processing.

3.4 Current Approaches for Fingerprint Image Processing and Classification

Currently, there are a number of approaches for pre-processing, feature extraction, classification and matching that have been investigated for the purpose of automatic fingerprint recognition. These approaches are discussed in two categories: approaches for general fingerprint recognition and processing, such as, structural, statistical, syntactic, geometric, mathematical, hybrid approaches and artificial neural networks.; and approaches for fingerprint classification, such as, inexact graph matching, dynamic masks, multi-space Karhunen-Loe've Transform (KLT), directional image, nearest-neighbor classifier, hybrid fingerprint matcher and Fourier Transform based.

3.4.1 General Approaches for Fingerprint Processing

- *Structural-based*

The *structural-based* approach is to analyse the global configuration of fingerprint patterns with a sampling square, which describes the distribution of ridge directions as well as determines the existence or rough positions of feature points, such as core and delta, and exploiting the topology of the features [Mehtre89]. According to the different structures encountered, a fingerprint image would be classified into a number of classes. In the structural approach one extracts features based on minutiae and represents the features using a graph data structure. An alignment-based minutiae matching algorithm has been implemented in [Jain97]. This algorithm is capable of finding the correspondences between input minutiae and the stored template without resorting to exhaustive search and has the ability to adaptively compensate for the non-linear deformations and inexact matching. Fingerprint minutiae matching has been approached using several different strategies including image-based and ridge pattern matching [Ratha96].

- *Statistical-based Approach*

In the statistical-based approach, statistical characteristics are calculated as the attributes of ridges and feature points for fingerprint classification [Vermwa87], [Murthy92], [Roddy97]. Among the various frameworks in which pattern recognition has been formulated, the statistical approach has been most intensively studied and used in practice. However, the statistical-based approach suffers from sensitivity and noise, which usually appears in fingerprint images. Local fingerprint features essential to the classification process, such as cores and deltas, might be missing due to noise and as a result would most likely lead to erroneous classification.

- *Geometric-based Approach*

Shape analysis methods play an important role in systems for object recognition, matching, registration, and analysis. Research in shape analysis has been motivated, in part, by studies of human visual form perception systems [Gorsky90]. Shape analysis methods are classified into several groups. Classification is determined according to the use of shape boundary or interior, and according to the type of result. In geometric-based approach [Chong97] and [Murthy92], an analysis of the global geometric shape of fingerprint ridges is used to discriminate between the different classes. The geometric approach also depends on local features, thus, similar to the structural and statistical approaches, and faces the same problem of sensitivity to image noise.

- *Artificial Neural Networks-based Approach*

In the Artificial Neural Networks (ANNs) approach [Jain99b], [George99], and [Tsao93], a connectionist system takes as input the ridge directions and features. The system could be able to learn about the different classes and be trained to classify accurately. Self-organising feature maps, in particular, have been used to cope with uncertainty, in order to deal with fingerprint images having distorted regions [Halici96]. A probability neural network classifier was trained for a six-class problem, and an auxiliary whorl-detecting classifier was added to trace and analyse pseudo-ridges (approximate trajectories through the ridge flow), and finally, combining the outputs of the neural network and auxiliary classifiers so as to decide on a hypothesised class and a confidence level [Candela95]. Other research used ANNs in [Neto97], [Watson94], [Blue94]. The neural networks-based approach is most viable among the above approaches, if sufficiently extensive sample images of all different classes are provided. However, due to complexity of fingerprints, thousands of training instances are often required to train the network, which slows the computation speed and may not perform well for large problems.

- *Syntactic and Mathematical based*

In the syntactic-based approach [Moayer75], [Fu82], a grammar is used to represent and to classify fingerprint patterns. The representation is in the form of strings of primitives, which would be parsed according to a set of production rules. A variety of grammars for the fingerprint classification problem have been considered. These include context-free grammar [Bruyne82], regular grammar [Verma89], and tree grammar [Moayer76].

In the mathematical-based approach [Cheung87] and [Rao74], a mathematical model is developed to compute local fingerprint ridge orientation from core and delta points for the purpose of fingerprint classification. It also depends on local features such as core and delta points similar to the structural and statistical approaches. This approach again faces the same problem of sensitivity to image noise.

- *Hybrid Approaches*

In the hybrid approaches, two or more of the foregoing approaches are combined to accomplish the classification task [Yoshtaka91], [Yong92], [Wilson97a]. An amalgamation of the structural and syntactic approaches, and the structural and statistical approaches have been implemented [Ammar98]. Decision-level fusion in fingerprint verification [Prabhakar97] is a scheme for classifier combination, which stresses the importance of classifier selection during combination. The proposed scheme is optimal when sufficient data are available to obtain reasonable estimates of the join densities of classifier outputs. Four different fingerprint-matching algorithms were combined using the proposed scheme to improve the accuracy of a fingerprint verification system. Experiments conducted on a large fingerprint database (2,700 fingerprints) confirm the effectiveness of the proposed integration scheme.

Most fingerprint matching systems rely on the distribution of minutiae on the fingertip to represent and match fingerprints [Jain00]. While the ridge flow pattern is generally used for classifying fingerprints, it is seldom used for matching. [Jain00], however, describes a hybrid fingerprint matching scheme that uses both minutiae and ridge flow information to represent and match fingerprints. A set of 8 Gabor filters [Murthy92], whose spatial frequencies correspond to the average inter-ridge spacing in fingerprints, is used to capture the ridge strength at equally spaced orientations. A square tessellation of the filtered images is then used to construct an eight-dimensional feature map, called the ridge feature map. The ridge feature map along with the minutiae set of a fingerprint image is used for matching purposes. The proposed technique has the following features: (i) the entire image is taken into account in constructing the ridge feature map, and every tessellated cell is equally weighted; (ii) minutiae matching is used to determine the line transformation parameters relating the query and the template images for ridge feature map extraction; (iii) filtering and ridge feature map extraction are implemented in the frequency domain thereby speeding up the matching process; (iv) filtered query images are cached to greatly increase the one-to-many matching speed. The hybrid matcher performs better than a minutiae-based fingerprint matching system. However, for the classification process the singular-points based classification system is more fast and reliable than the minutiae-based.

3.4.2 Techniques for Fingerprint Classification

- *Inexact Graph Matching*

In inexact graph matching [Maio96] a dynamic-clustering technique is used to segment the directional image of a fingerprint in homogeneous regions (that is, regular shaped regions containing elements having similar directions). These regions are used to construct a relational labelled graph, which is invariant to translation and rotations. The fingerprint is then classified by computing the distance between its graph and some fixed graph models [Lumini99a], [Maio96].

- *Fingerprint Classification using Dynamic Masks*

In the fingerprint classification using dynamic masks [Lumini99b] the fingerprint image is partitioned into homogeneous connected regions according to the fingerprint topology, thus, giving a synthetic representation, which can be exploited as a basis for the classification. A set of *dynamic masks*, directly derived from the most common fingerprint classes, together with an optimisation criterion are used to guide the partitioning. The adaptation of the masks produces a numerical vector representing each fingerprint as a multidimensional point, which can be conceived as a continuous classification or used to derive an exclusive classification. The method is invariant with respect to translation and rotation of the fingerprints and works on noisy fingerprints too.

- *Fingerprint Classification based on Multi-space KLT*

This approach is based on a generalisation of the KLT [Maio99], where multiple subspaces are used for representing the patterns. Given a training set of patterns, some subspaces are created according to an optimisation criterion, which attempts to minimise the average mean-square reconstruction error. Intuitively, a KLT subspace can be conceived as a specific view on the data, taken from a certain observation point; in Multi-space KLT (MKLT) several views on the same data are exploited in order to better represent and distinguish the patterns. The MKLT classifier draws linear decision boundaries in the feature space. The main disadvantage is when the data is not linearly separable, in which case no meaningful test classification can be obtained since the system does not have the ability to distinguish between the different classes. Cappelli et al. [Cappelli99] have proposed a fingerprint classification algorithm based on the multi-space KL transform.

- *Fingerprint Classification Techniques Using Directional Image*

Fingerprint classification techniques using directional digitized images find the directional image by checking the orientations of individual pixels, or in some cases the directional histograms, using overlapping blocks of the digitized image.

The complexity of the technique is on the order of the number of pixels in the fingerprint image. The technique does not require iterations or feedback, and is highly parallel. Fingerprint classification techniques using the directional image approach present an important, feature-based technique for automatically classifying fingerprints. In most cases, the techniques extract singular points in fingerprints obtained from directional histograms. This approach is investigated in this research.

- *The Nearest-Neighbor Classifier*

The classifier consists of a database containing the set of labeled template feature vectors, and an unknown test vector is classified by the distance between the test vector and every one of the template vectors [Coetzee90]. The unknown test vector is labeled as belonging to that class for which the distance is the smallest. One advantage of the nearest-neighbour classifier is that no training is associated with it. This means that the classifier is ready to use the moment labeled feature vectors are available. Another advantage is its ability to generate complex decision boundaries, which enables the classifier to generalize very effectively. However, it is extremely noise sensitive, requires extensive computation time for classification and large storage, because all the template vectors have to be stored for use in calculation for each classification. Classification based on a two-stage classifier which uses a K-nearest neighbour classifier in the first stage and a set of neural networks in the second stage [Jain98].

- *Classification using the Fourier Transform*

The Fourier transform is an important digital signal and digital image processing tool, which translates the spatial data into the frequency domain [Coetzee93]. In fingerprint classification, the Discrete Fourier Transform is used to decompose an image into its sine and cosine components. The output of the transformation represents the image in the frequency domain, while the input image is the spatial domain equivalent. By modifying frequency components of an image, periodic noise can be reduced and modifying frequency components of an image can enhance the image. However, the processing time is very long. Approaches, which combine Fourier Transform with other algorithms are, Fourier Transform and Wedge-Ring Detector [Coetzee90] and Fourier transform and Wavelet Transforms [Mallat89].

3.5 Conclusion

In this Chapter a review of fingerprint image processing and classification techniques has been presented. We described, in general, the current approaches for fingerprint recognition including statistical, structural, and neural networks. Current fingerprint classification techniques were discussed including inexact graph matching, dynamic masks, MKLT, nearest-neighbor classifier, and Fourier Transform. Evaluation of current classification techniques in term of classification accuracy on different databases has been discussed and concluded. The proposed implementation of fingerprint classification system will be discussed in subsequent Chapters.

Chapter 4

Fingerprint Feature Extraction

4.1 Introduction

The efficiency of image processing systems depends on the quality of the images, the basic models and evaluation algorithms, and also on the representation of the data. At the beginning of such a system the image is only represented as a matrix of values, such as, grey-level or colour. The task of the system is to obtain from this matrix an application dependent representation of the image content so that a user or subsequent process can take advantage of this information. Traditionally, characteristic features represent objects in automatic pattern recognition systems. Classifiers are optimised using examples of objects for training. Feature Extraction (FE) is a process through which geometric primitives within images are isolated in order to describe the image structure, i.e. used to extract important image information and to suppress redundant information or neglect information, which is not used in the following processes. Additionally, FE also identifies the topological neighbourhood relationships between the features. Features and relations provide a symbolic description of the image, represented in the gif format of the image. This Chapter reviews feature extraction techniques, and then discusses the proposed fingerprint feature extraction algorithm.

The rest of this Chapter is organized as follows. Section 4.2 discusses general issues of image feature extraction. Section 4.3 presents fingerprint feature extraction, including review of some existing techniques. Section 4.4 describes the proposed feature extraction algorithm. Finally section 4.5 draws some conclusions from the Chapter.

4.2 Overview of Image Feature Extraction

Though there exist many techniques for extracting features from images, the derivation of a meaningful symbolic description is difficult for many reasons. For example, consider the detection of edges in an image. First, the concept of a (meaningful) edge has to be clarified. A commonly used definition is: "an image contour across which the brightness changes abruptly" [Nalwa94]. But these edges may not be the ones that are desired by the human operator or an automatic interpretation system. Some physical discontinuities such as surface-reflectance, specular-reflectance, or depth discontinuity can also cause an intensity edge. It is not trivial to distinguish between the two edge types just by using image information. Furthermore, it is possible that some of the discontinuities are not visible due to low contrast in the image. Another problem is the fact that an image is a (physical) observation of the world and therefore we must expect uncertainties within the observation. For example, each sensor of the CCD camera introduces noise that leads to pixel values, which can vary within a certain range. Keeping this in mind, we have to accept that a symbolic description is erroneous up to a certain extent as long as we only use information from the image. An error in a symbolic description can be either quantitative (e.g. an edge is not as long as expected) or qualitative (e.g. an edge is missing or unwanted). Fingerprint images are not exempt from such errors.

As a consequence of the problems mentioned above, FE is often understood as deriving a symbolic image description [Marr82], [Fishler85], which is not necessarily complete, but serving a special purpose. This leads to a large amount of approaches for detecting the basic features such as points, edges, and regions.

4.2.1 Image Smoothing Techniques

Smoothing techniques are used for noise reduction and are usually referred to as low pass filtering. These techniques can be implemented either in the spatial domain or in the frequency domain. In the spatial domain, there are two commonly used smoothing techniques, which are neighbourhood (mean filtering) averaging and median filtering. In neighbourhood averaging, each pixel of the image is replaced by the average of odd-sized (e.g. 3x3, 5x5) masks. In median filtering, each pixel of the image is replaced by the median value of odd-sized masks. The basic approach of the mask operator is to sum the products of the mask coefficients and the intensities of the pixels under the mask at a specific location in the image. Figure 4.1 shows a general 3x3 mask. Denoting the grey-levels of pixels under the mask at any location by z_1, z_2, ..., z_9, the response of a linear mask is as follows:

$$R = w_1 z_1 + w_2 z_2 + \ldots + w_9 z_9 \tag{4.1}$$

w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

Figure 4.1. Example of 3x3 mask with arbitrary coefficient (weights).

If the centre of the mask is at location (x,y) in the image, the grey-level of the pixel at this location is replaced by R. The mask is then moved to the next pixel location in the image and the process is repeated until all the pixel locations are covered. The value of R is computed by using partial neighbourhoods for pixels that are located in the border of the image. Nonlinear spatial filters also operate on neighbourhoods. In general, however, their operation is based directly on the values of the pixels in the neighbourhood under consideration [Rafael93], and they do not explicitly use coefficients in the manner of Eq. (4.1). Noise reduction can be achieved with a nonlinear filter whose basic function is compute the median grey-level value in the neighbourhood in which the filter is located. Examples include the max-filter, whose response is shown in Eq. 4.2, which used to find the brightest point in an image.

$$R = \max\{z_k | k = 1,2,\ldots,9\} \qquad (4.2)$$

A lowpass (smoothing) spatial filter indicates that the filter has to have all positive confinements [Rafael93]. For a 3x3 spatial filter, the simplest arrangement would be a mask in which all coefficients have a value 1, the sum is scaled by dividing R by 9 (Figure 4.2). Note that the response R would simply be the average of all pixels in the area of the mask. For this reason, the use of masks of this form is often referred to as neighbourhood averaging.

$$\frac{1}{9}\times \begin{array}{|c|c|c|}\hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline \end{array}$$

Figure 4.2 Example of a spatial lowpass filter of 3x3 mask.

If the objective is to achieve noise reduction rather than blurring, an alternative of median filters may used. That is, the grey-level of each pixel is replaced by the median of the grey-levels in a neighbourhood of that pixel, instead of the average. This method is particularly effective when the noise pattern consists of strong, spikelike components and the characteristic to be preserved is edge sharpness. The median *m* of a set of values is such that half values in the set are less than *m* and half are greater than *m*. In order to perform medium filtering, values are sorted in ascending or descending order and each pixel is replaced by the medium one. For example in a 3x3 mask, the median value is the 5th largest value, while in a 5x5 mask the median value is 13th largest value. Thus the principal function of median filtering is to force points with distinct intensities to be more like their neighbour, actually eliminating intensity spikes that appear isolated in the area of filter mask.

A highpass (sharpening) spatial filter suggests that the filter should have positive coefficients near its centre, and negative in the outer periphery (Figure 4.3). Reducing the average value of an image to zero implies that the image must have some negative grey-levels. Dealing with only positive levels, the results of highpass filtering involve some form of scaling so that the grey-levels of the final results span a range. Taking the absolute value of the filtered image to make all the values positive is usually not a good idea because large negative values would appear brightly in the image.

$$\frac{1}{9}\times\begin{array}{|c|c|c|}\hline -1 & -1 & -1 \\ \hline -1 & \mathrm{w} & -1 \\ \hline -1 & -1 & -1 \\ \hline\end{array}$$

Figure 4.3 Example of a spatial highpass filter of 3x3 mask.

As averaging is analogous to integration, differentiation can expect to have the opposite effect. The common method of differentiation in image processing applications is the gradient. For a function *f(x,* y), the gradient of *f* at coordinates *(x, y)* and the magnitude of its vector are defined in Eq. (4.3) and (4.4).

$$\Delta f = \left[\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}\right] \qquad (4.3)$$

$$mag(\Delta f) = \left[\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2\right]^{1/2} \qquad (4.4)$$

Consider a 3x3 image region, denoted by z values of grey-levels. Eq. (4.4) can be approximated at point z_5 in a number of ways. The simplest is to use the difference (z_5-z_8) in the x direction and (z_5-z_6) in the y direction, combined as:

$$\Delta f \approx \left[(z_5 - z_8)^2 + (z_5 - z_6)^2\right]^{1/2} \tag{4.5}$$

Another approach is to use cross differences as:

$$\Delta f \approx \left[(z_5 - z_9)^2 + (z_6 - z_8)^2\right]^{1/2} \tag{4.6}$$

Eq. (4.5) and (4.6) can be implemented by using 2x2 mask. For example, Eq. (4.6) can be implemented by taking absolute value of the response of the two masks shown in Figure 4.4(a) and summing the results. These are called the Roberts cross-gradient operators. Masks of even sizes are awkward to implement [Rafael93]. An approximation to Eq. (4.4) using 3x3 neighbourhoods is:

$$\Delta f \approx \left|(z_7 + z_8 + z_9) - (z_1 + z_2 + z_3)\right| + \left|(z_3 + z_6 + z_9) - (z_1 + z_4 + z_7)\right| \tag{4.7}$$

The difference between the third and the first row of the 3x3 region approximates the derivative in the x direction, and the difference between the third and first column approximates the derivative in the y direction. The masks shown in Figure 4.4(b), called the Prewitt operators, can be used to implement Eq. (4.7).

1	0
0	-1

0	1
-1	0

(a) Roberts 2x2 operators

-1	-1	-1
0	0	0
1	1	1

-1	0	-1
-1	0	1
-1	0	1

(b) Prewitt 3x3 operators

Figure (4.4). Different masks used to compute the derivative.

4.2.2 Extraction of Points

Points are image objects, whose geometric properties can be represented by only two co-ordinates (*x,y*), and they could take several types such as circular symmetric points, line-end points, corners, and junctions. The symbolic description of points can be given as a list containing geometric (e.g. co-ordinates), radiometric (e.g. strength), and relational attributes (e.g. intersection points). Deriving the point coordinates normally follows one of two methods:

(i) Point template matching: one possibility to detect point regions is to define a point pattern (template) which represents the point structure we are looking for. The main idea of template matching is to find the places in the image where the template fits best. The similarity between the template and the image can be evaluated by multiplication of the template values with the underlying image intensities, or by estimation of the correlation coefficients [Forstner87]. The disadvantages of template matching, in general, are the limitation by the number and types of templates, and sensitivity to changes in scale and to image rotation.

(ii) Point detection by optimisation: measures the suitability or interest of an image point by the estimation of the variances in a small window (e.g. 4x4, 8x8 Pixel). This method is used in many stereo matching algorithms and relies on interest operators proposed by [Paderes84]. Similar to the Moravec-Operator [Forstner87], the objective of this method is the detection of interesting points (but with higher accuracy). It is able to detect different point types with the same algorithm and can be used either for image matching or image analysis. The interest operator consists of two processing steps: selection of optimal windows and finding the point location within these windows.

4.2.3 Extraction of Edges

An edge is an image contour, where a certain image property like brightness, depth, colour, or texture changes abruptly perpendicular to the edge [Ballard82]. According to these characteristics, edges can be classified into two general types [Nalwa94]: step edges (edges) and bar edges (lines). Edges represent boundaries between two regions, and lines are either a discontinuity in orientation of surfaces, or they are small elongated objects, like streets in a high scale image.

Edge extraction normally leads to incomplete description of the image, that is, edges do not build closed boundaries of the homogeneous image regions. The types of representation of single edges are manifold depending very much on the approach. Edges can be applied to a broad range of problems, such as, relative orientation (e.g. edge-based matching in stereo pairs [Li91]), absolute orientation (e.g. matching edges with frame wire models [Wolff94]), and object recognition and reconstruction.

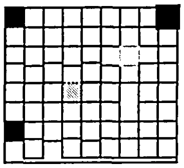
Both edge types can be detected by the discontinuity in the image domain. Since the beginning of digital image processing, edge detection has been an important and very active research area. As a result, a lot of edge detectors have been developed, which differ in the image or edge model they are based on, the complexity, the flexibility and the performance. In particular, the performance depends on (1) the goodness of detection, i.e. the probability of missing and spurious edges and (2) the accuracy of the edge location. For simplicity, we present only the principles of the main approaches, i.e. our focus is on the main processing steps that most edge detector algorithms have in common. A typical approach consists of four steps:

1. Extraction of edge regions: the extraction of all pixels, which probably belong to an edge. The results are elongated "edge regions". The aim is to extract all pixels from an input image, which are likely to be edge pixels.

2. Extraction of edge pixels: extraction of the most probable edge pixels within the edge regions reducing the regions to "one pixel wide edge pixel chains". The aim is to thin the edge regions.

3. Extraction of edge elements (edgels): to estimate edge pixel attributes, e.g. real valued position of the edge pixels, accuracy, strength, orientation. This is the first transition stage from the edge pixels in the discrete image domain to the symbolic description of the edge.

4. Extraction of streaks: an aggregation or grouping of the edgels, which belong to the same edge. The grouping process should provide that the streaks (1) consist of connected edge elements, where each pixel pair is connected by a non-ambiguous pixel path and (2) bound to at most two regions. To provide the second criteria we define a streak as an edge pixel chain between two edge pixels, which are either end-pixel(s) and/or node-pixel(s). For example, dependent on the number of neighbours in a N_8-neighbourhood we can classify the pixel as a node, line or end pixels as shown in Figure 4.5.



■ end pixel: if it has exactly one neighbour pixel

□ line pixel: if it has exactly two neighbour pixel

▨ node pixel: if it has at least three neighbour pixel

Figure 4.5 Classification of edge pixels with respect to the number of neighbours

5. Extraction of edges: the approximation of the streaks by the set of analytic functions, for example polygons. Up to step 4, the extracted streaks are still defined in the discrete image model as they are represented by a set of connected edge elements. This step will change the representation domain from the discrete image raster to a continuous image model.

4.2.4 Extraction of Regions

Regions are image areas, which fulfil a certain similarity criteria, which could be the intensity value of image pixels or some texture properties of the surrounded area of the pixel. The result of such a region extraction should divide or segment the image to a number of disjunct blobs. Ideally, the union of these blobs will reconstruct the image, such that the regions should be connected and bounded by simple lines. Region information has the advantage that it covers geometrically large parts of the image. It can be used for several applications such as data compression, analysing range and binary images, and high-level image interpretation. The large number of region extraction methods can be classified in several ways. One possibility is to separate the methods by the number of pixels, which are used for the grouping decision and therefore called local or global techniques. Furthermore, the methods can be differentiated depending on how the grouping is done. In the first place, the grouping process is defined in the image domain. That means connected pixels can be merged or split directly by the analysis of the neighbouring pixel properties. So, both the similarity and the proximity are considered in one joined processing step. Examples of these, are region growing, region splitting, and region merging algorithms [Zamperoni95]. The second approach applies the similarity and connectivity evaluation in two separate steps. First analyse the discriminating properties of the pixels of the entire image and use the result to define several classes of objects. Examples are thresholding and cluster techniques. In the second step, pixels of the same class and which are also connected in the image space are grouped to homogeneous regions. It performs the proximity criterion, which can be easily done by connected component algorithms [Deriche93].

4.3 Fingerprint Feature Extraction

Raw fingerprint image data offer a rich source of information for matching and classification of objects. For simplicity of pattern recognition system design, a sequential approach consisting of sensing, feature extraction is conventionally adopted where each stage transforms a particular component of information relatively independently. Given raw input measurements, automatically extracting the given representation is an extremely difficult problem, especially where input measurements are noisy, which is common in the case of fingerprints.

The central problem in designing a fingerprint classification system is to determine what features should be used and how categories are defined based on these features. There are, mainly two types of features that are useful for fingerprint recognition system: (i) local ridge and furrow details (minutiae details), which have different characteristics for different fingerprints, and (ii) global pattern configurations, which form special patterns of ridges and furrows in the central region of the fingerprint. The first type of features carry the individuality information about the fingerprints and the second type carry information about the fingerprint class. Therefore, for fingerprint classification, the features derived from the global pattern configurations should be used. These features should be invariant to the translation and rotation of the input fingerprint images. Generally, global fingerprint features can be derived from the orientation field and the global ridge shape. The orientation field of a fingerprint consists of the ridge orientation tendency in local neighbourhoods and forms an abstraction of the local ridge structures. It has been shown that the orientation field is highly structured and can be roughly approximated by the core and delta models [Monro93]. Therefore, singular points details (refer to Figure 4.11) and their relationships can be used to derive fingerprint categories. On other hand, global ridge shape and directional field also provides important clues about the global pattern configuration of fingerprint images.

Methods for fingerprint classification can roughly be divided into two approaches: (i) statistical approach and (ii) structural approach. A statistical approach classifies a fingerprint using vectors derived directly from the orientation field or the input images. Examples include extraction of singular points from directional fields of fingerprints. The method is based on the gradient vector $[G_x(x,y)\ G_y(x,y)]^T$ of the grey scale image $I(x,y)$, defined by [Asker01]:

$$\begin{bmatrix} G_x(x,y) \\ G_y(x,y) \end{bmatrix} = \nabla \mathrm{I}(x,y) = \begin{bmatrix} \dfrac{\partial \mathrm{I}(x,y)}{\partial x} \\ \dfrac{\partial \mathrm{I}(x,y)}{\partial y} \end{bmatrix} \qquad (4.7)$$

Where I(x,y) represents the grey-scale image. The notation of gradients can best be explained when the pixel values are regarded as heights in a continuous 2-dimensional landscape. In this case, the gradient vector is the vector that points in the direction of the steepest descent and the length of the gradient vector is a measure for the steepness [Heijden94]. Gradients can be considered as elementary orientations at each pixel of the image, and discussed further in section 4.4.2.

The Directional Field (DF) is, in principle, perpendicular to the gradients. However, the gradient orientations at pixel scale describe the orientation of the ridge and valley structures, which is a much coarser scale. Therefore, the directional field can be derived from the gradients by performing some averaging operation on the gradients, involving pixels in some neighbourhood [Lindeberg94]. This averaging is the central issue of this method. Then DF is estimated for each pixel in the image using a Gaussian window W for averaging, using the formula:

$$G_{xx} = \sum_w G_x^{\,2}\ ,\ G_{yy} = \sum_w G_y^{\,2}\ \ and\ \ G_{xy} = \sum_{\mathrm{w}} \mathrm{G_x G_y} \tag{4.8}$$

The average gradient direction θ, is given by:

$$\theta = 1/2 \angle (G_{xx} - G_{yy}\, ,\, 2G_{xy});\qquad -\frac{1}{2}\pi < \theta \le \frac{1}{2}\pi \tag{4.9}$$

Following a counter-clockwise closed contour in the DF around a core results in a cumulative change of π in the orientation, while carrying out this procedure around the delta results in -π [Tojo84]. On the other hand, when applied to a location that does not contain a SP, the cumulative orientation change will be zero.

In another fingerprint classification technique using directional images [Mehtre99] the technique enhances images using adaptive clipping and extracts the SP in fingerprints obtained from directional histograms as follows.

S is an NxN reduced directional image quantized in one of four directions (0, π/4 π/2, and 3π/4 respectively).

In order to obtain the delta point(s), the method checks the neighbourhoods of S(i,j) as follows [Ballan98b]:

$$\begin{gathered} S(i-1,j+1) <= S(i,j)) < S(i+1,j+1) \\ S(i-1,j+1) <= \pi/2 \quad \text{and} \quad S(i+1,j+1) >= \pi/2 \end{gathered} \tag{4.10}$$

Next the algorithm eliminates the false delta points as follows:

For each delta point (DP) candidate at pixel S(i,j), check the adjacent horizontal and vertical pixels as shown in (4.12) as follows:

$$\mathrm{DP} = \begin{pmatrix} x & DP(i\text{-}1,j) & x \\ DP(i,j\text{-}1) & DP(i,j) & DP(i,j+1) \\ x & DP(i+1,j) & x \end{pmatrix} \tag{4.11}$$

$$DP(i,j\text{-}1) <= \pi/2 \text{ and } DP(i,j+1) > \pi/2$$

$$DP(i+1,j) <= \pi/4 \quad \pi/4 <= DP(i\text{-}1,j) <= 3\pi/4 \qquad (4.12)$$

Where DP = S and the x positions are not of interest. Candidate satisfying (4.11) and (4.12) correspond to true delta points.

If a point S corresponds to a core point, then it has to satisfy the following inequality:

$$\pi/2 <= (S(i,j)\text{-}S(i,j\text{-}1)) < 3\pi/4 \qquad (4.13)$$

For each core point candidate at pixel S(i,j), the algorithm examines the 2x2 neighbourhood to Northwest, Southwest, Southeast, and Northeast corners to form directional core point matrices H1, H2, H3, and H4, respectively.

$$\begin{bmatrix} H1 & x & H3 \\ x & C(i,j) & x \\ H2 & x & H4 \end{bmatrix} \qquad (4.14)$$

Where C = S. The directional core point matrices are

$$H1 = \begin{bmatrix} C(i\text{-}2,j+1) & C(i\text{-}2,j+2) \\ C(i\text{-}1,j+1) & C(i\text{-}1,j+2) \end{bmatrix}$$

$$H2 = \begin{bmatrix} C(i\text{-}2,j\text{-}1) & C(i\text{-}2,j\text{-}1) \\ C(i\text{-}1,j\text{-}2) & C(i\text{-}1,j\text{-}1) \end{bmatrix}$$

$$H3 = \begin{bmatrix} C(i+1,j\text{-}2) & C(i+1,j\text{-}1) \\ C(i+2,j\text{-}2) & C(i+2,j\text{-}1) \end{bmatrix} \qquad (4.15)$$

$$H4 = \begin{bmatrix} C(i+1,j+1) & C(i+1,j+2) \\ C(i+2,j+1) & C(i+2,j+2) \end{bmatrix}$$

Next, the dominant direction HDn for each Hn matrix is found. Each Hn matrix has four entries and therefore four possible directions. The dominant directions must satisfy the following:

$$(HD1>=\pi/2 \text{ and } HD2<=\pi/4) \text{ or } (HD3<\pi/2 \text{ and } \pi/2<=HD4<\pi) \qquad (4.16)$$

In case that two directions occur which are equal, then the core point is discarded. If the core point is concave, then H1 must be left skewed and H2 must be right skewed. If H3 is right skewed and H4 is left skewed, the point must be convex. Both concave and convex core points are kept.

In the structural approach, fingerprint features are extracted using a number of salient properties and their relationships [Hong00]. The algorithm mainly depends on the ridge verification from a thinned ridge map as an input and generates a refined orientation field and quality index, which indicates the goodness of the input ridge map. Let R, Q', and R' be the input ridge map, the interpolated orientation field, and the verified ridge map, respectively. The major steps of this method as follows:

1. Initialise Q', R', and R which is a map used to indicate the genuine regions.

2. Delete all ridge pixels in R with more than 8-connected pixels to ensure that each ridge is a single 8-connected chain.

3. Trace and label all the ridges in R. For each traced ridge, $r = \{(x_1,y_1), (x_2,y_2), ..., (x_n,y_n)\}$, do the following:

(i) Smooth r,

(ii) Let $(x_{i\in},y_{i\in})$ and $(x_{(i+1)\in},y_{(i+1)\in})$ denote the starting point (i+1) and the ending point of segment in r, where $\in$ is the length of r. Find the 4 neighbouring ridge points, (u,v), (p,q), (u',v'), and (p',q'), which form a quadrilateral at one side of the segment,

(iii) For each quadrilateral, find the minimum rectangle that contains the quadrilateral. Compute the ratio, η, between the area of the quadrilateral and the area of the minimum rectangle. If η is larger than a threshold (e.g. $\eta_0 = 0.75$), then label all the pixels inside the quadrilateral as foreground pixels. Otherwise, label them as background pixels.

Many other different algorithms for singular points extraction are known from literature. Examples of these algorithms are sliding neural networks [Drets99], local energy of the directional image [Perona98], ratio of the sine of the fingerprint image in two adjacent regions [Jain99c], and singular point indexing [Kawagoe84]. However, these algorithms provide somewhat unsatisfactory results, since they only provide continuous measures that indicate how much the local directional image resembles a singular point. Post-processing steps are necessary to interpret the outputs of the algorithms and to make the final decisions, resulting in missed and false singular points.

4.4 Proposed Fingerprint Feature Extraction

A feature point is a point of interest in an image, such as an intersection between two lines. Such points serve to define the relationship between different features. In a character recognition problems, for example, two strokes could fully cross each other, come together in a "Y" or a "T" intersection, form a corner, or avoid each other altogether [Mohamed97]. These are particularly sensitive relationships; the fact that the lines cross in a certain way is more important than the individual lengths of those lines. These relationships are what could be used for character identification. One procedure for extracting these feature points, utilising a thinning algorithm [Lam92], is fairly straightforward. The algorithm is based on examining adjacent pixels in a neighbourhood. Considering 3x3 block of pixels, it is viable to simply loop through the entire block and examine each pixel in turn. If a pixel is "on", its eight neighbours are checked. Since each neighbour can also only be "on" or "off", there are merely 256 possible combinations of neighbourhoods. Extracting feature points thus reduces to calculating a number between 0 and 255 to describe a pixel's neighbourhood and then comparing that number against a table of known feature points.

Generally, FE does not require specific knowledge about the image, and hence can be applied for a broad range of application fields or image types. The image model assumes just the following prerequisites: the image area can be segmented into homogeneous areas with well defined boundaries, where the boundaries can be approximated by straight line segments. Before feature location takes place, some algorithms extract areas of interest by addressing the given image and investigating the features on its surface [Mehtre98]. This allows selection of feature-type dependent on location procedures, which are appropriate and fit best to the local image function. This leads to distinct features, but also non-connected features. The resultant FE code allows users to select or combine any feature types and relations, which are appropriate and useful for the particular application.

In the case of fingerprint image, once a high-quality image is captured, there are several steps required to convert its distinctive features into a compact template. These steps, known as the feature extraction, are at the core of fingerprint recognition technology. Once a quality image is captured, it must be converted to a usable format. Minutiae and Singular-Points (SP) localisation begins with the processed image. At this point, even a very precise image will have distortions and false minutiae or SP that need to be filtered out. Thus, an algorithm may search and eliminate one of two adjacent minutiae, because minutiae are very rarely adjacent. Anomalies caused by scars, sweat, or dirt also appear as false minutiae, and algorithms can locate such points or patterns that don't make sense, such as a spur on an island (probably false) or a ridge crossing perpendicular to 2-3 others (probably a scar or dirt). A large percentage of would-be minutiae are discarded in this process. The points at which minutiae and the singular-point begin are the most logical features, and are used in most applications. The reasoning behind the proposal algorithm is the evidence of the number of singular points and their relative positions on the fingerprint patterns, which is given by the following empirical observations:

(1) If the fingerprint has 0 delta point or 0 core point, then the fingerprint is an arch;

(2) Else if the fingerprint has 1 delta point or 1 core point, then the fingerprint is classified by treating the positions of the delta and core as follows:

(i) If the core and delta points are aligned in the vertical direction, then the fingerprint is a tended arch,

(ii) If the delta point is to the left of the core point, then the fingerprint is a right loop.

(iii) Else if the delta point is to the right of the core point, then the fingerprint is a left loop.

(3) Else if there are exactly 2 core points and exactly 2 delta points, then the fingerprint is whorl.

One may ask what about the other six possibilities such as (a) 0 delta and 1 core, (b) 1 delta and 0 core, (c) 1 delta and 2 cores, (d) 2 deltas and 1 core, (e) 0 delta and 2 cores, or (f) 2 deltas and 0 core. These cases are not normally applicable to the five-class problem. Therefore, if any one of these possibilities occurred we would reprocess or reject the sample. In this research, however, it is the role of the classification system to discern and resolve such inconsistency.

We have developed a Fingerprint Features Extraction (FFE) algorithm based on the structural approach. The basic idea of the proposed FFE method is to deduce from the fingerprints a directional field image to locate the singular-points (SPs), and then create a feature-encoded vector. The feature extraction technique, checks the orientations of individual pixels, computes directional fields using overlapping blocks in the image, and then detects the SPs to classify the fingerprint into the required classes. Determination of directional fields in fingerprints is a fast method for estimation of, and detection of the singular points, because ridge orientations, and the directional field, describe the coarse structure of a fingerprint. In summary, the feature extraction is comprised of the following steps:

- Segmentation of the image,
- Estimation of Directional Field,
- Extraction of Singular Points,
- Encoding of Feature Vector.

4.4.1 Segmentation of Fingerprint Image

Segmentation is a process to isolate features, from an image sample and is often the key step in interpreting the image. It is a process in which regions or features sharing similar characteristics are identified and grouped together. Image segmentation may use statistical classification, thresholding, edge detection, region detection, or any combination of these techniques [Perona98]. The output of the segmentation step is usually a set of elements, such as regions or boundaries. Most segmentation techniques are either region-based or edge-based. Region-based techniques rely on common patterns in intensity values within a cluster of neighbouring pixels [Hong00]. The cluster is referred to as the region, and the goal of the segmentation algorithm is to group regions according to their anatomical or functional roles. Edge-based techniques rely on discontinuities in image values between distinct regions, and the goal of the segmentation algorithm is to accurately demarcate the boundary separating these regions [Gouet00].

Thresholding is the simplest way to perform segmentation, and it is used extensively in many image-processing applications. Thresholding is based on the notion that regions corresponding to different feature types can be classified by using a range function applied to the intensity values of image pixels. The assumption is that different feature types will have a distinct frequency distribution and can be discriminated on the basis of the mean and standard deviation of each distribution. For example, given a two-dimensional image we can define a threshold rule to isolate different regions. A suitable frequency distribution *f(I)* of intensity values in a grey image can be used to threshold into black and white image, especially where a bimodal histogram can be obtained. In this research, threshold of the grey-level images to black and white is carried out, using either a Regional Average Thresholding (RAT) scheme or a General Threshold (GT) scheme. In the GT scheme, the process of binarising of the grey-level image to a black and white image is carried out by comparing each numeric pixel of grey-level image with a fixed number called a threshold level T. If the pixel intensity is less than the threshold level, the pixel value is set to zero; otherwise it is set to *255*. The thresholding scheme can be expressed as follows:

$$P(i,j) = \begin{cases} 255 & \text{if } I(i,j)>T \\ 0 & \text{if } I(i,j)<=T \end{cases} \qquad (i=0,1...,N, j=0,1...,M) \qquad (4.17)$$

Where *I(i,j)* indicate the original image, *P(i,j)* indicates the output binary image, *T* is the threshold level, and (*i= 0,1,...,N, j = 0,1,...,M*) are the rows and columns in the image respectively. To decide *T,* we usually need to apply histogram stretching of the grey scale images.

A histogram is a graph of the relative occurrence of grey levels in an image. The luminance spectrum is shown on the horizontal axis of the graph, ranging from zero luminance (pure black) to full luminance (pure white). The vertical axis of the graph indicates the proportion of the image's pixels that matches each point on the luminance spectrum. The histogram of an image can be calculated easily as follows:

$$Hist(r_k) = \frac{n_k}{n} \qquad (k = 0,1,2,\ldots, L-1) \qquad (4.18)$$

Where, *Hist(r*$_k$*)* is the discrete histogram function, r_k is the k^{th} grey level, n_k is the number of pixels in the image with that k^{th} grey-level, *L* is the total number of grey-levels and *n* is the total number of pixels in the image.

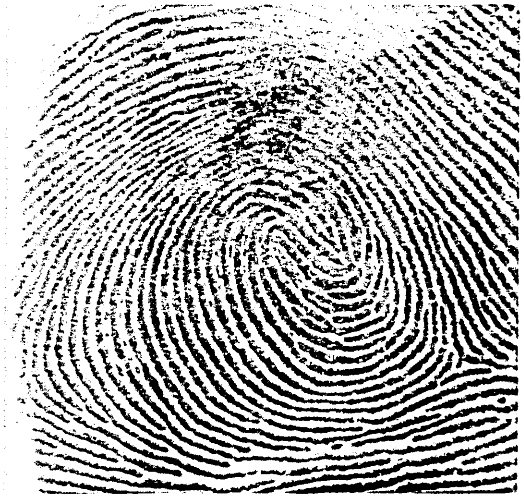


Figure 4.6 An original fingerprint image

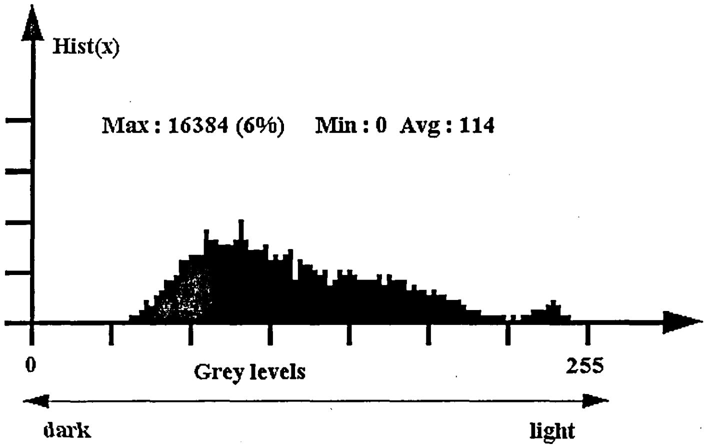


Figure 4.7 Histogram of fingerprint image shown in Figure 4.6.

Histogram stretching is a commonly used grey-level scaling algorithm, which automatically scales all the grey-levels within an image to the minimum and maximum grey-levels used in the image. It increases the brightness and contrast of the image. For *8-bit (256* grey-level) images, the following equation gives a linear transformation for histogram stretching:

$$g_i = \frac{255}{\mathrm{I}_{\max} - \mathrm{I}_{\min}}(\mathrm{I}_i - \mathrm{I}_{\min}) \qquad (4.19)$$

Where, g_i is any element of the scaled output image, I_i is the corresponding input image element, $\mathrm{I}_{\max}$ and $\mathrm{I}_{\min}$ are maximum and minimum grey-level values in the input image respectively. Histogram stretching is a reversible image processing application, such that, if the maximum and minimum grey-level values of the original image are known, the original image can be reconstructed from the scaled image using the following equation (4.20). Figure 4.6 shows an original image and Figure 4.7 shows its histogram.

$$\mathrm{I}_i = \frac{\mathrm{I}_{\max} - \mathrm{I}_{\min}}{255} + \mathrm{I}_{\min} \qquad (4.20)$$

Applying threshold value to the whole image, as in the GT scheme, may cause some feature loss. That is because; the average grey-level is not always the same in different parts of the original image. This is an important consideration in fingerprint images, which can be affected by different skin types. Regional Average Thresholding (RAT) [Emiroglu97] is a threshold scheme proposed for fingerprint images to overcome the problem of the GT. The original image may be divided into small square sized regions such as *8*x*8* region size windows.

Fingerprint images are first divided into any one of 16x16, 16x8, 8x8 or 8x4 regions (window sizes), and thresholding is done on each region by using the grey level average of the related region. In a RAT scheme, a window of 4, 8 or 16 pixel squares scans the image starting from the left hand corner of the image at the bottom. An average threshold level is calculated within the current window but only applied to the first half of the current window. The window then moves by one half of a window-size to the adjacent square. This time again, the leftmost half of the image window is thresholded, although the average threshold level is calculated for the whole window. The process continues until the entire image is thresholded in this way. Since the average threshold levels are calculated regionally, many more of the features are preserved in comparison with global thresholding. This stage also eliminates the fields that contain no information on the edges of the fingerprint.

Thresholding is done within each region by using the grey-level average of the window, given by:

$$T = \frac{1}{N^2} \sum_{i=0}^{N} \sum_{j=0}^{N} I(i,j) \quad . \qquad (4.21)$$

Where, T is the average grey level of the window, N is the size of each region, and $I(i,j)$ are the pixel intensities. Example of applying the RAT technique to a fingerprint image from NIST-4 database is shown in Figure 4.8, which shows the test fingerprint image and the thresholded image, using 16x8 window size.

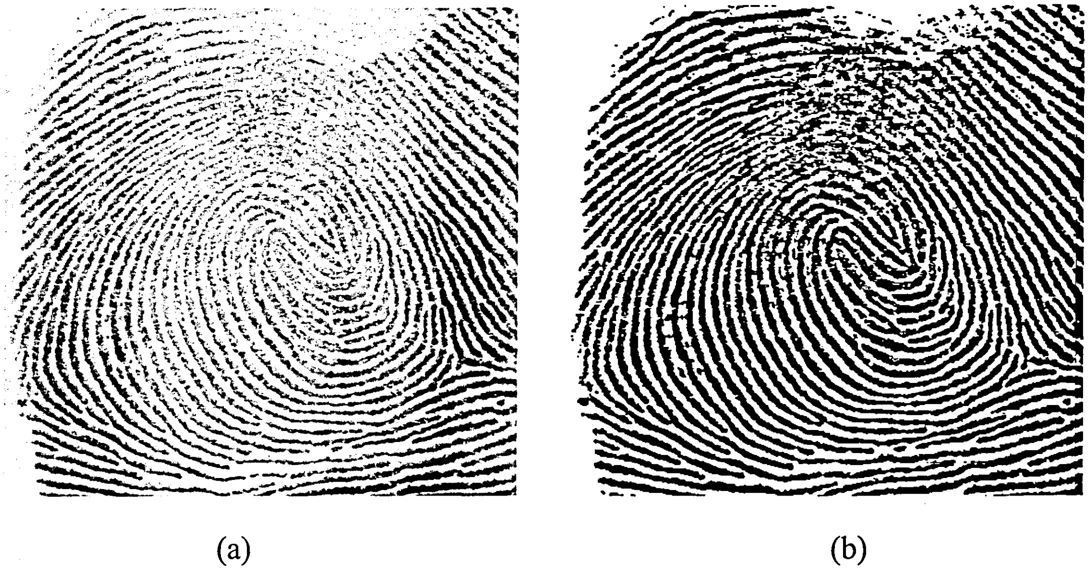


Figure 4.8 (a) The test image, (b) Thresholded image using 16x8 region size.

4.4.2 Directional Image Estimation

In order to extract the singular points, it is important to calculate the directional image using the pixel intensities of the thresholded images. A directional field describes the coarse structure of a fingerprint. It describes the local orientations of the ridge and valley structures. In general, the directional field at some location in the image is estimated by averaging the directions in some window around the desired location. A common method is the use of gradients [Asker00] (presented early in this Chapter). The local ridge-valley orientation is then perpendicular to the average gradient direction. The gradients are distributed over a cyclic space, ranging from 0 to

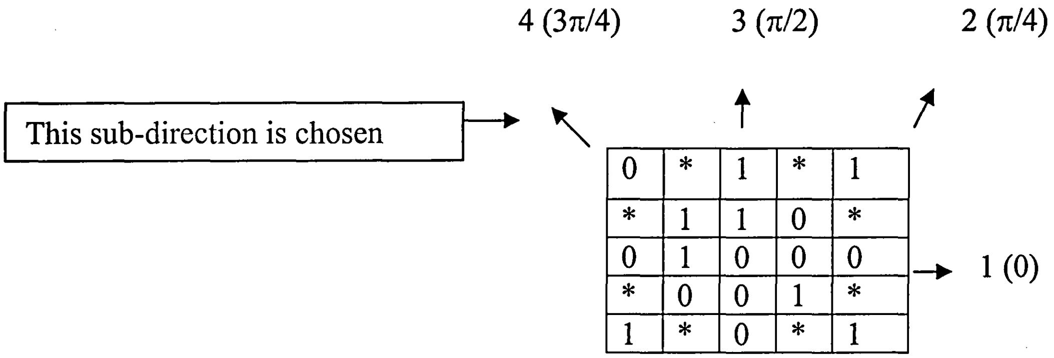
2π. However, the ridge-valley orientations have no direction, and therefore, they are distributed from 0 to π. The standard method to estimate the average orientation is given in [Jain99c] and [Candela95]. This method is said to solve most of the problems that originate from the cyclic space and the direction-less orientations, such as, strong orientations that have a higher contribution in the average orientation than weaker orientations [Asker01].

We have proposed in this research a new method of directional field estimation based on the concept of a directional image developed in [Mehtre98]:

$$V(i,j) = Min\left(\sum_{m=1}^{n} \left|p(i,j) - p_d(i_m, j_m)\right|\right) \tag{4.22}$$

Where, $V(i,j)$ is the directional value at the point (i,j), $p(i,j)$ is the binary value at the point (i,j), and $p_d(i_m,j_m)$, $(m = 1, 2, ..., n,$ and $d = 0,1,..., N-1)$ is the average of the pixel values in a particular direction, n is the number of pixels chosen for this computation in each direction, and N is the number of directions. This directional image is known as the pixel-wise directional image.

The sub-direction with the least variations is chosen as dominant ridge direction of the block. Considering Figure 4.9, the values of p_d for the different sub-directions are: $p_1 = 0.2$, $p_2 = 0.4$, p_3=0.4 and p_4=0.6. The sum of the absolute differences are $v_1 = 4.0$, $v_2 = 3.0$, $v_3 = 3.0$ and $v_4 = 2.0$. For this example, therefore, sub-direction 4 is the dominant direction as it has the smallest sum of the absolute differences value. In cases where more than one dominant sub-direction is found we re-evaluate Equation (4.22) for a smaller 3x3 block, however, if more than one dominant sub-direction is also found in the new block, we may choose any one of them. This will be most likely the case of an accident sample of fingerprint, or due to a noisy sample.

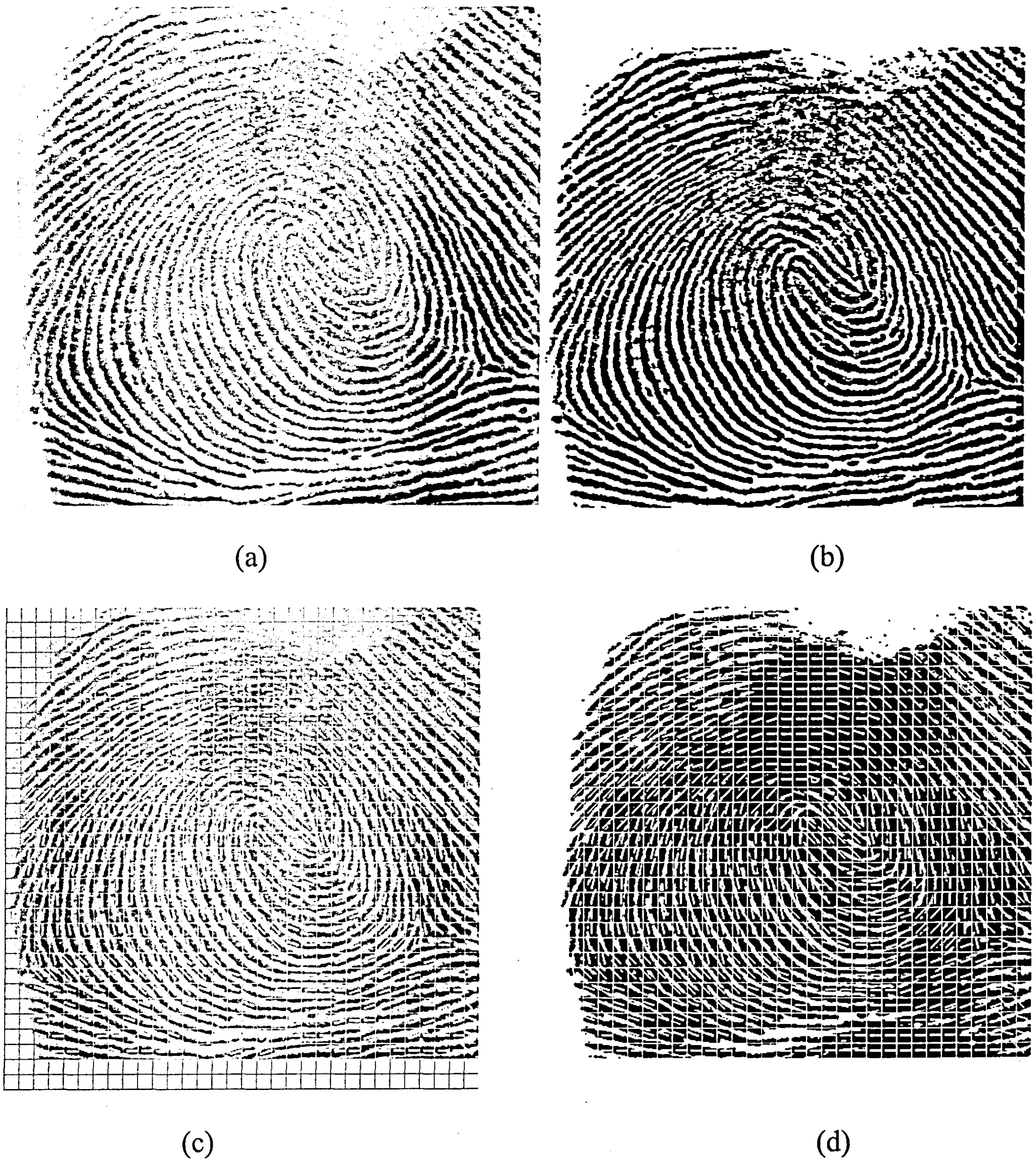


0	*	1	*	1
*	1	1	0	*
0	1	0	0	0
*	0	0	1	*
1	*	0	*	1

Figure: 4.9. 5x5 mask with different pixel values.

Directional field estimation was carried out by scanning a 5x5 window over the thresholded image; and replacing each central pixel by the estimated direction as described above. The directional image creates an MxN reduced image *R*, which decreases the complexity and increases the speed of the processing. The logic behind the directional method is that a peak in the histogram of a directional image in a region indicates that there exists a clear ridge, because a ridge-line results in points of the same direction in the region. That is, if a clear ridge exists in a region, it expressly means it is foreground, and it gives rise to peak in the histogram.

The limitation of the directional field method is that in perfectly uniform region, $p_d = p(i_m, j_m)$ for m varying in any direction, hence the directional field becomes undefined because there will be no dominant direction. However, the directional criterion is very good for low contrast and noisy images, besides giving good results for modest quality fingerprint images.



(a) (b)

(c) (d)

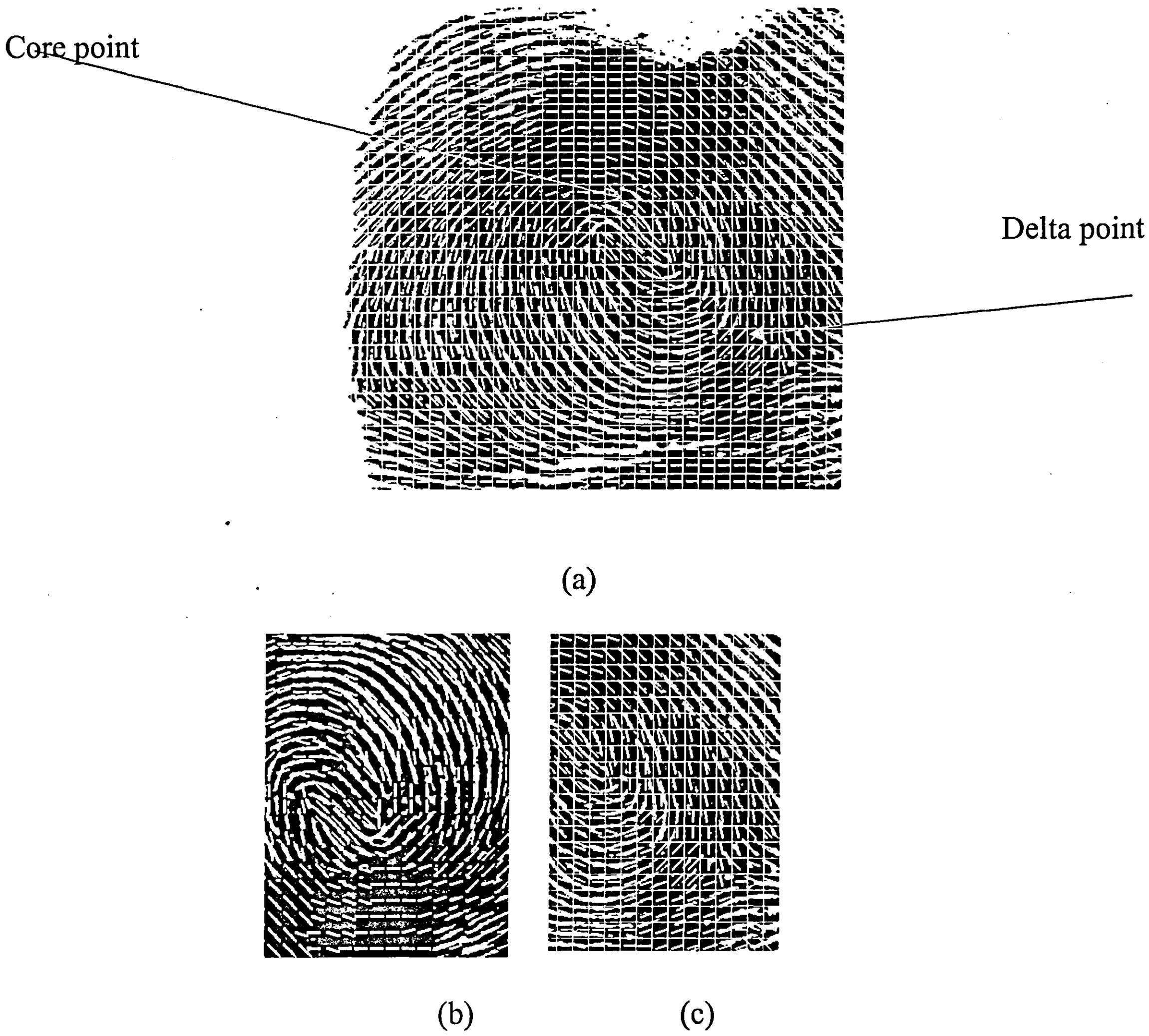
Figure 4.10 (a) Original image (b) thresholded image (c) directions are mapped on the original image, (d) directions are mapped on the thresholded image.

4.4.3 Image direction estimation

The image direction is the approximate orientation of the whole fingerprint pattern. To estimate the image direction; first the centre of the directional fingerprint pattern is located, and a 5x5 window placed at this centre. The mode or mean of the orientations in the eight neighbouring windows is then determined. This is the estimate of the entire image orientation. The image direction can also be estimated with different sizes of windows and neighbourhoods.

4.4.4 Singular points extraction

The singular points (namely the delta and the core) are shown as discontinuities in the directional image, shown in Figure 4.11. Delta points lie on a ridge at or in front of, and nearest the centre of the divergence of the type lines. Core points lie in the approximate centre of the finger impression.



(a)

(b) (c)

Figure 4.11: (a) Singular points on fingerprint (b) directional field for a core point; and (c) directional field for a delta point.

The method presented for SP extraction is based on the *Poincare* index [Tojo84]. In the DF, *R* is an *M*x*N* image, where R(i,j) represents the local ridge orientation at pixel (i,j). A counter-clockwise closed contour in the DF around a core results in a cumulative change of π in the orientation, while carrying out this procedure around the delta results in -π. On the other hand, when applied to a location that does not contain a SP, the cumulative orientation change will be zero. Let $\Psi_x(.)$ and $\Psi_y(.)$ represent the *x* and *y* coordinates of a closed digital curve with N_Ψ pixels. The *Poincare* index at pixel *(i,j)*, which is enclosed by the digital curve can be computed as follows:

For each point *k* along the curve

$$\delta(k) = R(\Psi_x(k+1), \Psi_y(k+1)) - R(\Psi_x(k), \Psi_y(k)) \qquad (4.23)$$

$$\Delta(k) = \begin{cases} \delta(k), & if|\delta(k)| \quad < \pi/2, \\ \pi + \delta(k), & if\delta(k) \quad \leq -\pi/2, \\ \pi - \delta(k) & otherwise, \end{cases} \qquad (4.24)$$

$$Poinc\,Poincare(i,j) = \frac{1}{2\pi}\sum_{k=0}^{N_\Psi}\Delta(k) \qquad (4.25)$$

The size of the closed digital curve is crucial to the performance of a singular point detection algorithm using the *Poincare* index. If it is too small, then a small perturbation of orientations may result in spurious singular points being detected. On the other hand, if it is too large, then a true pair of core and delta, which are close to each other, may be ignored because the *Poincare* of a digital curve that includes an equal number of cores or deltas is 0.

In an orientation field, the *Poincare* index of a core-shaped singular point has a value of (1/2) and the *Poincare* index of delta-shaped singular point has a value of (-

1/2) [Hong99]. Although, the *Poincare* index provides means for consistent detection of singular points, the question may arise how to calculate this measure. A part from the problem of the efficiency of calculating cumulative orientation changes over the contour, a choice has to be made on the optimal size and shape of contour. However, many researchers including [Jain00], [Hong99] have suggested that the Poincare index is the most reliable algorithm compared to other existing algorithms for singular-points detection.

The singular point detection algorithm uses a closed square (8x8) with a length of *28* pixels. We empirically determined that a curve of *28* pixels is a good trade-off between detection and misses of singular points. The extractor routine starts scanning the image from the bottom left hand corner of the image which is accepted as the origin point. Let *R* be the orientation field. The main steps in the singular points detection algorithm are as follows [Hong99], [Jain99c]:

1. Initialise *Q*, a label image used to indicate the singular points,
2. For each pixel *(i,j)* in *R*, compute the *Poincare* index and assign the corresponding pixel in *Q* a value *1* if the *Poincare* index is *(1/2)* and a value *2* if the *Poincare* index is *(-1/2)*,
3. Find each connected component in *Q* with pixel value *1*. If the area of connected component is larger than *7*, a core is detected at the centred of the connected component. If the area of the connected component is larger than *20*, then two cores are detected at the centred of the connected component,
4. Find each connected component in *Q* with pixel values *2*. If the area of the connected component is larger than *7*, a delta is detected at the centred of the connected component,

5. If more than two cores or more than two deltas are detected, we let the system the system accepts as two cores or deltas. The heuristic that at most two cores and two deltas exist in a fingerprint is assumed true [Jain99b].

- Core direction estimation: to compute the core-point direction, a 5x5 block is, centred at the detected core point. The mode or median direction is assigned as the direction for the core point. In the case that two cores are detected, the dominant direction for the core point closest to the centre of the image will be assigned as the centre direction.
- Delta position estimation: to compute the position of a delta relative to a core the following procedure is adopted:
  1. Estimate the symmetric axis which crosses the core in its local neighbourhood,
  2. Compute the angle, α, between the line segment from the core to the delta and the symmetric axis,
  3. Compute the average angle difference, β, between the line segment from the core to the delta and the local ridge orientation of the centre of the core,
  4. If ($\alpha < 15^{o}$) or ($\beta < 15^{o}$), then the delta position is vertical to the core position,
  5. Otherwise the delta position is to the left or right of the core position,
  6. If more than one core or more than one delta are detected the above process is repeated for each core-delta combinations. The value of "both" is assigned if two different relative positions are observed.

Figure 4.12 illustrates symmetric axis which crosses the core in its local neighbourhood.

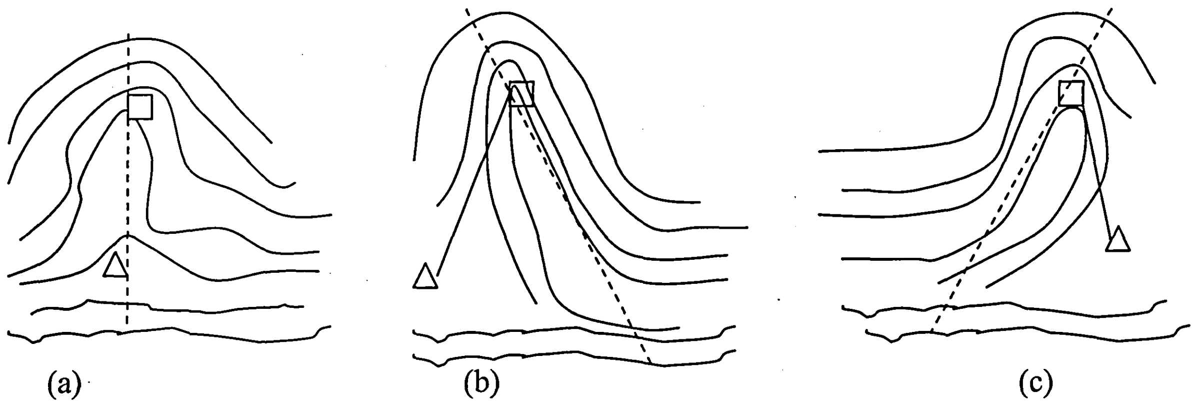


Figure 4.12 The dashed lines are the symmetric axis in (a) tended arch, (b) right loop and (c) left loop prototypes respectively.

4.4.5 Feature Encoder

The feature vector is a list of singular points plus their relative position or direction and image direction estimation with accompanying attribute values. The details of the features of interest are the following:

I- Number of deltas, *DeltaNo,* computed using the *Poincare* index, (equation *(4.23), (4.24) and (4.25)).*

II- Number of cores, *CoreNo*, computed using the *Poincare* index, (equation *(4.23), (4.24) and (4.2*5)).

III- Image direction estimation, *ImageDir,* described in section 4.4.3.

IV- Core direction estimation, CoreDir, described in section 4.4.4.

V- The delta position estimation, *DeltaPos*, also described in section 4.4.4.

The typical features for the five classes of the fingerprints are as shown in Table 4.1. The process of feature extraction, however, produces feature vectors that are often different from the expected values. While this can be verified by human inspection in manual classifications systems, it is not easily so in automatic classification systems. It is for this reason that robust classifiers, such as neural networks have been proposed in this research. Figure 4.13 shows a flowchart of the feature extraction scheme.

True Type	***DeltaNo***	***CoreNo***	***ImageDir***	***CoreDir***	***DeltaPos***
A	0	0	1 0)	3 (π/2)	2 (right)
T	1	1	3 (π/2)	3 (π/2)	1 (vertical)
W	2	2	3 (π/2)	2 (π/4)	4 (both)
R	1	1	4 (3π/4)	4 (3π/4)	2 (right)
L	1	1	2 (π/4)	2 (π/4)	3 (left)

Table 4.1 Typical desired features for different classes

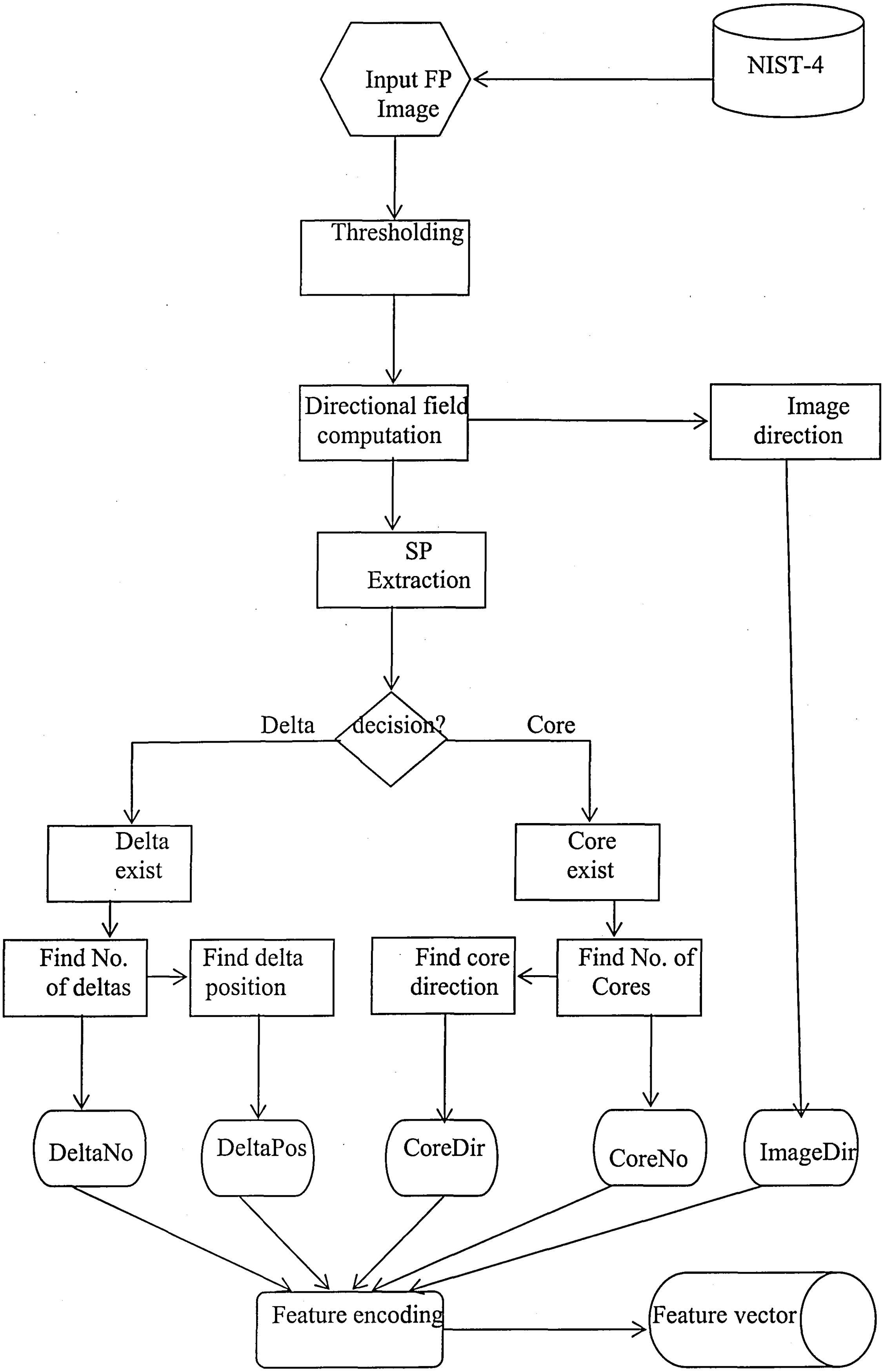


Figure: 4.13. Flow-chart of calling routines to perform the FFE.

4.5 Conclusion

This Chapter has discussed the problem of feature extraction, in general, by reviewing the different techniques of feature extraction. Different approaches of detecting basic features such as points, edges, and regions were described. Some image processing techniques for fingerprint images have been discussed with respect to the proposed FFE. Fingerprint feature extraction is automatic algorithm, which includes thresholding the input image, image directional field estimation, singular points extraction, and feature vector encoding. The next Chapter discusses feature vector analysis and a classification algorithm

Chapter 5

Fingerprint Features Classification Using Neural and Fuzzy-Neural Networks

5.1 Introduction

This Chapter describes the process of classification of the feature vectors extracted and encoded, as presented in Chapter 4. The statistical properties of the feature vectors are also examined, using the Statistical Package for Social Sciences (SPSS), to determine the suitability of the extracted features for classification of fingerprints. The implementation of features classification is carried out using two different neural network classifiers and one fuzzy-neural network classifier.

Neural networks, allow many type of problems to be solved by simply feeding large amounts of 'raw data' (e.g. images, sound signals, stock market index ranges) to a neural network. During training, the network learns the relationship between the features, and in many such situations, neural networks offer the best solution, especially when dealing with non-linear and complex problems. Fuzzy-neural networks are hybrid systems, which possess the advantages of both neural networks and fuzzy systems [Chen93]. In the former, learning abilities, optimisation abilities, and connectionist structures; and in the later human-like reasoning capabilities and ease of incorporating expert knowledge [Lin99]. In addition, the hybrid systems can alleviate the shortcomings of the respective techniques. These include common problems encountered in the design of fuzzy systems, such as, the determination of the membership functions; the identification of the fuzzy rules as well the operation of fuzzy inferences, which can be resolved using neural network techniques. One well acknowledged drawback of neural networks is their opaqueness. The integration of fuzzy concepts, in fuzzy-neural systems greatly improves the transparency for a better understanding of their inner working.

The rest of this Chapter is organised as follow. Section 5.2 presents an overview of SPSS and the statistical analysis of the features data. Section 5.3 discusses background overview of neural network, fuzzy logic and fuzzy-neural network techniques. Section 5.4 discusses implementation of Multi-Layer Perceptrons (MLP) and Radial Basis Function (RBF) network classifiers. Section 5.5 presents implementation of a fuzzy-neural classifier, incorporating automatic network construction provided by Neufuzzy^{TM} software. Finally, Section 5.6 gives some conclusions.

5.2 Statistical Analysis of Features Data

Statistical analysis of feature data was carried out to determine the distribution of the feature vectors, and hence the ability to distinctly separate them into five classes. The purpose of this analysis is to identify the level of overlap in the features space, and justify the usefulness of selected features for the purpose of classification. The SPSS statistical package was used [Samuel00].

SPSS can perform a variety of data analysis and presentation functions, including statistical analyses and graphical presentation of data. Among its features are modules for statistical data analysis, including descriptive statistics, such as plots, frequencies, charts, and lists. It also uses sophisticated inferential and multivariate statistical procedures, such as Analysis of Variance (ANOVA), factor analysis, cluster analysis, and categorical data analysis.

Since the feature data in this research is comprised mainly of discrete, and qualitative values (represented as discrete quantities), it is beneficial to carry out a histogram (frequency) analysis rather than clustering to evaluate the degree of overlap of feature vectors. Tables 5.1 through 5.5 show the occurrence frequencies of feature values for the five fingerprint classes. This also shown in Figures 5.1 through 5.5 as feature histograms. A cluster distribution of all feature vectors, determined by SPSS, is shown in Figure 5.6.

Valid	Frequency	Percent	Valid Percent
DeltaNo=0	664	83	100
DeltaNo=1	136	17	0
CoreNo=0	720	90	100
CoreNo=1	80	10	0
ImageDir=1	786	98.25	100
ImageDir=2	14	1.75	0
CoreDir=2	4	.5	0
CoreDir=3	796	99.5	100
DeltaPos=1(Vertical)	696	87	100
DeltaPos=2(Right)	104	13	0
No. of fingerprints	800	100	100

Table: 5.1 Arch

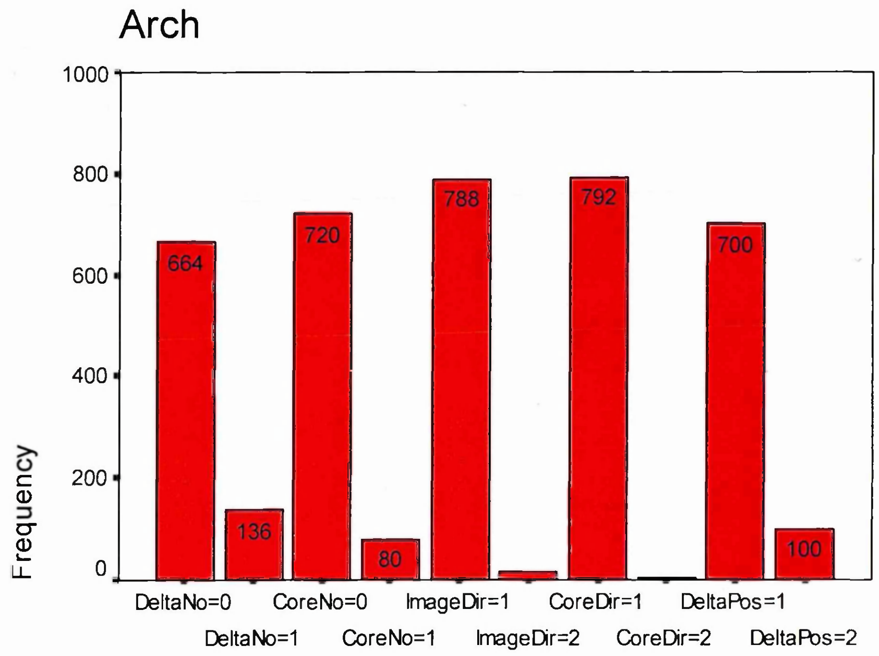


Figure: 5.1. Feature Histogram for Arch-type

Valid	Frequency	Percent	Valid Percent
DeltaNo=0	36	4.5	0
DeltaNo=1	764	95.5	100
CoreNo=0	123	15.37	0
CoreNo=1	677	84.63	100
ImageDir=2	7	.87	0
ImageDir=3	793	99.13	100
CoreDir=2	4	.5	0
CoreDir=3	796	99.5	100
DeltaPos=1(Vertical)	197	24.63	0.0
DeltaPos=2(Right)	593	75.37	100
No. of fingerprints	800	100	100

Table: 5.2 Tended-arch

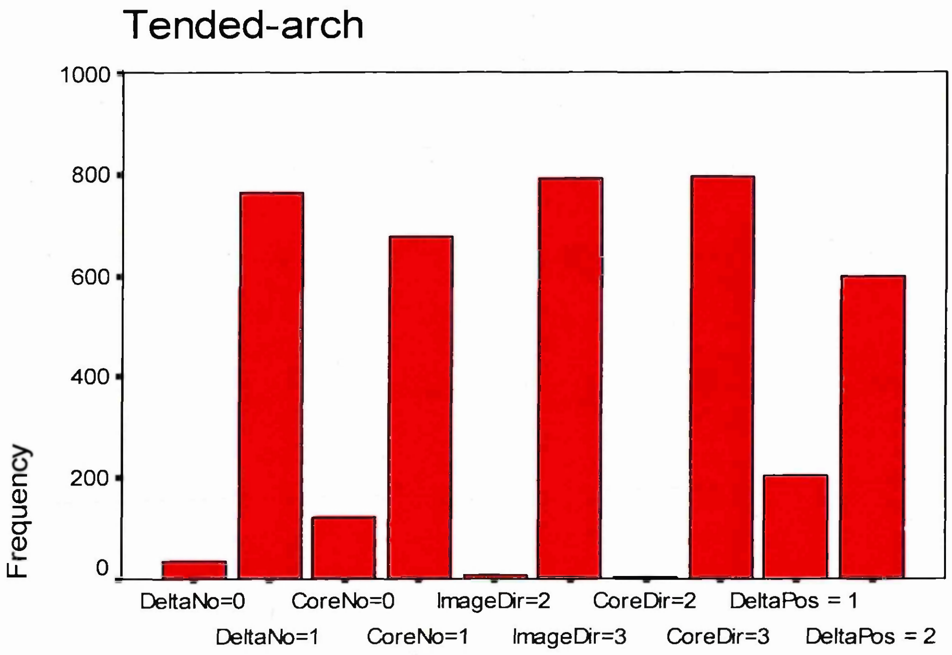


Figure: 5.2 Feature Histogram for Tended-arch-type

Valid	Frequency	Percent	Valid Percent
DeltaNo=1	6	.75	0
DeltaNo=2	794	99.25	100
CoreNo=1	111	13.87	0
CoreNo=2	689	86.63	100
ImageDir=3	784	98	100
ImageDir=4	16	2	0
CoreDir=2	796	99.87	100
CoreDir=4	5	.63	0
DeltaPos=3(Left)	67	8.38	0
DeltaPos=4(both)	733	91.62	100
No. of fingerprints	800	100	100

Table: 5.3 Whorl

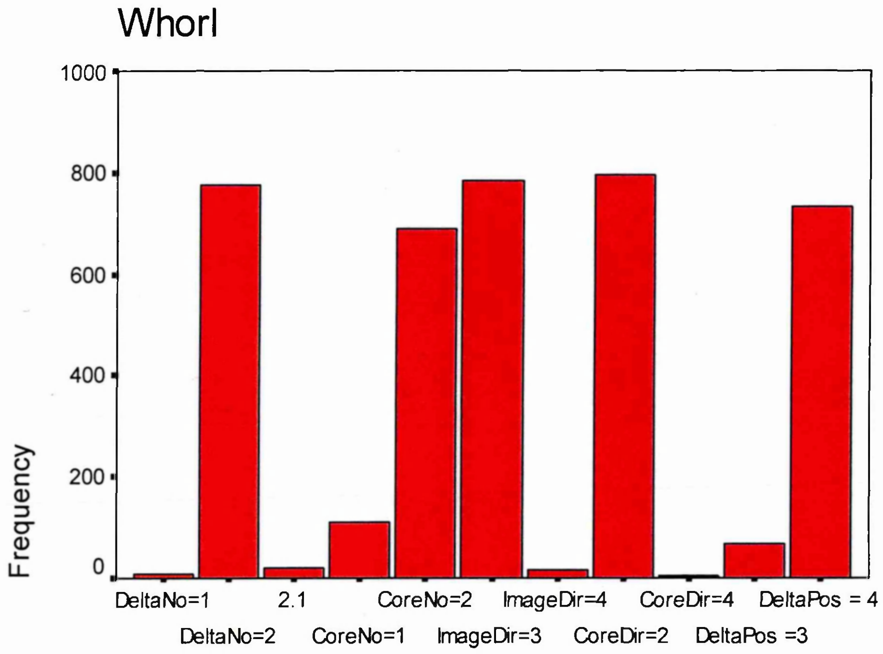


Figure: 5.3 Feature Histogram for Whorl-type

Valid	Frequency	Percent	Valid Percent
DeltaNo=0	18	2.25	0
DeltaNo=1	782	97.75	100
CoreNo=0	12	1.5	0
CoreNo=1	784	98	100
CoreNo=2	4	.5	0
ImageDir=3	137	17.13	0
ImageDir=4	663	82.87	100
CoreDir=3	3	.38	0
CoreDir=4	797	99.62	100
DeltaPos=1(Vertical)	271	33.88	0
DeltaPos=2 (Left)	529	66.2	100
Total	4000		

Table: 5.4 Right-loop

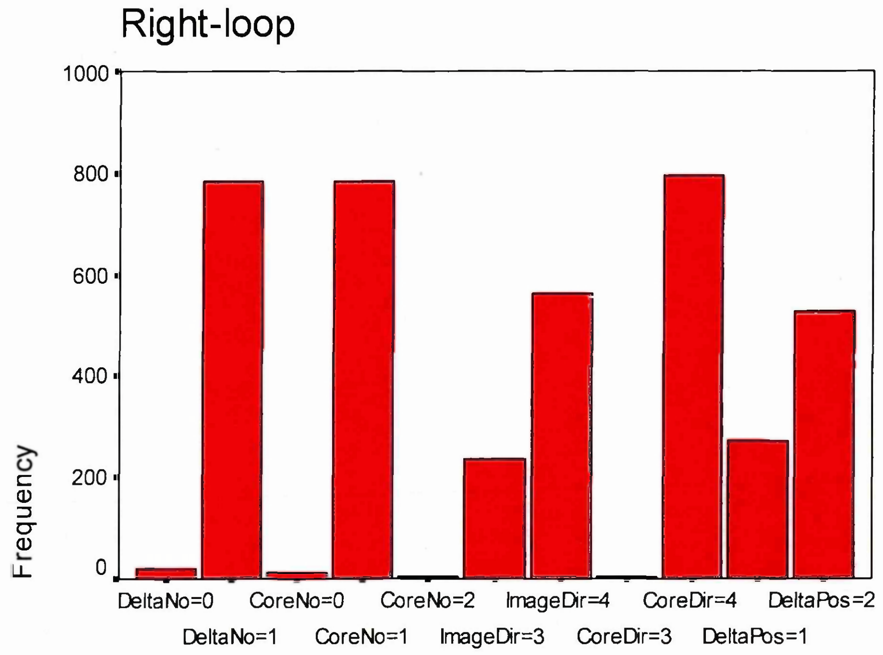


Figure: 5.4 Feature Histogram for Right-loop type

Valid	Frequency	Percent	Valid Percent
DeltaNo=0	20	2.5	0
DeltaNo=1	780	97.5	100
CoreNo=0	17	2.12	0
CoreNo=1	783	97.87	100
ImageDir=1	14	1.75	0
ImageDir =2	796	98.25	100
CoreDir=1	6	.75	0
CoreDir=2	794	99.25	100
DeltaPos is 1(Vertical)	88	11	0
DeltaPos=3(Right)	712	89	100
No. of fingerprints	800	100	100

Table: 5.5 Left-loop

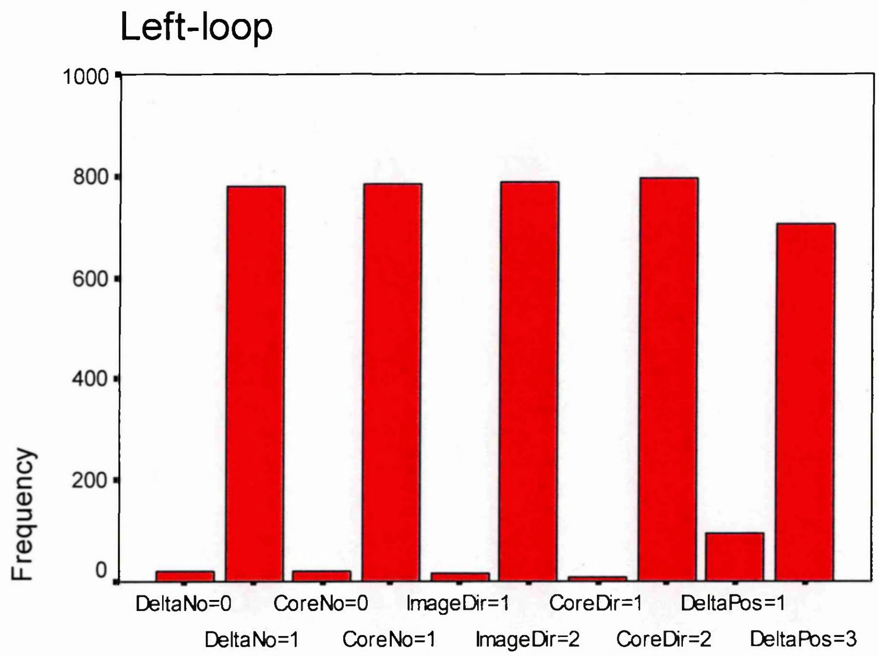


Figure: 5.5 Feature Histogram for Left-loop type

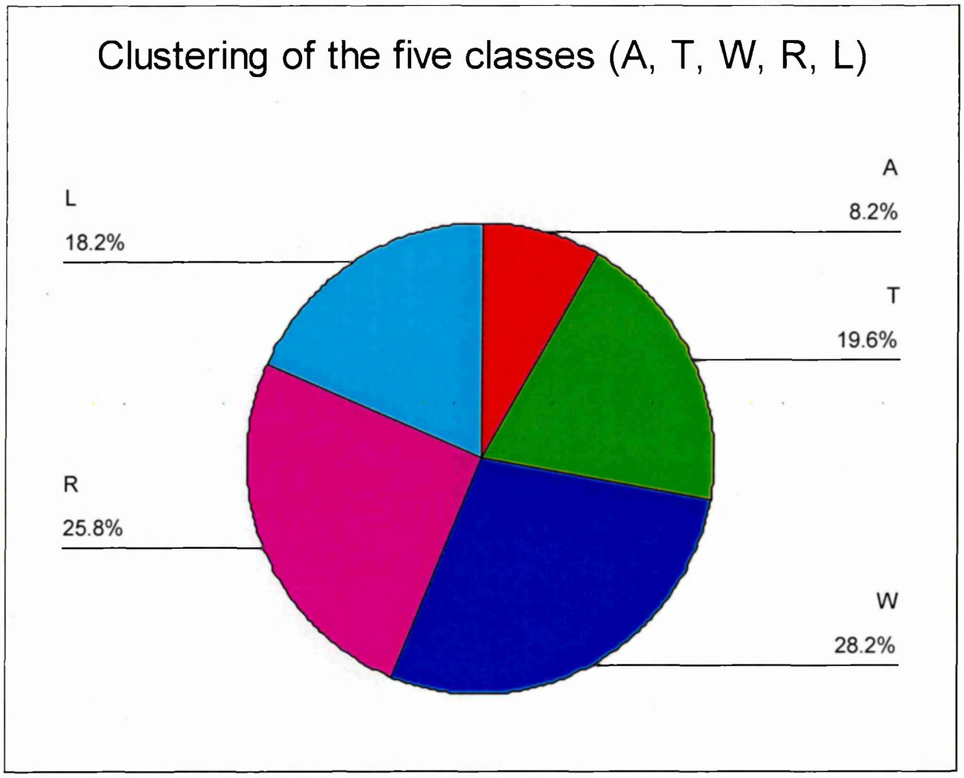


Figure: 5.6 Clustering distribution percentages of the five classes using SPSS.

Although, the expected statistical distribution is even among the five classes, the distribution shown in Figure 5.6 indicates that there is significant feature overlap, and thus potential for misclassification, to suggest that the features cannot be satisfactorily classified using simple techniques, such as, linear discriminant analysis or K-nearest neighbour. For example, the classification error in the case of A class is as high as 59%. It is for this reason it was proposed to investigate computational intelligence approaches, that is, neural and fuzzy-neural classifiers.

5.3 Background of Neural Network, Fuzzy logic and Fuzzy-Neural Techniques

5.3.1 Overview of Neural Networks Techniques

Neural networks represent a unique methodology by which knowledge is acquired from sets of training examples and stored in a distributed manner in the connectionist structure of the network. The distributivity contributes to increase learning capabilities of the neural networks because the individual elements in the network are capable of adjusting their connections to achieve near-optimal input-output mappings. Distributed learning is also advantageous because it permits a response to a novel situation to be inferred using knowledge of previously learned, similar but not exactly the same circumstances. Thus, the major advantage of Neural Networks (NNs) learning is the ability to accommodate poorly modelled, non-linear dynamical systems. The fundamental philosophy of NNs is to view a system as mapping between input states and the output states, with learning being regarded as the modification of this mapping to improve the system performance objective. However, distributed knowledge representation usually makes it almost impossible to come to a reasonable interpretation of the overall structure of the network in terms of humanly understandable concepts, such as an "if … then" symbolic framework [Linkens96].

Structurally, neural networks are organised as a number of input nodes, equal to the number of independent variables, and a number of output nodes, equal to the number of dependent variables. Between these input and output node levels is the third set of nodes, so-called the hidden nodes. These nodes are hidden in the sense that they do not directly interact with outside world. These three groups of nodes are referred to as the input level, hidden level, and the output level of nodes, respectively. Linking the input and hidden levels of nodes is a set of connections referred to as the input-to-hidden layer weights. Similarly, connecting the hidden and output levels of nodes are the hidden-to-output layer weights. A neural network is trained by presenting it with vectors of training exemplars, one at a time. For each exemplar, successively presented to the neural network, the network generates a set of output values. These outputs can be considered to be the guesses, or estimates, that the network makes as to what the values of the dependent variables for the particular training exemplar should be. Each output node represents one of the dependent variables of the relationship the network is to learn. At the start of network training the actual outputs of the network will, in general, not be close to the desired outputs for each of the training exemplars. The corresponding output generated by the network is compared with the target outputs. The difference, called the training error, is used as the basis for a scheme that modifies the network weights. The weights are modified according to a rule, typically, error backpropagation rule [Rumelhart86]. This rule is expected, over the course of training, to minimize the difference between the actual outputs of the network and the desired outputs.

Neural networks are a good methodology for data classification because of the following characteristics:

- Deal with non-linearities,
- Learn from data without an initial system model,
- Handle noisy or irregular data,
- Be easily and quickly updated,
- Interpret information from large number of variables or parameters,
- Provide generalised solutions.

As neural networks provide general techniques for modelling and pattern recognition, they have found applications in many fields, including business, finance, science and industry.

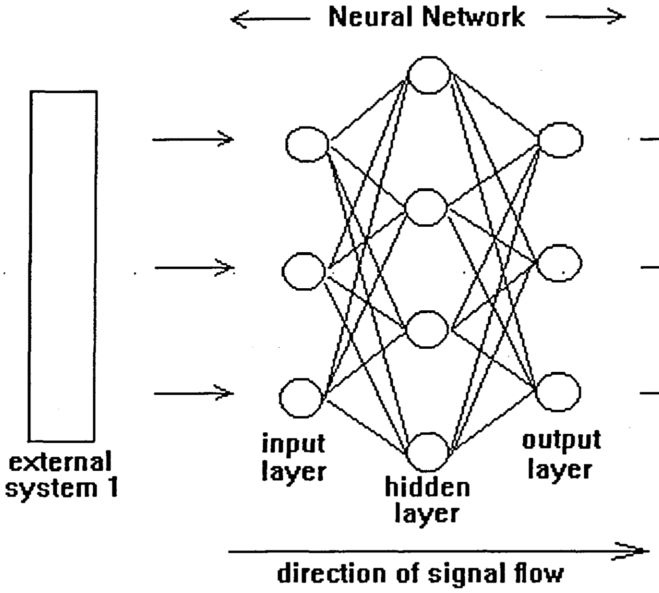


Figure 5.7. Feedforward neural network architectures.

There exist two primary types of neural network learning: supervised and unsupervised. Supervised learning is a process of training a neural network by giving it input-output examples of the task we want it to learn, i.e., learning with a teacher. This technique is mostly applied to feed-forward type of neural networks, such as, Multi-Layer Perceptron, Radial Basis Function and Hopfield networks. The most frequently used and effective supervised learning algorithm is the "Error Back-Propagation Algorithm". Figure 5.7 shows an example of a feedforward neural network architecture. Unsupervised learning is the process through which a network is able to discover statistical regularities in its input space and automatically develops different modes of behaviour to represent different classes of inputs. In practical applications 'labelling' is required after training, since it is not known at the outset which mode of behaviour will be associated with a given input class. Kohonen's self-organising (topographic) map neural networks, are used for this type of learning.

In order to understand the learning process of the main types of neural network we discuss, in brief, the learning process of a single perceptron, as an example of the basic development of NN. We shall then discuss multiplayer perceptons and the radial basis functions networks as examples of practical NN classifiers.

- The Perceptron

The Perceptron is a single processing unit (neuron) with incoming input lines and one output line, which works as a binary decision neuron. Each perceptron possesses a threshold value and each time an input pattern is presented to the perceptron, it computes a net weighted value from the pattern and compares it to the threshold. If the threshold is bigger than the net input then, a '0' value is given at the output, otherwise, a '1' is given as an output.

A single perceptron with two inputs can, learn a binary classification problem (presented in Figure 5.8), by moving the boundary line in the data plane until it gets to a position where the two classes are clearly separated. The perceptron algorithm simply moves a decision boundary in the data hyper-plane to classify all the data that belongs to a class on one side of the boundary and the rest on the other side.

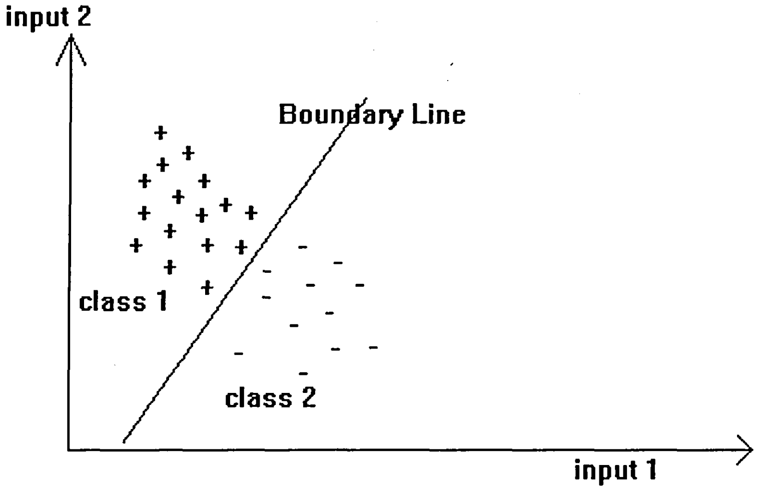


Figure: 5.8. Example of a 2-Dimensional Classification

After extensive research on perceptron applications and limitations, Minsky and Papert [Minsky69] found that the perceptron was incapable of solving a nonlinear classification problem, using the well-known XOR problem (Figure 5.9). This weakness of perceptrons showed neural networks to be unfavourable artificial cognitive systems, until networks of perceptrons with an input, output and at least one hidden layer, along with a variety of more sophisticated learning algorithms were developed and proven to be very powerful parallel-distributed processors.

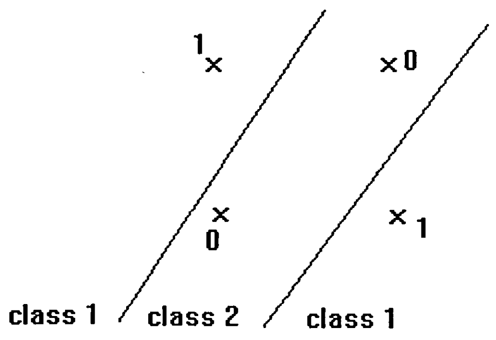


Figure: 5.9 The XOR problem nonlinear classifications.

- Multi-Layer Perceptrons (MLP)

Typically, the MLP network consists of a set of sensory nodes that constitute the input layer, one or more hidden layers of computation nodes, and an output layer. The input signal propagates through the network in a forward direction, on a layer-by-layer basis. The network architecture is set before training and training adjusts only the internal links so that the response from the net model conforms better to the proper response. This is an iterative process, that can take many passes of the data through the network before the model is good enough. Querying (operation) is a fast process that simply passes the data through the model and retrieves the answer. MLP are one of the most used of neural networks because they are robust and very good for general classification problems. MLP have been applied successfully to solve difficult and diverse problems by training them in a supervised manner, with the highly popular error Back-Propagation (BP) algorithm [Werbos74]. Basically, the error BP process consists of two passes through the different layers of the network: a forward pass and a backward pass. In the forward pass, an input vector is applied to the sensory nodes of the network, and its effect propagates through the network, layer by layer. Finally, a set of outputs is produced as the actual response of the network. During the forward pass the synaptic weights are all fixed. In the backward pass, on the other hand, the synaptic weights are all adjusted in accordance with an error-correction rule. Specially, the actual response of the network is subtracted from a desired response to reduce an error signal. This error signal is then propagated backward through the network, against the direction of the synaptic connections. The synaptic weights are adjusted so as to make the actual response of the network move closer to the desired response. It is for this reason they are sometimes referred to "back-propagation" neural networks. The model of each neuron in the network includes nonlinearity at the output end. A commonly used form of nonlinearity that satisfies this requirement is a sigmoidal nonlinearity defined by the logistic function:

$$y_i = \frac{1}{1+\exp(-v_j)} \qquad (5.1)$$

Where v_j is the net input to the neuron j, and y_j is the output of the neuron. The presence of nonlinearities is important because, otherwise, the input-output relation of the network could be reduced to that of a single-layer perceptron. Moreover, the use of the logistic function is biologically motivated, since it attempts to account for the refractory phase of real neurons [Pineda88].

The basic BP algorithm is simply an application of gradient descent methods. One of the things to note about the algorithm is that it requires the outputs to be continuous, monotonic, and differentiable functions of their inputs. The key steps of the Backpropagation algorithm, as described in Rumelhart et. al. [Rumelhart86], are outlined as follows:

The error in the output neurons is given by,

$$(t_k - y_k) \tag{5.2}$$

In addition, the contribution to the error by the preceding hidden layers is given by,

$$\delta\kappa = (t_k - y_k) f'\left(\sum_{j=1}^{l} z_j w_{jk} \right) \tag{5.3}$$

Where t_k is the target output, y_k is the actaul output, $f'(\ )$ is the derivative of the activation function, z_j is the output at j^{th} hidden neuron and w_{jk} is the connection weight between the hidden and the output neurons.

The weight correction for the output layer is given by,

$$\Delta w_{jk} = \eta \delta_k z_j \tag{5.4}$$

Where η is called the learning rate.

The errors in the hidden layer are given by,

$$\delta_j = \sum_{k=1}^{m} \delta_k w_{jk} f' \sum_{i=1}^{n} x_i w_{ij} \tag{5.5}$$

Where m is the number of units in the output layer, n is the number of inputs, x_i is the i^{th} input and w_{ij} is the connection weight between the i^{th} input and the jth hidden neuron. The weight correction for the hidden layer is then given by,

$$\Delta w_{ij} = \eta \delta_j x_i \qquad (5.6)$$

When a momentum term is used, the weight update for both layers is given by,

$$w_{jk}(t+1) = w_{jk}(t) + \Delta w_{jk} + \beta(w_{jk}(t) - w_{jk}(t-1)) \qquad (5.7)$$

Where β is the momentum term.

- Radial Basis Function (RBF)

Radial basis function networks are also feed-forward, but have only one hidden layer. Like MLP, RBF networks can learn arbitrary mappings: the primary difference is in the hidden layer. RBF hidden layer units have a receptive field, which has a centre, a particular input value at which they have a maximal output. Their output tails off as the input moves away from this point. Generally, the hidden unit function is a Gaussian transformation. The construction of the RBF network in its most basic form involves three entirely different layers. The input layer is made up of source nodes. The second layer is a hidden layer of high enough dimension, which responds to different regions of the input space. The output layer supplies the response of the network to the activation patterns applied to the input layer. The transformation from the hidden-unit space is linear. Generally, the centres and standard deviations are decided by examining the distribution of vectors in the training data. The output layer weights are then trained using the Delta rule [Cover65]. RBFs have the advantage that one can add extra units, without disturbing the previous training.

RBF networks are trained by deciding on how many hidden units there should be, deciding on their centres and the standard deviations of their Gaussians and training the output layer.

When a standard RBF is used to perform a complex pattern classification task, the problem can be solved by transforming it into a high-dimensional space in a nonlinear manner. The underling justification for so doing is provided by Cover's theorem on the separability of patterns, which states that a complex pattern classification problem cast in high dimensional space nonlinearly is more likely to be linearly separable than in low-dimensional space [Cover65].

RBF and MLP are both universal approximators. However, these two networks differ from each other in several important aspects, [Simon94]:

1. An RBF has a single hidden layer, whereas MLP may has one or more hidden layers.
2. Typically, the computation nodes of an MLP located in a hidden or output layer, share a common neuron model, while the computation nodes in the hidden layer of an RBF serve a different purpose from those in the output layer.
3. The hidden layer of an RBF is nonlinear, whereas the output layer is linear. In both MLP layers are usually nonlinear.
4. The argument of the activation function of each hidden unit in an RBF network computes the Euclidean norm (distance) between the input vector and the centre of that unit, while the activation function of each hidden unit in an MLP computes the inner product of the input vector and the synaptic weight vector of that unit.
5. MLPs construct global approximations to nonlinear input-output. Consequently, they are capable of generalization in regions of the input space where little or no

training data are available, while RBFs using exponentially decaying localised nonlinearities (e.g. Gausian functions) construct local approximations to nonlinear input-output mapping, with the result that these networks are capable of fast learning and reduced sensitivity to the order of presentation of the training data.

5.3.2 Overview of Fuzzy Logic Techniques

The fundamental idea behind fuzzy logic is based on the observation that human thinking is not just two-valued or multi-valued logic, but logic with continuous degree of truth. A theory of fuzzy sets has been developed as a complete mathematical abstraction for representing and operating fuzzy logic [Zadeh73], which combines elements of multi-valued logic, probability theory, and artificial intelligence. Currently, fuzzy logic is emerging in industry with applications in a variety of domains. Fuzzy logic is amenable to situations when only incomplete information is available, and hence has rapidly become one of the most successful technologies of today. It provides a remarkably simple way to draw definite conclusions from vague, ambiguous or imprecise information. In a sense, fuzzy logic resembles human decision-making, with its ability to work from approximate data, and yet find precise solutions. Unlike classical logic, which requires thorough understanding of a system and use of precise values, fuzzy logic incorporates an alternative way of thinking, which allows modeling complex systems using a higher level of abstraction originating from our knowledge and experience, by expressing this knowledge with subjective concepts that are mapped into exact numeric ranges.

Fuzzy logic uses degrees of membership in sets rather than a strict (yes/no) membership. The degree of membership is the certainty (expressed as a number between 0 and 1) of a particular value belonging to a fuzzy set [Zadeh84]. Accordingly, complex computing tasks can be made simpler if the questions have imprecise or fuzzy answers rather than precise or crisp ones. A common language with terms known as linguistic variables, such as "hot", "warm" and "cold" is used to describe fuzzy sets, which are used to model linguistic uncertainty. Several different functions can be used, to represent fuzzy sets, including triangular, trapezoidal, quadratic and Gaussian. An important field of fuzzy system application is fuzzy clustering [Zhang93]. In fuzzy clustering, the objective is to determine the fuzzy classification of each pattern so as to minimize some suitably defined function.

A simple fuzzy logic system has a number of IF-THEN rules that produce one or more responses depending on the fired rules. Each rule is weighted according to the degree of membership function of the input. All the outputs of the rules are aggregated to produce an output value. In order to illustrate some basic concepts in fuzzy logic, consider a simple example of controlling a heater fan. The room temperature detected through a sensor is input to a controller, which outputs a control signal to adjust the heater fan speed. A fuzzy controller works in shades of grey where the temperature is treated as a series of overlapping ranges. For example, a temperature of 78°F can be considered to be 60% "warm" and 20% "hot". The controller is programmed with simple IF-THEN rules that adjust the control signal to the fan. As a result, when the temperature changes the fan speed will continuously adjust to keep the temperature at the desired level.

To design such a fuzzy controller it is necessary to characterize the range of values for the input and output variables of the controller. Then we assign linguistic labels such as "cool" for the temperature, and "high" for the fan speed. A set of English-like rules is then derived to control the system. Inside the controller, all temperature regulating actions will be based on how the current room temperature falls into these ranges and the rules describing the system behaviour. The temperature controller described above can be defined by four simple rules:

IF temperature IS cold THEN fan_speed IS high

IF temperature IS cool THEN fan_speed IS medium

IF temperature IS warm THEN fan_speed IS low

IF temperature IS hot THEN fan_speed IS zero

Here the linguistic variables cool, warm, high, etc. are labels, which refer to the set of overlapping values shown in Figure 5.10. These triangular shaped values are called membership functions. A fuzzy controller works in a similar manner to a conventional system: it accepts an input value, performs some calculations, and generates an output value. This process is called the Fuzzy Inference process, and works in three steps illustrated in Figure 5.11: (a) Fuzzification where a crisp input is translated into a fuzzy value; (b) Rule Evaluation, where the fuzzy output truth-values are computed; and (c) Defuzzification where the fuzzy output is translated to a crisp value. During the fuzzification step, the crisp temperature value is translated into fuzzy truth-values. For example, 78°F is fuzzified into "warm" with truth-value 0.6 (or 60%) and "hot" with truth-value 0.2 (or 20%).

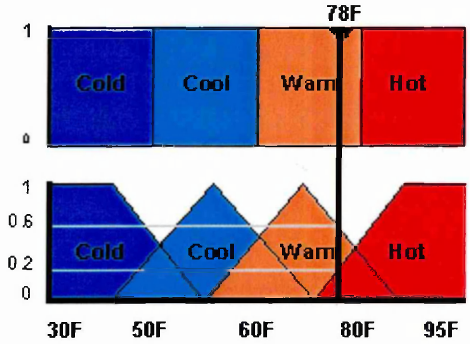
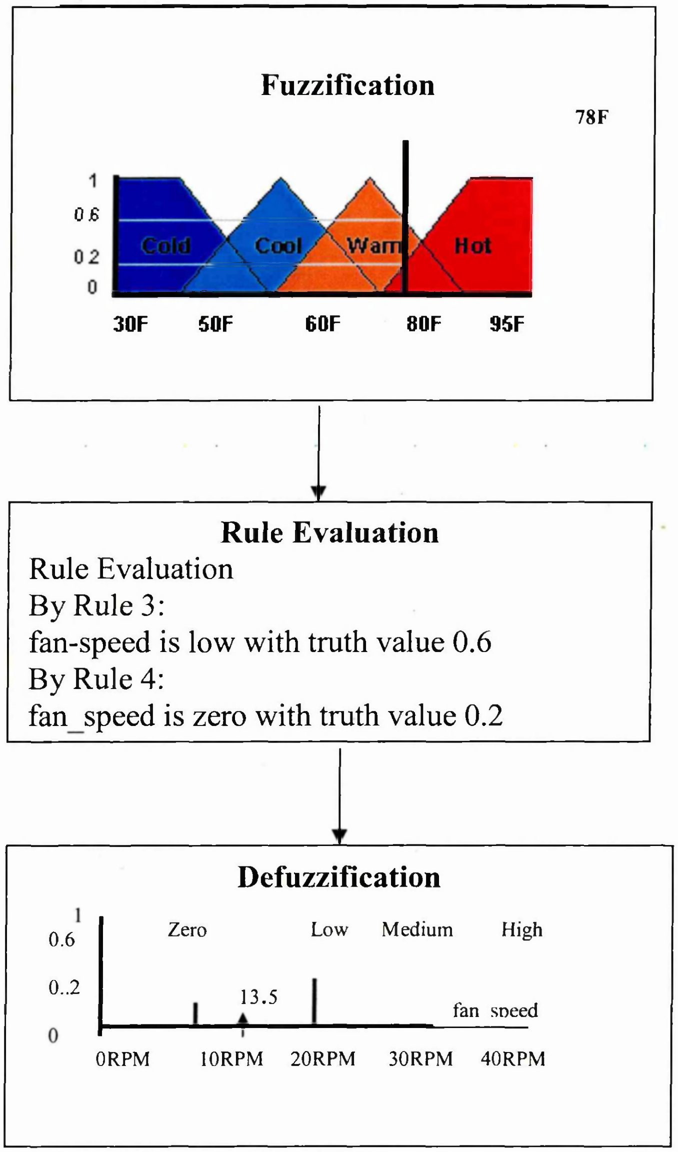


Figure: 5.10. Conventional and fuzzy sets



Rule Evaluation

Rule Evaluation
By Rule 3:
fan-speed is low with truth value 0.6
By Rule 4:
fan_speed is zero with truth value 0.2

Figure: 5.11 Fuzzy Inference Process

During the rule evaluation step the entire set of rules is evaluated, for which some rules may "fire". For a temperature of 78°F, only the last two of the four rules will fire. Specifically, using rule three the fan_speed will be "low" with degree of truth 0.6, and using rule four the fan_speed will be "zero" with degree of truth 0.2. During the defuzzification step, the 60% "low" and 20% "zero" labels are combined using a calculation method called the Centre of Gravity (COG) [Lee90], in order to produce the crisp output value of 13.5 RPM for the fan speed

Fuzzy systems can be used for the same tasks as neural networks. The difference is that a learning algorithm does not create fuzzy systems. They are built from explicit knowledge, which is expressed in form of linguistic (fuzzy) rules. However, it is sometimes difficult to specify all parameters of a fuzzy system (rules and membership functions). If the performance of the fuzzy system is not satisfactory, the parameters must be tuned manually. This tuning process is error prone and time consuming. So the idea of applying some kind of learning algorithm to a fuzzy system, in particular, the combination with neural networks is very popular.

5.3.3 Fuzzy-Neural Systems

Fuzzy-neural hybrid systems combine the advantages of fuzzy systems, which deal with explicit knowledge that can be explained and understood, and neural networks, which deal with implicit knowledge that can be acquired by learning. Neural network learning provides a good way to adjust the expert's knowledge and automatically generate additional fuzzy rules and membership functions, to meet certain specifications and reduce design time and costs. On the other hand, fuzzy logic enhances the generalisation capability of a neural network system by providing more reliable output when extrapolation is needed beyond the limits of the training data.

Neural networks and fuzzy systems, although very different, have a close relationship: they both can work with imprecision in a space that is not defined by crisp, deterministic boundaries. Neural networks and fuzzy systems each have their own shortcomings. When one designs with neural networks alone, the network is a black box that needs to be defined and thus, this is a computationally intensive process. Fuzzy systems, on the other hand, require a thorough understanding of the fuzzy variables and fuzzy membership functions, of the input-output relationships as well as the good judgement to select the fuzzy rules that contribute the most to the solution of the application [Zadeh73]. Neural networks are powerful in machine learning, associative memory and parallel processing, but they fail to do well in some symbol processing and indefinite reasoning. On the contrary, fuzzy logic systems are powerful in indefinite reasoning and symbol processing but they fail to do well in associative memory [Sheng01].

On the one hand, neural systems are useful if sufficient process data is available or is measurable, while on the other, fuzzy systems are appropriate if sufficient expert knowledge about the process is available. Despite this, there exists a lot of similarity and a synergetic relationship between neural networks and fuzzy logic systems. Formal equivalencies between different types of fuzzy and neural systems have also established, and it has been shown that neural networks can be constructed that are identical to fuzzy systems [Linkens96]. The shortcomings of neural networks and fuzzy systems may be overcome if we incorporate fuzzy logic operations into neural networks and learning and classification of neural networks into fuzzy systems. The result is called a Fuzzy Neural Network (FNN). In FNN, the neural network part is primarily used for its learning and classification and retrieval. The neural network part automatically generates fuzzy logic rules and membership functions during the training period. In addition, even after training, the neural networks keeps updating the membership functions and fuzzy logic rules as it learns more and more from its input signals. Fuzzy logic, on the other hand, is used to infer and provide a crisp or defuzzified output when fuzzy parameters exist.

Many related studies on fuzzy-neural applications aim at combining the advantages of the two paradigms. Some work has been carried out on fuzzy neural systems for pattern recognition. Enhancing neural systems by fuzzy logic and evolutionary reinforcement has also been addressed [Nyongesa98]. [Kosko90] proposed a Fuzzy Associative Memory (FAM), which defined mapping between fuzzy sets and neurons. A description of a simple fuzzy neurone model is discussed and used in a neural network for application in character recognition problems [O'Hagan91]. A fuzzy competitive learning algorithm has been described and used in a hybrid fuzzy neural systems for hand-written word recognition [Bum95]. This research has also reported fingerprint recognition (identification and classification) using fuzzy-neural techniques [Mohamed01], [Mohamed02]. Neural networks and fuzzy techniques are among the most promising approaches to pattern recognition.

In general, there are two major information sources for the construction of fuzzy-neural systems, namely, human beings who provide linguistic instructions and descriptions of the system, and sensors that provide numerical measurements of variables [Bastian95]. We refer to the information from the former as linguistic information and the information derived from the latter as numerical information. Therefore, from an information-source point of view we can divide fuzzy-neural systems into either of the following classes:

1. Linguistic fuzzy-neural networks: these are nets that are constructed using linguistic information and the adjustments of such networks are achieved by using neural network technique [Yager94]. The initial rule-base can be created either by using the expert knowledge. The incorporation of linguistic information in these systems prevents the random choice of the initial structure and parameters. Consequently, the fuzzy-neural network converges faster during training and performs better in decision-making. However, such linguistic information is not always available.

2. Numeric fuzzy-neural networks: these are the networks that are constructed using numerical information and the adjustments of the networks are also achieved by using

neural network techniques. This class of fuzzy-neural networks [Lee00], [Yager94] is similar to the previous one insofar as neural network techniques are concerned for the tuning and configuration of the parameters and structures. However, the initial set of parameters and the structure of such fuzzy-neural networks are derived using an unsupervised learning algorithm from a set of training data [Junhong94] and are fine-tuned on the basis of the numerical information. [Yager94] gives an example of this class of fuzzy-neural network, in which a self-organising procedure is used to determine the structure and initial weights of the fuzzy-neural networks. This class is suitable in applications where one may have direct observations from the system, but are unable to find experts who can provide an organised description of the system. However, since the set of training data is the only source of the information employed in such fuzzy-neural networks, it must be representative of the system's behaviour. Moreover, the unsupervised learning algorithms that are used to initialise the fuzzy-neural network must be properly selected in the absence of expert opinions.

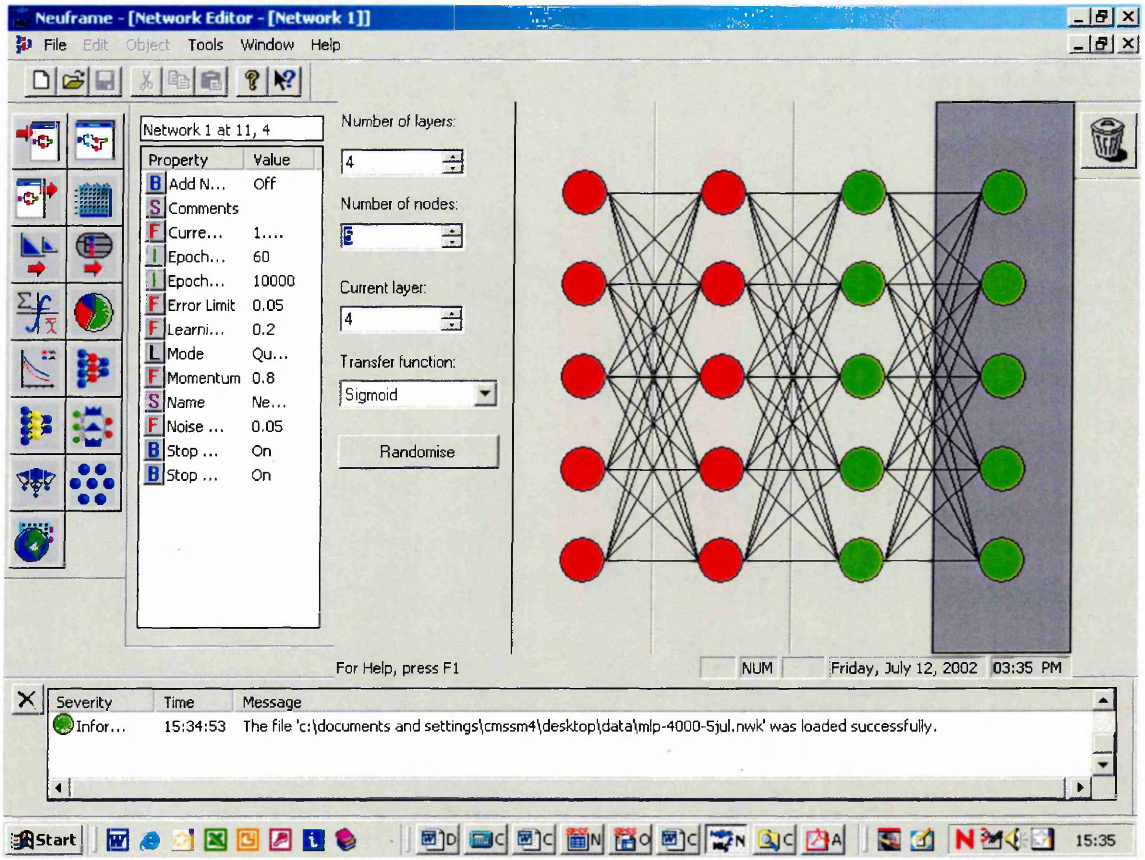
In this research, we investigated fuzzy-neural networks belonging to the first class. The proposed approach combines the features of neural networks (such as learning ability and high-speed parallel structure) and fuzzy systems (such as ability to process fuzzy information using fuzzy membership and rule-based systems). In the fuzzy-neural system, each neurone in the fuzzification layer represents an input membership function of the antecedent of a fuzzy rule. One common method to implement this layer is to express membership functions as discrete points [Cox95]. A better approach is to use a combination of one or two sigmoidal functions and a linear function to represent each membership function in the fuzzification and defuzzification layers [Lin99]. The parameters of these neurones can be trained to fine-tune the final shape and location of the membership functions.

5.4 Implementation of MLP and RBF Network Classifiers

Whichever network is used, the overall processes are very similar. The input part of the pattern is applied to the network and the network's response is stored. Each response value is then subtracted from its matching expected value to form an error value for each output. Weights adjustment are calculated for each weight. Either the weights are modified at this point or the weight adjustments are accumulated. If all of the patterns have been presented to the network, the errors for all the patterns are summed to give a performance value. This process repeats and the error, hopefully, improves. Neural network interrogation (query) is the process of passing data through a network to obtain an assessment of the correct answer. If the network is untrained, the outputs will be meaningless. If the network is trained to a given accuracy and the input pattern is one on which the net has been trained, the net will produce the output expected to the accuracy specified. If, however, the pattern is one that the network has not processed before, it will estimate the answer to the best of its ability. In order to train and test both MLP and RBF classifiers we used the Neuframe software package.

The experimental parameters for the two classifiers were setup as follows. The architecture of the MLP network used was 5x5x5x5, learning rate was 0.2 and the momentum rate was 0.8. The specifications were chosen after testing different alternatives, including network size, topology and maximum epochs. A typical network set-up is shown in Figure 5.12, while a typical training error curve of MLP as shown in Figure 5.13. Similarly, the architecture of the RBF network is shown in Figure 5.14. The network size was 5x6x5, with Gaussian transfer functions whose parameters were automatically estimated from the data. Noise level of zero mean and

0.05 amplitude was added to the input patterns. Again, these specifications were chosen after testing different alternatives. The training process was terminated when the performance value exceeded a threshold, or when the number of epochs processed is greater than a defined value. Experimental results and analysis are presented in the next Chapter.



Network 1 at 11, 4

Property	Value
B Add N...	Off
S Comments	
F Curre...	1....
I Epoch...	60
I Epoch...	10000
F Error Limit	0.05
F Learni...	0.2
L Mode	Qu...
F Momentum	0.8
S Name	Ne...
F Noise ...	0.05
B Stop ...	On
B Stop ...	On

Number of layers: 4

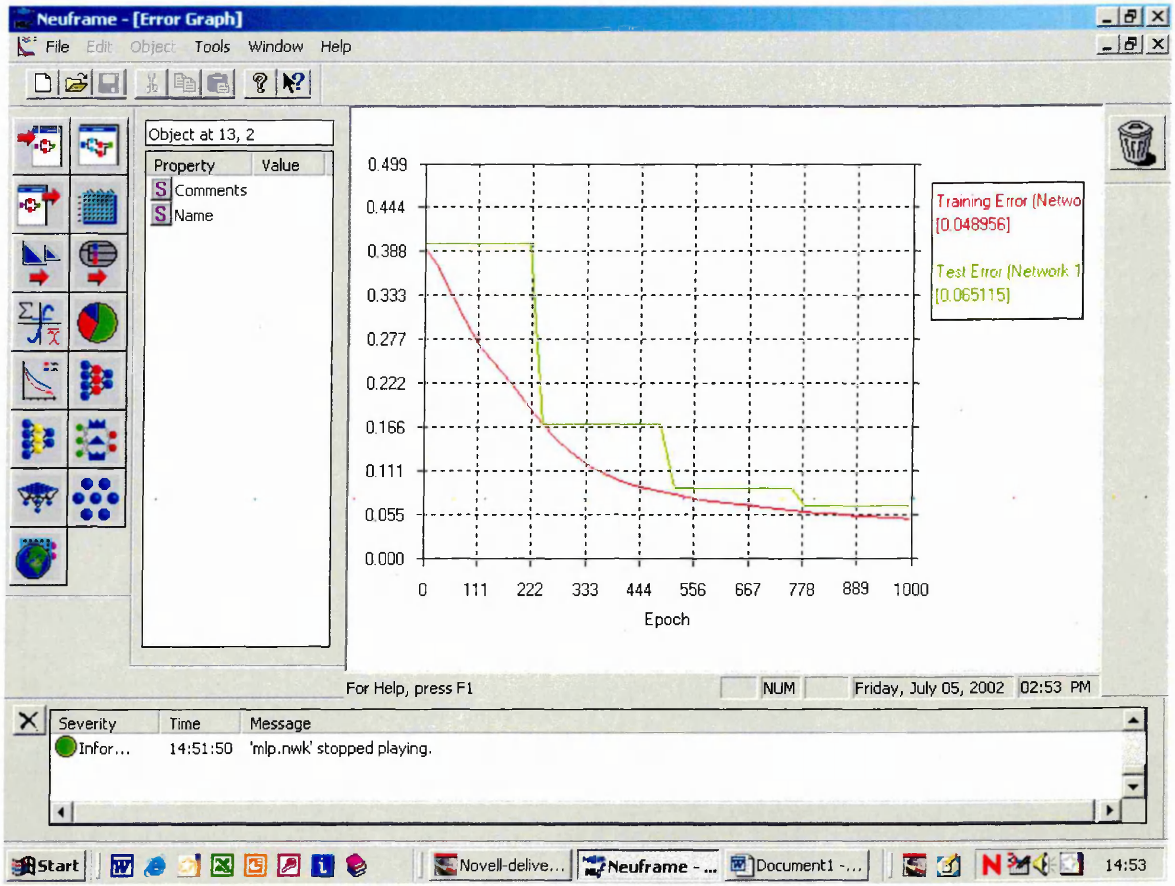
Number of nodes: 5

Current layer: 4

Transfer function: Sigmoid

Severity	Time	Message
Infor...	15:34:53	The file 'c:\documents and settings\cmssm4\desktop\data\mlp-4000-5jul.nwk' was loaded successfully.

Figure 5.12 Architecture of MLP network



Object at 13, 2

Property	Value
S Comments	
S Name	

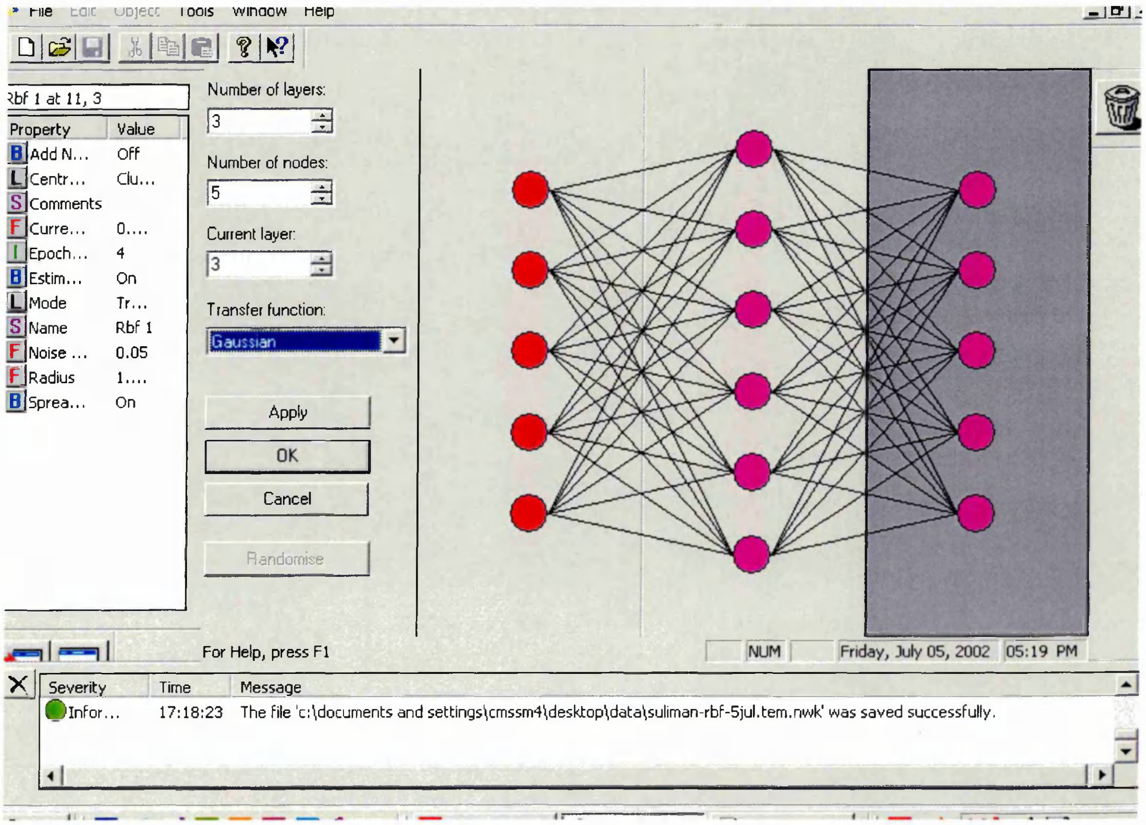
Training Error (Netwo [0.048956]

Test Error (Network 1 [0.065115]

For Help, press F1 NUM Friday, July 05, 2002 02:53 PM

Severity	Time	Message
Infor...	14:51:50	'mlp.nwk' stopped playing.

Figure 5.13 Training error curve of MLP network



Rbf 1 at 11, 3

Property	Value
B Add N...	Off
L Centr...	Clu...
S Comments	
F Curre...	0....
I Epoch...	4
B Estim...	On
L Mode	Tr...
S Name	Rbf 1
F Noise ...	0.05
F Radius	1....
B Sprea...	On

Number of layers: 3

Number of nodes: 5

Current layer: 3

Transfer function: Gaussian

For Help, press F1 NUM Friday, July 05, 2002 05:19 PM

Severity	Time	Message
Infor...	17:18:23	The file 'c:\documents and settings\cmssm4\desktop\data\suliman-rbf-5jul.tem.nwk' was saved successfully.

Figure 5.14 Architecture of RBF network.

Results of the performances of the neural classifiers are discussed in details in the next Chapter. In summary, MLP achieved a minimum classification accuracy of 91.97% a maximum classification accuracy of 98.72%, with mean accuracy of 96.09% and standard deviation of 1.01. RBF achieved a minimum classification accuracy of 83.07%, a maximum classification accuracy of 87.02%, with mean accuracy of 84.54% and standard deviation of 1.04. The poorer performance of the RBF classifier is not surprising. Application of RBF networks requires optimal placement of the centres and standard deviations of the Gaussian basis functions. In the present application, however, the feature vectors are comprised of discrete, singular points in an n-dimensional space. The presence of "noisy" feature vectors, that is vectors containing wrong feature values, as a result of the feature extraction process or noisy fingerprint images leads to attempts to place the centres of the basis functions at sub-optimal mid-way points. We believe this contributes to the poor performance of RBF classifiers. The MLP are shown to offer a superior performance but its operation is not transparent, unlike RBFs, which can be interpretable. A desire for transparency of a classifier, in order to validate results against expert knowledge, and the lack of such transparency by the MLP network is the motivation for investigation of a hybrid fuzzy-neural classifier. Fuzzy logic is a key technology for representing human knowledge and for constructing adaptive systems. Fuzzy logic is chosen when there is a definite a *priori* model, from which a rule set can be constructed.

5.5 Implementation of Fuzzy-Neural Network Classifier

The fuzzy-neural approach was chosen to achieve a good accuracy of feature classification, and also transparency of the classification process. We adopted Neuframe-4 software to train and test the data of fingerprint features. Neuframe-4 [Neuframe01] is an application development environment for combining NNs and fuzzy systems. The fuzzy-neural network, in general has a five-layer structure. The layers are characterised by the fuzzy operations that are performed. The processes among these layers as follows:

Layer 1 (the input layer): neurones in the input layer receive crisp data of the input features.

Layer 2 (the fuzzification layer): the neurones in this layer are the linguistic nodes, representing fuzzy concepts such as "small, medium, and large", which are fuzzification of inputs. The output of these membership function neurones are connected to the fuzzy rules layer, using links with fixed weights. The weights of these links represent the relative significance of the rules associated with the neurones. Their values can be pre-set according to the expert or initialised to be 1.0, and then trained to reflect their actual importance to the output membership functions contained in the defuzzification layer.

Layer 3 (the rule-base layer): the neurones in this layer are the rule nodes. They represent fuzzy rules, such as, "if CoreNo is "medium" AND DeltaNo is "medium"....THEN ...". The rules represent linguistic labels, such as, membership values of the corresponding output variables, and produce consequences. In most designs, the number of neurones in the rule-base layer is fixed, but it is possible to add or remove these neurones during training, according to the outputs produced on the training samples [Zhou96]. As result, when all the training data are derived from certain rules, the links denoting these rules have larger weights than others.

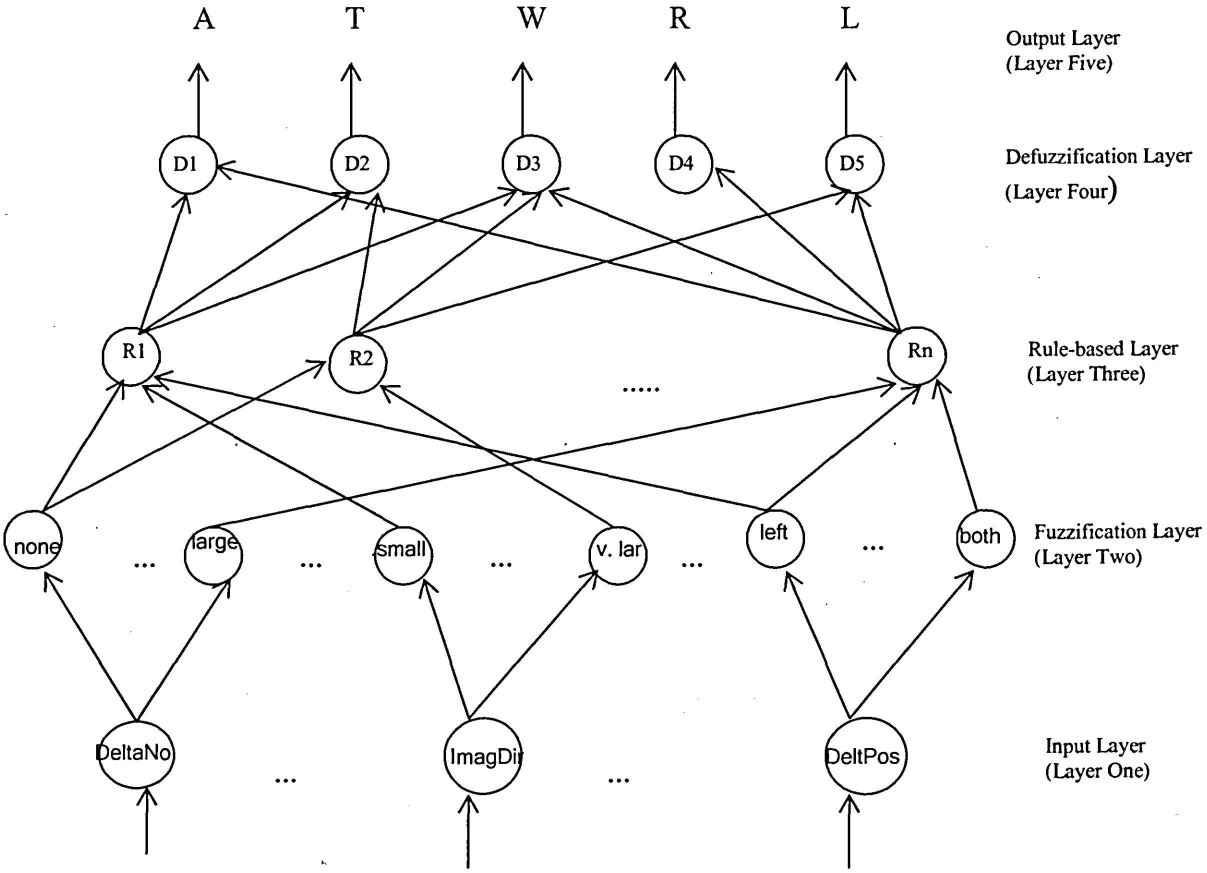


Figure: 5.15 Structure of the Fuzzy-Neural Network.

Layers 4 (the defuzzification layer): Each neurone in the defuzzification layer represents a consequent proposition "Y is B", and its membership function can be implemented by combining one or two sigmoidal functions and linear functions. The certainty of each consequent proposition is calculated, and is regarded as the goodness of fit of those fuzzy rules, which have the same consequent proposition. The neurones process linguistic information back to crisp values. The weight of each link from these neurones represents the output membership function of the consequent, and is trainable. Layer 5 (the output layer): the neurones in this layer provide the weighted average of the consequences of all rules, as the final output.

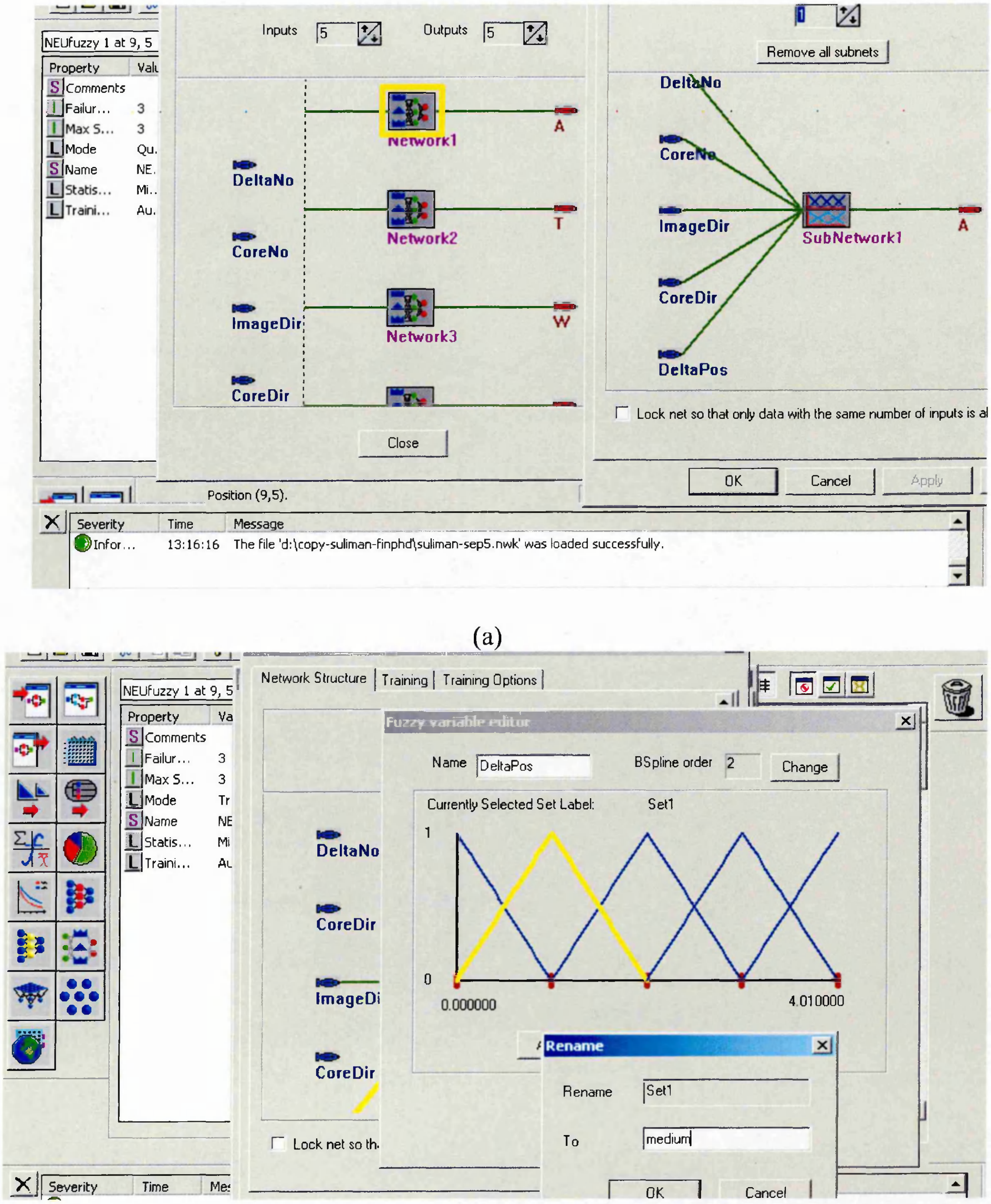
The classifier network was constructed through an automatic network construction process, a feature of NeuFrame^{TM}. The fuzzy variable editor, also a feature of the software, allows adjustment of membership functions. Once we have named the relevant inputs, the membership sets and the outputs, we can view the rules generated by each sub-network. A typical example of the rules from the sub-networks for each class is illustrated below. The complete listing of rules is presented in Chapter 6.

IF DeltaNo is "none" AND CoreNo is "none" AND
ImageDir is "small" AND CoreDir is "small" AND DeltaPos
is "none" THEN A is Equal (1.00) OR A is Equal (0.00)

The FNN system has three modes of operation:

1. Automatic Network Construction (ANC): this is used when you have no knowledge about the rules governing the problem but you have data describing it. ANC will extract rules from the data, where the data will contain such information. When this is complete, rules can be modified at any time based on subsequent "expert" knowledge and experience. Unlike MLP and RBF, Neufuzzy will only use variables which have a bearing on the result.
2. Weight Rule Confidence Training: this is for use when you have some "expert" knowledge of the rules governing the problem, and you then tune the rules subsequently using available data.
3. Fuzzy Logic: is used when there is no data available and all the knowledge for the rules is supplied by an "expert", who is also used subsequently to modify the rules in the light of further experience.

The architecture of the FNN network used was 5x20xNx5x5. The first hidden layer is the fuzzification of the input features into 20 heuristic terms chosen for the crisp values. The number of nodes in second hidden layer is initially undefined because the rules numbers is not limited. The FNN tool is shown in Figure 5.16, which illustrates the internal sub-networks for each fingerprint class, and the fuzzy variable editor. Once we have named the relevant inputs, outputs, the memberships sets and the outputs sets, we can view the (IF... AND....THEN.....) rules generated by each trained sub-network.



The file 'd:\copy-suliman-finphd\suliman-sep5.nwk' was loaded successfully.

(a)

Name: DeltaPos
BSpline order: 2
Currently Selected Set Label: Set1
Rename: Set1
To: medium

(b)

Figure 5.16. Neuframe Toolbox: (a) Classification subnetworks (b) fuzzy editor

The weights of the links between the rule-base layer and the output layer are initially set to zero. After the ANC learning process, these weights represent the strengths of the fuzzy rules having the corresponding output-label nodes as their real consequences. Amongst the links between a rule node and all the output-label of a defuzzification node, at most one link with the highest weight is chosen and the others are deleted. On the other hand, when all the weights of the links between a rule node and the output-label node of a defuzzification node are very small or almost equal to each other, this means that this rule node has little or no relation to the output variable represented by the particular defuzzification. Hence, all the corresponding links in this case can be deleted without affecting the outputs. The details of processing steps and implementation of ANC including loading, training and testing data, deffuzification, training the network, and query the results, are presented in the Appendix: 3.

5.6 Conclusion

This Chapter has described the implementation of fingerprint features classification for the five-class problem. Extracted features were analysed using SPSS software, and shown to exhibit significant overlap between class features. In order to efficiently match fingerprints in a given database, a good classification scheme plays a vital and necessary role. We have studied a flexible fingerprint classification system, which classifies input fingerprints according to the number of cores and deltas (singular points), and their relative positions. In this regard, we implemented RBF, MLP and FNN classifiers. Performance results, analysis and discussion are presented in the next Chapter.

Chapter 6

Experimental Results and Analysis

6.1 Introduction

This Chapter presents a discussion and an analysis of the results obtained from implementation of the fingerprint classification system. Although, many researchers have studied the problem of fingerprint classification using different techniques, the error rate is still relatively high. Hence improvement of the accuracy of fingerprint classification is still a crucial problem in automatic fingerprint identification systems. The feature vectors obtained utilising the proposed FFE algorithms were classified using RBF, MLP and FNN classifiers.

The rest of this Chapter is organised as follows: Section 6.2 presents a brief discussion on experimental procedure. Section 6.3 presents the experimental results. Section 6.4 presents discussion and analysis of results obtained. Section 6.5 discusses comparison of reviewed and proposed classification accuracies. Finally, section 6.6 draws some conclusions and remarks on this Chapter.

6.2 Experimental Procedure

The experimental procedure presented in this Chapter depends on the FFE algorithm developed in Chapter 4 and the implementation of RBF, MLP and FNN classifiers, described in Chapter 5. We have tested our fingerprint classification algorithm on the well-known NIST-4 database of fingerprint images. The NIST-4 database consists of 4,000 fingerprint images, (with image size of 512x512) from 2,000 fingers [Watson92]. Each finger has two impressions named f (first) and s (second). Each image is labelled with one of the five classes (A, T, W, R, and L). We have made use of all the labels of a fingerprint to train our system. During testing, we consider the output of the classifier to be correct if the output matches the target labels. The images in the NIST-4 database are numbered f0001 through f2000 and s0001 through s2000. Each number represents a fingerprint from a different finger. We form our training set with the first 2,000 fingerprints from 1,000 fingers (f0001 to f1000 and s0001 to s1000) and the test set with the remaining 2,000 fingerprints (f1001 to f2000 and s1001 to s2000).

The natural proportions of fingerprints belonging to each class is 0.279, 0.317, 0.338, 0.037, and 0.029 for the classes W, R, L, A, and T, respectively [Wilson94]. Classification accuracy can be significantly increased by using data-sets whose records follow the natural distribution of fingerprint classes because the more common types of fingerprints (loop and whorl) are easier to recognise. However, the distributions of the data-sets in this study does not follow this natural class distribution: we used an equal number of fingerprints (800) belonging to each of the five classes. Appendix 1 describes the fingerprints file format in the NIST-4 database.

The feature extraction process has been discussed in Chapter 4. Specifically, five features (DeltaNo, CoreNo, CoreDir, DeltaPos, ImageDir) were extracted from each fingerprint. The three types of classifiers have been discussed in Chapter 5. Preliminary experiments were carried out to determine the best network configuration for each classifier type. Experiments were carried out 30 times for each classifier using the Neuframe^{TM} environment, which implements all 3 types of classifiers, using

30 different samples of data created from the feature vectors. The results presented are averaged for the 30 experiments, where appropriate.

6.3 Experimental Results

The experimental results of the RBF classifier are presented in Table 6.1.

Dataset	Network Size	Number of Training Epochs	SSE Training	SSE Testing	Classification Accuracy
1	5x6x5	415	0.156	0.159	85.41%
2	5x7x5	405	0.152	0.154	85.70%
3	5x8x5	537	0.157	0.157	85.17%
4	5x9x5	575	0.153	0.158	85.50%
5	5x10x5	609	0.156	0.158	85.42%
6	5x11x5	645	0.955	0.956	83.64%
7	5x12x5	744	0.950	0.952	83.57%
8	5x9x5	333	1.618	1.624	83.11%
9	5x6x5	334	1.625	1.642	83.07%
10	5x6x5	354	1.590	1.607	83.13%
11	""	352	1.501	1.606	83.31%
12	""	355	1.520	1.616	83.23%
13	""	348	0.195	0.195	84.37%
14	""	361	0.196	0.196	84.33%
15	""	367	0.191	0.194	84.60%
16	""	452	0.158	0.159	85.09%
17	""	451	0.157	0.158	85.16%
18	""	463	0.197	0.197	84.32%
19	""	470	0.192	0.193	84.51%
20	""	467	0.194	0.195	84.37%
21	""	701	0.173	0.174	85.00%
22	""	689	0.223	0.243	84.13%
23	""	499	0.236	0.236	84.03%
24	""	479	0.228	0.231	84.11%
25	""	501	0.239	0.240	84.00%
26	""	512	0.152	0.153	85.80%
27	""	524	0.150	0.151	85.90%
28	""	533	0.141	0.145	87.02%
29	""	537	0.149	0.146	86.60%
30	""	589	0.153	0.147	86.48%

Table: 6.1 RBF Classifier results using Gaussain transfer functions.

Statistical analysis of the RBF testing results are shown in Table 6.2. In particular, we obtained a minimum accuracy of 83.07%, a maximum of 87.02 and an average accuracy of 84.66. Insight into the performance of the RBF classifier can be obtained by examining the confusion matrix in Table 6.3, for the maximum accuracy of the training results. This matrix has a row for each actual class and a column for each hypothesized (output) class. It shows un-parenthesized numbers, the number of fingerprints that fell into each class, and a parenthesized numbers corresponding to the percentage of each class. The entries shown in bold-face correspond to correct classifications. Figure 6.1 shows the same results as a bar-chart.

Expreiments	TRAINSSE	TESTSSE	ACCURACY	EPCHOS
Numbers	30	30	30	30
Mean	.47	.46	84.66	486.70
Std. Deviation	.55	.54	1.08	115.35
Minimum	.145	.141	83.07	333.00
Maximum	1.642	1.625	87.02	744.00

Table 6.2 Statistical analysis of the RBF results.

	Hypothesized Class				
Actual Class	A	T	W	R	L
A	**336 (84.0)**	42 (10.5)	10 (2.5)	8 (2.0)	4 (1.0)
T	41 (10.25)	**339 (84.75)**	8 (2.0)	6 (1.5)	6 (1.5)
W	8 (2.0)	13 (3.25)	**359 (89.75)**	11 (2.75)	9 (2.25)
R	8 (2.0)	11 (2.75)	17 (4.25)	**357 (89.25)**	7 (1.75)
L	6 (1.5)	11 (2.75)	23 (5.75)	9 (2.25)	**351 (87.75)**

Table 6.3 Confusion matrix of five-class classification results on RBF classifier.

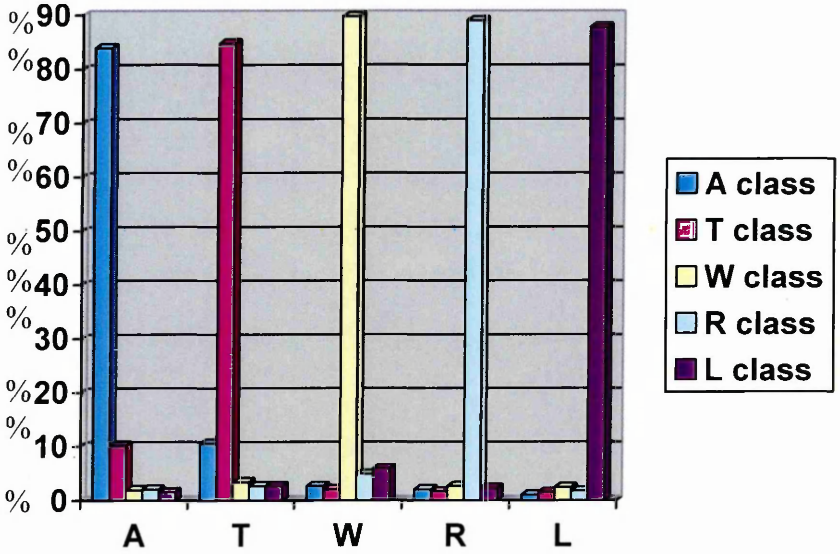


Figure 6.1 A Bar-chart of the RBF five-class results.

The experimental results of MLP classifier are presented in Table 6.4. Statistical analysis of the MLP testing results are shown in Table 6.5. We obtained a minimum accuracy of 91.97%, a maximum of 98.83 and an average accuracy of 96.15. Insight into the performance of the MLP classifier can be obtained by examining the confusion matrix in Table 6.6, for the case of the minimum accuracy of 91.97%. This matrix has the same properties as Table 6.3 above. Figure 6.2 shows a bar-chart of the MLP performance from data in the confusion matrix of Table 6.6.

Dataset	Network Size	Number of Epochs	SSE Training	SSE Testing	Classification Accuracy
1	5x5x5x5	189	0.049	0.049	94.23%
2	""	223	0.049	0.049	94.10%
3	""	285	0.047	0.049	94.61%
4	""	244	0.048	0.049	94.27%
5	""	285	0.047	0.049	94.60%
6	5x7x5	272	0.048	0.049	94.30%
7	5x5x5	275	0.048	0.049	94.27%
8	5x4x5	334	0.049	0.049	93.97%
9	5x5x5x5	283	0.038	0.039	96.39%
10	""	289	0.039	0.039	96.04%
11	""	291	0.037	0.038	97.38%
12	""	298	0.036	0.037	98.01%
13	""	318	0.036	0.037	98.09%
14	""	365	0.036	0.037	98.07%
15	""	370	0.034	0.036	98.41%
16	""	324	0.027	0.028	98.62%
17	""	401	0.028	0.029	98.46%
18	""	411	0.026	0.027	98.83%
19	""	398	0.027	0.028	98.51%
20	""	410	0.036	0.036	97.47%
21	""	401	0.037	0.037	97.30%
22	""	423	0.038	0.038	97.02%
23	""	442	0.038	0.039	96.39%
24	""	409	0.038	0.039	96.26%
25	""	421	0.042	0.042	94.77%
26	""	415	0.041	0.041	95.20%
27	""	407	0.040	0.041	96.03%
28	""	427	0.041	0.041	95.07%
29	""	432	0.040	0.041	95.90%
30	""	437	0.049	0.050	91.97%

Table: 6.4 Simulation results for MLP using different sizes of hidden neurons (Learning rate parameter =0.2 and momentum =0.8), stop training on training error tested between 0.01 and 0.05, and epochs=1000.

Experiments	TRAINSSE	TESTSSE	ACCURACY	EPCHOS
Numbers	30	30	30	30
Mean	.0404	.0396	96.15	349.30
Std. Deviation	.006	.007	1.81	73.16
Minimum	.027	.026	91.97	189.00
Maximum	.050	.049	98.83	442.00

Table 6.5 Statistical analysis of the MLP results.

	Hypothesized Class				
Actual Class	A	T	W	R	L
A	**354 (88.5)**	26 (6.5)	7 (1.75)	5 (1.25)	8 (2.0)
T	19 (4.75)	**367 (91.75)**	2 (0.5)	8 (2.0)	4 (1.0)
W	3 (0.75)	11 (2.75)	**372 (93)**	10 (2.50)	4 (1.0)
R	6 (1.5)	9 (2.25)	11 (2.75)	**371 (92.75)**	3 (0.75)
L	5 (1.25)	8 (2.0)	9 (2.25)	3 (0.75)	**375 (93.75)**

Table 6.6 Confusion matrix of five-class classification results on MLP classifier.

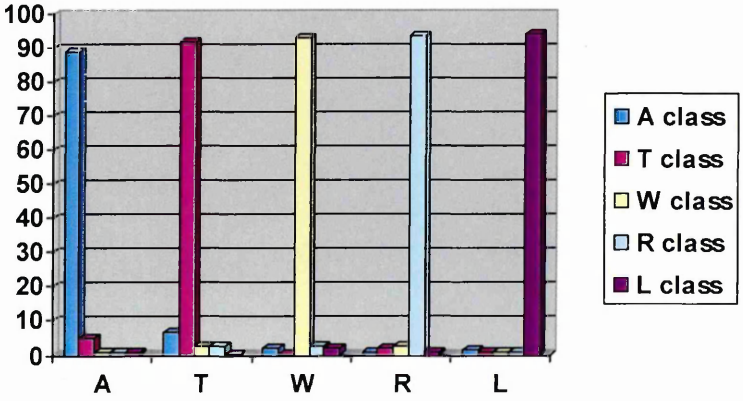


Figure 6.2 A Bar-chart of the MLP five-class results.

The experimental results of FNN classifier are presented in Table 6.7, which shows network size and classification accuracy. We tested FNN network, with different data sets and compared the results of the best network configuration.

Dataset	Network Size	Classification Accuracy
1	5x20x23x5x5	96.33%
2	5x20x23x5x5	96.71%
3	5x20x25x5x5	96.27%
4	5x20x27x5x5	97.63%
5	5x20x24x5x5	97.30%
6	5x20x23x5x5	95.80%
7	5x20x25x5x5	96.07%
8	5x20x25x5x5	97.72%
9	5x20x25x5x5	95.49%
10	5x20x25x5x5	95.84%
11	5x20x27x5x5	96.08%
12	5x20x25x5x5	95.01%
13	5x20x23x5x5	96.20%
14	5x20x25x5x5	97.07%
15	5x20x27x5x5	95.73%
16	5x20x24x5x5	96.09%
17	5x20x29x5x5	95.67%
18	5x20x23x5x5	96.14%
19	5x20x27x5x5	95.39%
20	5x20x23x5x5	96.37%
21	5x20x23x5x5	96.60%
22	5x20x24x5x5	97.22%
23	5x20x24x5x5	96.69%
24	5x20x28x5x5	97.26%
25	5x20x23x5x5	95.60%
26	5x20x27x5x5	95.20%
27	5x20x23x5x5	97.03%
28	5x20x25x5x5	95.26%
29	5x20x27x5x5	95.11%
30	5x20x27x5x5	95.02%

Table 6.7 Classification accuracy from FFN testing.

Statistical analysis of the FNN testing results are shown in Table 6.8. We obtained a minimum accuracy of 95.01%, a maximum of 97.72 and an average accuracy of 96.19. In-sight into the performance of the FNN classifier can be obtained by examining the confusion matrix in Table 6.9, for the case of the maximum accuracy of 97.72%. Again this matrix has the same properties as Table 6.3 above. Figure 6.3 shows a bar-chart of the FNN's five-class performance from data in the confusion matrix of Table 6.9.

Valid	Accuracy
Numbers	30
Mean	96.19
Std. Deviation	.790
Minimum	95.01
Maximum	97.72

Table 6.8 Statistical analysis of the FNN classification accuracy results

	Hypothesized Class				
Actual Class	A	T	W	R	L
A	**386 (96.5)**	8 (2.0)	3 (0.75)	1 (0.25)	2 (0.50)
T	9 (2.25)	**387 (96.75)**	1 (0.25)	0 (0.0)	3 (0.75)
W	1 (0.25)	1 (0.25)	**395 (98.75)**	2 (0.50)	1 (0.25)
R	2 (0.50)	2 (0.50)	3 (0.75)	**392 (98.0)**	1 (0.25)
L	1 (0.25)	2 (0.50)	3 (0.75)	0 (0.0)	**394 (98.5)**

Table 6.9 Confusion matrix of five-class classification results on FNN classifier.

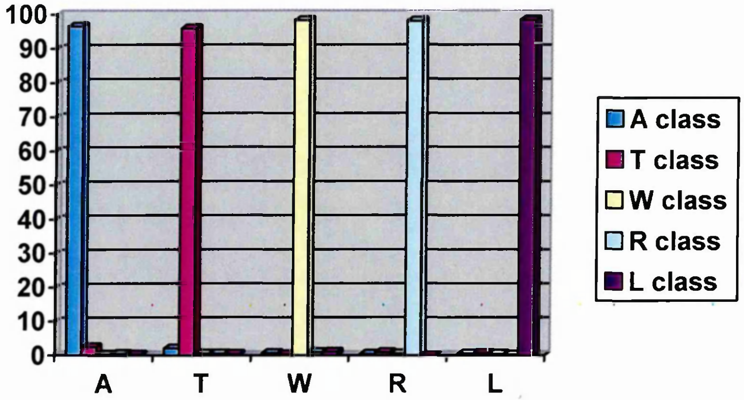


Figure 6.3 A Bar-chart of the FNN five-class results.

6.4 Analysis of Results

A summary of results is shown in Table 6.10. Comparison of the results obtained between the different network classifiers shows that the performance of MLP and FNN are satisfactory and very similar, but the RBF results does not perform as well. We argue that the poor RBF performance is because of the difficulty of placement of the Gaussian basis function for discrete data sets. The similarity of MLP and FNN performances, however, is not surprising. Afterall the FNN is, in fact, constructed using specially configured MLP-like layers to perform fuzzy functions. This research, in particular, sought to investigate the applicability of FNN classifiers in order to benefit from fuzzy decision making. These include:

- Ability to deal with uncertainty: Although the expected values for each feature are known, noisy images and feature extraction errors can results in variant feature values.

- Approximate reasoning: The essence of fuzzy decision making is to rely on aggregation of the rules for similar, but not exactly the same, instances.

	MLP	RBF	FNN
Mean	96.15	84.66	96.19
Std. Deviation	1.810	1.089	.790
Minimum	91.97	83.07	95.01
Maximum	98.83	87.02	97.72

Table 6.10 Summary results for accuracy of classifiers (RBF, MLP, FNN).

Table 6.11 and Table 6.12 show deterministic rules for the different classes of outputs. The FNN learns the actual rules for classification from data presented to it through the automatic network construction process, which can then be examined and validated by human experts, with respect to Table 6.11 and 6.12. Typical learned rules, for the specific data sets are shown in Tables 6.13 to 6.17. These rules were obtained for different data sets applied in succession; that is, learning was not initialised for each new data set. FNN allows examination of the activation fuzzy rules, in order to validate the efficiency of FNN learning. At the beginning of the FNN learning each rule is connected to all the output-label nodes of each defuzzification node. After the FNN learning process has been completed, some links have stronger strengths than the others. Ideally, the final rules should be similar to the desired set of fuzzy rules shown in Table 6.12. Each rule has a consequence, which is either a level of confidence or not defined. This is interpreted as a probability that the input vector in question belongs to the specific class. Therefore, if the two output fuzzy sets both have non-zero confidence levels, say at 0.65 and 0.35 (the total sums to 1), then it means that the input vector can be considered as 65% likely to be set 1 or 35% likely to be set 2. For a given input vector the selected output-class is the one that provides the maximum degree of confidence.

IF DeltaNo is 0 AND CoreNo is 0 AND ImageDir is 1 AND CoreDir is 1 AND DeltaPos is 1 THEN A
IF DeltaNo is 1 AND CoreNo is 1 AND ImageDir is 3 AND CoreDir is 3 AND DeltaPos is 1 THEN T
IF DeltaNo is 2 AND CoreNo is 2 AND ImageDir is 3 AND CoreDir is 2 AND DeltaPos 4 THEN W
IF DeltaNo is 1 AND CoreNo is 1 AND ImageDir is 4 AND CoreDir is 4 AND DeltaPos is 3 THEN R
IF DeltaNo is 1 AND CoreNo is 1 AND ImageDir is 2 AND CoreDir is 2 AND DeltaPos is 2 THEN L

Table 6.11 Desired crisp rules of the five classes

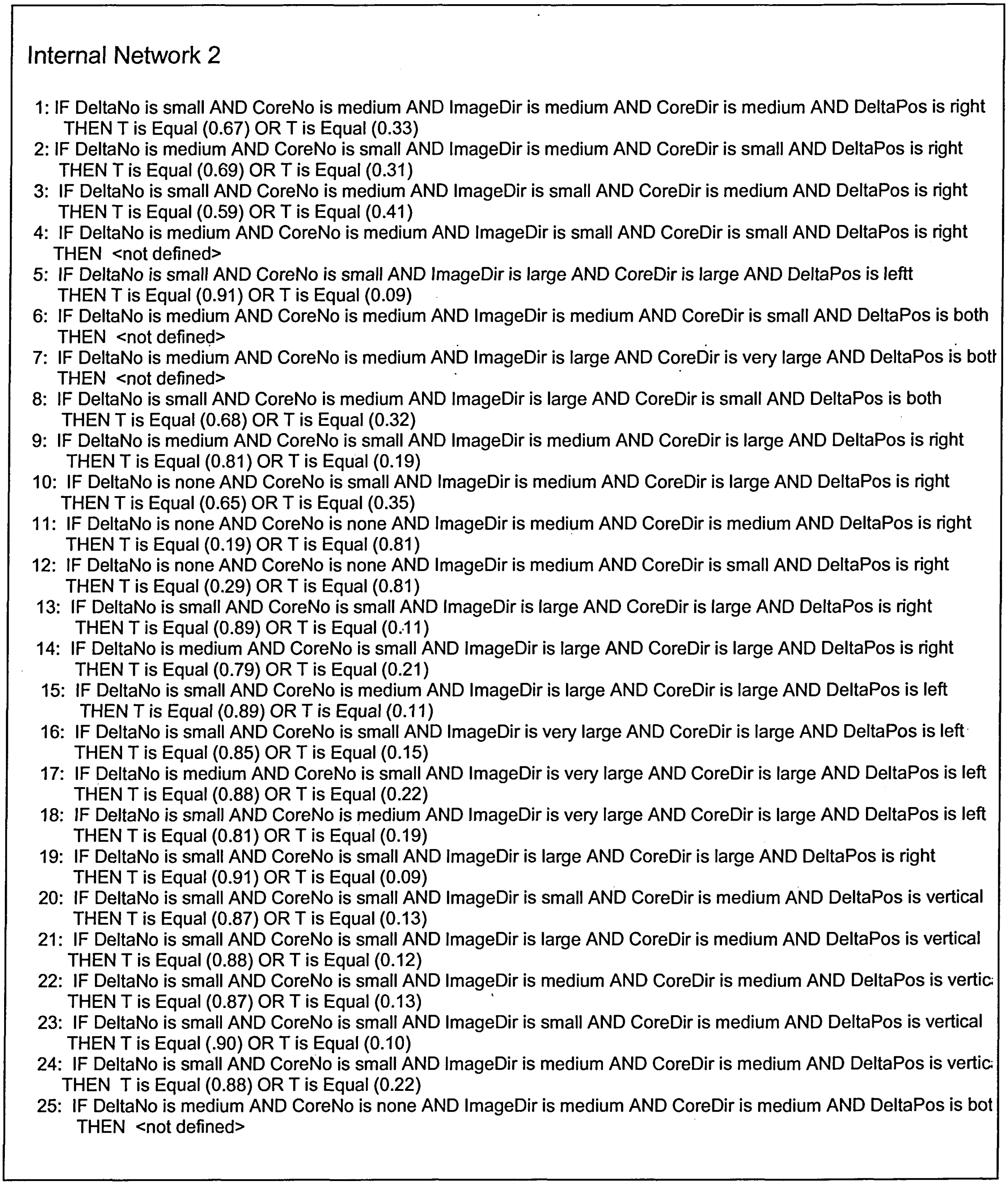
IF DeltaNo is none AND CoreNo is none AND ImageDir is small AND CoreDir is small AND DeltaPos is vertical THEN A
IF DeltaNo is small AND CoreNo is small AND ImageDir is large AND CoreDir is large AND DeltaPos is vertical THEN T
IF DeltaNo is medium AND CoreNo is medium AND ImageDir is small AND CoreDir is small AND DeltaPos is both THEN W
IF DeltaNo is small AND CoreNo is small AND ImageDir is v.-large AND CoreDir is v.-large AND DeltaPos is left THEN R
IF DeltaNo is small AND CoreNo is small AND ImageDir is medium AND CoreDir is medium AND DeltaPos is right THEN L

Table 6.12 Desired fuzzy rules of the five classes

Internal Network 1

1: IF DeltaNo is small AND CoreNo is small AND ImageDir is small AND CoreDir is small AND DeltaPos is right
THEN A is Equal (0.51) OR A is Equal (0.49)
2: IF DeltaNo is medium AND CoreNo is small AND ImageDir is small AND CoreDir is small AND DeltaPos is right
THEN A is Equal (0.45) OR A is Equal (0.55)
3: IF DeltaNo is small AND CoreNo is medium AND ImageDir is small AND CoreDir is small AND DeltaPos is right
THEN A is Equal (0.45) OR A is Equal (0.55)
4: IF DeltaNo is medium AND CoreNo is medium AND ImageDir is small AND CoreDir is small AND DeltaPos is right
THEN <not defined>
5: IF DeltaNo is small AND CoreNo is small AND ImageDir is large AND CoreDir is small AND DeltaPos is leftt
THEN A is Equal (0.28) OR A is Equal (0.72)
6: IF DeltaNo is none AND CoreNo is small AND ImageDir is large AND CoreDir is small AND DeltaPos is both
THEN A is Equal (0.55) OR A is Equal (0.45)
7: IF DeltaNo is none AND CoreNo is none AND ImageDir is large AND CoreDir is large AND DeltaPos is both
THEN A is Equal (0.65) OR A is Equal (0.35)
8: IF DeltaNo is none AND CoreNo is medium AND ImageDir is medium AND CoreDir is medium AND DeltaPos is right
THEN A is Equal (0.79) OR A is Equal (0.21)
9: IF DeltaNo is none AND CoreNo is none AND ImageDir is medium AND CoreDir is medium AND DeltaPos is right
THEN A is Equal (0.88) OR A is Equal (0.22)
10: IF DeltaNo is none AND CoreNo is none AND ImageDir is medium AND CoreDir is small AND DeltaPos is right
THEN A is Equal (0.89) OR A is Equal (0.11)
11: IF DeltaNo is small AND CoreNo is medium AND ImageDir is medium AND CoreDir is large AND DeltaPos is right
THEN A is Equal (0.30) OR A is Equal (0.70)
12: IF DeltaNo is small AND CoreNo is medium AND ImageDir is large AND CoreDir is large AND DeltaPos is right
THEN A is Equal (0.39) OR A is Equal (0.61)
13: IF DeltaNo is small AND CoreNo is small AND ImageDir is very large AND CoreDir is large AND DeltaPos is right
THEN A is Equal (0.31) OR A is Equal (0.69)
14: IF DeltaNo is medium AND CoreNo is small AND ImageDir is very large AND CoreDir is large AND DeltaPos is right
THEN A is Equal (0.21) OR A is Equal (0.79)
15: IF DeltaNo is medium AND CoreNo is small AND ImageDir is medium AND CoreDir is large AND DeltaPos is right
THEN A is Equal (0.21) OR A is Equal (0.79)
16: IF DeltaNo is small AND CoreNo is small AND ImageDir is small AND CoreDir is medium AND DeltaPos is vertical
THEN A is Equal (0.37) OR A is Equal (0.63)
17: IF DeltaNo is medium AND CoreNo is small AND ImageDir is medium AND CoreDir is medium AND DeltaPos is vertica
THEN A is Equal (0.28) OR A is Equal (0.72)
18: IF DeltaNo is small AND CoreNo is medium AND ImageDir is medium AND CoreDir is medium AND DeltaPos is vertica
THEN A is Equal (0.37) OR A is Equal (0.63)
19: IF DeltaNo is small AND CoreNo is medium AND ImageDir is medium AND CoreDir is medium AND DeltaPos is vertic
THEN <not defined>
20: IF DeltaNo is small AND CoreNo is none AND ImageDir is medium AND CoreDir is medium AND DeltaPos is vertical
THEN A is Equal (0.67) OR A is Equal (0.33)
21: IF DeltaNo is medium AND CoreNo is medium AND ImageDir is large AND CoreDir is large AND DeltaPos is both
THEN <not defined>
22: IF DeltaNo is medium AND CoreNo is none AND ImageDir is small AND CoreDir is large AND DeltaPos is left
THEN A is Equal (0.47) OR A is Equal (0.53)
23: IF DeltaNo is medium AND CoreNo is medium AND ImageDir is large AND CoreDir is medium AND DeltaPos is both
THEN <not defined>
24: IF DeltaNo is medium AND CoreNo is none AND ImageDir is medium AND CoreDir is medium AND DeltaPos is right
THEN A is Equal (0.31) OR A is Equal (0.69)

Table 6.13 Sample of fuzzy rules generated for class A



Internal Network 2

1: IF DeltaNo is small AND CoreNo is medium AND ImageDir is medium AND CoreDir is medium AND DeltaPos is right THEN T is Equal (0.67) OR T is Equal (0.33)

2: IF DeltaNo is medium AND CoreNo is small AND ImageDir is medium AND CoreDir is small AND DeltaPos is right THEN T is Equal (0.69) OR T is Equal (0.31)

3: IF DeltaNo is small AND CoreNo is medium AND ImageDir is small AND CoreDir is medium AND DeltaPos is right THEN T is Equal (0.59) OR T is Equal (0.41)

4: IF DeltaNo is medium AND CoreNo is medium AND ImageDir is small AND CoreDir is small AND DeltaPos is right THEN <not defined>

5: IF DeltaNo is small AND CoreNo is small AND ImageDir is large AND CoreDir is large AND DeltaPos is leftt THEN T is Equal (0.91) OR T is Equal (0.09)

6: IF DeltaNo is medium AND CoreNo is medium AND ImageDir is medium AND CoreDir is small AND DeltaPos is both THEN <not defined>

7: IF DeltaNo is medium AND CoreNo is medium AND ImageDir is large AND CoreDir is very large AND DeltaPos is bott THEN <not defined>

8: IF DeltaNo is small AND CoreNo is medium AND ImageDir is large AND CoreDir is small AND DeltaPos is both THEN T is Equal (0.68) OR T is Equal (0.32)

9: IF DeltaNo is medium AND CoreNo is small AND ImageDir is medium AND CoreDir is large AND DeltaPos is right THEN T is Equal (0.81) OR T is Equal (0.19)

10: IF DeltaNo is none AND CoreNo is small AND ImageDir is medium AND CoreDir is large AND DeltaPos is right THEN T is Equal (0.65) OR T is Equal (0.35)

11: IF DeltaNo is none AND CoreNo is none AND ImageDir is medium AND CoreDir is medium AND DeltaPos is right THEN T is Equal (0.19) OR T is Equal (0.81)

12: IF DeltaNo is none AND CoreNo is none AND ImageDir is medium AND CoreDir is small AND DeltaPos is right THEN T is Equal (0.29) OR T is Equal (0.81)

13: IF DeltaNo is small AND CoreNo is small AND ImageDir is large AND CoreDir is large AND DeltaPos is right THEN T is Equal (0.89) OR T is Equal (0.11)

14: IF DeltaNo is medium AND CoreNo is small AND ImageDir is large AND CoreDir is large AND DeltaPos is right THEN T is Equal (0.79) OR T is Equal (0.21)

15: IF DeltaNo is small AND CoreNo is medium AND ImageDir is large AND CoreDir is large AND DeltaPos is left THEN T is Equal (0.89) OR T is Equal (0.11)

16: IF DeltaNo is small AND CoreNo is small AND ImageDir is very large AND CoreDir is large AND DeltaPos is left THEN T is Equal (0.85) OR T is Equal (0.15)

17: IF DeltaNo is medium AND CoreNo is small AND ImageDir is very large AND CoreDir is large AND DeltaPos is left THEN T is Equal (0.88) OR T is Equal (0.22)

18: IF DeltaNo is small AND CoreNo is medium AND ImageDir is very large AND CoreDir is large AND DeltaPos is left THEN T is Equal (0.81) OR T is Equal (0.19)

19: IF DeltaNo is small AND CoreNo is small AND ImageDir is large AND CoreDir is large AND DeltaPos is right THEN T is Equal (0.91) OR T is Equal (0.09)

20: IF DeltaNo is small AND CoreNo is small AND ImageDir is small AND CoreDir is medium AND DeltaPos is vertical THEN T is Equal (0.87) OR T is Equal (0.13)

21: IF DeltaNo is small AND CoreNo is small AND ImageDir is large AND CoreDir is medium AND DeltaPos is vertical THEN T is Equal (0.88) OR T is Equal (0.12)

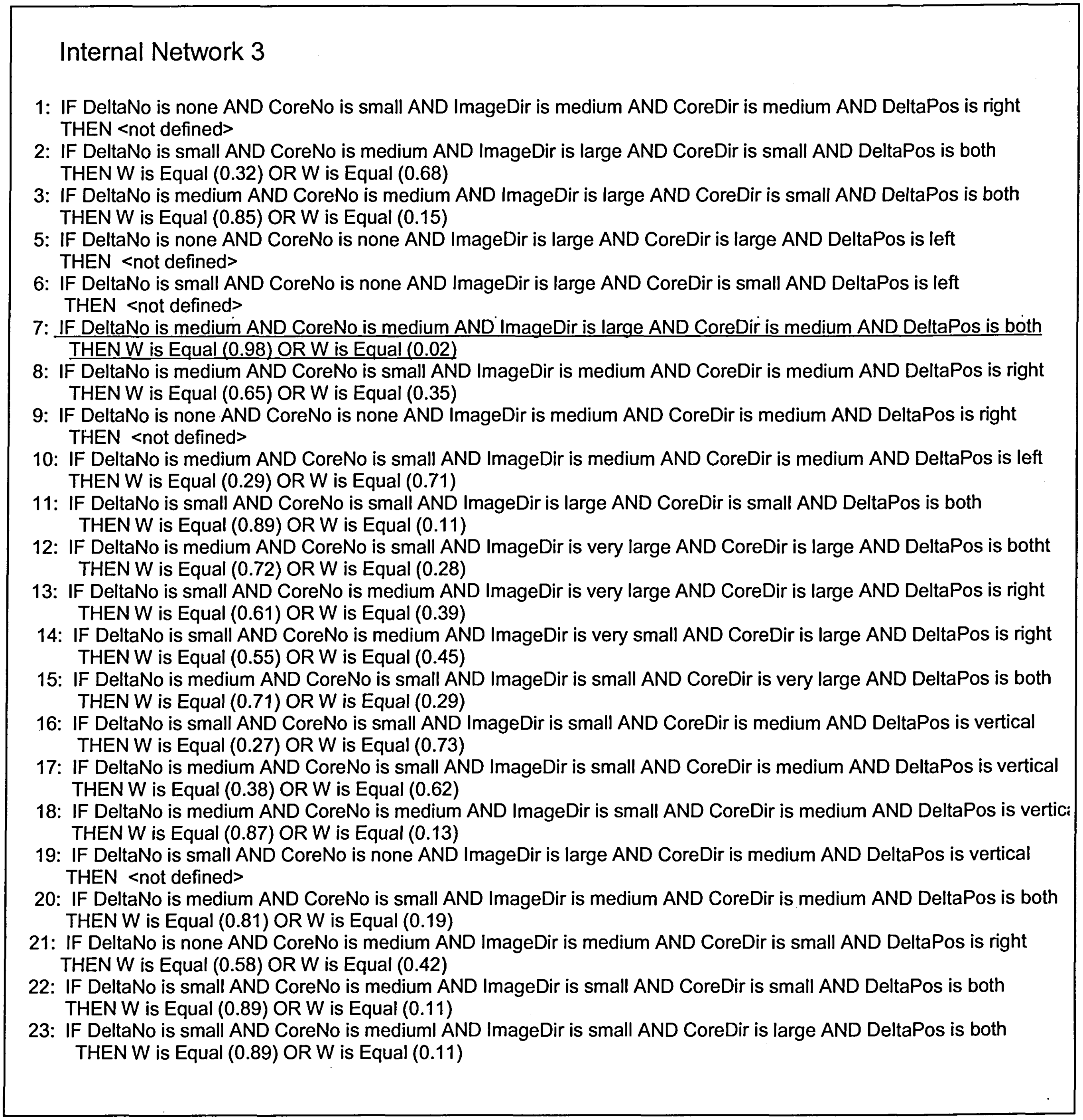
22: IF DeltaNo is small AND CoreNo is small AND ImageDir is medium AND CoreDir is medium AND DeltaPos is vertic THEN T is Equal (0.87) OR T is Equal (0.13)

23: IF DeltaNo is small AND CoreNo is small AND ImageDir is small AND CoreDir is medium AND DeltaPos is vertical THEN T is Equal (.90) OR T is Equal (0.10)

24: IF DeltaNo is small AND CoreNo is small AND ImageDir is medium AND CoreDir is medium AND DeltaPos is vertic THEN T is Equal (0.88) OR T is Equal (0.22)

25: IF DeltaNo is medium AND CoreNo is none AND ImageDir is medium AND CoreDir is medium AND DeltaPos is bot THEN <not defined>

Table 6.14 Sample of fuzzy rules generated for T class



```
Internal Network 3

 1: IF DeltaNo is none AND CoreNo is small AND ImageDir is medium AND CoreDir is medium AND DeltaPos is right
    THEN <not defined>
 2: IF DeltaNo is small AND CoreNo is medium AND ImageDir is large AND CoreDir is small AND DeltaPos is both
    THEN W is Equal (0.32) OR W is Equal (0.68)
 3: IF DeltaNo is medium AND CoreNo is medium AND ImageDir is large AND CoreDir is small AND DeltaPos is both
    THEN W is Equal (0.85) OR W is Equal (0.15)
 5: IF DeltaNo is none AND CoreNo is none AND ImageDir is large AND CoreDir is large AND DeltaPos is left
    THEN  <not defined>
 6: IF DeltaNo is small AND CoreNo is none AND ImageDir is large AND CoreDir is small AND DeltaPos is left
     THEN  <not defined>
 7: IF DeltaNo is medium AND CoreNo is medium AND ImageDir is large AND CoreDir is medium AND DeltaPos is both
     THEN W is Equal (0.98) OR W is Equal (0.02)
 8: IF DeltaNo is medium AND CoreNo is small AND ImageDir is medium AND CoreDir is medium AND DeltaPos is right
     THEN W is Equal (0.65) OR W is Equal (0.35)
 9: IF DeltaNo is none AND CoreNo is none AND ImageDir is medium AND CoreDir is medium AND DeltaPos is right
     THEN  <not defined>
10: IF DeltaNo is medium AND CoreNo is small AND ImageDir is medium AND CoreDir is medium AND DeltaPos is left
     THEN W is Equal (0.29) OR W is Equal (0.71)
11: IF DeltaNo is small AND CoreNo is small AND ImageDir is large AND CoreDir is small AND DeltaPos is both
      THEN W is Equal (0.89) OR W is Equal (0.11)
12: IF DeltaNo is medium AND CoreNo is small AND ImageDir is very large AND CoreDir is large AND DeltaPos is botht
      THEN W is Equal (0.72) OR W is Equal (0.28)
13: IF DeltaNo is small AND CoreNo is medium AND ImageDir is very large AND CoreDir is large AND DeltaPos is right
      THEN W is Equal (0.61) OR W is Equal (0.39)
 14: IF DeltaNo is small AND CoreNo is medium AND ImageDir is very small AND CoreDir is large AND DeltaPos is right
      THEN W is Equal (0.55) OR W is Equal (0.45)
 15: IF DeltaNo is medium AND CoreNo is small AND ImageDir is small AND CoreDir is very large AND DeltaPos is both
      THEN W is Equal (0.71) OR W is Equal (0.29)
 16: IF DeltaNo is small AND CoreNo is small AND ImageDir is small AND CoreDir is medium AND DeltaPos is vertical
      THEN W is Equal (0.27) OR W is Equal (0.73)
 17: IF DeltaNo is medium AND CoreNo is small AND ImageDir is small AND CoreDir is medium AND DeltaPos is vertical
     THEN W is Equal (0.38) OR W is Equal (0.62)
 18: IF DeltaNo is medium AND CoreNo is medium AND ImageDir is small AND CoreDir is medium AND DeltaPos is vertic
     THEN W is Equal (0.87) OR W is Equal (0.13)
 19: IF DeltaNo is small AND CoreNo is none AND ImageDir is large AND CoreDir is medium AND DeltaPos is vertical
    THEN  <not defined>
 20: IF DeltaNo is medium AND CoreNo is small AND ImageDir is medium AND CoreDir is medium AND DeltaPos is both
    THEN W is Equal (0.81) OR W is Equal (0.19)
21: IF DeltaNo is none AND CoreNo is medium AND ImageDir is medium AND CoreDir is small AND DeltaPos is right
    THEN W is Equal (0.58) OR W is Equal (0.42)
22: IF DeltaNo is small AND CoreNo is medium AND ImageDir is small AND CoreDir is small AND DeltaPos is both
    THEN W is Equal (0.89) OR W is Equal (0.11)
23: IF DeltaNo is small AND CoreNo is mediuml AND ImageDir is small AND CoreDir is large AND DeltaPos is both
     THEN W is Equal (0.89) OR W is Equal (0.11)
```

Table 6.15 Sample of fuzzy rules generated for W class

Internal Network 4

1: IF DeltaNo is none AND CoreNo is medium AND ImageDir is large AND CoreDir is medium AND DeltaPos is both
THEN <not defined>
2: IF DeltaNo is medium AND CoreNo is small AND ImageDir is large AND CoreDir is medium AND DeltaPos is right
THEN R is Equal (0.21) OR R is Equal (0.79)
3: IF DeltaNo is medium AND CoreNo is medium AND ImageDir is very large AND CoreDir is medium AND DeltaPos is both
THEN R is Equal (0.19) OR R is Equal (0.81)
4: IF DeltaNo is small AND CoreNo is small AND ImageDir is large AND CoreDir is large AND DeltaPos is left
THEN R is Equal (0.88) OR R is Equal (0.12)
5: IF DeltaNo is small AND CoreNo is medium AND ImageDir is very large AND CoreDir is small AND DeltaPos is right
THEN R is Equal (0.41) OR R is Equal (0.59)
6: IF DeltaNo is medium AND CoreNo is medium AND ImageDir is small AND CoreDir is small AND DeltaPos is right
THEN <not defined>
7: IF DeltaNo is medium AND CoreNo is small AND ImageDir is large AND CoreDir is small AND DeltaPos is right
THEN R is Equal (0.15) OR R is Equal (0.85)
8: IF DeltaNo is small AND CoreNo is small AND ImageDir is very large AND CoreDir is very large AND DeltaPos is left
THEN R is Equal (1.00) OR R is Equal (0.00)
9: IF DeltaNo is small AND CoreNo is medium AND ImageDir is large AND CoreDir is small AND DeltaPos is both
THEN R is Equal (0.22) OR R is Equal (0.78)
10: IF DeltaNo is medium AND CoreNo is medium AND ImageDir is large AND CoreDir is very large AND DeltaPos is both
THEN R is Equal (0.15) OR R is Equal (0.85)
11: IF DeltaNo is medium AND CoreNo is medium AND ImageDir is large AND CoreDir is large AND DeltaPos is both
THEN <not defined>
12: IF DeltaNo is none AND CoreNo is none AND ImageDir is small AND CoreDir is small AND DeltaPos is right
THEN <not defined>
13: IF DeltaNo is none AND CoreNo is small AND ImageDir is medium AND CoreDir is very large AND DeltaPos is left
THEN R is Equal (0.72) OR R is Equal (0.28)
14: IF DeltaNo is small AND CoreNo is none AND ImageDir is medium AND CoreDir is medium AND DeltaPos is left
THEN R is Equal (0.72) OR R is Equal (0.28)
15: IF DeltaNo is small AND CoreNo is none AND ImageDir is large AND CoreDir is very large AND DeltaPos is left
THEN R is Equal (0.85) OR R is Equal (0.15)
16: IF DeltaNo is small AND CoreNo is small AND ImageDir is very large AND CoreDir is large AND DeltaPos is right
THEN R is Equal (0.89) OR R is Equal (0.11)
17: IF DeltaNo is medium AND CoreNo is small AND ImageDir is large AND CoreDir is large AND DeltaPos is right
THEN R is Equal (0.65) OR R is Equal (0.35)
18: IF DeltaNo is small AND CoreNo is medium AND ImageDir is large AND CoreDir is large AND DeltaPos is left
THEN R is Equal (0.85) OR R is Equal (0.15)
19: IF DeltaNo is small AND CoreNo is medium AND ImageDir is large AND CoreDir is large AND DeltaPos is vertical
THEN R is Equal (0.75) OR R is Equal (0.25)
20: IF DeltaNo is small AND CoreNo is small AND ImageDir is very large AND CoreDir is large AND DeltaPos is vertical
THEN R is Equal (0.89) OR R is Equal (0.11)
21: IF DeltaNo is medium AND CoreNo is small AND ImageDir is very large AND CoreDir is very large AND DeltaPos is left
THEN R is Equal (0.85) OR R is Equal (0.15)
22: IF DeltaNo is small AND CoreNo is medium AND ImageDir is very large AND CoreDir is large AND DeltaPos is left
THEN R is Equal (0.82) OR R is Equal (0.18)
23: IF DeltaNo is medium AND CoreNo is small AND ImageDir is large AND CoreDir is very large AND DeltaPos is left
THEN R is Equal (0.82) OR R is Equal (0.18)
24: IF DeltaNo is small AND CoreNo is small AND ImageDir is small AND CoreDir is very large AND DeltaPos is vertical
THEN R is Equal (0.87) OR R is Equal (0.13)
25: IF DeltaNo is medium AND CoreNo is small AND ImageDir is very large AND CoreDir is medium AND DeltaPos is vertic
THEN R is Equal (0.78) OR R is Equal (0.22)
26: IF DeltaNo is small AND CoreNo is small AND ImageDir is small AND CoreDir is very large AND DeltaPos is vertical
THEN R is Equal (.80) OR R is Equal (0.20)

Table 6.16 Sample of fuzzy rules generated for R class

Internal Network 5

1: IF DeltaNo is small AND CoreNo is small AND ImageDir is small AND CoreDir is medium AND DeltaPos is right
THEN L is Equal (0.91) OR L is Equal (0.09)

2: IF DeltaNo is small AND CoreNo is small AND ImageDir is medium AND CoreDir is medium AND DeltaPos is right
THEN L is Equal (1.00) OR L is Equal (0.00)

3: IF DeltaNo is small AND CoreNo is small AND ImageDir is medium AND CoreDir is small AND DeltaPos is right
THEN L is Equal (0.91) OR L is Equal (0.09)

4: IF DeltaNo is medium AND CoreNo is small AND ImageDir is small AND CoreDir is small AND DeltaPos is right
THEN L is Equal (0.65) OR L is Equal (0.35)

5: IF DeltaNo is small AND CoreNo is medium AND ImageDir is small AND CoreDir is small AND DeltaPos is right
THEN L is Equal (0.65) OR L- is Equal (0.35)

6: IF DeltaNo is medium AND CoreNo is medium AND ImageDir is small AND CoreDir is small AND DeltaPos is right
THEN L is Equal (0.15) OR L is Equal (0.85)

7: IF DeltaNo is medium AND CoreNo is medium AND ImageDir is large AND CoreDir is small AND DeltaPos is leftt
THEN <not defined>

8: IF DeltaNo is small AND CoreNo is small AND ImageDir is medium AND CoreDir is medium AND DeltaPos is vertical
THEN L is Equal (0.95) OR L is Equal (0.05)

9: IF DeltaNo is small AND CoreNo is medium AND ImageDir is large AND CoreDir is small AND DeltaPos is both
THEN L is Equal (0.22) OR L is Equal (0.78)

10: IF DeltaNo is medium AND CoreNo is medium AND ImageDir is large AND CoreDir is small AND DeltaPos is right
THEN L is Equal (0.15) OR L is Equal (0.85)

11: IF DeltaNo is medium AND CoreNo is medium AND ImageDir is large AND CoreDir is large AND DeltaPos is both
THEN <not defined>

12: IF DeltaNo is medium AND CoreNo is none AND ImageDir is small AND CoreDir is large AND DeltaPos is left
THEN <not defined>

13: IF DeltaNo is none AND CoreNo is medium AND ImageDir is large AND CoreDir is small AND DeltaPos is both
THEN <not defined>

14: IF DeltaNo is medium AND CoreNo is medium AND ImageDir is large AND CoreDir is medium AND DeltaPos is both
THEN <not defined>

15: IF DeltaNo is medium AND CoreNo is smal AND ImageDir is medium AND CoreDir is medium AND DeltaPos is right
THEN L is Equal (0.81) OR L is Equal (0.19)

16: IF DeltaNo is none AND CoreNo is small AND ImageDir is medium AND CoreDir is medium AND DeltaPos is right
THEN L is Equal (0.88) OR L is Equal (0.12)

17: IF DeltaNo is none AND CoreNo is none AND ImageDir is medium AND CoreDir is medium AND DeltaPos is right
THEN L is Equal (0.35) OR L is Equal (0.65)

18: IF DeltaNo is none AND CoreNo is none AND ImageDir is medium AND CoreDir is small AND DeltaPos is right
THEN L is Equal (0.28) OR L is Equal (0.72)

19: IF DeltaNo is small AND CoreNo is small AND ImageDir is large AND CoreDir is very large AND DeltaPos is right
THEN L is Equal (0.78) OR L is Equal (0.22)

20: IF DeltaNo is small AND CoreNo is small AND ImageDir is medium AND CoreDir is large AND DeltaPos is right
THEN L is Equal (0.81) OR L is Equal (0.19)

21: IF DeltaNo is small AND CoreNo is medium AND ImageDir is large AND CoreDir is medium AND DeltaPos is right
THEN L is Equal (0.90) OR L is Equal (0.10)

22: IF DeltaNo is small AND CoreNo is medium AND ImageDir is very large AND CoreDir is medium AND DeltaPos is righl
THEN L is Equal (0.61) OR L is Equal (0.39)

23: IF DeltaNo is medium AND CoreNo is small AND ImageDir is very large AND CoreDir is large AND DeltaPos is right
THEN L is Equal (0.55) OR L is Equal (0.45)

24: IF DeltaNo is small AND CoreNo is small AND ImageDir is small AND CoreDir is medium AND DeltaPos is vertical
THEN L is Equal (0.79) OR L is Equal (0.21)

25: IF DeltaNo is small AND CoreNo is medium AND ImageDir is small AND CoreDir is medium AND DeltaPos is vertical
THEN L is Equal (0.65) OR L is Equal (0.35)

26: IF DeltaNo is small AND CoreNo is small AND ImageDir is large AND CoreDir is medium AND DeltaPos is vertical
THEN L is Equal (0.79) OR L is Equal (0.21)

27: IF DeltaNo is small AND CoreNo is small AND ImageDir is medium AND CoreDir is medium AND DeltaPos is vertical
THEN L is Equal (0.85) OR L is Equal (0.15)

Table 6.17 Sample of fuzzy rules generated for L class

6.5 Discussion

The results show that the worst classification performance is in the case of the arch types, A and T. For example, in confusion matrix of FNN the largest misclassification errors are shown to be between these two class types, that is, the misclassification of A as T is 2.0% and the misclassification of T as A is 2.25%. Similarly from the confusion matrix of MLP, the misclassification of A as T is 6.5% and the misclassification of T as A is 4.75%. From the confusion matrix of RBF, the misclassification of A as T is 10.5% and the misclassification of T as A is 10.25%. This is not surprising because both are arches and have similar features. By combining these two arch categories into a single class (four class), the error rate can be brought down to levels comparable with the other classes.

The result, on other hand, shows that the classification accuracy does not vary widely across the different classes. The poorer performance on the arch types may be due to the generalisation characteristic of neural networks, which causes misclassification among fingerprints with similar features. It is suggested that this can be overcome using a different feature scheme, which can distinguish between the classes. Alternatively, the occurrence of misclassification can be studied further, and the confusion probabilities used in resolving the final output classes.

Several approaches have been developed for fingerprint classification some of which were discussed in Chapter 3. These approaches can be broadly categorised into three main categories: (i) knowledge-based, (ii) structure-based, and syntactic. The knowledge-based fingerprint classification techniques use the locations of singular points to classify a fingerprint into five or four classes [Jain99c], [Hong99]. A knowledge-based approach tries to capture the knowledge of human expert by deriving rules for each category by hand-constructing the models and therefore, does not require training. Accuracies of 85% [Jain99c] and 87.5% [Hong99] have been reported on the NIST-4 database using these approaches. A structure-based approach uses the estimated orientation field in a fingerprint image for classification. An accuracy of 90.2% with 10% rejection [Wilson94] is reported on the NIST-4 database. A later version of this algorithm reported an accuracy of 92.2% with 0% rejection [Candela95] on the NIST-14 database, which is a naturally distributed database resulting in a better performance (in a naturally distributed database, the number of fingerprint images for particular fingerprint type is proportional to the probability of occurrence of that type in nature). However, this performance improvement should be expected since the NIST-14 database contains only small percentage of arch-type fingerprints, which pose the most difficulty for fingerprint classifiers, and the neural network used in the algorithm implicitly takes advantage of this information. A similar structure-based approach which uses B-spline curves to represent and classify fingerprints reported an accuracy of 96.5% with 0% rejection [Chong97]. A syntactic approach uses a formal grammar to represent and classify fingerprints. Hybrid approaches combine two or more approaches for classification. These approaches show some promise, although some of them have not been tested on large databases. For example, in [Fitz96] reported an accuracy of 85% with 0% rejection on 40 fingerprints, and [Tojo84] reported an accuracy of 91.5% with 0% rejection on 94 fingerprints. Cappelli et al. [Cappelli99] have proposed a fingerprint classification algorithm based on the multi-space KL transform applied the orientation field reported an accuracy of 99% with 20% rejection on the NIST-14. Recently, Prabhakar [Prabhakar01] proposed fingerprint classification based on Gabor filterbank representation applied on the NIST-4 database, and reported an accuracy of 97% with

1.8% rejection. Table 6.18 presents a summary of fingerprint classification accuracies compiled from the literature survey.

Authors	C	Features	Method	Acc. (RR)
Tojo and Kawagoe 1984	7	Singular points	Rule-based	91.5% (0%)
Blue et al. 1994	5	Orientation field	Neural network	92.8% (0%)
Wilson et al. 1994	5	Orientation field	Neural network	90.2% (10%)
Candela et al. 1995	6	Orientation field	Neural network	92.27% (0%)
Maio and Maltoni 1996	5 4	Orientation field	Inexact Graph Matching	85.5% 91.1%
Fitz and Green 1996	4	FFT	Nearest-neighbour	85% (0%)
Jain and Hong 1999	5 4	Singular points and ridges lines	Rule-based	87.5% (0%) 92.3%
Cappelli et al. 1999	5	Orientation field	Combination	99% (20%)
Salil Prabhakar 2001	5	Gabor response	Combination	97% (1.8%)
Mohamed 2002	5	Orientation field and singular points	Neural-network Fuzzy-neural	96.07% (0%) 96.19%

Table: 6.18 A comparison of fingerprint classification accuracies. *C: number of classes, Acc: accuracy, RR: reject rate.*

We believe that the FBI requirement of 1% classification error with maximum of 20% rejection rate is very challenging. Algorithms that have reported a performance close to or surpassing this requirement [Wilson97], [Cappelli99] have achieved their results on a naturally distributed database and, thus have circumvented the fact that the less frequently occurring classes are more difficult to classify. We have shown in this research that the features extracted and the classification scheme offer performance that exceeds many previous studies, and is comparable to the best result achieved in other studies.

6.6 Conclusion

This Chapter has presented an analysis of the implementation of a fingerprint features classification problem using radial basis functions, multilayer perceptron and fuzzy-neural network classifiers. We have proposed a fast and flexible fingerprint classification algorithm, which classifies input fingerprints into five categories according to the number of the core and delta (singular points), their relative (x,y) positions in an image, and their estimated directions. For the five-class problem, an average error rate of 2.575% was achieved without any rejection. A lower error rate can be achieved by adding a reject option, which is based on the quality index of the input image [Duda73]. However, if the classifier is allowed to reject some prints, i.e. to indicate that it is uncertain as to their class or that it does not accept the hypothesized class, then it can also achieve an error rate much lower that 2.57%. To implement the rejection, it is sufficient to set a confidence threshold then reject all prints for which the classifier produces a confidence below the threshold. The larger the threshold used, the greater is the percentage of the prints that are rejected (obviously), but also the smaller is the percentage of the accepted prints that are classified. The fuzzy-neural network classifier developed in this research is fast and highly transparent, compared to alternative algorithms.

Chapter 7

Conclusions and Further Work

7.1 Conclusions

With recent advances in fingerprint sensing technology and improvement in the accuracy and speed of fingerprint matching algorithms, reliable fingerprint classification is an attractive complement to the fingerprint identification system. The critical factor for widespread use of fingerprints is in meeting the performance (identification speed and accuracy) standards demanded by emerging applications. Unlike identification based on passwords or tokens, the reliability of fingerprint-based systems is very high and not easy to defraud. There will be a growing demand for faster, accurate and more reliable fingerprint classification algorithms, which can particularly handle poor quality images.

In spite of several years of research in pixel-based fingerprint image processing techniques, computerised image analysis systems are often unable to recognise characteristics that would be obvious to human visual inspection. The aim of this research has been to present implementation of a fingerprint classification problem combining conventional fingerprint image analysis, feature extraction, and fuzzy-neural classification techniques. Fingerprint classification is to categorise fingerprints into certain pre-specified categories based on global pattern configuration. If two fingerprints are from the same finger, they must belong to the same category.

Fingerprint classification provides an important first step in automatic fingerprint recognition systems and speed up the process of retrieval from a large database. An accurate automatic personal identification is critical to a wide range of application domains, such as access control, electronic commerce, criminal investigation and welfare benefits disbursements. Therefore, there is an increasing interest in inexpensive and reliable personal identification.

In order to process a fingerprint recognition system we need to deal with five main issues. These are, fingerprint acquisition, enhancement, feature extraction, classification, and matching. This research has studied a major sub-set of these issues, namely, enhancement, feature extraction and classification.

Although, fingerprint singular points provide very effective fingerprint class clues, methods relying on singular points alone may not be very successful due to lack of such information in some fingerprints and due to the difficulty in extracting these information from the noisy fingerprint images. As a result, the most successful approaches need to supplement the orientation field and ridge information, with reliable pattern recognition and classification. Neural and fuzzy-neural classifiers were proposed in this research for this purpose. The integration of fuzzy logic and neural network paradigms promises significant advantages for the realisation of flexible and intelligent classification systems in the future.

The complete system for fingerprint features classification is the combination system of a (e.g. NIST-4) database of fingerprints, Fingerprint Feature Extraction (FFE) algorithm, and fingerprint classification. This is presented in the block diagram of Figure 7.1. In the following we summarised the main contribution of this research.

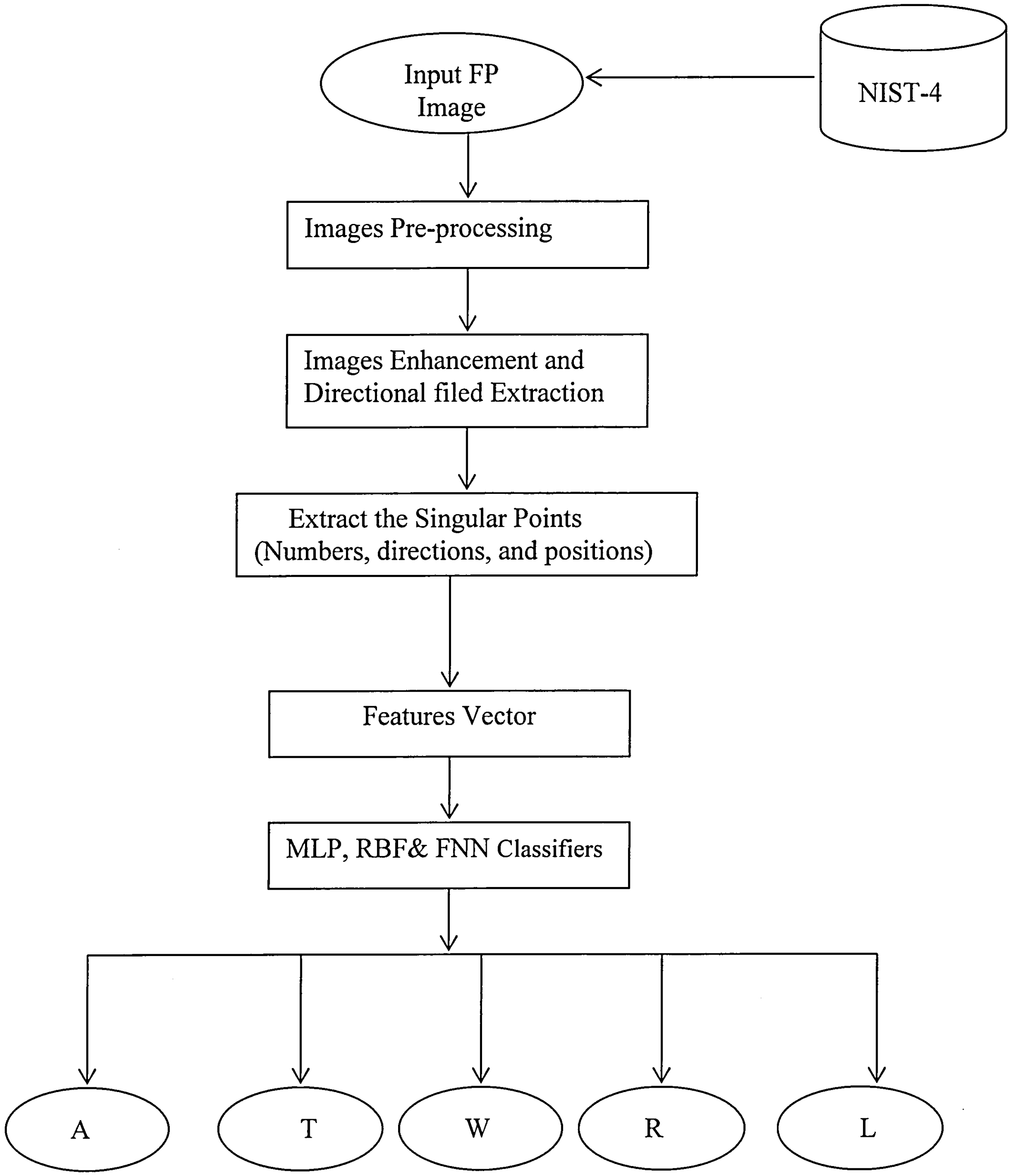

Input FP Image
NIST-4
Images Pre-processing
Images Enhancement and Directional filed Extraction
Extract the Singular Points (Numbers, directions, and positions)
Features Vector
MLP, RBF& FNN Classifiers
A
T
W
R
L

Figure: 7.1 Block diagram of the FFE and FNN Classification System

7.1.1 Pre-processing

Fngerprint image pre-processing includes segmentation and enhancement. Segmentation is a process to remove noise and artefacts, from an image sample and is often the key step in interpreting the image. We applied the threshold-based segmentation, as it is used extensively in many image-processing applications. The assumption is that different feature types will have a distinct frequency distribution and can be discriminated on the basis of the mean and standard deviation of each distribution. In this research, threshold of the grey level images to black and white is carried out using the Regional Average Thresholding (RAT) scheme or a General Threshold (GT).

7.1.2 Directional Image and Feature Extraction

The idea of a directional field image is to locate the singular-points (SPs), and then create a feature-encoded vector. A directional field describes the local orientations of the ridge and valley structures. In general, the directional field at some location in the image is estimated by averaging the directions in some window around the desired location. In this research, we developed a method for directional field estimation based on pixel-wise directional image. The studied directional image creates an *M*x*N* reduced image, which decreases the complexity and increases the speed of the processing. Moreover, we estimate the whole image direction, which is computed in four main directions

The feature extraction technique, checks the orientations around individual pixels, computes directional fields, estimates the whole image direction and then detects the

SP and their relative positions. A feature encoder stage, assigns a class label to the features extracted.

7.1.3 Classification

Statistical properties of the feature vectors were investigated using the SPSS statistical package. It was shown that the feature vectors could not be classified easily using simple techniques such as linear discrimunant analysis. The research investigated alternative classification methods, namely, neural and fuzzy-neural. Two neural classifiers were implemented namely MLP and RBF. In the fuzzy-neural classifier, the neural network part is primarily used for learning. The neural network also automatically generates fuzzy logic rules. In addition even after training, the neural networks keeps updating the fuzzy logic rules as it learns more and more from its inputs. Fuzzy logic, on the other hand, is used to provide transparency and understandability of the classification process.

7.1.4 Methodology

The methodology of this research was evaluated and tested using fingerprint images from the NIST-4 database. At a first glance, the fingerprint classification problem appears to be rather simple. But because of large intra-class and small inter-class variations in global pattern configuration and poor quality of input images, the desired accuracy of 1% error rate with a low percentage of reject rate is very difficult to achieve.

Although, good classification performance was obtained in this research, the main problem, which hampers the classification of a fingerprint image, namely, quality of the fingerprints, was not addressed. If the quality is not of an acceptable standard, fingerprint classification becomes extremely difficult. Some of the problems can be solved with pre-processing techniques, such as, segmentation and enhancement.

7.2 Summary of Contributions

This thesis has described and discussed a fingerprint feature extraction and fingerprint classification algorithm. Our attempt was to enable the accuracy and speed up of automatic fingerprint classification algorithms to improve the efficiency of the existing identification systems. Therefore, contribution to knowledge by this study can be summarised as follows:

- A review of human identification issues and biometrics-based identification systems,
- A new algorithm for fingerprint feature extraction, which results in more robust classification,
- Implementation of neural and fuzzy-neural classifiers for fast, efficient fingerprint classification.

7.3 Further Research

It is envisaged to carry on this research further, in order to achieve a complete automatic fingerprint recognition system. Therefore, future research will include the following:

- Fingerprint enhancement techniques to deal with poor quality images and different skin conditions,

- Apply thinning methodologies to enhance feature extraction,

- The issues involved in integrating fingerprint-based identification with other biometric or non-biometric technologies may also constitute an important research topic.

References

[Adler97] Adler F. H., "Physiology of the Eye; Clinical Application", C. V. Mosby Company, London, 1997.

[Akhan95] Akhan M. B., Emirroglu I., and Bahari E. G., "A Flexible Fingerprint Identification System", European Convention on Security and Detection, (Conf. Publ. No.408), IEE, London, pp. 284-287, May, 1995.

[Almansa00a] Almansa A. and Cohen L., "Fingerprint Image Matching by Minimization of A Thin-Plate Energy Using A Two-Step Algorithm with Auxiliary Variables", Proceedings Fifth IEEE Workshop on Applications of Computer Vision, IEEE Computer Society Los Alamitos, CA, USA, pp. 35-40, 2000.

[Almansa00b] Almansa A. and Lindeberg T., "Fingerprint Enhancement by Shape Adaptation of Scale-Space Operators with Automatic Scale Selection", IEEE-Transactions-on-Image-Processing, Vol. 9, No. 12, pp. 2027-2042, Dec. 2000.

[Ammar96] Ammar H. H. and Maio Z., "Performance of Parallel Algorithms for Fingerprint Image Comparison System", Proceedings

Eighth IEEE Symposium on Parallel and Distributed Processing, IEEE Comput. Soc. Press, Los Alamitos, CA, USA, pp. 410-413, 1996.

[Ammar98] Ammar H. H., Zeng S. and Maio Z., "Parallel Processing and Fingerprint Image Comparison", International Journal of Modelling and Simulation, Vol. 18, No. 2, pp. 85-99, 1998.

[Armstrong02] Armstrong I., "Biometrics Technology Making Moves in the Security Game", the International Journal of Computer Security, pp. 26-32, March 2002.

[Asker00] Asker M. Bazen M. and Sabih H. G, "Computational Intelligence in Fingerprint Identification", Proceedings of 2nd IEEE Benelux Signal Processing Symposium (SPS-2000), Hilvarenbeek, the Netherlands, pp. 1-4, March 23-24, 2000.

[Asker01] Asker M. Bazen M. and Sabih H. G., "Extraction of Singular Points from Directional fields of Fingerprints", Proceeding of the Annual CTIT Workshop, Enschede the Netherlands, pp. 111-114, February 2001.

[Baldi93] Baldi P. and Chauvin Y., "Neural Networks for Fingerprint Recognition", Neural Computation, Vol. 5, No. 3, pp. 402-418, 1993.

[Ballan98a] Ballan M., "Directional Fingerprint Processing", 4th International Conference on Signal Processing, IEEE, Piscataway, NJ, USA, Vol. 2, pp. 1064-1067, 1998.

[Ballan98b] Ballan M., Sakarya A. and Evans B L, "Fingerprint Classification Technique Using Directional Images", 31th Asilomar Conference on Signals, Systems and Computers, IEEE Comput. Soc, Los Alamitos, CA, USA, Vol. 1, pp. 101-104, 1998.

[Ballard82] Ballard D. H. and Brown C. M, "Computer Vision", Prentice-Hall, Englewood Cliffs, NJ, USA, 1982.

[Basak97] Basak J., Pal N. R. and Patel P. S., "Thinning in Binary and Grey Images", A Connectionist Approach, Journal of the Institution of Electronics and Telecommunication Engineers, Vol. 42, No. 4-5, pp. 305-313, 1997.

[Bastian95] Bastian A., "An Approach Towards Linguistic Instructions Understanding Using The Concept Of Flexible Linguistic Variables", Proceedings of 1995 IEEE International Conference on Fuzzy Systems, The International Joint Conference of the Fourth IEEE International Conference on Fuzzy Systems and The Second International Fuzzy Engineering Symposium, IEEE, New York, NY, USA, Vol. 2, pp. 927-934, 1995.

[Battley37] Battley H., "A New and Practical Method of Classifying and Filing Single Fingerprints and Fragmentary Impressions", New Haven: Yale University Press, 1937.

[Berfanger99] Berfanger D. M. and George N., "All-Digital Ring-Wedge Detector Applied to Fingerprint Recognition", Applied Optics, Vol. 38, No. 2, pp. 357-369, 1999.

[Bicz99] Bicz W., Banasiak D., and Bruciak P., "Fingerprint Structure Imaging based on an Ultrasound Camera", The Biometrics Report, SJB Services, ISBN 1-900-18009, 1999.

[Blue94] Blue, J. L., Canfela, G. T., Grother P. J., Chellappa R. and Wilson C. L., "Evaluation of Pattern Classifiers for Fingerprint and OCR Applications", Pattern Recognition, Vol. 27, No. 4, pp. 485-501, 1994.

[Blume89] Blume P., "The Personal Identity Number in Danish Law", Computer Law and Security Report, Vol. 5, No. 3, pp. 10-13, 1989.

[Bradley94] Bradley J. and Brislawn C. M., "The Wavelet/Scalar Quantization Standard for Digital Fingerprint Images", In Proceedings IEEE ISCAS-94, London, Vol. 3, pp. 205--208, 1994.

[Broomhead88] Broomhead D. S. and Lowe D., "Multivariable Functional Interpolation and Adaptive Networks", Complex Systems, No. 2, pp. 321-355, 1988.

[Bruyne82] Bruyne P. and Orum J., "Reducing Storage Requirements of Digitized Fingerprint Images", Proceedings-of-the-1982-Carnahan-Conference-on-Security-Technology, University of Kentucky, Lexington, KY, USA, pp. 1-6, 1982.

[Bum95] Bum R. L. and Chin H. C., "A Modified Fuzzy Competitive Learning Algorithm for Image Coding", Journal-of-the-Korea-Information-Science-Society, Vol. 22, No. 6, pp. 860-867, 1995.

[Bunke97] Bunke H. Yan Q. and Kandel A., "A Genetic Fuzzy Neural Network for Pattern Recognition", Proceedings-of-the-Sixth-IEEE International Conference on Fuzzy Systems, Cat-No. 97CH36032, IEEE, NY, USA, Vol. 1, pp. 75-78, 1997.

[Candela95] Candela G. T., Grother P. J., Watson C. I., Wilkinson R. A., and Wilson C. L., "A Pattern-Level Classification Automation System for Fingerprints (PCASYS)", NIST Tech. Report NISTIR 5647, August 1995.

[Cappelli99] Cappelli R., Maio D. and Maltoni D., "Fingerprint Classification based on Multi-space KL", Proceedings of AutoID'99 (Workshop on Automatic Identification Advances Technologies), Summit (NJ), pp. 117-120, 1999.

[Chaum85] Chaum D., "Security without Identification, Card Computer to Make Big Brother Obsolete", Comun. ACM 28, 10, pp. 1030-1044, 1985.

[Chen93] Chen Y., Sheng M., and Yongbao H. E., "A Method of Pattern Recognition Based on Synthetic Technology of Fuzzy Logic and Neural Network", Proceedings TENCON '93, IEEE Region 10 Conference on Computer, Communication, Control and Power Engineering, IEEE, New York, Vol. 4, pp. 577-580,1993.

[Cheung87] Cheung Y. S. and Yip W. M., "A Personal Computer-Based Fingerprint Identification System", Proceeding IEEE Asian Electronic Conference, pp. 290-294, Hong Kong, 1987.

[Chong92] Chong M. S., Gay R. L., Tan H. N. and Liu J., "Automatic Representation of Fingerprints for Data Compression by B-Spline Functions", Pattern Recognition, Vol. 25, No. 10, pp. 1199-1210, 1992.

[Chong97] Chong S. M., Han T N., Liu J. and Gay K. L., "Geometric Framework for Fingerprint Image Classification", Pattern Recognition, Vol. 30, No. 9, pp. 1475-1488, 1997.

[Chow72] Chow C. K. and Kaneko T., "Automatic Boundary Detection of the Left Ventricle from Cineangiograms", Computers and Biomedical Research, Vol. 5, No. 4, pp. 388-410, 1972.

[Chow94] Chow C. R., Chu. C. H., Naraghi P. M, and Hegde M., "Genetic Algorithm Approach to Fault-tolerant Neural Networks Design", World Congress on Neural Networks, Lawrence Erlbaum Associates, Hillsdale, NJ, USA, Vol. 3, pp. 696-701, March, 1994.

[Chu91] Chu C. H. and Kottapalli M. S., "A Genetic Algorithm Approach to Visual Model-Based Halftone Pattern Design", Proceedings of the SPIE, the International Society for Optical Engineering, Vol. 1, pp. 470-481, 1991.

[Clarke91] Clarke R.A., "The Tax File Number Scheme: A Case Study of Political Assurances and Function Creep", Policy 7, 4, Summer 1991.

[Clarke92] Clarke R.A., "The Resistible Rise of the Australian National Personal Data System", Software Learning Journal, Vol. 5, No. 1, pp. 15-21, January 1992.

[Clarke94a] Clarke R. A., "Human Identification in Information Systems: Management Challenges and Public Policy Issues", the Journal of Information Technology and People, Vol. 7, No. 4, pp. 6-37, 1994.

[Clarke94b] Clarke R.A., "Dataveillance by Governments, the Technique of Computer Matching", Information Technology and People, pp. 23-29, June 1994.

[Clarke94c] Clarke R.A., "Information Technology, Weapon of Authoritarianism or Tool of Democracy", Proceeding of World Congress, International Conference of Information Processing, Hamburg, PP. 36-47, September 1994.

[Coetzee90] Coetzee L. and Botha E. C., "Fingerprint Recognition with Neural Network Classifier", Proceeding of the First South African Workshop on Pattern Recognition, Vol. 1, pp. 33-40, 1990.

[Coetzee91] Coetzee L. and Botha E. C., "Preprocessing of two Dimensional Fingerprint Images for Fingerprint", South African Symposium on Communications and Signal Processing IEEE, NY, USA, pp. 69-73, 1991.

[Coetzee92] Coetzee L., "Fingerprint Recognition, Thesis Submitted in the Faculty of Electronics and Computer Engineering", University of Pretoria, Pretoria, 1992.

[Coetzee93] Coetzee L. and Botha E. C., "Fingerprint Recognition in Low Quality Images", Pattern-Recognition., Vol. 26, No. 10, pp. 1441-1460, 1993.

[Cover65] Cover T. M., "Geometrical and Statistical Properties of Systems of Linear Inequalities with Applications in Pattern Recognition", IEEE Transactions on Electronic Computers, EC-14, PP. 326-334, 1965.

[Cox95] Cox E., "Relational Database Queries Using Fuzzy Logic", AI Expert, pp 23-29, January, 1995.

[Dagli97] Dagli C H and Ozbayoglu A M, "Unsupervised Hierarchical Fingerprint Matching", IEEE International Conference on Neural Networks- Conference Proceeding, Vol. 3, pp. 1439-1442, 1997.

[Daugman93] Daugman J. G., "High Confidence Visual Recognition of Persons by a Test of Statistical Independence", IEEE Transactions Pattern Analysis and Machine Intelligence, Vol. 15, No. 11, pp. 1148-1161, 1993.

[Davies92] Davies S., "Big Brother Australia's Growing Web of Surveillance", Simon and Shuster, Sydney, 1992.

[Davies94] Davies S., "Touching Big Brother: How Biometric Technology Will Fuse Flesh and Machine", Information Technology and People, Vol. 7, No. 4, pp. 162-171, 1994.

[De00] De R K, Basak J. and Pal S. K., "Neuro-Fuzzy Feature Evaluation with Theoretical Analysis", Neural-Networks, Vol. 12, No. 10; pp. 1429-1455, 2000.

[Deriche93] Deriche R. and Blaszka T., "Recovering and Characterizing Image Features Using an Efficient Model Based Approach", Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, IEEE Comput. Soc. Press, Los Alamitos, CA, USA, pp. 530-535, 1993.

[Ding96] Ding Y. and Clarkson T., "Fingerprint Recognition by pRAM Neural Networks", Neural Networks World, Vol. 6, No. 4, pp. 525-543, 1996.

[Drets99] Drets G. A. and Liljenstrom H. G., "Fingerprint Sub-classification and Singular Point Detection Procedure using Neural Network Approach", Neural Networks in Engineering Systems., Proceedings of the International Conference on Engineering Applications of Neural Networks, Eng. Assoc, Turku, Finland, pp. 71-74, 1999.

[Duda73] Duda R. O. and Hart P. E, "Pattern Classification and Scene Analysis", John Wiley and Sons, 1973.

[EAN-Int01] EAN International, "EAN.UCC Solutions for the Textile and Apparel Industry", Bar Coding and Electronic Data Interchange, EDITEX, EURATEX, Belgium, 2001.

[Eaton86] Eaton J.W., "Card-Carrying Americans, Privacy, Security and the National ID Card Debate", Rowman and Little-field, Totowa NJ, 1986.

[Edward97] Edward J. R., "Neural Network Data Analysis Using Simulnet", Springer, 1997.

[Emiroglu97] Emiroglu I. and M. Akhan, "Pre-processing of Fingerprint Images", European Conference on Security and Detection, ECOS97 Incorporating, Symposium on Technology Used for Combating Fraud IEE, London, pp. 147-151, April 1997.

[FBI01] Federal Bureau of Investigation (FBI), "The Science of Fingerprints", U.S. Department of Justice, 2001.

[FBI84] Federal Bureau of Investigation (FBI), "The Science of Fingerprints Classification", U.S. Department of Justice, 1984.

[Fishler85] Fishler M and Bolles R, "Perceptual Organization and Curve Partitioning", IEEE-Transact. on Pattern Analysis and Machine Intelligence, Vol. PAMI-8, No. 1, pp. 100-105, Dec. 1985.

[Fitz96] Fitz A. P. and Green R. J., "Fingerprint Classification Using A Hexagonal Fast Fourier Transform", Pattern Recognition, Vol. 29, No. 10, pp. 1587-1597, 1996.

[Florack94] Florack L. M. J., Haar B., Romeny M., J. J. and Koenderink M. A., "General Intensity Transformations and Differential Invariant", Journal of Mathematical Imaging and Vision, Vol. 4, No. 2, PP. 171-187, 1994.

[Forstner87] Forstner W. and Gulch E., "Automatic Orientation and Recognition in Highly Structured Scenes", Proceedings-of-the-SPIE, The International Society for Optical-Engineering, PP. 2-13, 1987.

[Fu82] Fu K. S., "Syntactic Pattern Recognition and Applications", Prentice-Hall, Englewood Cliffs, NJ, USA, pp. 596-604, 1982.

[Gader00] Gader P D, Khabou M. A, and Koldobsky A., "Morphological Regularization Neural Networks", Pattern-Recognition, Vol. 33, No. 6, pp. 935-944, 2000.

[George99] George N. and Berfanger D. M., "All-Digital Ring-Wedge Detector Applied to Fingerprint Recognition", Applied Optics, Vol. 38, No. 2, pp. 357-369, 1999.

[Gorsky90] Gorsky W. I. and Mehrotra R., "Index based Object Recognition in Pictorial Data Management", Computer Vision, Graphics, and Image Processing, Vol. 52, No. 3, pp. 416-436, 1990.

[Gouet00] Gouet V., Montesinos P. and Pele D., "Matching Colour Uncalibrated Images using Differential Invariants", Image-and-Vision-Computing, Vol. 18, No. 9, pp. 659-671, 2000.

[Halici96] Halici U. and Ongun G., "Fingerprint Classification Through Self-Organizing Feature Maps Modified to Treat Uncertainties", Proceeding of The IEEE, Vol. 84, No. 1, pp. 1497-1512, Oct. 1996.

[Harry83] Harry E B, "Handbook of Bar Coding Systems", Van Nostrand Reinhold Company, New York, 1983.

[Heijden94] Heijden F. D., "Image Based Measurement Systems", John Wiley and Sons Ltd., Chichester, 1994.

[Hong96] Hong L., Jain A. K., Pankanti S., and Bolle R., "Fingerprint Enhancement", In Proceeding of 1st IEEE Workshop on Application of Computer Vision, Sarasota, FL, pp. 202-207, 1996.

[Hong97] Hong L., Jain A. K., Bolle R. and Pankanti S., "An Identity Authentication System Using Fingerprints", Proc. of First Int'l Conf. on Audio and Video-Based Biometric Person Authentication, Switzerland, pp. 103-110, March 1997.

[Hong98] Hong L. and Jain A. K., "Integrating Faces and Fingerprints for Personal Identification", IEEE Trans. Pattern Analysis Machine Intelligence, Vol. 20, No.12, pp. 1295-1307, December, 1998.

[Hong99] Hong L. and Jain A. K., "Classification of Fingerprint Images", Proceedings of 11th Scandinavian Conference on Image Analysis, Kangerlussuaq, Greenland, June 7-11, 1999.

[Howard01] Howard T, "Biometrics Come of Age", Personal Computer World Magazine, pp. 32-33, January 2001.

[Hung93] Hung D. and Douglas C., "Enhancement and Feature Purification of Fingerprint Images", Pattern Recognition, Vol. 26, No. 11, pp. 1661-1671, 1993.

[IAI83] International Association for Identification, "Fingerprint Filing Sequence Formula Identification News", International

Association for Identification, Mendota Hieghts MN 55120-1120, USA, February 1983.

[Ibrahim01] Ibrahim I. A., "A Brief Illustrated Guide to Understanding Islam" ISBN: 9960-34-011-2 Library of Congress Catalogue Card Number: 97-67654 Published by Darussalam, Publishers and Distributors, Houston, Texas, USA, 2001.

[Isenor86] Isenor D. K and Zaky S. G., "Fingerprint Identification Using Graph Matching", Pattern Recognition, Vol. 19, No. 2, pp. 113-122, 1986.

[Jablonowsky94] Jablonowsky M, "Fuzzy Risk Analysis: Using AI systems", AI Expert, pp 34-37, 1994.

[Jain00a] Jain A. K., Prabhakar S., Hong L., and Pankanti S., "Filterbank-based Fingerprint Matching", IEEE Transactions of Image Processing, Vol. 9, No. 5, pp. 846-859, 2000.

[Jain00] Jain A. K., Duin R. P., and Mao J., "Statistical pattern recognition", a Review, IEEE Trans., Pattern Analysis Machine Intelligence, Vol. 22, No. 1, pp. 4-37, 2000.

[Jain01] Jain A. K. and Pankanti S., "Automated Fingerprint Identification and Imaging Systems", Advances in Fingerprint Technology, (Ed. H. C. Lee and R. E. Gaensslen), Elsevier Science, New York, 2001.

[Jain97] Jain A. K., Hong L. and Bolle R., "On Line Fingerprint Verification", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 19, No. 4, PP. 302-314, 1997.

[Jain99a] Jain A. K., Bolle R., and Pankantis S., "Biometrics Personal Identification in Networked Society", New York, Kluwer Academic Publishers, 1999.

[Jain99b] Jain A. K., Prabhakar S., and Hong L., "A Multichannel Approach to Fingerprint Classification", IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 21, No., 4, pp. 348-359, 1999.

[John96] John S., "Improved Image Quality of Live-Scan Fingerprint Scanners Using Acoustic Backscatter Measurements", In Proceedings of Biometric Consortium Ninth Meeting, San Jose, California, pp. 301-312, 11-12, June 1996.

[Jouko98] Jouko L., Jorma L., and Erkki O., "Pattern Recognition", Chapter in Image Processing and Pattern Recognition Book, Neural Networks Systems Techniques and Applications, Academic Press, Vol. 5, pp. 1-10, 1998.

[Julian00] Julian A, "Biometrics Advanced Identity Verification", Springer, NY, USA, 2000.

[Junhong94] Junhong N. Linkens D. A., "FCMAC: a Fuzzified Cerebellar Model Articulation Controller with Self-Organizing Capacity", Automatica, Vol. 30, No. 4, pp. 655-664, April 1994.

[Kameshwar80] Kameshwar C. V. and Black K., "Type Classification of Fingerprints, A Syntactic Approach", IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 2, No. 3, pp. 1187-1216, 1980.

[Karu96] Karu K. and Jain A.K., "Fingerprint Classification", Pattern Recognition, Vol. 29, No. 3, pp. 389-404, 1996.

[Kasabov00] Kasabov N and Iliev G, "Hybrid System For Robust Recognition of Noisy Speech Based on Evolving Fuzzy Neural Networks and Adaptive Filtering", Proceedings of the IEEE-INNS-ENNS, International Joint Conference on Neural Networks, Neural Computing, IEEE Comput. Soc., Los Alamitos, CA, pp. 91-96, 2000.

[Kawagoe84] Kawagoe M. and Tojo A., "Fingerprint Pattern Classification", Pattern Recognition, Pergamon Press, Oxford, England, Vol. 17, No. 3, pp. 295-303, 1984.

[Khan85] Khan M. M. and Al-Hilali M. T., "Interpretation of Meanings of The NOBLE QURA'AN In The English Language", Islamic University Al-Madina Al-Munawwara, KSA, Dar-us-Salam Publications, Kingdom of Saudi Arabia, March 1985.

[Kohonen88] Kohonen K., "Self-Organization and Associative Memories", Berlin, Springer-Verlag, 1988.

[Kosko90] Kosko B. and Kong S.G., "Comparison of Fuzzy and Neural Truck Backer-Upper Control Systems", International Joint Conference on Neural Networks, IEEE, New York, NY, USA, Vol. 3, pp. 349-358, 1990.

[Kruschke89] Kruschke J. K., "Improving Generalisation in Back-Propagation Networks with Distributed Bottle-necks", International Journal of Neural Networks, Research and Applications, Vol. 1, No.3, pp.187-193., July 1989.

[Lam92] Lam L., Lee S. W., and Suen C. Y., "Thinning Methodologies A Comprehensive Survey", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 14, No. 9, pp. 869-885., September 1992.

[L-Dict95] Longman Dictionary, "Longman Group Ltd, Longman Hpiuse", Harlow Essex, England, p.705, 1995.

[Lee00] Lee S Y et al., "Automatic Segmentation of Multi-Spectral Brain Images Using a Neuro-Fuzzy Algorithm", Neuro-Fuzzy Pattern Recognition, Series in Machine Perception and Artificial Intelligence, Vol. 41, 2000.

[Lee90] Lee C. C., "Fuzzy Logic in Control Systems Part I and II", IEEE Trans. SMC, pp. 404-435, 1990.

[Lee96] Lee S. Y., Ham Y. K. and Park R. H., "Recognition of Human Front Faces Feature Extraction and Neuro-Fuzzy Algorithm", Pattern Recognition, Vol. 29, No. 11, PP. 1863-1876, 1996 .

[Leondes98a] Leondes C. T., "Image Processing and Pattern Recognition", Academic Press, NY, USA, 1998.

[Leondes98b] Leondes C. T., "Image Processing and Pattern Recognition", Neural Network Systems Techniques and Applications, Vol. 5, Academic Press, London, 1998.

[Li91] Li C L and Schenk T, "An Accurate Camera Calibration for the Aerial Image Analysis", Proceedings, 10th International Conference on Pattern Recognition, IEEE Comput. Soc. Press, Los Alamitos, CA, USA; Vol. 1, pp.207-209 , 1991.

[Lin00] Lin J. S., "Clustering Problem Using Fuzzy C-Means Algorithms and Unsupervised Neural Networks", Neuro-Fuzzy Pattern Recognition, Series in Machine Perception and Artificial Intelligence, Vol. 41, 2000.

[Lin99] Lin C. T. and Chung I. F., "A Reinforcement Neuro-Fuzzy Combiner for Multiobjective Control Systems", IEEE Trans on Fuzzy Systems, Man and Cybernetics, Part B (Cybernetics), Vol. 29, No. 6, pp.726-744, Dec. 1999.

[Lindeberg94] Lindeberg T., "Scale-Space Theory in Computer Vision", Kluwer Academic Publishers, Boston, 1994.

[Linkens96] Linkens D. A. and Nyongesa H. O., "Learning Systems in Intelligent Control: an Appraisal of Fuzzy, Neural and Genetic Algorithms Control Applications", Proceedings IEE, Control Theory and Application, Vol. 143, No. 4, pp. 367-386, 1996.

[Luk91] Luk A., Leung S. H., Lee C. K. and Lau W. H., "A Two-Level Classifier for Fingerprint Recognition", Processing IEEE Int. Symp. Circuits and System, Singapore, pp. 2625-2628, 1991.

[Lumini99a] Lumini A., Maio D., and Maltoni D., "Inexact Graph Matching for Fingerprint Classification", Machine Graphics and Vision, Special Issue on Graph Transformations in Pattern Generation and CAD, Vol. 8, No. 2, PP. 231-248, 1999.

[Lumini99b] Lumini A., Cappelli R., Maio D., and Maltoni D., "Fingerprint Classification by Directional Image Partitioning", IEEE Transactions on Pattern Analysis Machine Intelligence, Vol. 21, No. 5, PP. 402-421, 1999.

[Maio96] Maio D., and Maltoni D., "A Structural Approach to Fingerprint Classification", in Proceedings 3th ICPR, Vienna 96, PP. 71-83, Aug. 1996.

[Maio99] Maio D., Cappelli R,, and Maltoni D., "Fingerptint Classification based on Multi-space KL", Proceedings of AutoID'99 (Workshop on Automatic Identification Advances Technologies), Summit (NJ), pp. 117-120, Oct. 1999.

[Mallat89] Mallat S. G., "A Theory of Multiresolution Signal Decomposition the Wavelet Representation", IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 11, No. 7, pp. 674-693, 1989.

[Marija01] Marija J. N., "SPSS 10.01 Guide to Data Analysis", Prentice-Hall, Inc., New Jersey, 2001.

[Marr82] Marr D, "Vision", W.H. Freeman, NY, 1982.

[McCabe92] McCabe R., Willson C., and Grub D., "Research Considerations Regarding FBI-IAFIS Tasks and Requirements", NISTIR 4892, NIST, July 1992.

[M-Dict81] Macquarie Dictionary, "Macquarie Dictionary of English Language", p. 879, London, 1981.

[Mehtre89] Mehtre B. M. and Chatterjee B., "Segmentation of Fingerprint Images using Composite Method", Pattern Recognition, Vol. 22, No. 4, pp. 381-385, 1989.

[Mehtre93] Mehtre B. M., "Fingerprint Image Analysis for Automatic Identification", Machine Vision and Applications, Vol. 6, pp. 124-139, 1993.

[Mehtre98] Mehtre B. M., Murthy N. N., Kapoor S. and Chatterjee B., "Segmentation of Fingerprint Images Using the Directional Image", Pattern Recognition, Vol. 20, No. 4, pp. 429-435, 1998.

[Minsky69] Minsky M. L. and Papert S. A., "Perceptrons", MIT Press, Cambridge MA, 1969.

[Minsky88] Minsky M. L. and Papert S. A., "Perceptrons, Expanded Edition", Cambridge MA, MIT Press, 1988.

[Moayer75] Moayer B. and Fu K. S., "A Syntactic Approach to Fingerprint Pattern Recognition", Pattern Recognition, Vol. 7, No. 1, pp. 1-23, 1975.

[Moayer76] Moayer B. and Fu K. S., "An Application of Stochastic Languages to Fingerprint Pattern Recognition", Pattern Recognition, Vol. 8, No. 3, pp. 173-179, 1976.

[Moayer86] Moayer B. and Fu K. S., "A Tree System Approach for Fingerprint Pattern Recognition", IEEE-Transactions-on-Pattern-Analysis-and-Machine Intelligence, Vol. PAMI-8, No.3, pp. 376-387, 1986.

[Mohamed00] Mohamed M. M. and Nyongesa H. O., "Fingerprint Recognition System Using Fuzzy Neural Techniques", Proceeding of 8th International Conference in AI Applications, Egypt, Cairo, pp. 295-305, 2000.

[Mohamed01] Mohamed S. M. and Nyongesa H. O., "Automatic Fingerprint Classification System Using Fuzzy Neural", Proceeding of the 2001 International Conference on AI, Las Vegas, Nevada, USA, Vol. 1, pp. 395-401, June, 2001.

[Mohamed02] Mohamed S. M. and Nyongesa H. O., "Automatic Fingerprint Classification System Using Fuzzy Neural", 2002 IEEE World Congress on Computational Intelligence, IEEE International Conference on Fuzzy Systems, FUZZ-IEEE'02, Proceedings (Cat. No.02CH37291), Piscataway, NJ, USA, vol. 1, pp. 358-362, 2002.

[Mohamed97] Mohamed S. M., "Hand-written Character Recognition System Using Artificial Neural Networks Technology", MSc Thesis, Sudan University of Science and Technology, 1997.

[Mohamed99] Mohamed S. M. and Nyongesa H. O, "Image Pattern Recognition Using Fuzzy/Self-Organising Network", the Proceeding of the 6th

UK Workshop on Fuzzy Systems, Brunel University, Uxbridge, UK, pp. 115-120, September, 1999.

[Mohrman97] Mohrman D., "Biometrie ALS Quantensprung", W\&S, pp. 2833, Huethig Verlag, Heidelberg, July 1997.

[Monro93] Monro D. M.and Sherlock B. G., "A Model for Interpreting Fingerprint Topology", Pattern Recognition, Vo1. 26, No. 7, pp. 1047-1055, 1993.

[Monro94] Monro D. M., Sherlock B. G., and Millard K., "Fingerprint Enhancement By Directional Fourier Filtering", IEE Proceedings in Vision Image, Signal Processing, Vol. 141, No. 2, pp. 87-94, 1994.

[Moore88] Moore R.T., "Automated Fingerprint Identification Systems Glossary of Terms and Acronyms", American National Standard ANSI/IAI 2-1988, July 1988.

[Moraitakis00] Moraitakis I. and Fargues M. P., "Feature Extraction of Intra-Pulse Modulated Signals Using Time-Frequency Analysis", Proceedings of 21st Century Military Communications, Architectures and Technologies for Information Superiority (Cat. No.00CH37155). IEEE, Piscataway, NJ, USA, Vol. 2, pp. 737-741, 2000.

[Muenz99] Muenz J., "The Uniform Code Council Focuses on Supply Chain Management", EDI-Forum: The Journal of Electronic Commerce, Vol. 12, No.1 pp.36-41, 1999.

[Murthy92] Murthy N. N. and Srinivasan V. S., "Detection of Singular Point in Fingerprint Images", Pattern Recognition, Vol. 25, No. 2, pp. 139-153, 1992.

[Nalwa94] Nalwa V S, "The Basis of Computer Vision", Computer Systems and Education, Proceedings of the International Conference on Computer Systems and Education in Honour, Tata McGraw-Hill, New Delhi, India, pp. 130-144, 1994.

[Nelson92] Nelson M. M. and Illingworth W. T., "A Practical Guide to Neural Nets", Texas Instruments, 1992.

[Neto97] Neto H. V and Borges D. L., "Fingerprint Classification with Neural Networks", Proceedings of the Brazilian Symposium on Neural Networks, ABRN, Los Alamitos, CA, USA, pp. 66-72, 1997.

[Neuframe01] http://www.neusciences.com/Products/neuframe4_Download.htm

[NIST00] National Institute of Standards and Technology (NIST), American National Standard for Information Systems, "Data Format for the Interchange of Fingerprint, Facial, Scar Mark and Tattoo Information", NIST Special Publication 500-245, USA Department of Commerce, 2000.

[Nyongesa01] Nyongesa H. O., Otieno A. W. and Rosin L. P. "Neural Fuzzy Analysis of Determinated Composites from Shearography Imaging", Composite Structure, Vol. 54, PP. 313-318, Elsevier, 2001.

[Nyongesa98] Nyongesa H. O., "Enhancing Neural Control Systems by Fuzzy Logic and Evolutionary Reinforcement", Neural Computing and Applications, Springer-Verlag London, Vol. 7, pp. 121-130, 1998.

[O-Dict76] Oxford Dictionary, "Oxford Advanced Letter's Dictionary", Oxford University Press, p. 604, 1976.

[O'Hagan91] O'Hagan M., "A Fuzzy Neuron Based Upon Maximum Entropy Ordered Weighted Averaging", Uncertainty in Knowledge Bases, 3rd International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems, Springer-Verlag, Berlin, Germany, pp. 598-609, 1991.

[Okyay98] Okyay K., Zadeh L. A., Burhan T. and Imre J. R., "Computational Intelligence: Soft Computing and Fuzzy-Neuro Integration With Applications", Nato Asi Series Vol. 162, 1998.

[Ongun96] Ongun G. and Halici U., "Fingerprint Classification Through Self Organizing Feature Maps Modified to Treat Uncertainties", Proc. of the IEEE, Vol. 84, No. 10, PP. 1497-1512, Oct. 1996.

[Paderes84] Paderes F C, Forstner W and Mikhail E M, "Rectification of Single and Multiple Frames of Satellite Scanner Imagery Using Points and Edges Control", NASA Symposium on Mathematical Pattern Recognition and Image Analysis, Amsterdam, Netherlands, pp. 108-112, 1984.

[Pal00] Pal N R and Chakraborty D., "Mountain and Subtractive Clustering Method: Improvements and Generalizations", International

Journal of Intelligent Systems, Vol. 15, No. 4, pp.329-341, April 2000.

[Perona98] Perona P., "Orientation Diffusions", IEEE Trans. on Image Processing, Vol. 7, No. 3, pp. 457-467, 1998.

[Pineda88] Pineda F. J., "Generalization of Backpropagation to Recurrent and Higher Order Neural Networks", Neural Information Processing Systems, New York, pp. 602-611, 1988.

[Prabhakar01] Prabhakar S., "Fingerprint Classification and Matching using Filterbank", Department of Computer Science and Engineering", PhD Thesis, Michigan State University, 2001.

[Prabhakar97] Prabhakar S., Jain A. K., "Decision-level Fusion in Fingerprint Verification", Multiple Classifier Systems, Second International Workshop, Proceedings (Lecture Notes in Computer Science Vol.2096). Springer-Verlag, Berlin, pp. .88-98, 1997.

[Quran592] God the Creator Revealed Verse, "Quran", in the Noble Quran, Chapter 75, Verse 4 (Surat Alqiyamah.75:4), 592.

[Rafael93] Rafael C. G. and Richard E. W., "Digital Image Processing", Addison-Wesley Publishing Company, 1993.

[Rao74] Rao C. K., Prasada B. and Sarma K. R., "An Automatic Fingerprint Classification System", Proceeding Second Int., Conf. Pattern Recognition, Copenhagen, Denmark, pp. 180-184, 1974.

[Rao80] Rao C. K. and Black K., "Type Classification of Fingerprints", IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 2, pp. 223-231, 1980.

[Ratha95a] Ratha N. K., Jain A.K., and Rover D.T., "An FPGA-based Point Pattern Matching Processor with Application to Fingerprint Matching", In Proceedings, Conference on Computer Architectures for Machine Perception, Como, Italy, pp. 394-401, 18-20, September 1995.

[Ratha95b] Ratha N. K., Chen S., and Jain A. K., "Adaptive Flow Orientation Based Feature Extraction in Fingerprint Images", Pattern Recognition, Vol. 28, No. 11, pp. 1657-1672, 1995.

[Ratha96] Ratha N. K., Karu K., Chen S., and Jain A. K., "Real-time Matching System for Large Fingerprint Database", IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 18, No. 8, pp. 799-813, 1996.

[Roddy97] Roddy A. and Stosz J., "Fingerprint Features Statistical Analysis and System Performance Estimates", Proceedings of IEEE, Vol. 85, No. 9, pp. 1390-1421, 1997.

[Rudy98] Rudy S., "Algorithmic Techniques and their Applications", Department of IS and Computing, National University of Singapore, Academic Press, Image Processing and Pattern Recognition, Vol. 5, pp. 287-319, 1998.

[Rumelhart86] Rumelhart D. E., Hinton G E. and Williams R J., "Learning Internal Representations by Error Backpropagation", Parallel Distibuted Processing, Vol. 1, MIT Press, Cambridge, MA, 1986.

[Ryoo00] Ryoo Y. J., Lim Y C, and Kim K H, " Classification of Materials Using Temperature Response Curve Fitting and Fuzzy Neural Network", Sensors and Actuators A (Physical), Vol. A94, No. 1-2, pp.11-18, 2000.

[Samuel00] Samuel B. Green, Neil J. Salkind and Theresa M. A., "Using SPSS for Windows, Analysing and Understanding Data", Second Edition, Published by Prentice-Hall, Inc., New Jersey 2000.

[Sheng01] Sheng C. C. Wei S. L., "Robust Neurofuzzy Controller Design of a Class of Uncertain Multivariable Nonlinear Systems", Proceedings of the 2001 IEEE International Conference on Control Applications, IEEE, Piscataway, NJ, USA, 1207 pp. 902-907, 2001.

[Simon94] Simon H., "Neural Networks, A Comprehensive Foundation", Macmillan College Publishing Company, New York, 1994.

[Srinivasan92] Srinivasan V. S. and Murthy N. N., "Detection of Singular Points in Fingerprint Images", Pattern Recognition, Vol. 25, No. 2, pp. 139-153, February 1992.

[Suen99] Suen C Y, Liu K and Strathy N. W., "Sorting and Recognizing Cheques and Financial Documents", Document Analysis Systems: Theory and Practice,Third IAPR Workshop, DAS'98.

Selected Papers (Lecture Notes in Computer Science), Springer-Verlag, Berlin, Germany, Vol. 5, pp. 173-187, 1999.

[Tang91] Tang Y. Y., Cheng H. D., and Suen C., "Transformation Ring Projection (TRP) Algorithm and Its VLSI Implementation", Character and Handwriting Recognition, Expanding Frontiers, World Scientific Publishing Co. Pte. Ltd., Singapore, 1991.

[Tojo84] Tojo A. and Kawagoe M., "Fingerprint Pattern Classification", Pattern Recognition, Vol. 18, No. 3, pp. 201-210, 1984.

[Tsao93] Tsao E. C., Wei C. Lin and Chin T C., "Constraint Satisfaction Neural Networks for Image Recognition", Pattern Recognition, Vol. 26, No. 4, pp. 523-567, 1993.

[USDJ74] U.S. Department of Justice, "The Science of Fingerprints", Washington, D.C., 1974.

[Verma89] Verma M R, and Chatterjee B, "Partial Fingerprint Pattern Classification", Journal of the Institution of Electronics and Telecommunication Engineers, Vol. 35, No.1, pp. 28-33, 1989.

[Vermwa87] Vermwa M. R., Majumdar A. K.and Chatterjee B., "Edge Detection in Fingerprints", Pattern Recognition, Vol. 20, No. 5, pp. 513-523, 1987.

[Walker86] Walker G., "End-User Searching: The Beginning or the End?" Reference-Librarian, No.14; Spring-Summer, pp. 39-51, 1986.

[Watson92] Waston C. I., Wilson C. I., "NIST Special Database 4, Fingerprint Database", National Institute of Standards and Technology, Advanced Systems Division, Image Recognition Group, 1992.

[Watson94] Watson C. I., Candela J., Grother P., "Comparison of FFT Fingerprint Filtering Methods for Neural Network Classification", Technical Report NISTIR 5493, NIST, September 1994.

[Wegstein69] Wegstein J. H., "A Semi-automated Single Fingerprint Identification System", NBS Technical Note 481, April 1969.

[Wegstein82] Wegstein J. H., "An Automated Fingerprint Identification System", Technical Report 500-89, National Bureau of Standards, Bethesda, Maryland, 1982.

[Werbos74] Werbos P. J., "Beyond Regression: New Tools for Predication and Analysis in the Behavioural Sciences", Ph.D. Thesis, Harvard University, Cambridge, MA, 1974.

[Wilson94] Wilson C. L., Candela G. T. and Watson C., "Neural Networks Fingerprint Classification", Journal of Artificial Neural Networks Vol. 1, No. 2, pp. 203-228, 1994.

[Wilson97a] Wilson C. L., Watson C.I., Paek E. G., "Combined Optical and Neural Network Fingerprint Matching", Optical Pattern Recognition Vol. 3, SPIE Proceedings of the International Society For Optical Engineering, Vol. 3073, pp. 373-382, April 1997.

[Wilson97b] Wilson C. L., Blue J. L., and Omidwar O. M., "Neurodynamics of Learning and Network Performance", Journal of Electronic Imaging, Vol. 6, No. 3, pp. 379-385, 1997.

[Wolff94] Wolff U, "Parameter-free Information-Preserving Surface Restoration", Nuclear-Physics-B, Proceedings Supplements, Vol. 34, pp. 243-245, 1994.

[Wu94] Wu K., Narasimhalu A. D., Mehtre B. M., Lam C. P., and Gao Y. J., "Identifying Faces and Fingerprints using Multiple Retrievals", IEEE Multimedia, pp. 27-38, Summer 1994.

[Xiao91] Xiao Q. and Raafat H., "Fingerprint Image Post-Processing, A Combined Statistical and Structural approach", Pattern Recognition, Vol. 24, No. 10, pp. 985-992, 1991

[Xiao95] Xiao Q. and Raafat H., "A Content Based Retrieval Engine for Multimedia Information Systems", Multimedia Systems, Springer Verlag, Vol. 3, pp. 25-41, 1995.

[Yager94] Yager R. R., Goldstein L S, and Mendels E., "Ernest, FUZMAR: an Approach to Aggregating Market Research Data Based On Fuzzy Reasoning", Fuzzy Sets and Systems, Vol. 68, No. 1, pp. 1-11, 1994.

[Yong92] Yong C., Minghao S. and Yongbao H., "A Method of Pattern Recognition Based Upon Synthetic Technology of Fuzzy Logic and Neural Network", Department of Computer Science, Fudan University Shanghai 200433 P. R. China, 1992.

[Yoshtaka91] Yoshitaka A., Kishida S., Hirakawa M., and Ichikawa T., "Fingerprint Image Postprocessing, a Combined Statistical and Structural Approach", Pattern Recognition, Vol. 24, No. 10, pp. 985-992, 1991.

[Zadeh73] Zadeh L. A., "Outline of a New Approach to the Analysis of Complex Systems and Decision Processes", IEEE Transactions on Systems, Man and Cybernetics Vol. SMC-3, No. 1, pp. 28-44, 1973.

[Zadeh84] Zadeh L. A., "Making Computer Think Like People", IEEE Spectrum, Vol. 2, No. 8, pp. 26-32, 1984.

[Zamperoni95] Zamperoni P., "Model-free Texture Segmentation Based on Distances Between First-Order Statistics", Digital-Signal-Processing, Vol. 5, No. 4, pp. 197-225, Oct. 1995.

[Zhang93] Zhang D., Kamel M.and Elmasry M. I., "Fuzzy Clustering Neural Network using Fuzzy Competitive Learning", World Congress on Neural Networks, International Neural Networks Society Annual Meeting, NY, USA, pp. 179-183, 1993.

[Zhou96] Zhou R W and Quek C., "A Pseudo Outer-Product Based Fuzzy Neural Network", Neural Networks, Vol. 9, No. 9, pp.1569-1581, 1996.

[Zsolt99] Zsolt M., Alessandro F., and Kovacs V., "Fingerprint Minutiae Extraction from Skeletonized Binary Images", Journal of the Pattern Recognition Society, Vol. l, No.32, pp. 877-889, 1999.

GLOSSARY

We including some abbreviations and terms definitions of frequently used when discussing fingerprint-based biometric in this thesis as follows:

1:1: A one-to-one match of a fingerprint against a stored fingerprint.

1:N: A one-to-many search of a fingerprint against an entire database of fingerprints.

AFIS: Automatic Fingerprint Identification System.

AFRS: Automatic Fingerprint Recognition System.

ANC: Automatic Network Construction.

Biometrics: Personal biological or behavioural characteristics that distinguish one person from another.

COG: Centre of Gravity

EAN: the "European Article Numbering Association" was created in 1977 as a non- profit body to develop a UPC-compatible system.

FAR: False Accept Rate error

FBI: The Federal Bureau of Identification

FC: Fingerprint Classification.

FEE: Automatic Fingerprint Feature Extraction.

Fingerprint: an impression of the skin pattern on the inner surface of a finger tip, used for purpose of identification and verification.

Finger-scan: The finger is placed on a prepared surface, usually glass, and a picture is captured.

FL: Fuzzy Logic

FNN: Fuzzy-Neural Network

FRR: False Reject Rate error

GT: General Thresholding

H-ID: Human Identification

IAFIS: The FBI's Integrated Automated Fingerprint Identification System

ID: Identification

Identification: the user submits his/her live sample and the system attempts to identify a database of templates. More complex than verification and may generate a multiple result.

IS: Information System

IT: Information Technology

JPEG: The Joint Photographic Experts Group

Live Scan: Optical ten print scanner for production of a standard ten-print card for AFIS use.

LRO: Local Ridge Orientation

Matching Algorithms: Series of software codes used to match finger minutiae.

Minutiae: Unique features (anomalies) of a fingerprint that occur where ridge endings begin end and/or bifurcate.

MLP: Multilayer perceptron neural network

NIST: National Institute of Standards and Technology

NNs: Neural Networks

Pattern Based: Method of matching the entire pattern of the finger image.

PINs: Personal Identification Number, commonly used with bank ATM cards and to access accounts.

PR: Pattern Recognition.

RAT: Regional Average Thresholding

RBF: Radial Basis Functions neural network

RCP: Ridge Continuity Point

REP: Ridge Ending Point

RMP: Ridge Meeting Point

Singular Points: are the Delta and Core points

SP: Singular Points

Template: A format used to store fingerprint minutiae, size ranges from 60 - 500 bytes per finger image.

Verification: identity is claimed by calling a particular template from storage by PIN or presentation of a token, and then presenting a live sample for comparison, resulting in a match/on match according to predefined parameters.

Appendix 1

NIST Special Database-4 Fingerprint Image File Format

1. Introduction

NIST Special Database 4 contains 8-bit grey scale images of randomly selected fingerprints. The database is being distributed for use in the development and testing of automated fingerprint classification systems on a common set of images. The CD-ROM contains 4000 (2000 pairs) fingerprints stored in NIST's IHead raster data format and compressed using a modified JPEG lossless compression algorithm. Each print is 512 X 512 pixels with 32 rows of white space at the bottom of the print. Approximately 636 Megabytes of storage are needed when the prints are compressed where as 1.1 Gigabytes are needed when uncompressed (1.6 : 1 average compression ratio). In the database fingerprints are classified into one of five categories (L = left loop, W = whirl, R = right loop, T = tented arch, and A = arch) with an equal number of prints from each class (400). Each filename contains a reference to the hand and digit number so the classes can be converted to other classification techniques (i.e. radial and ulnar). All classes are stored in the NIST IHead id field of each file, allowing for comparison with hypothesised classes.

2. Fingerprint File Format

Image file formats and effective data compression and decompression are critical to the usefulness of image archives. Each fingerprint was digitised in 8-bit grey scale form at

19.6850 pixels/mm (500 pixels/inch), 2-dimensionally compressed using a modified JPEG lossless algorithm, and temporarily archived onto computer magnetic mass storage. Once all prints were digitised, the images were mastered and replicated onto ISO-9660 formatted CD-ROM discs for permanent archiving and distribution. After digitisation, certain attributes of an image are required to correctly interpret the 1-dimensional pixel data as a 2-dimensional image. Examples of such attributes are the pixel width and pixel height of the image. These attributes can be stored in a machine readable header prefixed to the raster bit stream. A program which manipulates the raster data of an image is able to first read the header and determine the proper interpretation of the data which follows it.

Numerous image formats exist, but most image formats are proprietary. Some are widely supported on small personal computers and others on larger workstations. A header format named IHead has been developed for use as a general purpose image interchange format. The IHead header is an open image format which can be universally implemented across heterogeneous computer architectures and environments. Both documentation and source code for the IHead format are publicly available and included with this database. IHead has been designed with an extensive set of attributes in order to adequately represent both binary and grey level images, to represent images captured from different scanners and cameras, and to satisfy the image requirements of diversified applications including, but not limited to, image archival/retrieval, character recognition, and fingerprint classification. Figure A2.1 illustrates the IHead format.

Header Length

ASCII Format Image Header

8 bit Grey Scale Raster Stream

11010100110100111101001011 10

- Representing the digital scan across the page left to right, top to bottom
- 8 bits to a pixel
- 256 levels of grey
- 1 pixel is packed into a single byte of memory

Figure A1.1. An illustration of the Ihead raster file format

Since the header is represented by the ASCII character set, IHead has been successfully ported and tested on several systems including UNIX workstations and servers, DOS personal computers, and VMS mainframes. All attribute fields in the IHead structure are of fixed length with all multiple character fields null-terminated, allowing the fields to be loaded into main memory in two distinct ways. The IHead attribute fields can be parsed as individual characters and null-terminated strings, an input/output format common in the `C' programming language, or the header can be read into main memory using record-oriented input/output. A fixed-length field containing the size in bytes of the header is prefixed to the front of an IHead image file as shown in Figure A1.1.

The IHead structure definition written in the `C' programming language is listed in Figure A1.2.

```
/***********************************************************/

/*    File Name: IHead.h                                   */
/*    Package:  NIST Internal Image Header                 */
/*    Author:   Michael D. Garris                          */
/*    Date:     2/08/90                                    */
/***********************************************************/

/* Defines used by the ihead structure */
#define IHDR_SIZE    288      /* len of hdr record (always even bytes) */
#define SHORT_CHARS  8        /* # of ASCII chars to represent a short */
#define BUFSIZE      80       /* default buffer size */
#define DATELEN      26       /* character length of data string */

typedef struct ihead{
  char id[BUFSIZE];                    /* identification/comment field */
  char created[DATELEN];               /* date created */
  char width[SHORT_CHARS];             /* pixel width of image */
  char height[SHORT_CHARS];            /* pixel height of image */
  char depth[SHORT_CHARS];             /* bits per pixel */
  char density[SHORT_CHARS];           /* pixels per inch */
  char compress[SHORT_CHARS];          /* compression code */
  char complen[SHORT_CHARS];           /* compressed data length */
  char align[SHORT_CHARS];             /* scanline multiple: 8|16|32 */
  char unitsize[SHORT_CHARS];          /* bit size of image memory units */
  char sigbit;                         /* 0->sigbit first | 1->sigbit last */
  char byte_order;                     /* 0->highlow | 1->lowhigh*/
  char pix_offset[SHORT_CHARS];        /* pixel column offset */
  char whitepix[SHORT_CHARS];          /* intensity of white pixel */
  char issigned;                       /* 0->unsigned data | 1->signed data */
  char rm_cm;                          /* 0->row maj | 1->column maj */
  char tb_bt;                          /* 0->top2bottom | 1->bottom2top */
  char lr_rl;                          /* 0->left2right | 1->right2left */
  char parent[BUFSIZE];                /* parent image file */
  char par_x[SHORT_CHARS];             /* from x pixel in parent */
  char par_y[SHORT_CHARS];             /* from y pixel in parent */
}IHEAD;
```

Figure A1.2 . The IHead `C' programming language structure definition.

Appendix 2

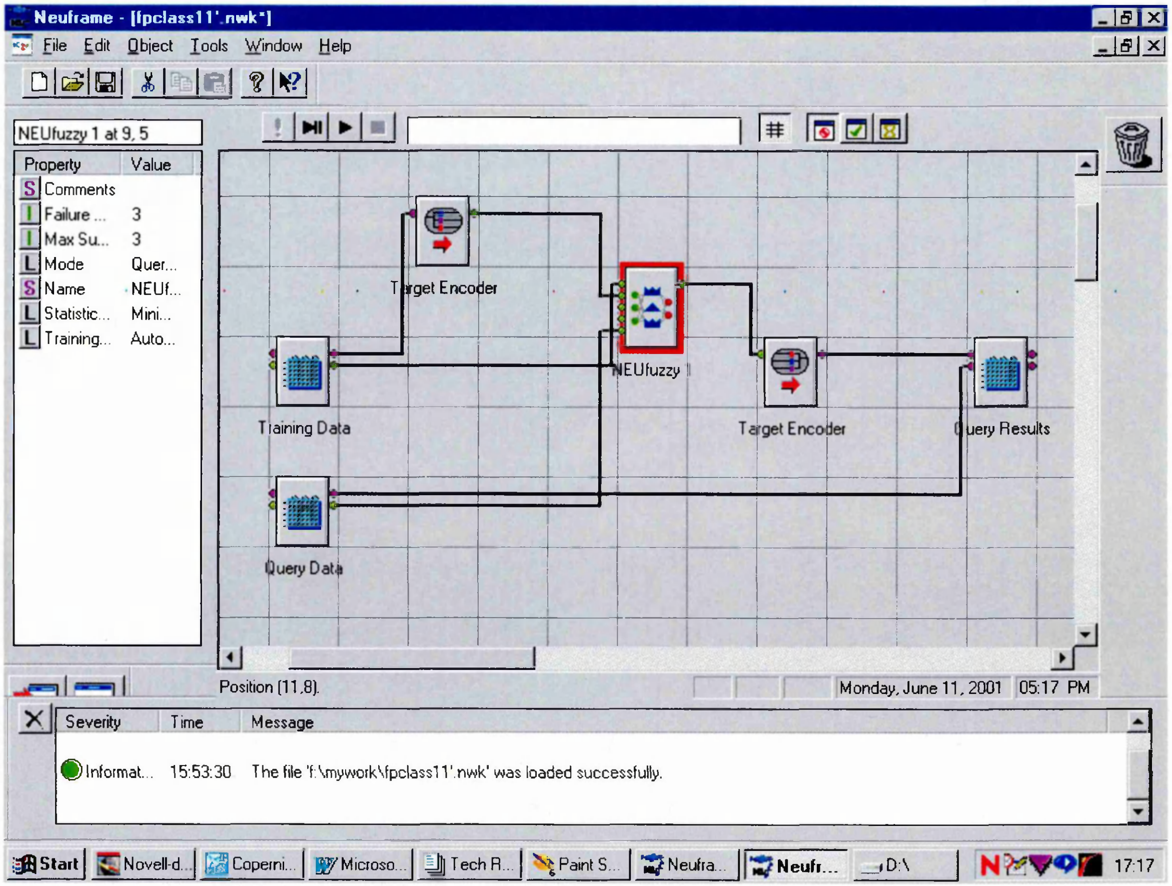
Implementation Steps of Fuzzy-Neural Classifier

2.1 Fuzzy-Neural Network Construction

We relied on Neuframe^{TM} commercially available educational software. The Neuframe's Neufuzzy object has three modes of operation:

1. Automatic Network Construction (ANC): used when you have no knowledge about the rules governing the problem but you have data describing it. ANC will extract rules from the data, where the data will contain such information. When this is complete you may go back and modify the rules at any time based on your subsequent "expert" knowledge and experience.
2. Weight Rule Confidence Training: used when we have some "expert" knowledge of the rules governing the problem we are addressing and we then tune the rules up subsequently using available data.
3. Fuzzy Logic: used when there is no data available and all the knowledge for the rules is supplied by an "expert" who also may subsequently modify the rules in the light of further experience.

In our implementation of fingerprint feature classification, we used the automatic network construction method (ANC). ANC mode is used to automatically build fuzzy-neural model and enables the expert to have 'the last say' and to tune the automatically generated model according to their expert knowledge. It provides the ability to automatically construct a fuzzy application model using fuzzy-neural logic techniques. The ANC mode runs an optimisation process, which tries a number of combinations of inputs and membership functions and reduces the number of parameters used to the minimum, required for a robust model. The structure of the ANC is shown in Figure A*2.1*. As shown in this figure the objects that are used by ANC from the Template Network are Datasheets, Encoder, and Neufuzzy. Once the feature data model has been generated from the feature encode using FFE, ANC allows the expert to then inspect and edit the fuzzy-neural logic structure, the fuzzy set memberships and the fuzzy rule base. The Fuzzy-neural module also provides weight rule confidence (WRC) training mode. The WRC mode in Neufuzzy enables an expert to build the fuzzy network structure, the fuzzy rule set and fuzzy set memberships from his or her expert knowledge. This enables us to start with our expert knowledge and then tune it with data, which used in the process of defuzzification. The Neufuzzy network structure, which includes the fuzzy rule set and fuzzy set memberships settings are then locked and are not, altered during the WRC training. Once this is done, the Neufuzzy training then adjusts the rule weights of the expert defined fuzzy model configuration, based on the information in actual data. The network and fuzzy variables are represented graphically and may be edited directly in the form of the above rules.



Property	Value
S Comments	
I Failure ...	3
I Max Su...	3
L Mode	Quer...
S Name	NEUf...
L Statistic...	Mini...
L Training...	Auto...

Severity	Time	Message
Informat...	15:53:30	The file 'f:\mywork\fpclass11'.nwk' was loaded successfully.

Figure A2.1 Neufuzzy's Automatic Network Construction.

2.2 Steps for Implementation of ANC

Data used: Fingerprint Features Data. This data consists of 5 input columns: CoreNo, DeltaNo, ImageDir, CoreDir, and DeltaPos; and 5 (target true types) columns: A, T, W, R, and L (these are the columns we are trying to predict). The structure of the ANC is shown in Figure A2.1 with objects, such as, Training and Query Datasheets, Encoders, Neufuzzy and Query Results Datasheet.

In order to operate the implementation of ANC we need to follow the following steps:

1. Opening the Template (Template used: nfz.tem), Click File, Open, Templates, nfz.tem
2. Importing the Data (Training and Query Data)

Training Data: Double click on the dataview "Training Data". With the cursor on cell A1, right hand click and select import data. Select /Mywork/FPFC-Train.csv and choose yes to "Would you like to import the first row of data as column names".

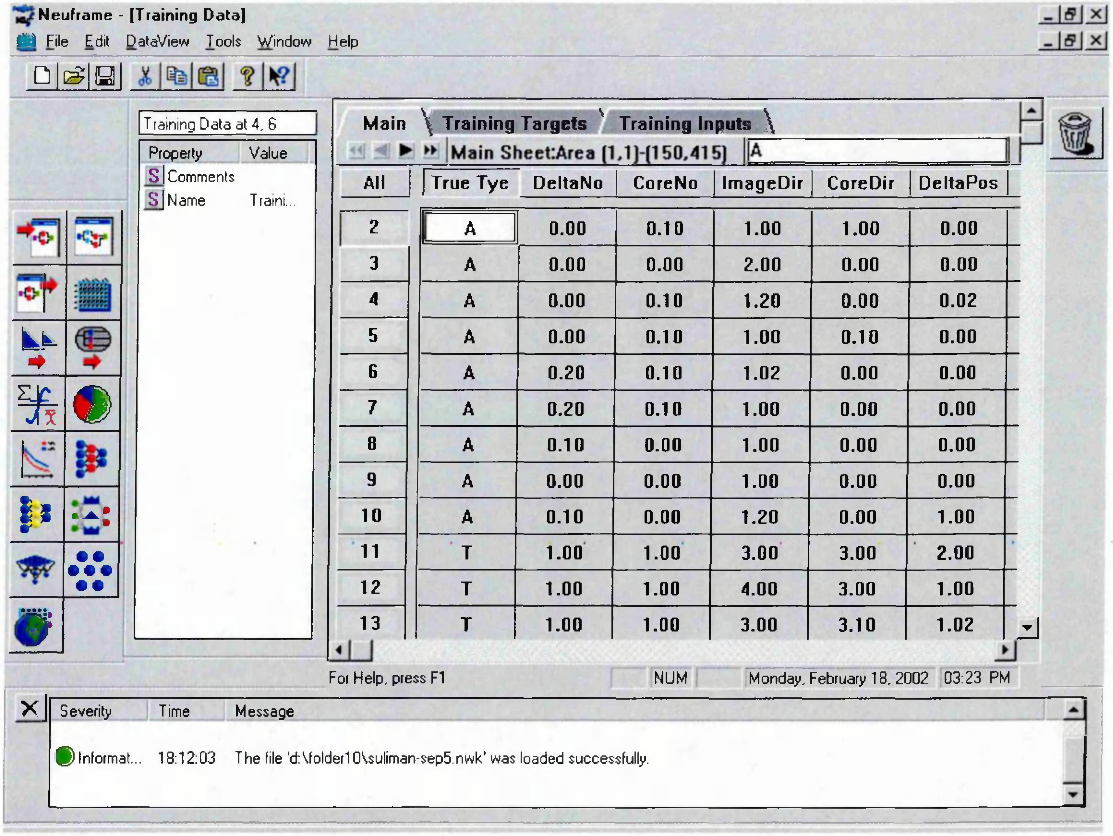
Highlight the data in the first column, right hand click and select Map Area to Dataview. Select Training Targets.

Query Data: Double click on the dataview "Query Data". With the cursor on cell A1, right hand click and select import data. Select /Mywork/FPFC-Test.csv and choose yes to "Would you like to import the first row of data as column names".

Highlight the data in the first column, right hand click and select Map Area to Dataview. Select Known Results.

Highlight the data in the first column, right hand click and select Map Area to Dataview. Select Training Targets.

Highlight the remainder of the data, right hand click and select Map Area to Dataview. Select Query Inputs. Close the datasheet.


Neuframe - [Training Data]
File Edit DataView Tools Window Help
Main
Training Targets
Training Inputs
Main Sheet:Area (1,1)-(150,415)
A

Training Data at 4, 6

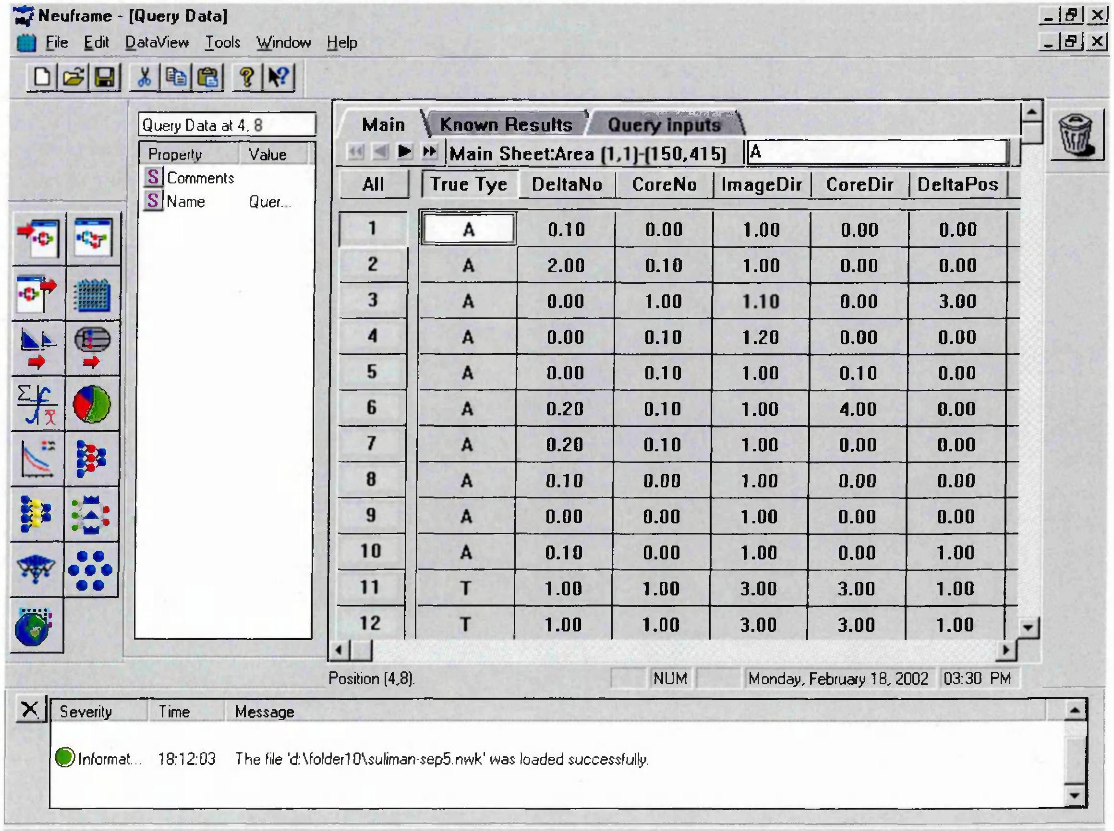
Property	Value
S Comments	
S Name	Traini...

All	True Tye	DeltaNo	CoreNo	ImageDir	CoreDir	DeltaPos
2	A	0.00	0.10	1.00	1.00	0.00
3	A	0.00	0.00	2.00	0.00	0.00
4	A	0.00	0.10	1.20	0.00	0.02
5	A	0.00	0.10	1.00	0.10	0.00
6	A	0.20	0.10	1.02	0.00	0.00
7	A	0.20	0.10	1.00	0.00	0.00
8	A	0.10	0.00	1.00	0.00	0.00
9	A	0.00	0.00	1.00	0.00	0.00
10	A	0.10	0.00	1.20	0.00	1.00
11	T	1.00	1.00	3.00	3.00	2.00
12	T	1.00	1.00	4.00	3.00	1.00
13	T	1.00	1.00	3.00	3.10	1.02

For Help, press F1 NUM Monday, February 18, 2002 03:23 PM

Severity	Time	Message
Informat...	18:12:03	The file 'd:\folder10\suliman-sep5.nwk' was loaded successfully.

Figure A2.2. Training data has been loaded into the network.


Neuframe - [Query Data]
File Edit DataView Tools Window Help
Main
Known Results
Query Inputs
Main Sheet:Area (1,1)-(150,415)
A

Query Data at 4, 8

Property	Value
S Comments	
S Name	Quer...

All	True Tye	DeltaNo	CoreNo	ImageDir	CoreDir	DeltaPos
1	A	0.10	0.00	1.00	0.00	0.00
2	A	2.00	0.10	1.00	0.00	0.00
3	A	0.00	1.00	1.10	0.00	3.00
4	A	0.00	0.10	1.20	0.00	0.00
5	A	0.00	0.10	1.00	0.10	0.00
6	A	0.20	0.10	1.00	4.00	0.00
7	A	0.20	0.10	1.00	0.00	0.00
8	A	0.10	0.00	1.00	0.00	0.00
9	A	0.00	0.00	1.00	0.00	0.00
10	A	0.10	0.00	1.00	0.00	1.00
11	T	1.00	1.00	3.00	3.00	1.00
12	T	1.00	1.00	3.00	3.00	1.00

Position (4,8). NUM Monday, February 18, 2002 03:30 PM

Severity	Time	Message
Informat...	18:12:03	The file 'd:\folder10\suliman-sep5.nwk' was loaded successfully.

Figure A2.3. Testing data has been loaded into the network.

This step is to load data from training and testing feature tables, which have been performed from FFE in Chapter 4 will be loaded to training datasheet and query datasheet respectively. That is by importing the data, the result of this process can be viewed as shown in Figures *A2.2* and A2.*3* receptively.

3. Training the Network

We will notice that we have only encoded the targets. Neufuzzy does not require numerical values to be scaled, however the target values, although numerical, are in fact categories. We therefore want to encode them as such.

Note: If a single categorical output has been presented as numbers then NeuFuzzy will not recognize it correctly and will assume it represents a single continuous variable output. There will then be only one output variable and only two membership sets, which will probably not make sense regarding the original problem

Now press Play ▶

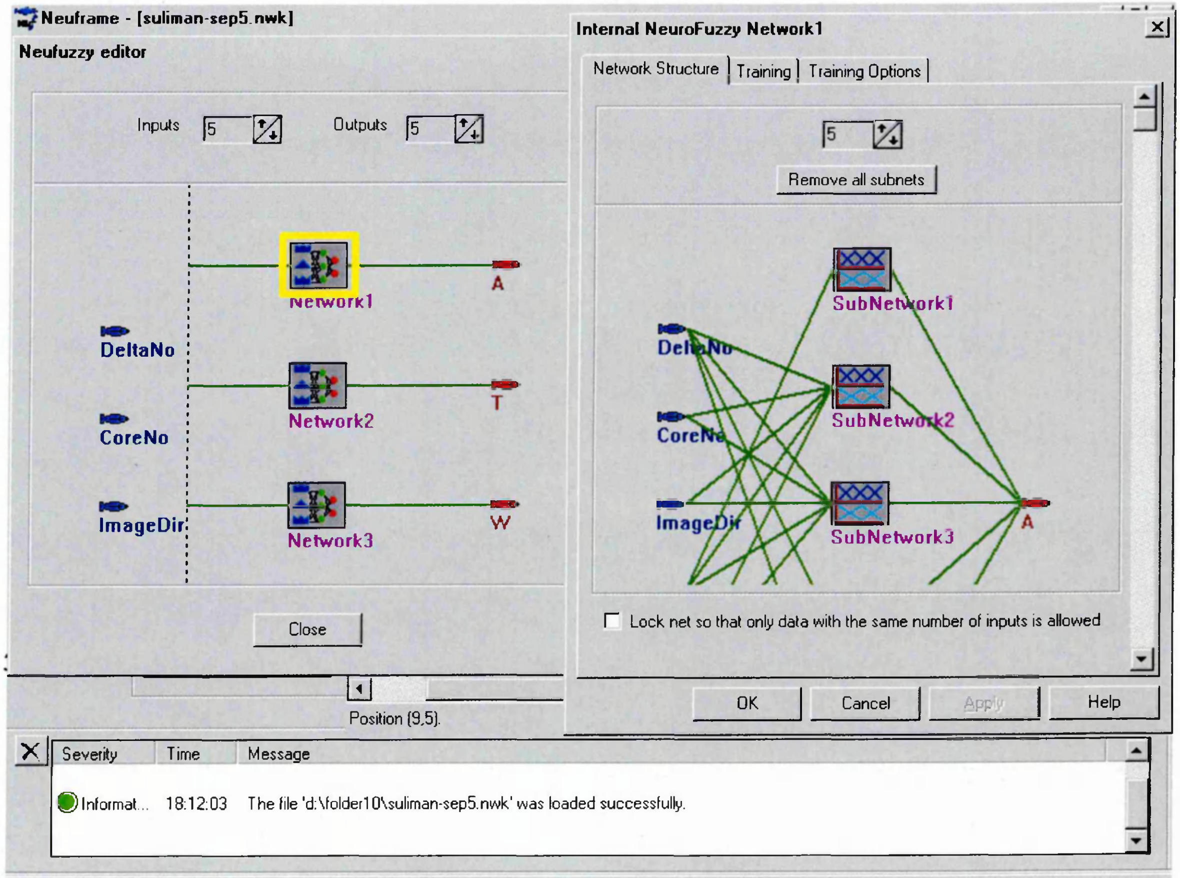
Once training has stopped, this is represented by a moving blue icon in the left-hand side of the worksheet, double click on the Neufuzzy object. We will notice there are five networks each with an output. The five outputs are in order, A, T, W, R and L. Double clicks on the first network.

4. Naming the Inputs and the Outputs: We need on each input in the network to turn and rename as per the list of variables of the five features.

Naming the Inputs: Right hand click on each input in turn and rename as per the list of variables above.

Naming the Outputs: Similarly right hand click on the outputs and rename to Classes A, T, W, R, and L. We need to do this for each of the networks.

Note that Fuzzy-Neural, unlike the MLP for example, will only use the variables, which have a bearing on the result. The generation of the five-inputs (the features) and the five-outputs (the five classes A, T, W, R, L) are shown in the Figure *A2.4* as seen in this Figure, the middle sub-networks can be expanded into another *5* networks.



Inputs 5 Outputs 5

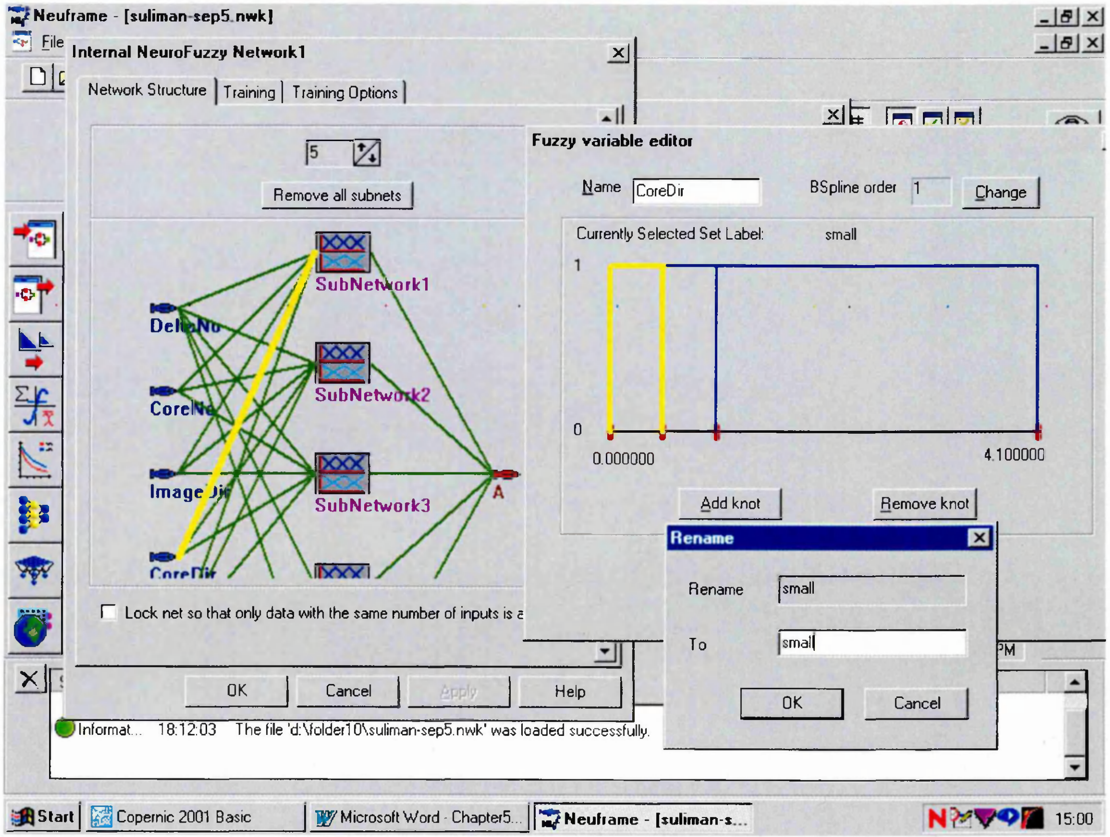
5

□ Lock net so that only data with the same number of inputs is allowed

```
Informat...  18:12:03  The file 'd:\folder10\suliman-sep5.nwk' was loaded successfully.
```

Figure A2.4. Right first-network of 5 inputs and 5 outputs, left the internal-network

specific fuzzy set. For example, we know that the CoreDir data varies from 1 to 4; we could therefore name them as (0...1 = small), (1...2 = medium), (2...3= large), and (3...4 = very-large).



Name: CoreDir

BSpline order: 1

Currently Selected Set Label: small

Rename: small

To: small

Informat... 18:12:03 The file 'd:\folder10\suliman-sep5.nwk' was loaded successfully.

Figure A2.5. The process naming the membership sets

In each network, you will need to click on the green connections. They will turn thick yellow, right hand click and select edit. Double click on each line in the networks and rename appropriately as shown in a Figure *A2.5* this is the equivalence to the process of defuzzification of the input data in the structure of our fuzzy-neural network.

6. Naming the output set: Right hand click on the output, select edit.

Note that there are only two sets for each one of the five classes; e.g. for class A either it is A or it is not.

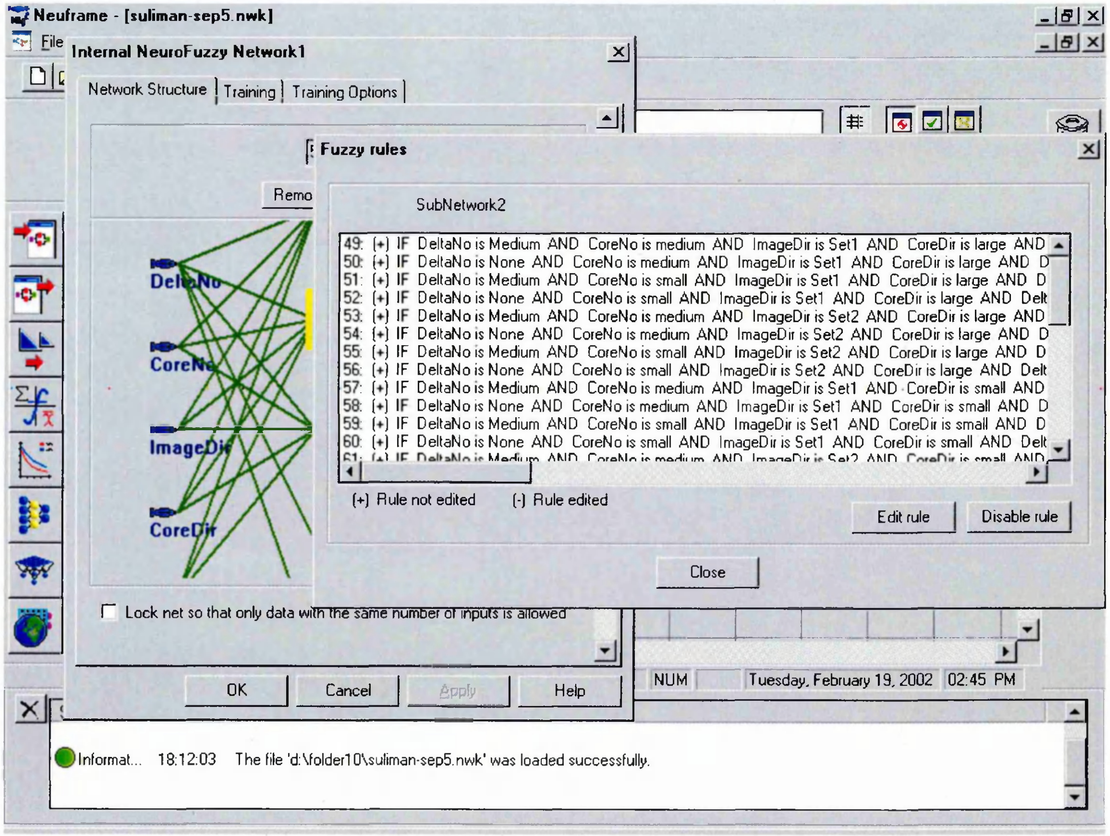
Note: The membership graphs should be interpreted as probabilities, and the two lines as being 'in the set' (the line which increases towards the right hand side) and 'not in the set' (the line which begins high and gets lower). This is true even when the desired output is numeric not categorical, but the defuzzification addresses this and ends up with a sensible value. Again, ranges along the bottom axis may or may not correspond to data ranges.

Once we have named the relevant inputs, outputs, the memberships sets and the outputs sets, we can view the (IF...THEN....AND....) rules generated by each sub-network.

Right hand click on the network and choose Query Display.

Double click on a sub-network to view these.

For using ANC will notice that we have only encoded the targets. Fuzzy-neural does not require numerical values to be scaled, however the target values, although numerical, are in fact categories. Therefore, encode them as such, and if a single categorical output has been presented as numbers then fuzzy-neural will not recognize it correctly and will assume it represents a single continuous variable output. There will then be only one output variable and only two membership sets, which will probably not make sense. Rules can be edited at any point if prior knowledge is known. Examples are shown in Figure A2.6.



SubNetwork2

```
49: (+) IF DeltaNo is Medium AND CoreNo is medium AND ImageDir is Set1 AND CoreDir is large AND
50: (+) IF DeltaNo is None AND CoreNo is medium AND ImageDir is Set1 AND CoreDir is large AND D
51: (+) IF DeltaNo is Medium AND CoreNo is small AND ImageDir is Set1 AND CoreDir is large AND D
52: (+) IF DeltaNo is None AND CoreNo is small AND ImageDir is Set1 AND CoreDir is large AND Delt
53: (+) IF DeltaNo is Medium AND CoreNo is medium AND ImageDir is Set2 AND CoreDir is large AND
54: (+) IF DeltaNo is None AND CoreNo is medium AND ImageDir is Set2 AND CoreDir is large AND D
55: (+) IF DeltaNo is Medium AND CoreNo is small AND ImageDir is Set2 AND CoreDir is large AND D
56: (+) IF DeltaNo is None AND CoreNo is small AND ImageDir is Set2 AND CoreDir is large AND Delt
57: (+) IF DeltaNo is Medium AND CoreNo is medium AND ImageDir is Set1 AND CoreDir is small AND
58: (+) IF DeltaNo is None AND CoreNo is medium AND ImageDir is Set1 AND CoreDir is small AND D
59: (+) IF DeltaNo is Medium AND CoreNo is small AND ImageDir is Set1 AND CoreDir is small AND D
60: (+) IF DeltaNo is None AND CoreNo is small AND ImageDir is Set1 AND CoreDir is small AND Delt
61: (+) IF DeltaNo is Medium AND CoreNo is medium AND ImageDir is Set2 AND CoreDir is small AND
```

(+) Rule not edited (-) Rule edited

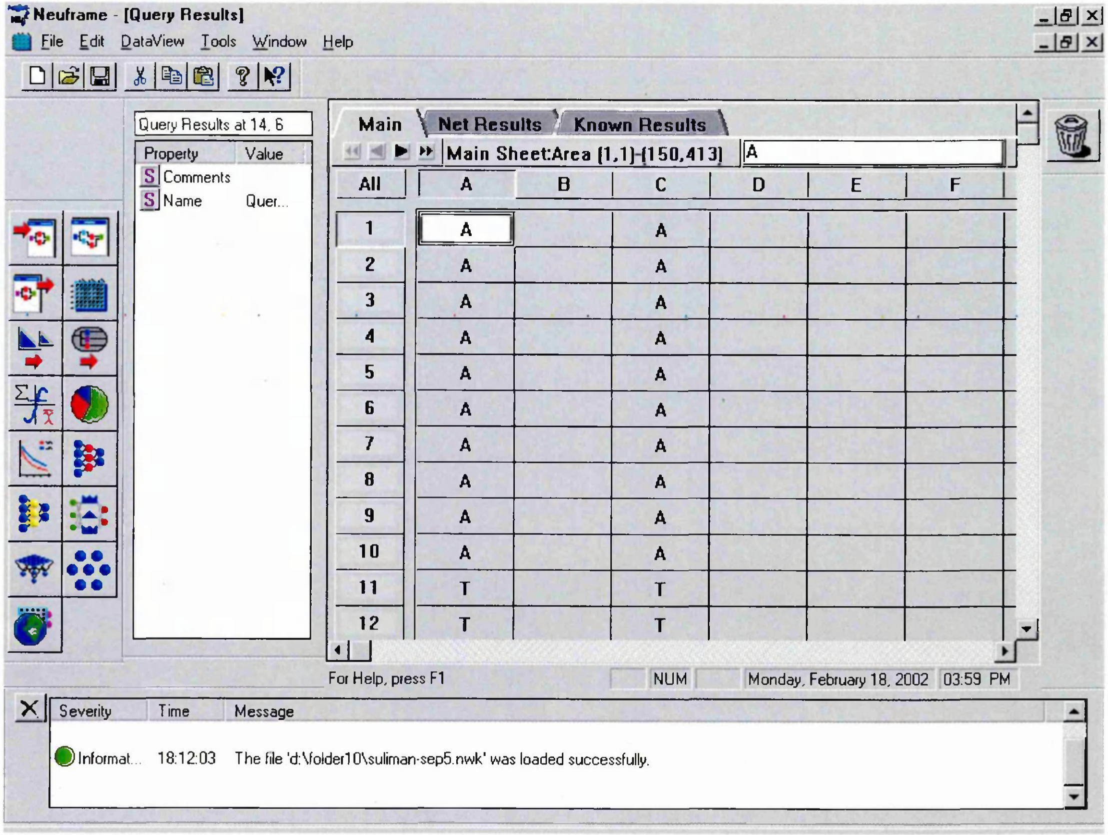
☐ Lock net so that only data with the same number of inputs is allowed

Tuesday, February 19, 2002 02:45 PM

Informat... 18:12:03 The file 'd:\folder10\suliman-sep5.nwk' was loaded successfully.

Figure A2.6. Fuzzy rules data generated by each sub-network

7. Querying the Network: To query the network we need to check the target and put the mode into training. Using automatic execution to run the network. Once the training has stopped, double click on the fuzzy-neural object. We then go back to the target encoder and change the mode to query and then run the network again.


Neuframe - [Query Results]
File Edit DataView Tools Window Help
Main
Net Results
Known Results
Query Results at 14. 6
For Help, press F1
NUM
Monday, February 18, 2002 03:59 PM

Property	Value
S Comments	
S Name	Quer...

Main Sheet:Area [1,1]-[150,413] A

All	A	B	C	D	E	F
1	A		A			
2	A		A			
3	A		A			
4	A		A			
5	A		A			
6	A		A			
7	A		A			
8	A		A			
9	A		A			
10	A		A			
11	T		T			
12	T		T			

Severity	Time	Message
Informat...	18:12:03	The file 'd:\folder10\suliman-sep5.nwk' was loaded successfully.

Figure: A2.7. The matching of the known-results with the net-results from the query results data-sheet.

8. Reviewing the Results: Finally, to view the results, click on the query results to view the matching of the classification as shown in Finger A2.7. This is from within datasheet Query Results, comparing the results from the net with the known results.

Net Results Known Results

Close the datasheet.

9. Saving the Worksheet: To save the worksheet, select File, Save As the

   Yourfile.nwk

Appendix 3

Copies of Some Significant Publications

The copies of some Papers, which have been published in Journals and Proceedings of Conferences are attached in this Appendix, starting from the latest.

1. Fingerprint Classification using Fuzzy Neural Networks; accepted for 2002 Fuzz-IEEE Hawaii, USA 12-17 May 2002.

2. Automatic Fingerprint-based Biometric Recognition using Fuzzy/Neural Networks Techniques; paper accepted and due to printed By the Editor of the international journal of Computers and Industrial Engineering, 2002.

3. Automatic Fingerprint Classification System Using Fuzzy Neural, Proceeding of the 2001 International Conference on AI, Vol. 1, PP. 395-401, Las Vegas, Nevada, USA, June, 2001, ISBN: 1-892512-78-5.

4. Suliman Mohamed, Sayed Horbaty, and Aboul-Ella Hassanien, An Image Metamorphosis Algorithm Based on Navier Spline Interpolation, Egyptian Computer Society Journal vol. 22, no. 1, PP. 9-12, Jan. 2000.

5. Automatic Fingerprint Identification System Using Fuzzy Neural, Proceeding of the 2000 International Conference on AI, Vol. 2, PP. 859-865, Las Vegas, Nevada, USA, June, 2000, ISBN: 1-892512-57-2.

6. Fingerprint Recognition System Using Fuzzy Neural Techniques, Proceeding of 8th International Conference in AI Applications, PP. 295-305-45, Egypt, Cairo, 2000.

7. Image Pattern Recognition Using Fuzzy/Self-Organising Network, the Proceeding of the 6th UK Workshop on Fuzzy Systems, PP. 115-120, Brunel University, Uxbridge, UK, September, 1999.

Automatic Fingerprint Classification System Using Fuzzy Neural Techniques

Suliman M Mohamed and Henry O Nyongesa

School of Computing and Management Sciences
Sheffield Hallam University
Sheffield S1 1WB, U.K.

bstract–The paper presents a fingerprint classification system nd its performance in an identification system. The lassification scheme is based on fingerprint feature extraction, hich involves encoding the singular points (Core and Delta) ogether with their relative positions and directions obtained rom a binarised fingerprint image. Image analysis is carried in our stages, namely, segmentation, directional image estimation, ingular-point extraction and feature encoding. A fuzzy-neural etwork classifier is used to implement the classification of input eature codes according to the well known Henry system. Fingerprint images from NIST-4 database were tested and. 98.5% classification accuracy was obtained for the five class-problem.

I. INTRODUCTION

Fingerprint identification and verification are one of the most significant and reliable identification methods. It is virtually impossible that two people have the same fingerprint. (probability 1 in 1.9E15) [1]. In fingerprint identification and verification applications world-wide, a large volume of fingerprints are collected and stored for a wide range of applications, including forensics, civilian, commercial and law-enforcement applications. Automatic identification of humans based on fingerprints requires the input fingerprint to be matched with a large number of fingerprints in a database (for example, the FBI database contains approximately 70 million fingerprints). To reduce the search time and computational complexity, it is desirable to classify the database into accurate and consistent classes so that input fingerprint is matched only with a subset of the fingerprints in the database. The nature of each application will determine the degree of accuracy required. For example, a criminal investigation case may require higher degree match than access control case systems.

Many automatic fingerprint classification methods, such as methods introduced in [3], [5] and [9]-[12], rely on point patterns in fingerprints, which form ridge endings and bifurcation unique to each person. Traditionally, activities to solve a pattern recognition task are twofold. First, a set of features has to be found describing the object(s) being classified. Second, after a set of features has been found, a classification mechanism is chosen and optimised. These two steps are highly interdependent, since the choice of features influences the conditions under which a classifier operates, and vice versa. With the advent of neural networks however, more and more problems are solved by simply feeding large amounts of 'raw data' (e.g. images, sound signals, stock market index ranges) to a neural network. This approach, however, is not feasible in fingerprint classification, which are highly susceptible to noise and elastic distortions. Therefore, it is desirable to extract features from the images that are invariant to such distortions. During training the classification network learns the association and significance of features. An attempt has been made previously to study fuzzy logic and artificial neural network techniques in fingerprint identification [2]. It was shown that a trade-off exists between the trainability of simple networks and its understandability: the larger the network, the easier to train and the most reliable training results can obtain. The conclusion was that fuzzy-neural networks could be useful as adaptive filters in fingerprint classification tasks, but that great care has to be taken in choosing the network architecture and training algorithm. In this paper an implementation of a fuzzy-neural network for fingerprint classification system is presented. The rest of this paper is organised as follows. In section II the proposed feature extraction algorithm is reported. Section III presents a brief discussion of fingerprint classification using a fuzzy-neural network (FNN) learning approach. Section IV presents the results of FNN classification after training and testing. Finally section V draws some conclusions from the study.

II. FINGERPRINT FEATURE EXTRACTION (FFE)

The central problem in designing a fingerprint classification system is to determine what features, should be used and how categories are defined based on these features. There are, mainly two types of features that are useful for fingerprint recognition system: (i) local ridge and valley details (minutiae) which have different characteristics for different fingerprints, and (ii) global pattern configurations, which form special patterns of ridges and valleys in the central region of the fingerprint. The first type of features carries for the information about the individuality of fingerprints and the second type of features carry information about the fingerprint class. Therefore, for fingerprint classification, the features derived from the global pattern configurations should be used. These features should be invariant to the translation and rotation of the input fingerprint images. Generally, global fingerprint features can be derived from the orientation field and the global ridge shape.

The orientation field of a fingerprint consists of the ridge rientation tendency in local neighbourhoods and forms an bstraction of the local ridge structures. It has been shown at the orientation field is highly structured and can be ughly approximated by the core and delta models [13], hich are known as singular points details. Therefore, ingular points details (see Figure 3) and their relationships an be used to derive fingerprint categories. On other hand, obal ridge shape and directional field also provides portant clues about the global pattern configuration of the igerprint image.

Many different algorithms for singular points extraction are iown from the literature. Examples of these algorithms are, liding neural networks [3], local energy of directional image rocessing [4], ratio of the sine of the fingerprint image in /o adjacent regions [1], and singular point indexing [5]. However, these algorithms give somewhat unsatisfactory results. in particular the rate of accuracy is very low in most cases. Post-processing steps are necessary to interpret the outputs of the algorithms and to make the final decisions. resulting in missed and false singular points. In this paper. we show that a singular points verification stage based on re-examining the gray-scale profile in a detected singular-points spatial neighbourhood of the image can improve the classification performance. Additionally, we show that a feature encoding stage which relies on the images estimated directional field can improve the classification performance.

A. Segmentation of Fingerprint Image

Segmentation of an image is used to pre-process appropriately; in order to remove noise from an image sample and it is often a key step in interpreting the image. Image segmentation is a process in which regions or features sharing similar characteristics are identified and grouped together. Image segmentation may use statistical classification. thresholding, edge detection. region detection, or any combination of these techniques [9,11,12]. The output of the segmentation step is usually a set of classified elements. such as regions or boundaries. Thresholding is the simplest way to perform segmentation. and it is used extensively in many image-processing applications. It is based on the notion that regions corresponding to different object types can be classified by using a range function applied to the intensity values of image pixels. The assumption is that different object types will have distinct frequency distributions and can be discriminated on the basis of the mean and standard deviation of each distribution. Thus. given a two-dimensional image $I(x,y)$. we can define a simple threshold rule to classify different object types. Threshold of gray-level images to black and white is based on a two-stage process: General Threshold (GT) of the whole image in the first stage. and Regional Average Thresholding (RAT) in the second stage. A hypothetical frequency distribution $f(I)$ of intensity values is used such that. low intensity values correspond to black while high intensity values correspond to white.

- *General Thresholding (GT)*

In the GT scheme, the process of binarising of the gray level image to a black and white image is carried out by looking at each pixel on the fingerprint image and deciding whether it should be converted into black (0) or white (255), i.e. converted to 0 and 1 values. The decision is made by comparing each numeric pixel of gray-level image with a fixed number called a threshold level to make the decision. If the pixel is less than the threshold level, the pixel value is set to zero: otherwise it is set to 255. The thresholding scheme can be expressed as follows in equation (1).

$$P(i,j) = \begin{cases} 255 & \text{if } I(i,j) > T \\ 0 & \text{if } I(i,j) <= T \end{cases} \qquad (1)$$

Where $I(i,j)$ indicate the original image. $P(i,j)$ indicates the output binary image. T is the threshold level. and ($i= 0,....N, j = 0,.....M$) represent the image size.

- *Regional Average Thresholding*

Applying GT to an image may cause some feature lose? This is because: the average gray level is not, usually, the same in different parts of the original image (e.g. background and foreground). This is particularly the case in fingerprint images, which are directly effected by different kinds of the skin affections or noise. Regional Average Thresholding (RAT) is a threshold scheme for fingerprint images. which has been proposed to overcome the problem of the GT. Thus. the original image may be partitioned into small regions. such as. 32x32 or 16x16 pixel windows. Thresholding is then carried out within each region, using the gray-level average of each window. The average Grey levels is calculated as shown in equation (2).

$$T = 1/N^2 \sum_{i=0}^{N} \sum_{j=0}^{N} I(i,j) \qquad (2)$$

In this paper a 16x16 pixel window scans the image starting from the left most corner of the image. An average threshold level is calculated within each window and moved to next window. The process continues until the bottom right corner of the image. Since the average threshold levels are calculated regionally, more features are preserved in comparison with GT. This stage also eliminates the fields that contain no information, such as, the edges of the fingerprint images.

- *Directional Image Estimation*

In order to extract singular point we have proposed to calculate the directional field of an image. The directional field describes the local orientations of the ridge and valley structures in a fingerprint. In this paper, the directional field of a fingerprint image is computed in four sub-directions, as shown in Figure 2. Firstly, the image is partitioned into small blocks (we chose 5x5 blocks).

Numeric gradients are computed for each sub-direction from the pixel intensities (equation 3). The dominant direction is then given by the sub-direction with the smallest numerical value. By sliding the 5x5 mask in Figure 2 over the threshold image P, we calculate the minimum sum of differences (sod) for the central pixel, c (Figure 2). Each block is then represented by the gradient value of the dominant sub-direction:

$$V(c,c) = \min_{d}\left(\sum\left|P(c,c) - P(i,j)\right|\right) \qquad (3)$$

where, *P(i,j)* are binary values, in a given direction, d.

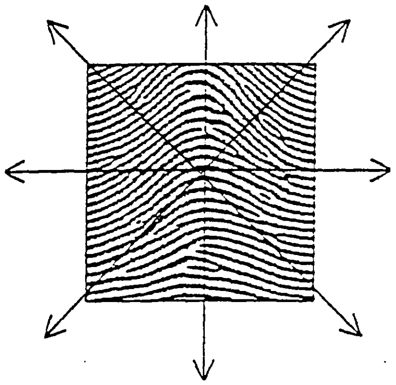


Fig. 1. Direction computation in 4 main directions

4		3		2
1		c		1
2		3		4

Fig. 2. A 5X5 direction mask with its geometric orientations

The directional field of an image, V creates a M/q x N/q reduced-size image, which decreases the dimensionality of the input features and hence the complexity of the feature extraction algorithm. The logic behind the working of the directional field method is that a peak in the histogram of a directional image in a region indicates that there exists a clear ridge, because a ridgeline results in points of the same direction in the region. That is, if a clear ridge exists in a region, it expressly means it is foreground, which gives rise to a peak in the histogram. The limitation of this method is that in perfect uniform region $P(c,c)=P(i_m,j_m)$, for m varying in any direction, thus equation (3) become undefined. However, the directional criterion is very good for low contrast and noisy images, besides giving good results for modest quality (clarity in ridges) of fingerprint images.

C. Singular Points Extraction

Singular points, namely the Delta and the Core, are manifest as discontinuities in the directional image. They are clearly visible in the fingerprint image in Figure 3. Delta point lies on a ridge at or in front of and nearest to the centre of the divergence of the type lines. A Core point is the approximate centre of the finger impression. Using the reduced-size directional image, we determine the candidate singular points, including their relative orientations and directions in the fingerprint image as follows:

A pixel, c (Figure 2) is a Delta point if:

$$16 \le \sum_{c} P(x,y) \le 20 \qquad (4)$$

A pixel, c (Figure2) is a Core point if:

$$\sum_{c} P(x,y) \ge 21 \qquad (5)$$

Otherwise, the point, c is undefined: where the pixel intensities *P(x,y)* are summed around the pixel, c.

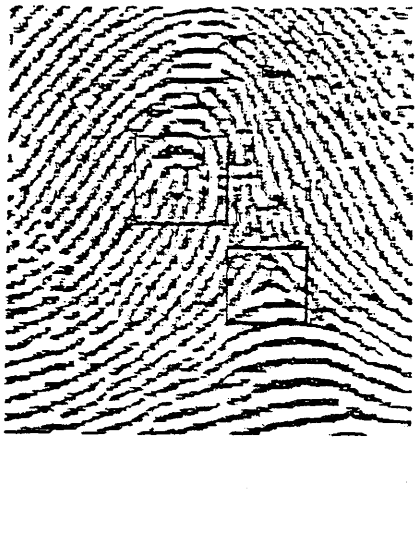


Fig. 3. Singular points on fingerprint

D. Feature Encoder

A feature encoder is applied for representing the vector of features extracted from fingerprints. This is a list of singular points with accompanying attribute values. The information we are interested includes:

1. Number of deltas, *DeltaNo;*
2. Number of cores, *CoreNo*;
3. Global directional field orientation, *ImageDir*
4. Core direction, *CoreDir;*
5. Relative Core-Delta position *DeltaPos.*

Table 1 shows an example of typical feature vectors for different fingerprint classes, namely, Arch, Tended arch, Whorl, Right-loop, Left-loop (see Figure 5). Due to noise and errors in segmentation and feature extraction algorithms, it is generally the case that the actual feature vectors deviate significantly from the canonical case. For this reason classifiers that can cope with such deviations are desirable. In this paper, it has been proposed to use a fuzzy-neural classifier.

TABLE 1: TYPICAL FEATURES FOR DIFFERENT CLASSES

Type	DeltaNo	CoreNo	ImageDir	CoreDir	DeltaPos
A	0	0	1	0	0
T	1	1	3	3	1
W	2	2	3	2	4
R	1	1	4	4	2
L	1	1	2	2	3

III. FINGERPRINT CLASSIFICATION USING FUZZY-NEURAL CLASSIFIER

Fuzzy-neural hybrid systems combine the advantages of fuzzy systems, which deal with explicit knowledge that can be explained and understood, and neural networks, which deal with implicit knowledge that can be acquired by learning [6]-[8]. In the fuzzy-neural network, the neural network part is primarily used for learning and classification and retrieval. The neural network part automatically generates fuzzy logic rules and membership functions during the training period. In addition even after training, the neural networks keeps updating the membership functions and fuzzy logic rules as it learns more and more from its input signals. Fuzzy logic, on the other hand, is used to infer and provide a crisp or defuzzified output where ambiguities exist in the input fuzzy parameters. In order to train the classifier, two data sets of feature codes were prepared. The first data set is used for training the network and the second for testing. Fuzzification of the operation of the classifier by generating membership functions around the typical values of feature codes, easily explained with linguistic terms. As an example we know that the CoreNo varies from 0 to 2. We could therefore form "fuzzy" CoreNo as none (0..1), small (1..2), and large (>= 2).

The overall network was constructed through an automatic network construction process, a feature of NeuFrame™ software [14]. Typical rules from the network are illustrated below:

1. IF DeltaNo is small AND CoreNo is small AND ImageDir is small AND CoreDir is small AND DeltaPos is right THEN L is equal (0.91) OR L is equal (0.09)
2. IF DeltaNo is medium AND CoreNo is small AND ImageDir is small AND CoreDir is AND DeltaPos is right THEN L is equal (0.91) OR L is equal (0.09)
3. IF DeltaNo is small AND CoreNo is medium AND ImageDir is small AND CoreDir is small AND DeltaPos is right THEN L is equal (0.91) OR L is equal (0.09).
4. IF DeltaNo is medium AND CoreNo is medium AND ImageDir is small AND CoreDir is small AND DeltaPos is right THEN L is equal (0.91) OR L is equal (0.09).

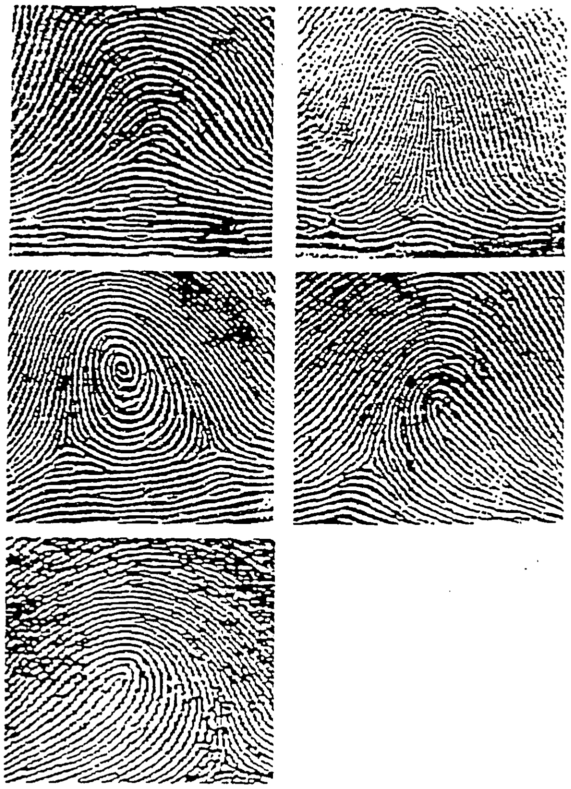


Fig. 4: Fingerprint classes - top left - Arch; top right Tended arch ; middle left - Whorl; middle Right-loop; bottom left - Left loop

IV. EXPERIMENTAL RESULTS

Results of the performance of the classifier were obtained y querying the fuzzy-neural classifier using the test set, and omparing known class labels against the classifier outputs. ie classifier was trained and tested on 4,000 images in the IST-4 database for the five-class problem. We note, erefore, that the overall network consists of five networks, ach corresponding to the output classes, A, T, W, R, L. The sults, presented in Table II, were obtained after passing feature encoded vectors of the FFE algorithm. The result iows that the classification accuracy varies widely across the different classes. Initial investigation has indicated that this may be due to the generalisation characteristic of neural networks, which causes mis-classification among fingerprints with similar features. It is suggested that this can be overcome using a different feature extraction scheme. Alternatively, the occurrence of mis-classification can be studied further and the confusion probabilities used in resolving the final output classes.

TABLE II
EXPERMENTAL RESULTS FROM ANC

Class Type	Accuracy
A	86.5
T	96.0
W	98.5
R	96.2
L	85.0

V. CONCLUSIONS

The aim of this paper has been to present an implementation of a fingerprint classification problem using fuzzy-neural networks. Fingerprint classification provides an important mechanism for automatic fingerprint recognition systems. We have proposed a simple and flexible fingerprint classification algorithm, which classifies input fingerprints into five categories according to the number of the core and delta (singular points), and their relative (x,y) positions in an image. The classifier was tested on 4,000 images in the NIST-4 database. For the five-class problem, classification accuracy as high of 98.5% is achievable. By incorporating a reject option, the classification accuracy can be increased further. The feature extraction algorithm demonstrates how, from directional fields of an image, accurate detection of the singular points and the orientations of those points can be obtained. While it is true that this method was not tested for all possible features of fingerprints, it has been shown to be effectively in identifying singular-point in all cases tested.

VI. REFERENCES

[1] L. Hong, S. Prabhakar, A. K. Jain, and S. Pankanti, "Filterbank-Based Fingerprint Matching," IEEE Transactions of Image Processing, Vol. 9:5, PP. 846-859, 2000.

[2] S. Mohamed, H. O. Nyongesa, and J Siddiqi, "Automatic Fingerprint Identification System Using Fuzzy Neural Techniques," Proceedings of the International Conference on Artificial Intelligence, Volume 2 PP. 859-865, CSREA Press, Las Vegas, June 2000.

[3] G. Drets and H. Liljenstrom, "Fingerprint Sub-Classification, A Neural Network Approach," Intelligence Biometric Techniques In Fingerprint and Face Recognition, PP. 109-134, 1999.

[4] P. Perona, "Orientation diffusions," IEEE Transactions on Image Processing, Vol. 7, PP. 457-467, 1998.

[5] M. Kawagoe and A. Tojo, "Fingerprint Pattern Classification," Pattern Recognition, Vol. 17, PP. 295-303, 1984.

[6] D. Zhang, M. Kamel and M. Elmasry, "Fuzzy Clustering Neural Network Using Fuzzy Competitive Learning," World Congress on Neural Networks, International, 1993.

[7] L. Zadeh, "Fuzzy Sets," Information and Control, Vol. 8, PP. 338-352, 1965.

[8] D. Patterson, Artificial Neural Network Theory and Application, 1996.B.

[9] Mehtre and B. Chatterjee, "Segmentation of Fingerprint Images Using Composite Method," Pattern Recognition, Vol. 22:4 PP. 381-385, 1989.

[10] Fitz and R. Green, "Fingerprint Classification Using A Hexagonal Fast Fourier Transform," Pattern Recognition, 29, No. 10, PP. 1587-1597 (1996).

[11] S. Michael, M. Chong and T. Han Ngee, "Geometeric Framework for Fingerprint Image Classification," Pattern Recognition, Vol. 30:9, PP. 1475-1488, 1999.

[12] B. Mehtre, N. Murthy, S. Kapoor and B. Chatterjee. "Segmentation of Fingerprint Images Using the Directional Image," Pattern Recognition, Vol. 20: 4, PP. 429-345, 1999.

[13] D. Monro and B. Sherlock, A Model of Interpreting Fingerprint Topology, Pattern Recognition, Vol. 26:7, PP. 1047-1055, 1993.

[14] NeuFrame: www.neusciences.com

Automatic Fingerprint-based Biometric Recognition Using Fuzzy/Neural Networks Techniques

Suliman M Mohamed[1] and Henry O Nyongesa

Computing Research Center
School of Computer and Management Sciences
Sheffield Hallam University
Sheffield S1 1WB, U.K.

Abstract: Fingerprint technology, is one of the most mature biometrics technologies. Biometrics identification deals with identification of individuals based on their biological or behavioral characteristics (so-called positive personal identification). The use of fingerprint for identification has been employed in law enforcement for about century, as it's one of the most reliable personal verification methods. However, manual fingerprint recognition is so tedious, time-consuming, and expensive that it is incapable of meeting today's increasing performance requirements. In this regard we have investigated the limitations of current approaches and strategies, and attempt to overcome these limitations by using the strengths of the computational intelligence. In particular, we introduce a flexible algorithm of feature extraction and investigates of fingerprint image enhancement and classification by using Fuzzy Self-Organizing Map Learning (FSOM), with the learning and the self-organizing features of artificial neural networks and the ability to process fuzzy data using fuzzy membership.

Keywords: *fingerprint recognition, biometrics. fingerprint enhancement, feature extraction, fingerprint classification/matching, fuzzy neural learning.*

[1] Corresponding author email: S.Mohamed@shu.ac.uk. Tel: 0044-1142253169, Fax: 1142253161

1. Introduction

Associating an identity with an individual is called personal identification. Biometrics identification deals with identification of individuals based on their biological or behavioral characteristics. Therefore, fingerprint technology, is one of the most, mature biometrics technologies. As information becomes the key to wealth in the 21st century, biometrics security will play a central role in providing a high level of security to existing and future products. Fingerprints are graphical flow-like ridges present on human fingers. The fingerprint image is made of foreground ridges which are separated by background valleys. ridge flow direction forms different patterns like arches, loops, wholes, and also gives rise to various minutiae like ridge endings. ridge bifurcation's. cores. deltas etc. Both foreground and background consist of a similar set of minutiae. the tiny patterns used for fingerprint classification, [2]. Each individual has a unique fingerprint and the uniqueness of a fingerprint, is exclusively determined by the local ridge characteristics and their relationship. These local ridge characteristics are not evenly distributed. Most of them depend heavily on the impression conditions and quality of fingerprints and rarely observed in fingerprints. Police departments have long been interested in the improvement of fingerprint recognition methods as an important factor in making more effective the administration of criminal justice. security control business. Although. fingerprint verification systems are usually associated with criminal identification, and police work it has now become more popular in civilian applications. Such as access control systems. high-security areas in prominent organizations. financial security, verification of firearm purchasers, and driver license applicants. These useful applications have been conceived due to the suspects when they are arrested. In civilian applications finger images may be captured by placing a finger on a scanner or by electronically scanning inked impressions on paper. Fingerprint recognition technology is already a key player in the information security device, and the Police departments have long been interested in the improvement of AFIS methods as an important factor in making more effective the administration of criminal justice, security and record business transactions. The adaptability of fingerprint recognition hardware to the computer keyboard and mouse make it a viable alternative to the workstations, password. New optical components like molded lens and single chip fingerprint imaging devices have lowered hardware costs from the thousands dollars per unit to under fifty, [7,8].

2.1.2 Face Recognition

Face is one of the most acceptable biometric because it is one of the most common methods of identification which humans use in their visual interactions. In addition, the method of acquiring face images is non-intrusive. Two primary approaches to the identification based on face recognition are the following [5]:

(i) Transformation approach: the universe of the face image domain is represented using a set of orthonormal basis vectors.

(ii) Attribute-based approach: facial attributes like nose. eyes, mouse. etc. are extracted from the face image and the invariance of geometric properties among the face landmark features is used for recognizing features.

2.1.3 Iris Recognition:

The iris is composed of elastic connective tissue. the trabecular meshwork. It consists of pectinate ligaments adhering into a tangled mesh revealing striations, ciliary processes. crypts. rings. furrows. a corona. sometimes freckles. vasculature.

fingerprint favorable characteristics such as unchangeability and uniqueness in an individual's lifetime. Inherently, using current technology of fingerprint identification is much more reliable than the kinds of popular biometric identification methods based on signature, face, iris, ear, hand geometry, and speech. With increasingly large volumes of fingerprints being collected and stored, there is an urgent need to develop automatic fingerprint recognition systems to improve the efficiency and reliability of personal identification. Usually, fingerprint recognition is performed manually by professional fingerprint experts. However, manual fingerprint recognition is so tedious. time-consuming, and expensive that is does not meet the performance requirements of the new applications. Various approaches for preprocessing and fingerprint recognition have been investigated for the purpose of automatic fingerprint recognition. These can fall into either one of the following categories: structural, statistical, syntactic, geometric. mathematical, hybrid approaches and artificial neural networks. [3.4,6.9.11.12]. In section 2 the discussion and overview of biometrics, biometric terms, biometric selection criteria, and biometric system steps are briefed. In section 3 the discussion of the main issues of automatic fingerprint image processing. including (fingerprint acquisition, enhancement, classification, and matching), here the research briefed on most of these issues but a concentration on enhancement. fingerprint feature extraction and classification has been investigated and implemented. Section 4 reports the overview of fuzzy neural technique. Finally section 5 draw some conclusions.

2. Overview of Biometrics

Biometric technology offers the possibility of replacing PINs, passwords, keys and other conventional terms. Eventually, with iris or fingerprint recognition systems (identifications), a user ID or card may no longer be necessary, figure: 1 depict samples of some biometric items. Theoretically, any human physiological or behavioral characteristic can be used to make a personal identification as long as it satisfies the following requirements:

(i) universality, which means that every person should have the characteristic, (ii) uniqueness, which indicates that no two persons should be the same in terms of the characteristic, (iii) permanence, which means the characteristic should be invariant with time, and (iv) collectability, which indicates that the characteristic can be measured quantitatively. In practice, there are some other important requirements like (i) Performance, which refers to the achievable identification accuracy, the resource requirements to achieve acceptable identification accuracy, and the working environmental factors that effect the identification accuracy. (ii) Acceptability, which indicates to what extent people are willing to accept the biometric system. and (iii) circumvention, which refers to how easy it is to fool the system by fraudulent techniques, [4].

2.1 Biometric Terms

Terms relating to specific biometrics technologies and techniques are summarized as below with a surface introducing given to each one of these terms:

2.1.1 Automatic Fingerprint Identification System (AFIS):

AFIS is some times known as Automatic Fingerprint Recognition System (AFRS). The AFIS highly specialized biometric system that compares a single finger image with a database of finger images. AFIS is predominantly used for law enforcement, but is also being put to use in civil applications, [12]. For law enforcement, finger images are collected from crime scenes. known as latent. or are taken from criminal and other features. During the first year of life a blanket of chromatophore cells usually changes the color of the iris, but the available clinical evidence indicates that the trabecular pattern itself is stable throughout the lifespan. [Clinical reference, Adler, 1965]. The protected internal organ, which can be imaged adequately at distance up to about a meter, reveals about a number of independent degrees-of-freedom of textural variation across individuals. The iris has in excess of 250 characteristics that are unique to each person, which is more than ten times the number of identifiers carried by a fingerprint. [British Telecommunications Engineering Journal, 1997]. Being an internal organ of the eye, the iris is immune (unlike fingerprint) to environmental influences, except for it's papillary response to light. The computer reads your iris in much the same way as it scans a bar code. Which is what this part of your eye effectively becomes after it has been entered into the non-contacted system (it takes few seconds for your iris image to be captured, by simply looking into a standard camera). Consider the applications for future. Iris scanning is going to open more doors than you thought possible. We predict that iris and fingerprint technologies will be used for everything from the identification of CMC, to car security and fast tracking through passport control at ports and airports.

2.1.4 Voice Recognition

Voice is a characteristic of an individual, however, its not expected to be sufficiently unique to permit identification of an individual from a large database of identities, [5]. Moreover, a voice signal available for authentication is typically degraded in quality by the microphone, communication channel, and digitizer characteristics. Before extracting features, the amplitude of the input signal may be normalized and decomposed into several hand-pass frequency channels. The features extracted from each band may be either time-domain or frequency domain features. Voice capture is unobtrusive and voiceprint is an acceptable biometric in most societies.

2.1.5 Signature Verification

Signature has long been accepted as a legitimate means of authentication. The way a person signs his/her name is known to be a characteristic of that individual. Although, signatures require contact and effort with the writing instrument, they seem to be acceptable in many government. legal, and commercial transactions as a method of personal authentication. Signatures are a behavioral biometric, evolve over a period of time and are influenced by physical and emotional conditions of the signatories. There are two approaches to signature recognition verification namely static and dynamic, [5]. In static signature verification. only shape (geometric) features of the signature are used for authentication and identity. Typically, the signature impressions are normalized to a known size and decomposed into simple components (strokes). The shapes and relationships of strokes are used as features. In dynamic signature verification, not only the shape features are used for authenticating the signature but also the dynamic feature like acceleration, velocity, and trajectory profiles of the signature are employed.

2.1.6 Other Biometrics

A number of other biometrics-based technologies are available and being developed in the educational and commercial research laboratories worldwide. Currently, these are include the following biometrics:

(i) *DNA:* DNA (DeoxyriboNucleic Acid) is the one-dimensional ultimate unique code for one's individuality. except for the fact that identical twins have the identical DNA pattern. It is, however, currently used mostly in the context of forensic applications of identification, [5]. Three issues limit the utility of this biometric for other applications:

- Contamination and sensitivity: it is easy to steal a piece of DNA from an unsuspecting subject to be subsequently abused for an ulterior purpose;
- Automatic real-time identification issue: the present technology for genetic matching is not geared for online unobtrusive identifications.
- Privacy issues: information about susceptibilities of a person to certain diseases could be gained from the DNA pattern and there is a concern that the unintended abuse of genetic code information may result in discrimination in e.g., hiring practices.

(ii) *Body Odor*: It is known that each object exudes an odor that is characteristic of its chemical composition and could be used for distinguishing various objects. The feature vector consists of the signature comprising of the normalized measurements from each sensor. After each act of sensing, the sensors need to be initialized by a flux of clean air. Body odor serves several functions including communication. attracting mates, assertion of territorial rights, and protection from a predator, [5]. A component of the odor emitted by a human (or any animal) body is distinctive to a particular individual. It is nor clear if the invariance in a body odor could be detected despite that deodorant smells varying chemical odor-based identity authentication systems exist.

(iii) *Ear Shape:* In the ear the structure of the cartilaginous tissue of the pinna are distinctive. The features of an ear are not expected to be unique to each individual, [5]. The ear recognition approaches are based on matching vectors of distances of salient points, on the pinna from a landmark location on the ear. No commercial systems are available yet and authentication of individual identity based on ear recognition is still a research topic.

(iv) *Facial Thermogram:* The facial thermogram is an image taken with an infrared camera that shows the heat patterns of the face. These images are unique, and combined with highly sophisticated pattern matching algorithms that check for relative temperature differences across the face, mean that this technique is unaffected by age, health, or even the temperature of the body. With 19,000 'data points' it's extremely accurate and will distinguish identical twins even in the dark, [12]. The development of this technology continues in the direction of improving cost effectiveness in order to increase its applicability to wider range identification and verification applications. The facial thermogram offers the promise of providing accurate, effective and highly secure identification technology once technology costs are reduced, and if there is a flexible method of acquisition.

(v) *Gait:* Gait is the peculiar way one walks and is a complex spatio-temporal behavioral biometric. Gait is not supposed to be unique to each individual, but is sufficiency characteristic to allow identity authentication. Gait is a behavioral biometric and may not stay invariant especially over a large period of time. Due to large fluctuations of body weight, major shift in the body weight (e.g. waddling gait during pregnancy, major injuries involving joints or brain (e.g., cerebella lesions in Parkinson disease, or due to inebriety), [5]). Humans are quite adept at recognizing a person at a distance from his gait.

Although, the characteristic gait of a human walk has been well researched in biometrics community to detect abnormalities in lower extremity joints, the use of gait for identification purposes is very recent.

(vi) *Hand geometry/Hand recognition:* Similar uses to voice recognition but unsuitable for arthritis sufferers and affected by changes in the weight.

(vii) *Keystroke Dynamics:* the keystroke is the behavioral biometric for some individuals, one may expect to observe large variations from typical typing patterns. Keystroke dynamic features are based on time duration between the keystrokes. Some variants of identity authentication use features based on inter-key delays as well as dwell times, how long a person holds down a key. Typical matching approaches use neural networks architecture to associate identity with the keystroke dynamics features, [11]. Some commercial systems already appearing in the market.

(vii) Retina: Eyes-Retina is the retina scan, offers the promise of extremely accurate identification but is invasive and currently requires the head to stabilized so that a light can be directed to the back of the retina, [11].

(viii) Vein tree (Hand vein): A related technology using near infrared imaging is the facial thremogram technology, [5]. Identification based on hand veins is an infrared image of the back of a clenched human fist. The structure of the vasculature could be used for identification.

2.2 Biometrics Selection

Which biometric will eventually gain preeminence in the information security identification and authentication/verification will depend on a number of factors including technological improvements that provide for improved performance, cost, univerisabilty, and consumer acceptance. Mapping the application requirements with all the attributes of the specific biometric will determine optimal biometric technology.

No single biometric technology, or single security device for that matter, can deliver guaranteed 100 percent security. Most customers that purchase information security identification and verification devices want the best possible security solution that is affordable, easy to implement, yet is unilaterally accepted by the intended user population. The answer to these needs will most likely be combination of security techniques to include multiple biometrics, which may effect the need of easy to implement. Growth in the biometrics research and industry has led to an ever-increasing number of vendor products available to the prospective buyer. As inmost new and emerging technology, there are many small vendors, many large product claims, providing an atmosphere that is difficult for customers to make strong differentiation amongst products. *Table: 2.1* gave a comparison of biometrics technologies in terms of some important factors.

The public may perceive biometric information as a confidential piece of information, much like a social security number. The use of biometric information as a means of access control and/or non-repudiation purposes is likely to enter the privacy issues debate. Even if access to that information requires a lot of effort, it can be treated as a public relations problem. To choose the right approach to biometric authentication and identification, implementers must understand the application, the user base and the characteristics of the biometric device it self. One also must consider the conditions under which it will be used and how fallback authentication methods, such as passwords or tokens, will be instituted when biometrics are not available. There are some factors to consider before choosing a biometric system.

These can fall as, multiple levels of requirements in industry on recognition systems. It can be broken down into two classes, verification and identification. The obvious choice when combining a token such as smartcard/ID with biometrics is using verification. A verification (matching the live fingerprint with the template stored on the token (ID/Smartcard), known as one-to-one) typically requires a more simplistic algorithm and has better performance. While identification (matching the given biometric with the database of stored information, known as one-to-many). Both verification and identification require more math computations then the present day biometrics technologies microprocessor can perform in a reasonable period.

2.3 Biometric System

A biometric which is characterized by a behavioral trait that is learnt and acquired over time rather than a physiological characteristic, i.e. technique for keystroke dynamics, signature verification, and speaker verification. Contrast with physical/physiological biometric.

2.3.1 An automatic biometric system capable of the following issues:

1. Capturing (acquisition) a biometric sample from an end user;
2. Extracting biometric data(feature) from that sample;
3. Comparing the biometric data with that contained in one or more reference templates.
4. Deciding how well they match; and
5. Indicating whether or not an identification or verification of identity has been achieved.

The submission of a biometric sample to a biometric system for identification or verification (authentication) is principle issue. The biometric system may allow more than one attempt to identify or verify. Biometric data are the extracted information taken from the biometric sample and used either to build a reference template or to compare against a previously created reference template. *Figure: 2* depict the general steps of a biometric identification system. The mechanism of the biometric system is the biometric engine & Certification. Biometric engine is the software element of the biometric system, which processes biometric data during the stages of enrolment and capture, extraction, comparison and matching.

The biometric certification is the process of testing a biometric system to ensure that it meets certain performance criteria. Systems that meet the testing criteria are said to have passed and are certified by the testing organization, [13]. Therefore, a number of advantages of Biometric System can include:

- Unique recognition of a claimant, the system is able to reliably establish people's identity. Eventually tokens and password no longer be required.
- Ease of use, it merely requires some user and applications depend.
- Secure, especially using iris or fingerprint, which are their biological feature, can not altered easily.
- Robust, the system generates a unique code for every individual.

2.3.2 Feature Encoder

For the purpose of simplicity we discussed the feature encoding of fingerprint technology. The problems are to develop image processing algorithms that are efficient and not computationally intensive for pre-processing of fingerprint images. Although, it may need many steps to overcoming this problem like to experiment with several image pre-processing algorithms and compare the effectiveness of them. One of the overriding requirements is that the image processing technique applied to fingerprint images does not create new features or lose existing features, [5]. In this regard we analysis the fingerprint image and use a simple technique to extract the most prominent features (minutiae), namely ridges endings & bufircations. Once extracted, feature properties are allowed to deviate by a user definable amount. Additionally number of features to be matched for successful identification may also be defined, depending on the nature of the application area.

The encoder uses a Grey-scale fingerprint images to extracts basic features (minutiae). The minutiae information of the fingerprint image are extracted by using the position minutiae in x, y co-ordinate, minutiae type, and minutiae direction. Each feature encoder has the information of position, type, and direction. There are only two types of minutiae, ridge endings and bifurcations. The position of each feature is expressed in (x,y axis) top left-hand corner being the origin (0,0).

2.3.3 Users difficulties

The most common user difficulties deal with alignment in image capture area. If image is translated or rotated excessively, the verification algorithm has difficulty in matching the users live image to the stored template. The majority of these problems are corrected the first few times a person uses the scanner. It may be important for the system to provide users with feedback on translation or rotation.

Some populations have difficulty using biometric devices. People with light ridge definition in their fingers may have difficulty using fingerprint-identification systems. Those who work with abrasive substances, construction workers or even people who handle large volumes of paper can have their ridges worn down. These also are substantial physical differences based on age, gender and ethnicity.

Users with excessively dry, wet or dirty hands have experienced problems with finger and palm recognition systems. People wearing gloves generally can't use these systems; however, the ultrasound-based systems have had limited success detecting prints through thin latex gloves.

Many systems may be turned to do less strict checking at the expense of weakening the security provided by the system. Administrators have to balance false acceptances versus false rejects, the possibility of fraud versus user convenience, [9]. Individual thresholds may be used to lower the threshold for only clients who have poor biometric characteristics. The threshold should still be within acceptance limits, as to not allow a "anyone can pass with my card" scenario.

3. Automatic Fingerprint Recognition

As a result of many studies, automatic fingerprint recognition systems are in great demand. Although, a significant progress has been made in designing automatic fingerprint recognition systems, over the past 25 years, a number of limitations in achieving the desired faster matching of the fingerprint image. An automatic fingerprint recognition system is concerned with some or all of the following issues:

- Fingerprint Acquisition
- Fingerprint Enhancement
- Fingerprint Classification
- Fingerprint Matching: (verification/identification)

In this paper we go brief on most of these issues but a concentration on enhancement, fingerprint feature extraction and classification has been investigated and implemented.

3.1 Fingerprint Acquisition

Fingerprint Acquisition is how to acquire fingerprint images and how to present them in a proper format. A number of methods are used to acquire fingerprints. Among them, the inked impression method remains the most popular. It has been essentially a standard technique for fingerprint acquisition for more than 100 years, (Battley H., 1937). The first step in capturing an inked impression of a fingerprint is to place a few dabs of ink on a slab then rolling it out smoothly with a roller until the slab is covered with a thin. Then the finger is rolled from one side of the nail to the other side over the inked slab, which inks the ridge patterns on top of the finger completely. After that the finger is rolled on a piece of paper so that the inked impression of the ridge pattern of the finger appears on the paper. Obviously, this method is time-consuming and unsuitable for an on-line fingerprint verification system. The second method is a more efficient and reliable optical data generation system. It consists of a prism and a uniform light beam that transforms the three-dimensional data into two-dimensional data, which can be photographed. The optical method of fingerprint data generation is not perfect either because the contrast and focus of the image obtained are sometimes poor. However, the method is clean, fast, and most of the problem can be overcome by good preprocessing techniques such as grayscale-to-binary conversion. Innovations in optical devices have been made recently as mid-1990s, an optical sensor was housed in a box about 6x3x6 inches. The third method is the ink-less fingerprint scanners are now available which are capable of directly acquiring fingerprints in digital form. This method eliminates the intermediate digitization process of inked fingerprint impressions and makes it possible to build an on-line system. The fourth method so-called solid-state sensors have appeared on the market recently. These are microchips containing a surface that images the fingerprint via one of the several technologies. including electrical measurements and temperature sensitive sensors. One of the most important factors that will decide when fingerprint verification will be commercially successful in the large-volume personal verification market are low cost and compact size.

3.2 Fingerprint Enhancement

Fingerprint Enhancement is to clear the quality of fingerprint images. In practice. due to variations in impression conditions, ridge configuration, skin conditions (aberrant formations of epidermal ridges. postnatal marks. occupational marks). acquisition devices, and non-cooperative attitude of subjects, a significant percentage of acquired fingerprint images is poor of quality. The ridge structures in poor-quality fingerprint images are not always well defined hence; they can not be correctly detected. In order to ensure that the performance of the minutiae extraction algorithm will be robust with respect to the quality of input fingerprint images, an enhancement that can improve the clarity of the ridge structures is necessary, (D. C. Douglas, 1993). A fingerprint is often able to correctly identify the minutiae by using various visual clues such as local ridge orientation. ridge continuity. ridge tendency, as long as the ridge are not corrupted completely. It is possible to develop an enhancement algorithm that exploits these visual clues to improve the clarity of ridge structures in corrupted fingerprint images.

Since the objective of a fingerprint enhancement algorithm is to improve the clarity of ridge structures of input fingerprint images to facilitate the extraction of ridge and minutiae. a fingerprint enhancement algorithm should not result in any spurious ridge structures. This is very important because spurious ridge structure may change the individuality of input fingerprints.

3.3 Fingerprint Classification

Fingerprint Classification is to assign a given fingerprint to one of the pre-specified categories according to its geometric appearance. A fingerprint classification algorithm presented in this paper is classified fingerprints into five main classes as arch, tended arch, left loop, right loop, and whorl.

- Arch: is a pattern of convex ridges with a peak in the middle. Arch pattern in which the ridges enter on one side and flow out the opposite side. Figure (4-a).
- Tented Arch: consists of a global structure of convex ridges on top. Figure (4-d) depicts an arch fingerprint.
- Whorl: had two deltas and global convex ridges. at least one ridge makes a complete circle, giving an overall circular effect within the pattern area. Figure (4-b) depicts a whorl fingerprint.
- Wright Loop: in the left right, the ridges are toward right from center of the print arrange themselves in the form of a hairpin, making a backward turn without twist, with the delta at the left of the core. Figure (4-d) depicts a left loop fingerprint.
- Left Loop: in the left loop. the are ridges toward left from centre of the print arrange themselves in the form of a hairpin, making a backward turn without twist, with the delta at the right of the core. Figure: (4-e) depicts a left loop fingerprint.

As can be seen in Figures: 4, an arch fingerprint image contains no cores or deltas, tented arches and loops contains one delta and one core, and whorls have two cores and two deltas.

The main purpose of fingerprint classification is to facilitate the management of large fingerprint databases and to speedup the process of fingerprint matching. Generally, manual fingerprint classification is performed within a specific framework such as well-known Henry system. Different frameworks use different sets of properties. However, no matter what type of framework is used, the classification is based on ridge patterns, local ridge orientations and minutiae. Therefore. if these properties can be described quantitatively and extracted automatically from a fingerprint image then fingerprint classification will become an easier task. The system has main two important stages, converting the gray level image of fingerprint to binary level and the extraction of singular points (core & delta) for implementation of classification.

3.3.1 Threshold the gray level images to black and white

Thresholding of the gray level image to black and white binary image. It involves looking at each pixel and deciding whether it should be converted into white (255) or black (0). Comparing each numeric pixel of gray level with fixed number called a threshold level makes the decision. If the pixel is less than the threshold level, the pixel value is set to zero; otherwise it is set to 255. The thresholing scheme can be expressed as follows:

$$G(i,j) = \begin{cases} 255 & \text{if } F(i,j)>T \\ 0 & \text{if } F(i,j)<=T \end{cases} \quad (1)$$

Where F(i,j) indicate the original image, G(i,j) indicates the out but binary image, T is the threshold level, (i= 0,1,...,N, j = 0,1,...,M), and N, M are the number of rows and columns in the image respectively. In our case the gray level of 512x512(NxM), and here we assume that T=50.

3.3.2. Singular point extraction

The algorithm extracts the *X, Y* positions of the singular points which the cores and deltas in a fingerprint image, and performs classification.

The singular point represent by point$_{CD}$ is used to determine the number of cores (N_c) and the number of deltas (N_d) points & positions in the fingerprint image.

Assume that a pixel on thresholding i.e. (eight-connected), then it has a value 1and otherwise 0. Let (x,y) denote a pixel at a ridge and n_1, n_2,....n_8 denote it's eight neighbours:

A pixel (x,y) is a Delta point if $(\sum n_i,\ i=1,2,..,8) = 5$ (2)

A pixel (x,y) is a Core point if $(\sum n_i,\ i=1,2...,8) > 5$ (3)

The classification diagram used in our algorithm is as follows:

1. If (N_c=0) and (N_d=0), then an arch is identified.

2. If (N_c=2) and (N_d=2), then a whorl is identified.

3. If (N_c=1) and (N_d=1), then classify the input by using the core and delta positions assessment by estimating the symmetric axis which crosses the core in its local neighborhood and compute the angle, β, between the line segment from the core to the delta and the symmetric axis:

a. If the core and delta in the same vertical (x,y) positions, or (β<5), then a tended arch is identified.

b. If the delta to the left of the core's (x,y) axis, and (β>10), then a right loop is identified.

d. If the delta to the right of the core's (x,y) axis, and (β>10), then a left loop is identified.

4. If none of the above conditions is satisfied, then repeat the processing again.

3.4 Fingerprint Matching: Verification and Identification

Fingerprint matching determines whether two fingerprints are from the same finger or not (fingerprint verification), or search given fingerprint in database of template (fingerprint identification). It is widely believed that if two fingerprints are from the same source, then their local ridge structures (minutiae details) match each other topologically. Matching can be separated into two categories: verification and identification, (Lawrence O. G., 1999). Verification is the comparison of a claimant fingerprint against an enrollee fingerprint. where the intention is that the claimant fingerprint matches the enrollee fingerprint. To prepare for verification, a person initially enrolls his her fingerprint into the verification system. A representation of that fingerprint is stored in some compressed format along with the person's name or other identity. Subsequently. each access is authenticated by the person identifying him or herself, then applying the fingerprint to the system such that, the identity can be verified. Verification is also known as. *one-to-one matching. Identification* is the traditional domain of criminal fingerprint matching. A fingerprint of unknown ownership is matched against a database of known fingerprints to associate a crime with an identity. Identification is also known as. *one-to-many matching.* A number of different types of local ridge descriptions have been identified. The two most prominent structures are ridge endings and ridge bifurcation's which are usually called minutiae. Fig. 4 shows examples of ridge endings and ridge bifurcations.

4. Fuzzy Neural Approach

One of the characteristics of artificial neural networks (ANNs) is that they can classify inputs. This is useful if plasticity is maintained to be, that is, the ANNs can continuously classify and also update classifications. We have also seen the stability of ANNs and how robust when inputs become less defined (i.e., fuzzy inputs). In addition, we have seen that fuzzy systems deal with current fuzzy information and are capable of providing crisp outputs. Fuzzy logic is one of the key technologies for representing human knowledge in the brain and for constructing adaptive systems. However, in fuzzy systems there are no learning and. even vaguely, the input-output relationships the fuzzy rules must been known a priori, neural networks and fuzzy systems each have their own limitations. When one designs with neural networks alone, the network is a black box that needs to be defined, (Zhang D. M. and Elmasry M. I., 1993). This is a highly compute-intensive process. One must develop a good sense, after extensive experimentation and practice, of the complexity of the network and the learning algorithm to be used and of the degree of accuracy acceptable by the application. On the other hand fuzzy systems, require a thorough understanding of the fuzzy variables and membership functions, of the input-output relationships as well as the good judgment to select the fuzzy rules that contribute the most to the solution of the application. A large number of rules. and many may not contribute significantly to the problem. Hence, good judgment is needed to eliminate unnecessary rules. As the tide of using neural network and fuzzy logic grows up a number of studies have shown that although. neural networks are powerful in machine learning, associative memory, and parallel processing but they fails to do well in some symbol processing and indefinite reasoning. On the contrary, the fuzzy logic systems are powerful in indefinite reasoning and symbol processing but they fail to do well in associative memory, (Adeli H. and Hung S. L., 1995). So it is supposed to use the synthetic method of these two techniques which can be a complement to each other, so as to use the fusion of these two methods in fingerprint image processing. In the fuzzy artificial neural network the neural network part is primarily used for its learning and classification and retrieval. The neural network part automatically generates fuzzy logic rules and membership functions during the training period. In addition even after training. the neural networks keeps updating the membership functions and fuzzy logic rules as it learns more and more from its input signals. Fuzzy logic, on the other hand. used to infer and provide a crisp or defuzzified output when fuzzy parameters exist. In this regard. we attempt to combine Fuzzy Logic and Kohonen's Self-organized Map. have been proposed. Our algorithm investigated the fuzzy-neural models by integrating fuzzy membership function and the self-organizing map network learning techniques. for purpose of image clustering. In particular, we have investigated the combination of features of neural networks (with learning ability. self-organizing and high-speed parallel structure) and fuzzy systems (with ability to process fuzzy information using fuzzy membership) to form a Fuzzy Self-Organizing Map Networks Learning (FSOML), which can learn from environments.

5. Conclusions

One of the major problems in the automatic fingerprint recognition is the quality of the original print. If the quality is not of an acceptable standard. automatic fingerprint identification becomes extremely difficult. However. pre-processing and automatic enhancements will help more. Fingerprint classification provides an important mechanism for automatic fingerprint recognition systems. We have approached a simple and flexible fingerprint classification algorithm which classifies input fingerprints into five categories according to the number of the core and delta (singular points) and their relative x,y position. Extracted features by this algorithm are been input to our FSOM network for training and testing which give us good experimental results. A FSOM paradigm adopting a principle of learn according to how well it wins is proposed, unlike the SOM where only one neuron will win and learn at each competition, every neuron in the FSOML to a certain degree wins, depending on it is distance to the input pattern.

Acknowledgements: We would like to acknowledge our grateful appreciation to the university of Sheffield Hallam University, moreover the authors want to thank the Sudanese Administration of Human Resources Development for their financial support for the project of this research.

References

1. Adeli H. and Hung S. L. (1995). *Machine Learning, Neural Networks, G. Algorithms and Fuzzy Systems*. New York: Academic Press.

2. Alessandro F., Zsolt M., & Kovacs V. (1999). Fingerprint minutiae extraction from skeletonized binary images. *The Journal of the Pattern Recognition Society,* 1(32), 877-889.

3. Ammar H. H. & Miao Z. (1996). In: Performance of parallel algorithms for fingerprint image comparison system. Proceeding of the Parallel Proceeding Symposium, (PP. 410-413), Lasvegas.

4. Anil J., Lin H. & Ruud B. (1997). On line fingerprint verification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 19(4), 302-313.

5. Anil J., Ruud B., and Sharath P. (1999). *Biometrics Personal Identification in Networked Society*. New York: Kluwer Academic Publishers.

6. Baldi P. and Chauvin Y. (1993). NNs for Fingerprint Recogition. *Journal of Neural Computation.* 5(3), 402-418.

7. Basak J., Pal N. R. and Patel P. S. (1997). Thinning in Binary and Gray Images: A Connectionist Approach. *Journal of the Institution of ETEE.* 42(4-5), 305-313.

8. Battley H. (1937). *A New and Practical Method of Classifying and Filing Single Fingerprints and Fragmentary Impressions*. New Haven: Yale University Press.

9. Berfanger D. M. and George N. (1999). All-Digital Ring-Wedge Detector Applied to Fingerprint Recognition. *Applied Optics*. 38(2), 357-369.

10. Blue, J. L., Canfela, G. T., Grother P. J., Chellappa R. and Wilson C. L. (1994). Evaluation of Pattern Classifiers SFOR Fingerprint and OCR Applications. *Pattern Recognition.* 27(4), 485-501.

11. Chen Y., Sheng M.. & Yongbao H. E. (1992). *A Method of Pattern Recognition Based on Synthetic Technology of Fuzzy Logic & Neural Network* (pp. 20-43). Shanghai: Dep. Computer Science. Fudan

12. Cheung Y. S. & Yip W. M. (1987). A Personal Computer-Based Fingerprint Identification System. Proceeding IEEE Asian Electronic Conference (pp. 290-294). Hong Kong.

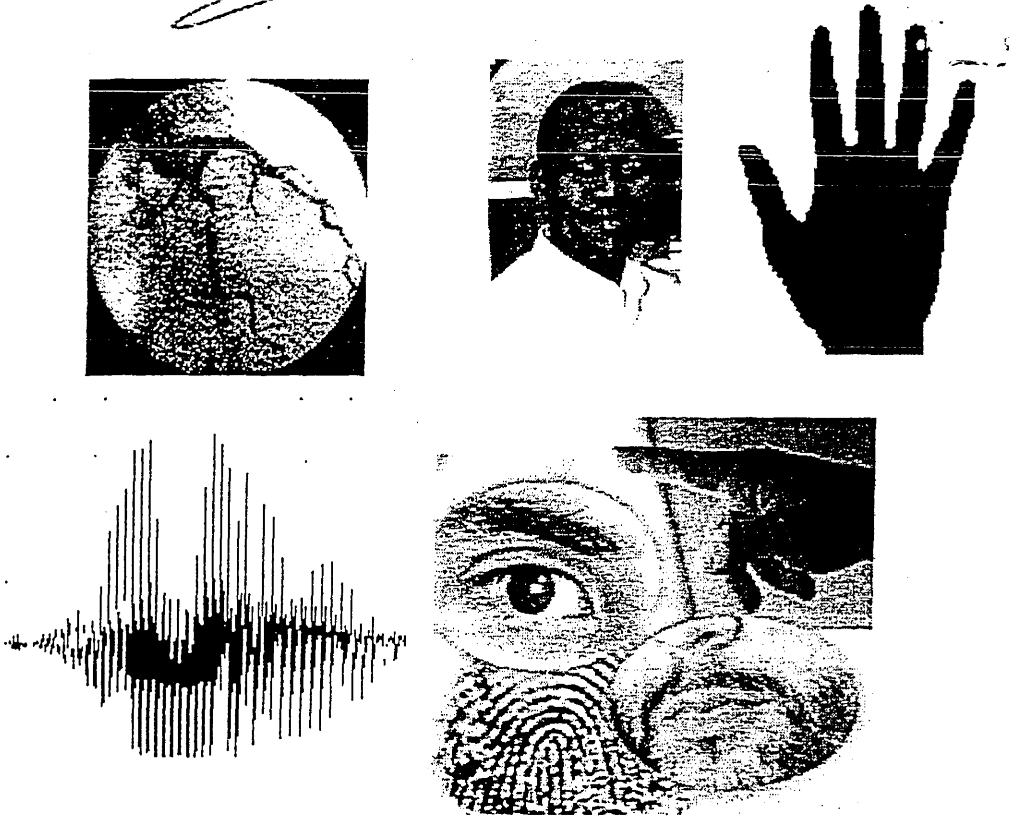


Figure: 1 Some Biometrics Samples

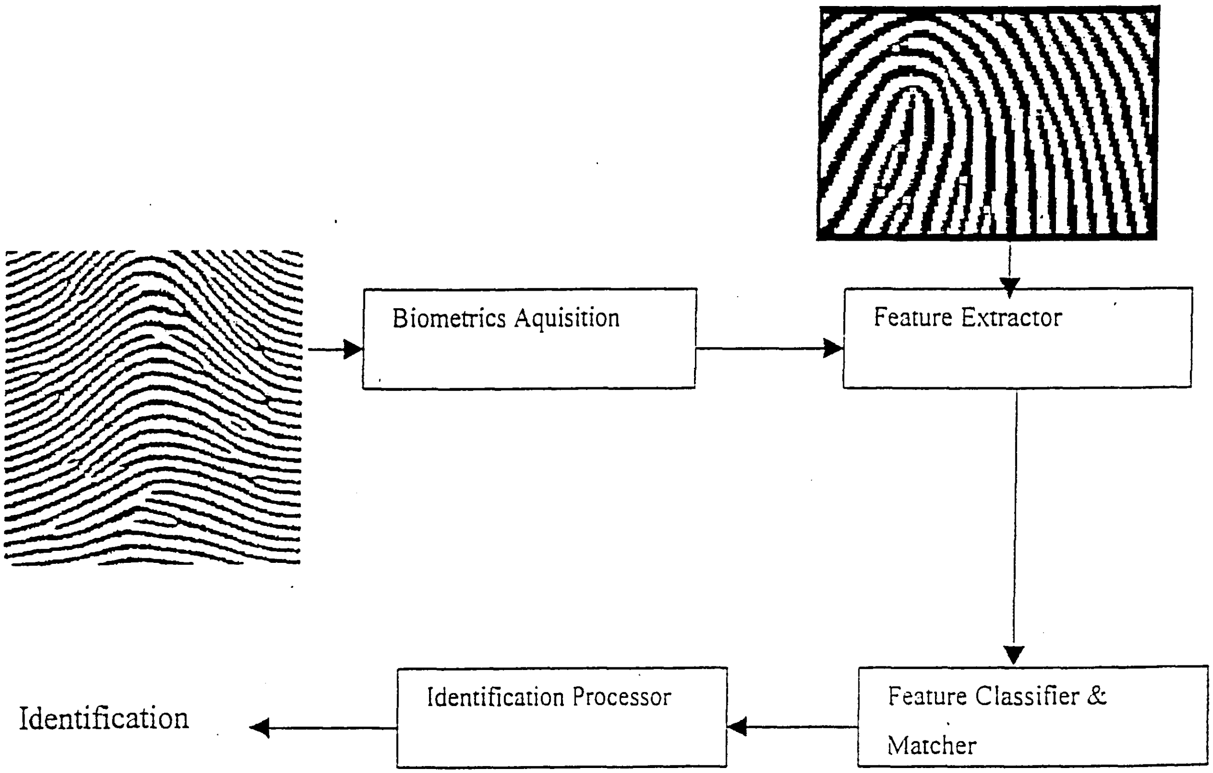


Figure: 2 General Steps of a Biometric Identification System

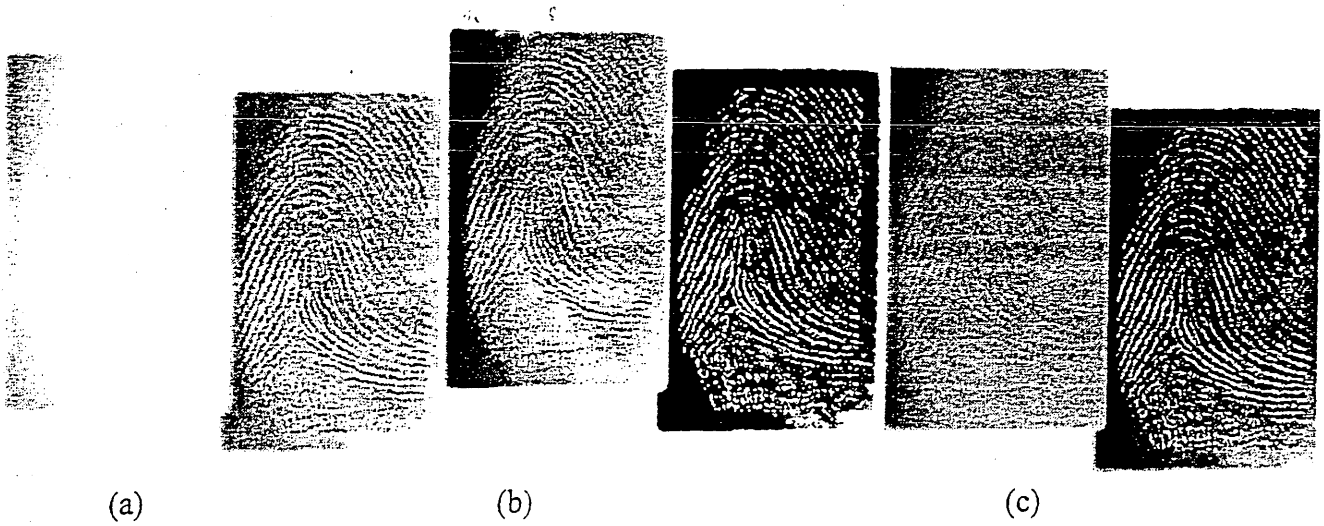


Figure: 3 Enhancement: Automatically enhancing fingerprint images without introducing artifacts is a challenging problem: (a) poor quality fingerprints, (b) and (c) result of image enhancement of fingerprint image shown in (a).

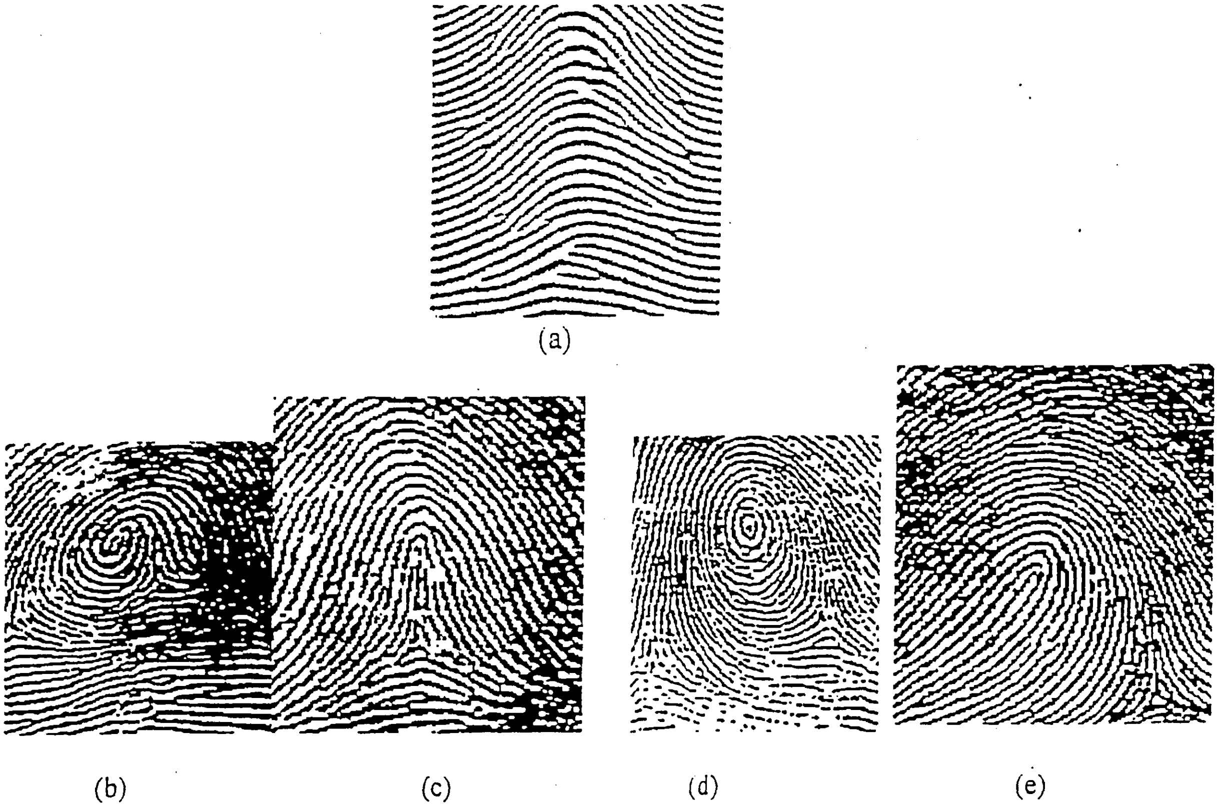


Figure:4 Five Classes are a. Arch b. Whorl c. Tended Arch d. right loop e. Left loop

Biometrics	Universality	Uniqueness	Permanence	Collectability	Performance	Acceptability	Circumvention
Face	High	Low	Medium	High	Low	High	Low
Fingerprint	Medium	High	High	Medium	High	Medium	High
Hand geometry	Medium	Medium	Medium	High	Medium	Medium	Medium
Keystrokes	Low	Low	Low	Medium	Low	Medium	Medium
Hand Vein	Medium	Medium	Medium	Medium	Low	Medium	High
Iris	High	High	High	Medium	High	Low	High
Retinal	High	High	Medium	Low	High	Low	High
Signature	Low	Low	Low	High	Low	High	Low
Voice	Medium	Low	Low	Medium	Low	High	Low
Facial Thermogra	High	High	Low	High	Medium	High	High
Odor	High	High	High	Low	Low	Medium	Low
DNA	High	Medium	High	Low	High	Low	Low
Gait	Medium	Low	Low	High	Low	High	Medium
Ear	Medium	Medium	High	Medium	Medium	High	Medium

Table: 1. The Comparison of biometrics technologies.

Automatic Fingerprint Classification System Using fuzzy Neural Techniques

Suliman M Mohamed and Henry O Nyongesa
Computing Research Center
School of Computer and Management Sciences
Sheffield Hallam University, Sheffield S1 1WB, U.K.

Abstract: *This paper reviews the fingerprint classification for its performance in the fingerprint identification systems. Fingerprint recognition can be applied to access control systems used in restricted areas, criminal recognition, forensic labs, and as a substitute for PIN numbers on cards or electronics access. The scheme is based on fingerprint classification feature extraction and testing a simple and flexible fingerprint classification algorithm. Our attempt is to allow the accuracy and speed up of the automatic fingerprint identification algorithms to improve the quality of the existing and the future systems. We used Fuzzy Self Organizing Map (FSOM) to implement the classification, in this purpose a FSOM classifier is trained to analyze. The technique extracts the singular points (Core and Delta points) in fingerprints obtained from directional threshold not minutiae neighborhoods and to decide whether they are valid or not.*

Keywords: *fingerprint, biometrics, fingerprint recognition, fingerprint* classification.

1. Introduction

In the implementation of the fingerprint applications everywhere, a large volumes of fingerprints are collected and stored everyday in wide range of applications including forensics, civilian and law-enforcement applications. An automatic recognition of human based on fingerprints requires that the input fingerprint to be matched with a large number of fingerprints in a database (FBI database contains approximately 70 million fingerprints), [10]. To reduce the search time and computational complexity, it is desirable to classify these fingerprints into an accurate and consistent manner so that the input fingerprint does required to be matched only with a subset of the fingerprints in the database.

Fingerprint identification and verification are one of the most significant and reliable biometric identification methods. It is virtually impossible that two people have the same fingerprint. (Probability 1 in 1.9E15), [11]. Therefore fingerprints are used in many applications such as criminal investigations, access control systems, national frontier and many more. The nature and security of each application require a different degree of accuracy. For example a criminal investigation case may require higher degree match than an access control case systems. Because the database of an access control is much smaller than the criminal case database. In this regard we analysis the fingerprint image features and use a simple technique to extract the singular points features for classification, namely cores & deltas. Feature revision by using *Fuzzy Self Organizing Map (FSOM)* for training and testing to gave a flexible classification system. In section 2 an overview of a biometric technology is reported. Section 3 analysis and discussion of how to extract the singular points, and report the general step of the classification algorithm. Section 4 gave a short report of *FSOM* learning approach. Section 5 discussed the automatic fingerprint matcher (identification and verification steps). Finally section 6 draw some remarks.

2. Biometric Technology

Human identity is a delicate notion that requires consideration at the levels of philosophy and psychology. Biometrics has long been recognized as an effective and highly accurate way of determining the identity of individuals. Some biometrics already used by government agencies and others physical security, immigration control, and prisoner/parolee control. Other business applications, such as minimize fraud in benefits programs, CMC, ATM, building security, point-of-sale terminals and credit card payment systems, benefit payment (currently, UK benefit fraud accounts for more than £3 billion per annum), home shopping, and video on demand, [13]. Most of the biometrics now has being developed and marketed as computer & network security technologies. The awareness and development of the biometric related technologies will provides an international forum for research and development, system design and integration, application development, market development and other issues.

There are many different biometrics in use today and several others under investigation or development for identification and verification purposes. Each of these, biometrics (physiological or behavioural characteristics) has its own feature of strengths and weaknesses, [16]. Some are too intrusive to be accepted by the general user population while others may not afford the high degree of accuracy required in some applications. The biometrics which seem best suited for the most information security applications are fingerprint, eye iris, voice, and face recognition. either singularly or in combination. These. biometrics are likely to dominate the portion of the computer/network identification and authentication devices market captured by the biometrics industry. Currently, the performance of biometric systems is gauged mostly by error rates. A false accepts occurs when an unauthorized user is identified as an authorized user and therefore accepted by the system. A false reject occurs when an authorized user is not recognized as such. and is rejected by the system. In order to describe the performance of a system. both the false accept rate (FAR) and false reject rate (FRR) must be determined. These FARs and FRRs are accepted as the metrics by which biometric system performance is judged today. Although, the final judgement dependent on many other issues, such as ergonomics (ease of use), universality, uniqueness, permanence, acceptability, speed, hardware simplicity, plus the area of the application. Here we concentrate on fingerprint and iris technologies as they have some extra advantages in terms of speed, hardware simplicity, accuracy, uniqueness, performance, and applicability.

3. Fingerprint Classification

Manual fingerprint classification methods are very time consuming, and usually not accurate. Fast and accurate fingerprint classification system is essential to speed up both existing and future automatic fingerprint identification. Fingerprint classification is to assign a given fingerprint to one of the pre-specified categories according to its geometric appearance. A fingerprint classification algorithm presented in this paper is classified fingerprints into five main classes as arch, tended arch, left loop, right loop, and whorl.

- *Arch:* is a pattern of convex ridges with a peak in the middle. In which the ridges enter on one side and flow out the opposite side. Figure: 1 (a).
- *Tented Arch*: consists of a global structure of convex ridges on top. Figure: 1 (b) depicts tended arch fingerprint.
- *Whorl:* have two deltas and global convex ridges. at least one ridge makes a complete circle, giving an overall circular effect within the pattern area. Figure: 2 (a) depicts a whorl fingerprint.
- *Right Loop:* in the right loop, the ridges are toward right from center of the print arrange themselves in the form of a hairpin, making a backward turn without twist, with the delta at the

left of the core. Figure: 2 (b) depicts a left loop fingerprint.

Left Loop: in the left loop, the are ridges toward left from center of the print arrange themselves in the form of a hairpin, making a backward turn without twist, with the delta at the right of the core. Figure 2 (c) depicts a left loop fingerprint. As can be seen in Figures: 1 & 2, an arch fingerprint image contains no cores or deltas, tented arches and loops contains one delta and one core, and whorls have two cores and two deltas.

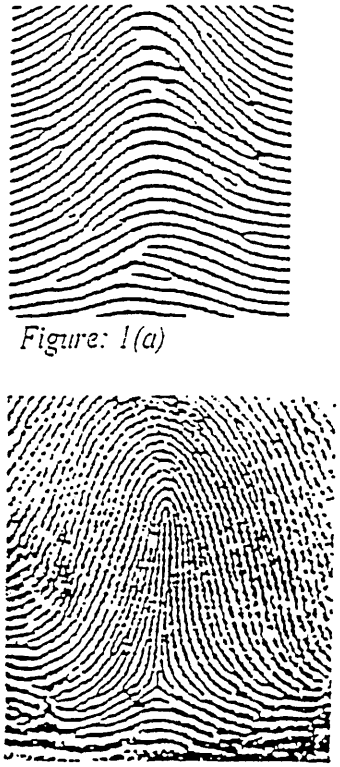


Figure: 1(a)

Figure:1 (b)

Figure 1: (a) An Arch and (b) Tend Arch

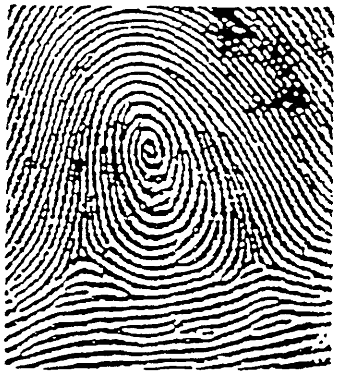


Figure: 2 (a)

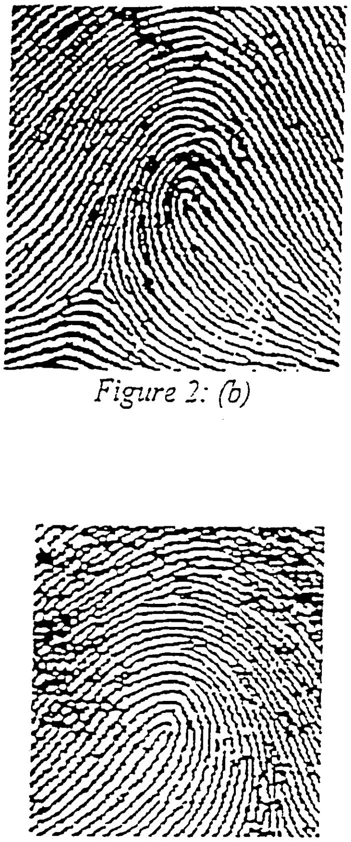


Figure 2: (b)

Figure 2: (c)

Figure 2: (a) A Whorl, (b) A Right Loop and (c) Left Loop

The main purpose of fingerprint classification is to facilitate the management of large fingerprint databases and to speedup the process of fingerprint matching. Generally, manual fingerprint classification is performed within a specific framework such as well-known Henry system. [17]. Different frameworks use different sets of properties. However, no matter what type of framework is used, the classification is based on ridge patterns, local ridge orientations and minutiae. Therefore, if these properties can be described quantitatively and extracted automatically from a fingerprint image then fingerprint classification will become an easier task

3.1 Basic Feature for Classification

The two most prominent (basic) features used for the fingerprint classification are the local characteristics, are called the *Cores* and *the Deltas*, which are also known as the *Singular Points. The* core is the most inner point of the fingerprint is where the ridges are ended with semicircle shape. The delta is abruptly where Bifurcation and ridges are formed a semi-triangle shape. Figure. 3 illustrate a sample of fingerprint image with its core and delta. Other features such as ridges ends, ridges bifurcation, lakes, pores, hooks are various basic features are very useful for the fingerprint identification or verification.

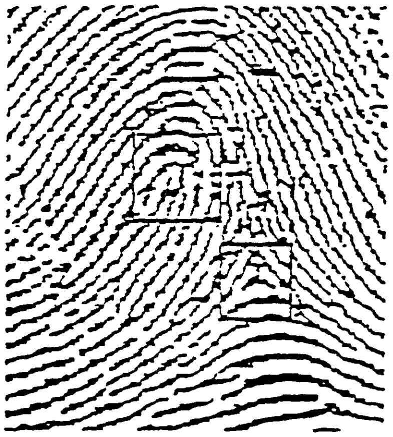


Figure 3: Examples of core and delta on a left loop fingerprint image

3.2 Singular point extraction

A simple and flexible new structural algorithm for classification of fingerprint is based on basic features *"Core"* and "*Delta*" and their orientation. The algorithm extracts the (x,y) positions of the singular points which represented on a fingerprint image, and performs classification by using fuzzy neural classifier based on the number and location of the detected cores and deltas. The classifier is rule-based, where are the rules are generated independent of given data set. The structure of our algorithm diagram is shown in figure 4.

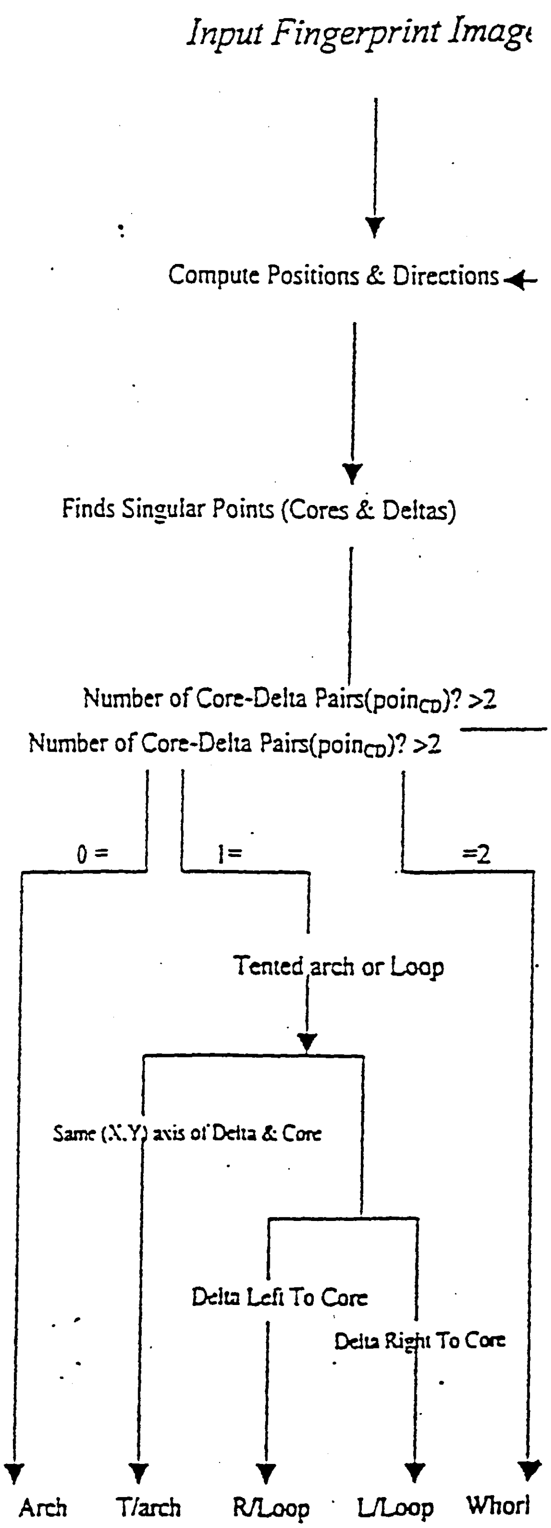

Input Fingerprint Image
Compute Positions & Directions
Finds Singular Points (Cores & Deltas)
Number of Core-Delta Pairs(poin$_{CD}$)? >2
Number of Core-Delta Pairs(poin$_{CD}$)? >2
0 =
1=
=2
Tented arch or Loop
Same (X.Y) axis of Delta & Core
Delta Left To Core
Delta Right To Core
Arch
T/arch
R/Loop
L/Loop
Whorl

Figure: 4. Block Diagram of the Fingerprint Classification Algorithm

The singular points represent by poin$_{CD}$ is used to determine the number of cores (N_c) and the number of deltas (N_d) points in the fingerprint image. A digital closed [illegible] curve, φ; about 16 pixel long, around each pixel is used to compute the poin$_{CD}$ index as defined below:

$$\mathrm{poin}_{CD}(i,j) = 1/2\pi\sum_{k=0}^{N\varphi}\Delta(k)$$

Where

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

$$\delta(k) = O'(\varphi_{x(i')},\varphi_{y(i')})-O'(\varphi_{x(i)},\varphi_{y(i)}),$$

$$i' = (i+1)\mathrm{mod}N\varphi,$$

O is the orientation field, and $\varphi_{x(i)}$ and $\varphi_{y(i)}$ denote co-ordinates of the i^{th} point on the arc length parameterized closed curve φ. The fingerprint classification algorithm classifies input fingerprints into five predefined classes according to the number of singular points detected, and their relative positions. Let O' be the oriented image, N_c and N_d be the number of cores and deltas detected from O', respectively. The classification diagram used in our algorithm is as follows:

1. If (N_c=0) and (N_d=0), then an arch is identified.

2. If (N_c=2) and (N_d=2), then a whorl is identified.

3. If (N_c=1) and (N_d=1), then classify the input by using the core and delta positions assessment by estimating the symmetric axis which crosses the core in its local neighborhood and compute the angle, β, between the line segment from the core to the delta and the symmetric axis:

a. If the core and delta in the same vertical (x,y) positions, or (β<5), then a tended arch is identified.

b. If the delta to the left of the core's (x,y) axis, and (β>10), then a right loop is identified.

d. If the delta to the right of the core's (x,y) axis, and (β>10), then a left loop is identified.

4. If none of the above conditions is satisfied, then repeat the processing again.

Then we had input these features of the fingerprint classification criteria to our fuzzy neural network classifier.

4. Fuzzy Neural Approach

In the fuzzy artificial neural network the neural network part is primarily used for its learning and classification and retrieval. The neural network part automatically generates fuzzy logic rules and membership functions during the training period. In addition even after training, the neural networks keeps updating the membership functions and fuzzy logic rules as it learns more and more from its input signals, [1,2]. Fuzzy logic, on the other hand, used to infer and provide a crisp or defuzzified output when fuzzy parameters exist, [3,4]. In this regard, we attempt to combine Fuzzy Logic and Kohonen's Self-organized Map, have been proposed. Our algorithm investigated the fuzzy-neural models by integrating fuzzy membership function and the self-organizing map network learning techniques, for purpose of image clustering. In particular, we have investigated the combination of features of neural networks (with learning ability, self-organizing and high-speed parallel structure) and fuzzy systems (with ability to process fuzzy information using fuzzy membership) to form a Fuzzy Self-Organizing Map Networks Learning (FSOML), which can learn from environments.

5. Fingerprint Matching

Fingerprint matching determines whether two fingerprints are from the same finger or not (fingerprint verification), or search given fingerprint in database of template (fingerprint identification). It is widely believed that if two fingerprints are from the same source, then their local ridge structures (minutiae details) match each other topologically. Matching can be separated into two categories: verification and identification, [8]. *Verification* is the comparison of a claimant fingerprint against an enrollee fingerprint, where the intention is that the claimant fingerprint matches the enrollee fingerprint. To prepare for verification, a person initially enrolls his her fingerprint into the verification system. A representation of that fingerprint is stored in some compressed format along with the person's name or other identity. Subsequently, each access is authenticated by the person identifying him or herself, then applying the fingerprint to the system such that, the identity can be verified. Verification is also known as, *one-to-one matching*. *Identification* is the traditional domain of criminal fingerprint matching. A fingerprint of unknown ownership is matched against a database of known fingerprints to associate a crime with an identity. Identification is also known as, *one-to-many matching*. A number of different types of local ridge descriptions have been identified. The two most prominent structures are ridge endings and ridge bifurcation's which are usually called minutiae. Fig. 3 in section 3 shows examples of ridge endings and ridge bifurcations.

6. Conclusions

Fingerprint classification provides an important mechanism for automatic fingerprint recognition systems. We have approached a simple and flexible fingerprint classification algorithm which classifies input fingerprints into five categories according to the number of the cores and deltas (singular points) and their relative x,y position. Extracted features by using this algorithm has been input to our FSOM network for training and testing which gave us very good experimental results. A FSOM paradigm adopting a principle of learns according to how well it wins is proposed. Unlike the Self Organizing Map (SOM) where only one neuron will win and learn at each competition, every neuron in the FSOM to a certain degree wins, depending on it is distance to the input pattern.

Reference

[1] L. A. Zadeh, Fuzzy Sets, Information and Control, vol. 8, pp. 338-352. 1965.

[2] J. H. Chian and P. D. Gader, Hybrid Fuzzy-Neural Systems in Handwritten Word Recognition. IEEE Transactions on Fuzzy Systems. Vol. 5, No. 4, 1997.

[3] C. Yong, S. Minghao and H. Yongbao, A Method of Pattern Recognition Based upon Synthetic Technology of Fuzzy Logic and Neural Network. Department of Computer Science, Fudan University, Shanghai 200433 P. R. China, 1998.

[4] D. W. Patterson, Artificial Neural Network Theory and Application, 1996.

[5] D. A. Linkens and H. O. Nyongesa. Learning Systems in Intelligent Control: an appraisal of fuzzy. neural and genetic algorithms control applications, 1996.

[6] H. Adeli and S. L. Hung, Machine Learning. Neural Networks. Genetic Algorithms and Fuzzy Systems, 1995.

[7] D. Zhang, M. Kamel and M. I. Elmasry, fuzzy clustering neural network using fuzzy competitive learning. world congress on neural networks. international. 1993.

[8] S Mohamed, H Noyngesa and J Seddiqi, Automatic Fingerprint Identification System Using Fuzzy Neural Techniques, Proceeding of the IC-AI-2000, Vol. 2, PP. 859-865, June 2000.

[9] C. T. Leondes, Image Processing and Pattern Recognition, Vol. 5 of Neural Network Systems Techniques and Applications, 1997.

[10] J. L. Blue,. G. T. Canfela, P. J. Grother, R.Chellappa and C. L. Wilson, Evaluation of Pattern Classifiers for Fingerprint and OCR Applications, Pattern Recognition, Vol. 27, No. 4, PP. 485-501, 1994.

[11] D. K Isenor And S. G. Zaky, Fingerprint Identification Using Graph Matchin-, Pattern Recognition, Vol. 19, No. 2, Pp. 113-122, 1986.

[12] B. M. Mehtre & B. Chatterjee, Segmentation of Fingerprint Images Using Composite Method, Pattern Recognition, Vol. 22, No. 4, PP. 381-385, 1989.

[13] M. S. Michael Chong, K. L. Robert Gay, H. N. Tan & J. Liu, Automatic Representation Of Fingerprints for Data Copression by B-Spline Functions, Pattern Recognition, Vol. 25. No. 10. PP. 1199-1210, 1992.

[14] O. Lawrence Gorman & V. Jeffrey Nickerson, an Approach to Fingerprint Filter Design, Pattern Recognition, Vol. 22, No. 1, PP. 29-38, 1988.

[15] A. P. Fitz and R. J. Green, Fingerprint Classification Using a Hexagonal Fast Fourier Transform, Pattern Recognition, Vol. 29, No. 10, PP. 1587-1597, 1996.

[16] S. Michael M. Chong, T. Han Ngee, Liu Jun and Robert K. L. Gay, Geometeric Framework for Fingerprint Image Classification, Pattern Recognition, Vol. 30, No. 9, PP. 1475-1488, 1997.

[17] V. S. Srinivasan and N. N. Murthy, Detection of Singular Points in Finger Print Images, Pattern Recognition, Vol. 25, No. 2, PP. 139-153, 1992.

[18] 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. 429-345, 1999.

An Image Metamorphosis Algorithm Based on Navier Spline Interpolation

Suliman M Mohamed[1], *El- Sayed M. El-Horbaty*[2], *Aboul-Ella Hassanien*[3]

[1]Department of Information Systems & Computing, Brunel University, Uxbridge, UK

[2]Faculty of Computer & Information Sciences, Ain Shams University, Egypt

[3]Faculty of Computers and Information, Cairo University, Egypt

Abstract: In this paper, we propose an image metamorphosis algorithm that uses Navier spline to generate warp functions for interpolating scattered data points. The Navier spline interpolation can be expressed as the linear combination of an affine transformation and a solution of Navier partial differential equation. Our algorithm generates a smooth warp that reflects feature point correspondences. The algorithm allows each feature point in the source image to be mapped to the corresponding feature point in the destination image. Once the images are warped to align the positions of features and their shapes, the in-between facial animation from two given facial images can be defined by cross dissolving the positions of correspondence features, shapes and colors The algorithm is efficient in time complexity and smoothly interpolated morphed images with only a remarkably small number of specified feature points. The implementations of the algorithm with some experimental result are given.

Keywords: Metamorphosis, Animation, Image processing, Complexity time.

1. Introduction

Image metamorphosis (image morphing, for short) algorithms have been widely used in creating special effects for television commercials, music videos such as Michael Jackson's Black or White, and movies such as Willow and Indiana Jones and the Last Crusade [9]. Applications of image morphing in the entertainment industry date back to the old cross-dissolving process, which originated at Industrial Light and Magic (ILM). Cross dissolving is the process of mixing the colors of a source image with those in a destination image to form new colors in an intermediate image.

The problem in image morphing is how to generate an in-between image from two given images. An animator establishes the correspondence between the two images with pairs of points or line segments. In most feature based morphing algorithms, these features are defined manually using an interactive user interface. The user interface presents two images side by side such that correspondence points can be defined by alternately picking points in the two images.

Once the corresponding points from two given facial images are paired, we can construct a warping function to interpolate the position of the feature across the morphing sequence. A warp is a two dimensional geometric transformation which generates a warped image when applied to an input image. When two images are given, the image morphing process first establishes the feature correspondence between them. Based on these feature correspondence pairs a warping function is then constructed. The warping function distorts the images to align the positions of the features and their shapes. Finally, the in-between images can be generated by cross dissolving the colors at each corresponding pair of pixels in the warped images. Such correspondences can be obtained by interpolating the positions of the correspondence feature points. Therefore, the image morphing process should allow convenient feature specification, and show a predictable distortion which reflects the feature correspondence.

The rest of this paper is organized as follows. In Section 2, we describe the related published work. The required computation of the algorithm is given in Section 3 .The design and analysis of the algorithm are given in Section 3. In Section 4, the implementation of the algorithm with some experimental results are given. Finally, conclusions and future work are discussed in Section 5.

2. Related Work

Most of the literature on image morphing falls into three categories. One is mesh morphing. which is an early morphing technique. Industrial Light and Magic to create the special effects for the movie Willow [9] used it. Wolberg [9] used a nonuniform control mesh and computed a warp by using cubic spline interpolation. Nishita [7] also used a nonuniform control mesh and computed a warp by using two dimensional free form deformations and a Bezier clipping algorithm. Moving the control points of the mesh on the images performed the deformation. These methods have the drawback that they require a control mesh on an image. whereas its features may have an arbitrary structure. Also, using the user interface to define the feature correspondence is very difficult and time consuming.

The second morphing category is called field morphing. This technique depends on pairs of lines, one line in a source image and a corresponding line in a destination image. Field morphing was used by Pacific Data Images to create the morphing sequence in Michael Jackson's video clip Black or White [3]. The field warp mapping is specified by defining a weighted average of the influence fields around each of the features of the images to be warped. This algorithm, of course. simplifies the work of user interface but the warp generation is complicated and it suffers from unexpected distortions referred to as ghost's [3].

The last category is point-based morphing. It is an important particular case of morphing where each feature is distinguished by a set of points in the image. Constructing the surfaces that interpolates scattered data points can derive the warp function. Recently. two warping algorithms have been proposed which are based on radial basis functions [1] and thin plate splines [4], and which formulate warp generation as scattered data interpolation. These techniques generate a smooth warp that exactly reflects the feature correspondence. An energy minimization method has been proposed for deriving the warp function [6]. This method allows extensive feature specification primitives such as points, polylines, and curves. Internally, all primitives are sampled and reduced to a collection of points. These points are then used to generate a warp.

3. Navier Spline Interpolation

The warping of scattered data points is a problem that appears frequently in science and engineering. The basic problem involves the construction of a reasonable interpolation function, which goes through a set of data points. In recent years, digital image warping has received a great deal of interest. It plays an important role in a wide range of applications, including remote sensing, medical imaging and machine vision as well as in computer graphics. In remote sensing application length surface curvature and oblique viewing angles cause distortions. For example, when a photo of the Earth's surface is transmitted from a satellite to ground, it is typically warped to correct the surface curvature. In medical imaging [2] image warping may be applied to the registration of medical data sets between various modalities, i.e., sensor types such as Computer Tomography and Magnetic Resonance Imaging, or between patients and an anatomical atlas. For example, one often wishes to compare two views of the same part of the anatomy acquired using the same imaging modality in order to detect differences. In computer graphics [9], image warping may be used to create an intermediate image for two given images. A well-known example of image warping is a computer movie production technique called image morphing.

In this section, we propose a new algorithm for constructing the mapping function that interpolates scattered data points. It is based on the Navier spline [10]. It allows each feature point in the source image to be mapped to the corresponding feature point in the warped image. It is based on a partial differential equation by Navier [5,18] that describes the equilibrium displacement of an elastic body subjected to forces. Once image features are paired with correspondence points $(\bar{p}_i, \bar{q}_i)$, we can construct the Navier spline transformation and use it as interpolation map from R^2 to R^2 relating the set of corresponding feature points.

In point based warping, sets of n point pairs $(\bar{p}_i, \bar{q}_i)$ are selected in the source and destination images. For instance, if p_1 is the coordinate of a feature point in the source image, q_1 is the corresponding point of the same feature in the warped image. The displacement between a pair of points is:

$$\bar{d}_i = \bar{q}_i - \bar{p}_i \qquad (1)$$

The coordinate transformation must be determined such that it matches the displacements $\bar{r}_i$ and interpolates them elsewhere. The coordinate transformation $\bar{C}_{trans}(\bar{x})$ is defined by

$$\bar{C}_{trans}(\bar{x}) = (f_x(\bar{x}), f_y(\bar{x})) \qquad (2)$$

$f_x(\bar{x})$ and $f_y(\bar{x})$ are Navier Splines that represent displacements and take the form:

$$\bar{C}_{trans} = \sum_{i-1}^{N} K(\bar{x} - \bar{p}_i) w_i + A\bar{x} + \bar{b} \qquad (3)$$

$A\bar{x}+\bar{b}$ is the affine part of the Navier spline in which

$$\bar{u}(\bar{x}) = K(\bar{x})\bar{w} \tag{4}$$

Where the coefficient, $\bar{w} = [w_1 \ w_2]^T$, is the strength of the force field. $\bar{u}(\bar{x})$ is the displacement of a point within the body from the original position $\bar{\mathbf{x}}$. Above, $K(\bar{x})$ is defined as:

$$K(\bar{x}) = [r(\bar{x})^2(M_1)I - (M_2 - 5)\bar{x}x^T]r(\bar{x})$$

Where $M_1 = 3\alpha\, ln[r(\bar{x})] - \beta$, $M_2 = 12ln[r(\bar{x})]$.
Here, I is a $2x2$ identity matrix, $r(\bar{x}) = |\bar{x}|$, and $\alpha = 12(1-k)-1$, $k = \lambda/\{2(\lambda+\mu)\}$ is Poisson's ratio. $\beta = 18(1-k)-2$.

Equation (4) is the fundamental solution of the Navier equilibrium partial differential equations for the elastic body subjected to forces which serve as the constraint equations in the elastic body:

$$\mu\nabla^2\bar{u}(\bar{x}) + (\mu + \lambda)\nabla[\nabla.\bar{u}(\bar{x})] + \bar{F}(\bar{x}) = 0$$

Where ∇^2 and ∇ denote the Laplacian and Gradient. respectively, μ and λ are the Lame coefficients which describe the physical properties of the elastic material. $\bar{F}(\bar{x})$ are the external forces distributed everywhere in the body; we should select these forces so that the warping of the Navier spline is smooth. There are many different ways to derive the forces[5], such as using information from the input data or from external knowledge (i.e., interactively or from a knowledge base). These forces should be selected to generate a smooth warp:

$$\bar{F}(\bar{x}) = w_i r(\bar{x})^2 \ln[r(\bar{x})]$$

$\bar{F}(\bar{x})$ and $\bar{x}$ are all 2D vectors.

The Navier spline coefficients are computed by solving the following linear system:

$$W = L^{-1}Y \tag{5}$$

Spline can be evaluated by using Equation (3) as the interpolation function that interpolates scattered data points that satisfy $\bar{f}(\bar{p}_i) = \bar{q}_i$ if and only if L is not singular.

3.1. Intermediate Images

In-between images can easily be implemented using a sequence of interpolation between the source and the destination images. Given two images I_s and I_d, with variable $\alpha \varepsilon [0,1]$, an in-between image I_α is created such that I_α is similar to I_s at $\alpha \Rightarrow 0$ and similar to I_d as $\alpha \Rightarrow 1$. We assume that the variable α varies from 0 to 1, so that the source image I_s continuously changes to the destination image I_d. The in-between images I_α are defined by interpolating a new set of feature points from their positions in I_s and I_d.

Let W_s be the warp function, which specifies the corresponding point in I_d for each point in I_s. When it is applied to I_s, W_s have to distort I_s to match I_d in the positions and shapes of features. Let W_d be the warp function from I_d to I_s. It is required to map the features on I_d to the features on I_s when it distorts I_d. The in-between images can be defined using the following deformed cross dissolve function:

$$I_\alpha = [1 - K(\alpha)] W_s^\alpha (I_s) + K(\alpha) W_d^{1-\alpha} (I_d) \quad (6)$$

Here, $K(\alpha)$ is the transition control defined on the image. It determines how fast each point on I_s moves to the corresponding point on the destination image I_d. Also, it determines how much the color of each point on I_s is reflected on the corresponding point in $I(\alpha)$. $K(\alpha)$ controls the rate of transition in Equation (6). For the color transformations, linear interpolation is not defined on the distorted images $I_s(\alpha)$ and $I_d(\alpha)$, but on the given images I_s and I_d, respectively. Hence, we used the transition control $K(\alpha)$ to attenuate the color intensities of I_s and I_d before applying the warp function. The transformation of positions and colors can be independently handled by specifying a different transition function for each. We can verify that $I_{\{\alpha=0\}} = I_s$ **and** $I_{\{\alpha=1\}} = I_d$. The complete algorithm for generation of the in-between sequences of images is described as follows:

<u>1- Initialization step</u>

- Frame =0
- **n** control points are selected in the source and destination images

<u>2- Distance points calculations</u>

```
- for i =1 to n do
- calculate $\bar{d}_i = \bar{q}_i - \bar{p}_i$
```

<u>3- Coordinate Transformation functions</u>

```
- Solve equation number (3) to determine $\bar{f}(\bar{x})$ and $\bar{f}(\bar{y})$
- For each frame do
    -Interpolate the points between $\bar{d}_i$ and $\bar{p}_i$ points
    -Warp $I_s$ to $I_1$ using the control points $\bar{d}_i$ and $\bar{p}_i$
```

-resampling and attenuating the result

-Warp I_d to I_2 using control points $\bar{\mathbf{d}}_i$ and $\bar{\mathbf{q}}_i$

-resampling and attenuating the result

4-Cross dissolving step

- Apply equation (6) to generate the intermediate image
- Frame = Frame +1
- Go to step 3

where I_1 and I_2 are the intermediate state of the source deformed toward the destination and vice versa.

3.2 The Analysis of the Algorithm

Now, we will examine the time complexity of previous morphing algorithms and compare them with our algorithm. Examining Equation (5) and Equation (3) of Section 3, the complexity time of our algorithm is $O(n^3)$ for Equation (3) and $O(n G)$ for Equation (5), where G is the number of pixels in the image, n is the total number of feature points specified. The complexity time of Beier and Neely's method [3] can be estimated to be about $O(t G w_l)$ where t is the number of the feature lines and w_l is amount of computation required for one pair of feature line. The complexity time of Lee's algorithm [6] is $O(r w_2 G)$ where r is the number of relaxation required on each of a grid, w_2 is the amount of computation required for one relaxation on the finest grid, which is apparently proportional to G. However, in these methods, all pixel points constitute unknowns, so that the entire set of pixel points must fully converge to a tolerable level before the solution is visible as the warped image. The complexity time for thin plate spline algorithm[4] is $O(G+n^3)$, where is n is the number of defined points and G is the total number of pixels.

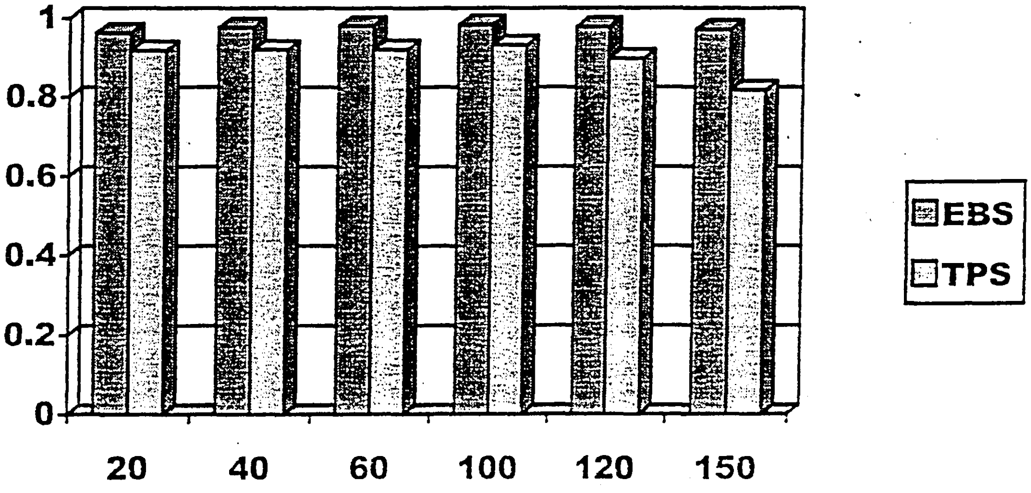


Figure 1 Correlation coefficient of Navier spline and thin plate spline

On the other hand, we study the smoothness for our algorithm by computing the correlation coefficients (i.e. smoothness) of the computed values f_x,f_y, and compare them with the same values calculated from the thin plate spline (TPS) algorithm[4]. Figure 1 represents the correlation coefficients of both Navier spline and TPS, higher values indicate a smoother image. One can see that our spline algorithm is smoother than thin plate spline algorithm. For instance, the correlation coefficient with 60 control points for both spline are .979 and .920, respectively. At the same time, the correlation coefficients for the thin plate spline decrease dramatically for more than 100 control points, whereas the coefficients remain about the same for our algorithm. So, it seems that our algorithm is suitable for obtaining satisfactory shape interpolations with more than 100 control points. We found that with around one hundred and fifth points, our algorithm adequately gives satisfied shape interpolation. The computation speed of our method is also fast enough for an interactive environment.

4. Experimental Results and Discussion

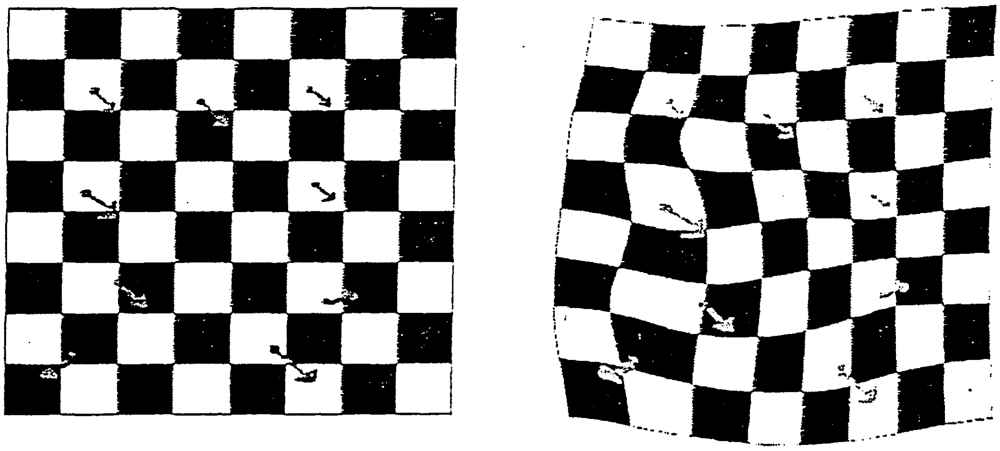
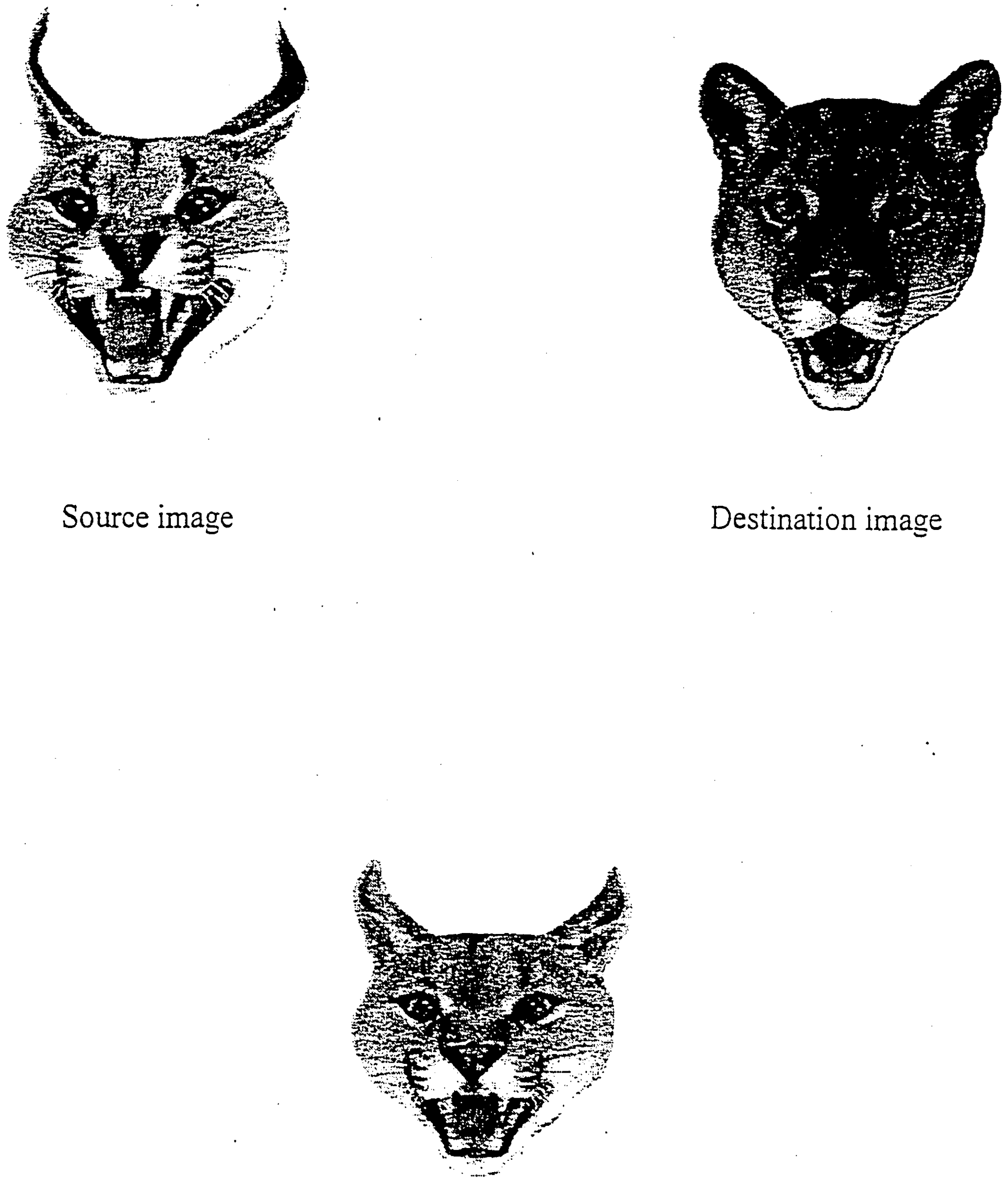


Figure 2 : *2D Navier spline warping checkerboard image*

Figure 2 shows the warping of a checkerboard image using the Navier spline. Figure 2(a) shows the checkerboard with a set of correspondence points. The point at the tail of the arrow represents the location of the points in the source image and the head of the arrow represents the location of the corresponding point in the warped image. The lines between them are the displacement. Figure 2(b) shows the result of warping.



Source image

Destination image

Figure 3: *Intermediate image morphing result*

Figure 3 shows the intermediate image between two facial images. The source image is in the left and the destination image is in the right. The intermediate image was generated using our algorithm. Figure (4) shows the in-between sequence between two given images.

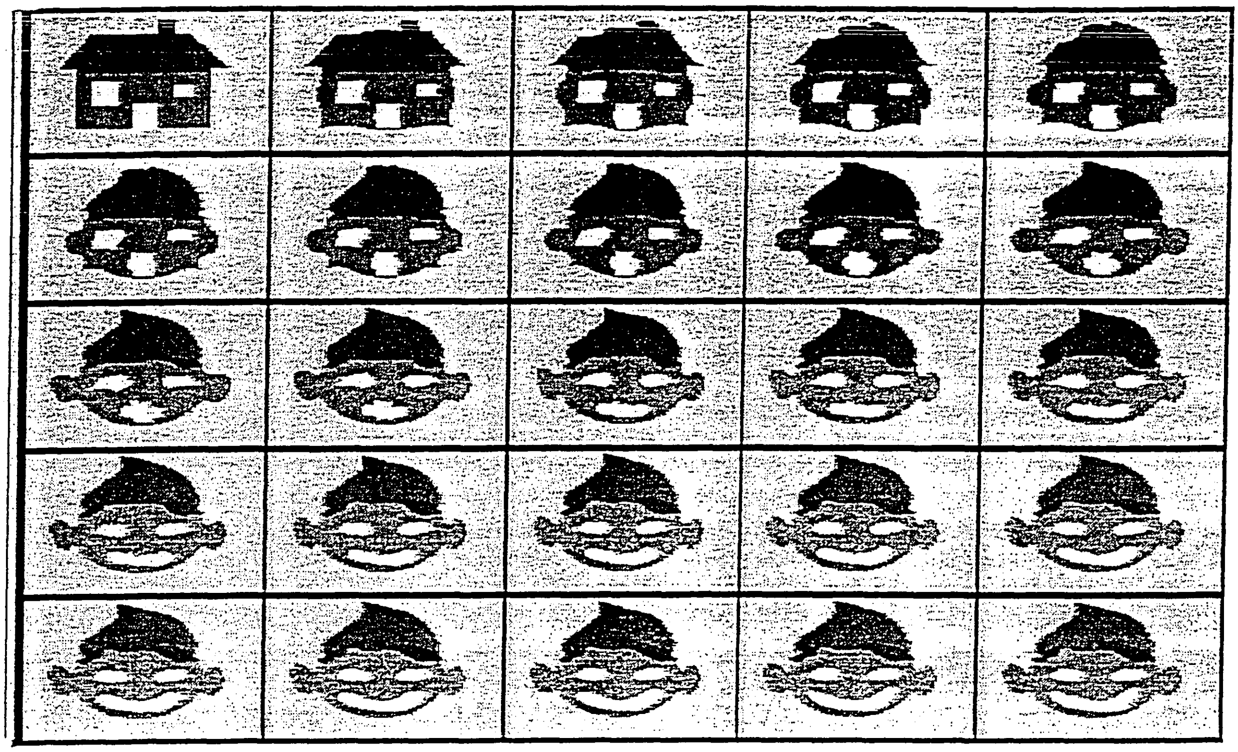


Figure (4): *Animated in between images*

We should note that, the most tedious part of image morphing is to establish the correspondence of features between images by an animator. Algorithms from computer vision may be employed to reduce human intervention, such as an active contour model [5] or active net model [8]. An edge detection algorithm can provide important features on images, and image analysis techniques may be used to find the correspondence between detected features. One of the most challenging problems in image morphing is to develop an efficient method for specifying features and their correspondence, especially when morphing between two given image sequences.

5. Conclusion and Future Work

In this paper, we have proposed a new image morphing algorithm which uses an Navier spline to construct an interpolation map from R^2 to R^2 constrained by a set of corresponding feature points. Navier spline is a technique for geometric transformation in 2D and 3D that is motivated by a physical model for the deformation of elastic materials. It is efficient in time complexity and smoothly interpolated morphed images with only a remarkably small number of specified feature points. It allows each feature point, in the source image to be mapped to the corresponding feature point in the warped image. To generate the in-between images, we have described an efficient cross dissolve algorithm.

Our future work will be to apply the proposed method to the interpolation of intermediate planar slices in medical data sets. Since, the traditional formulation for image morphing considers two input images at a time, i.e., source and target images. In that case, morphing among multiple images is understood to mean a series of transformations from one image to another. This limits any morphed image to take on the features and colors blended from just two input images. Given the success of morphing using this paradigm, it is reasonable to consider the benefits possible from a blend of more than two images at a time. For instance, consider the generation of a facial image that is to have its eyes, ears, nose, and profile derived from five different input images. In this case, morphing among multiple images is understood to mean a seamless blend of several images at once. Morphing among multiple images is ideally suited for image composition applications where elements are seamlessly blended from two or more images. A composite image is treated, as a metamorphosis of selected regions is several input images. The regions seamlessly blend together with respect to geometry and color. Since the proposed algorithm depends only on the distance between pairs of points, it can easily be extended to volume deformation applications, such as registration of volumetric data.

References

1 - N. Arad, N. Dyn, and D. Reisfeld, "*Image Warping by Radial Basis Functions: Application to Facial Expressions*", Computer Vision, Graphics, and Image Processing, vol. 56, no.3, Mar. 1994, pp. 161-172.

2- R. Bajcsy and S. Kovacic, "*Multiresolution Elastic Matching, Computer vision, Graphics, and Image*", Processing, vol. 46, no. 1,1989, pp. 1-21.

3 - T. Beier. and S. Neely, "*Feature based Image Metamorphosis*". SIGGRAPH'92, pp. 35-42.

4 - F.L.Bookstein," *Principal Warps: Thin Plate Spline and the Decomposition of Deformations*", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 11, 1989, pp. 567-583.

5 - M. Kass, A. Witkin and S. Terzopulos, "*Snakes: Active Contour Model*", Int. Journal of Computer Vision. vol. 1, 1988, pp. 321-331.

6- S. Lee, S.Sung, and G. Wolberg, "*Image Metamorphosis Using Snakes and Free-Form Deformations*", Computer Graphics Proceeding, 1995, pp. 439-448.

7 - T. Nishita and E. Nakamae,"*Metamorphosis using Bezier Clipping*", Proceeding of the 1st Pacific Conference in computer graphics and application, Korea, 1993, pp. 162-173.

8 - K.Sakaue and K. Yamamoto, "*Active Net Model and Its Applications to Region Extraction*", Journal of TV. vol. 45, no.10, 1991, pp. 1155-1163, (In Japanese).

9 - G. Wolberg, "*Digital Image Warping*", IEEE Computer Society Press.1990.

10 - S.P. Timoshenko and J. N. Goodier, "*Theory of Elasticity*", McGraw Hill, 1934, Ch.6.

Automatic Fingerprint Identification System Using fuzzy Neural Techniques

Suliman M Mohamed, Henry O Nyongesa and Jawed Siddiqi
Computing Research Center
School of Computer and Management Sciences
Sheffield Hallam University, Sheffield S1 1WB, U.K

Abstract: The successful use of the fingerprint identification has been employed in law enforcement for many years ago. Fingerprint technology, is one of the most mature biometrics technologies. Biometrics identification deals with identification of individuals based on their biological or behavioral characteristics (so-called positive personal identification). However, manual fingerprint identification system is so tedious, time consuming and incapable of meeting today's increasing performance requirements. A good performance of the Automatic Fingerprint Identification System (AFIS) highly demands. In this paper we described the AFIS which allows variations on the basic feature properties extracted from the fingerprint image for a match, then we used fuzzy neural networks learning techniques for testing and training these features. Relative performance of AFIS and the Eye Iris Recognition, among the other Biometrics items, is examined by using issues as FRR, FTA, and FAR.

Keywords: Fingerprint, Biometrics, Features Encoding, Matching, Fuzzy Neural Learning.

1. Introduction

As information becomes the key to wealth in the 21st century, biometrics security will play a central role in providing a high level of security to existing and future products. Police departments in most world countries have long been interested in the improvement of *AFIS* methods as an important factor in making more effective the administration of criminal justice, security and record of the offender's transactions. Although, fingerprint verification systems are usually associated with criminal identification, and police work it has now become more popular in civilian applications. Such as access control, high-security areas in prominent organizations, financial security, verification of firearm purchasers, driver license applicants, and computer networks security. These useful applications have been conceived due to the fingerprint favorable characteristics such as unchangeability and uniqueness in an individual's lifetime. Using the current technology of the fingerprint identification system is much more reliable than the different kinds of popular biometrics identification items technologies based on signature verification, face recognition, speech recognition, eye iris recognition, ...etc. With increasingly large volumes of fingerprints being collected and stored, there is an urgent need to develop fast and accurate AFIS to improve the efficiency and reliability of personal identification and authentication. Fingerprint *recognition* or *identification* system, is made by comparing the ridge detail of two different sets of fingerprints. The expert makes sure that the ridge detail of the two sets of fingerprints is in the same coincidental sequence. Usually, fingerprint recognition is performed manually by professional fingerprint experts. However, manual fingerprint recognition is so tedious, time-consuming, and expensive that is does not meet the performance requirements of the today's new applications. Various approaches for preprocessing and fingerprint recognition have been investigated for the purpose of automatic fingerprint recognition. These can fail into either one of the following categories: structural, statistical, syntactic, geometric, mathematical, hybrid approaches and artificial neural networks. [1,2]. In fingerprint matching (identification or verification), one of the problems is to develop image processing algorithms that are efficient and not computationally intensive for pre-processing of fingerprint images. Although, it may need many steps to overcoming this problem like to experiment with several image pre-processing algorithms and compare the effectiveness of them. One of the overriding requirements is that the image processing technique applied to fingerprint images does not create new features or lose existing features, [5]. In this regard we analysis the fingerprint image features and use a simple technique to extract the most prominent features, namely ridges endings & bufircations or known as *minutiae*. Once extracted, feature properties are allowed to deviate by a user definable amount. Additionally number of features to be matched for successful identification may also be defined, depending on the nature of the application area. Feature revision by using *Fuzzy Neural Learning Techniques (FNNT)* for training and testing to gave a flexible identification system. In section 2 report of fingerprint acquisition methods. Section 3 analysis and discussion of how to extract the basic features. Section 4 gave a short report of fuzzy neural learning approach. Section 5 discussed the AFIS matcher (classification, identification, and verification steps). Finally section 6 draw some remarks.

2. Fingerprint Acquisition

A number of methods are used to acquire fingerprints. Among them, the inked impression method remains the most popular. It has been essentially a standard technique for fingerprint acquisition for more than 100 years, [6]. The first step in capturing an inked impression of a fingerprint is to place a few dabs of ink on a slab then rolling it out smoothly with a roller until the slab is covered with a thin. Then the finger is rolled from one side of the nail to the other side over the inked slab, which inks the ridge patterns on top of the finger completely. After that the finger is rolled on a piece of paper so that the inked impression of the ridge pattern of the finger appears on the paper. Obviously, this method is time-consuming and unsuitable for an on-line fingerprint verification system. The second method is a more efficient and reliable optical data generation system. It consists of a prism and a uniform light beam that transforms the three-dimensional data into two-dimensional data, which can be photographed. The optical method of fingerprint data generation is not perfect either because the contrast and focus of the image obtained are sometimes poor. However, the method is clean, fast, and most of the problem can be overcome by good preprocessing techniques such as grayscale-to-binary conversion and enhancement. Innovations in optical devices have been made recently as mid-1990s, an optical sensor was housed in a box about 6x3x6 inches. The third method is the ink-less fingerprint scanners are now available which are capable of directly acquiring fingerprints in digital form. This method eliminates the intermediate digitization process of inked fingerprint impressions and makes it possible to build an on-line system. The fourth method so-called solid-state sensors have appeared on the market recently. These are microchips containing a surface that images the fingerprint via one of the several technologies, including electrical measurements and temperature sensitive sensors. One of the most important factors that will decide when fingerprint verification will be commercially successful in the large-volume personal verification market are low cost and compact size.

3. Fingerprint Features Analysis

The fingerprint image is made of foreground ridges which are separated by background valleys, ridge flow direction forms different patterns like arches, loops, wholes, and also gives rise to various minutiae like ridge endings, ridge bifurcation's, cores, deltas etc. Both foreground and background consist of a similar set of minutiae, the tiny patterns used for fingerprint classification. Each individual has a unique fingerprint and the uniqueness of a fingerprint, is exclusively determined by the local ridge characteristics and their relationship. These local ridge characteristics are not evenly distributed. Most of them depend heavily on the impression conditions and quality of fingerprints and rarely observed in fingerprints. A number of useful applications have been conceived due to the fingerprint favorable characteristics such as unchangeability and uniqueness in an individual's lifetime. Inherently, using current technology of the fingerprint identification is much more reliable than the kinds of popular biometrics identification methods. The adaptability of fingerprint recognition hardware to the computer keyboard and mouse make it a viable alternative to the workstations, password. The reliability of the minutiae features is relatively very high and depends on the quality of the prints, skin condition, and capture method.

3.1 Fingerprint Basic Feature Extraction

Figure: 1 (a) illustrates a sample of fingerprint image with different kinds of the classification features. The two most prominent (basic) features of the fingerprint in the local characteristics, known as the *minutiae*, are: The *Ridge Endings* and *Ridge Bifurcations. The* ridge ending is where a ridge ends abruptly. The ridges bifurcation is formed where a ridge, which has previously run parallel to another ridge, joins that ridge, as illustrated in Figure: 1(b). Other features such as lakes, pores, hooks are various combinations of these basic features.

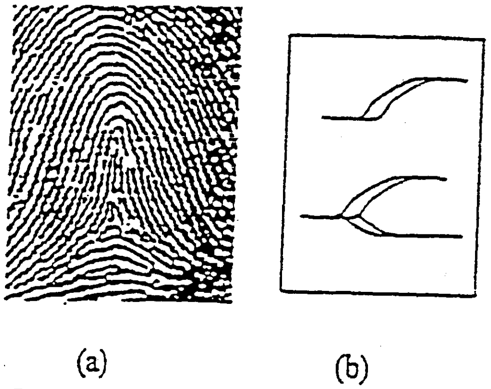


Figure:1 (a) fingerprint image sample (b) ridge end & bifurcation

3.2 Feature Encoding

To encoding fingerprint features we used a gray-scale fingerprint images to extracts basic features (minutiae). Using the minutiae position in x, y co-ordinate, minutiae type (ridge ending (E) and bifurcations (B)), and minutiae direction (Angle) to extract the minutiae information of the fingerprint images.

Each feature encoder has the following information stored for it:

- *Position*
- *Type*
- *Feature direction*

Table 1. A typical three encoded fingerprint basic features

X: Position	Y: Position	Type: E=end B=bifurcation	Angle
275	357	E	10
1217	500	B	8
6011	1982	B	16

There are only two types of minutiae, ridge endings and bifurcations. *Table 1.* depicts three typical encoded features. The position of each feature is expressed in x,y axis values, top left hand corner being the origin (0,0). Feature direction is expressed by 16 level increments covering 360 degrees. Each increment corresponds to 22.50 degrees.

4. Fuzzy Neural Approach

Fuzzy logic and neural networks are the most computational intelligence key technologies for representing human knowledge in the brain, and for constructing adaptive systems. As the tide of using neural network and fuzzy logic grows up a number of studies have shown that although, neural networks are powerful in machine learning, associative memory, and parallel processing but they fails to do well in some symbol processing and indefinite reasoning. On the contrary, the fuzzy logic systems are powerful in indefinite reasoning and symbol processing but they fail to do well in associative memory, [2,3]. So it is supposed to use the synthetic technology of these two techniques which can complement to each other. In particular, we investigated fingerprint image features clustering by using Fuzzy Self-Organizing Map (*FSOM*) learning, with the learning and self-organizing. features of Neural Networks and the ability to process fuzzy data using Fuzzy Membership. Finally we used supervised Neural Networks for training and testing the encoded features information. In *FSOM* by considering the win as a fuzzy set, every neuron to a certain degree wins, depending on its distance to the current training pattern. Hence, it has to learn according to its win membership during the competition. In this way, a learn according to how well it wins fuzzy self-organizing map learning (FSOML) paradigm results, based upon which a fuzzy class of SOM algorithm can be developed. In this section, we present the derivation of a fuzzy version of the SOM algorithm. Furthermore, we comment on the advantages of the derived fuzzy algorithm.

4. AFIS Matcher

To implement automatic fingerprint identification (some times known as fingerprint recognition), we need to match the encoded features. As a result of many studies, automatic fingerprint recognition systems are in great demand. Although, a significant progress has been made in designing automatic fingerprint identification systems, over the past 25 years, a number of limitations in achieving the desired faster matching of the fingerprint image. An automatic fingerprint identification system is concerned with some very important issues include the following: Fingerprint Acquisition: How to acquire fingerprint images and how to present them in a proper format, as reported in section 2. Fingerprint Classification: To assign a given fingerprint to one of the pre-specified categories according to its geometric appearance. Fingerprint Matching: verification or identification. Fingerprint verification is to determine whether two fingerprints are from the same finger or not. Fingerprint Identification: is to search for a query fingerprint in a database.

4.2 Fingerprint Classification

The goal of fingerprint classification is to assign a given fingerprint to a specific category according to its geometric properties. In general there are six classes: *(i) Arch (ii) Tented arch (iii) Right loop (iv) Left loop (v) Twin loop and. (vi) whorl.*. The main purpose of fingerprint classification is to facilitate the management of large fingerprint databases and to speedup the process of fingerprint matching. Generally, manual fingerprint classification is performed within a specific framework such as well-known Henry system. Different frameworks use different sets of properties. However, no matter what type of framework is used, the classification is based on ridge patterns, local ridge orientations and minutiae. Therefore, if these properties can be described quantitatively and extracted automatically from a fingerprint image then fingerprint classification will become an easier task.

4.3 Fingerprint Matching

Matching can be separated into two categories: known as verification and identification, [7]. *Verification* is the comparison of a claimant fingerprint against an enrollee fingerprint, where the intention is that the claimant fingerprint matches the enrollee fingerprint.

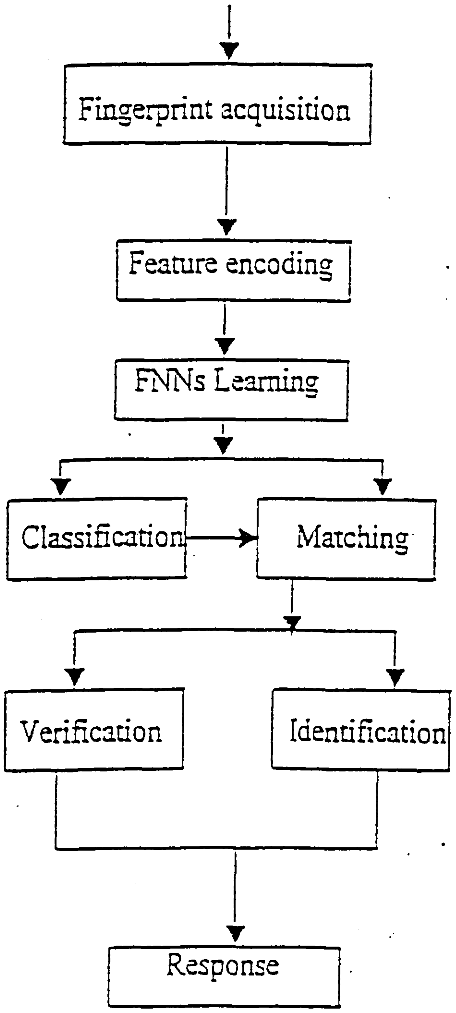


Figure: 2 Steps taken by a general-purpose of AFIS

To prepare for verification, a person initially enrolls his/her fingerprint into the verification system. A representation of that fingerprint is stored in some compressed format along with the person's name or other identity, [8]. Subsequently, each access is authenticated by the person identifying him or herself, then applying the fingerprint to the system such that, the identity can be verified. Verification is also known as, *one-to-one matching*. *Identification* is the traditional domain of criminal fingerprint matching. A fingerprint of unknown ownership is matched against a database of known fingerprints to associate a crime with an identity. Identification is also known as, *one-to-many matching*.

5. AFIS Performance

Most biometrics are used for *verification* using biometric measurement to authenticate a claimed identity. Some biometrics, including fingerprint and iris features, are highly capable of *identification, Figure: 2* illustrates the general steps of AFIS. This means that we determine the true identity of an unknown person by comparing his or her sample measurement to a collection of templates in a biometrics database, without requiring a claim of identity. Errors can occur. For example, when biometric measurement from a live subject is compared to that subject's enrolled template and the system fails to match the two, a "false reject" event occurs. The theoretical probability of this happening, is known as the *False Reject Rate* or *FRR*. A special case of the False Reject event occurs when, for any of a variety of reasons, the biometric system is unable to collect a useable sample of the biometric measurement. In the case of the eye iris recognition, the iris might be obscured by eyelids, eyelashes, contact lens, sunglasses, etc. [9]. In the fingerprint case, some or all of the object's fingers may be dried, abraded, specially for elderly and manual workers people, or otherwise injured in such away as to remove or damage the fingerprint and render it unusable. These events are often called a *"Failure To Acquire"* or *FTA* and lead to false rejection of an *authentication* of the individuals. Because they are caused by events that are very difficult to control or predict. There is always possibility that the measurement from a live subject will be sufficiently similar to a template from another, different person that a match will be (erroneously) declared. This type of error is called a "false accept" event and the associated probability is called the *False Accept Rate* or *FAR*. The FAR achieved by a particular biometric directly reflects the fundamental power and specificity of the technology. To achieve a low FAR the biometric entity measured must be absolutely unique to the individual, where the *fingerprint* and the *iris* can be ideal, and the algorithm used to measure the entity must capture the uniqueness in perfect and effectively. In AFIS the FAR is very low, and in verification system, where the sample is compared only to the template corresponding to the claimed identity, false accepts are very rare. But in identification system the sample is compared to all entries in a database, and the chance of falsely matching at least one of them will be much higher than the single match FAR. Some biometrics technologies, including fingerprint identification, mitigate the false accept limitation by reducing the size of the database to be searched using binding and filtering, and by combining multiple biometrics measurements. In combining multiple biometrics measurements using an "OR" logic rule increases false accepts, while combining using an "AND" logic rule increases false rejects. An important difference between the fingerprint and the others biometrics based identification systems is the size of the templates used to store the unique biometric features. Fingerprint may be stored and transmitted as compressed digital images, and require about 15 Kbytes of the storage in this format, [9]. Alternatively, we encoding fingerprint features as a series of minutiae, locations of ridge endings and bifurcations, and angles of the ridges at each location point.

6. Conclusion

One of the major problems in AFIS is the False Reject Rate (FRR). The false minutiae are created by different kinds of skin impression, such as manual worker people, elderly people, and other people who deal with some skin effecting tools. However, using some image enhancement techniques can reduce this problem.

In the FSOM Learning paradigm adopting a principle of learn according to how well it wins is proposed. Unlike the SOM Learning where only one neuron will win and learn at each competition, every neuron in the FSOM to a certain degree wins, depending on it is distance to the input pattern. Encoding the features information for matching purpose is much more flexible than dealing with other fingerprint image pre-processing steps.

References

[1] Adeli H. and Hung S. L. (1995), Machine Learning, Neural Networks, Genetic Algorithms and Fuzzy Systems.

[2] Adeli H. and Hung S. L., 1995, Machine Learning, Neural Networks, Genetic Algorithms and Fuzzy Systems.

[3] Alessandro F., Zsolt M., & Kovacs V., 1999, Fingerprint minutiae extraction from skeletonized binary images, The Journal of the Pattern Recognition Society, Vol. 32, PP. 877-889.

[4] Ammar H. H. & Miao Z., 1996, Performance of parallel algorithms for fingerprint image comparison system, Proceeding of the Parallel Proceeding Symposium, IPPS, PP. 410-413.

[5] M. B. Akhan, I. Emirroglu, and E. G. Bahari, 1999, A Flexible Fingerprint Identification System

[6] Anil J., Lin H. & Ruud B., 1997, On line fingerprint verification, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 19, No. 4, PP. 302-313.

[7] Anil J., Ruud B., and Sharath P., 1999, Biometrics Personal Identification in Networked Society, Kluwer Academic Publishers, The Kluwer International Series in Engineering and Computer Science.

[8] Anil K. J., Salil P., and Lin H., 1999, A Multi-channel Application to Fingerprint Classification by Directional Image Partitioning, IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol. 21, No. 4.

[9] James Cambier, 2000, Biometric Identification in Large Populations, The International Jouneral for IT Security professionals, Vol. 5, No. 2, PP.17-26.

[10] Baldi P. and Chauvin Y., 1993, Neural Networks for Fingerprint Recognition, Neural Computation, Vol. 5, No. 3, PP. 402-418.

Automatic Fingerprint Recognition System Using Fuzzy/Neural Networks Techniques

Suliman M Mohamed[1], Henry O Nyongesa[1], George Gagaudakis[2]

[1]Department of Information Systems and Computing

Brunel University, Uxbridge, Middx. UB8 3PH, UK

[2]Department of Computer Science

Cardiff University, Newport Road, PO Box 916, CF24 3XF

Cardiff Wales

e-mail:{cspgssm3, cssthkn, cspgggg2}@brunel.ac.uk

Abstract: The use of fingerprint for identification has been employed in law enforcement for about century as it's one of the most reliable personal verification methods, and now a day becomes one of the most strengths biometrics technologies. However, manual fingerprint identification/verification is so tedious, time-consuming, and expensive that it is incapable of meeting today's increasing performance requirements. An Automatic Fingerprint Recognition System (AFRS) is widely needed. This paper describes the steps of design and implementation of the fingerprint recognition system, which operate different classification approaches and strategies are discussed. In this regard we have investigated the limitations of current approaches and strategies, and attempt to overcome these limitations by using the strengths of the computational intelligence. In particular, investigation of fingerprint image clustering by using Fuzzy Self-Organizing Map Learning (FSOM), with the learning and the self-organizing features of artificial neural networks and the ability to process fuzzy data using fuzzy membership.

Keywords: fingerprints, biometrics, fingerprint acquisition, fingerprint enhancement/matching/classification, fuzzy neural learning.

1. INTRODUTION

Associating an identity with an individual is called personal identification. Biometrics identification deals with identification of individuals based on their biological or behavioral characteristics (so-called positive personal identification). As information becomes the key to wealth in the 21st century, biometrics security will play a central role in providing a high level of security to existing and future products. Fingerprint technology, is one of the most, mature biometrics technologies.

Fingerprints are graphical flow-like ridges present on human fingers. The fin image is made of foreground ridges which are separated by background ridge flow direction forms different patterns like arches, loops, wholes, a gives rise to various minutiae like ridge endings, ridge bifurcation's, cores etc. Both foreground and background consist of a similar set of minutiae, t patterns used for fingerprint classification. Each individual has a unique fin and the uniqueness of a fingerprint, is exclusively determined by the loca characteristics and their relationship. These local ridge characteristics are not distributed. Most of them depend heavily on the impression conditions and qu fingerprints and rarely observed in fingerprints. Police departments have lon interested in the improvement of AFRS methods as an important factor in r more effective the administration of criminal justice, security and record bı transactions. Although, fingerprint verification systems are usually associate criminal identification, and police work it has now become more popular in c applications. Such as access control, high-security areas in prominent organiz financial security, and verification of firearm purchasers, and driver l applicants. These useful applications have been conceived due to the fing favorable characteristics such as unchangeability and uniqueness in an indivi lifetime. Inherently, using current technology fingerprint identification is much reliable than the kinds of popular identification methods based on signature. iris, ear, hand geometry, and speech. With increasingly large volumes of finger being collected and stored, there is an urgent need to develop automatic finge recognition systems to improve the efficiency and reliability of pe. identification. Usually, fingerprint recognition is performed manuall professional fingerprint experts. However, manual fingerprint recognition tedious, time-consuming, and expensive that is does not meet the perforn requirements of the new applications. Various approaches for preprocessin Fingerprint recognition have been investigated for the purpose of auto fingerprint recognition. These can fall into either one of the following categ *structural, statistical, syntactic, geometric, mathematical, hybrid approaches artificial neural networks*. Furthermore, the robustness of the AFRS has evaluated over a manually constructed fingerprint. This paper will discuss the issues of automatic fingerprint image processing, including (fingerprint acquis enhancement, classification, matching, identification, and verification), and r the overview of fuzzy neural technique with. it's advantages to clustering fingerprint images.

2. Fingerprint Image Processing

Automatic fingerprint matching depends on the comparison of the local r characteristics and their relationships to make a personal identification. Becaus the large volume of fingerprints and recent advances in the computer technol there has been increasing interest in automatic processing of fingerprints. The most prominent local ridge characteristics, called minutiae, are

- Ridge ending and
- Ridge bifurcation.

bifurcation is defined as the point where ridge forks or diverges into [illegible] good quality fingerprint typically contains about 40-100 minutiae, (Lin H. and Yifei W., 1998). Figure 1. Shows sample of fingerprint image.

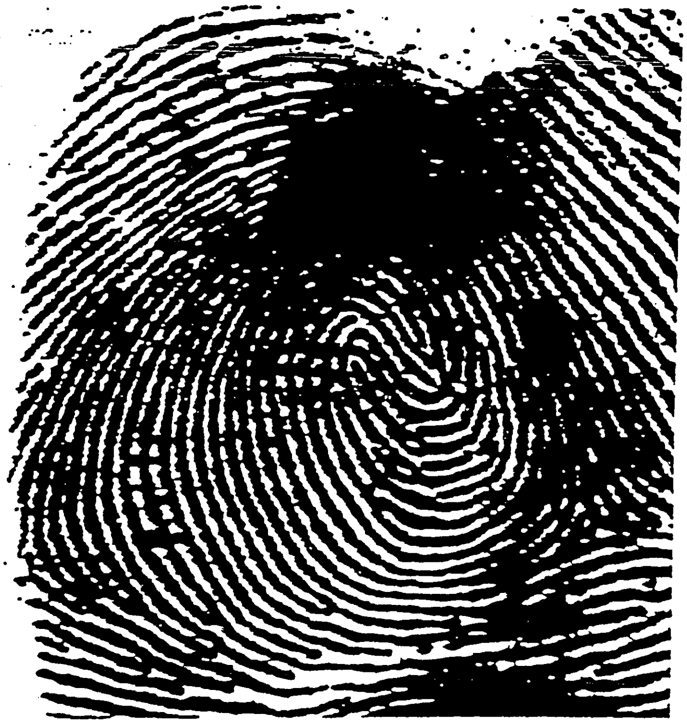


Figure 1. Sample of fingerprint image

2.1 Automatic Fingerprint Recognition

As a result of many studies, automatic fingerprint recognition systems are in great demand. Although, a significant progress has been made in designing automatic fingerprint recognition systems, over the past 25 years, a number of limitations in achieving the desired faster matching of the fingerprint image.

An automatic fingerprint recognition system is concerned with some or all of the following issues:

- Fingerprint Acquisition: How to acquire fingerprint images and how to present them in a proper format.
- Fingerprint Enhancement: To clear the quality of fingerprint images.
- Fingerprint Classification: To assign a given fingerprint to one of the pre-specified categories according to its geometric appearance.
- Fingerprint Matching: (verification/identification)

# Verification: To determine whether two fingerprints are from the same finger or not.

2.1.1 Fingerprint Acquisition

A number of methods are used to acquire fingerprints. Among them, the inked impression method remains the most popular. It has been essentially a standard technique for fingerprint acquisition for more than 100 years, (Battley H., 1937). The first step in capturing an inked impression of a fingerprint is to place a few dabs of ink on a slab then rolling it out smoothly with a roller until the slab is covered with a thin. Then the finger is rolled from one side of the nail to the other side over the inked slab, which inks the ridge patterns on top of the finger completely. After that the finger is rolled on a piece of paper so that the inked impression of the ridge pattern of the finger appears on the paper. Obviously, this method is time-consuming and unsuitable for an on-line fingerprint verification system. The second method is a more efficient and reliable optical data generation system. It consists of a prism and a uniform light beam that transforms the three-dimensional data into two-dimensional data, which can be photographed. The optical method of fingerprint data generation is not perfect either because the contrast and focus of the image obtained are sometimes poor. However, the method is clean, fast, and most of the problem can be overcome by good preprocessing techniques such as grayscale-to-binary conversion. Innovations in optical devices have been made recently as mid-1990s, an optical sensor was housed in a box about 6x3x6 inches. The third method is the ink-less fingerprint scanners are now available which are capable of directly acquiring fingerprints in digital form. This method eliminates the intermediate digitization process of inked fingerprint impressions and makes it possible to build an on-line system. The fourth method so-called solid-state sensors have appeared on the market recently. These are microchips containing a surface that images the fingerprint via one of the several technologies, including electrical measurements and temperature sensitive sensors. One of the most important factors that will decide when fingerprint verification will be commercially successful in the large-volume personal verification market are low cost and compact size.

2.1.2 Fingerprint Enhancement

In practice, due to variations in impression conditions, ridge configuration, skin conditions (aberrant formations of epidermal ridges, postnatal marks, occupational marks), acquisition devices, and non-cooperative attitude of subjects, a significant percentage of acquired fingerprint images is poor of quality. The ridge structures in poor-quality fingerprint images are not always well defined hence, they can not be correctly detected. In order to ensure that the performance of the minutiae extraction algorithm will be robust with respect to the quality of input fingerprint images, an enhancement that can improve the clarity of the ridge structures is necessary, (D. C. Douglas, 1993). A fingerprint is often able to correctly identify the minutiae by using various visual clues such as local ridge orientation, ridge continuity, ridge tendency, as long as the ridge are not corrupted completely. It is possible to develop an enhancement algorithm that exploits these visual clues to improve the clarity of ridge structures in corrupted fingerprint images.

(b)

(c)

Enhancement: Automatically enhancing fingerprint images introducing artifacts is a challenging problem: (a) poor quality ·ints, (b) and (c) result of image enhancement of fingerprint 'iown in (a).

objective of a fingerprint enhancement algorithm is to improve the clarity ructures of input fingerprint images to facilitate the extraction of ridge and a fingerprint enhancement algorithm should not result in any spurious :tures. This is very important because spurious ridge structure may change luality of input fingerprints.

ngerprint Classification and Matching

: classification, fingerprint matching (verification or identification), are sks of fingerprint image processing.

ıgerprint Classification

. f fingerprint classification is to assign a given fingerprint to a specific cording to its geometric properties. In general there are six classes:

Tented arch (iii) Right loop (iv) Left loop (v) Twin loop and (vi) whorl.

ɔurpose of fingerprint classification is to facilitate the management of :print databases and to speedup the process of fingerprint matching. manual fingerprint classification is performed within a specific such as well-known Henry system. Different frameworks use different perties. However, no matter what type of framework is used, the n is based on ridge patterns, local ridge orientations and minutiae.

Therefore, if these properties can be described quantitatively and extracted automatically from a fingerprint image then fingerprint classification will become an easier task.

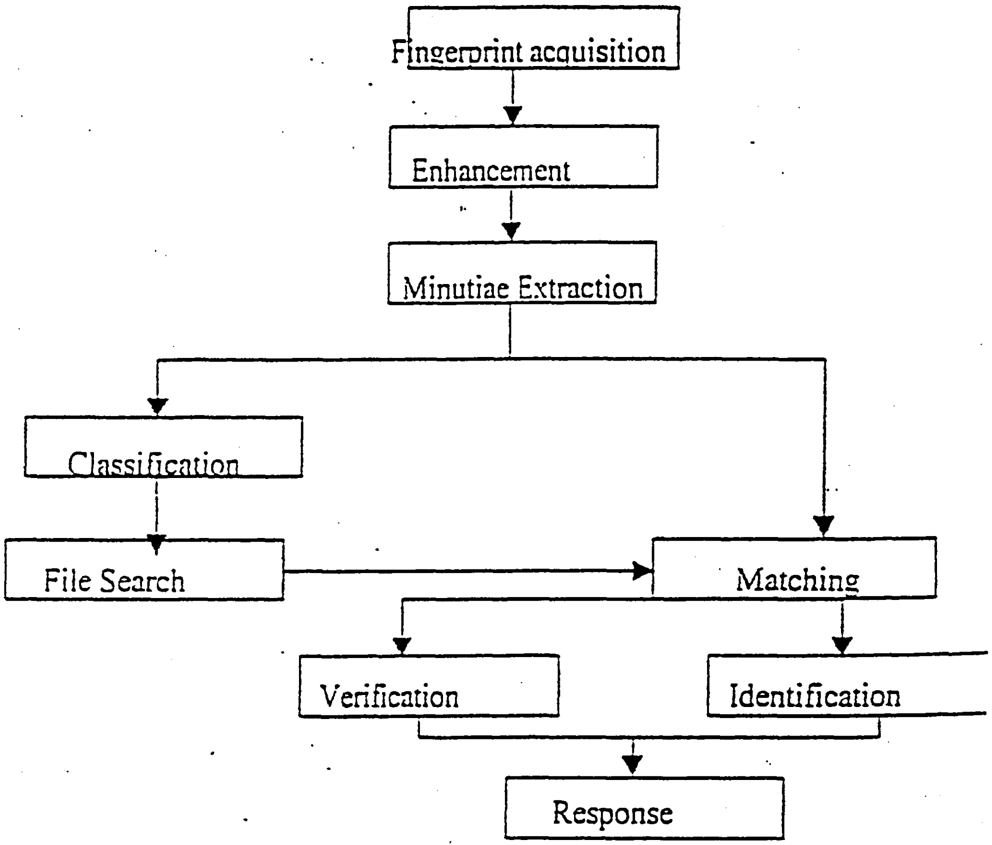


Figure 3. Steps taken by a general-purpose of fingerprint matcher

2.3.2 Fingerprint Matching: *Verification and Identification*

Fingerprint matching determines whether two fingerprints are from the same finger or not (fingerprint verification), or search given fingerprint in database of template (fingerprint identification). It is widely believed that if two fingerprints are from the same source, then their local ridge structures (minutiae details) match each other topologically. Matching can be separated into two categories: verification and identification, (Lawrence O. G., 1999). *Verification* is the comparison of a claimant fingerprint against an enrollee fingerprint, where the intention is that the claimant fingerprint matches the enrollee fingerprint. To prepare for verification, a person initially enrolls his her fingerprint into the verification system. A representation of that fingerprint is stored in some compressed format along with the person's name or him or herself, then applying the fingerp

be verified. Verification is also known as, *one-to-one matching*. *Identification* is the traditional domain of criminal fingerprint matching. A fingerprint of unknown ownership is matched against a database of known fingerprints to associate a crime with an identity. Identification is also known as, *one-to-many matching*.

A number of different types of local ridge descriptions have been identified. The two most prominent structures are ridge endings and ridge bifurcation's which are usually called minutiae. Fig. 4 shows examples of ridge endings and ridge bifurcations.

Based on this observation and by representing the minutiae as a point pattern, an automatic fingerprint verification/identification problem may be reduced to a point pattern matching (minutiae matching) problem. In the ideal case, if

ie correspondences between the template and input fingerprint are known,

here are no deformations such as translation, rotation and nonlinear deformations, een them, and

each minutia present in a fingerprint image is exactly localized, then fingerprint .cation/identification consists of the trivial task of counting the number of spatially hing pairs between the two images.

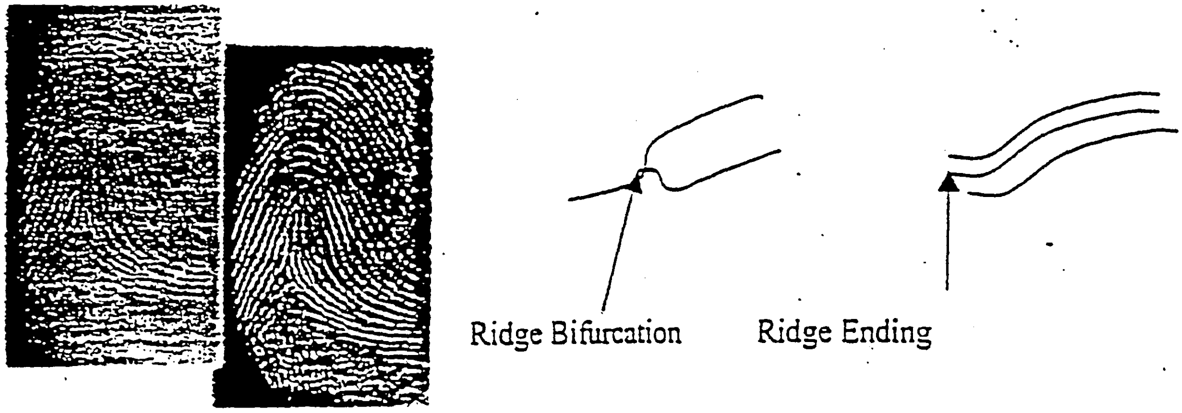


Figure 4. Ridge ending and ridge bifurcation

3. Fuzzy Neural Approach

One of the characteristics of artificial neural networks (ANNs) is that they can classify inputs. This is useful if plasticity is maintained to be, that is, the ANNs can continuously classify and also update classifications. We have also seen the stability of ANNs and how robust when inputs become less defined (i.e., fuzzy inputs). In addition, we have seen that fuzzy systems deal with current fuzzy information and are capable of providing crisp outputs. Fuzzy logic is one of the key technologies for representing human knowledge in the brain and for constructing adaptive systems. However, in fuzzy systems there are no learning and, even vaguely, the input-output relationships the fuzzy rules must been known a priori, neural networks and fuzzy systems each have their own limitations. When one designs with neural networks alone, the network is a black box that needs to be defined, (Zhang D. M. and Elmasry M. I., 1993). This is a highly compute-intensive process. One must develop a good sense, after extensive experimentation and practice, of the complexity of the network and the learning algorithm to be used and of the degree of accuracy acceptable by the application. On the other hand fuzzy systems, require a thorough understanding of the fuzzy variables and membership functions, of the input-output relationships as well as the good judgment to select the fuzzy rules that contribute the most to the solution of the application. A large number of rules, and many may not contribute significantly to the problem. Hence, good judgment is needed to eliminate unnecessary rules. As the tide of using neural network and fuzzy logic grows up a number of studies have shown that although, neural networks are powerful in machine learning, associative memory, and parallel processing but they fails to do well in some symbol processing and indefinite reasoning. On the contrary, the fuzzy logic systems are powerful in indefinite reasoning and symbol processing but they fail to do well in associative memory, (Adeli H. and Hung S. L., 1995). So it is supposed to use the synthetic method of these two techniques which can be a complement to each other, so as to use the fusion of these two methods in fingerprint image processing.

In the fuzzy artificial neural network the neural network part is primarily used for its learning and classification and retrieval. The neural network part automatically generates fuzzy logic rules and membership functions during the training period. In addition even after training, the neural networks keeps updating the membership functions and fuzzy logic rules as it learns more and more from its input signals. Fuzzy logic, on the other hand, used to infer and provide a crisp or defuzzified output when fuzzy parameters exist. In this regard, we attempt to combine Fuzzy Logic and Kohonen's Self-organized Map, have been proposed. Our algorithm investigated the fuzzy-neural models by integrating fuzzy membership function and the self-organizing map network learning techniques, for purpose of image clustering. In particular, we have investigated the combination of features of neural networks (with learning ability, self-organizing and high-speed parallel structure) and fuzzy systems (with ability to process fuzzy information using fuzzy membership) to form a Fuzzy Self-Organizing Map Networks Learning (FSOML), which can learn from environments.

3.1 An Overview of Fuzzy Self-Organizing Map Approach

By considering win as a fuzzy function, every neuron in the FSOM to a certain degree wins, depending on its distance to the current training pattern. In this way, the learning according to how well it wins paradigm results. The clustering version of the SOM combines conventional Kohonen's learning with fuzzy membership generalization toward image processing.

For our problem of fingerprint image processing, the input pattern vector

$X = XI, X2, ..., Xn$

Each Xi is a pixel value from the image, can be used as input vector to the FSOM network.

Given a number of input pattern vectors x(1), x(2), ..., x(p), our objective is to divide them into several clusters with each cluster comprising similar pattern vectors X, with membership function of belongingness.

3.2 FSOM classification technique consist of two stages as follows:

1- The SOM classification process. In this regards the SOM classify the set of training instance into a set of clusters and the values of the mean vector (prototype) for each cluster are stored in the weight associated with the links between inputs and output nodes.

2- The fuzzification process which fuzzy membership values for each training instance in the set of supports, classified clusters, are evaluated.

Why FSOM?

For the SOM the number of input nodes is equal to the number of input vector for each training instance, number of output nodes is equal to the number of clusters, and is determined through the classification process. The topology of NN is changed and self-organizing during the classification process. After the process of classification is completed classified clusters may be disjoint or partly overlapping and based on the learning only if it wins, where we can apply fuzzy membership.

4. Conclusion and Discussion

One major problem in the automatic fingerprint recognition is the quality of the original print. If the quality is not of an acceptable standard, automatic fingerprint identification becomes extremely difficult. The reason for this is that normal methods of fingerprint recognition use the small unique features (known as minutiae) in the fingerprint pattern to identify the fingerprint. However, it is extremely difficult to extract these minutiae from the fingerprint image if the quality of the print is not perfect. Problems also exist in the extracting these minutiae from the fingers of elderly people as well as manual labourers. The problem with elderly people's prints is that the prominence of the ridges diminishes, with result that the fingerprint pattern is not very clear. Manual workers (laborers) have the same problem that the skin on the hands is subject to severe punishment, with the result that false minutiae are created by cuts in the skin and some cases the ridges are worn away.

- In solving fingerprint image processing using NN thousands of training instance are required, SOM will, applied for the purpose of feature abstraction.
- Fuzzy membership function can apply to represent the ambiguous relationship between training instance and clusters.

- A FSOML paradigm adopting a principle of learn according to how well it wins is proposed, unlike the SOM where only one neuron will win and learn at each competition, every neuron in the FSOML to a certain degree wins, depending on it is distance to the input pattern.

References

1. Adeli H. and Hung S. L. (1995), Machine Learning, Neural Networks, Genetic Algorithms and Fuzzy Systems.
2. Adeli H. and Hung S. L., 1995, Machine Learning, Neural Networks, Genetic Algorithms and Fuzzy Systems.
3. Alessandro F., Zsolt M., & Kovacs V., 1999, Fingerprint minutiae extraction from skeletonized binary images, The Journal of the Pattern Recognition Society, Vol. 32, PP. 877-889.
4. Ammar H. H. & Miao Z., 1996, Performance of parallel algorithms for fingerprint image comparison system, Proceeding of the Parallel Proceeding Symposium, IPPS, PP. 410-413.
5. Ammar H. H., Zeng S. & Miao Z., 1998, Parallel Processing & Fingerprint Image Comparison. International Journal Of Modelling & Simulation, Vol. 18, No. 2, Pp.P 85-99.
6. Anil J., Lin H. & Ruud B., 1997, On line fingerprint verification, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 19, No. 4, PP. 302-313.
7. Anil J., Ruud B., and Sharath P., 1999, Biometrics Personal Identification in Networked Society, Kluwer Academic Publishers, The Kluwer International Series in Engineering and Computer Science.
8. Anil K. J., Salil P., and Lin H., 1999, A Multi-channel Application to Fingerprint Classification by Directional Image Partitioning, IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol. 21, No. 4.
9. Baldi P. and Chauvin Y., 1993, Neural Networks for Fingerprint Recognition, Neural Computation, Vol. 5, No. 3, PP. 402-418.
10. Basak J., Pal N. R. and Patel P. S., 1997, Thinning in Binary and Gray Images: A Connectionist Approach, J. of the Institution of ETEE, Vol. 42, No. 4-5, PP. 305-313.
11. Battley H., 1937, A New and Practical Method of Classifying and Filing Single Fingerprints and Fragmentary Impressions, New Haven: Yale University Press.
12. Berfanger D. M. and George N., 1999, All-Digital Ring-Wedge Detector Applied to Fingerprint Recognition, Applied Optics, Vol. 38, No. 2, PP. 357-369.
13. Blue, J. L., Canfela, G. T., Grother P. J., Chellappa R. and Wilson C. L., (1994), Evaluation of Pattern Classifiers SFOR Fingerprint and OCR Applications, Pattern Recognition, Vol. 27, No. 4, PP. 485-501.
14. Chen Y., Sheng M., & Yongbao H. E., 1992, A Method Of Pattern Recognition Based Upon Synthetic Technology Of Fuzzy Logic & Neural Network, (Department of Computer Science, Fudan University Shanghai 200433 P. R. CHINA.
15. Cheung Y. S. & Yip W. M., 1987, A Personal Computer-Based Fingerprint Identification System, Proceeding IEEE Asian Electronic Conference, Hong Kong, PP. 290-294.

Image Pattern Recognition Using Fuzzy Self-Organizing Map Network Learning

Suliman M Mohamed and Henry O Nyongesa
Department of Information Systems and Computing
Brunel University, UK

Abstract: In this paper, a learning algorithm for fuzzy self-organising map (SOM) network, is discussed in the domain of image processing. Unsupervised classification algorithms are often based on the concepts of data clustering and feature abstraction. However, unlike the crisp SOM learning algorithm where only one neuron will win and learn at each competition, every neuron in this technique wins to a certain degree, depending on a distance measure to the input pattern. Thus, the concept of win is formulated as a fuzzy relation. Furthermore, the self-organising network is combined with a clustering network. The technique is applied to image filtering and compression.

1. Introduction

In real-world image analysis, the input dimensionality can be very high order and the discriminate functions to approximate are nonlinear and complex. A classifier based on the measured objects (i.e. images) directly would require a large number of parameters in order to approximate and generalize well all over the input domain. Images of real scenes very frequently contain data, which is incomplete and ambiguous. In this case, fuzzy interpretations of data can be a natural and intuitively plausible way to formulate and solve problems in image analysis.

Conventional self-organising feature map (SOM) classification algorithms are often based on data clustering and feature abstraction. A best matching processor (the winner processor) is found and the weight vectors of this processor and its topological neighbors are adjusted accordingly. The process is repeated for all the inputs and several iterations are performed until the weights converge, resulting in output clusters. Since these crisp clusters provide output from only one neuron, which may, however, overlap or is not completely disjoint with other clusters fuzzy membership values can be used to represent the relationship among the given training instances.

The RBF network can be used for classification or function approximation, using two distinctive layers. The first layer is composed of adaptable basis functions, which are usually Gaussians. The location and width of the basis functions are adapted so that they cover the input space. The output of these basis functions becomes your new representation of the input, which is input to a simple supervised network. Typically, a RBF network performs a clustering of the input. Inputs that are similar will "fire" similar basis functions, producing very similar inputs. This feature is exploited in our algorithm to create the fuzzified SOM algorithm in which instead of learning being applied to topological neighbours in the classic Kohonen framework, it can be based on general fuzzy functions.

There are a number of previous studies that have addressed fuzzy clustering approaches in image analysis. Most of these have described fuzzy and neural models which were successfully used for applications in character recognition and image classification[1,2]. This paper investigates the issue of image pre-processing and analysis, which includes compression, feature extraction and indexing[3]. The method is a novel combinatipon of the Kohonen self-organising map and clustering based on radial basis functions.

2. Kohonen's Self-Organising Feature Map Networks

The self-organizing feature map is a competitive network, which learns from the environment without the aid of a teacher, based on a concept data clustering or feature abstraction. The objective of the clustering process is to classify a given training set into a certain number of homogeneous clusters or classes relying on regularities in the training data. The unsupervised feature extraction scheme is especially suitable for general image analysis in computer vision, since it is fairly inexpensive to collect large amounts of data to be used in training, as long as the images need no manual analysis and classification.

In SOM, the input feature vector X = $(X_1,X_2,...,X_n)$ is mapped onto an output Y= $(Y_1,Y_2,...,Y_m)$, through an adjustable weight vector W^{nxm}. The output unit with a weight vector closest to the input pattern wins the competition and responds maximally driving all other units to zero output. The competitive action is implemented through lateral (fixed-weight) connections between neighboring units where both excitations and inhibitions are generated. Thus, the winning unit shares the learning experience with its closest neighbors and the learning process is executed in such a way that nearby elements tend to align their weights in the same direction as the input pattern while more distant units have their weights aligned in opposing directions[5]. Given a number of input pattern vectors $X(1), X(2),...,X(p)$, the objective is to divide them into several clusters with each cluster comprising similar pattern vectors.

For the sake of clarity, a general SOM algorithm is described below:

Step 1. Initialization:

-Set the number of competing neurons, *m*.

-Initialize the network weight vector, W.

Step 2. Distance computation:

-For input pattern X_i compute distances

$D_j = ||X_i - W_i||$ for all competing neurons j.

Step 3. Competition:

-Determine the winning neuron J having $D_{iJ} = \arg\min [D_{ij}], \forall_j$.

Step 4. Learning:

-Update the winning neuron's weight vector as,

$W_{ij}(t+1)=W_{ij}(t)+ \alpha(t)[X_i-W_{ij}(t)]$

Where $\alpha(t)$, is the learning rate that is usually monotonically decreasing.

Step 5. Termination:

-Repeat steps 2-4 until the terminating criterion.

3. Radial Basis Function (RBF) Networks

The fundamental principle of operation of RBF is a fixed non-linear mapping of the input, followed by a linear adjustable output mapping [4]. The standard architecture of RBF is that of a conventional three-layer feedforward network. The processing layer, however, is comprised of so-called *basis function* units, such that they only respond to locally tuned regions of the input space. These units compute a distance measure between the basis function and the input vector, and their output is a function of these distances.

The critical choices in application of RBF networks are the location and width of the basis functions, which must be adapted so that they appropriately cover the input space. Since the response of the RBF networks, they have a tendency to generalize better. On the other hand, the number of basis functions increases exponentially with the dimensionality of the input. In addition, since the RBF are essentially clustering the data, RBF will perform better if the input data can be naturally clustered into regions.

4. Clustering SOM Learning Algorithm

By considering win as a fuzzy function, every neuron in the SOM network to a certain degree wins, depending on its distance to the current training pattern. In this way, a learn according to how well it wins learning paradigm results. The clustering version of the SOM combines conventional Kohonen learning with RBF (radial basis function) clustering. The advantage of the derived technique is faster convergence and better generalisation.

The RBF SOM learning algorithm only differs from the conventional SOM algorithm in so far as the computation of the distance metric between the inputs and the weight vectors. Thus:

Step 3. Fuzzy competition:

-Based on a distance metric, determine the degree of similarity μ_j between the input vector and the weight vector for each neuron.

Step 4. Fuzzy learning:

-Update each competing neuron's weight vector as

$W_{ij}(t+1) = W_{ij}(t) + \alpha(t)\lambda_j(t)[X_i - W_{ij}(t)]$

where $\alpha(t)$ is the learning rate and $\lambda_j(t)$ is a function of μ_j.

We applied RBFSOM to image filtering and compression. In this method we are interested in removing background noise and preserving the main spatial features of the image after compression. This is necessary in order to reduce the dimensionality of an image analysis task. The degree of compression determines the network size, and in this case has been predefined. There are 3 parameters for each pixel in an image, its x-y coordinates and the Greyscale intensity. The 704x480 images were compressed to 88x60. The neural network thus, is comprised of 3 input nodes and 88x60 processing nodes. Thus, the weight vector comprises fixed links connecting the x-y inputs and adjustable links connecting the intensity input to processing layer. This is illustrated in Figure 1.

Based on the above specification and the network architecture, learning proceeds as follows.

- A random pixel is selected from the image.
- For each of the processing nodes, the proximity, μ_{xyj} between the pixel coordinates and node's receptive region is determined using fixed basis functions.
- The mean, I and standard deviation, σ_i of the intensity of each neuron are dynamically adjusted in proportion to μ_{xyj}:

$$I(t+1) = \frac{I(t).n + i(t)}{n+1}$$

$$\sigma_i^2(t+1) = \frac{\sigma_i^2(t).n + \mu_{xy}.(I(t) - i(t))^2}{n+1}$$

- The learning rate is modified by,

$$\lambda_j = \min(\mu_{xyj}, \exp(-\frac{(I-i)^2}{\sigma^2}))$$

An example of the application of FSOM learning to image filtering and compression is shown in Figure 2.

5. Conclusion

We have combined RBF clustering and SOM competitive learning in a network suitable for an image processing task. Through the fuzzy competitive learning scheme it is possible to identify and preserve overlapping characteristics in input data, while RBF clustering enables faster convergence in learning. The main feature and advantage of this technique is the fast convergence and better input generalisation.

6. References

1. T. Yamakawa and S. Tomoda, "A fuzzy neuron and its applications to pattern recognition," in Proc. Third Int. Fuzzy Systems Associations (IFSA) Congress, Seattle, 1989, pp30-38.

2. J. H. Chian and P. D. Gader, "Hybrid fuzzy-neural systems in handwritten word recognition", IEEE Trans. on Fuzzy Systems, Vol. 5, No. 4, 1997.

3. J. Jiang, "An image compression and indexing system using neural networks", J. Visual Communication and Image Representation, vol. 8, no. 2, 1997, pp135-145.

4. D. A. Linkens and H. O. Nyongesa, "Learning Systems in Intelligent Control: an appraisal of fuzzy, neural and genetic algorithms control applications", IEE Proc. Pt D, vol. 143, no. 4, 1996, pp367-386.

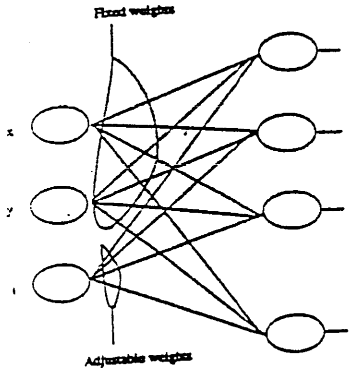
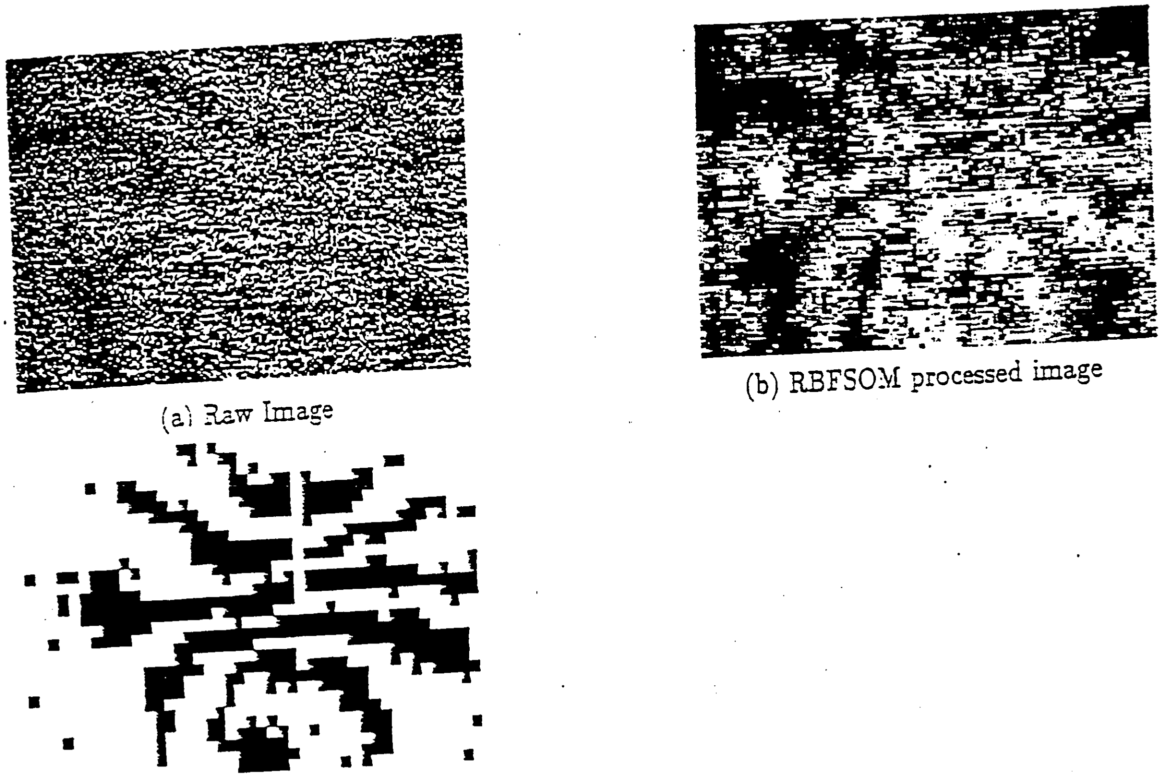


Figure 1: FSOM Neural Network



(a) Raw Image

(b) RBFSOM processed image

(c) Median filtered threshold image

Figure 2: Identifying patterns in an image