A Fingerprint Verification System Using Minutiae and Wavelet Based Features

By

Umair Mateen Khan

2005-NUST- MS PhD- CSE-22
Submitted to the Department of Computer Engineering
in fulfillment of the requirements for the degree of

Masters of Sciences
in
Computer Software Engineering

Thesis Supervisor

Dr. Shoab Ahmed Khan

MS-06

College of Electrical & Mechanical Engineering

National University of Science and Technology

2009

ACKNOWLEDGMENTS

There is no success without the will of God. I am grateful to Almighty, for showing me the path to success in this work and ever before. I owe my gratitude to all who have made this thesis possible and because of whom my experience has been one that I will cherish forever.

My special thanks to my supervisor Lt. Col. Dr. Shoab Ahmed Khan for his guidance and technical support in the development of the project. It has been a pleasure to work with and learn from him.

I would also like to thank Lt. Col. Dr.Alamas Anjum for the knowledge he imparted in the course of ‘Digital image processing’, which prompts me to work on this interesting area. I also want to express my sincere thanks to Dr. Shaleeza, Lt. Col. Dr. Farooq-e-Azam and Dr. Aasia Khanum for their advices and tremendous support. I also thank all of them for sparing their invaluable time reviewing my work. All of them also deserve thanks for their acceptance to become the members of guidance and evaluation committee of this thesis and for sparing their invaluable time reviewing the manuscript.

I also thank the lab administrators of College of EME, Rawalpindi for their support in required software installations in the lab. I also thank my colleague Mr. Khurram Yasin for his guidance and support throughout the work. I also thank my other colleagues for their moral support throughout the project.

I owe my deepest thanks to my family - my mother and father who have always stood by me and guided me through my career, and have pulled me through against impossible odds at times. Words cannot express the gratitude I owe them.

It is impossible to remember all, and I apologize to those I've inadvertently left out.

I would like to dedicate this thesis work to

my teachers, family and friends.

ABSTRACT

Fingerprint matching is an important and challenging research area of Digital Image Processing. Now a day's, every country of the world is much more concerned about their safety and security concerns than ever. That's where a fingerprint verification system helps in forbidding unauthorized access to different facilities, and in situations of any breach it also helps in tracing the intruders.

Minutiae based approach is one of the most famous technique for fingerprints matching. Minutiae are actually the features that are attained by the ridge discontinuities. Ridge endings and bifurcation are most commonly used minutiae types and researchers have used them in many flavors by using their attributes like Minutia type, minutia coordinates, Distance between minutiae, Ridges count between minutiae, Direction and relative angles etc. The performance of the minutiae based classification depends on the strength of these features. Due to the noise or corruption in the image integrity of these features reduces. Therefore it is required that the features extracted from these minutiae should be robust enough in such a way that it can minimize the effects of problem caused by some kind of noise and should have features from multiple domains.

The purpose of this research revolves around the fact described in the previous passage. Efforts are made to extract a very rich feature sets that covers not only features from spatial domain but also some features are extracted from frequency domain after applying different wavelets. And remarkably for classification using features that are actually attained by fusing both types of features really helped and certainly improved system performance. The system was tested on standard fingerprint database and very good results are obtained.

TABLE OF CONTENTS

List of FIGURES

List of TABLES

Chapter 1

INTRODUCTION

1.1 Biometrics

Biometrics means identifying any individual based on his/her physiological or behavioral characteristics (identifiers). To make a positive personal identification it relies on "something which you are or you do". On the other the hand there are other identification techniques like Knowledge-based technique and token-based technique.

Knowledge-based technique uses "something that you know" to make a personal identification. In this technique individuals are identified by showing some information or knowledge, which only they themselves know, such as password and personal identification number (PIN). While in token-based approach Individuals are identified by showing something or a token that is in their possession, such as passport, driver's license, ID card, credit card, and keys. The physiological or behavioral features are more reliable and more capable than knowledge-based and token based techniques in distinguishing between an authorized person and an impostor, because these are unique to every person. Also, the person to be identified is required to be physically present at the point-of-identification thus adding extra level of security. Moreover although these techniques have major advantages over other that are

a) These are very simple techniques and
b) They can be easily integrated into different systems with a low cost

However, since these conventional techniques do not have any inherent attributes of an individual to make a personal identification, they also have a number of disadvantages when comparing with biometrics: tokens may be lost, stolen, forgotten, or misplaced; PIN may be forgotten or guessed by the impostors. Therefore, they are unable to satisfy the security requirements of our modern and much more complex electronically inter-connected information society.

Hence biometrics is the only thing left that provides a high-quality solution for the security requirements society and would be the leading automatic personal identification standard in the near future.

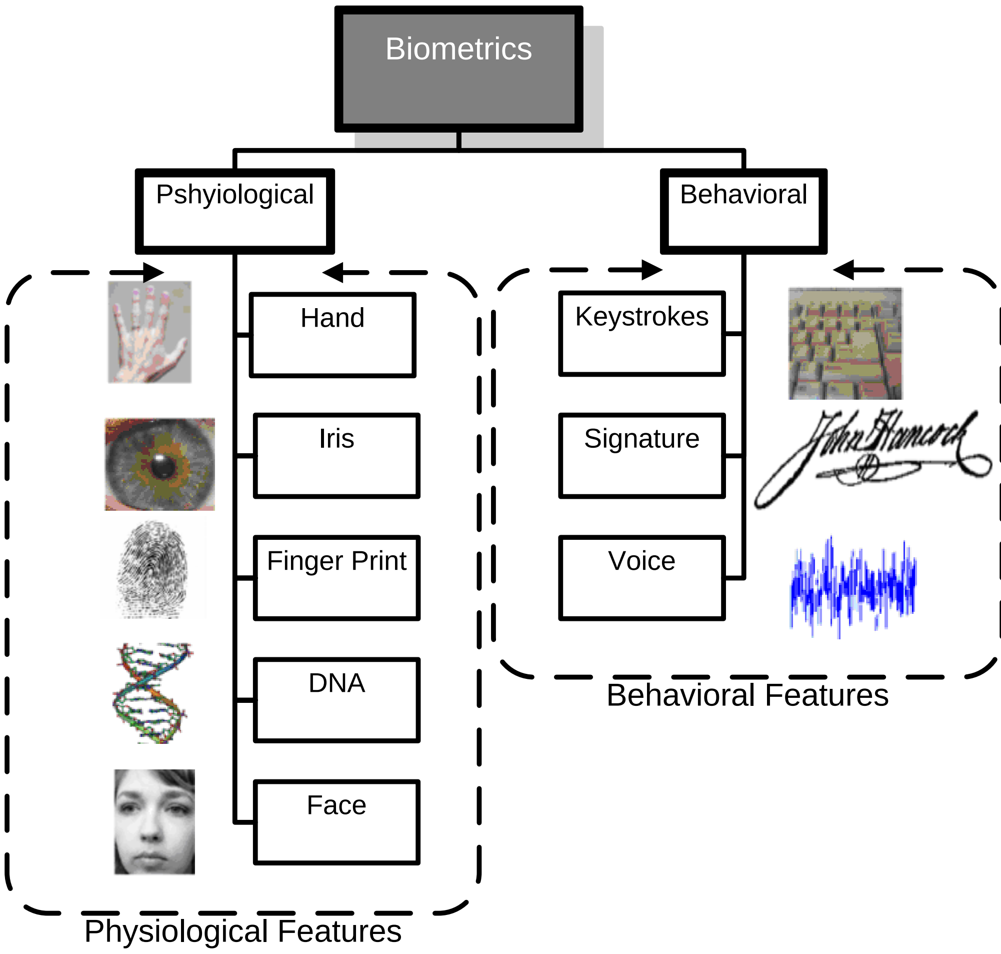


Figure 1.1: Different types of biometrics

So from the above diagram it can be deduced that any human physiological or behavioral characteristic can be used as a biometric-characteristic or identifier to make a personal identification as long as it satisfies the following requirements as described by [12]:

a) **Universality** means that each person should have that characteristic.
b) **Uniqueness** shows that no two persons should have same characteristics.
c) **Permanence** means that these characteristics should not be changeable and are permanent.
d) **Quantitative** indicates that the characteristic can be measured quantitatively.

All above mentioned requirements serves as preconditions of being a biometric feature. However, in practice, it is quite possible that any biometric characteristic that satisfies all the above requirements may not always be feasible for a practical biometric system. In a practical biometric system, there are a number of other issues that should be considered, that are:

a) **Performance** refers that whether using this feature it is achievable to have high accuracy, speed, robustness and whether it is possible to fulfill the resource requirements to achieve the desired identification accuracy and speed, as well as operational or environmental factors that affect the identification accuracy and speed.

b) **Acceptability** indicates that to how much extent people are willing to accept this biometric identifier in their daily life.

c) **Circumvention** reflects how easy it is to fool the system by deceptive methods.

Biometric techniques mentioned in Figure 1.1, to some extent satisfy the criteria mentioned above and have been used in practical systems or have the potential to become a valid biometric technique. Table 1.1 [10] compares these biometrics techniques to the degree they coincide with the requirement of a biometrics system, discussed above. Every biometric technique is given a rating against each criterion. This rating is assigned as Low (L), Medium (M) and High (H) value.

Biometrics Identifiers	Universality	Uniqueness	Permanence	Collectability	Performance	Acceptability	Circumvention
Face	H	L	M	H	L	H	L
Fingerprint	M	H	H	M	H	M	H
Hand geometry	M	M	M	H	M	M	M
Keystrokes	L	L	L	M	L	M	M
Hand veins	M	M	M	M	M	M	H
Iris	H	H	H	M	H	L	H
Retinal scan	H	H	M	L	H	L	H
Signature	L	L	L	H	L	H	L
Voice	M	L	L	M	L	H	L
Facial thermograph	H	H	L	H	M	H	H
Odor	H	H	H	L	L	M	L
DNA	H	H	H	L	H	L	L
Gait	M	L	L	H	L	H	M
Ear Canal	M	M	H	M	M	H	M

Table 1.1: Comparison of biometric technologies. High, Medium, and Low (High=H, Medium=M, and Low=L)

It is quite clear from the table that fingerprint as a biometric feature has a very good balance of all the desirable properties. As every human being have fingerprints with the omission of any disabilities related to hand. Every fingerprint has distinctive features like the fingerprint details are permanent, even if they may slightly change due to cuts and marks on the skin or any weather conditions. As for as the problem of capturing a fingerprint is concerned, live-scan fingerprint sensors can easily capture high-quality images and they do not suffer from the problem of segmentation of the fingerprint from the background.

So unlike other biometric features, fingerprint recognition is one of the most mature biometric technologies and is suitable for a large number of recognition applications. The fingerprint as a biometric feature has a lions share when taking about the percentage of its usage in the market among other biometric technologies. A comparing of various biometric technologies is made by [46] and is shown in figure 1.2.

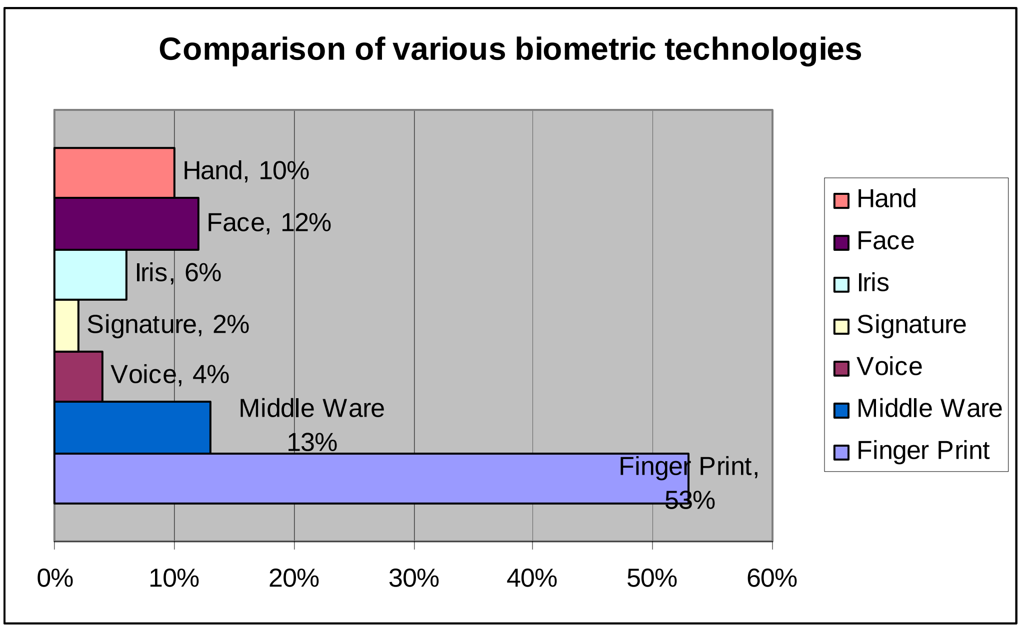


Figure 1.2: Percentage usage of different biometric technologies

1.2 Biometric System

Every biometric system is actually a pattern recognition system, which tries to identify any individual by determining the authenticity of a specific physiological or behavioral characteristic own by that user. A biometric system can be decomposed into two main modules:

a) The Enrollment Module
b) The Identification Module

The enrollment module is responsible for enrolling or signing up individuals into the biometric system. During this phase, a device that is called a biometric reader produces a raw digital representation of the characteristic by scanning the biometric features of an individual. Now, in order to facilitate matching, this raw digital representation is usually further processed by a feature extractor to generate a compact but meaningful representation, called a template. The template is either stored in the central database of the biometric system or be recorded on a magnetic card or smart card issued to the individual. During this 2nd phase which is identification module, the biometric reader captures the characteristic of the individual to be identified and converts it to a digital format, which is also processed by feature extractor to produce the same representation. The resulting features are not stored instead they are given to matcher that directly compares them to the features of the template(s) and tries to find any identity.

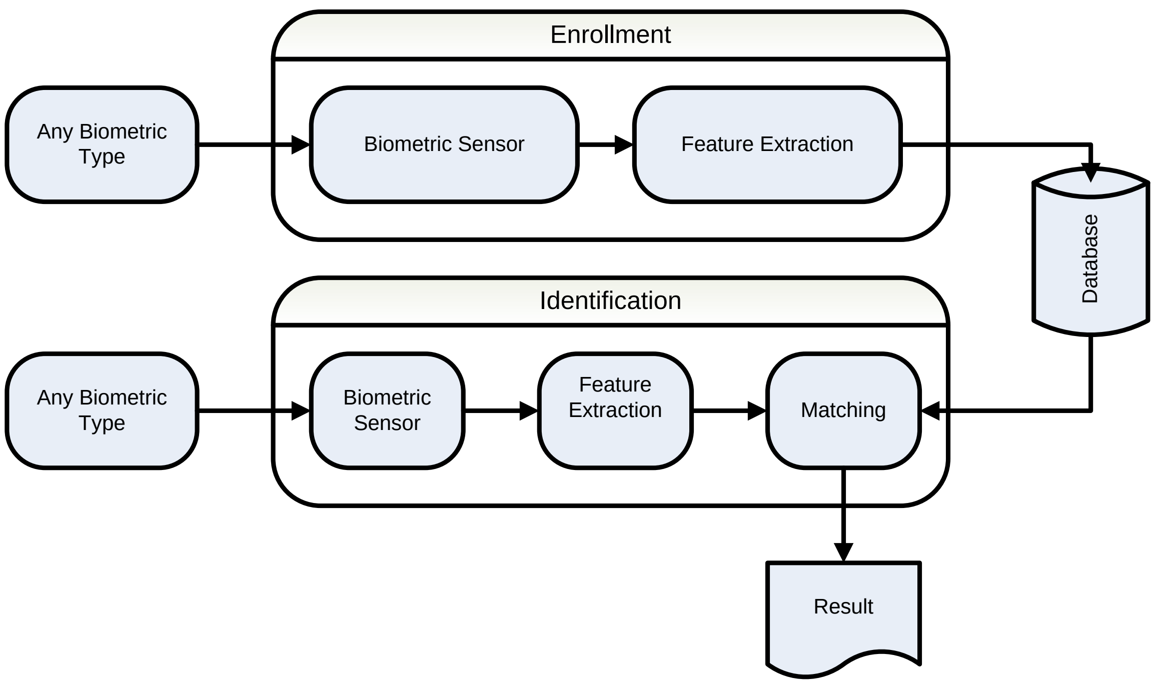


Figure 1.3: Operation of a biometric system

Now in case of fingerprints, fingerprint identification or verification is to verify the authenticity of one person by his fingerprint. The user provides his/her fingerprint together with his identity information like his PIN. The fingerprint identification system retrieves the fingerprint template according to the PIN number and matches the template with the real-time acquired fingerprint from the user. Usually it is the main design principle of AFAS (Automatic Fingerprint Authentication System).

1.3 Finger Print Overview

A fingerprint is the pattern of ridges and furrows or valleys on the surface of a fingertip. It is formed by the heap of dead cornfield cells that constantly slouch as scales from the exposed surface. Extensive studies have been conducted on fingerprints and fingerprint identification. Humans have used fingerprints for personal identification for centuries and in fact, fingerprint technology is so common in personal identification that it has almost become the synonym of biometrics.

Figure 1.4: Fingerprint image acquired by an Optical Sensor

The patterns on fingerprint surface that are ridges and valleys help us to grip objects by providing certain amount of friction. These patterns exhibit very rich structural information when examined as an image which can be called as features. The features that are gathered from a fingerprint are categorized as global and local features. The global features that a fingerprint has are orientation of the ridge, spacing between ridges and other singular points such as core and delta. While the local features are marked by ridge discontinuities and are called Minutiae points. There are about 18 distinct types of minutiae some of them are ridge endings, bifurcations, crossovers and islands and are shown in Figure 1.5. Ridge endings and bifurcation are the commonly used features. A ridge ending occurs when the ridge flow abruptly terminates and a ridge bifurcation is marked by a fork in the ridge flow. Most matching algorithms do not even differentiate between these two types since they can easily get exchanged under different pressures during acquisition. Global features do not have sufficient discriminative power on their own and are therefore used for binning or classification before the extraction of the local minutiae features.

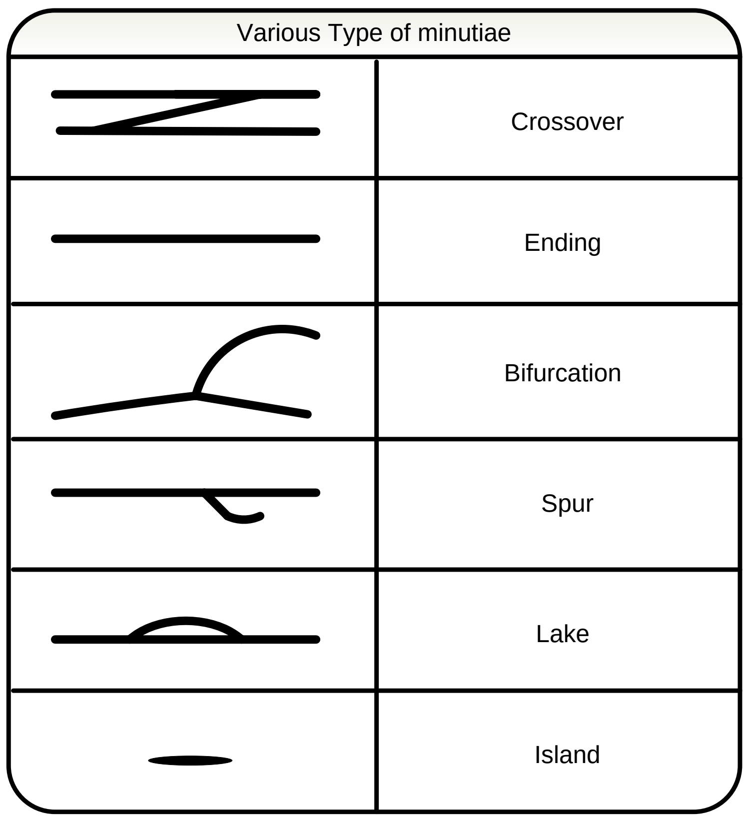


Figure 1.5: Some common minutiae points

1.4 Problem Statement

Minutia based approach is one of the most famous for fingerprint recognition. Many researchers have used this information in different flavors using different techniques. Some have calculated different features across a close neighborhood of each minutia. These features are Type, Co-ordinate, Distance, Ridges count, Direction and relative angles etc. This approach is also good when working with large databases of fingerprints. The extraction of minutiae points correctly is serious challenge specially when there is a case of a bad noisy image as it can generate many false minutiae during this phase. As the matching solely depends on the structural information that is extracted from these minutiae points so these false minutiae also causes problems during matching and generate undesirable results. So to overcome the deficiencies of this approach we should have a strong and robust feature set that confiscates the problem caused by those false minutiae. And also the features extracted should be simple enough that it improves the system accuracy without disturbing its performance and without adding much complexity in the system.

1.5 Objectives

The main objective of this thesis is to make a hybrid technique by using different conventional techniques for fingerprint matching and to perform efficient fingerprint matching and to achieve maximum accuracy. For that purpose two different types of features are extracted from two different domains. First types of features are extracted directly from a gray scaled image without any preprocessing requirements. For the extraction of these features frequency domain is used. The second approach is minutiae based in which the gray scale image is applied with a series of preprocessing steps and then after those minutiae based features are extracted from that image. These features are called spatial domain features.

The primary objective of this thesis is to design such a rich feature set that can help to achieve maximum accuracy while not much degrading the other important matters e.g. performance and efficiency. Hence features both types of features are fused together to get such feature vector. The secondary objective includes design/modification of the algorithms for classification of fingerprints, design and development of a system for testing the performance of the designed classifier and finally detailed analysis of the obtained results in terms of accuracy, performance and success rate.

1.6 Thesis Organization

Rest of the research is organized as follows. Chapter 2 gives a review of the techniques that have been used for finger print enhancement, feature extraction and matching techniques. Chapter 3 gives the formal definition and provides a framework for the solution of the problem in hand. It also lists the assumptions and conditions that define the scope of the work. Chapter 4 illustrates the detailed design of algorithms for different modules. It also further explains how these modules are integrated. Chapter 5 gives an in depth analysis of the results obtained during the experimentation and. Chapter 6 concludes the research and highlights the future work, which can be done to carry forward this effort.

1.7 Summary

This Chapter covers the different aspects of the research topic. It has also presented the motivation behind the selection of this subject as final thesis. The problem statement is given to clarify the scope of the project and then the main objectives that are required to be achieved are also defined. At the end an organization of the rest of the document is provided.

LITERATURE REVIEW

2.1 Introduction

This chapter includes the summary of various approaches used to deal with the problem of Fingerprint matching or recognition. The chapter initially describes different approaches used for fingerprint matching. Then it covers the background work on preprocessing, feature extraction and post processing techniques for fingerprint verification systems. Considering the importance of feature extraction module after preprocessing, the work done by researchers on the extraction of different type of features is also reviewed.

2.2 Fingerprint Matching Approach

When talking about automatic fingerprint matching a large number of methods have been designed over the last 40 years. Overall these approaches can be divided into three main (classes) categories of fingerprint matching:

- **Correlation-based matching:** In this approach two fingerprint images are superimposed on each other and the correlation or correspondence (at the intensity level) between corresponding pixels is computed for different alignments (e.g., various displacements and rotations) [6] [7].
- **Minutiae-based matching:** As the name states, firstly minutiae are extracted from the two fingerprints and stored as sets of points in the two-dimensional plane. Minutiae matching essentially consists of finding the alignment between the template and the input minutiae sets that results in the maximum number of minutiae pairings [15] [16].

- **Ridge feature-based matching:** As a matter of fact minutiae extraction is difficult in very low-quality fingerprint images, whereas other features of the fingerprint ridge pattern (e.g., local orientation and frequency, ridge shape, texture information) may be extracted more reliably than minutiae, even though their uniqueness is generally lower. In this approach the comparison of fingerprint is made using features extracted from the ridge pattern. Features of ridges are extracted from the gray-scale images e.g. a Gabor filter based method uses this scheme [17] [18].

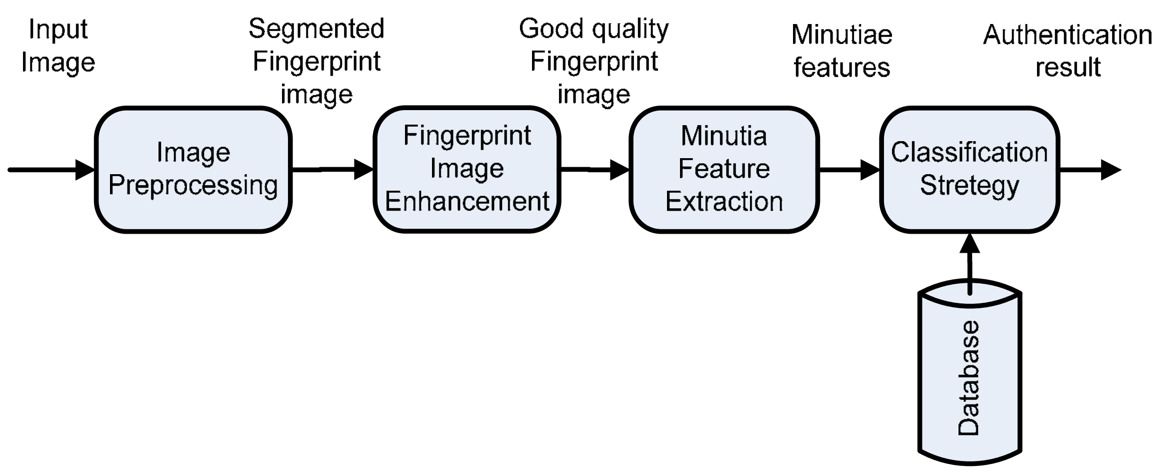


Figure 2.1: Complete process of Fingerprint identification system

2.3 Pre-processing and Fingerprint Enhancement

The Pre-processing and Enhancement steps in a fingerprint verification system serve many important purposes. As they not only help to enhance the image quality but also remove the background and unnecessary information from an image. These steps represent the input image into such form that makes it possible to extract the features set from image.

Ridge structures in a fingerprint image possess important characteristics and carry various features. Minutiae are one of them and for their extraction; fingerprint image should be in good quality. Ideally, in a well-defined fingerprint image, the ridges and valleys should be oscillated one by one and flow in locally constant direction. This regularity facilitates the detection of ridges and consequently, allows the minutiae to be extracted precisely from the thinned ridges. However, in practice, a fingerprint image may not always be well defined due to elements of noise that corrupts the clarity of the ridge structures. This corruption is due to variations in skin impression such as scars, humidity, dirt, and non-uniform contact with the fingerprint capture device [9].

Figure 2.2 shows a poor quality image and it is clear from the figure that the ridge information in them is very much lost. So in order to ensure that the performance of the minutiae extraction process is good even if the quality of input fingerprint image is poor, an enhancement algorithm is required. This algorithm improves the clarity of the ridge structures, if necessary.

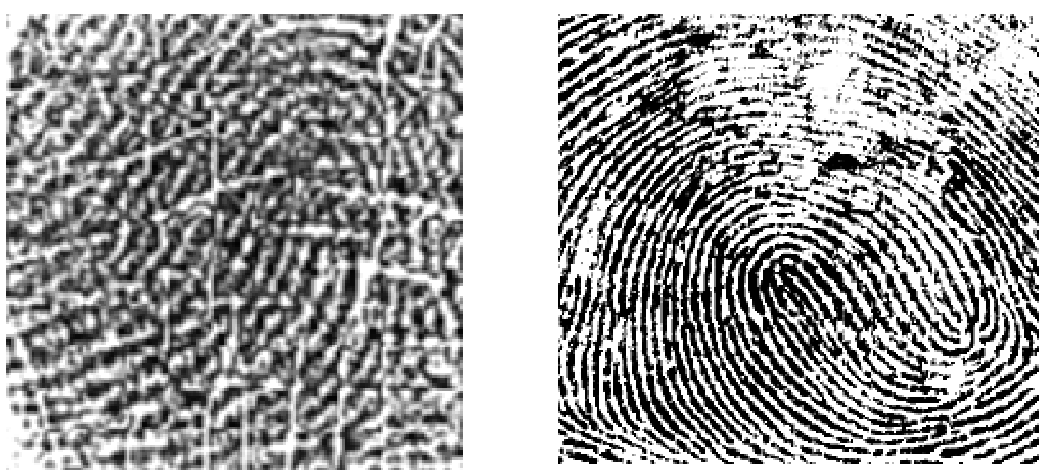


Figure 2.2: Poor quality fingerprint images

Thus, many image enhancement techniques are employed to reduce the noise and enhance the definition of ridges against valleys. In the following section I will review existing technique in the field of fingerprint image enhancement.

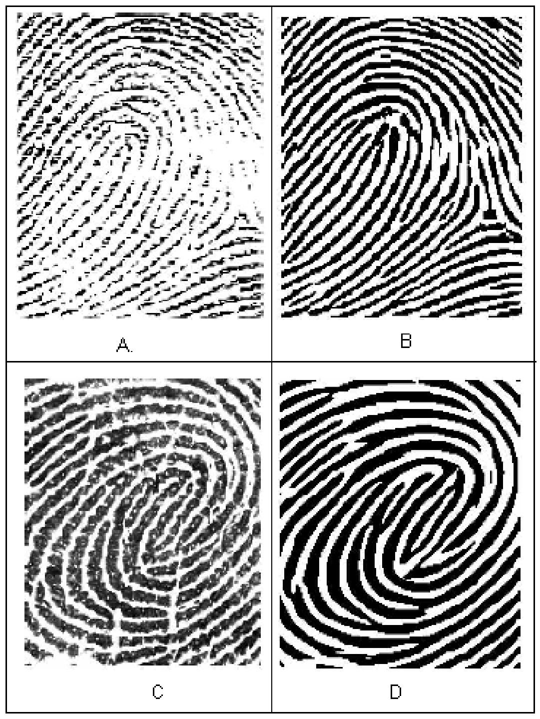


Figure 2.3: Images after enhancement

2.4 Fingerprint Enhancement Techniques

There are many approaches that are used for fingerprint enhancement but there is one which is most widely used by many researchers that is suggested by Lin Hong [11]. There is also another technique which is called Frequency Based techniques. In the next section I will discuss both of these techniques briefly.

2.4.1 Lin Hong Technique

This technique is based on the convolution of the image with Gabor filters tuned to the local ridge orientation and ridge frequency. This algorithm consists of four main stages:

- Normalization,
- Orientation estimation,
- Ridge frequency estimation, and
- Gabor filtering.

Now let's briefly see each of above mentioned step.

2.4.1.1 Normalization

Normalization is the first step in this approach. It includes the normalization of the fingerprint image so that it has a pre-specified mean and variance. As imperfection and distortion caused while fingerprint image capturing process due to non-uniform ink intensity or non-uniform contact with the fingerprint capture device, a fingerprint image may show variations in gray-level values along the ridges and valleys. Thus, it is very important to normalize the effect of these variations, which facilitates the subsequent image enhancement steps [38].

2.4.1.2 Orientation Estimation

After the process of normalization the Orientation of the image is calculated. As the name suggests orientation estimation of the image is a matrix of direction vectors representing the ridge orientation at each location in the image. Among many approaches the mostly commonly employed technique is gradient-based. This approach is used to calculate the gradient, which works on the fact that the orientation vector is orthogonal to the gradient. Firstly, the image is partitioned into square blocks and the gradient is calculated for every pixel, in the x and y directions. Then the orientation vector for each block is derived by performing an averaging operation on all the vectors orthogonal to the gradient pixels in the block. Due to noise in the image and some corrupted parts in the image, the ridge orientation may not always be properly determined. Given that the ridge orientation varies slowly in a local neighborhood, the orientation image is then smoothed.

A.Jain and D.Manio first computed the orientation followed by the filtering suggested by [10], [11]. A Low pass Gaussian filter was used and again angles were computed to find the smoother field as shown in Figure 2.4.

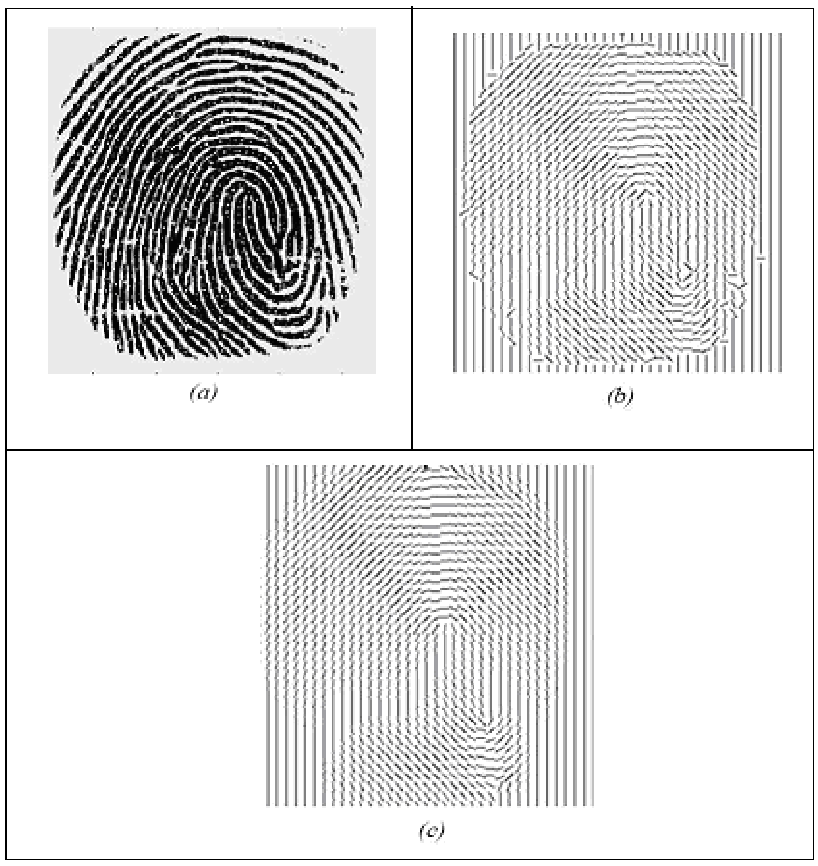


Figure 2.4: Local ridge orientation of a fingerprint image computed over a square-meshed grid: (a) original image, (b) orientation image, (c) smoothed orientation image.

2.4.1.3 Frequency Estimation

The next step in the image enhancement process is the estimation of the ridge frequency image. The frequency image defines the local frequency of the ridges in the fingerprint. First, the image is divided into square blocks and an oriented window is calculated for each block. Then for each of this block, an x-signature signal is constructed using the ridges and valleys in the oriented window. The x-signature is the projection of all the gray level values in the oriented window along a direction orthogonal to the ridge orientation. As a result, the projection outlines a sinusoidal-shape wave in which the center of a ridge maps itself as a local minimum in the projected wave. The distance between consecutive peaks in the x-signature can then be used to estimate the frequency of the ridges.

2.4.1.4 Gabor Filtering

Fingerprint enhancement methods based on the Gabor filter have been widely used to facilitate various fingerprint recognition applications. Actually Gabor filters are band-pass filters that have both frequency-selective and orientation-selective properties [8]. This means that the filters can be effectively tuned to precise frequency and orientation values. One valuable property of fingerprints is that they are known to have well defined local ridge orientation and ridge frequency. Therefore, the improvement or enhancement algorithm takes advantage of this regularity of spatial structure by applying Gabor filters that are tuned to match the local ridge orientation and frequency. Based on the local orientation and ridge frequency information around each pixel, the Gabor filter is applied to each pixels location in the image. As a result the filter enhances the ridges oriented in the direction of the local orientation, and decreases anything oriented differently. Hence, the filter increases the contrast between the foreground ridges and the background, while effectively reducing noise.

2.4.2 Frequency Based technique

The approach discussed in the previous section was a spatial domain technique. That technique actually involves spatial convolution of the image with filters, which can be computationally expensive. There is another approach to enhance the features in a fingerprint image employed by Sherlock [13] called “Directional Fourier filtering”. Operating in the frequency domain allows us to efficiently convolve the fingerprint image with filters of full image size. This technique can be divided into following main steps.

- Image Orientation
- Frequency Domain Filtering
- Filtered Image
- Binarization

Let's briefly see each of this technique one by one.

2.4.2.1 Image Orientation

In this second approach of fingerprint enhancement, the process starts by computing the orientation of the image. If compared to the previous method, which calculates the ridge orientation using a continuous range of directions, this method uses a set of only 16 directions to calculate the orientation. An image window is centered at a point in the raw image, which is then used to obtain a projection of the local ridge information. The image window is then rotated in each of the 16 equally spaced directions, and in each direction a projection along the window's y axis is calculated. The projection with the maximum variance is used as the dominant orientation for that point in the image. This process is then repeated for each pixel to form the orientation of image.

2.4.2.2 Frequency Domain Filtering

As filtering applied by [11], in this approach after the orientation image has been figured out, the raw image is then filtered using a set of band-pass filters tuned to match the ridge orientation. The image is first converted from the spatial domain into the frequency domain by appliance of the two-dimensional discrete Fourier transform. The Fourier image is then filtered using a set of 16 Butterworth filters with each filter tuned to a particular orientation.

The number of directional filters corresponds to the set of directions used to compute the orientation image. After each directional filter has been independently applied to the Fourier image, the inverse Fourier transform is used to convert each image back to the spatial domain, thereby producing a set of directionally filtered images called pre-filtered images.

2.4.2.3 Filtered Image

After the frequency domain filtering has been applied the next step in the enhancement process is to construct the final filtered image using the pixel values from the pre-filtered images. This requires the value of the ridge orientation at each pixel in the raw image and the filtering direction of each pre-filtered image. Each point in the final image is then computed by selecting, from the pre-filtered images the pixel value whose filtering direction most closely matches the actual ridge orientation. The output of the filtering stage is an enhanced version of the image that has been smoothed in the direction of the ridges.

2.4.2.4 Binarization

Finally the approach that is used at the last is Binariazation. During this process local adaptive thresholding is applied to the directionally filtered image, which produces the final enhanced binary image. This involves calculating the average of the gray-level values within an image window at each pixel, and if the average is greater than the threshold, then the pixel value is set to a binary value of one; otherwise, it is set to zero. The gray-level image is converted to a binary image, as there are only two levels of interest, the foreground ridges and the background valleys.

2.5 Minutiae Extraction Techniques

Overall the minutiae extraction algorithms can be categorized into two main approaches that are mostly described in literature. First approach is based on binarization based method and the second is direct extraction from gray scale images.

2.5.1 Binarization Approach

In this approach, the gray scale image is converted into a binary image prior to minutiae detection. The algorithms differ have the following common stages.

- Segmentation/Binarization: In it, the gray scale image is converted to a binary image through the process of simple thresholding or some form of adaptive binarization. The quality of the binarization output is improved if the gray scale image is enhanced prior to this process.
- Thinning: The resulting binary image is thinning by an iterative morphological process resulting in a single pixel wide ride map. This thinned image helps minutia extraction.
- Minutiae Detection/Marking: The most commonly employed method of minutiae extraction is the Crossing Number (CN) concept [4]. This method involves the use of the binarized thin image and an eight-connected neighbor window is calculated for each ridge pixel. The minutiae are detected by scanning the local window of size 3x3 for each ridge pixel in the image. Then using this window the CN value is then computed. This value is defined as half the sum of the differences between pairs of adjacent pixels in the eight-neighborhood. Using the properties of the CN as shown in Table 2.1, the ridge pixel can then be classified as a ridge ending, bifurcation or non-minutiae point. For example, a ridge pixel with a CN of 3 shows a ridge bifurcation, and a CN of 1 determines a ridge ending.

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

Table 2.1: Properties of the Crossing Number.

Other author such as Jain et al. [9] has also performed minutiae extraction using the skeleton binary image. Their approach involves using a 3x3 window to examine the local neighborhood of each ridge pixel in the image. A pixel is then classified as a ridge ending if it has only one neighboring ridge pixel in the window, and classified as a bifurcation if it has three neighboring ridge pixels. Consequently, it can be seen that this approach is very similar to the Crossing number method.

- Post-processing: As a result of minutiae extraction process there are normally two types of errors that are injected in the system. The detection may introduce spurious minutiae where they do not exist in the original or may skip genuine minutiae. While nothing can be done about the missing minutiae, spurious minutiae can be eliminated by considering their spatial relationships. Several heuristic rules may then be applied to filter out these false positives.

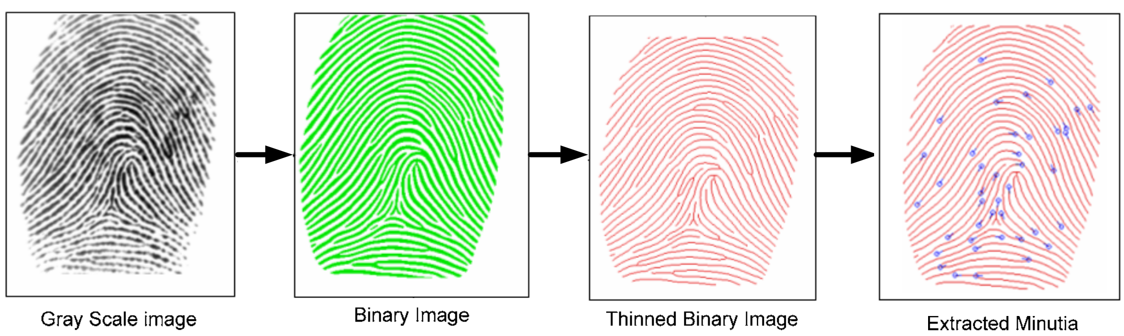


Figure 2.5: Different stages of binarization approach for minutiae extraction

2.5.2 Direct Gray-Scale Approaches

Approaches working with gray level images are mostly based on ridge following. Maio and Maltoni [5] proposed a feature extraction algorithm that directly operates on gray scale image. This algorithm is based on tracking the ridges by following the location of the local maxima along the flow direction. The ridge line algorithm attempts to locate at each step, the local maxima relative to a section perpendicular to the local ridge direction. Given a starting point (x0, y0) and a direction $\theta 0$, the algorithm computes the next ridge point by moving μ pixels in the direction $\theta 0$. Since the position (x1, y1) is only approximate, the algorithm computes the ridge position by finding the peak along a line orthogonal to $\theta1$. The algorithm then computes the next point by moving μ pixels in the new direction $\theta1$. The algorithm avoids revisiting the same ridge by keeping track of the points traced so far. With this approach, the local maxima cannot be reliably located in poor quality images and therefore, false positives are still introduced. Whole process is shown in figure by [26].

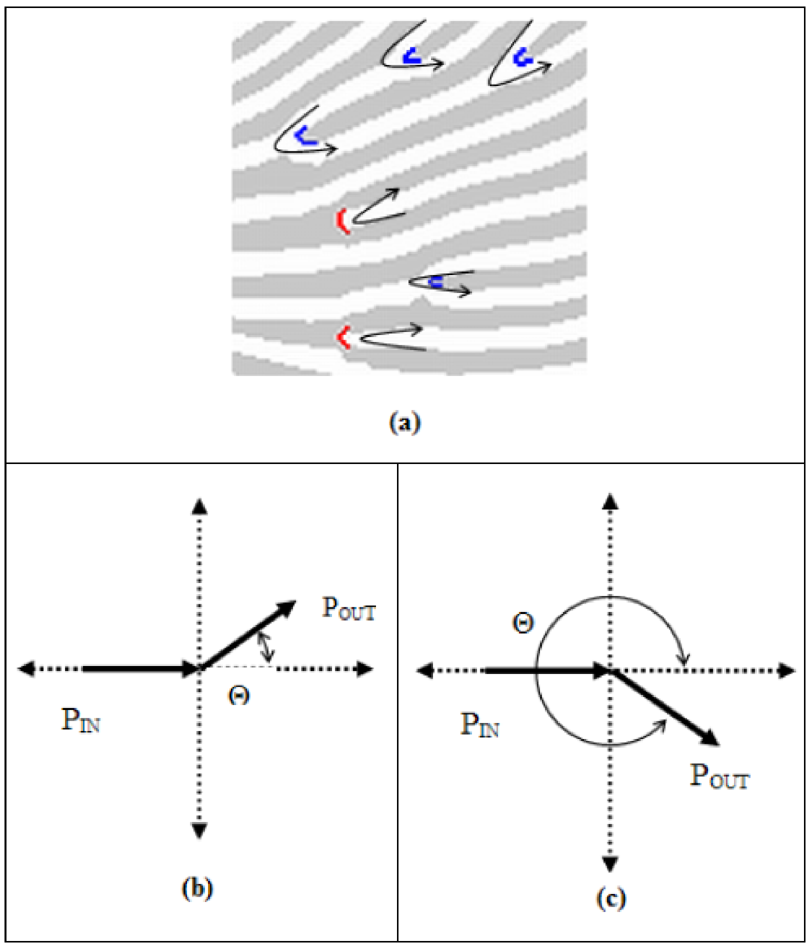


Figure 2.6: (a) Minutiae marked by significant turn in the contour (b) Left turn (b) Right turn

2.6 Matching Techniques

When talking specifically about fingerprint recognition techniques there are many algorithms that are proposed in the literature for finger print matching. As my work mostly covers minutiae based matching so I would only like to include a study of some minutiae based matching algorithms since these are the most widely approach for fingerprint recognition. When looking at these algorithms in the broader perspective they can be classified as follows.

- **Global Matching:** As the name suggests in this approach, the matching process tries to simultaneously align or match all points at once. Identical transformation function is considered for all points of a finger print. However, this is only estimation since two occurrences of the fingerprint belonging to the same individual are often related by a non-linear transformation. The global matching approach can be further categorized into two sub categories:
    a) Implicit Alignment Most point pattern matching problem belong to this class. Here the process of finding the matched points and finding the optimal alignment are performed at the same time.
    b) Explicit Alignment In this approach, the optimal transformation is obtained after explicitly aligning one of more corresponding points.
- **Local Matching:** In this approach, the matching tries to match minutiae points within a small local neighborhood. The trustworthiness of matching all the points is obtained by merging these local matches. Local matching algorithms are much more vigorous to non-linear distortion and partial overlaps when compared to global approaches.

As my work is related to the local matching so in the next section I will closely look at this strategy in little bit more detail.

2.6.1 Local Matching

In this approach, to decide whether a fingerprint is a match or not depends upon gathering evidence from matching local neighborhood structure of each minutia. Each local neighborhood is related to its structural properties that are invariant under translation and rotation. However, local neighborhood minutiae do not sufficiently confine the global structural relationships making false accepts very frequent. Therefore in practice, matching algorithms that rely on local neighborhood information are implemented in two stages

- Local structure matching: As the name suggests in this step, local structures are compared to derive candidate matches for each structure in the reference print.
- Consolidation: This step involves the validation of candidate matches based on how it agrees to the global match and a score is generated by consolidating all the valid matches.

The various local structure based algorithms mentioned in literature can be differentiated in the following perspective.

- Local features/representation chosen: Hreichak and McHugh [14] associate an eight dimensional vector {v1, v2...v8} to each minutia encoding the local neighborhood information, where each dimension represents the number of minutiae of a given type are in the local neighborhood of that minutia. The minutia types that are considered are dots, ridge endings, bifurcation, island etc. Jian and Yau [20] propose a eleven dimensional feature vector (See Figure 2.7) constructed by considering two nearest neighbors (mj,mk) of each minutia mi. The feature vector [d_{ij}, d_{ik}, θ_{ij}, θ_{ik}, Φ_{ij}, Φ_{ik}, n_{ij}, n_{ik}, t_i, t_j, t_k]. Here d_{ab} represents the Euclidean distance between minutiae m_a and m_b. n_{ab} represents the ridge count between the minutiae m_a and m_b. θ_{ab} represents the relative orientation of minutia mb with respect to minutia ma. Φa represents the orientation of the edge connecting ma to mb measured relative to θi.

Jea and Govindaraju [3] simplify this representation to include only the information related to minutiae. They reduce the feature vector to a five dimensional feature vector, retaining only [$d$ij, $d$ik, θij, θik, Φij $-\Phi$ik] without considerable loss in accuracy. Where as Ratha et al. [2] represent the local structure (star) in the form of a minutiae adjacency graph (MAG). The graph represented by G (V, E) consists of a center vertex vi representing mi and all vertices vj such that the Euclidean distance $d$ij < Thd where Thd is some threshold. Each edge $e$ij belongs-to E is labeled with five features (i, j, $d$ij, $n$ij, Φij) with the elements having the same meaning as mentioned before.

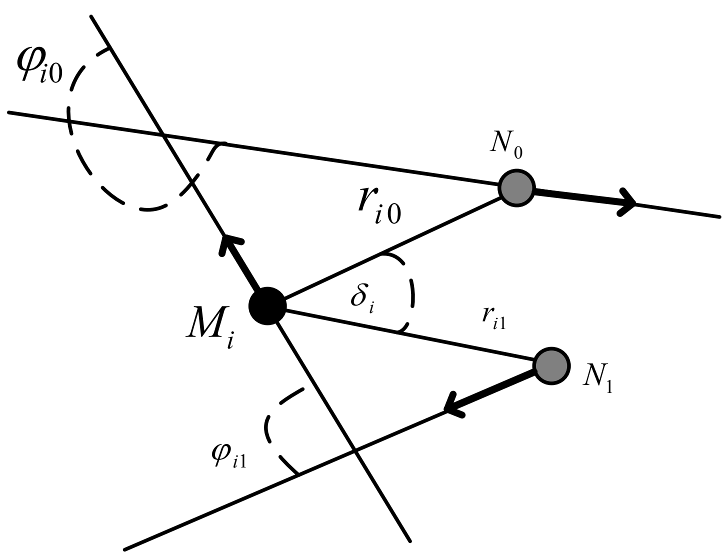


Figure 2.7: Illustration of the local secondary features used by [45] [43]

- Criterion for matching local neighborhoods: In Jian and Yau's approach [20] the neighborhoods are matched considering a weighted distance between the vectors. A similar approach is followed in Jea and Govindaraju [3]. Ratha et al. [2] matched each star by considering minutiae in increasing angles of Φ. The result of local matching process is a set of candidate star pairs.

- Consolidation process: In Jian and Yau's approach [20], the pair of best correspondence is obtained by considering the pair that has the least weighted distance. This pair is then used to align the two point sets with each other. In the consolidation process, the weighted distance of all the registered (aligned) pairs are used to determine the final match score. There is always an explicit alignment stage involved in this process. In Ratha's [2] approach each candidate star pair is checked for consistency in the second stage. Jea and Govindaraju [3] adopt a different approach for consolidation of the local matches. After the candidate secondary feature pairs have been obtained they actually build a histogram of the global rotation parameter. The histogram peak will correspond to the optimal rotation angle. After that secondary feature pairs are then pruned by checking their local rotation parameter with the global optimum. The center of the secondary feature pair with the minimum cost of misalignment is chosen as the reference point for alignment.

2.7 Summary

This Chapter reviews the techniques and algorithms developed and implemented for the fingerprint matching. Initially some techniques related to the fingerprint enhancement are listed. Overall, it can be seen that most techniques for fingerprint image enhancement are based on filters that are tuned according to the local characteristics of fingerprint images. Both of the examined techniques employ the ridge orientation information for tuning of the filter. However, only the approach by Hong [11] takes into account the ridge frequency information, as Sherlock's [13] approach assumes the ridge frequency to be constant. By using both the orientation and ridge frequency information, it allows for accurate tuning of the Gabor filter parameters, which consequently leads to better enhancement results. Hence, the Gabor filtering approach by Hong has been employed to perform fingerprint image enhancement.

After the phase of enhancement, the chapter lists different approaches for the extraction of minutiae. And finally reviews both local and global techniques for the recognition of fingerprint. Since the purpose of this work is related to local matching technique therefore the focus has been kept on this approach.

C h a p t e r 3

METHODOLOGY

3.1 Introduction

This chapter gives establishes the theoretical background of composition of this research work. First of all a formal definition of the problem is given then it is further extended to the list of various conditions and assumptions that are made in this research work. Then the problem has been decomposed into various modules. The theoretical background of adopted approach for the solution of each of these modules is also presented.

3.2 Proposed System Description

In automated fingerprint recognition systems, the input of the system is actually in t the form of a fingerprint images. For testing such systems images are usually taken from standard Fingerprint Verification Competition FVC databases. These databases are made in such a form that a large amount of scenarios can be addressed. Usually for a single fingerprint there are total of 8 different types of sample images. In my work the database that is used is FVC 2002. This database contains total of 100 fingerprints. The main goal of this thesis is to get maximum accuracy with very low FAR and FRR.

In the proposed system, first training is being performed. During the training phase features of each trainee images are extracted and then stored in the database. Now during the testing phase, features are first extracted from the testing image one by one and are matched with the stored feature vectors of template images. After a match is found the image whose features are best matched is loaded from the image database.

The proposed system is designed in Windows based environment and program is written in MATLAB 7.0 thus it requires the MATLAB software at the desired system to perform the desired task.

3.3 Problem Decomposition into Modules

To solve every problem it is very important that it must be divided into modules. So in my case the system is also divided into several modules. Each of these modules performs very important functionality that helps to attain over all system wise solution. There two operating modes of the system as described earlier, that are enrollment and testing. As in both cases of the system operating modes the primary input of the system is the fingerprint image, whether the system as a working in enrollment or training mode or working as identification or recognition mode. So for this system the fingerprint images database that is used is FVC 2002, which is most widely used database and is considered as standard for checking performance of fingerprint verification systems. This database has already been used by many researchers. As described earlier this database has a total of 100 different fingerprints, and each image has 7 other variations for testing purpose. So a total of 800 fingerprint images are in it. The dimension of each image is (388 * 374).

If looking at this system in the broader view it has two main components i.e. Training and Testing. Both of them have almost the same modules up to a certain level except the Recognition module is only present in Testing Component. The overall modules, that actually form both components of the system includes; *The Pre-processing Module*, *The Minutiae Extraction Module*, *The Wavelet Transformation Module*, *The Feature Extraction Module* and finally *The Classification Module*. As told earlier the training component uses all modules except the classification or the recognition one. Testing component uses all modules but it doesn't store the features into the database. Process diagram in Figure 3.1 depicts this thing.

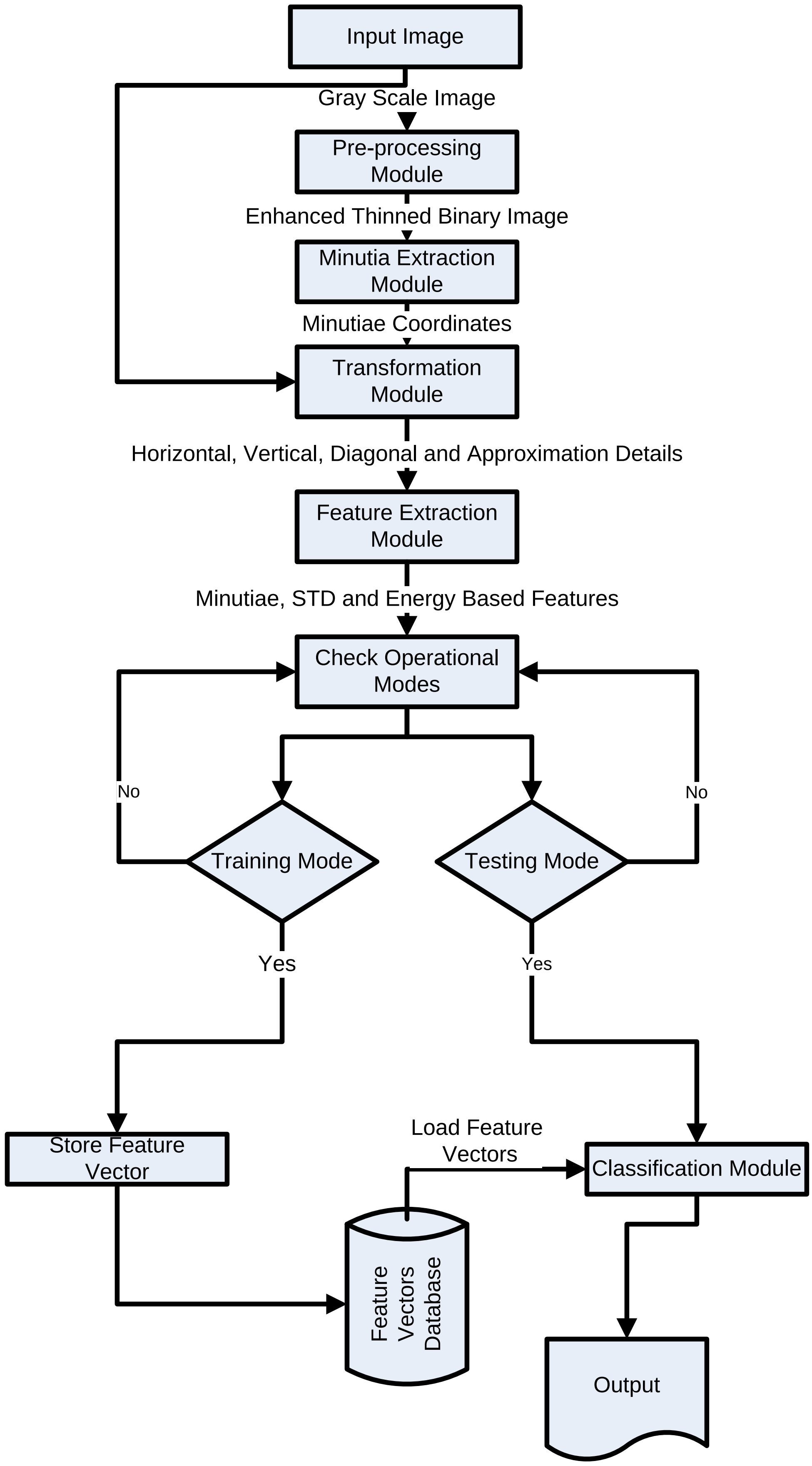


Figure 3.1: System Process Diagram

3.3.1 Pre-processing Module

The input of the system is raw fingerprint image and to extract features from it there are series of operations needed to be carried out. The purpose of these operations is to enhance the image quality. Different pre-processing steps that are most widely used for enhancing input fingerprint image were listed briefly in the previous chapter. Now the out of those the following enhancement steps some are employed for this system. The algorithm I have implemented is built on the techniques developed by Hong et al. [9] and few new steps that were actually added by Raymond [12]. Few morphological operations are also added and are applied to the thinned binary image. Figure 3.2 illustrates these steps in proper order.

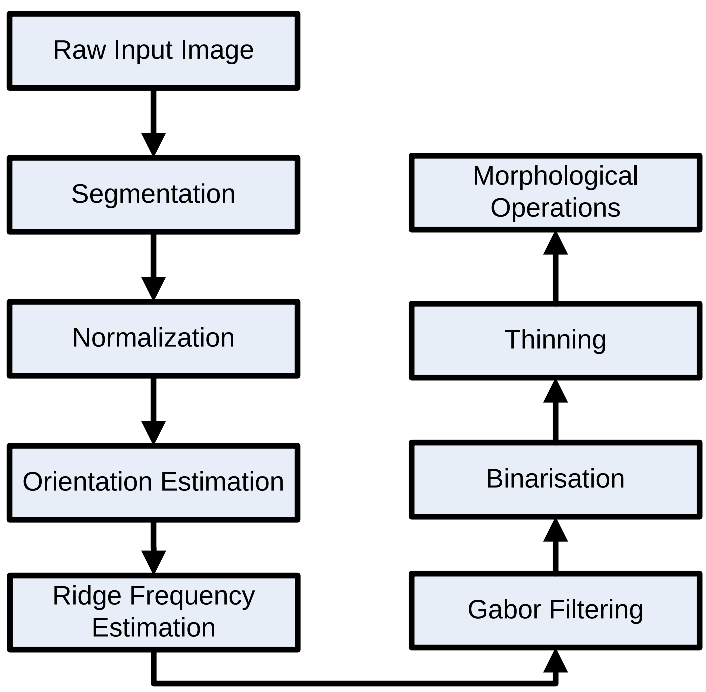


Figure 3.2: Pre-processing Steps

Finally the thinned ridge map or the skeleton image is then filtered by other three Morphological operations to remove some H breaks, isolated points and spikes. Let us quickly discuss purpose of these steps, and see why these steps are necessary. The input of the first step is a gray scaled fingerprint image. The next section describes a brief introduction to these steps.

Figure 3.3: The input gray scaled image

- **Segmentation**: Segmentation is the process of separating the foreground regions in the image from the background regions. The foreground regions correspond to the clear fingerprint area containing the ridges and valleys, which is definitely our area of interest. The background contains the region outside the borders of the fingerprint area and do not have any valid fingerprint information. When minutiae extraction process is applied to the background of an image, it extracts some false minutiae. Thus, segmentation is used to discard these background regions, which facilitates the reliable extraction of minutiae.

Figure 3.4: Segmentation of image

- **Normalization**: From the figure 3.2 it is clear that the next step in the fingerprint enhancement process is image normalization. Normalization is used to standardize the intensity values in an image by adjusting the range of grey-level values so that it lies within a desired range of values.

Figure 3.5: The normalized image

- **Orientation Estimation**: The orientation estimation step of the pre-processor defines the local orientation of the ridges contained in the fingerprint. The orientation estimation is a fundamental step in the enhancement process as the subsequent stage of Gabor filtering relies on the local orientation in order to effectively enhance the fingerprint image.

Figure 3.6: The result of orientation estimation

- **Ridge Frequency Estimation**: In addition to the orientation of the ridge in an image, another important parameter that is used in the construction of the Gabor filter is the local ridge frequency. The frequency image describes the local frequency of the ridges in a fingerprint.

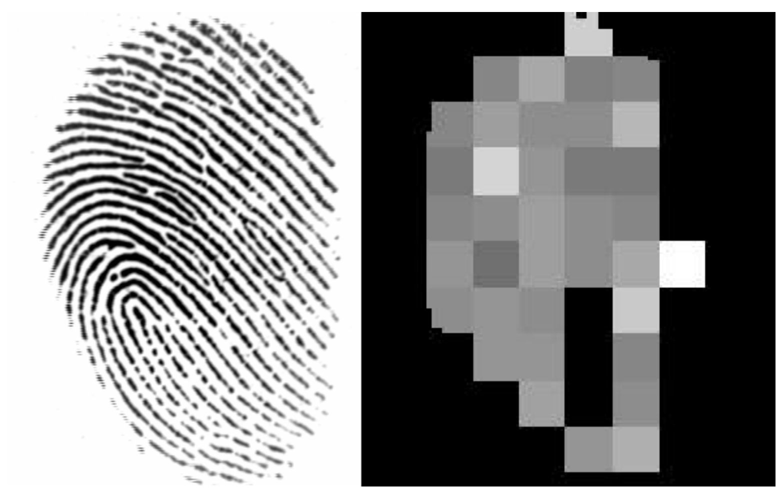


Figure 3.7: The result of frequency estimation

]

- **Gabor Filtering**: Once the ridge orientation and ridge frequency information has been determined in previous steps of the pre-processor, these parameters are used to construct the even-symmetric Gabor filter. A two dimensional Gabor filter consists of a sinusoidal plane wave of a particular orientation and frequency, modulated by a Gaussian envelope [14]. Gabor filters are employed because they have frequency-selective and orientation-selective properties and we have to take full advantage of this property. That's why we calculated these two parameters earlier. These properties allow the filter to be tuned to give maximal response to ridges at a specific orientation and frequency in the fingerprint image. Therefore, a properly tuned Gabor filter can be used to effectively preserve and enhance the ridge structures while reducing the effect of noise.

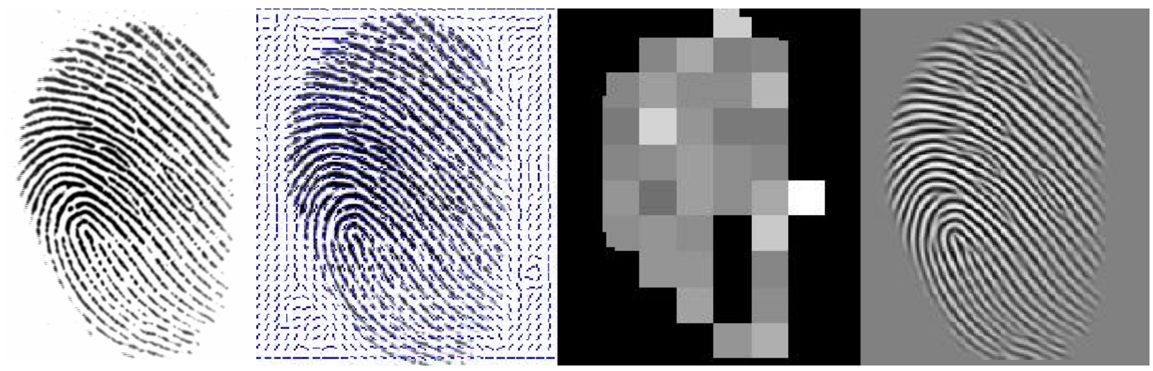


Figure 3.8: The process of performing Gabor Filtering

- **Binarization**: As in the previous chapter we have seen two strategies of the extraction of minutiae. The most minutiae extraction algorithms operate on binary images where there are only two levels of interest: the black pixels that represent ridges, and the white pixels that represent valleys. Now the binarization is the process that converts a grey level image into a binary image. This improves the contrast between the ridges and valleys in a fingerprint image, and consequently facilitates the extraction of minutiae.

  One useful property of the Gabor filter is that it has a DC component of zero, which means the resulting filtered image has a mean pixel value of zero. Hence, straightforward binarization of the image can be performed using a global threshold of zero. The binarization process involves examining the grey-level value of each pixel in the enhanced image, and, if the value is greater than the global threshold, then the pixel value is set to a binary value one; otherwise, it is set to zero. The outcome is a binary image containing two levels of information, the foreground ridges and the background valleys.

Figure 3.9: The process of performing binarization

- **Thinning**: The final image enhancement step is performed prior to minutiae extraction is thinning. The application of the thinning algorithm to a fingerprint image preserves the connectivity of the ridge structures while forming a skeleton version of the binary image. A standard thinning algorithm [24] is employed, which performs the thinning operation using two sub-iterations. This algorithm is accessible in MATLAB via the `thin' operation under the bwmorph function. Each sub-iteration begins by examining the neighborhood of each pixel in the binary image, and based on a particular set of pixel-deletion criteria, it checks whether the pixel can be deleted or not. These sub-iterations continue until no more pixels can be deleted. The skeleton image is then used in the subsequent extraction of minutiae.

Figure 3.10: Result of enhancement a thin binary image

3.3.2 Minutia Extraction Module

This section describes the methodology for performing the minutiae extraction stage. The minutiae extraction technique that I have implemented is based on the widely employed Crossing Number method [4] as briefly described earlier. The process involving the extraction of minutiae using this method will be discussed in the next chapter in more detail.

3.3.3 Wavelets Transformation Module

Wavelet transformation is not applied on the whole image. Each minutia point is used as center and window of size (35 x 35) is cropped across it, on which the transformation is performed. As minutiae based features are rotational invariant so to incorporate wavelets based features with them cropped image are also taken from the rotated versions of the print. The most important thing is that those wavelets should be applied which are good for pattern recognition. So for this research work, three families of wavelets have been used namely debauches, symlets and coiflets. For each family, further more types are applied e.g. db1, db4, db8, sym1, sym2, sym8, coif1 and coif8 etc. Wavelet transformation of the image results in the horizontal, vertical, diagonal and low level details which are then used for the feature extraction.

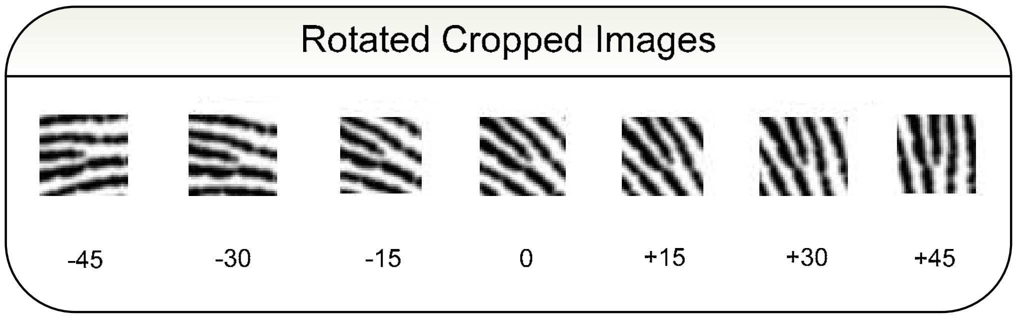


Figure 3.11: Cropped images taken after -45 to +45 degree of rotation

3.3.4 Feature Extraction Module

Once the minutiae have been extracted and the wavelet transformation is applied, features are extracted from them for the purpose of classification. Some features are taken as advised by A. Wahab [1]. In addition to these features some new features are also extracted as a result of wavelet transformations. These new features are of two types that are: standard deviation based features and Energy based features. These two features are associated to each minutia point of the image. After the extraction of all these features the template is formed for that image which is then stored in the database. Now in the following section let's see each type of these features in bit more detail.

3.3.4.1 Spatial Domain Features

After the extraction of minutiae the following features are extracted as by [1]. Overall spatial domain features are extracted for the two levels of classification. Features from each level are used at its stage of classification operations. For each minutia keeping it as a center minutia its neighborhood minutiae are calculated with in a certain radius. For the 1st stage of classification for each center minutia point, it's X, Y coordinates, type, neighbor-type, neighbor-distance, relative angle and ridge count from neighbors are calculated. And for the 2nd stage of classification some new features are calculated in the similar fashion. The features that are extracted in this stage are the minutia type, direction, relative angle and the distance of the neighborhood feature.

3.3.4.2 Standard Deviation Based Features

For the extraction of these features, for each minutia location by keeping it as center a window of (35 x 35) is cropped from the gray scaled image. For each cropped image wavelet is applied. Standard deviation is calculated for all details except low level details thus resulting in a total of 3 features. The use of wavelets at this cropped image reflects its specific recurrent ridge-valley structure. So a compact illustration of this information based on the standard deviation of each wavelet sub-image is being used as a feature vector for the classification purposes.

3.3.4.3 Energy Based Features

Energy based features are calculated exactly in the same way as standard deviation was calculated. So at the end a feature vector of length 3 based on energy is also obtained. The recurring ridge-furrow structure makes the wavelet coefficients of fingerprint image different from most natural images. Therefore, the energy distribution and orientations is a quite informative feature for fingerprint identification and offer good discriminatory properties. The energy of different sub-bands gives information regarding both the edge spatial frequency as well as the ridge orientation.

3.3.5 Classification Module

Proposed system uses distance based classification technique. In this classification technique distances are computed between the feature vectors, and on the basis of this distance classification is performed. The distances that is used in this work is Euclidian distance. As briefly described earlier the classification process is actually a two staged process. At each stage its concerning features participate in classification operation and a matching score is constructed. If the matching score of an image is higher then a particular threshold then that image moves to the 2nd stage of classification. The purpose of the 2nd stage is to make the classification criteria more stringent so that False Acceptance Rate (FAR) can be reduced. The 2nd stage of classification finally determines that whether the input image is a match or not. Figure 3.12 shows the operation of the two stages of classification. The details of this classification are given in Chapter 4.

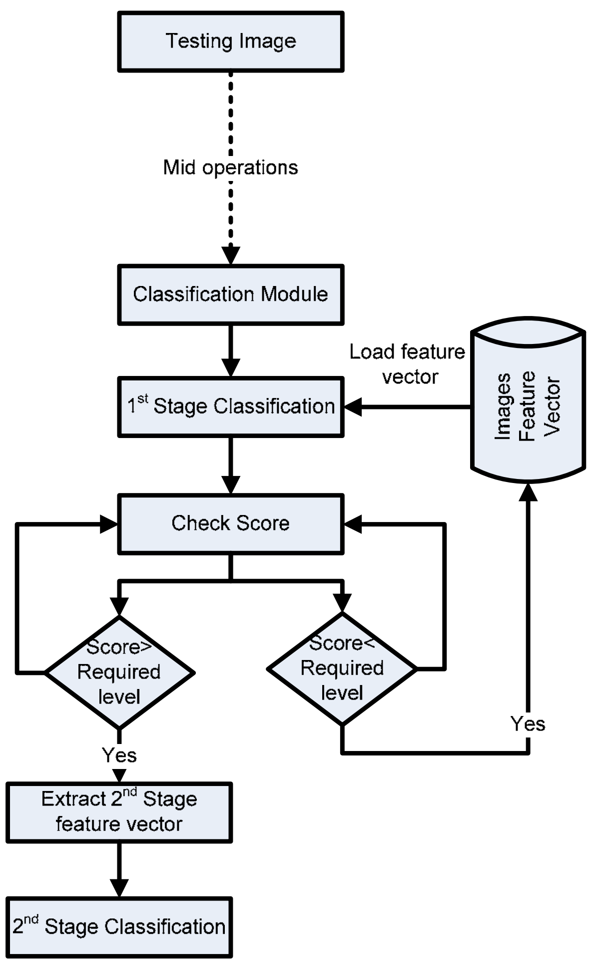


Figure 3.12: Brief description of two staged classification process

3.4 Summary

Chapter 3 sets up the basis of this research. It narrows down the vastness of the topic to the conditions and assumptions under which this work has been done. The process diagram is also given which lists the steps involved in the work. The chapter breaks down the process into modules and briefly explains the functioning of each the individual module.

DESIGNING AND IMPLEMENTING THE FRAMEWORK

4.1 Introduction

This chapter concentrates on the various algorithms that are actually adopted in this system. These algorithms are for Minutia extraction, Wavelets transformation, features extraction and classification modules. At the end the algorithm for the distance based classifier is also listed. In this chapter the implementation details of the different approaches adopted are described.

4.1.1 Minutia Extraction (Marking) Algorithm

The Crossing Number (CN) method is used to perform minutiae extraction. This method extracts the ridge endings and bifurcations from the skeleton image by examining the local neighborhood of each ridge pixel using a $3x3$ window. The CN for a ridge pixel P is given by [30]:

$$CN = 0.5\sum_{i=1}^{8}\left|P_i - P_{i+1}\right|, \qquad P_9 = P_1 \qquad \textbf{(4.1)}$$

where Pi is the pixel value in the neighborhood of P.

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

Table 4.1: Properties of the Crossing Number.

Algorithm

Begin

1. *For a pixel P, its eight neighboring pixels are scanned in an anti-clockwise direction.*
2. *Compute the CN for that pixel*
3. *Classify the pixel according to its CN value*
4. *If CN is equal to 0 then*

   This pixel is only an isolated point, not a minutia
5. *If CN is equal to 1 then*

   Mark that pixel as ending minutia

   Keep record of its X, Y coordinates

   Orientation of the associated ridge segment is calculated
6. *If CN is equal to 2 then*

   Do nothing and continue traversing
7. *If CN is equal to 3 then*

   Mark that point as Bifurcation minutia

   Keep record of its X, Y coordinates

   Orientation of the associated ridge segment is calculated
8. *If CN is equal to 4 then*

   That is only a crossing point not a minutia

End

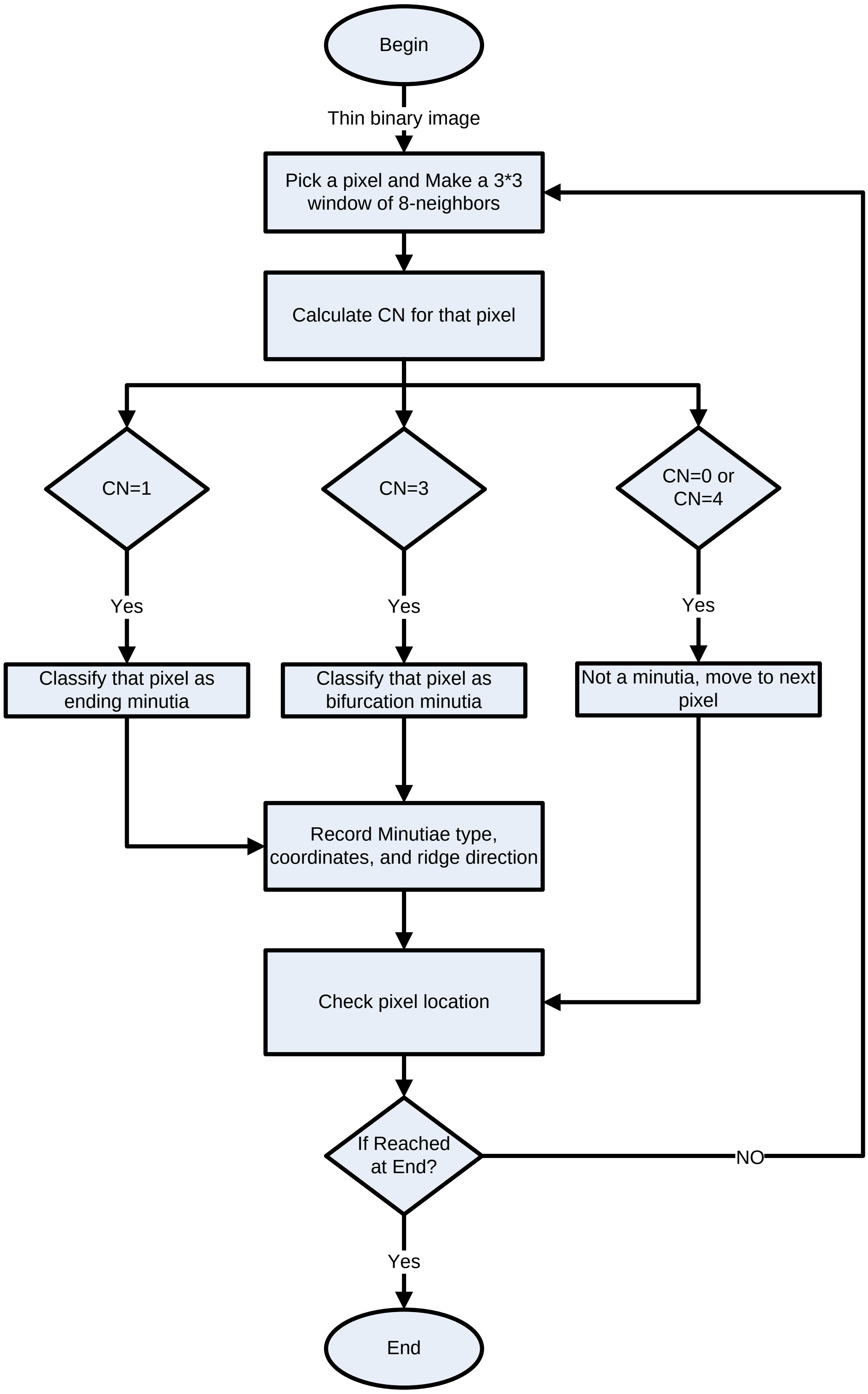


Figure 4.1: Flow chart for minutiae extraction algorithm

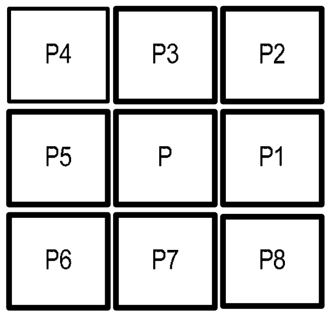
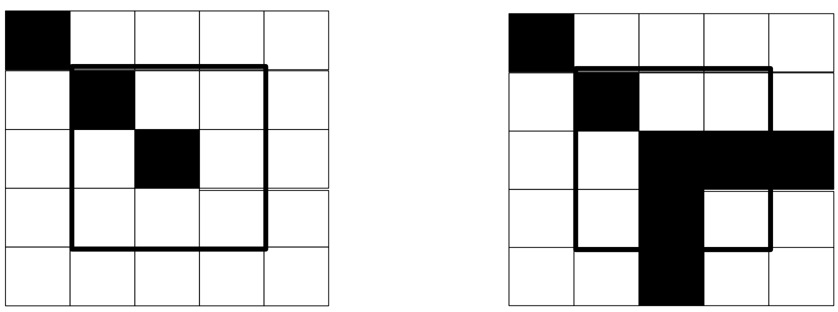


Figure 4.2: 3x3 neighboring window containing pixel locations

For each extracted minutiae point, the following information is normally recorded:

- *x* and *y* coordinates,
- orientation of the associated ridge segment, and
- type of minutiae (ridge ending or bifurcation).



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

Figure 4.3: Examples of a ridge ending and bifurcation pixel. (a) A Crossing Number of one corresponds to a ridge ending pixel. (b) A Crossing Number of three corresponds to a bifurcation pixel.

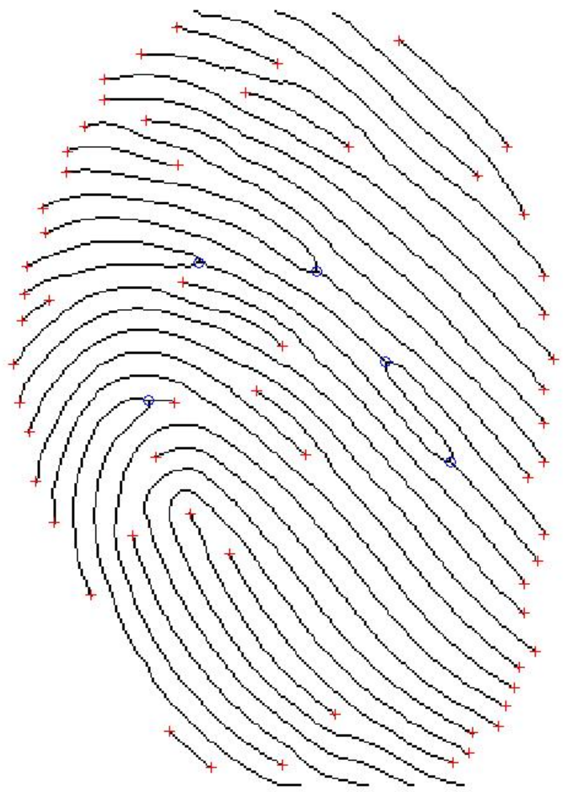


Figure 4.4: The extracted minutiae of fingerprint image

4.2 Wavelet Transformation

Wavelet analysis provide the time and frequency information which is not being provided by the Fourier and Short Time Fourier Transforms. Wavelet analysis tells us that which frequencies will exist during the specific intervals. This information is not provided by other transformations. Wavelet analysis allows the use of long time intervals where more precise low-frequency information is required and shorter regions where high-frequency information is required i.e. multi resolution analysis (MRA) of the image.

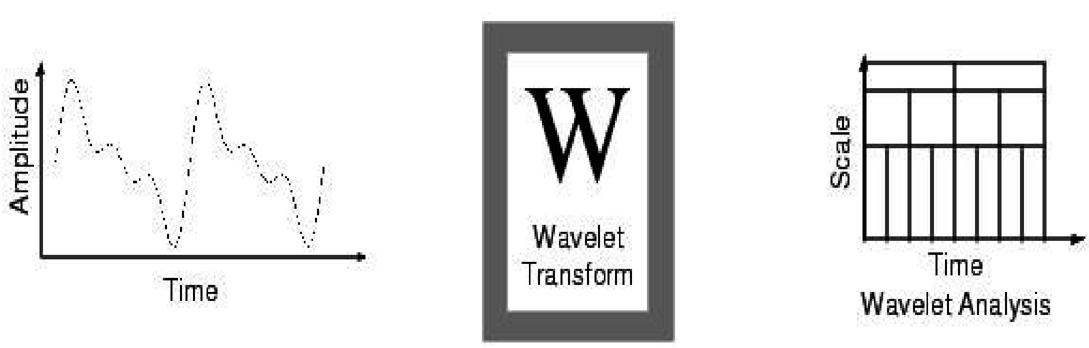


Figure 4.5: Wavelet transformation [35]

Wavelet analysis does not use a time-frequency region, but rather a time-scale region as compared to Fourier transform as shown in Figure 4.5. Wavelet analysis reveals such aspects of data that other signal analysis techniques miss like trends, breakdown points, discontinuities in higher derivatives and self-similarity. There are two main types of wavelet transforms [25].

4.2.1 Continuous Wavelet Transform

The continuous wavelet transform (CWT) is defined as the sum over all time of the signal multiplied by scaled, shifted versions of the wavelet function Ψ:

$$C(scale, position) = \int_{-\infty}^{\infty} f(t)\psi(scale, position, t)dt \qquad \textbf{(4.2)}$$

The results of the CWT are many wavelet coefficients C, which are a function of scale and position. Then each coefficient is multiplied by the properly scaled and shifted wavelet yields the constituent wavelets of the original signal. Scaling a wavelet simply means either it is stretched or compressed. The smaller scale factor means more compressed wavelet while larger factor means expanded wavelets. In wavelet analysis, the scale factor is related inversely to the frequency. In shifting a wavelet we simply delay (or hasten) its onsets. In continuous wavelet, mother (defined) wavelet is first placed at time interval t_o. It is then multiplied with the portion of the signal that it covers. Now mother wavelet is placed at new position e.g. t_1. Again wavelet will be multiplied with signal. Finally at the end results will be integrated to find one coefficient value. Next mother wavelet is scaled and shifting will be repeated to get another coefficient value. So this process results in number of coefficients which can be used for different applications.

4.2.2 Discrete Wavelet transform

Calculating wavelet coefficients at every possible scale is a fair amount of work and it generates an awful lot of data. So by choosing scales and positions based on powers of two then the analysis will be much more efficient and just as accurate. Such an analysis is referred as discrete wavelet transforms (DWT). This type of analysis is commonly used in pattern recognition. Let consider an image f(x,y) whose forward discrete transform can be expressed in the form of following general relation :

$$W_{\varphi}(j_0,m,n)=\frac{1}{\sqrt{MN}}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}f(x,y)\varphi_{j0,m,n(x,y)} \quad \textbf{(4.3)}$$

$$W_{\psi}(j_0,m,n)=\frac{1}{\sqrt{MN}}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}f(x,y)\psi_{j0,m,n(x,y)} \quad \textbf{(4.4)}$$

Where j_0 is an arbitrary starting scale and it refers to the horizontal, vertical and diagonal details. $W_{\varphi}(j_0,m,n)$ Coefficients define the approximation details and $W_{\psi}(j_0,m,n)$ coefficients add horizontal, vertical and diagonal details of image *f(x,y)* at scale j_0. Normally j_0 is set to zero and $N=M=2^j$ is selected in such a way that j=0,1,2,….,J-1 and m,n =0,1,2,…,2^j-1. φ represents the scaling function while ψ represents the wavelet function.

4.2.2.1 Different Types of Wavelets

I have used a total of three different wavelets from three wavelet families namely debauches, symlets, coiflets. Debauches wavelets have no explicit expression except for db1, which is the Haar wavelet.

Let $$P(y)=\sum_{k=0}^{N-1}C_k^{N-1+k}y^k \quad \textbf{(4.5)}$$

Where C_k^{N-1+k} denotes the binomial coefficients

$$\text{Then} \quad |m_0(w)|^2 = \left(\cos^2\left(\frac{w}{2}\right)\right) N_p\left(\sin^2\left(\frac{w}{2}\right)\right) \tag{4.6}$$

and

$$m_0(w) = \frac{1}{\sqrt{2}}\sum_{k=0}^{2N-1} h_k e^{-ikw} \tag{4.7}$$

To generate these wavelets, first polynomial coefficients are calculated form the binomial formula. This polynomial is then factored with the roots command in MATLAB. Then convolution is performed followed by the normalization to generate the desired filters. On the other hand Symlets are only near symmetric. Daubechies proposes modifications of her wavelets actually increase their symmetry, can be boomed while retaining great simplicity. The idea consists of reusing the function m_0 introduced in the dbN, considering the $|m_0(w)|^2$ as a function W of $z = e^{iw}$. Then W can be factored in several different ways in the form of

$$W(z) = U(z)\ \overline{U\left(\frac{1}{z}\right)} \tag{4.8}$$

By selecting U such that the modulus of all its roots is strictly less than 1, Daubechies wavelets dbN are built. The U filter is actually a "minimum phase filter." By making another choice, more symmetrical filters are obtained and called symlets.

Coiflets are considered to be more symmetrical. The scaling and wavelet functions have a support of length 6N-1. With respect to the supported length, coifN needs to compare with db3N or sym3N. To generate these wavelets, first the filter length is specified e.g. 6 for coif1, 12 for coif2 and so on. A starting vector is then guessed for the filter. Then fsolve () command of the MATLAB is used to solve a system of non-linear equations. Defined function is being used in fsolve () command with starting guess vector which performs number of operations including advancements by 2L, norms, double shifts etc. Finally low and high pass decomposition filters will be obtained at the end.

4.2.3 Feature Extraction

After performing all the enhancements and minutiae extraction procedures the system is now ready to perform the process of feature extraction. As told earlier there are basically two different types of features that are extracted. The first type is of features that are spatial domain features and are extracted from the minutiae information. The second type of features also use the information of minutiae, but are obtained after the application of wavelets transformations across a certain region of each minutia. These features use the co-ordinates of each minutia but actually are extracted from a gray scaled image. The extraction of these features does not need any enhanced image.

The next section describes the methodology involved in the extraction of features from both of these types in detail.

4.2.3.1 Minutia Based Features

As discussed earlier during the minutiae extraction phase initially reveals 2 type of information. These are minutiae X, Y coordinates and minutia type (ridge ending or bifurcation). Using this information many new features are extracted as suggested by A. Wahab [1] [9]. The idea is that instead of identifying each of single minutiae in both images the matching depends on the identification of a whole bunch of minutiae that are organized in a closes neighborhood. Different features are extracted among the minutiae in this neighborhood. These features help in identifying this neighborhood minutiae set in any image. If some of these minutiae neighborhood are matched then the image is considered to be a match.

But before taking a final decision about the matching of two images some new features are extracted in the second stage of matching. This stage not only helps in the removal of false minutia matched during first stage but also helps to align both the images. Once both images are aligned the newly calculated features are also matched. After the matching results of this stage the decision about the matching of image is then taken.

4.2.3.2 First Stage Features

As Wahab [1] suggested the features for the first stage of matching are extracted for a close neighborhood of minutiae. The idea that he suggested can be described as under:

Algorithm

Begin

1. *Load each minutia from the minutiae list*
2. *Consider each minutia as a central minutia among all*
3. *Calculate distance of all minutiae from this center one*
4. *Sort the minutiae distance list in ascending order*
5. *Pick the 5 closest distance minutiae from center and save their distance for this central minutia*
6. *For each neighboring minutia calculate the relative from the central minutia and a reference point calculated on the ridge of central minutia*
7. *For each neighboring minutia calculate the ridge count from center*
8. *Store all that information in form of a vector*

End

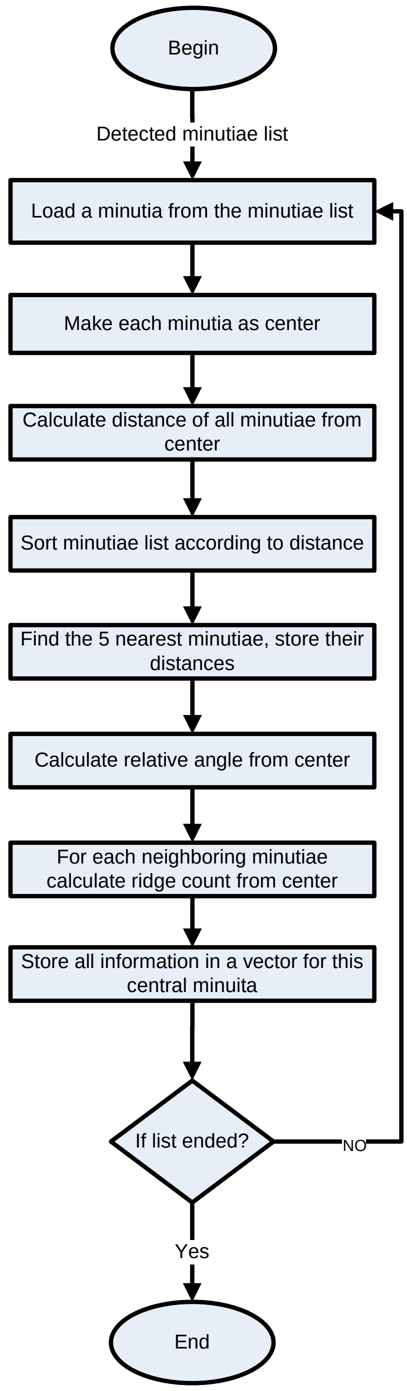


Figure 4.6: Flow chart for Feature extraction algorithm for 1st stage of classification

To organize this information over all two classes of feature vectors, *vector 1* and *vector 2* are defined. *Vector 1* is a one-dimensional vector containing the minutia type of the central minutia while *vector 2* is a three-dimensional vector containing the minutia type, distance, relative angle and ridge count of the neighborhood minutiae. Both of these features vectors are shown in Figure 4.7;

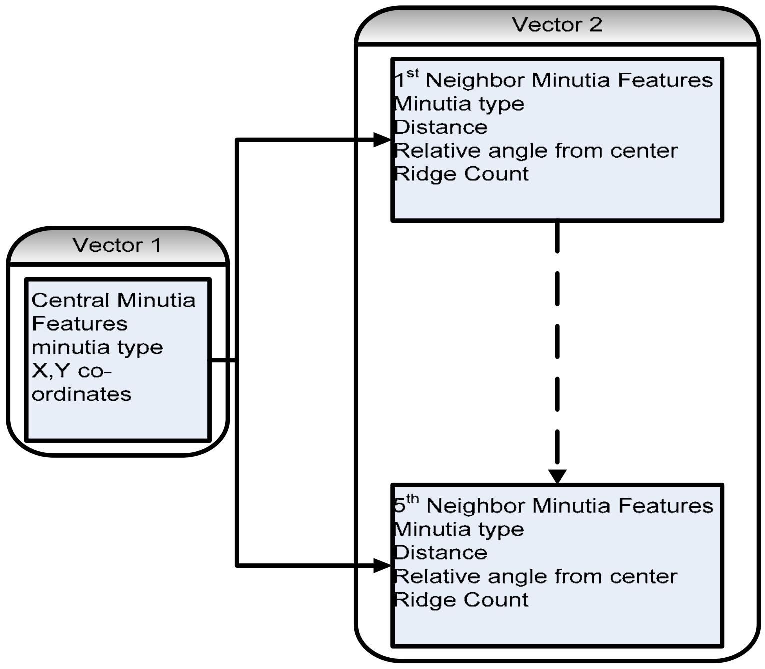


Figure 4.7: Organization of local feature vectors, Vector 1 and Vector 2 [1]

4.2.3.3 Second Stage Features

For the second stage of matching the global features are organized. The global features of the finger prints consist of all the central minutiae of the local features that have at least one minutia feature vector matched between the two prints. The features used in 2nd stage makes the matching process much more stringent. And they are organized in four dimensional features *Vector 3*. Figure 4.8 shows the features Vector 3.

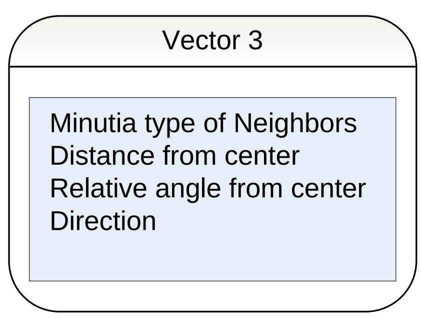


Figure 4.8: Organization of global features vector, Vector 3

4.2.3.4 Calculation of Feature Vectors

Previous section discussed about different type of features that are extracted and how they are organized. Following sections discusses about the methodology adopted for the extraction of these features one by one.

4.2.3.4.1 Minutia Type and Minutia X, Y Coordinates

The minutiae were extracted using Crossing numbers algorithm [30] discussed in the previous section. According to the algorithm if the CN is equal to one then it corresponds to a ridge ending, and if the CN is equal to three then it corresponds to a bifurcation. And also after fulfilling any of the above the condition the X, Y coordinate of the minutia are also available. The information for these features is stored during the minutiae extraction phase.

4.2.3.4.2 Distance

From the center minutia the distance from all other minutia is calculated. The method used for the calculation of distance is the Euclidean distance. The distance between the coordinates of two minutiae A(x1, x2) and B(y1, y2) is given by equation 4.9.

$$Dis\tan ce = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad \textbf{(4.9)}$$

After the calculation of these distances the next step is to find the 5 nearest neighbors. So the calculated distances are sorted. After sorting, the 5 nearest minutiae are selected as a neighborhood for this central minutia. The extraction of all following features depends on these neighboring minutiae set.

4.2.3.4.3 Relative Angle

Originally as advised by the Wahab [1] the relative angle should be between two neighboring minutiae. And it is fairly clear from the Figure 4.9 by [1] that the relative angle is the angle calculated between any two neighboring minutiae.

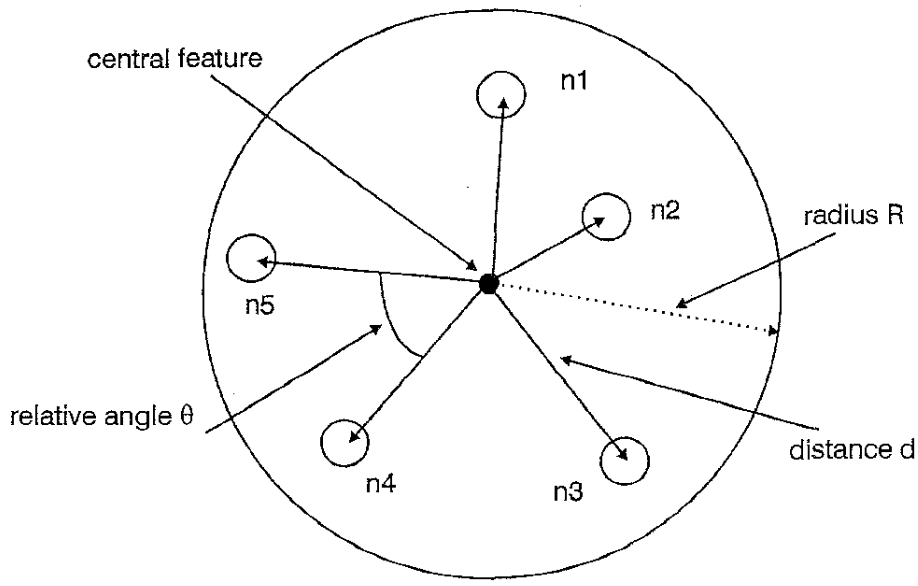


Figure 4.9: Calculation of different features [1]

But calculating relative angle using this method has some serious shortcomings that are:

- During the phase of minutia extraction for different images if the test image is little bit change from the template then due to various reasons it can produce some new minutiae. And due the detection of these new minutiae the neighborhood across same central minutia of the two different images can change.
- As the relative angle is calculated between the neighboring minutiae as advised by [1], so the angle for a neighboring minutia can change due to the occurring of these new minutiae. And it causes problem during the matching stage and this important feature becomes useless.

To overcome this problem the relative angle for each neighboring minutia is calculated from center. A new point called point X is calculated by tracing the ridge of the central minutia by 10 pixels. The selection of this new point had some serious issue of the traversal path selection. So that issue was resolved by selecting the directional side of the ridge. Then the relative angle for each of the neighboring minutia is calculated with the reference of this new point X. Figure 4.10 shows how the relative angle is calculated using this approach.

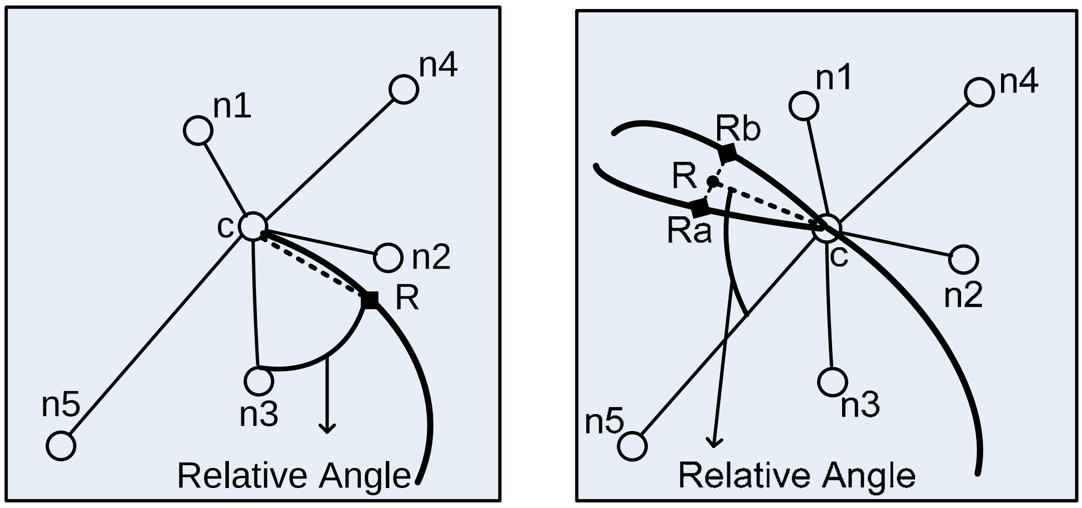


Figure 4.10: Calculation of relative angle

Let’s see how this new approach actually worked to solve the above mentioned problems.

- As the relative angle is dependent on the reference point detected on the ridge of the central minutia it is not dependent on any neighboring minutia.
- Calculating relative angle using this approach made this feature rotational invariant. As the ridge of the central minutia will also rotate so as neighboring minutiae.

Algorithm (*Calculate ridge point*)

Begin

1. *For each minutiae point*
    a. *Make a (21*21) rectangle around it in such a way that it is at the center location (11,11)*
    b. *Construct a 3*3 window around this central location*

c. *Find the detection of any new minutia (ridge ending or bifurcation) while traversing and applying CN method on forwarding window*

d. *If non of them is detected then move window to the next pixel and continue ridge traversal until 10 pixels are traversed*

e. *If the central minutia is a ridge ending then traverse on this ridge until 10 pixels traversed or any minutia detected. Mark point after completing traversal and save its coordinate.*

f. *If the central minutia is a bifurcation then we have 3 ridges to go. Traverse on all 3 ridges one by one and find new points on them either after traversing 10 pixels or due to occurrence of new minutia. Find the closest 2 points among 3 and take their mean to find a new imaginary point. Mark point and save its coordinate.*

2. *The new points for all central minutiae are obtained and are used for calculation of relative angle.*

End

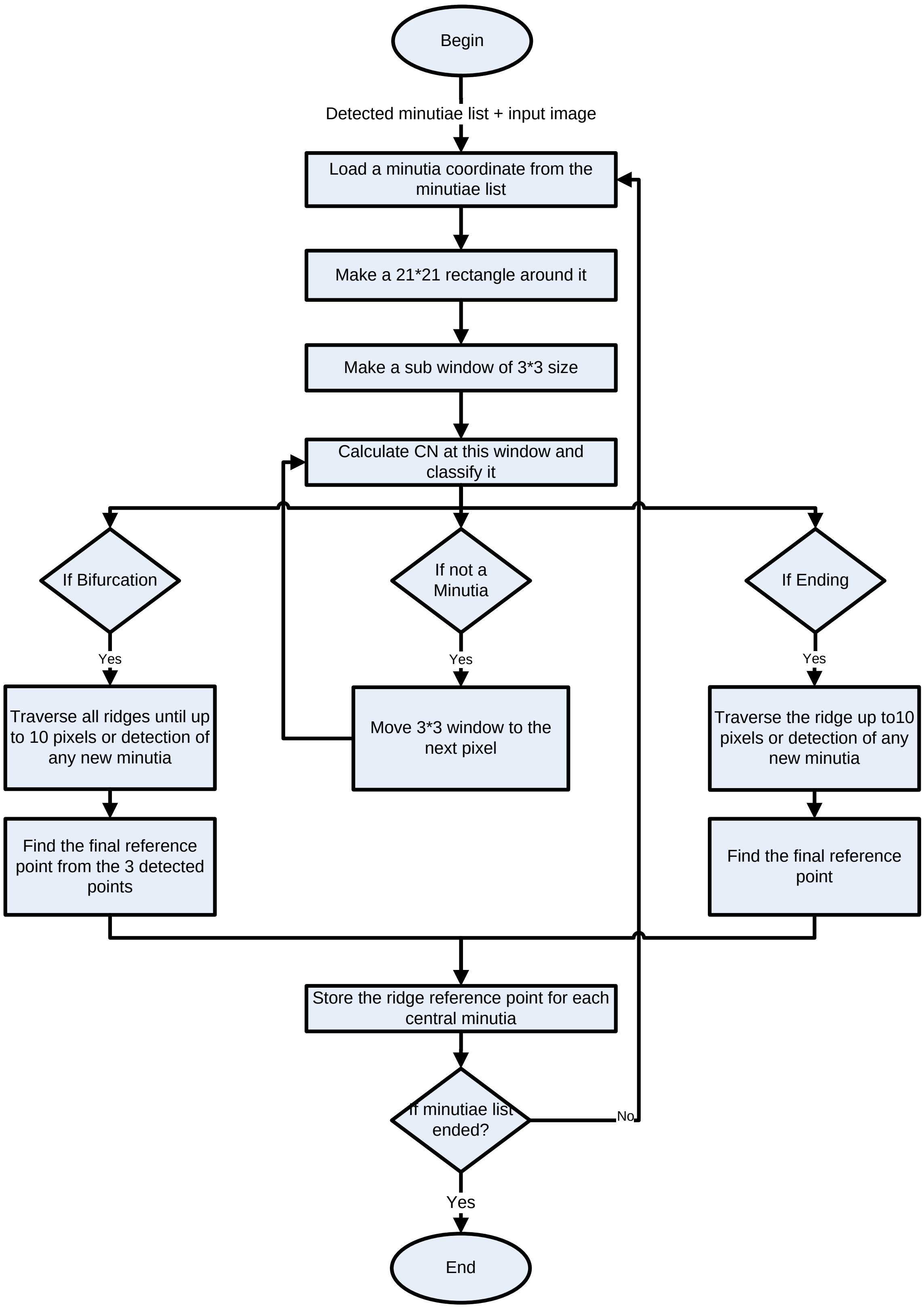


Figure 4.11: Flow chart for ridge point calculation

4.2.3.4.4 Ridge Count

The feature of ridge count shows the number of ridges pass between two minutiae and is calculated using following approach. The ridge count is derived by counting the number of transitions from white to black pixels as traveling on the straight line joining the central feature with the neighborhood feature.

Algorithm

Begin

1. *Select the central minutia*
2. *For each neighboring minutia*
3. *Start traversing pixels from central to a neighboring minutia in a straight line*
4. *Count the total transitions from white to black*
5. *Store the total number of ridge counts for that neighboring minutia.*

End

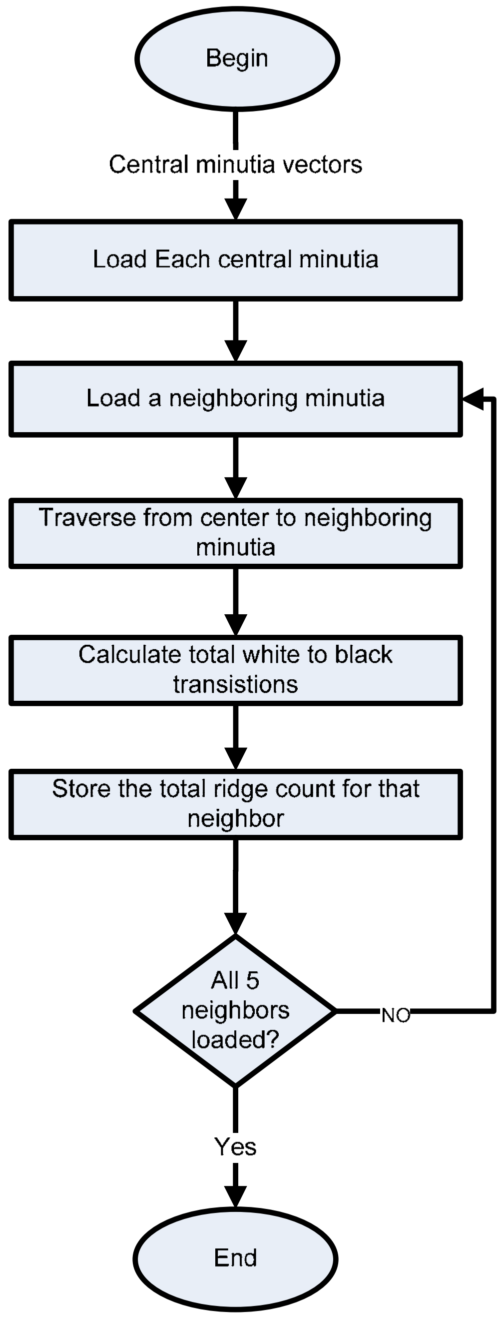


Figure 4.12: Flow chart for Ridge count calculation

4.2.4 Standard Deviation Based Features and Energy Based features

In probability and statistics, the standard deviation of a probability distribution, random variable, or population or multi-set of values is a measure of the spread of its values. It is usually denoted with letter σ. It is defined as square root of the variance.

All operations are performed on cropped image to get the feature vector based on standard deviation. The wavelet transformation is directly performed on the cropped image without any decomposition. Let's say that the level of decomposition is represented by J. As no decomposition is performed so we set the level at J=0. After applying transformation there are three detailed sub-images and one approximated sub-image as shown in equation 4.10.

$$[a_J, \{d_j^1 d_j^2 d_j^3\}_{\ j=1,\ldots\ldots,J}] \tag{4.10}$$

The standard deviation is now calculated from the horizontal, vertical and diagonal details referred by $\{d_j^1 d_j^2 d_j^3\}$. Following formula is used for deviation:

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \bar{x})^2} \tag{4.11}$$

Where $\bar{x}$ is the arithmetic mean. So for one minutia there will be 3 features based on the standard deviation as shown in Figure 4.13.

0	1
2	3

Figure 4.13: Extraction of 3-Features after Applying Wavelet Transformation

As told above the standard deviation is found for all details except low level details thus resulting in a total of 3 features. The application of wavelets at this cropped image reflects its specific recurrent ridge-valley structure. The recurring ridge-furrow structure makes the wavelet coefficients of fingerprint image different from most natural images. Therefore, the energy distribution and orientations is a quite informative feature for fingerprint identification. The energy of different sub-bands gives information regarding both the edge spatial frequency as well as the ridge orientation. So a compact representation of this information based on the standard deviation of each wavelet sub-image is being used as a feature vector for the classification purposes. This information is then utilized for the classification of the input image.

The feature vectors extracted for each minutia have discriminatory properties for different fingerprint images as shown in Figure 4.14.

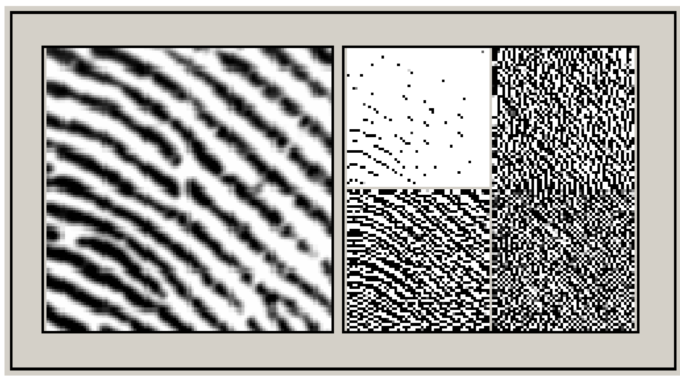


Figure 4.14: Standard Deviation Based Features for each minutia point [35]

Algorithm *(Wavelet transformation)*

Begin

1. *Determine the user selected wavelet type*
2. *Get the corresponding filters values*
3. *Crop the 101*101 image across each central minutia by keeping it in middle*
4. *Make the rotated versions of the main cropped image of size 101*101*
5. *Make a set of small cropped images of size 35*35 from all versions of the main cropped image*
6. *Apply wavelet transformation on each of small cropped image. It will reveal the 3 high level detail sub images and one approximated low level sub-image.*
7. *Calculate the standard deviation of 3 detailed sub-images obtained after applying wavelets transformation on each version of small cropped*

image. Save 3 values resulted from the calculation of STD for each detailed sub-image for this minutia.

8. *Calculate the Energy of 3 detailed sub-images obtained after applying wavelets transformation on each version of small cropped image. Save 3 values resulted from the calculation of Energy for each detailed sub-image for this minutia.*
9. *Move to the next central minutia*

End

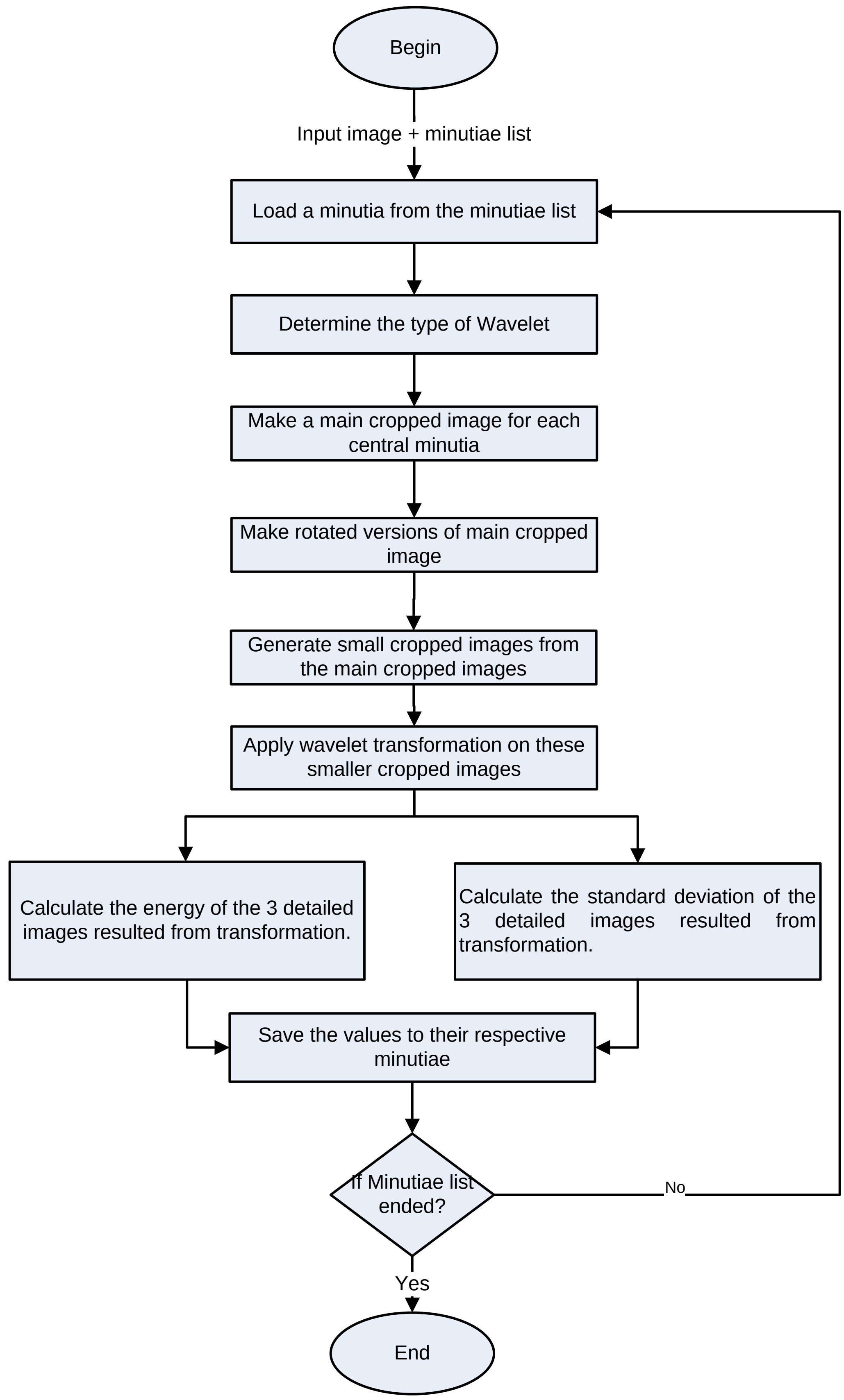


Figure 4.15: Flow chart for Extraction of Wavelet Features

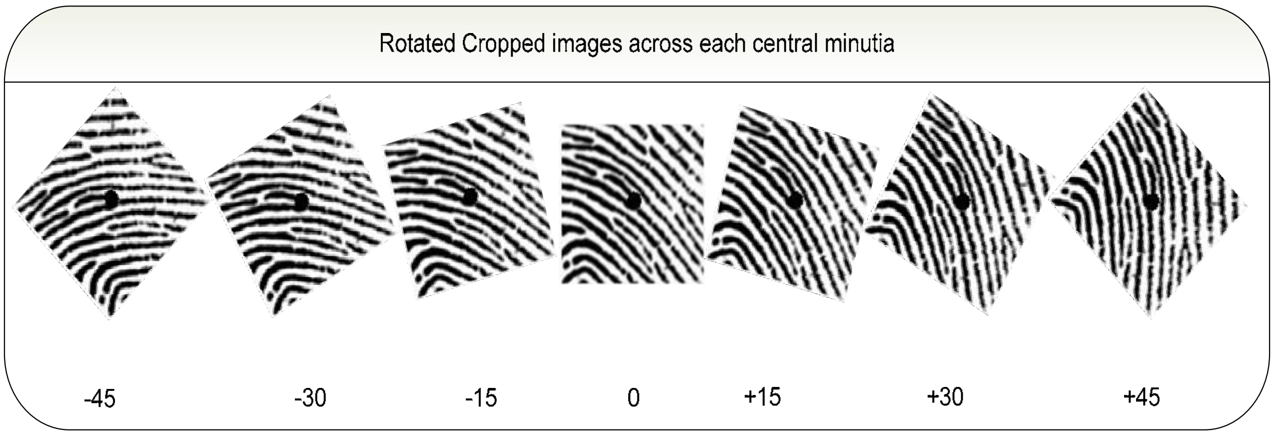


Figure 4.16: Rotated version of the main cropped image of size 101*101

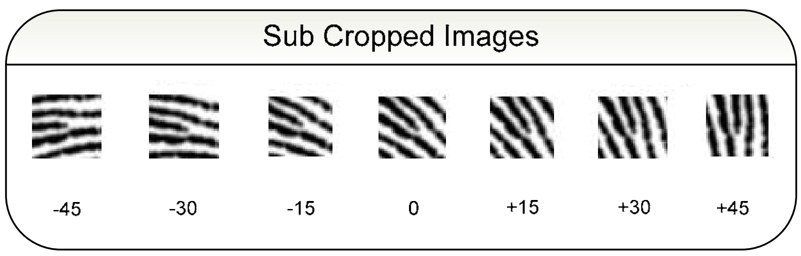


Figure 4.17: Sub-images of size 35*35 taken from their respective main cropped image

4.3 Summary

This chapter provides the implementation details of this thesis work and describes the adopted approaches for materializing the earlier studied theoretical baseline. This baseline knowledge was obtained after in depth study of the related research work discussed in Chapter 2 and explained in Chapter 3. The adopted techniques and some related issues are explained in depth in this chapter.

Chapter 5

RESULTS AND ANALYSIS

5.1 Introduction

In this chapter, the results of the algorithms and techniques, given in Chapter 4, have been presented. The analysis made here is by keeping in mind the main objective of this work. As the main focus of this work was to make a system which can provide rotation and translation invariance and can also keeps in mind to have maximum accuracy in recognition. Moreover for every efficient fingerprint matching system, it is necessary that it should generate good matching performance with faster speed.

As in my work efforts are made to achieve maximum accuracy so a very rich features set is obtained from an image that covers the features from spatial and frequency domains. The fingerprint data base that is used for matching is FVC 2002. In this database there are 100 different fingers with 8 impressions per finger. The resolution of the images is (388 x 374). In this database for every fingerprint there are 7 different variations to the first one. These are actually the rotated and translated versions of the original fingerprint images etc as shown in Figure 5.1.

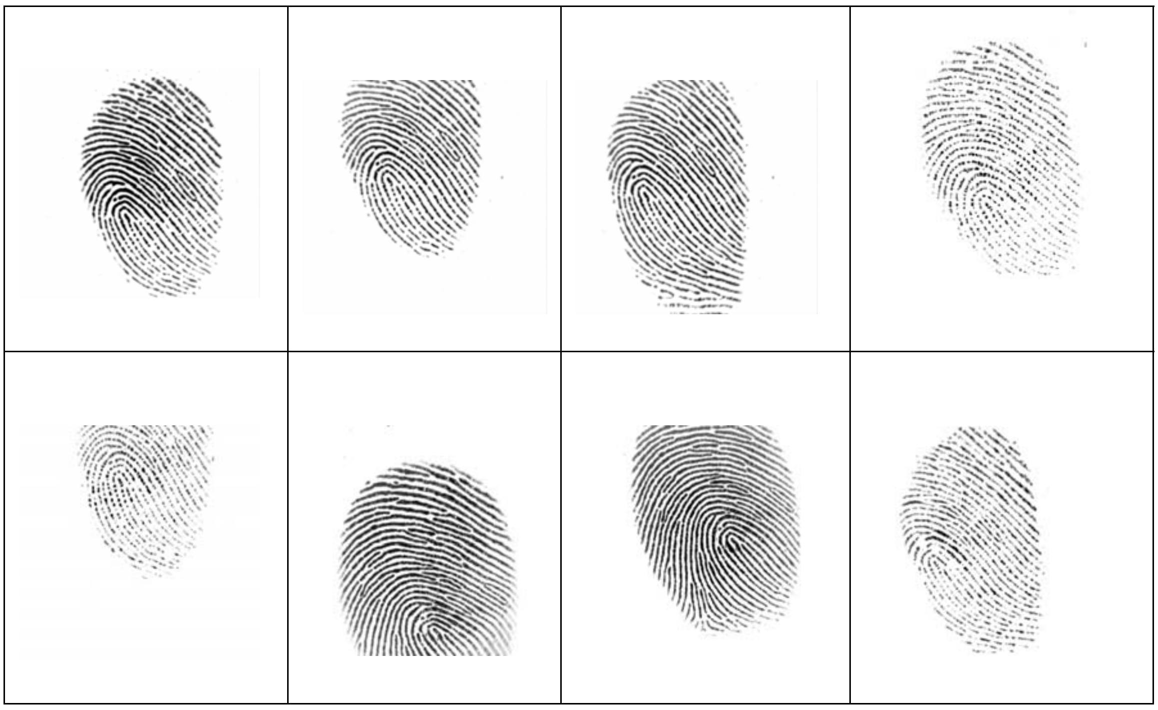


Figure 5.1: Different variation of same fingerprint

5.2 Proposed System

The system that I designed is actually a minutiae based system. Using minutiae based features during classification produced good accuracy and such systems are also good for larger databases. I further improved the performance of this approach by adding some new features that are actually from frequency domain. These new features surpluses the performance by a large factor. Frequency domain features are obtained after applying wavelet transformations. These features are intelligently used during different stages of classification process.

For preprocessing the preprocessor that is actually used is as suggested by Hong et al. [11] and Raymond [12]. Some modifications are also made in the preprocessor by adding some new morphological operations. These operations are also enhanced the over all image quality and helped in correctly identifying the minutiae.

5.3 System Formulation

To test the system behavior before and after adding the new features I divided the system into four set of formulations. Initially the system that was implemented only had minutiae based features and it was given the name as System 1.Later the classification process was solely carried out using only features that are extracted after applying wavelet transformations. This formulation was named as System 2. A third system was also formed by only the Standard deviation based features in minutiae based features. This system was named as System 3. System 4 combines the Energy based features with minutiae based features and System 5 finally includes all the minutiae based and wavelets based features during classification. Table 5.1 shows that formulation and also the maximum accuracy attained is also shown.

Formulation	Featrues	Output Accuracy
System 1	Minutia Based Features	84.75%
System 2	Standard Deviation Based + Energy Based Features	65.50%
System 3	Minutia Based + Standard Deviation Based Feature	87.25%
System 4	Minutia Based + Energy Based Features	91.50%
System 5	Minutia Based + Standard Deviation Based + Energy Based Features	~~95~~94.75%

Table 5.1: Formulation of five different systems to assess the performance of the features set extracted.

Table highlights the important issues which have been learnt during this research work that Minutia Based features system alone did not performed quite well as claimed by Wahab [1]. And the accuracy that was achieved was in fact about 85% which was surely not good enough. There were some shortcomings of this research work that can be identified. Firstly the preprocessor that was used by Hong et al.[11] and Raymond [12], did not perform the enhancement quite well for poor noisy images and resulted into large number of false minutiae in the system. Moreover due to the improper enhancement in the fingerprint image, features like minutia type and ridge count partially became non functional in actual scenario. As far as distance among minutiae was concerned it also showed significant variation for rotated version of same finger print. The direction feature on the other hand was calculated across any of the two axis x or y. So in case of rotated images, this feature also did not perform. So keeping in view all that shortcomings, attaining accuracy more than 90% looks quite impractical as claimed by [11].

So it was needed that some other features should be added in the system that could resolve the problem of such a low accuracy. So two new types of features were added that are actually extracted after applying wavelets. The two new types of features are i.e. Standard Deviation and energy based features. Both these features can be obtained directly from gray scale image without the need of any preprocessing steps. But in my scenario these features are actually dependent on extracted minutiae. So they are extracted after minutiae extraction phase. The performance of the system was analyzed after adding these features one by one as shown in Table 5.1.

5.4 Proposed System Success Rate

Any fingerprint matching system typically returns a matching score that quantifies the similarity between the input and the database template image. The higher the matching score the more certain is that the two fingerprints come from the same finger and large numbers of matches are actually found between the two prints. On the other hand the lesser the score is, the smaller is the system confidence that the two fingerprints come from the same finger and less number of matches found between the two prints. To see that whether the distance between features is close enough to the required level a threshold is set. And if the difference of distance is less or equal to that threshold then it is considered as matched feature.

5.5 Biometrics system errors

A typical biometric verification system commits two types of errors: mistaking biometric measurements from two different fingers to be from the same finger (called false acceptance) and mistaking two biometric measurements from the same finger to be from two different fingers (called false rejection) and their rates are normally calculated. There is a strict tradeoff between FAR and FRR in every biometric system [5]. In fact, both FAR and FRR are functions of the system threshold t, and we should, therefore, refer them as FAR(t) and FRR(t), respectively. If t is decreased to make the system more tolerant with respect to input variations and noise, then FAR(t) increases; vice versa, if t is raised to make the system more secure, then FRR(t) increases accordingly. Whole the evaluation process is carried out in two steps. Firstly total *genuine attempts* are calculated that is each fingerprint image is compared to the remaining images of the same finger. Secondly *impostor attempts* are calculated that is first impression of each finger is compared to the first image of the remaining fingers. For this database there are total of 2800 genuine attempts and 4950 impostor attempts.

A Receiver Operating Curve (ROC) is a plot of False Rejection Rate (FRR) against False Acceptance Rate (FAR) for all possible values of thresholds. It measures overall performance of the system. Figure shows performance of my matcher in terms of FAR(t) and FRR(t) parameters.

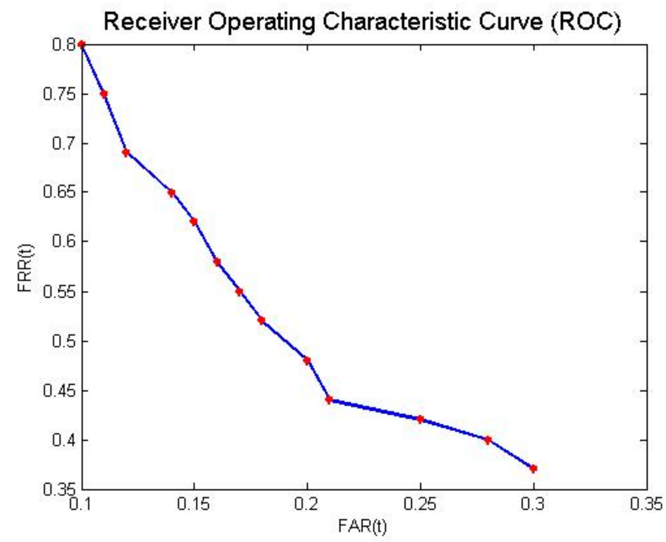


Figure 5.2: System ROC curve FRR(t) vs. FAR(t)

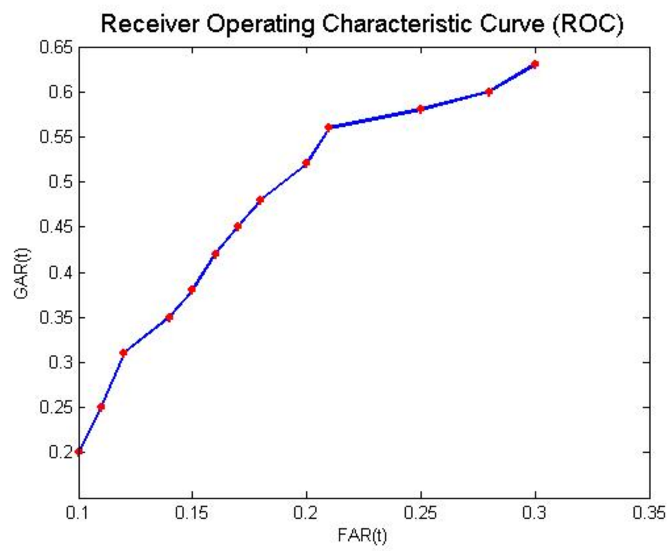


Figure 5.3: System ROC curve GAR(t) vs. FAR(t)

There is another evaluation method that is used to assess the accuracy and performance of any automated fingerprint verification system is in terms of its total success rate. In proposed system as there are some wavelets based features that works along with other spatial domain features. Wavelets based features are obtained by applying different wavelets and then different thresholds are employed for them to identify the Total Success rate as shown in Table 5.2. By looking at table it is clear that system provides an accuracy of about 95 % at maximum. However, this accuracy can be increased if a more effective preprocessor along with some method of false minutiae removal step are adopted.

No.	Filter	Total Success Rate
1	Minutia + Wavelets based features ‘Db1’ wavelets	92.50%
2	Minutia + Wavelets based features ‘Db3’ wavelets	93.25%
3	Minutia + Wavelets based features ‘Db5’ wavelets	89.75%
4	Minutia + Wavelets based features ‘Db8’ wavelets	94.75%
5	Minutia + Wavelets based features ‘Sym2’ wavelets	90.75%
6	Minutia + Wavelets based features ‘Sym3’ wavelets	88.75%
7	Minutia + Wavelets based features ‘Sym4’ wavelets	92.90%
8	Minutia + Wavelets based features ‘Sym5’ wavelets	91.25%
9	Minutia + Wavelets based features ‘Coif3’ wavelets	90.75%
10	Minutia + Wavelets based features using ‘ Coif 4’ wavelets	88.25%
11	Minutia + Wavelets based features ‘ Coif 5’ wavelets	89.75%

Table 5.2: System Success Rate obtained from different families of wavelets

Parameter	**FC100**	**Fingerscan**	**Previous System by [2]**	**Our System**
Enrolment time	10s	25s	10s	<10s
Verification time (One-on-one matching)	1-2s	<0.5s	<1s	<1s
False Acceptance rate (FAR)	<0.0001%	<0.0001%	Unknown	<3%
False Rejection Rate (FRR)	<1%	<1%	<1%	<8%

Table 5.3: System comparison with previous approaches

Table 5.4: Comparison with FVC 2002 systems

No.	Filter	Total Success Rate
1	Minutia + Wavelets based features ‘Daubechies’ wavelets	94.75%
2	Minutia + Wavelets based features using ‘Symlets’ wavelets	92.90%
3	Minutia + Wavelets based features ‘Coiflets’ wavelets	91.25%

Table 5.5: Maximum System Success Rate obtained from each family of wavelets

5.6 Computational Cost

In the previous section the accuracy of the system has been analyzed. Another important method to evaluate any fingerprint matching system is in terms of its efficiency. As it is required that the running time of the system must be less enough so that the system can be considered as practically implement able. So for this system the running times (in seconds) have been computed for different modules in order to show the efficiency. To produce a trust worthy results the performance of the system has been checked on two **environments** namely **A** and **B. Environment A** is a 1.6 GHz and **environment B** is 2.0 GHz Pentium IV machines with 512 MB of RAM.

5.6.1 Feature Extraction Module

The feature extraction module is one of the most important modules. So in the following section time analysis has been provided related to this module. The main goal is to find out that how much time it takes to extract different features. Feature extraction is required for both training and testing images. As described earlier over all the features are divided in two kinds. So it requires different time to extract both types of features. As the spatial domain features that are minutiae based require that the minutiae had already been extracted while the other features that are wavelets based features and are extracted after applying wavelet transformation directly on the gray scaled image. This feature is only extracted for those images that reach in the second stage of matching. So it is quite important to analyze the time that is required to extract both features. First of all the time required to compute wavelets based features for different families is shown in seconds in Table 5.6.

No.	Filter	Environment A	Environment B
1	Wavelets based features using 'Db1' wavelets	1.00	0.75
2	Wavelets based features using 'Db3' wavelets	0.90	0.68
3	Wavelets based features using 'Db5' wavelets	0.80	0.60
4	Wavelets based features using 'Db8' wavelets	0.85	0.70
5	Wavelets based features using 'Sym2' wavelets	1.25	0.95
6	Wavelets based features using 'Sym3' wavelets	0.95	0.60
7	Wavelets based features using 'Sym4' wavelets	0.85	0.65
8	Wavelets based features using 'Sym5' wavelets	1.00	0.80
9	Wavelets based features using 'Coif3' wavelets	1.35	0.95
10	Wavelets based features using 'Coif 4' wavelets	0.90	0.75
11	Wavelets based features using 'Coif 5' wavelets	0.95	0.70

Table 5.6: Computation Time for Wavelets based Features (in seconds)

So from Table 5.6 it is clear that the overall feature extraction time in all kind of wavelets is less than 2 second.

The second features that are spatial domain features are extracted in two stages. Initially only some features are extracted that are Center minutia coordinates, minutia type, Distance from Centre minutia, Ridge count and relative angle etc. All these feature are only required during first stage of matching. Then if the matching score is good enough that it requires to perform second stage matching then some new features are calculated. These features are minutia direction, relative angle and distance from the neighborhood minutia etc. So the running time for the features extraction of both stages are calculated separately and listed as under. These facts are shown in Table 5.7 and 5.8.

Spatial Domain Features	1st Stage features (Environment A)	1st Stage features (Environment B)
Ridge Count	0.25	0.10
Relative Angle	0.85	0.75
Distance	0.90	0.55

Table 5.7: Spatial Domain Features Computation Cost for stage 1 of classification (in seconds)

Spatial Domain Features	2nd Stage features (Environment A)	2nd Stage features (Environment B)
Direction	0.75	0.55
Relative Angle	0.51	0.45
Distance	0.65	0.30

Table 5.8: Spatial Domain Features Computation Cost for stage 2 of classification (in seconds)

5.6.2 Classification and Authentication Module

This is another very important module. The time complexity of this module affects the overall system efficiency. The Classification and Authentication Module performs all the matching of features and decides that whether any features is matched by looking at its threshold. At the end of matching an overall image score is determined that tells that whether an image is matched or not. The classification is performed in two stages i.e. Stage 1 and Stage 2. If the matching score after 1st stage is greater than the minimum required number of matches then the system enters into 2nd stage of matching. So it is very important that the time complexities of each stage are analyzed separately.

5.7 Summary

As from this detailed analysis it can be deduced quite clearly that the adopted approach has been proved to be highly successful in automatic fingerprints verification system. The choosing of suitable approaches after detailed analysis has made the overall system robust. Three feature vectors from two different domains have provoked the benefits of both domains and when they are used in integration, it provided best recognition for different types of fingerprints. Also there have been mentioned few shortcoming of previous research. The time analysis at the end indicates the efficiency of the proposed system for real time environment. The hardware implementation of the proposed algorithm will be faster than the result produced by P-IV platform and thus can be used in real time applications.

Chapter 6

CONCLUSION AND FUTURE WORK

6.1 Overview

This research was aimed towards the efficient classification of fingerprint images. As the main requirement of a fingerprint recognition system is its fast processing and precise decision making. The proposed methodology has been implemented on a desktop PC in Microsoft Windows operating system and was implemented in MATLAB. The main focus of this work was to obtain a robust feature set that can operate well when the input image is noisy and corrupted. This feature set was actually obtained by including the features from spatial and frequency domain. These features were then fused while using for classification process. The system was run and tested on standard fingerprint databases. To make the results more trust worthy the system performance was checked on different platforms of different specifications. During the testing it was found that mixing these two types of features actually gave really good results. There were some procedures in the proposed framework that were computationally little bit expensive. However, considering the total contribution these features made in increasing the total reliability and accuracy of the system it is justified either to save the computational time. Moreover for actual implementation, the technique can be implemented on real time processing hardware. However in the current scenario the program takes less than 1 second on a Pentium IV machine i.e. 1.6 GHz with 512 MB RAM to recognize a fingerprint.

This system of fingerprints recognition and classification is robust enough that it makes the system rotation and translation invariant with maximum accuracy.

6.2 Future Work

The proposed method of automated fingerprint matching has provided the platform for achieving maximum accuracy with considerable amount of less time in the field of fingerprint based biometric systems.

Talking about fingerprint recognition techniques researcher have worked in almost every aspect of fingerprints. But they actually failed to make an ideal system due to many reasons. Some time the problem is with the sensors, some time the preprocessor is not good enough that fails to extract features from the fingerprint correctly. Due to which the benefit of even a very good set of feature vectors defined for classification is not attained. And some time the problem lies with the less strong features vectors set. So all of these phases are very much dependent on each other. The new research area on which mostly researchers are interested are level-3 features that are pores on a fingerprint ridge. These features are very handy in fingerprint recognition. Again the extraction of these features is very difficult because they require very high quality images of up to 1000 dpi. Normal sensors that are commonly in use can produce up to 500 dpi image.

The preprocessor and feature extraction phases adopted in this work takes some time due to the richness of feature set. But the time it takes to extract them does not fit this system in some scenarios where it is required that the system should give results quickly and running on very large database of fingerprint. So for that kind of scenario current system can be used in a bit different manner. That is it must not be used for all fingerprint images specially that images which are in good quality. As we know that these images can be correctly identified even with a reduced features set. But for all that images which are poor in quality or corrupted due to some noise the much enhanced features set used in current system can be adopted for their correct classification.

Also over all system running time (Both feature extraction and matching) can also be reduced by doing some variations in the current system. As the extraction of minutiae and energy based features takes considerable amount of time a scheme can be adopted to reduce that time. What if before extraction of all types of features if some image based classification algorithm is run on the system. If this algorithm gives reasonable results then its worthy to proceed by extraction of enhanced features otherwise it is unnecessary to extract them.

The distance based classification approach used in this research work is quite simple but fast. That is compelled by the need for real time decision making. However there are other classifiers which provide good recognition rates. A further improvement can be studied in the classification area of the research by the use of a 'Hybrid Classification' approach based on the utilized classifiers or any other new type of classifiers. Moreover Energy feature vector has provided best results in this research work. So focus should be made in the future to identify the different ways to extract the energy based features form the images to get the good recognition rates not only in fingerprints but in other biometric systems too. This will result in an improved and efficient success rate due to the reduction of the complexity.

Appendix A

FEATURE EXTRACTION MODULE ALGORITHM

procedure [Minutia_Vector**] = Feature_Extraction(Image A, MinutiaeDetails)**

begin

//Input of this function is List of Minutia Detected

// The first step is to calculate the distance of each minutia from all others

Distance_Vector=CalculateDistance(MinutiaeDetails);

//Next we have to select the 5 nearest as each minutia neighborhood. So first

//there is need to sort those distances

Minutia_Vector=Sorting(Distance_Vector, MinutiaeDetails);

// Minutia_Vector contains the 5 neighborhood structure for each minutia

// Now relative angles are calculated for each 5 neighborhood minutia

//structure

Minutia_Vector =RelativeAngle(Minutia_Vector);

// Now Ridge Count is count between central and neighboring minutia

Minutia_Vector = RidgeCount(Minutia_Vector, A);

//Computation Wavelet based features

//Computation of Features based on Standard Deviation and Energy

//Energy and deviation features are 3 in length

Initialize a vector Feature_std of size 5*3.

Initialize a vector Feature_energy of size 5*3.

for (each Minutia)

Crop the 101*101 sub image across each minutia

Apply wavelet transformation on sub image.

```
        Find standard deviation and energy of all details except the approximation
        details.
        //Store the 3 features from the temporary arrays of Std_Feature and
        //Egy_Feature into global feature arrays
        for (start i from 1 and to 5)
            for (start j from 1 and to 3)
                Features_std (1, j, i) = Std_Feature(1,j);
                Features_energy (1, j, i) = Egy_Feature(1,j);
            end
        end
    end
// Also assign these two feature vectors to Minutia_Vector
Assign Fetaures_std vector to Minutia_Vector.
Assign Fetaures_energy vector to Minutia_Vector.


end
```

BIBLIOGRAPHY

[1] A. Wahab, S.H.Chin and E. C. Tan. “A Novel approach to automated fingerprint verification” IEE Proc -Vis Image Signal Process, Vol 145 No 3 June I998

[2] N. K. Ratha, V. D. Pandit, R. M. Bolle, and V. Vaish. “Robust fingerprint authentication using local structure similarity” In Workshop on applications of Computer Vision, pages 29–34, 2000.

[3] Tsai-Yang Jea and Venu Govindaraju. “A minutia-based partial fingerprint recognition system” Submitted to Pattern Recognition, 2004.

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

[5] D. Maio and D. Maltoni. “Direct gray scale minutia detection in fingerprints” Transactions on PAMI, 19(1), 1997.

[6] Lee, C.J., and Wang, S.D. “Fingerprint feature extraction using Gabor filters” Electron. Lett., 1999, 35, (4), pp.288-290.

[7] M. Tico, P.Kuosmanen and J.Saarinen. “Wavelet domain features for fingerprint recognition” Electroni. Lett., 2001, 37, (1), pp.21-22.

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

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

[10] D. Maltoni, D. Maio, Anil Kumar Jain and S. Probhaker. “Handbook of Fingerprint Recognition” Springer Verlag, CSC Press, New York, USA, 2003

[11] Hong, L., Wan, Y., and Jain, A. K. “Fingerprint image enhancement: Algorithm and performance evaluation” IEEE Transactions on Pattern Analysis and Machine Intelligence 20, 8 (1998), 777–789.

[12] Raymond Thai Honors Thesis, “Fingerprint Image Enhancement and Minutiae Extraction”, 2003

[13] Sherlock, D. B. G., Monro, D. M., and Millard, K. “Fingerprint enhancement by directional Fourier filtering” In IEE Proc. Vis. Image Signal Processing (1994), vol. 141, pp. 87–94.

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

[15] D. S. Zorita, J.O. Garcia, S. C. Llanas and J.G. Rodriguez, "Minutiae Extraction Scheme for Fingerprint Recognition Systems", Proceedings of Seventh International Conference on Image Processing, Vol 3, No.10, pp. 254-257, Tesalónica, Greece, 2001

[16] M. Arantes, A. N. Ide and J. H. Saito, "System for Fingerprint Minutiae Classification and Recognition", Proceedings of the Ninth International Conference on Image Processing (ICONIP 2002), Vol. 5, pp. 2474-2478, Orchid Country Club, Singapore, 2002

[17] C. J. Lee and S. D. Wang, "Gabor Filter-Based Approach to Fingerprint Recognition", Proceedings of IEEE Workshop on Signal Processing Systems (SiPS 99), pp. 371-378, Taipei, Taiwan, 1999

[18] C. J. Lee and S. D. Wang, "Fingerprint Features Extraction Using Gabor Filters", Electronics Letters, Vol. 35, No. 4, pp. 288-290, 1999

[19] Marius Tico, Pauli Kuosmanen, ”An Algorithm for Fingerprint Image Postprocessing” 0-7803-65 14-3/00/$10.0002000 IEEE

[20] Xudong Jiang and Wei-Yun Yau. “Fingerprint minutiae matching based on the local and global structures” In International Conference on Pattern Recognition, pages 1038–1041, 2000.

[21] Simon-Zorita, D., Ortega-Garcia, J., Fierrez-Aguilar, J., Gonzalez-Rodriguez, J.: “Image quality and position variability assessment in minutiae-based fingerprint verification” IEE Proceedings - Vis. Image Signal Process. 150 (2003) 402–408

[22] Mehtre, B. M. “Fingerprint image analysis for automatic identification-Machine Vision and Applications” 6, 2 (1993), 124–139.

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

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

[25] S. Mallat, “A Wavelet Tour of Signal Processing”, Academic Press, New York, USA, 1998

[26] Sharat S. Chikkerur PhD thesis, “Online Fingerprint Matching System”, June 2005

[27] Seonjoo Kim, Dongjae Lee, and Jaihie Kim, “Algorithm for Detection and Elimination of False Minutiae in Fingerprint Images” J. Bigun and F. Smeraldi (Eds.): AVBPA 2001, LNCS 2091, pp. 235–240, 2001.

[28] Luo, Jie Tian and Yan Wu, “A Minutia Matching Algorithm in Fingerprint Verification”, 0-7695-0750-6/00 2000 IEEE

[29] Finger print Verification Contest 2002; FVC2002, http://bias.csr.unibo.it/ fvc2002/

[30] Tamura, H. “A comparison of line thinning algorithms from digital geometry viewpoint” In Proc. of the 4th Int. Conf. on Pattern Recognition (1978), pp. 715–719.

[31] Jason Zalev, and Reza Sedaghat, “Complex Energy Signal Model for Digital Human Fingerprint Matching”, proceedings of world academy of science, engineering and technology volume 13 may 2006 issn 1307-6884

[32] Linda Shaprio and George Stockman, “Computer Vision Book”, Michigan State University, USA, 2000

[33] S. Mallat, “A Wavelet Tour of Signal Processing”, Academic Press, New York, USA, 1998

[34] Peng Shi, Jie Tian, Qi Su, and Xin Yang, “A Novel Fingerprint Matching Algorithm Based on Minutiae and Global Statistical Features”, 978-1-4244-1597-7/07/$25.00 ©2007 IEEE

[35] Nabeel Younus, Master Thesis, “Efficient Fingerprint Matching Technique Using Wavelet Based Features” 2007

[36] Biometric System Lab., University of Bologna, Cesena-Italy, http:// csr. unibo.it/research/biolabl/

[37] T.B.J. Andrew and N.C.L. David, "Integrated Wavelet and Fourier-Mellin Invariant Feature in Fingerprint Verification System", Proceedings of the ACM SIGMM workshop on Biometrics methods and applications (WMBA'03), No.1, pp.82-88, California, USA, 2003

[38] Mr. Zhi Li Wu, “Fingerprint Recognition using Minutia Details”, PhD Thesis, Baptist University of Hong Kong, Hong Kong, 2002

[39] Sen Wang, Wei Wei Zhang and Yang Sheng Wang, “Fingerprint Classification by Directional Fields”, Proceedings of the Fourth IEEE International Conference on the Multimodal Interfaces (ICMI’02), Vol. 2, No. 5, pp.395-399, Pennsylvania, 2002

[40] S. Tachaphetpiboont and T. Amornraksa ,”Applying FFT Features for Fingerprint Matching”, Proceedings of the First International Symposium on Wireless Persuasive Computing,Vol.1, pp.1-5, Phuket, Thailand, 2006

[41] M. Tico, E. Immonen, P. Ramo, P. Kuosmanen, and Saarinen, “Fingerprint Recognition using Wavelet Features”. Proceedings of IEEE International Symposium on Circuits and Systems, Vol 2, pp. 21 -24, Sydney, Australia, 2001

[42] Mr. Tsai – Yang Jea, “Minutia-Based Partial Fingerprint Recognition”, PhD Thesis, The State University of New York, New York, 2005

[43] Tu Van Le, Ka Yeung Cheung, Minh Ha Nguyen, “A Fingerprint Recognizer Using Fuzzy Evolutionary Programming” Proceedings of the 34th Hawaii International Conference on System Sciences – 2001

[44] Jason Zalev, Reza Sedaghat, “Automatic Fingerprint Recognition Algorithm”, CCECE 2004- CCGEI 2004, Niagara Falls, May/mai 2004

[45] Koichi Ito, Ayumi Morita, Takafumi Aoki, Hiroshi Nakajima, Koji Kobayashi, and Tatsuo Higuchi, “A Fingerprint Recognition Algorithm Combining Phase-Based Image Matching and Feature-Based Matching”, ICB 2006, LNCS 3832, pp. 316–325, 2005.

[46] Davide Maltoni, Dario Maio, Anil K. Jain, Salil Prabhakar, “Handbook of Fingerprint Recognition”