# APPLICATION OF NEURAL NETWORKS IN FINGERPRINT IDENTIFICATION

Thesis submitted in partial fulfillment of the requirements for the award of degree of

**Master of Engineering**
**In**
**Software Engineering**

By:
**Kushal Veer Singh**
**(800831005)**

Under the supervision of:
**Mr. Karun Verma**
**Assistant Professor, CSED**

COMPUTER SCIENCE AND ENGINEERING DEPARTMENT
THAPAR UNIVERSITY
PATIALA – 147004

**JULY - 2010**

# Certificate

I hereby certify that the work which is being presented in the thesis entitled, **"Application of Neural Networks in Fingerprint Identification"**, in partial fulfillment of the requirements for the award of degree of Master of Engineering in Software Engineering submitted in Computer Science and Engineering Department of Thapar University, Patiala, is an authentic record of my own work carried out under the supervision of ***Mr. Karun Verma*** and refers other researcher's works which are duly listed in the reference section.

The matter presented in this thesis has not been submitted for the award of any other degree of this or any other university.

*Kushal Veer Singh*

This is to certify that the above statement made by the candidate is correct and true to the best of my knowledge.

**Karun Verma**
**Assistant Professor**

Computer Science and Engineering Department
Thapar University,
Patiala

**Countersigned by**

15/07/10

**Dr. RAJESH BHATIA**
Head
Computer Science & Engineering. Department
Thapar University
Patiala

**Dr. R.K.SHARMA**
Dean (Academic Affairs)
Thapar University,
Patiala.

# Acknowledgment

I would first like to express my sincere gratitude to my thesis supervisor **Mr. Karun Verma,** Assistant Professor, of Computer Science and Engineering Department for his great positive encouragement, advice, and guidance professionally and personally for the completion of the thesis work. His consistent support and intellectual guidance made me energize and innovate new ideas.

I would also like to express my thanks to all professors and lecturers, and members of the department of Computer Science and Engineering, Thapar University, Patiala for their generous help in various ways for the completion of this thesis.

Finally, I thank my parents, my soul mates, truly and sincerely, for their patience, support and constant encouragement throughout my whole life.

Kushal Veer Singh

*Kushal Veer Singh*
*800831005*
*M.E.(Software Engineering)-2nd year*
*Computer Science & Engineering Department*
*Thapar University*
*Patiala-147004*

# Abstract

The goal of this thesis is to investigate the current techniques for fingerprint identification system. This target can be mainly decomposed into image preprocessing, feature extraction and feature match. For each sub-task, some classical and up-to-date methods in literatures are analyzed. Based on the analysis, an integrated solution for fingerprint identification system is developed for demonstration.

Fingerprint images are rarely of perfect quality. To achieve good minutiae extraction in fingerprints with varying quality, preprocessing in form of image enhancement and binarization is first applied on fingerprints before they are evaluated. We applied histogram equalization method for image enhancement to obtain a more reliable estimation of minutiae locations. To extract minutia points an algorithm based on crossing number method is used. Neural network trained for the location of these minutia points and to improve the performance of the system. An alignment-based elastic matching algorithm has been developed for minutia matching. This algorithm is capable of finding the correspondences between input minutia pattern and the stored template minutia pattern without resorting to exhaustive search. All the implementation work has been done in **MATLAB 7.5.0** Image Processing Toolbox. Performance of the developed system is evaluated on a database with fingerprints from different people.

# Table of Contents

# List of Tables

# List of Figures

# Chapter-1: Introduction

## 1.1 Overview

In an increasingly digital world, reliable personal authentication has become an important human computer interface activity. National security, e-commerce, and access to computer networks are some examples where establishing a person's identity is vital. Most existing security measures rely on knowledge-based approach like passwords or token-based approach such as swipe cards and passports to control access to physical and virtual spaces. However, all over the place, such methods are not very secure. Token such as access cards may be shared or stolen. Passwords and PIN numbers may be stolen electronically. Furthermore, they cannot differentiate between authorized user and a person having access to tokens or knowledge. Biometrics such as fingerprint, face and voiceprint offers reliable personal authentication that can address these problems and is gaining citizen and government acceptance. Among biometrics,fingerprint systems have been one of most widely researched and deployed because of their easy access, low price of fingerprint sensors, non-intrusive scanning, and relatively good performance [2].

## 1.2 What is Fingerprint?

Fingerprints are graphical flow-like ridges present on human fingers is shown in Figure 1.1. They are fully developed under pregnancy and do not change throughout the life except due to serious accidents such as bruises and cuts or surgery on the fingertips. This property makes fingerprints a very attractive biometric identifier. Typically, a fingerprint image is captured in one of two ways [1]:

i. Scanning an inked impression of a finger is shown in Figure 1.1(a)
ii. Using a live-scan fingerprint scanner is shown in Figure 1.1(b).

(a) (b)

Figure 1.1: Fingerprint images (a) inked fingerprint (b) live-scan fingerprint [3].

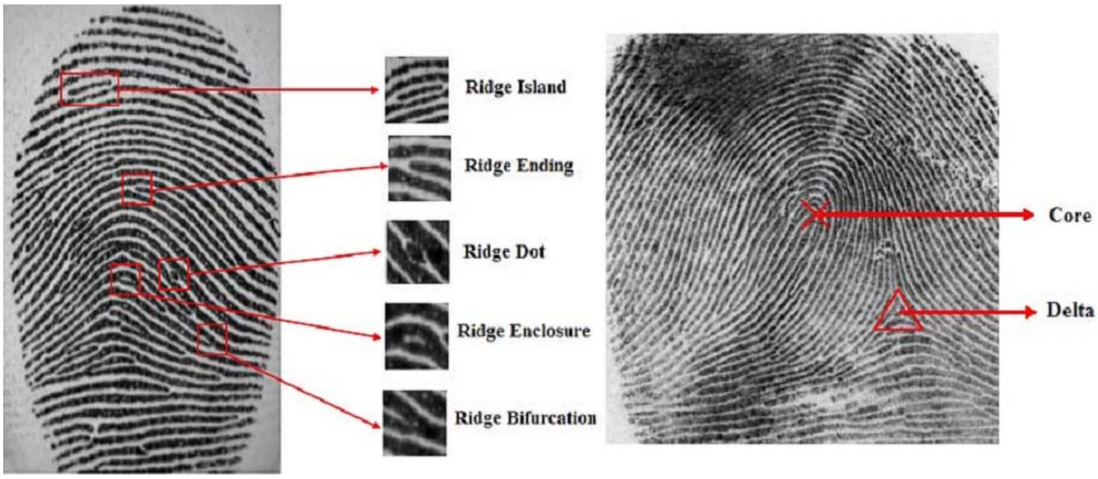
In the above figure dark lines is called ridges and the white area that exists between the ridges is called valley or furrow.

## 1.3 Fingerprint Features

Fingerprint features are those attributes of a fingerprint that may be useful either to classify or to uniquely identify the fingerprint. There are two main types of features, namely, the local features and the global features. The below figure 1.2(a) shows the local features and the below Figure 1.2 (b) shows the global features.

### 1.3.1 Global Features

The fingerprint global features are identified by means of the local orientation of the fingerprint ridges, that is, the Orientation Field Curves (OFCs).These features occur in the form of a Core and/or a Delta, and they are normally located in the central region of the fingerprint. These features are referred to as the singular points of a fingerprint, or simply as the singularities. A Core is the area around the center of the fingerprint loop and a Delta is the area where the fingerprint ridges tend to triangulate. Due to their vague nature, it can be concluded that both the Core and the Delta have a useful purpose in a fingerprint classification problem [4].



(a) (b)

Figure1.2: (a) Local Features: Minutiae (b) Global Features: Core and Delta [3]

### 1.3.2 Local Features

The fingerprint local features are those attributes that give the minute details about the fingerprint pattern. Minutia refers to various ways that the ridges can be discontinuous. For example, a ridge can suddenly end (termination), or can divide into two ridges (bifurcation). These minutiae are what constitute the uniqueness of every human fingerprint pattern. Due to their detailed nature, it is apparent that the local features have an important role to play in a fingerprint-matching problem [4].

## 1.4 Fingerprint Classification

An automatic recognition of people based on fingerprints requires that the input fingerprint be matched with a large number of fingerprints in a database. To reduce the search time and computational complexity, it is necessary to classify these fingerprints in a precise and consistent manner so that there is only need a subset of fingerprint in the database to be matched input fingerprint image.

According to Galton-Henry classification (Galton, 1892 and Henry, 1900) classification, we classifyfingerprint images into 5 major classes: plain arch, tented arch, left-loop, right-loop and whorl (a plain and twin loop, respectively).

- **Arch:** Arches are encountered in only 5 % of the patterns received. Arch patterns consist of ridges that run from one side of the pattern to the other. There are two types of arches – Plain arches and Tented arches. Generally, plain arch has no singular points. While tented arch have one core and one delta.

- **Loop:** Loops occur in about 60-70 % of fingerprint patterns encountered. One or more of the ridges enters on either side of the impression, re-curves, touches or crosses the line running from the delta to the core and terminates on or in the direction of the side where the ridge or ridges entered. Each loop pattern has is one delta and one core. There are two types of loops - left loop and right loop.
- **Whorl:** Whorls are seen in about 25-35 % of fingerprint patterns encountered. In a whorl, some of the ridges make a turn through at least one circuit. Any fingerprint pattern which contains 2 or more deltas will be a whorl pattern. There are basically two types of whorl plain whorl and double whorl. Plain whorls consist of one or more ridges which make or tend to make a complete circuit with two deltas, between which an imaginary line is drawn and at least one re-curving ridge within the inner pattern area is cut or touched. Double loop whorls consist of two separate and distinct loop formations with two separate and distinct shoulders for each core, two deltas and one or more ridges which make, a complete circuit. Between the two at least one re-curving ridge within the inner pattern area is cut or touched when an imaginary line is drawn.

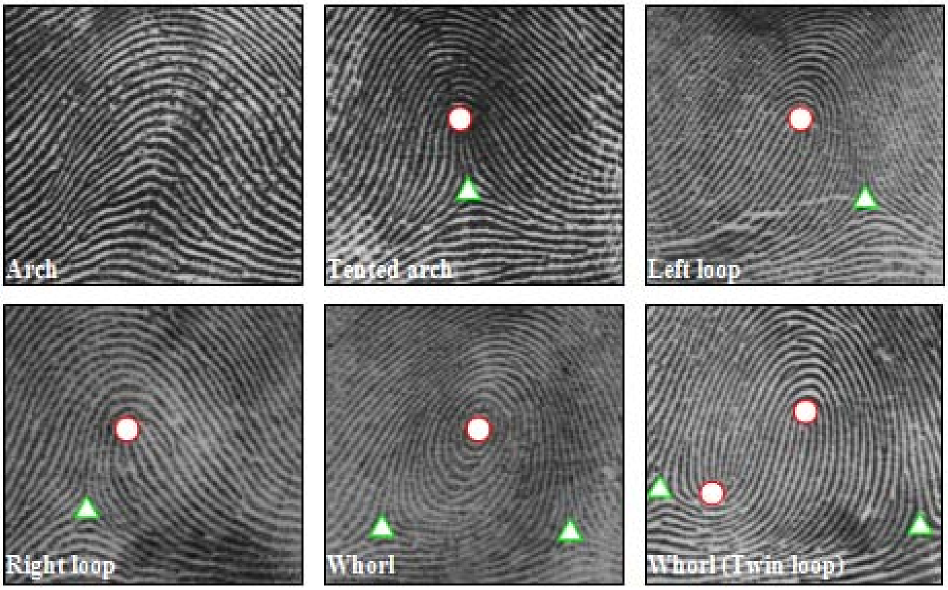



Figure1.3: Fingerprint Classes [5]

## 1.5 Fingerprint Identification

Fingerprint identification refers to specifying one's identity based on his fingerprints. The fingerprints are captured without any information about the identity of the person. It is then matched across a database containing numerous fingerprints. The identity is only retrieved when a match is found with one existing in the database. So, this is a case of one-to-n matching where one capture is compared to several others. This is widely used by law enforcement. By using this technology, law enforcement officials can identify criminals, verify identification, and preserve public safety.

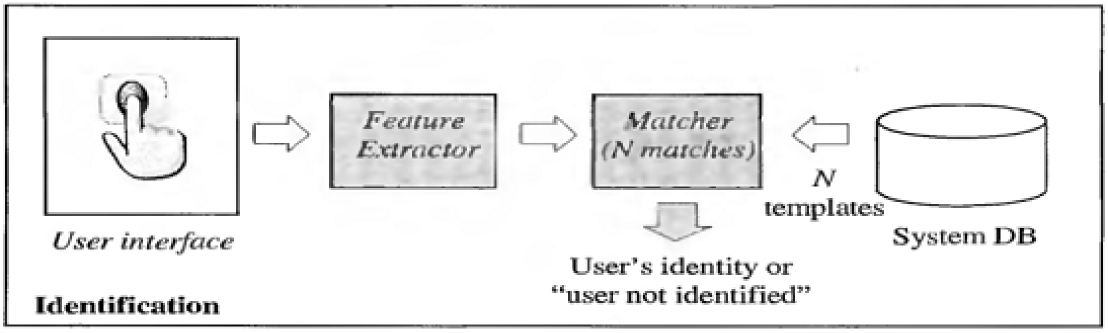



Figure 1.4: Fingerprint Identification system [1]

## 1.6 Fingerprint Matching Techniques

The large number of approaches to fingerprint matching can be coarsely classified into three families [1].

- **Correlation-based matching:** Two fingerprint images are superimposed and the correlation between corresponding pixels is computed for different alignments (e.g. Various displacements and rotations).
- **Pattern-based (or image-based) matching:** Pattern based algorithms compare the basic fingerprint patterns (arch, whorl, and loop) between a previously stored template and a candidate fingerprint. This requires that the images be aligned in the same orientation. To do this, the algorithm finds a central point in the fingerprint image and centers on that. In a pattern-based algorithm, the template contains the type, size, and orientation of patterns within the aligned fingerprint image. The candidate fingerprint image is graphically compared with the template to determine the degree to which they match.

- **Minutiae-based matching:** This is the most popular and widely used technique, being the basis of the fingerprint comparison made by fingerprint examiners. Minutiae are extracted from the two fingerprints and stored as sets of points in the two-dimensional plane. Minutiae-based matching essentially consists of finding the alignment between the template and the input minutiae sets that result in the maximum number of minutiae pairings.

  In this thesis we have implemented a minutiae based matching technique. This approach has been intensively studied, also is the backbone of the current available fingerprint identification products.

In this chapter, we discuss a brief introduction of neural network and image processing. Traditionally, the term neural network had been used to refer to a network or circuit of biological neurons. Neural networks are inspired by our brains. The modern usage of the term often refers to artificial neural networks, which are composed of artificial neurons or nodes. Thus, the term has two distinct usages:

## 2.1 Biological Neural Network

In general, a biological neural network is composed of a group or groups of chemically connected or functionally associated neurons. The human brain has about $10^{11}$ neurons and $10^{14}$ synapses. A neuron consists of a soma (cell body), axons (sends signals), and dendrites (receives signals). A synapse connects an axon to a dendrite. Given a signal, a synapse might increase (excite) or decrease (inhibit) electrical potential. A neuron fires when its electrical potential reaches a threshold (If the signals are strong enough). Thus, noise and other unwanted signals are removed. The most interesting thing to note here is that the synaptic connections and the threshold amount of signals needed to activate the neuron changes every time. This helps the neuron to learn. It learns when to fire signal by changing its synaptic connections and the threshold [30].

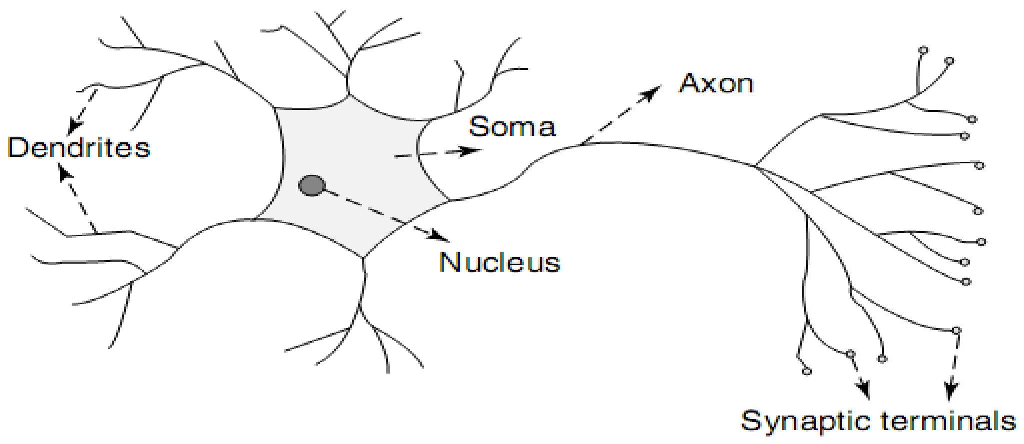



Figure 2.1: Biological Neuron [30]

## 2.2 Artificial Neural Network

In order to find relationship between the input and output data driven from accelerated experimentations, a powerful method than traditional modeling is necessary. ANN is an especially efficient algorithm to approximate any function with finite number of discontinuities by learning the relationships between input and output vectors (Hagan, M.T., H.B. Demuth, 1996; Biglin, M., 2004). These algorithms can learn from the experiments, and also are fault tolerant in the sense that they are able to handle noisy and incomplete data.

The ANNs are able to deal with non-linear problems, and once trained can perform prediction and generalization at high speed (Sozen, A., E. Arcakilioglu, 2004). They have been used to solve complex problems that are difficult for conventional approaches, such as control, optimization, pattern recognition, classification, properties and desired that the difference between the predicted and observed (actual) outputs be as small as possible (Richon, D., S. Laugier, 2003). Artificial neural networks are biological inspirations based on various characteristics of the brain functionality. They are composed of many simple elements called neurons as shown in Figure 2.2, that are interconnected by links that act like axons and determine an empirical relationship between the inputs and outputs of a given system. Where the inputs are independent variables and the outputs are dependent.

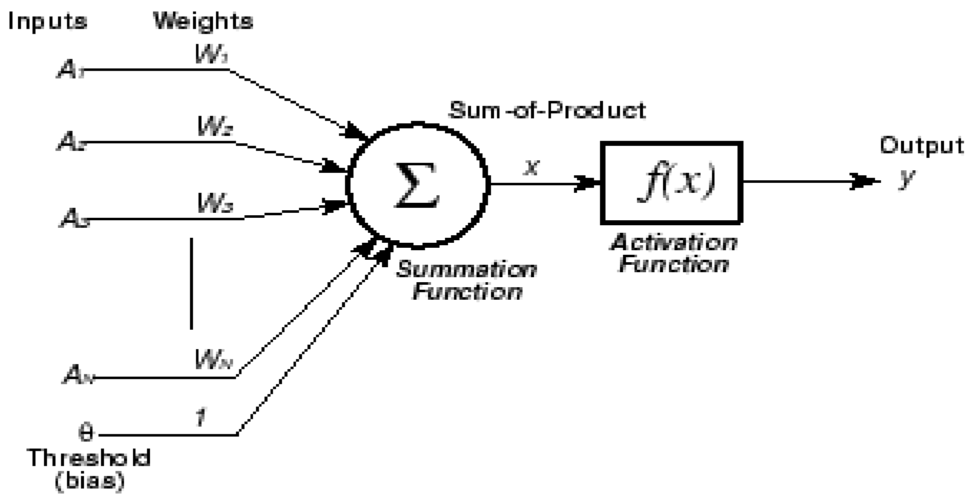



Figure 2.2: Neuron Model of an Artificial Neural Network [7]

A typical interconnected neural network that arranged in multiple layers has an input layer, an output layer, and hidden layer(s), that each one of them plays different roles.

In a network, each connecting line has an associated weight. Artificial neural networks are trained by adjusting these input weights (connection weights), so that the calculated outputs approximate the desired [6].

## 2.3 Basic Building Block of Artificial Neural Networks

The basic building blocks of the artificial neural network are [8]:

- Network architecture.
- Setting the weights.
- Activation function.

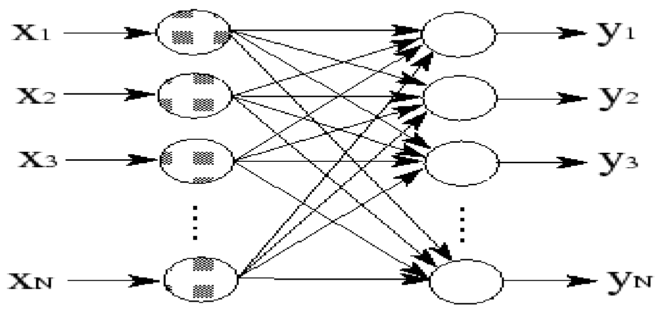
### 2.3.1 Network Architecture

There are several types of architecture of ANN. However, the two most widely used ANN are discussed below:

**1) Feed-forward networks**

Feed-forward ANNs allow signals to travel one-way only, from input to output. There is no feedback (loops) i.e. the output of any layer does not affect that same layer. Feed-forward ANNs tend to be straightforward networks that associate inputs with outputs. They are extensively used in pattern recognition. There are two types of Feed-forward neural networks, namely Single-layer and Multi-layer Feed-forward neural network [8].

- **Single-layer Feed-forward Network:** This type of network comprise of two layers, namely the input layer and output layer. The input layer neurons receive the input signals and the output layer neurons receive the output signals. The synaptic links carrying the weights connect every input neuron to the output neuron but not vice-versa (See Figure 2.3) [8].




Input layer Output layer

Figure 2.3: Single-layer Feed-forward Network [7]

- **Multi-layer Feed-forward Network:** To achieve higher level of computation capabilities, a more complex structure of neural network is required. Figure 2.4 shows the Multi-layer neural network, which distinguishes itself from the Single-layer neural network by having one or more hidden layers. In Multi-layer structure, the input nodes pass the information to the units in the first hidden layer, and then the outputs from the first hidden layer are passed to the next layer and so on. The designer of an artificial neural network should consider how many hidden layers are required, depending on complexity in desired computation [8].

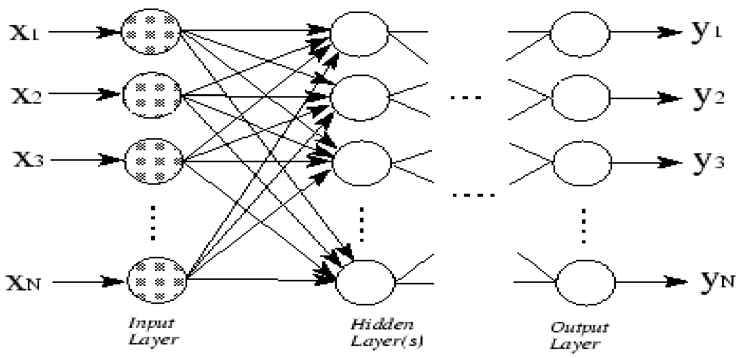



Figure 2.4: Multi-layer Feed-forward Network [7]

**2) Feedback or Recurrent network**

Feedback networks can have signals travelling in both directions by introducing loops in the network. Feedback networks are very powerful and can get extremely complicated. Feedback networks are dynamic; their 'state' is changing continuously until they reach an equilibrium point. They remain at the equilibrium point until the input changes and a new equilibrium needs to be found [8].

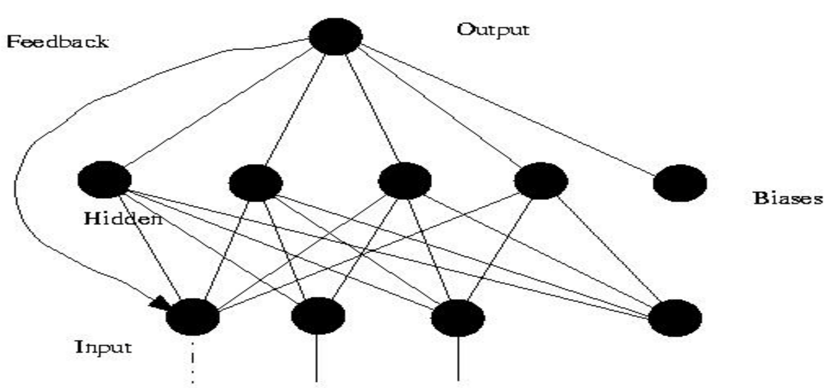



Figure 2.5: Feedback network [7]

### 2.3.2 Setting the weights

The method of setting the values for the weights enables the process of learning or training. The process of modifying the in the connections between network layers with the expected output is called training a network. The internal process that takes place when a network trained is called learning. Generally, there are two types of learning as follows [8]:

1) **Supervised learning**

In supervised learning, both the input and the actual response and the desired response are available and are used to formulate a cost (error) measure. If the actual response differs from the target response, the Neural Network generates an error signal, which is then used to calculate the adjustment that should be made to the network's weights so that actual output matches the target.

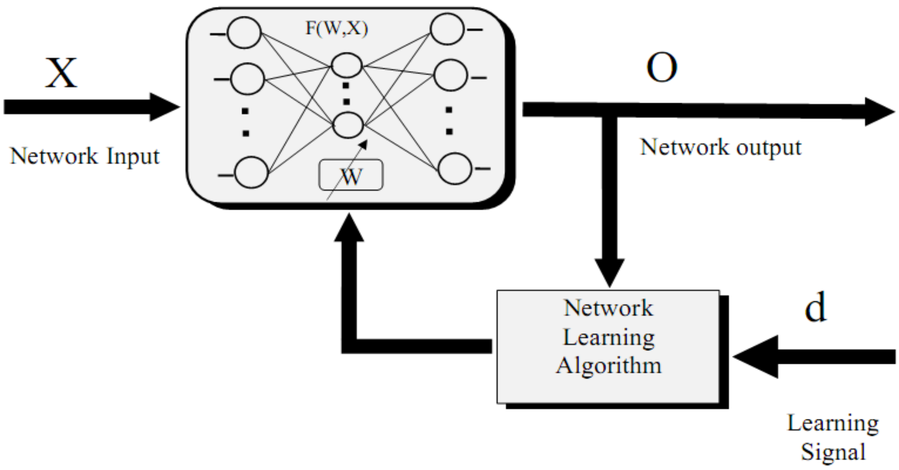



Figure 2.6: Supervised Learning Rule [10]

**2) Unsupervised learning**

A training scheme in which the network is given only input data, network finds out about some of the properties of the data set, learns to reflect these properties in its output. E.g., the network learns some compressed representation of the data. This type of learning presents a biologically more likely model of learning. What exactly these properties are, that the network can learn to recognize, depends on the particular network model and learning method. This type of learning sometimes referred to as self-organizing learning.

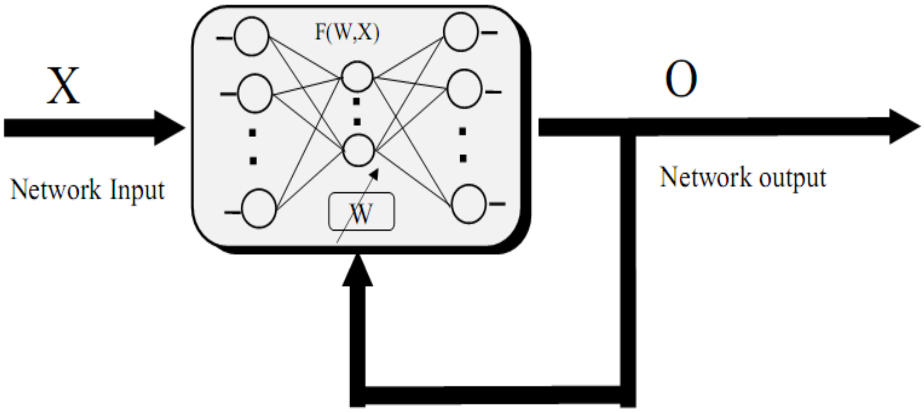



Figure 2.7: Unsupervised Learning Rule [10]

### 2.3.3 Transfer Function

The behavior of an ANN (Artificial Neural Network) depends on both the weights and the input-output function (transfer function) that is specified for the units. This function typically falls into one of three categories [8]**:**

- **Linear (or ramp):** The output activity is proportional to the total weighted output (see Figure 2.8).

$$f(x) = x\ ;\ for\ all\ x$$

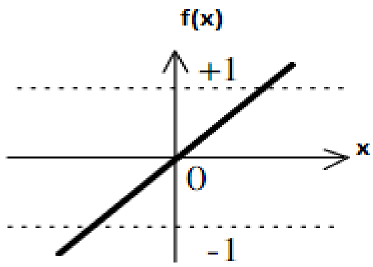



Figure 2.8: Linear activation function

- **Threshold:** The output is set at one of two levels, depending on whether the total input is greater than or less than some threshold (say$\theta$) value (See Figure 2.9).

$$f(x) = \begin{cases} 1 & x \geq \theta \\ -1 & x < \theta \end{cases}$$

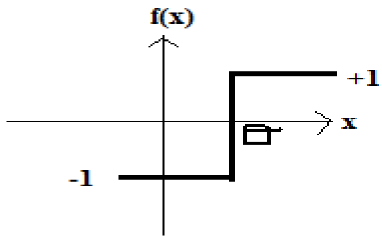



Figure 2.9: Threshold activation function

- **Sigmoid:** the output varies continuously but not linearly as the input changes. Sigmoid units bear a greater resemblance to real neurons than do linear or threshold units, but all three must be considered rough approximations (See Figure 2.10).

$$f(x) = \frac{1}{1+e^{-x}} \text{ ; for all x.}$$

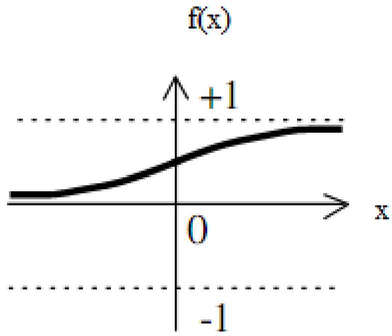



Figure 2.10: Sigmoid activation function

## 2.4 Learning Algorithms

The most common learning algorithms are [9]:

- **Hebbian learning:** Hebbian learning occurs when a connection weight on an input path to a processing element is incremented if both the input is high (large) and the desired output is high. This is analogous to the biological process in which a neural pathway is strengthened each time it is used.
- **Delta-rule learning:** Delta-rule learning (sometimes called mean square error learning) occurs when the error signal (difference between the desired output and the actual output) is minimized using a least-squares process. Back propagation is the most common implementation of Delta-rule learning and probably is used in at least 75% of ANN applications.

- **Competitive learning:** Competitive learning occurs when the processing elements compete; only the processing element yielding the strongest response to a given input can modify itself, becoming more like the input. In all cases, the final values of the weighting functions constitute the "memory" of the ANN.

## 2.5 Structuring the Network

The number of layers and the number of processing elements per layer are important decisions. These parameters to a feed forward, back-propagation topology are also the most ethereal - they are the "art" of the network designer. There is no quantifiable, best answer to the layout of the network for any particular application. There are only general rules picked up over time and followed by most researchers and engineers applying this architecture to their problems [11].

- **Number of hidden layers:** The hidden layer(s) provide the network with its ability to generalize. In theory, a neural network with one hidden layer with a sufficient number of hidden neurons is capable of approximating any continuous function. In practice neural network with one and occasionally two hidden layers are widely used and have to perform very well.
- **Number of hidden nodes:** There is no magic formula for selecting the optimum number of hidden neurons. As the complexity in the relationship between the input data and the desired output increases, the number of the processing elements in the hidden layer should also increase. However, some thumb rules are available for calculating number of hidden neurons. A rough approximation can be obtained by the geometric pyramid rule proposed by Masters (1993). For a three layer network with n input and m output neurons, the hidden layers would have *sqr (n*m)* neurons.
- **Number of output nodes:** Neural networks with multiple outputs, especially if these outputs are widely spaced, will produce inferior results as compared to network with a single output.
- **Activation function:** Activation function is mathematical formulae that determine the output of a processing node. Each unit takes its net input and applies an activation function to it. Transfer function such as sigmoid, are commonly used because they are nonlinear and continuously differentiable which are desirable for network learning.

## 2.6 What is Image processing?

Image processing is a physical process used to convert an image signal into a physical image. The image signal can be either digital or analog. The actual output itself can be an actual physical image or the characteristics of an image. The most common type of image processing is photography. In this process, an image is captured or scans using a camera to create a digital or analog image. In order to produce a physical picture, the image is processed using the appropriate technology based on the input source type. In digital photography, the image is stored as a computer file. This file is translated using photographic software to generate an actual image. The colors, shading, and nuances are all captured at the time the photograph is taken the software translates this information into an image. When creating images using analog photography, the image is burned into a film using a chemical reaction triggered by controlled exposure to light. The image is processed in a darkroom, using special chemicals to create the actual image. This process is decreasing in popularity due to the opening of digital photography, which requires less effort and special training to product images. The field of digital imaging has created a whole range of new applications and tools that were previously impossible. Face recognition software, medical image processing and remote sensing are all possible due to the development of digital image processing. Specialized computer programs are used to enhance and correct images [30] [22].

### 2.6.1 Types of Digital Image

For photographic purposes, there are two important types of digital images: color and grayscale. Color images are made up of colored pixels while grayscale images are made of pixels in different shades of gray [12].

- **Grayscale Images:** A grayscale image is made up of pixels, each of which holds a single number corresponding to the gray level of the image at a particular location. These gray levels span the full range from black to white in a series of very fine steps, normally 256 different grays. Assuming 256 gray levels, each black and white pixel can be stored in a single byte (8 bits) of memory.
- **Color Images:** A color image is made up of pixels, each of which holds three numbers corresponding to the red, green and blue levels of the image at a

particular location. Assuming 256 levels, each color pixel can be stored in three bytes (24 bits) of memory. Note that for images of the same size, a black & white version will use three times less memory than a color version.

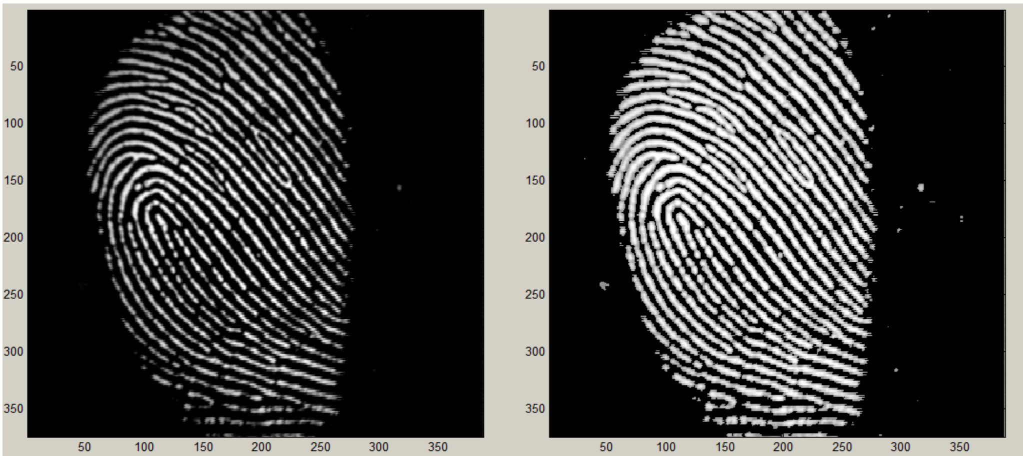
- **Binary Images:** Binary images use only a single bit to represent each pixel. Since a bit can only exist in two states- ON or OFF, every pixel in a binary image must be one of two colors, usually black or white. This inability to represent intermediate shades of gray is what limits their usefulness in dealing with photographic images.

### 2.6.2 Basic operation of Image processing

- **Image Enhancement**

  Image enhancement is the improvement of digital image quality (wanted e.g. for visual inspection or for machine analysis), without knowledge about the source of degradation. If the source of degradation is known, one calls the process image restoration. Hence, by the term image enhancement we mean improvement of the appearance of an image by increasing dominance of some feature, or by decreasing ambiguity between different regions of the image. The enhancement techniques can be dividing into three categories [22]:

  - Contrast intensification
  - Noise cleaning or smoothing
  - Edge sharpening or crispening




Original Image Enhancement by Histogram Equalization

Figure 2.11: Example of Image Enhancement [19]

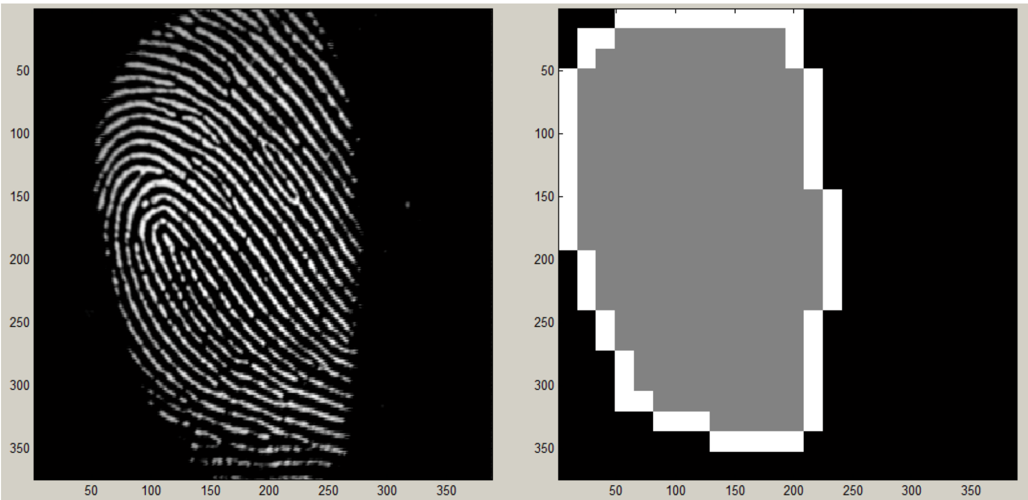
- **Image Segmentation**

Segmentation refers to the process of partitioning a digital image into meaningful segments (sets of pixel) or regions. This meaningful region may be a complete object or may be a part of it. Segmentation is an important part of practically any automated image recognition system, because it is at this moment that one extracts the interesting objects, for further processing such as description or recognition. Segmentation of an image is in practice the classification of each image pixel to one of the image parts. More precisely, image segmentation is the process of assigning a label to every pixel in an image such that pixels with the same label share certain visual characteristics. Each of the pixels in a region is similar with respect to some characteristic or computed property, such as color, intensity, or texture. Adjacent regions are significantly different with respect to the same characteristic.

In mathematical sense the segmentation of the image *I*, which is a set of pixels, is the partition of *I* into *n* disjoint sets *R1*,*R2*, . . . , *Rn*, called segments or regions such that their union of all regions equals *I*.

$$I = R_1 \cup R_2 \cup \ldots \cup R_n$$

There two types of segmentation approach Pixel-based, Region-based. But in this thesis we focus only on Region-based segmentation [22].




Original Image Region of Interest (After Segmentation)

Figure 2.12: Example of Image Segmentation [19]

- **Image Compression**

  Image compression is minimizing the size in bytes of a graphics file without degrading the quality of the image to an unacceptable level. The reduction in file size allows more images to be stored in a given amount of disk or memory space. It also reduces the time required for images to be sending over the Internet or downloaded from Web pages [22].

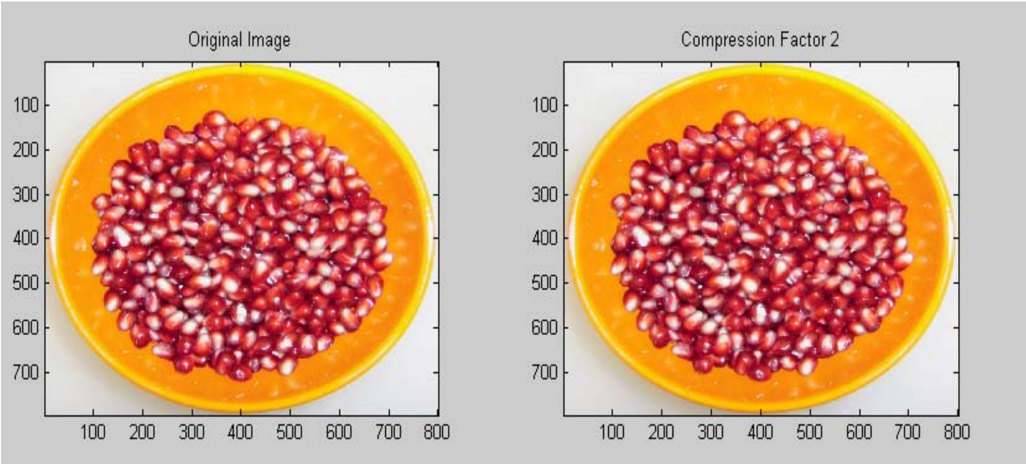



Figure 2.13: Example of image compression [29]

- **Image Restoration**

  Image restoration removes or minimizes some known degradations in an image. It can be seen as a special kind of image enhancement. The most common degradations have their origin in imperfections of the sensors, or in transmission [29].

Original image Crack removed & restored image

Figure 2.14: Example of Image Restoration [14]

### 2.6.3 Feature Extraction

To generate the description of the scene or for image understanding, we usually need to recognize each of these regions. One of the prerequisites of identification and recognition is feature extraction. By the term feature extraction, we mean determining various attributes as well as properties associated with a region or object [22].

## 2.7 Survey on fingerprint identification system

In this section, detail of work has been done on fingerprint identification system is given:

In 2000 Xiping Luo, Jie Tian, proposed a rule-based method to do fingerprint enhancement. They introduced human knowledge about fingerprints into the enhancement process in the form of rules and simulate what an expert will do to enhance a fingerprint image Fingerprint image is a special kind of image and has its own characteristics. Most of the known techniques for fingerprint image enhancement are direction-oriented and rely heavily on the correct estimate of local ridge direction. But the estimate of local ridge directions is unreliable in the areas corrupted by noise where enhancement is most needed. Rule-based method to do fingerprint enhancement tried to simulate what an expert will do to enhance a fingerprint image. The skeleton image is used to give information of the connect relation of ridges for the enhancement of the binary image [15].

In 2003 Wang Sen, Zhang Weiwei, Wang Yangsheng, present useful and effective fingerprint image segmentation. They extract two new features, with which their algorithm can distinguish noisy area from the foreground, and, therefore, can reduce the number of false minutiae. They use supervised RBF neural network to classify patterns and select typical patterns to train the classifier. Firstly, they divided the input fingerprint image into w×w blocks and computed the four features in each block. After that, we put these features into RBF network and classified each block in the input fingerprint image into foreground and background. If a block is foreground then B(i,j) =1, else B(i,j) = 0. After image B is obtained, the classification is completed. They also compute the percentage of foreground. If the percentage of foreground regions is smaller than a threshold, then the input fingerprint image is rejected [16].

In 2004 Madasu Hanmandlu, Vamsi Krishna Madasu, Shantaram Vasikarla, developed a texture segmentation technique for the segmentation of texture image. For this purpose, a membership function is constructed to represent the effect of the neighboring pixels on the current pixel in a window. Using these membership function values, find a feature by weighted average method for the current pixel. This is repeated for all pixels in the window treating each time one pixel as the current pixel. Using these fuzzy based features, they derive three descriptors such as maximum, entropy, and energy for each window. To segment the texture image, the modified mountain clustering that is unsupervised and Fuzzy C-means clustering have been used. This approach is spurred by the fact that a Texel has an ambiguity in the spatial arrangement of gray levels of pixels. Moreover, all the pixels do not have the same property and there is uncertainty in the property as well. Here, they explore the possibility of using interaction type model to evolve texture features. Next, these features are utilized to cluster the image. For extracting features, the given image is partitioned into sub-images of size w×w and is replaced with texture features [17].

In 2005 Tsai-YangJea,Venu Govindaraju, proposed a minutia based partial fingerprint recognition system. They present an approach that uses two major types of features that are used in fingerprint matching: local and global features. Local features, such as the minutiae information and our secondary features, contain the information that is in a local area only and invariant with respect to global transformation. On the other hand, global features, such as number, type, and position of singularities, spatial relationship and geometrical attributes of ridge lines, size and shape of thefingerings, are characterized by the attributes that capture the global spatial relationships of a fingerprint. Because of the nature of partial fingerprints, partial fingerprint matching requires a set of local features that do not depend on global singular structures. Moreover, localized features have the ability to tolerate more distortions localized secondary features derived from relative minutiae information. Aflow network -based matching technique is introduced to obtain one-to-one correspondence of secondary features. This method balances the tradeoffs between maximizing the number of matches and minimizing total feature distance between query and reference fingerprints. A two-hidden-layer fully connected neural network is trained to generate the final similarity score based on minutiae matched in the overlapping areas. Since the minutia-based fingerprint representation is an ANSI -NIST standard [American

National standards Institute, New York, 1993], this approach has the advantage of being directly applicable to existing databases [18].

In 2006 Hamed Vahdat Nejad, Hameed Reza Pourreza, and Hasan Ebrahimi, developed a new fuzzy filter is presented for noise reduction of images corrupted with additive noise. The filter consists of two stages. In the first stage, all the pixels of image are processed for determining noisy pixels. For this, a fuzzy rule based system associates a degree to each pixel. The degree of a pixel is a real number in the range [0, 1], which denotes a probability that the pixel is not considered as a noisy pixel. In the second stage, another fuzzy rule based system is employed. It uses the output of the previous fuzzy system to perform fuzzy smoothing by weighting the contributions of neighboring pixel values. They proposed a new fuzzy filter for additive Gaussian noise reduction. The main feature of the proposed filter is that it tries to determine corrupted pixels to reduce their contribution in smoothing process. Hence, it performs fuzzy smoothing by the previous knowledge of the pixels [19].

In 2007 Guo Lei,Chen Da-hai, Li Hai, Chang Jiang, proposed a new method for binarizing fingerprint image while preserving image characteristics. Firstly, a band-stop filter method is adopted to remove the noises in high frequency. At the same time, a simply threshold original image is passed to Gaussian low pass filter, and outlines of fingerprint region are get by applying region growing method to the filtered image. Then, background in band-pass image is filled with outlines of the fingerprint region. After these procedures, background intensity is close to zero, thus fixed threshold one is used for binarization. Experimental results demonstrate that band stop method can eliminate noises effectively in the fingerprint images. Moreover, binarization in outline of fingerprint can solve the problem of disturbance of the background. Segmenting foreground from background helps select threshold and keep the characteristics of fingerprint.

In 2007 Jianping Yin, En Zhu, Xuejun Yang, Guomin Zhang, Chunfeng Hu, proposed a new algorithm for fingerprint segmentation. They say a fingerprint image usually consists of different regions: non-ridge regions, high quality ridge regions, and low quality ridge regions. Fingerprint segmentation is usually to exclude non-ridge regions and unrecoverable low quality ridge regions as background to avoid detecting false features. In ridge regions, including high quality and low quality, there are often some remaining ridges which are the afterimage of the previously scanned finger and are expected to be excluded as background. However, existing segmentation methods do not take this case into consideration, and often, the remaining ridge regions are falsely taken as foreground. They proposed two steps for fingerprint segmentation to exclude the remaining ridge region from the foreground. The non-ridge regions and unrecoverable low quality ridge regions are removed as background in the first step, and then the foreground produced by the first step is further analyzed to remove the remaining ridge region [20].

In 2008 NeetaNain , Deepak BM,Dinesh Kumar, Manisha Baswal and BijuGautham proposed a new minutiae-based approach to matchfingerprint images using similar structures. Distortion poses serious threats through altered geometry, increases false minutiae, and hence makes it very difficult to find a perfect match. This Algorithm divides fingerprint images into two concentric circular regions–inner and outer–based on the degree of distortion. The algorithm assigns weight ages for a minutiae–pair match based on the region in which the pair exists [31].

In 2009 Hoi Le, T. Duy Bui present a specific contribution by introducing a new robust indexing scheme that is able not only to fasten the fingerprint recognition process but also improve the accuracy of online fingerprint identification the system with a fast and distortion tolerance hashing scheme. Fingerprint features stored in database can be considered as a very large set. Moreover, the distribution of fingerprint feature points is not uniform and unknown. One of the traditional techniques to reduce the searching time in a database is hashing. However, if the input set is not uniform, this traditional hashing becomes inefficient because it results in a hash table of which the size of hash entry is unlikely uniform; therefore the average expected searching time is not much lower than the case where linear search strategy is used. Therefore, in order to improve the search speed, they proposed an indexing scheme such that for the large and not uniformly distributed input sets of feature points, it can produce uniformly distributed hash values for searching process. To match this requirement, there is one technique called random extractor, which is widely used in cryptography, and other purposes related to randomness generation. A random extractor is a function that takes an input a high entropy and not uniform source and generates an output shorter and uniformly distributed. A randomized hash function is a construction of random extractors. Following is the description of scheme, which builds such randomized hash function to perform indexing on fingerprints [21].

# Chapter-3: Problem Statement and Methodology

## 3.1 Problem Statement

Fingerprint identification is the technology that distinguishes between a user from others using the unique information infingerprint. Fingerprint identification is one of the oldest known biometric techniques, but still is widely used because of its simplicity and good levels of accuracy. Fingerprint identification system consists of three main processes, namely, acquisition, feature extraction, and matching.

Firstly, the system obtains the digitalized fingerprint images using a sensor. Fingerprint images are rarely of perfect quality. These images may be degraded and corrupted, with elements of noise due to many factors including variations in skin and impression conditions such as scars, humidity, oily, dirt, and non-uniform contact with the fingerprint capture device. This degradation can result in a significant number of spurious minutiae being created and genuine minutiae being ignored. A critical step in studying the statistics of fingerprint minutiae is to reliably extract minutiae from fingerprint images. Thus, it is necessary to employ image enhancement techniques prior to minutiae extraction to obtain a more reliable estimate of minutiae locations.

Accurate segmentation of fingerprint images influences directly the performance of minutiae extraction. If more background areas are included in the segmented fingerprint of interest, more features that are false are introduced; if some parts of the foreground are excluded, useful feature points may be missed. Thus after the enhancement of the image accurate segmentation is the second problem.

So, In order to achieve high-accuracy minutiae with varied quality fingerprint images, segmentation algorithm needs to separate foreground from noisy background without excluding any ridge-valley regions and not include meaningless background. Image enhancement algorithm needs to keep the original ridge flow pattern without altering the singularity, join broken ridges, clean artifacts between pseudo-parallel ridges, and not introduce false information.

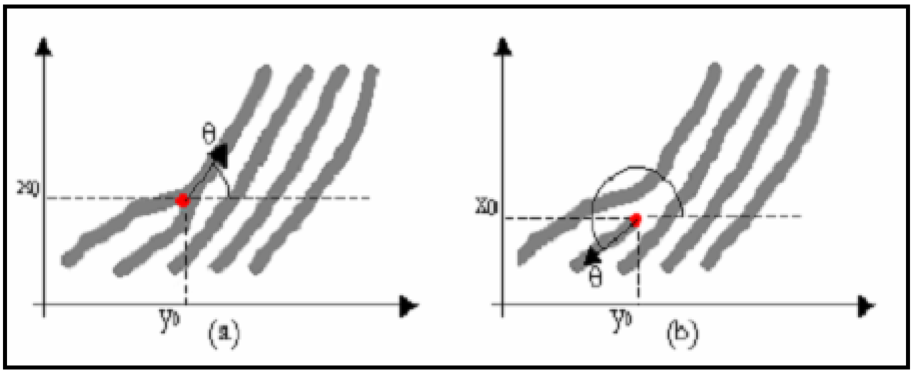
Finally minutiae detection algorithm needs to locate efficiently and accurately the minutiae points. However, it is not an easy task to extract accurate minutiae point from fingerprint. In addition, there may be some false minutiae point that makes the problem more difficult. So, finding false minutiae and their removal is the third problem.

## 3.2 Methodology

The following modules constitute the major methodologies for fingerprint identification system.

- Image Acquisition
- Image Enhancement
- Image Binarization
- Image Segmentation
- Thinning
- Minutiae Marking
- Remove False Minutiae
- Minutiae Matching

We have concentrated our implementation on Minutiae based method. In particular, we are interested only in two of the most important minutia features i.e. Ridge Ending and Ridge bifurcation as shown in Figure 4.1.




(a) (b)

Figure 4.1: (a) Ridge Bifurcation, (b) Ridge Ending [3]

The outline of our approach can be broadly classified into three stages – Acquiring device, Minutiae Extraction and Minutiae matching.

The system takes one fingerprints as input to be match this fingerprint with another fingerprints, which already exist in database and gives a percentage score of the extent of match between the two. Based on the score and threshold match value it can distinguish whether the two fingerprints match or not.

## 4.1 Design Description

The above system is further classified into various modules and sub-modules as given in Figure 4.2. Minutia extraction includes Preprocessing (Image enhancement, Image binarization and Image Segmentation), Minutiae extraction (Ridge Thinning and Minutiae Marking) and Post processing (Remove false minutiae) while Minutiae matching include Minutiae Alignment and Match processes.

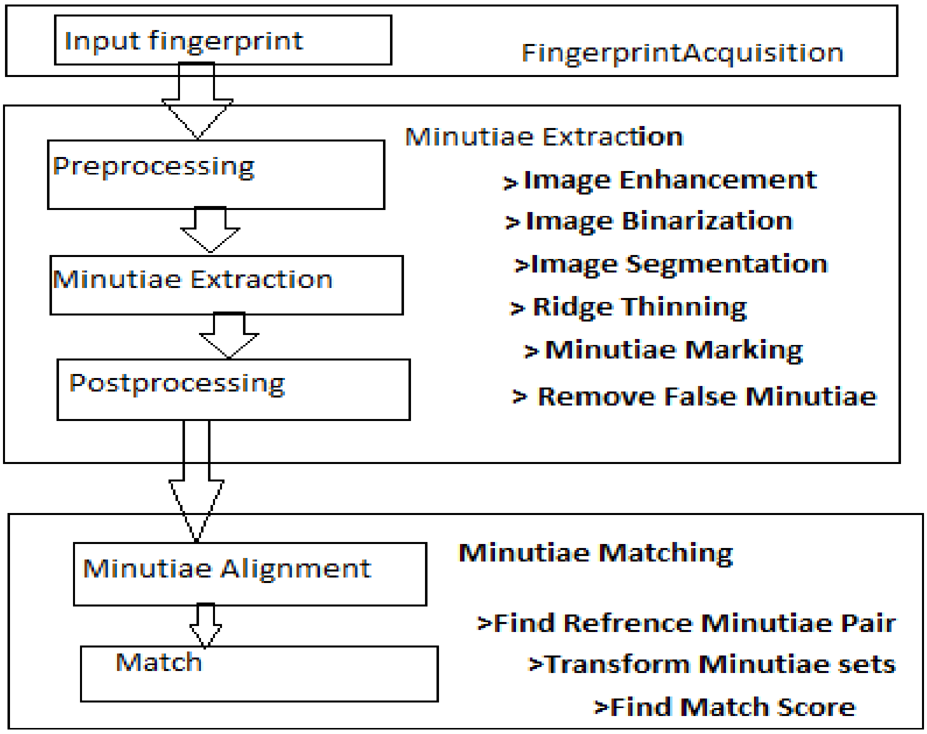



Figure 4.2: Detailed Design Descriptions

Under Preprocessing step, Image Enhancement using Histogram Equalization increases the quality of the input image, Image Binarization converts the grey scale image to a binary image and Image Segmentation is performed which extracts a Region of Interest using Ridge Flow Estimation and MATLAB's morphological functions.

Thereafter the minutia points are extracted in the Minutiae Extraction and Post processing steps by Ridge Thinning, Minutia Marking and Removal of False Minutiae processes.

Using the above Minutia Extraction process, we get the Minutiae sets for the two fingerprints to be match. Minutiae Matching process iteratively chooses any two minutiae as a reference minutia pair and then matches their associated ridges first. If the ridges match well, two fingerprint images are aligned and matching is conducted for all remaining minutia to generate a Match Score.

## 4.2 Fingerprint Acquisition

For fingerprint acquisition, optical or semi-conduct sensors are widely used. They have high efficiency and acceptable accuracy except for some cases that the user's finger is too dirty or dry. The input fingerprints are taken from the database provided by FVC2004 (Fingerprint Verification Competition 2004). So no acquisition stage has been implemented.

## 4.3 Fingerprint Image Enhancement

The first step in the minutiae extraction stage is Fingerprint Image enhancement. This is mainly done to improve the image quality and to make it clearer for further operations. Often fingerprint images from various sources lack sufficient contrast and clarity. Hence, image enhancement is necessary and a major challenge in all fingerprint techniques to improve the accuracy of matching. It increases the contrast between ridges and furrows and connects the some of the false broken points of ridges due to insufficient amount of ink or poor quality of sensor input. In our thesis, we have implemented two techniques: Histogram Equalization and Image Binarization.

### 4.3.1 Histogram Equalization

Histogram equalization is a technique of improving the global contrast of an image by adjusting the intensity distribution on a histogram. This allows areas of lower local contrast to gain a higher contrast without affecting the global contrast. Histogram equalization accomplishes this by effectively spreading out the most frequent intensity values. The original histogram of a fingerprint image has the bimodal type, the histogram after the histogram equalization occupies all the range from 0 to 255 and the visualization effect is enhanced (See Figure 4.3).


Original image Enhance image

Figure 4.3: Enhance image

### 4.3.2 Fingerprint Image Binarization

Image Binarization is a process, which transforms the 8-bit Gray image to a 1-bit image with 0-value for ridges and 1-value for furrows. After the operation, ridges in the fingerprint are highlight with black color while furrows are white.

A locally adaptive binarization method is performed to binarize the fingerprint image. In this method image is divide into blocks of 16 x 16 pixels. A pixel value is then set to 1 if its value is larger than the mean intensity value of the current block to which the pixel belongs (Figure 4.4).


Enhance image Binarize image

Figure 4.4: Binarize Image

## 4.4 Fingerprint Image Segmentation

After image enhancement, the next step is fingerprint image segmentation. In general, only a Region of Interest (ROI) is useful to be recognized for each fingerprint image. The image area without effective ridges and furrows is first discarded since it only holds background information. Then the bound of the remaining effective area is sketched out since the minutiae in the bound region are confusing with those spurious minutiae that are generated when the ridges are out of the sensor.

To extract the region of interest, two steps are followed: Block direction estimation and ROI extraction by Morphological methods.

### 4.4.1 Block direction estimation

Here the fingerprint image is divided into blocks of size 16 x 16 pixels (W x W), after which the block direction of each block is calculated according to the algorithm:

i. Calculate the gradient values along *x*-direction ($g_x$) and y-direction ($g_y$) for each pixel of the block. Two Sobel filters are used to fulfill the task.

ii. For each block, use following formula to get the Least Square approximation of the block direction [3].

$$tan2\beta = \frac{2\Sigma\Sigma\left(\mathrm{g_x} * \mathrm{g_y}\right)}{\Sigma\Sigma\left(\mathrm{g_x^2} - \mathrm{g_y^2}\right)}$$

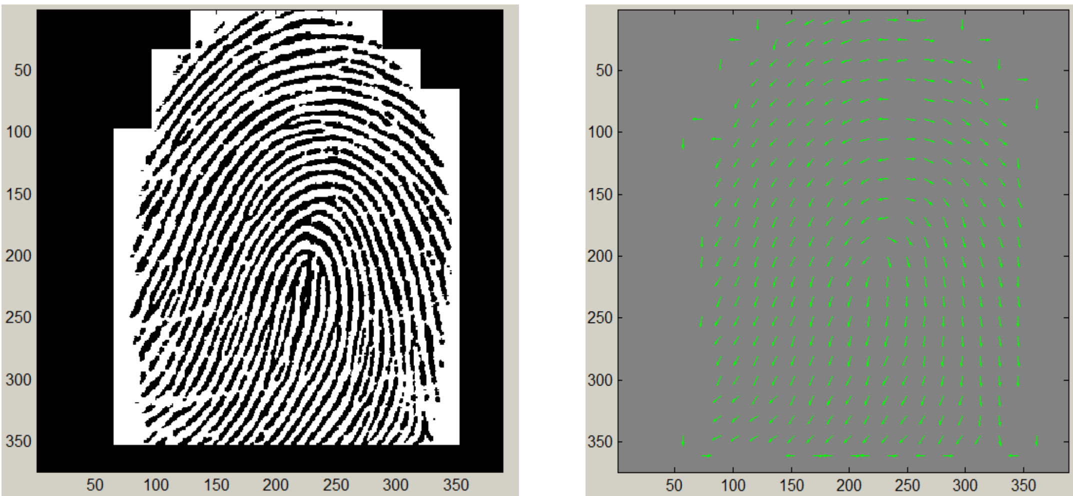
The formula is easy to understand by regarding gradient values along x-direction and y-direction as cosine value and sine value. So the tangent value of the block direction is estimated nearly the same as the way illustrated by the following formula [3].

$$tan2\theta = \frac{2sin\theta cos\theta}{cos^2\theta - sin^2\theta}$$

After finished with the estimation of each block direction, those blocks without significant information on ridges and furrows are discarded based on the following formulas:

$$E = \frac{2\Sigma\Sigma\left(g_x * g_y\right) + \Sigma\Sigma\left(\mathrm{g_x^2} - \mathrm{g_y^2}\right)}{W * W * \Sigma\Sigma\left(g_x^2 + g_y^2\right)}$$

For each block, if its certainty level E is below a threshold, then the block is regarded as a background block. The direction map is shown in Figure 4.5.     .




Binarize image

Direction map image

Figure 4.5: Direction map image

### 4.4.2 ROI Extraction by Morphological operations

ROI extraction is done using two Morphological operations called OPEN and CLOSE. The OPEN operation can expand images and remove peaks introduced by background noise. The 'CLOSE' operation can shrink images and eliminate small cavities.

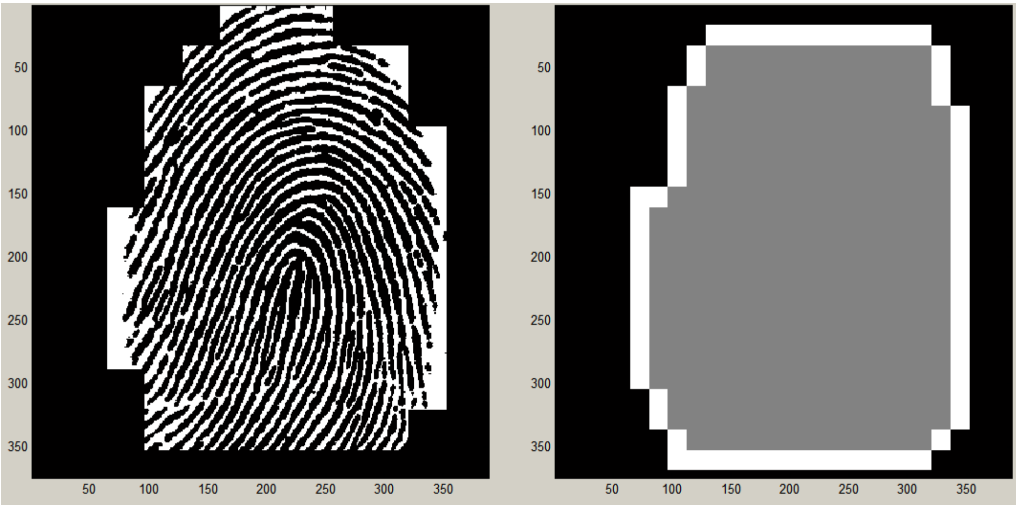



Figure 4.6: Fingerprint ROI

Figure 4.6 shows the ROI of the fingerprint, which is the bound area after subtraction of the closed area from the opened area. Then the leftmost, rightmost, uppermost and bottommost blocks out of the bound area are discarded.

## 4.5 Minutiae Extraction

Now that we have enhanced the image and segmented the required area, the job of minutiae extraction closes down to four operations: Ridge Thinning, Minutiae Marking, False Minutiae Removal and Minutiae Representation.

### 4.5.1 Ridge Thinning

In this process, we eliminate the redundant pixels of ridges until the ridges are just one pixel wide. This is done using the MATLAB's built in morphological thinning function.

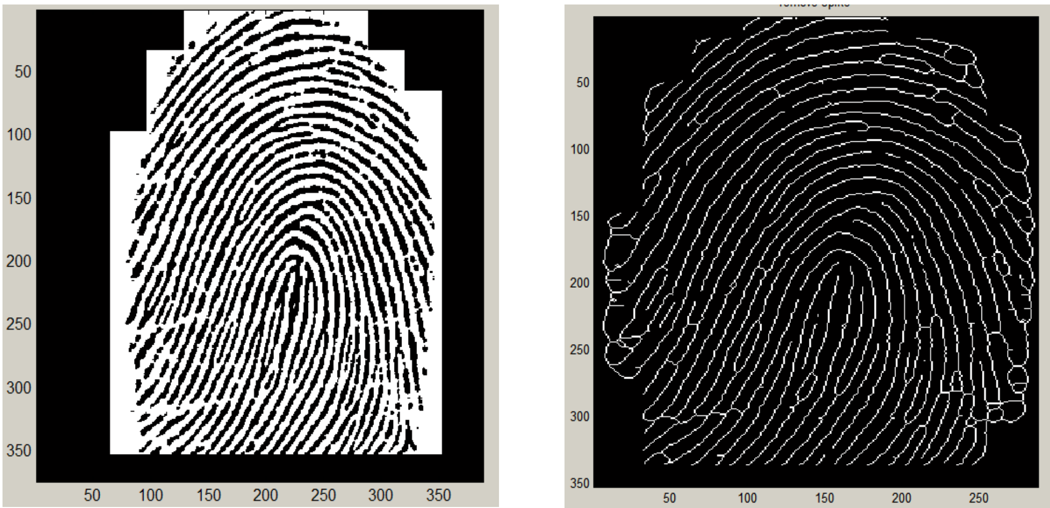
```
bwmorph(binaryImage,'thin',Inf)
```

The thinned image is then filtered,again using MATLAB's three morphological functions to remove some H breaks, isolated points and spikes (See Figure 4.7)

```
bwmorph(binaryImage,'hbreak',k)
```

```
bwmorph(binaryImage,’clean’,k)

bwmorph(binaryImage,’spur’,k)
```




Binarize image Thinning image

Figure 4.7: Thinning image

### 4.5.2 Minutiae Marking

The concept of Crossing Number (CN) is used for minutiae marking. Minutiae marking are now done using templates for each 3 x 3 pixel window as follows.

If the central pixel is 1 and has exactly 3 one-value neighbors, then the central pixel is a ridge branch (See Figure 4.8).

| | | |
|---|---|---|
| 0 | 1 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 1 |

Figure 4.8: Ridge Bifurcation

If the central pixel is 1 and has only 1 one-value neighbor, then the central pixel is a ridge ending (See Figure 4.9).

| | | |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 0 | 0 | 1 |

Figure 4.9: Ridge Termination

There is one case where a general branch may be triple counted (See Figure 4.10). Suppose both the uppermost pixel with value 1 and the rightmost pixel with value 1 have another neighbor outside the 3x3 window due to some left over spikes, so the two pixels will be marked as branches too, but actually only one branch is located in the small region.

| | | |
|---|---|---|
| 0 | 1 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 0 |

Figure 4.10: Triple counting branch

### 4.5.3 False Minutiae Removal

At this stage false ridge breaks due to insufficient amount of ink & ridge cross connections due to over inking are not totally eliminated. Also some of the earlier methods introduce some spurious minutia points in the image. So to keep the recognition system consistent these false minutiae need to be removed.

Here we first calculate the inter ridge distance D which is the average distance between two neighboring ridges. For this scan each row to calculate the inter ridge distance using the formula [23]:

$$Inter\ ridge\ distance = \frac{sum\ all\ the\ pixels\ in\ the\ row\ whose\ value\ is\ one}{row\ length}$$

Finally, an averaged value over all rows gives D. All we label all thinned ridges in the fingerprint image with a unique ID for further operation using a MATLAB morphological operation BWLABEL.

Now the following 7 types of false minutia points are removed using these steps (See Figure 4.11) [23] [27].

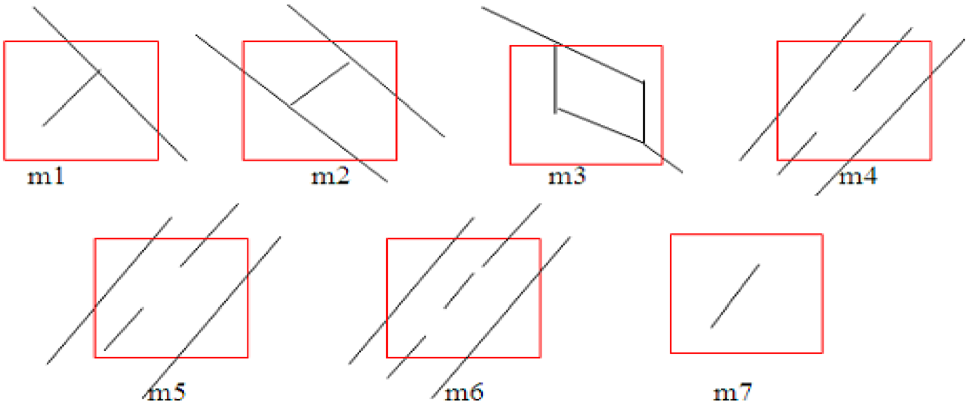



Figure 4.11: False minutia structures

- If d (bifurcation, termination) < D & the two minutia are in the same ridge then remove both of them (case m1).
- If d (bifurcation, bifurcation) < D & the two minutia are in the same ridge them remove both of them (case m2, m3).
- If d (termination, termination) ≈ D & the their directions are coincident with a small angle variation & no any other termination is located between the two terminations then remove both of them (case m4, m5, m6).
- If d (termination, termination) < D & the two minutia are in the same ridge then remove both of them (case m7).

  Where d(X, Y) is the distance between two minutia points.

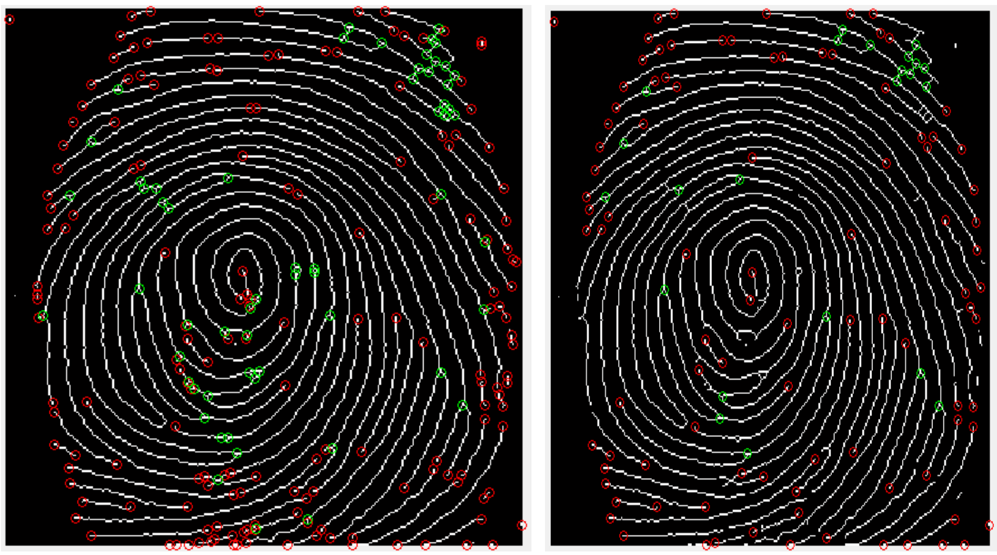


Image with Minutiae Image with Remove spurious minutiae

Figure 4.12: Remove spurious minutiae

### 4.5.4 Minutiae Representation

Finally, after extracting valid minutia points from the fingerprint they need to be stored in some form of representation common for both ridge ending and bifurcation. So each minutia is completely characterized by the following parameters

- x-coordinate,
- y-coordinate,
- orientation and
- ridge associated with it

Actually, a bifurcation can be broken down to three terminations each having their own x-y coordinates (pixel adjacent to the bifurcating pixel), orientation and an associated ridge. The orientation of each termination (tx, ty) is estimated by following method. Track a ridge segment whose starting point is the termination and length is D. Sum up all x-coordinates of points in the ridge segment. Divide above summation with D to get sx. Then get sy using the same way.

Get the direction from:

$$tan^{-1}\frac{sy-ty}{sx-tx}$$

## 4.6 Training

To train minutiae point location by neural network using back propagation algorithm. All detail about neural network has been given in chapter 2. Back propagation is most popular supervisor learning algorithm used to training in various systems [25] [26] [27] [28]. With back propagation, the input data is repeatedly presented to the neural network. With each presentation, the output of the neural network is compared to the desired output and an error is computed. This error is then fed back (back propagated) to the neural network and used to adjust the weights such that the error decreases with each iteration and the neural model gets closer and closer to producing the desired output. For input values whole image is taken, and fix the target as location points in image and then train the system. System is trained many times to give the batter performance as shown in Figure 4.13. Figure 4.14 shows that after 20 epochs, the performance is 0.0118664, which increase up to 0.00342491 after taking 40 epochs (Figure4.15). Figure 4.16 show the performance is 0.000941088 after 68 epochs.

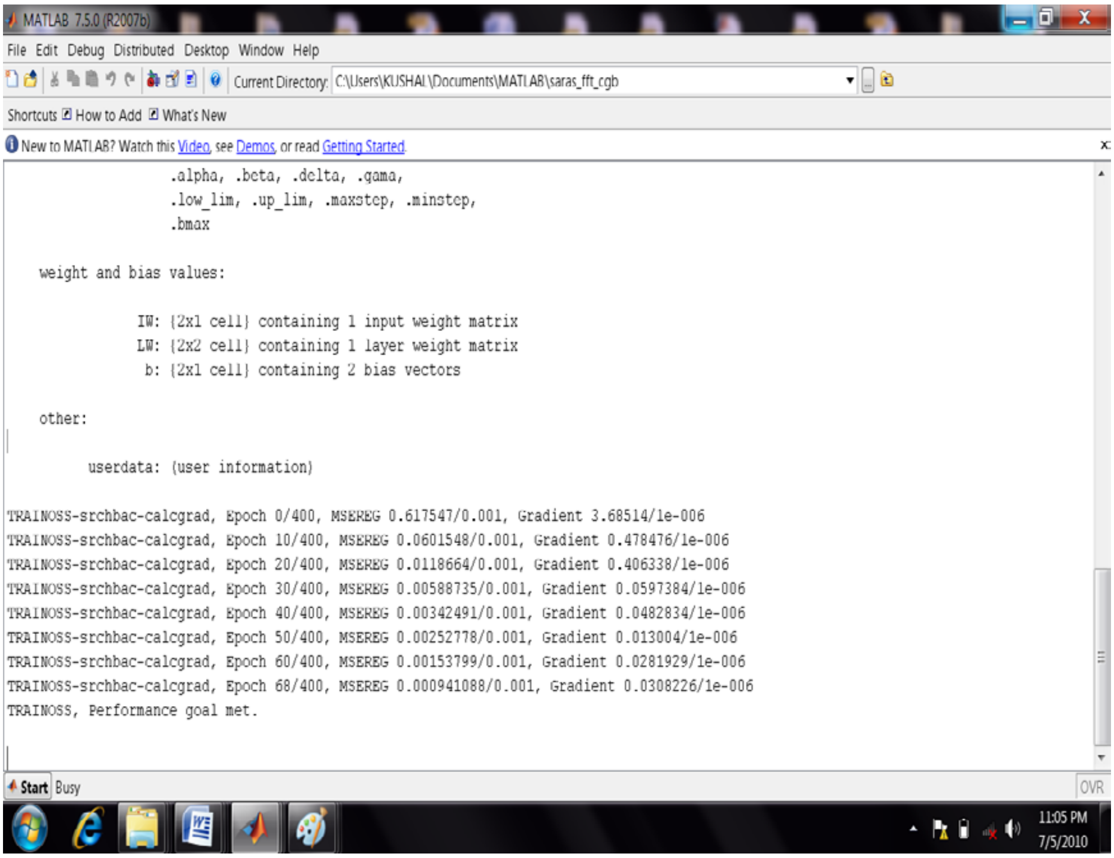



Figure 4.13: Training by neural network

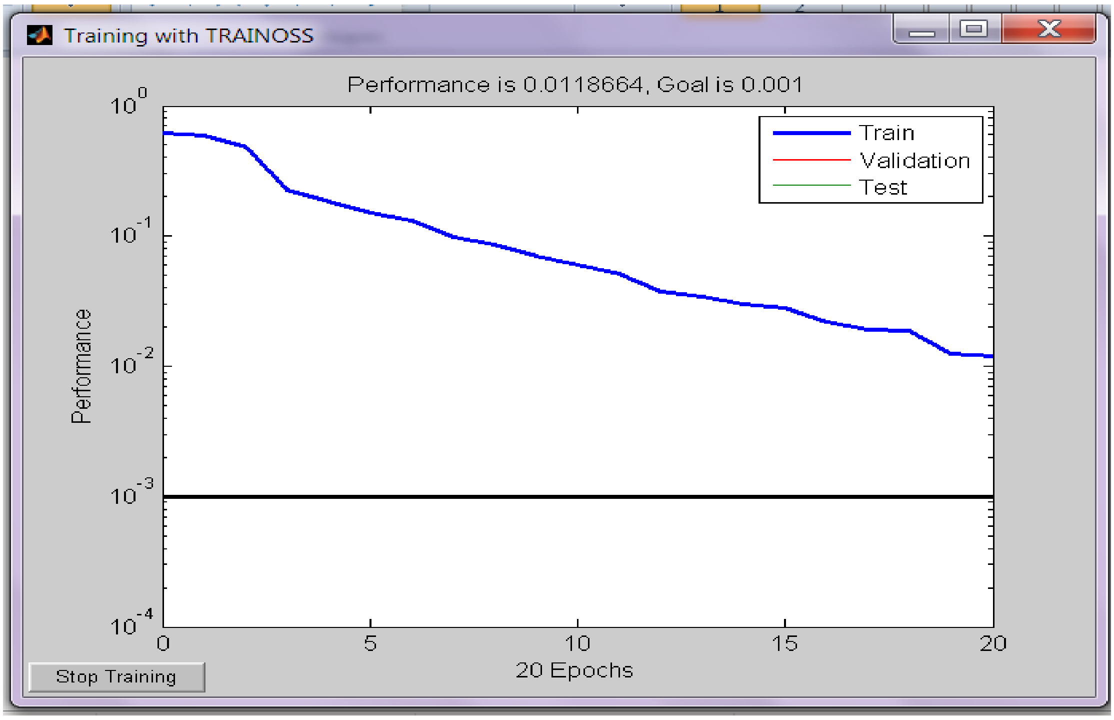



Figure 4.14: Training by neural network after 20 Epochs

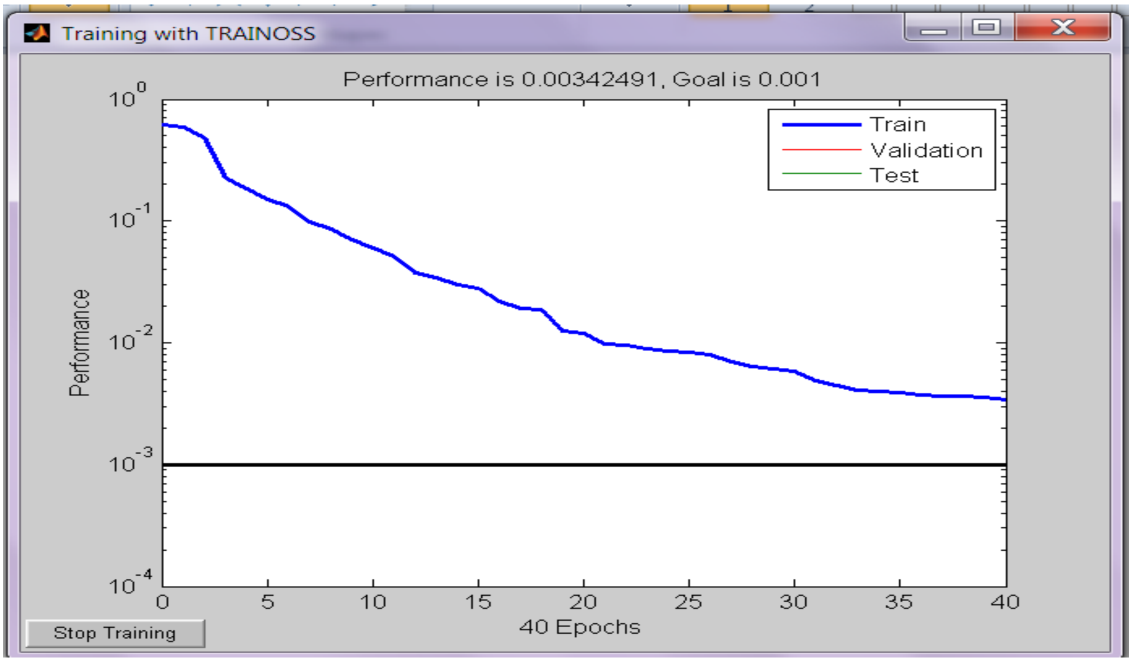



Figure 4.15: Training by neural network after 40 Epochs

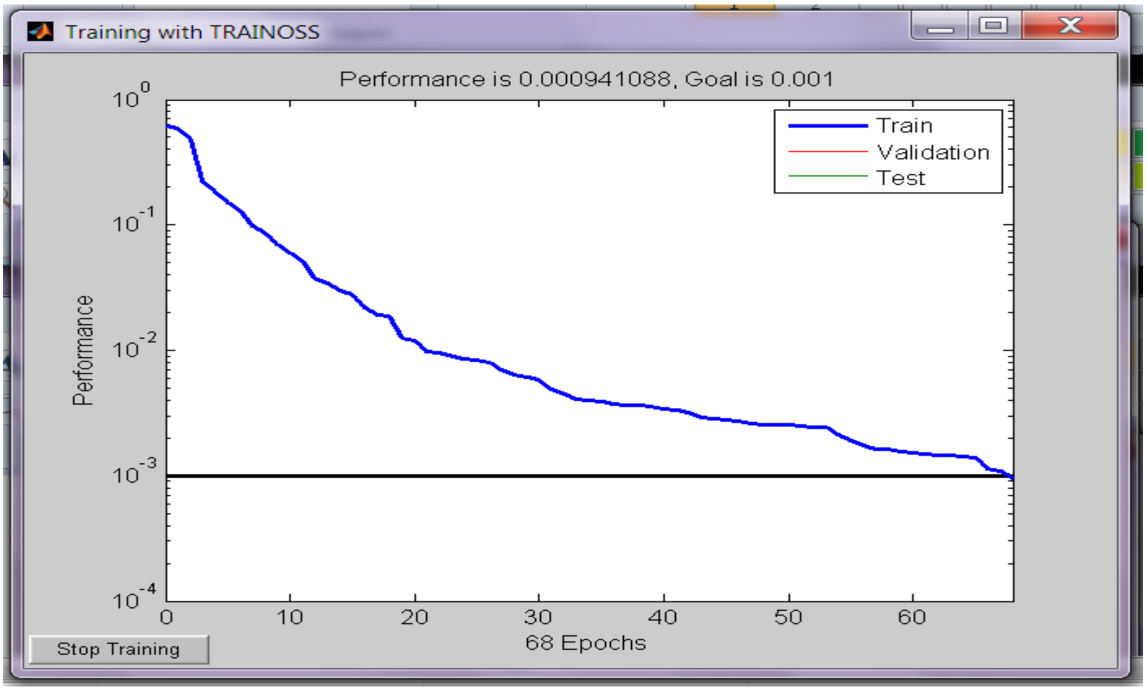



Figure 4.16: Training by neural network after 68 Epochs

## 4.7 Minutiae Matching

After successfully extracting the set of minutia points of two fingerprint images to be tested, we perform Minutiae Matching to check whether they belong to the same person or not. We use an iterative ridge alignment algorithm to first align one set of minutiae with respect to other set and then carryout an elastic match algorithm to count the number of matched minutia pairs.

### 4.7.1 Minutiae Alignment

i. Let $I_1$ & $I_2$ be the two minutiae sets given by,

$I_1 = \{m_1, m_2, m_3, \ldots\ldots, m_M\}$ Where $m_i = (xi, yi, \theta i)$

$I_2 = \{n_1, n_2, n_3, \ldots\ldots\ldots, n_N\}$ Where $n_i = (xi', yi', \theta i')$

Now we choose one minutia from each set to find the ridge correlation factor between them.

The ridge associated with each minutia is represented as a series of *x*-coordinates *(x1, x2…xn)* of the points on the ridge. A point is sampled per ridge length L starting from the minutia point, where the L is the average inter-ridge length. In addition, n is set to 10 unless the total ridge length is less than 10*L.

Therefore, the similarity of correlating the two ridges is derived from:

$$s = \sqrt{\frac{\sum_{i=0}^{m} xiXi}{\sum_{i=0}^{m} xi^2\, Xi^2}}$$

Where *(xi,…..,xn)* and *(Xi,……,XN)* are the set of x-coordinates for each of the two minutia chosen. Moreover, m is minimal one of the n and N value. If the similarity score is larger than 0.8, then go to step 2, otherwise continue to match the next pair of ridges.

ii. The approach is to transform each set according to its own reference minutia and then do match in a unified *x-y* coordinate.

Let M *(x, y,θ)* be reference minutia found from step 1. For each fingerprint, translate and rotate all other minutiae *(xi, yi, θi)* with respect to the M according to the following formula:

$$\begin{pmatrix} xi\_new \\ yi\_new \\ \theta i\_new \end{pmatrix} = TM * \begin{bmatrix} (xi - x) \\ (yi - y) \\ (\theta i - \theta) \end{bmatrix}$$

Where *(x, y,θ)* is the parameters of the reference minutia, and TM is

$$TM = \begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

The new coordinate system is originated at reference minutia M and the new *x*-axis is coincident with the direction of minutia M. No scaling effect is taken into account by assuming two fingerprints from the same finger have nearly the same size. So we get transformed sets of minutiae $I_1^{'}$ $and$ $I_2^{'}$.

### 4.7.2 Minutiae Match

An elastic string *(x, y,θ)* match algorithm is used to find number of matched minutia pairs among$I_1^{'}$ $and$ $I_2^{'}$. According to the elastic string match algorithm minutia 'mi' in $I_1^{'}$ and a minutia 'mj' in $I_2^{'}$ are considered "matching," if the spatial distance (*sd*) between them is smaller than a given tolerance *r* and the direction difference (*dd)* between them is smaller than an angular tolerance Θ.

$$sd = \sqrt{(xi - xj)^2 + (yi - yj)^2} \leq r$$

$$dd = min(|\theta i - \theta j|, 360 - |\theta i - \theta j|) \leq \Theta$$

Let mm(.) be an indicator function that returns 1 in the case where the minutiae mi and mj match according to above equations.

$$mm(mi, mj) = \begin{cases} 1, & sd(mi, mj) \leq r \; and \; dd(mi, mj) \leq \Theta \\ 0, & otherwise \end{cases}$$

Now the total number of matched minutiae pair given by,

$$num(matched\ minutiae) = \Sigma\, mm(mi, mj)$$

And final match score is given by

$$Match\ Score = \frac{num(matched\ minutiae)}{\max(number\ of\ minutiae\ in\ I_1^{'}, I_2^{'})} * 100$$

Its ranges vary from 0 to 100. If the Match Score is larger than a pre-specified threshold, the two fingerprints are from the same finger.

## 5.1 Result

Experiments were done in around 100 images and some of the results of them being shown below. Result is shown on the bases of two-fingerprint image, as shown in Figure 5.1and Figure 5.5. Table 5.1 and Table 5.3 are show the buffrication points while Table 5.2 and Table 5.4 shows termination points location in ROI of Fingerprint 1 and Fingerprint 2 respectively.

Figure 5.1: Fingerprint 1

Figure 5.2: Enhance Image of fingerprint 1 using Histogram equalization

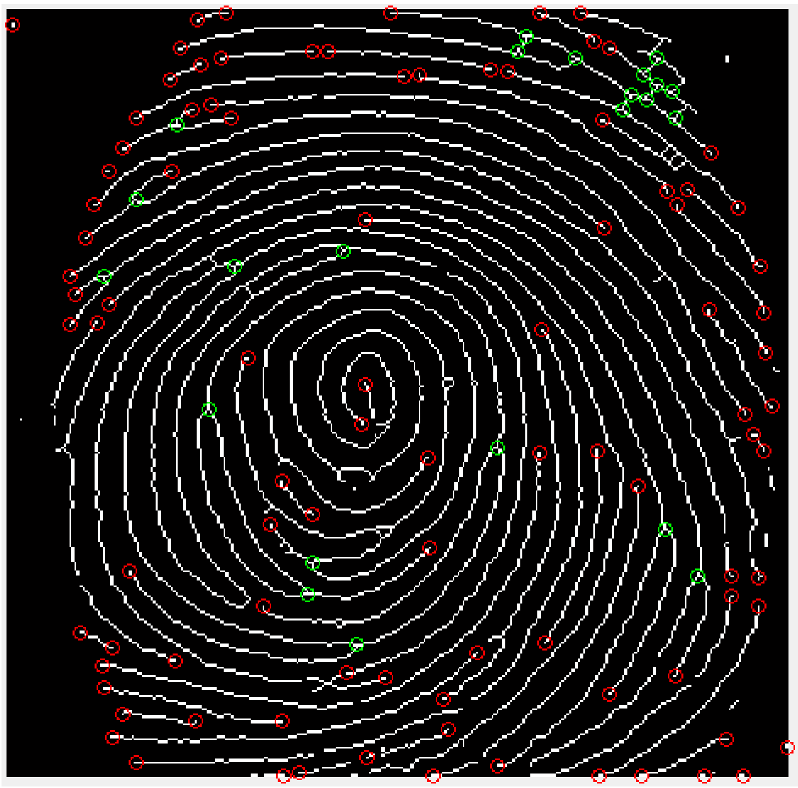


Figure 5.3: Minutiae points of fingerprint 1

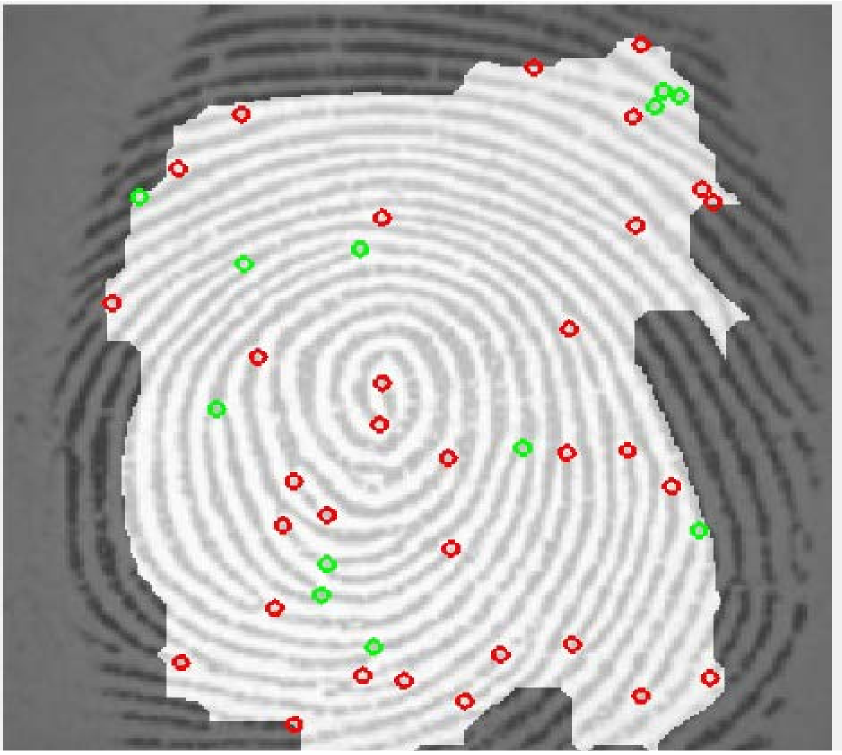


Figure 5.4: Minutiae points in ROI of fingerprint 1

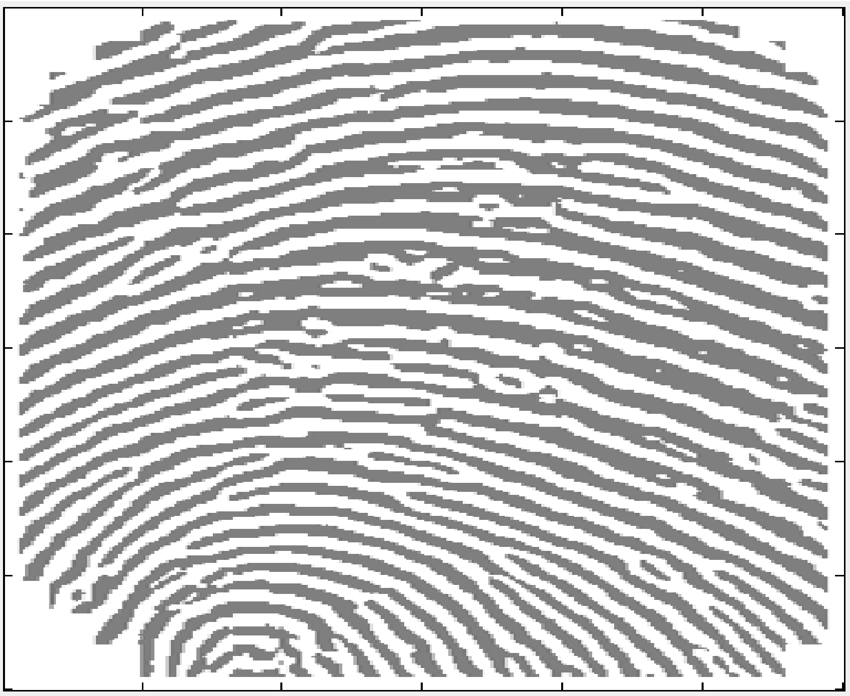


Figure 5.5: Fingerprint 2

Figure 5.6: Enhance Image of fingerprint 2 using histogram equalization

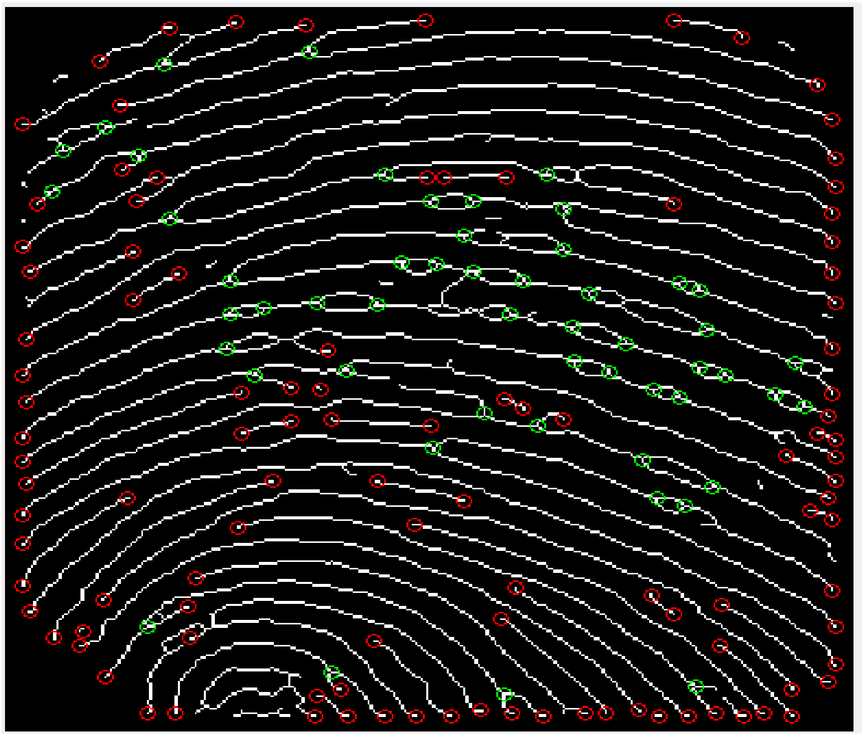


Figure 5.7: Minutiae points of fingerprint 2

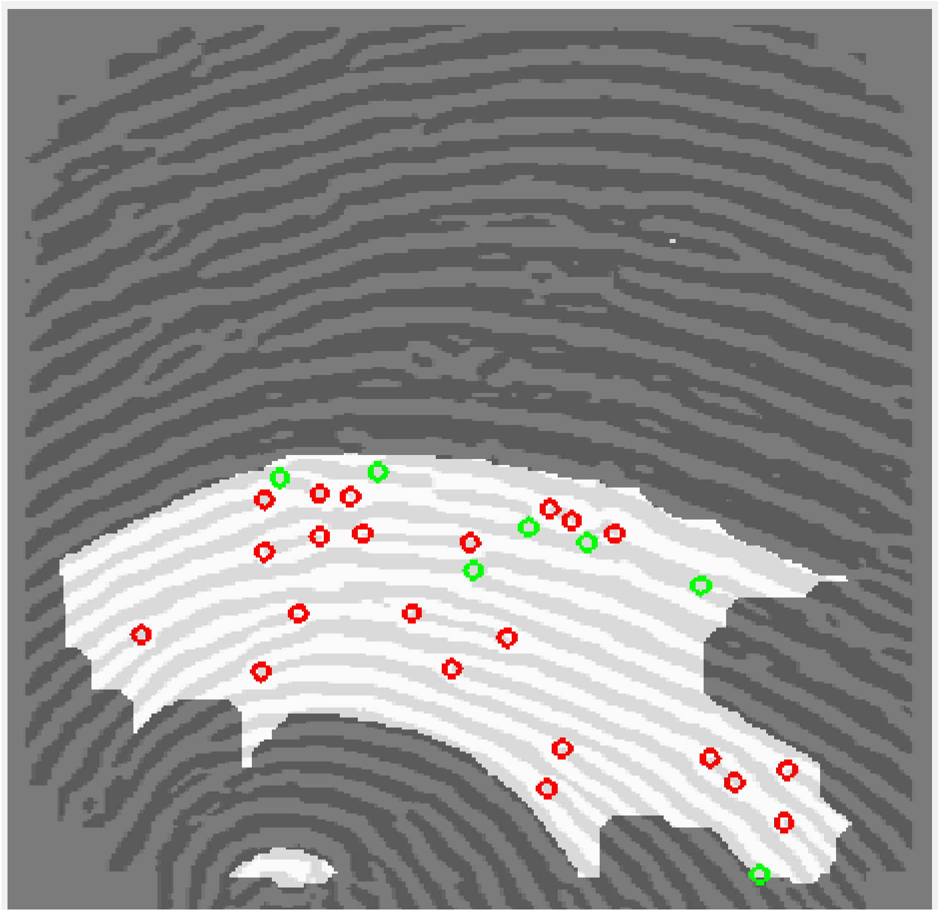


Figure 5.8: Minutiae points in ROI of fingerprint 2

| X | Y | Angle 1 | Angle2 | Angle3 |
|---|---|---|---|---|
| 240 | 34 | 2.62 | -2.09 | 0.00 |
| 246 | 26 | 2.62 | 1.05 | -0.52 |
| 237 | 40 | 3.14 | 1.05 | -0.79 |
| 50 | 75 | -2.36 | 1.57 | -0.52 |
| 130 | 95 | 2.36 | -2.36 | 0.52 |
| 88 | 101 | 2.62 | -1.57 | 0.52 |
| 78 | 157 | -2.36 | 1.57 | -0.79 |
| 189 | 172 | -2.62 | 1.57 | -1.05 |
| 253 | 204 | -2.36 | 1.57 | -0.52 |
| 118 | 217 | 2.62 | -1.57 | 0.52 |
| 116 | 229 | -2.62 | 2.09 | 0.00 |
| 135 | 249 | 2.62 | -2.09 | 0.00 |

Table 5.1: Buffrication Points Location in ROI of fingerprint 1

| X | Y | Angles |
|---|---|---|
| 232 | 16 | -0.52 |
| 193 | 25 | -0.52 |
| 87 | 43 | 3.14 |
| 229 | 44 | -0.52 |
| 64 | 64 | -2.62 |
| 254 | 72 | 2.09 |
| 258 | 77 | -1.57 |
| 138 | 83 | 0.00 |
| 230 | 86 | 2.36 |
| 40 | 116 | 0.79 |
| 206 | 126 | -0.79 |
| 138 | 147 | -1.05 |
| 137 | 163 | 2.09 |
| 227 | 173 | -1.57 |
| 205 | 174 | -2.09 |
| 162 | 176 | -2.36 |
| 102 | 202 | -1.05 |
| 106 | 185 | -1.57 |
| 243 | 187 | -2.09 |
| 118 | 198 | 2.62 |
| 163 | 211 | -2.09 |
| 99 | 234 | -1.05 |
| 207 | 248 | 0.79 |
| 181 | 252 | -2.36 |
| 65 | 255 | 3.14 |
| 131 | 266 | 0.00 |
| 257 | 261 | 1.05 |
| 146 | 262 | -2.62 |
| 232 | 268 | 1.05 |
| 168 | 270 | 0.52 |
| 106 | 279 | 3.14 |

Table 5.2: Termination Points Location in ROI of fingerprint 1

| X | Y | Angle 1 | Angle 2 | Angle 3 |
|---|---|---|---|---|
| 121 | 151 | -2.62 | 1.05 | -0.52 |
| 89 | 153 | 2.62 | 1.05 | -0.52 |
| 170 | 169 | -2.62 | 1.57 | -0.52 |
| 189 | 174 | 3.14 | 1.05 | -0.79 |
| 152 | 183 | 2.62 | -1.05 | 0.79 |
| 226 | 188 | 2.62 | -1.05 | 0.52 |
| 245 | 282 | 2.62 | -1.57 | 0.00 |

Table 5.3: Buffrication Points Location in ROI of fingerprint 2

| X | Y | Angle |
|---|---|---|
| 102 | 158 | 2.62 |
| 84 | 160 | -2.62 |
| 117 | 163 | -0.52 |
| 116 | 171 | -0.52 |
| 198 | 171 | 2.36 |
| 102 | 172 | -2.62 |
| 151 | 174 | 3.14 |
| 84 | 177 | 0.52 |
| 95 | 197 | -2.62 |
| 132 | 197 | -0.52 |
| 44 | 204 | -2.62 |
| 163 | 205 | 3.14 |
| 145 | 215 | 0.00 |
| 83 | 216 | 0.52 |
| 229 | 244 | -1.05 |
| 254 | 248 | -0.52 |
| 237 | 252 | 2.36 |
| 176 | 254 | -0.52 |
| 253 | 265 | -0.52 |

Table 5.4: Termination Points Location in ROI of fingerprint 2

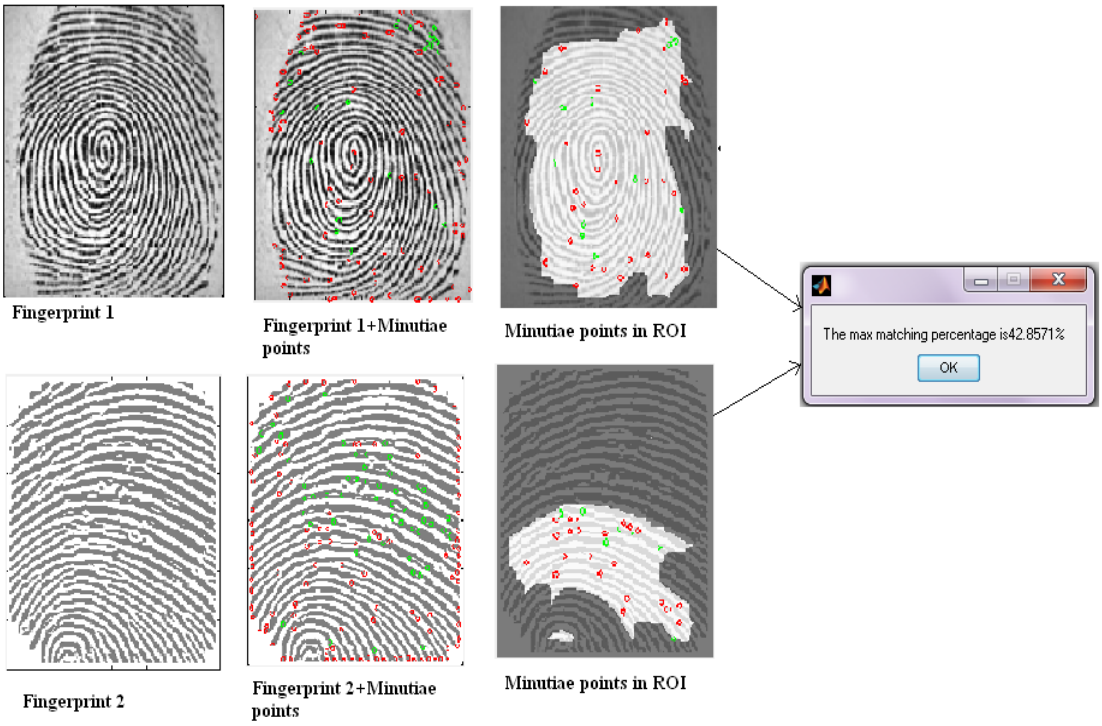



Figure 5.9: Minutiae points Matching

# Chapter-6: Conclusion and Future scope

Image quality is related directly to the final performance of fingerprint identification systems. Good quality fingerprint images need only minor preprocessing and enhancement for accurate feature detection algorithm. But low quality fingerprint images need preprocessing to increase contrast, and reduce different types of noises. When some background region is included in the segmented regions of interest, noisy pixels also generate many spurious minutiae because these noisy pixels are also enhanced during preprocessing and enhancement steps. In this thesis, enhancement using histogram equalization, segmentation using Morphological operations, minutia marking by specially considering the triple branch counting, minutia unification by decomposing a branch into three terminations, an alignment-based elastic matching algorithm has been developed for minutia matching were implemented. The proposed alignment-based elastic matching algorithm is capable of finding the correspondences between minutiae without resorting to exhaustive search.

There is a scope of further improvement in terms of efficiency and accuracy, which can be achieved by improving the image enhancement techniques or by improving the hardware to capture the image. So that the input image to the thinning stage could be made, better this could improve the future stages and the outcome.

# References

[1] D. Maltoni, D. Maio, A.K. Jain, S. Prabhakar, Handbook of Fingerprint Recognition,Second Edition. London Springer, 2009.

[2] B.M. Mehtre and B. Chatterjee,“Segmentation of fingerprint images–a composite method”, Pattern Recognition, 22(4):381–385, 1989.

[3] Lin Hong,"Automatic Personal Identification Using Fingerprints", Ph.D. Thesis, 1998.

[4] Ishmael S. Msiza, Brain Leke-Betechuoh, Fulufhelo V. Nelwamondo and Ntsika Msimang,“A Fingerprint Pattern Classification Approach Based on the Coordinate Geometry of Singularities”,Proceedings of the IEEE International Conference on systems, Man, and Cybernetics San Antonio, TX, USA-October 2009.

[5] Raffaele Cappelli, Alessandra Lumini, Dario Maio and Davide Maltoni, “Fingerprint Classification by Directional Image Partitioning”,IEEE Transactions On Pattern Analysis And Machine Intelligence, vol. 21, no. 5, pp. 402-421, 1999.

[6] Abdolreza Moghadassi, Fahime Parvizian, SayedMohsen Hosseini,“A New Approach Based on Artificial Neural Networks for Prediction of High Pressure Vapor-liquid Equilibrium”,Australian Journal of Basic and Applied Sciences, 3(3): 1851-1862, 2009, ISSN 1991-8178.

[7] Neural network URL http://www.ai.rug.nl/vakinformatie/ias/slides/3_NeuralNetworksAdaptation.pdf Retrieved June 2010.

[8] S.N.Sivanandam, S.Sumathi, S.N. Deepa, Introduction To Neural Networks using MATLAB 6.0,Second Reprint, Tata Mcgraw Hill, 2006.

[9] Artificial Neural Networks URL http://www.neuworldfinancial.com/ann.htm Retrieved June 2010.

[10] Artificial Intelligence: An Introductory Course URL www.icgst.com Retrieved June 2010.

[11] E. Sackinger, B.E. Boser, J. Bromley, Y. Lecun, and L. D. jacket, “Application of the ANNA neural network chip to high speed character recognition”, IEEE transaction on neural networks, vol.3.3 pp. 498-505, 1992.

[12] Pravesh Kumar,“Use of Fuzzy Set and Neural Network to Extract Fingerprint Minutiae Points and Location”, Master of Engineering,Thesis-2009.

[13] FVC2004-Third International Fingerprint Verification Competition URL http://bias.csr.unibo.it/fvc2004/default.asp Retrieved June 2010.

[14] Image Restoration URL http://restoreinpaint.sourceforge.net/ Retrieved June 2010

[15] Xiping Luo, Jie Tian, Knowledge Based Fingerprint Image Enhancement, 15th ICPR, Vol.4, P783-786, Barcelona, 2000.

[16] Wang Sen, Zhang Wei-wei, Wang Yang-sheng,“ New Features Extraction and Application in Fingerprint Segmentation”,Acacta Automatica Sinica Vol. 29,No.4,July2003.

[17] Madasu Hanmandlu, Vamsi Krishna Madasu, Shantaram Vasikarla: A Fuzzy Approach to Texture Segmentation. ITCC (1) 2004: 636-642.

[18] Tsai-Yang Jea, Venu Govindaraju: A minutia-based partial fingerprint recognition system. Pattern Recognition 38(10): 1672-1684 (2005).

[19] Hamed Vahdat Nejad, Hameed Reza Pourreza, and Hasan Ebrahimi,“A Novel Fuzzy Technique for Image Noise Reduction” ,Proceedings Of World Academy Of Science, Engineering And Technology Volume 14 August 2006 ISSN 1307-6884.

[20] Jianping Yin, En Zhu, Xuejun Yang, Guomin Zhang, Chunfeng Hu: Two steps for fingerprint segmentation. Image Vision Comput. 25(9): 1391-1403 (2007).

[21] Hoi Le and The Duy Bui (2009), Online fingerprint identification with a fast and distortion tolerant hashing, Journal of Information Assurance and Security, 4 (2), pp 117-123.

[22] B.Chanda, D.Dutta Majumder, Digital Image Processing and Analysis, Prentice-Hall of India 2000.

[23] N. Ratha, S. Chen and A.K. Jain, "Adaptive Flow Orientation Based Feature Extraction in Fingerprint Images", Pattern Recognition, Vol. 28, pp. 1657-1672, November 1995.

[24] L.C. Jain, U.Halici, I. Hayashi, S.B. Lee and S.Tsutsui., Intelligent biometric techniques in fingerprint and face recognition.,the CRC Press 1999.

[25] Andrew Blais and David Mertz , “An Introduction to Neural Networks – Pattern Learning with Back Propagation Algorithm”, Gnosis Software, Inc., July 2001

[26] E. Czogala, J. Leski, “Fuzzy and Neuro-Fuzzy Intelligent Systems”, Physica-Verlag Heidelberg, New York, 2000.

[27] H. Demuth, and M. Beale, “Neural Network Toolbox for use with MATLAB”, The MathWorks Inc,1998.

[28] Hassoun, M. H., Fundamentals of Artificial Neural Networks, MIT Press,1995.

[29] Rafael C. Gonzalez, Richard E. Woods, Digital Image Processing, second edition.Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA 2004.

[30] Peter H. Sydenham, Richard Thorn, Handbook of Measuring System Design, John Wiley & Sons, Ltd. ISBN: 0-470-02143-8, 2005.

[31] NeetaNain , Deepak BM,Dinesh Kumar, Manisha Baswal and BijuGautham, "Optimized Minutiae-based Fingerprint Matching", IASTED Intermnational Conference on Signal and Image Processing (SIP 2008), August 18- 20, 2008, Kailua-Kona, USA.

# List of Paper Published/Communicated

[1] Kushal Veer Singh, Tarun Kumar and Karun Verma, "Preprocessing Stage for Image Recognition using MATLAB",National Conference on Next Generation Computing (NGC-10), GITM Gurgaon, India, March 2010 (Published).

[2] Kushal Veer Singh, Karun Verma, "Fingerprint identification using minutia alignment-based matching", International Conference on Advances in Computing and Communication, NIT Hamirpur, India, December 2010 (Communicated).