Use of Fuzzy Set and Neural Network to Extract Fingerprint Minutiae Points and Location

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

Master of Engineering

in

Software Engineering

Submitted By:

Pravesh Kumar

(80731019)

Under the supervision of:

Mr. Karun Verma

Lecturer CSED

COMPUTER SCIENCE AND ENGINEERING DEPARTMENT
THAPAR UNIVERSITY
PATIALA – 147004

June 2009

Certificate

I hereby certify that the work which is being presented in the thesis entitled, **"Use of Fuzzy Set and Neural Network to Extract Fingerprint Minutiae Points and Location"**, 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.

Pravesh

Pravesh Kumar

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

Karun Verma

Karun Verma
Lecturer, CSED
Thapar University, Patiala

Countersigned by

13/07/09.

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

RK Sharma 15/7/09

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

Acknowledgment

First and foremost, I would like to express my sincere gratitude to my advisor **Karun Verma**, Lecturer, Computer Science & Engineering Department for his immense help, guidance, stimulating suggestions and encouragement all the time. He always provide a motivating and enthusiastic atmosphere to work with, it was a great pleasure to do this thesis under his supervision

I am thankful to Dr Rajesh Bhatia, Head of Computer Science & Engineering Department, TU, Patiala, for the motivation and inspiration that triggered me for the thesis work. I would also like to thank all the staff members who were always there at the need of the hour and provided with all the help and facilities, which I required for the completion of the thesis.

Last but not the least, I express my heartfelt thanks to my parents and all of my friends for encouraging me and providing me useful information during my work.

Finally, my special thanks go to authors whose works I have consulted and quoted in this work. Finally, my special thanks go to authors whose works I have consulted and quoted in this work.

Pravesh

Pravesh Kumar
80731019
M.E.(Software Engineering)-2nd year
Computer Science & Engineering Department
Thapar University
Patiala -147004

Abstract

In biometric identification, fingerprint recognition is most popular and widely used method Fingerprints were used as a means of positively identifying a person as an author of the document and are used in law enforcement. Fingerprint recognition has a lot of advantages, a fingerprint is compact, unique for every person, and stable over the lifetime. A predominate approach to fingerprint technique is the uses of minutiae. This thesis presents an investigation and comparative study to extract minutiae points in a particular fingerprint image. In most cases, fingerprint images available are not of good quality; they may be corrupted and degraded due to variation in skin and effective condition. So first a fuzzy logic based image enhancement method has been applied to obtain a more reliable estimation of minutiae points and their location and then a different algorithm used to extract them. Neural network is used to give the training to the location of these minutiae point and to improve the performance of the system. All the implementation work has been done in MATLAB 7.0 Image Processing Toolbox. Experimental result shows that proposed algorithm gives the better result compare to others.

Table of Contents

List of Tables

List of Figures

Chapter 1
Introduction

1.1 Fingerprints

A fingerprint is comprised of ridges and valleys. The ridges are the dark area of the fingerprint and the valleys are the white area that exists between the ridges. Figure 1.1 shows an example of fingerprint image. According to [23], the biological properties of fingerprint formation are well understood and fingerprints have been used for identification purposes from centuries. Fingerprint has been widely used for identification of criminals since 20th century. So most people do not feel comfortable in provide their fingerprints in many applications. Fingerprint-based authentication is very popular in a number of civilian and commercial applications such as, welfare disbursement, cellular phone access, and laptop computer log-in due to its many advantages such that fingerprint based biometric system provide high degree of confidence in positive identification and also it can be embedded in various system (e.g., cellular phones). The availability of cheap and compact solid state scanners as well as robust fingerprint matchers are two important factors in the popularity of fingerprint-based identification systems. There are also many disadvantages of fingerprint based authentication systems as compared to other biometrics. For example, approximately 4% of the population does not have good quality fingerprints, manual workers get regular scratches on their fingers which poses a difficulty to the matching system, finger skin peels off due to weather, fingers develop natural permanent creases, temporary creases are formed when the hands are immersed in water for a long time, and dirty fingers can not be properly imaged with the existing fingerprint sensors. Also users should have a good knowledge to capture the fingerprint, so it is not suited for many applications such as surveillance. [23].

Figure 1.1: Fingerprint

1.2 Minutiae Points

In the biometric process of fingerprint scanning, minutiae are specific points in a finger image. There are two main types, known as ridge endings and bifurcations. Sometimes, other details, such as the points at which scars begin or terminate, are considered minutiae. A ridge ending is defined as the point where the ridge ends abruptly and the ridge bifurcation is the point where the ridge splits into two or more branches.

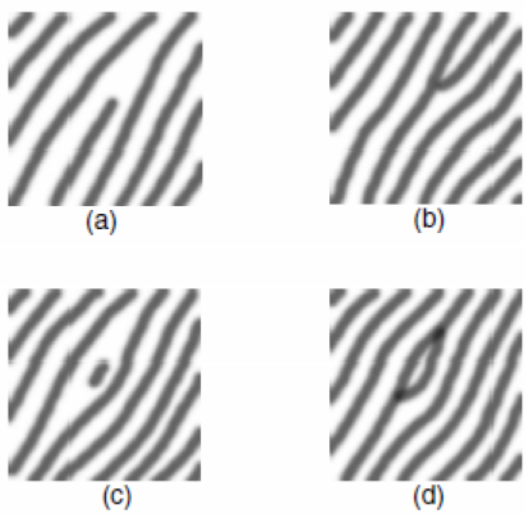

Figure 1.2: (a) Ridge ending; (b) Buffrication; (c) Ridge dot; (d) Ridge enclosure

1.3 Fingerprint Classification

Everyday, a large number of fingerprints image is acquired and stored in a big range of applications including forensics, access control, and driver license registration. 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. There are three main structures that make up fingerprints. These are loops, whorls and arches.

1.3.1 Loops

Loops are making of one or more ridges entering from one side, curving, and then going out the same side it entered. The ridges in loops double back on themselves. All loops have elements called a delta and a core. The delta is a triangular area usually shaped like a T-junction, while a core is the centre of the pattern. About 65% of fingerprints have loops.

Loops can be divided into two groups:

- **Radial loops:** these flow downward and toward the radius (or the thumb side)
- **Ulnar loops:** these flows toward the ulnar (or the little finger side). The ulnar loop is more common.

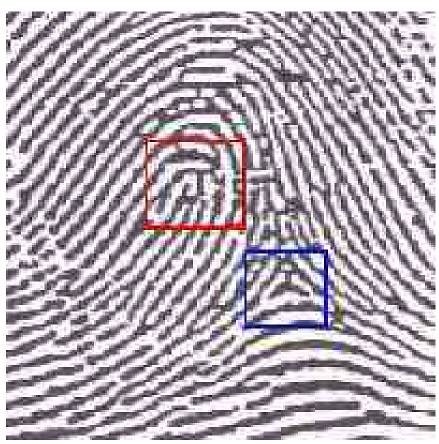

Figure 1.3: Loop

1.3.2 Whorls

Whorls have a circular pattern and have at least two deltas and a core. Whorls look a little like target shapes or whirlpools – circles within circles. Whorls make up 35% of patterns seen in human fingerprints and can be sub-grouped into four categories:

- **Plain whorls:** these are either concentric circles like a bull's eye or spirals like a wound spring.
- **Central pocket loop whorls:** these resemble a loop with a whorl at its end.
- **Double loop whorls:** these occur when two loops collide to produce an "S" shaped pattern.
- **Accidental loop whorls:** these are slightly different from other whorls and are irregular.

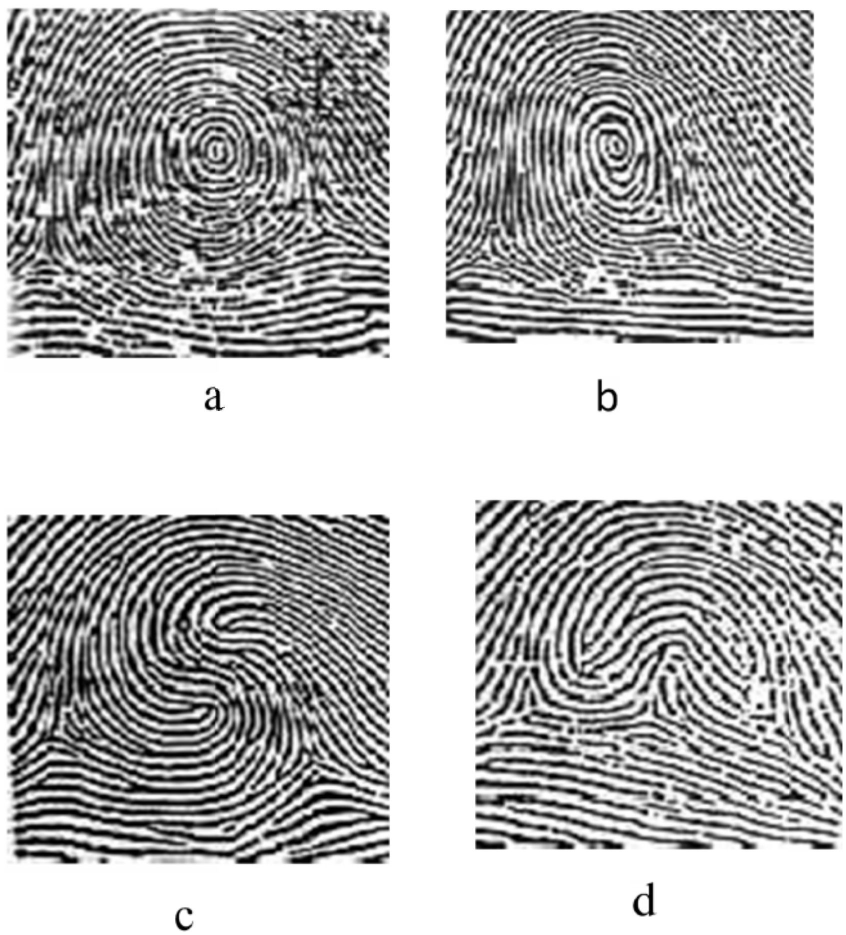

Figure 1.4(a): Plain Whorl, (b): Central Pocket Loop (c): Double Loop Whorl (d): Accidental Whorl.

1.3.3 Arches

Arches are the least common pattern making up only 5% of all pattern types. Arches are ridgelines that rise in the centre and create a wave like pattern. The ridges enter from one side and exit the other side with a rise in the middle. They do not have a delta or a core and can be broken into two sub-groups:

- **Plain arch:** This has a gentle rise.
- **Tented arch:** This has a steeper rise than plain arches.

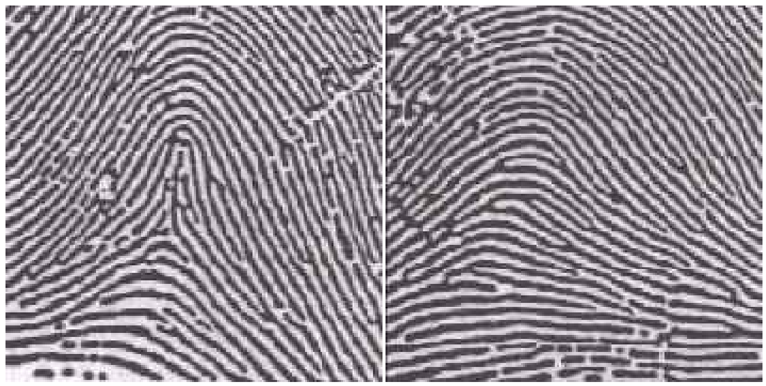

Figure 1.5: Tented Arch and Plain Arch

1.4 Fingerprint Matching Method:

Generally there are two methods are use for fingerprint matching:

- Correlaton based fingerprint matching.
- Minutiae point based fingerprint matching.

Minutiae-based matching is most popular and widely used method. Minutiae-based techniques first find minutiae points and then map their relative placement on the finger. However, there are some difficulties when using this approach. It is difficult to extract the minutiae points accurately when the fingerprint is of low quality. Also this method does not take into account the global pattern of ridges and furrows.

The correlation-based method is able to overcome some of the difficulties of the minutiae-based approach. In this method two fingerprints are superimposed and correlation between pixels is computed for different alignments. However, it has some of its own shortcomings. Correlation-based techniques require the precise location of a registration point and are affected by image translation and rotation. Figure 1.6 and 1.7 shows the example of both of these two methods.

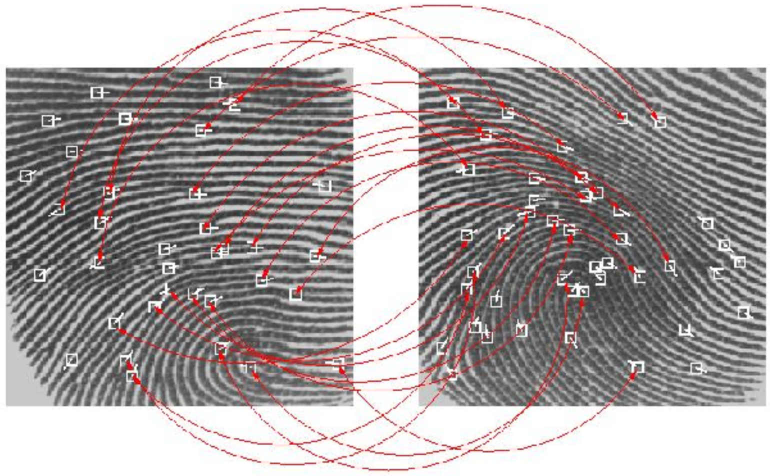

Figure 1.6: Correlation Based Matching

Figure 1.7: Minutiae Based Matching

Chapter 2

Neural Network, Fuzzy Set and Image Processing

2.1 Neural Network

Artificial neural networks, commonly referred to as neural networks are systems that are constructed to make use of some organizational principles similar to those of the human brain. They represent a promising new generation of information processing systems. Neural networks have a large number of highly interconnected processing elements (nodes or units). A neural network is a large and heavily parallel distributed processor inspired by the real biological neuron in the brain; therefore, it has the ability to learn, recall, and generalize as a consequence of training patterns or data. There are two types of neural network, first is biological neural network and other is artificial neural network. Basic details of these types of the neural network are given below:

2.1.1 Biological Neural Networks

According to the theory of modern brain science, the brain is made up of different processing elements called neurons. There are approximately 10^{11} neurons of different shapes in human brain. The neurons are like a polarized cell. The neurons have three main regions to its structure. First is cell body that is heart of the cell and contain the nucleus and maintain protein synthesis. Second is dendrites which branch out is in a treelike structure that receive signal from other neuron. Third is axon by which a neuron sends information to other neurons. The end of an axon splits into strands. Each strand terminates in a small bulblike shape called a synapse (There are approximately 10^4 synapses per neuron in a human brain), where the neuron introduces its signal to the neighbouring neurons. The signal in the form of an electric impulse is then received by dendrites. This type of signal transmission involves a complex chemical process in which specific transmitter substances are released from the sending side. This raises or lowers the electric potential inside the cell body called soma of the receiving neuron. The receiving neuron fires if its electric potential reaches a certain level called threshold, and a pulse or action potential of fixed strength and duration is sent out through the axon to synaptic junctions to other neurons. After firing, a neuron has to wait for a period of time called the refractory period before it can fire again. Synapses are excitatory if they let passing impulses cause the firing of the receiving neuron, or inhibitory if they let passing impulses hinder the firing of the neuron [14]

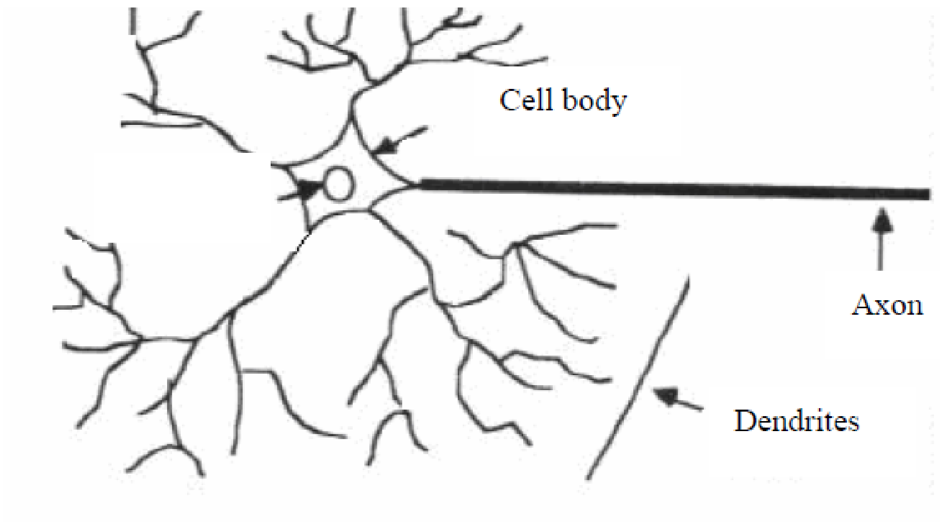

Figure 2.1: Biological Neuron

2.1.2 Artificial Neural Networks

Artificial neural network research is developed to make mathematical models of its biological counterpart in order to mimic the capabilities of biological neural structures

Warren McCulloch and Walter Pitts [18] introduced the first mathematical model of neuron. Figure 2.2 shows an example of this model. It is known as the McCulloch-Pitts model, it does not possess any learning or adaptation capability. Many of the later neural network models use this model as the basic building block. This model consists of a single neuron, which receives a set of inputs $(x_1, x_2, x_3 \ldots \ldots x_n)$. This set of inputs is multiplied by a set of weights ($w_1, w_2 \ldots \ldots w_n$). Here, weights are referred to as strengths of the synapses. These weighted values are then summed and the output is passed through an activation (transfer) function. The activation function is also referred to as a squashing function in that it squashes (limits) the permissible range of the output signal to some finite value [14].

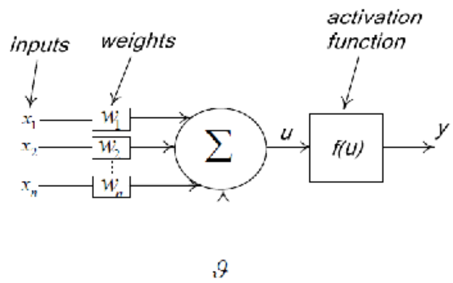

Figure 2.2: Artificial Neuron

2.1.3 Transfer Function

The behaviour 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:

- **Linear (or ramp):** the output activity is proportional to the total weighted output.
- **Threshold:** the output is set at one of two levels, depending on whether the total input is greater than or less than some threshold value.
- **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.

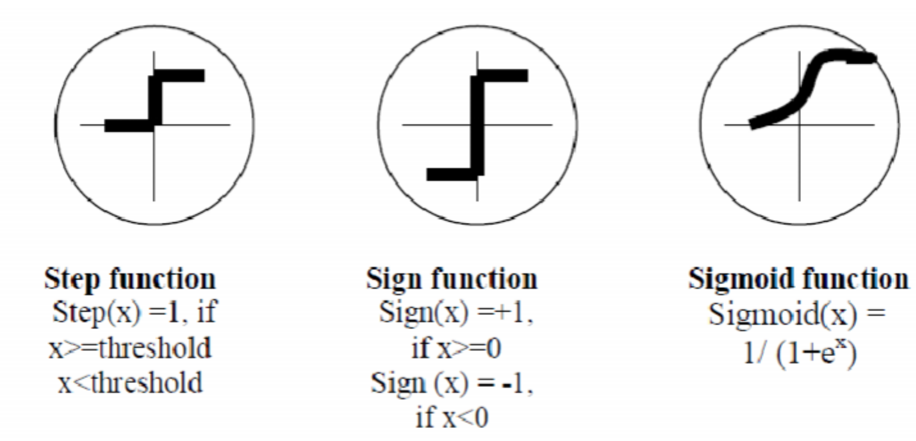

Figure2.3: Transfer Function

2.1.4 Type of Neurons:

Neuron can be classified in two types namely, simple neuron and complicated neuron. Following is the basic information of these two given in detail:-

- **Simple neuron** An artificial neuron is a device with many inputs and one output. The neuron has two modes of operation; the training mode and the using mode. In the training mode, the neuron can be trained to fire (or not), for particular input patterns. In the using mode, when a taught input pattern is detected at the input, its associated output becomes the current output. If the input pattern does not belong in the taught list of input patterns, the firing rule is used to determine whether to fire or not. Figure 2.4 show an example of simple neuron.

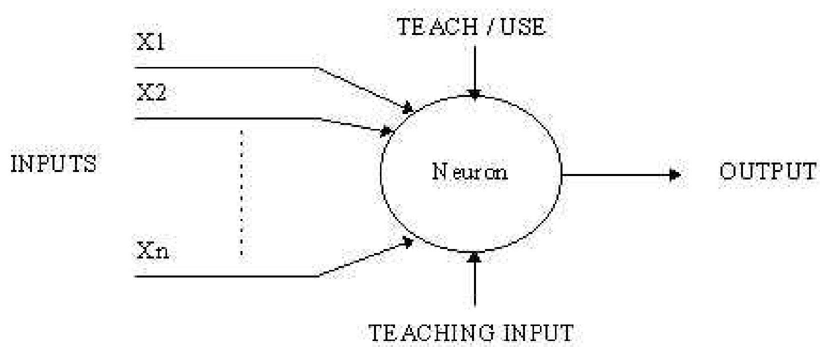

Figure 2.4: Simple Neuron

- **Complicated neuron**: The simple neuron doesn't do anything that conventional computers don't do already. Figure 2.5 is the example of complicated neuron. The difference from the previous model is that the inputs are ‘weighted’; the effect that each input has at decision making is dependent on the weight of the particular input. The weight of an input is a number which when multiplied with the input gives the weighted input. These weighted inputs are then added together and if they exceed a pre-set threshold value, the neuron fires. In any other case the neuron does not fire.

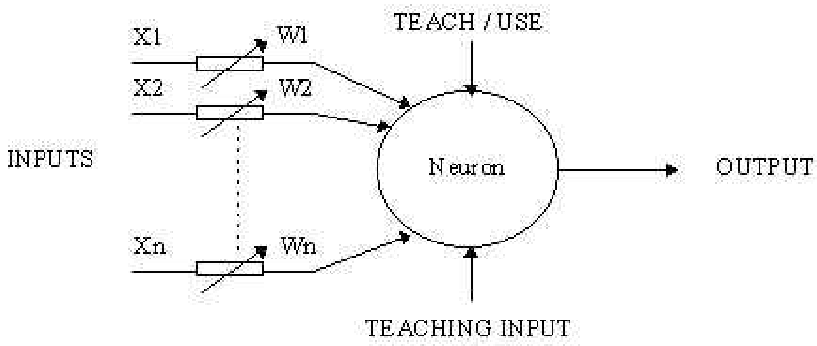

Figure 2.5: A Complicated Neuron

In mathematical terms, the neuron fires if and only if;

$$X_1W_1 + X_2W_2 + X_3W_3 + \ldots > T$$

The addition of input weights and of the threshold makes this neuron a very flexible and powerful one. The complicated neuron has the ability to adapt to a particular situation by changing its weights and/or threshold. Various algorithms exist that cause the neuron to 'adapt'; the most used ones are the Delta rule and the back error propagation.

2.1.5 Architecture of Neural Networks

- **Network Layers:** The commonest type of artificial neural network consists of three groups, or layers, of units: a layer of "input" units is connected to a layer of "hidden" units, which is connected to a layer of "output" units [20].
  - The activity of the input units represents the raw information that is fed into the network.
  - The activity of each hidden unit is determined by the activities of the input units and the weights on the connections between the input and the hidden units.
  - The behaviour of the output units depends on the activity of the hidden units and the weights between the hidden and output units.

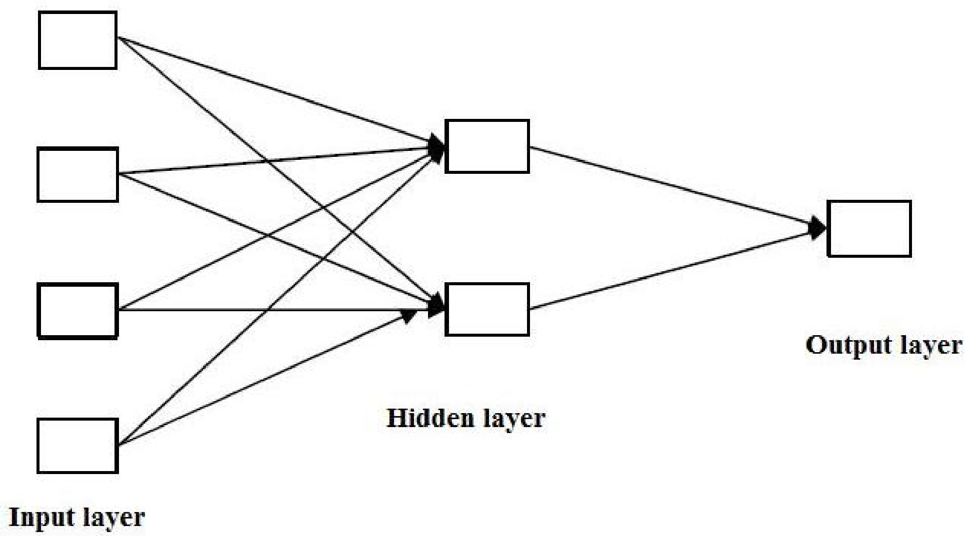

Figure 2.6: Layer in Neural Network

This simple type of network is interesting because the hidden units are free to construct their own representations of the input. The weights between the input and hidden units determine when each hidden unit is active, and so by modifying these weights, a hidden unit can choose what it represents. We also distinguish single-layer and multi-layer architectures. The single-layer organization, in which all units are connected to one another, constitutes the most general case and is of more potential computational power than hierarchically structured multilayer organizations. In multi-layer networks, layer, instead of following a global numbering, often numbers units.

- **Feed-Forward Networks:**
  The feedforward neural network is the first and simplest type of artificial neural networks. In this network, the data moves in only one direction, forward, from the input nodes, through the hidden nodes (if any) and to the output nodes. There are no cycles or loops in the network. Feedforward Neural Networks are Neural Networks that utilize more than one neuron (node) and contain no feedback path within the network. There are two different types of feedforward neural networks, namely Single-Layer and multilayer feedforword neural network.
  - **Single-Layer Feedforward Neural Networks:** A Single-Layer Feedforward Neural Network is shown in Figure 2.7. In the single layer feedforward neural network, there is only one input layer and one output layer. In this

network, several neurons (nodes) can be connecting in parallel to a layer. The network is strictly feedforward, that is, there is no feedback connections from the outputs back to the inputs. Usually, no connections exist between the neurons (nodes) in a particular layer. The network shown in Figure 2.7 is fully connected, that is, all inputs are connected to all the nodes. Partially connected networks are those where some of the connection links are missing [14].

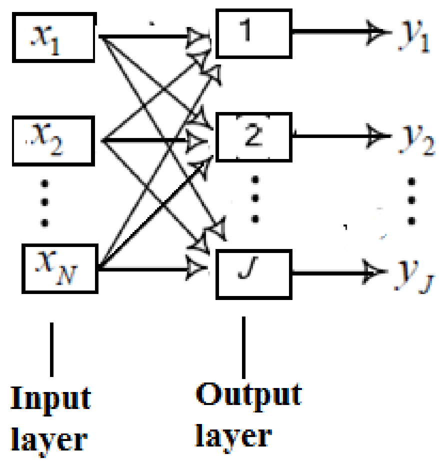

Figure 2.7: Single Layer Feedforword Neural Network

- **Multilayer Feedforward Neural Networks (MFNN):** A Multilayer Feedforward Neural Network is shown in Figure 2.8 is the most widely used neural networks, particularly within the area of systems and control. Similar to the single-layer feedforward neural networks, there is one input layer and one output layer, and no interconnections between the nodes in a particular layer. But different from the single-layer feedforward neural networks, multilayer neural networks have a number of intermediate or hidden layers (any layer between the input and output layers, is called a hidden layer because it is internal to the network and has no direct contact with the external environment) existing between the input and output layer. One, two or even no hidden layers are used for most applications. The small number of hidden layers is due to the fact that the training process becomes too long and tedious if the architecture of the neural network becomes large. In Figure 2.8 one hidden layer is present in this multilayer neural network, where $J \neq K \neq$

N; $J, K, N \in$ R. To get the output from the network, a set of input data is first presented as inputs to the input layer in turn. The outputs from this layer are then fed, as inputs to the first hidden layer, and subsequently the outputs from the first hidden layer are fed, as weighted inputs (the outputs from the first hidden layer are multiplied by the weights), to the second hidden layer. This process carries on until the output layer is reached. An example of a feedforward neural network is the multilayer perceptron (MLP) (commonly called the multilayer feed forward network) [14].

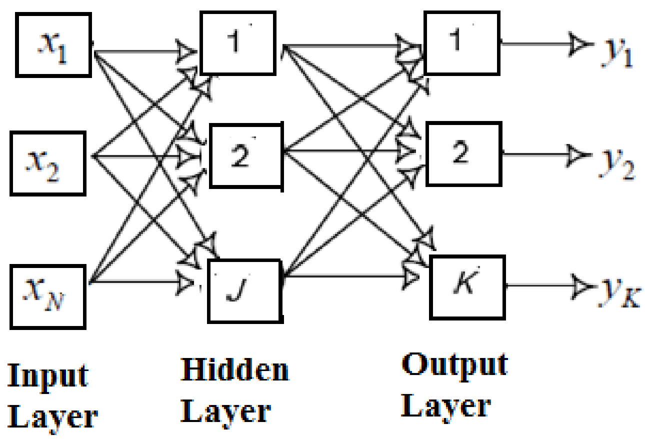

Figure 2.8: Multilayer Feedforward Neural Network

- **Feedback Networks**

  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. Feedback architectures are also referred to as interactive or recurrent, although the latter term is often used to denote feedback connections in single-layer organizations.

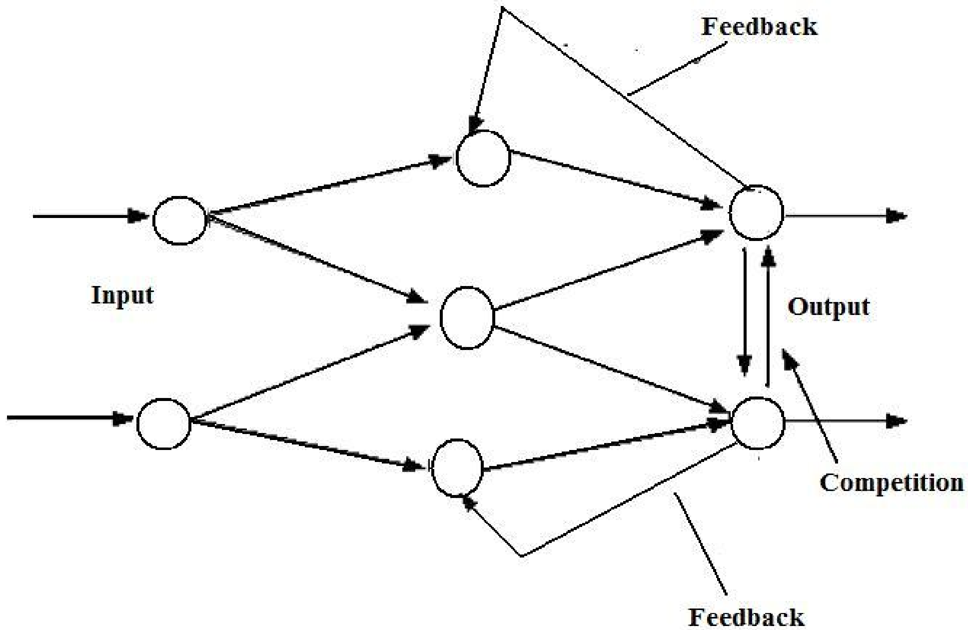

Figure 2.9: Feedback Neural Network

2.1.6 Learning in Neural Networks

The ability of neural network is to learn their environment and to adaptive finetune their parameter to improve the system performance. Generally, learning is the process by which the NN adapts itself to a stimulus, and eventually it produces a desired response. During the process of learning, the network adjusts its parameters, the synaptic weights, in response to an input stimulus so that its actual output response converges to the desired output response. When the actual output response is the same as the desired one, the network has completed the learning phase. Learning rules for networks are described by mathematical expressions called learning equations. The neurons in NNs may be interconnected in different ways; however, the learning process is not same for the all. It is known that, different learning methodologies suit different people. There are two general categories of learning in NNs, supervised and unsupervised learning.

- **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 NN 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

- **Unsupervised Learning:** Unlike supervised learning, there is no target output in unsupervised learning. During the training period, the network receives at its input many different input patterns and it arbitrarily organizes the pattern into categories. When a stimulus is later applied, the network provides an output response indicating the class to which the stimulus belongs. If a class cannot be found for the stimulus, a new class is generated. This type of learning sometimes referred to as self-organizing learning.

2.1.7 Learning Algorithms

A learning algorithm is a mathematical tool that outlines the methodology and the speed for NN to reach the steady state of its parameters, weights and thresholds successfully. It starts with an error function (energy function), which is expressed in terms of weights. The objective is to minimize the error in the set of weights. When the error function is zero or small enough, the steady state of the network and of the weights is reached. During learning, the error function decreases and the weights are updated. The decrease may be accomplished with different optimization techniques such as the Delta rule, Boltzman's algorithm, the backpropagation learning algorithm and simulation annealing. The selection of the error function and the optimization method is important, because it may increase stability, instability or a solution trapped in a local minimum. Generally we use backpropagation learning algorithm. Backpropagation learning algorithm is the basic learning mechanism and it is very popular. In this algorithm, the network output, on presentation of input data, is compared with the desired output and a measure of the error is obtained. This error measure is then used to incrementally modify appropriate weights in the connection matrices in order to reduce the error.

2.2 Fuzzy Set

A "fuzzy set" is a simple extension of the definition of a classical set in which the characteristic function is permitted to have any values between 0 and 1. A "fuzzy set" *A* in *X* can be defined as a set of ordered pairs:

$$A = \{(x, \mu_A(x) : x \in X)$$

Where $\mu_A(x)$ is called membership function for the fuzzy set *A*. It maps each *x* to a membership grade between 0 and 1. Examples of membership functions (Triangular, Trapezoidal and Gaussian) can be seen in Figure 2.10 and described with the following formulas:

- **Triangular MFs:** Triangular membership function is given by following equation:

$$Triangle(x; a, b, c) = \begin{cases} 0 & \text{if } x \le a \\ \frac{x-a}{b-a} & \text{if } a \le x \le b \\ \frac{c-x}{c-b} & \text{if } b \le x \le c \\ 0 & \text{if } x \ge c \end{cases}$$

- **Trapezoidal MFs:** Trapezoidal membership function is given by following equation:

$$Trapezoidal\ (x; a, b, c, d) = \begin{cases} 0 & \text{if } x \le a \\ \frac{x-a}{b-a} & \text{if } a \le x \le b \\ 1 & \text{if } b \le x \le x \\ \frac{d-x}{d-c} & \text{if } c \le x \le d \\ 0 & \text{if } x \ge d \end{cases}$$

- **Gaussian MFs:** Gaussian membership function is given by following equation:

$$Gaussian\ (x; c, \sigma) = \mathbf{e}^{-(x-c)^2/2}\, \sigma$$

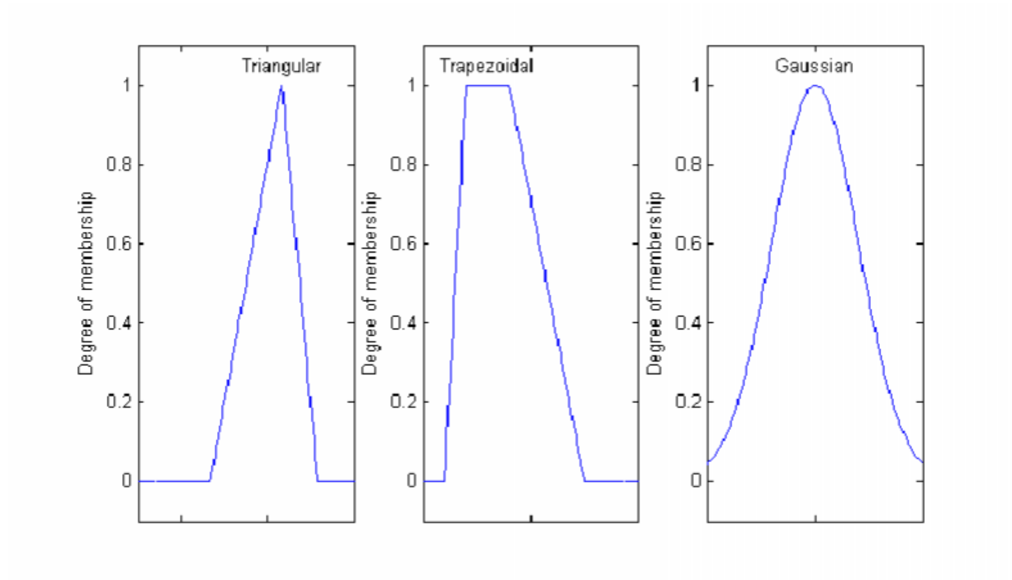

Figure 2.10: Examples of Membership Functions

2.2.1 Linguistic Variables

The concept of linguistic variables was introduced by Zadeh [32] to provide a basis for approximate reasoning. A linguistic variable was defined as a variable whose values are words or sentences. For instance, Age can be linguistic variable if its values are linguistic rather than numerical, i.e., young, very young, old, very old, etc., rather than 20, 21, 53, 55…. Figure 2.11 illustrates the term set Age expressed by the Gaussian MFs.

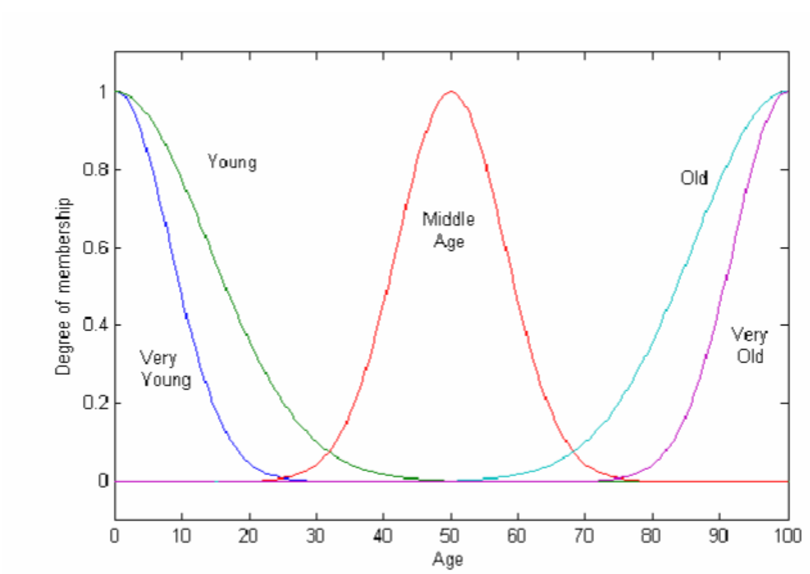

Figure 2.11: Membership Functions of the Term Set Age

2.2.2 Fuzzy if-then Rules

A fuzzy if-then rule (fuzzy rule, fuzzy implication, or fuzzy conditional statement) is expressed as follow:

If x is A then y is B

Where *A* and *B* are linguistic values defined by fuzzy sets. "x is *A*" is called "antecedent" or "premise", while "y is *B*" is called the "consequence" or "conclusion". Some of the if-then rule examples can be given below:

- If age is 25, then person is young.
- If the speed is low AND the distance is small, then the force on brake should be small.
- If height is 180 cm then person is tall

2.2.3 Fuzzy Reasoning

Fuzzy reasoning, approximate reasoning, is an inference procedure whose outcome is conclusion for a set of fuzzy if-then rules. The steps of fuzzy reasoning can be given as follows:

1. "Input variables are compared with the MFs on the premise part to obtain the membership values of each linguistic label (fuzzification).
2. The membership values on the premise part are combined through specific fuzzy set operations such as: min, max, or multiplication to get firing strength (weight) of each rule.
3. The qualified consequent (either fuzzy or crisp) is generated depends on the firing strength.
4. The qualified consequents are aggregated to produce crisp output according to the defined methods such as: centroid of area, bisector of area, mean of maximum, smallest of maximum and largest of maximum (deffuzification) [15].

2.2.4 Fuzzy Systems

Fuzzy systems are made of a knowledge base and reasoning mechanism called fuzzy inference engine. The structure of fuzzy inference engine is shown in Figure 2.12. A fuzzy inference engine combines fuzzy if-then rules into a mapping from the inputs of the system into its outputs, using fuzzy reasoning methods. That is, fuzzy systems represents nonlinear mapping accompanied by fuzzy if-then rules from the rule base.

Each of these rules describes the local mappings. The rule base can be constructed either from human expert or automatic generation that is extraction of rules using numerical input-output data [5].

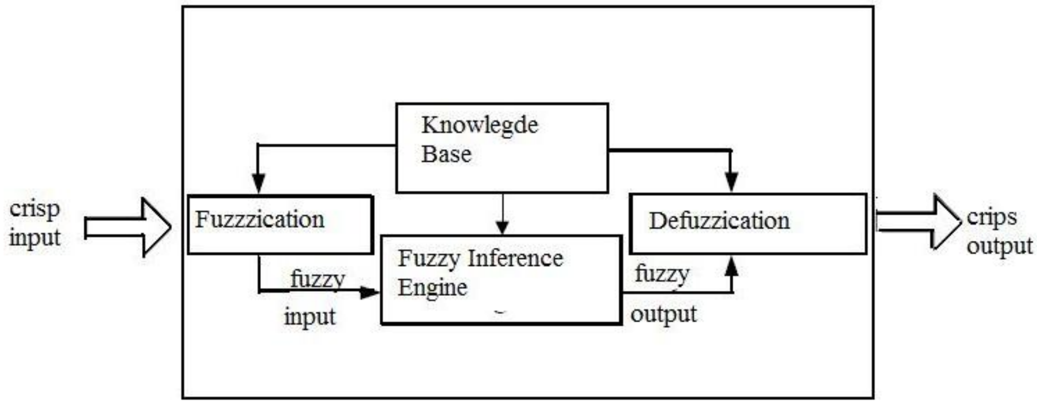

Figure 2.12: Fuzzy Inference Engine

A fuzzy inference system (FIS) consists of four functional blocks as shown in Figure 2.12:

- **Fuzzification:** transforms the crisp inputs into degrees of match with linguistic values.
- **Knowledge base:** consists of a rule base and a database. A rule base contains a number of fuzzy if-then rules. A database defines the MFs of the fuzzy sets used in the fuzzy rules.
- **Fuzzy inference engine:** performs the inference operations on the rules.
- **Defuzzification:** transforms the fuzzy results of the inference into a crisp output.

Mamdani and Takagi-Sugeno fuzzy systems are the examples of fuzzy inference systems. Mamdani fuzzy inference system was first used to control a steam engine and boiler combination by a set of linguistic rules obtained from human operators [17]. Figure 2.13 illustrates how a two rule Mamdani fuzzy inference system derives the overall output z when subjected to two numeric inputs x and y. Takagi-Sugeno fuzzy inference system was first introduced by Takagi and Sugeno [24]. The difference of Takagi-Sugeno model is that each rule has a crisp output, and the overall output is determined as weighted average of single rules output. This type of fuzzy inference system is shown in Figure 2.14

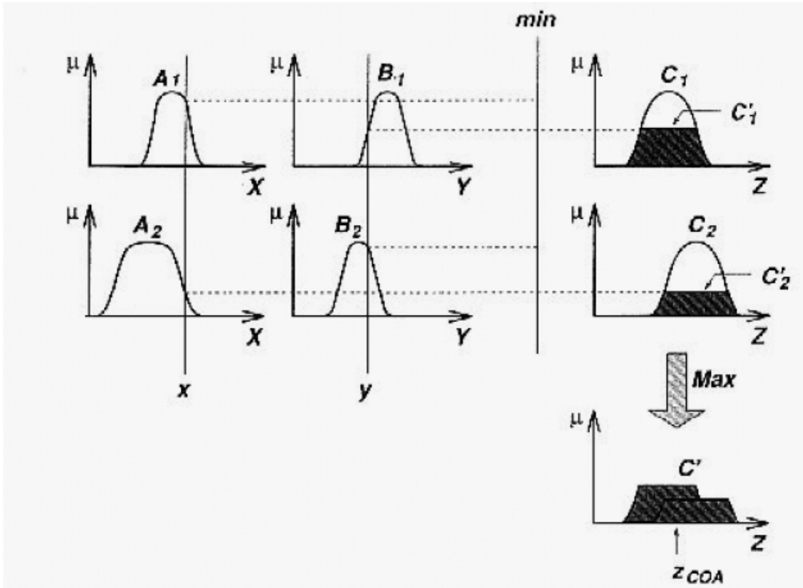

Figure 2.13: Mamdani Fuzzy Inference System

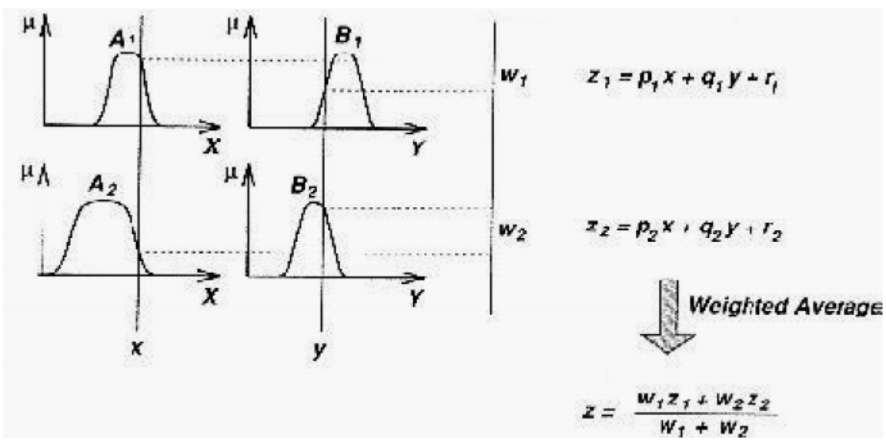

Figure 2.14: Takagi-Sugeno Fuzzy Inference System

2.3 Image Processing:

In electronic engineering and computer science, image processing is any form of signal processing for which the input is an image, such as photographs or frames of video the output of image processing can be either an image or a set of characteristics or parameters related to the image. Most image-processing techniques involve treating the image as a two-dimensional-signal and applying standard signal-processing techniques to it. Image processing is used to solve identification problems, such as in forensic medicine or in creating weather maps from satellite pictures. It deals with images in bitmapped graphics format that have been scanned in or captured with digital cameras.

2.3.1 Types of Digital Image

For photographic purposes, there are two important types of digital images: color and black & white. Color images are made up of colored pixels while black & white images are made of pixels in different shades of gray.

- **Black & White Images:** A black & white 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.

2.3.2 Image Processing Operation:

Image processing operation can be divided into 4 categories:

- **Image Enhancement:** Image Enhancement alters an image to makes it’s meaning clearer to human observers. It is often used to increase the contrast in images that are substantially dark or light. Enhancement algorithms often play attention to humans’ sensitivity to contrast.

Figure 2.15: Example of Image Enhancement

- **Image Segmentation:** The process of partitioning a digital image into multiple regions (set of pixel) is called image segmentation. Actually, partitions are different objects in image which have the same texture or calor. The result of image segmentation is a set of regions that collectively cover the entire image, or a set of contours extracted from the image. All of the pixels in a region are 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 characteristics. Some of practical applications of image segmentation are: image processing, computer vision, face recognition, medical imaging, digital libraries, image and video retrieval

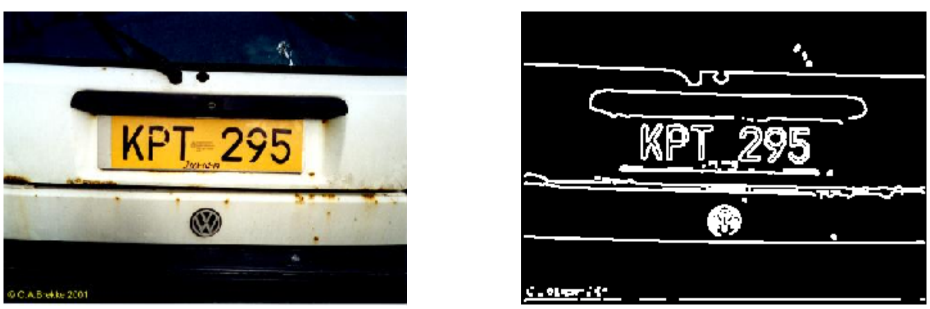

Figure 2.16 :Example of Image Segmentation

- **Image Compression:** Image compression means minimizing the size in bytes of graphics file without degrading the quality of image to an unacceptable level. The reduction in file size allows more images to store in a given amount of disk or memory space. It also reduces time required for image to be sent over internet or download from web.

Figure 2.17: Example of Image Compression

- **Image Restoration:** Restoration takes a corrupted image and attempts to recreate a clean image. As many sensors are subject to noise, they results in corrupted images that don't reflect the real world scene accurately and old photograph and film archives often show considerable damage

2.3.3 Feature Extraction

When the input data to an algorithm is too large to be processed and it is suspected to be notoriously redundant (much data, but not much information) then the input data will be transformed into a reduced representation set of features (also known as features vector). Transforming the input data into the set of features is called features extraction. If the features extracted are carefully chosen it is expected that the features set will extract the relevant information from the input data in order to perform the desired task using this reduced representation instead of the full size input.

2.3.4 Fuzzy Image Processing

Fuzzy image processing is not a unique theory. It is a collection of different fuzzy approaches to image processing. Fuzzy image processing is the collection of all approaches that understand, represent and process the images, their segments and features as fuzzy sets. The representation and processing depend on the selected fuzzy technique and on the problem to be solved [12].

Fuzzy image processing has three main stages: image fuzzification, modification of membership values, and, if necessary, image defuzzification (see Fig2.18.)

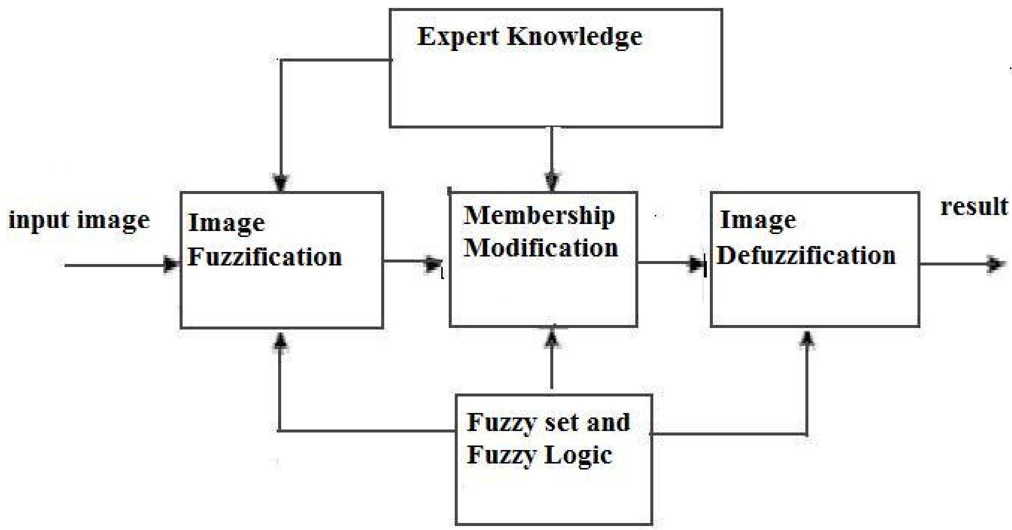

Figure 2.18**:** The General Structure of Fuzzy Image Processing.

The fuzzification and defuzzification steps are due to the fact that we do not possess fuzzy hardware. Therefore, the coding of image data (fuzzification) and decoding of the results (defuzzification) are steps that make possible to process images with fuzzy techniques. The main power of fuzzy image processing is in the middle step (modification of membership values, (Figure 2.19). After the image data are transformed from gray-level plane to the membership plane (fuzzification), appropriate fuzzy techniques modify the membership values. This can be a fuzzy clustering, a fuzzy rule-based approach, and a fuzzy integration approach and so on [12].

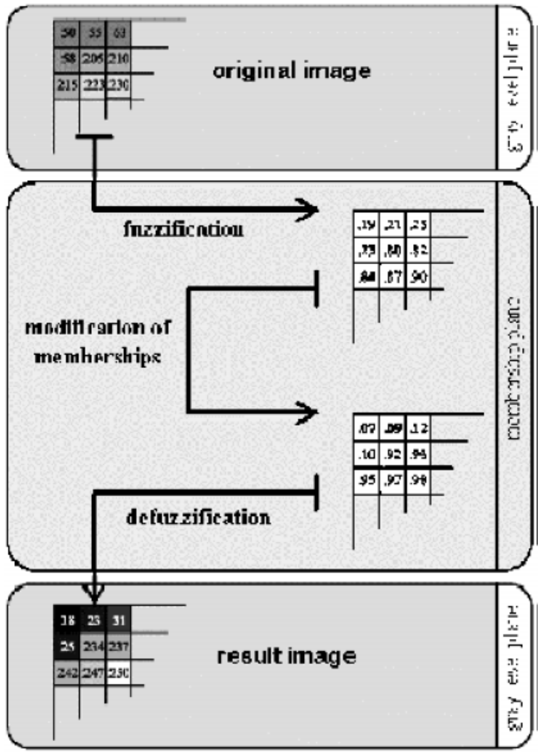

Figure 2.19: Steps of Fuzzy Image Processing

2.3.5 Motivation behind Fuzzy Image Processing

There are many reasons to use fuzzy logic in image processing. Some of important regions are as follows:

- Fuzzy techniques are powerful tools for knowledge representation and processing
- Fuzzy techniques can manage the vagueness and ambiguity efficiently
- In many image processing applications, we have to use expert knowledge to overcome the difficulties (e.g. object recognition, scene analysis). Fuzzy set theory and fuzzy logic offer us powerful tools to represent and process human knowledge in form of fuzzy if-then rules. On the other side, many difficulties in image processing arise because the data/ tasks/results are uncertain. This uncertainty, however, is not always due to the randomness but to the ambiguity and vagueness. Beside randomness which can be managed by probability theory we can distinguish between three other kinds of imperfection in the image processing.

i. Grayness ambiguity
ii. Geometrical fuzziness
iii. Vague (complex/ill-defined) knowledge

- General observations about fuzzy logic are:
  - Fuzzy logic is conceptually easy to understand. The mathematical concepts behind fuzzy reasoning are very simple.
  - Fuzzy logic is flexible. With any given system, it's easy to manage it or layer more functionality on top of it without starting again from scratch.
  - Fuzzy logic is tolerant of imprecise data. Everything is imprecise if we look closely enough, but more than that, most things are imprecise even on careful inspection. Fuzzy reasoning builds this understanding into the process rather than tacking it onto the end.
  - Fuzzy logic can model nonlinear functions of arbitrary complexity. We can create a fuzzy system to match any set of input-output data. This process is made particularly easy by adaptive techniques like ANFIS (Adaptive Neuro-Fuzzy Inference Systems), which are available in the Fuzzy Logic Toolbox.
  - Fuzzy logic can be blended with conventional control techniques. Fuzzy systems don't necessarily replace conventional control methods. In many cases fuzzy systems augment them and simplify their implementation.
  - Fuzzy logic is based on natural language. The basis for fuzzy logic is the basis for human communication. This observation underpins many of the other statements about fuzzy logic.

Chapter 3
Literature Review

There is a lot of work has been done in the field of fingerprint minutiae point detection. In this section, work done in the area of fingerprint and neural fuzzy set is reviewed and focus has been made on detecting the minutiae point of fingerprint.

3.1 Survey on Fuzzy Set:

The concept of fuzziness was first proposed by Zadeh (1965). He aimed to describe complex and complicated systems using fuzzy approximation and introduced fuzzy sets. “Generally, fuzzy logic can be considered as a logical system that provides a model for modes of human reasoning that are approximations rather than exact”. Table 3.1 shows brief history of fuzzy set and logic.

Fuzzy Logic (FL) is a powerful problem solving methodology in the past few years. It shows a rapid growth in the applications of FL, especially in the image understanding applications, such as edges detection, feature extraction, classification and clustering. FL provides a simple and easy way to draw a definite conclusion from ambiguous, imprecise or vague information. The adaptation of FL is mimicking the human decision making, which provide a precise solution from approximate data. With FL, we can apply rule based decision making in terms of words rather than numeric figure. It is more similar with the way of human thinking which generate solution from the expertise experiences (represent in rules), knowledge and even skill. Although we know that the rule-based systems have a long history in the field of Artificial Intelligence (AI), but the conventional rule-based system have difficulty to tackle the fuzzy consequents and fuzzy antecedents' problems. The if-then rules in FL will combine with an inference engine to be more flexible and accurately to solute the problems mentioned above [21].

From the official web page of fuzzyTECH [9], we find that the FL technology has very good results in two main application fields, which consists of industrial applications and business and finance applications. Those FL applications can categorize under automated control or decision-making support. The common used of

FL in automated control industrial applications is due to three main factors: complex design of control systems and involve multiple parameters the optimization of most system based on engineering expertise rather than mathematical methods, and the competitive automotive engineering on an international scale. In another hand, FL has been widely used in risk assessment and can be used to help investors evaluate data in the field of business and finance applications. Just like automated engineering~ in market investment, a manager may use his knowledge, which consists of a lot of rules~ and also his experiences and skill to analyze the investment situation. FL will provide a mechanism to users especially when rules and experiences are important in problem solving [21].

Other currently available applications of FL in automated control consists of automatic control of dam gates for hydroelectric-power plants, wind energy converter control, camera aiming for the telecast of sporting events, efficient and stable control of car-engines, cruise-control for automobiles, positioning of wafer-steppers in the production of semiconductors, back light control for camcorders, automatic motor control for vacuum cleaners with recognition of surface condition and degree of soiling, single button control for washing-machines, flight aid for helicopters, software-design for industrial processes, controlling of machinery speed and temperature for steel-works, controlling of subway systems in order to improve driving comfort, precision of halting and power economy, improved fuel-consumption for automobiles, improved sensitiveness and efficiency for elevator control, and improved safety for nuclear reactors [3] [21].

1965 Zadeh	Introduction to fuzzy set
1966 Zadeh	Pattern recognition as interpolation of membership function
1969 Ruspini	Concept of fuzzy portioning
1970 Prewitt	First approach towards fuzzy image understanding
1973 Dunn, Bezdek	First fuzzy cluster algorithm

1977 Pal	Fuzzy approach to speech recognition
1979 Rosenfeld	Fuzzy geometry
End of 1980s-90s Dave/Krishnapuram/ Bezdek	Different fuzzy cluster algorithm
Russo/ Krishnapuram	Rule based filter
Bloch et al. / Di Gesu / Sinha et al./De Baets /and many others...	Fuzzy morphology

Table 3.1: History of Fuzzy Set [12]

3.2 Survey on Available Neural Networks Applications

According to [6], neural network models are usually implemented in a few areas such as computer vision, speech recognition, signal analysis, robotics, expert systems and scheduling. As more improvement are made to neural networks models architecture and learning algorithms, neural networks models will be able to be implemented into more areas. Generally neural networks model are used in data mining, matching and clustering.

Below are the examples of applications that use neural network models from [6].

- **Computer Vision**: One of the neural network applications is character recognition, The Nestor Learning System claim that their recognizer has the ability to recognize approximately 2,500 Japanese handwritten characters with 92% accuracy, recognize handwritten zip code with 98% accuracy and verify signature recognition with 4% false reject rate. Besides implemented in character recognition, neural network was also implemented in face recognition system, image compression, object recognition, edge detection, data classification and biometric recognition.
- **Speech Recognition**: Neural network were implemented in speech recognition for several purposes such as to convert text to speech and speech to text application, and also use speech as a biometric. The Phonetic Typewriter and Net talk are two of a few successful applications that able to convert text to

speech (aid the keyboard to enter words) and speech to text (to enhance the accessibility of computers). A speech recognition system will automatically extracts speeches and store in the memory and to be used again for matching. This speech recognition process can be implemented with Recurrent Neural Networks (RNN), Radial Basis Functions (RBF) and Vector Learning Quantization (LVQ).

- **Signal Analysis**: Signal analysis is one of the largest neural network models research areas. Neural network is used because radar is required to perform tracking, recognizing, and classifying on an object by analysing signal that receive by a receiver. Besides that, neural network are used in radar technology because a radar system must handle a lot data such as image angles and targets which required long computational time by conventional algorithms and neural network are able to solve the problems with its generalization and parallel processing characteristic. Helicopter recognition for smart weapons, radar target tracking, classification and recognition, and sonar classifier are the examples of radar that use neural network technology.
- **Robotic**: A robotic movement can be divided in to Autonomous Vehicles and Manipulator Trajectory Control. The Autonomous Vehicles work as the robot decision module, it make decision of the robot movement based to the input provided by sensors. Manipulator Trajectory Control is used to control the robot's kinematics, to design a manipulator control is difficult and time consuming with conventional programming. The adaptability and generalization characteristic of a neural network is able to solve the problem.
- **Expert System:** The different between a neural expert system with other expert system is that the neural expert system does not require a knowledge engineer to formulate rules. The neural expert system is applied in medical services and financial services. One of the example of medical services expert system is the Saito's Medical Diagnostic Expert System which able to diagnose 23 diseases from 216 symptoms with the 67% accuracy after 300 examples training. Nestor's Mortgage Origination Underwriter is one of the financial services expert system is a system that determines a mortgage loan application based upon the applicant's information.

3.3 Survey on Fuzzy Set and Neural Network in Fingerprint Recognition

In this section detail of work has been done on fuzzy set and neural network in the field of fingerprint recognition is given:

In 1999 *Vijay Kumar Sagar and Koh Jit Beng Alex* used neuro-fuzzy set technology in automated fingerprint recognition for minutiae point extraction. In contrast to the classical approach, the fuzzy approach makes use of the grayscale information for the extraction of minutiae. A grayscale fingerprint image consists of two distinct levels of gray pixels. The darker pixels, constituting the ridges, form one such level. The lighter pixels, constituting the valleys or furrows, form one other such level. Using human linguistics, these two levels of gray can be described as dark and bright levels correspondingly. By using fuzzy logic, these levels can be modelled and used along with the appropriate fuzzy rules to extract minutiae accurately. The fuzzy neural approach is essentially an extension of the fuzzy approach. The surrounding areas of a potential minutia can be considered as a pattern of dark and bright windows. Neural networks can then be used to recognize these patterns, and therefore, model the fuzzy rules of the system [30].

In 2001 *Tu Van Le, Ka Yeung Cheung and Minh Ha Nguyen*, developed a fingerprint recognise system that was based on fuzzy evolutionary programming. A fingerprint recognizing system is built with two principal components: the fingerprint administrator and the fingerprint recognizer. Fingerprints are identified by their special features such as ridge endings, ridge bifurcation, short ridges, and ridge enclosures, which are collectively called the minutiae. The fingerprint administrator uses the method of gray scale ridge tracing backed up by a validating procedure to extract the minutiae of fingerprints. The fingerprint recognizer employs the technique of fuzzy evolutionary programming to match the minutiae of an input fingerprint with those from a database [29].

In 2003 *Vijayaprasad.PI, Ashraf Gasim and Elsid Abdalla*, proposed a new algorithm to improve fingerprint image quality by using Neuro-fuzzy technique. It was very difficult to detect minutiae point from bad quality fingerprint image. In this research a new method has been developed that was based on neural fuzzy set algorithm to enhancement the image. First give the membership function to each pixel according to its gray level and find image quality. After that a new membership function has been developed to enhance the fingerprint image and train the new image by neural network. In last by compare with many other image enhancement methods the result as neural fuzzy based technique gives batter result then others [31].

In 2005 *Benno Stain* introduced a particular form of fuzzy-fingerprints-their construction, their interpretation, and their use in the field of information retrieval. Though the concept of fingerprinting in general is not new, the way of using them within a similarity search as described here is: Instead of computing the similarity between two fingerprints in order to access the similarity between the associated objects, simply the event of a fingerprint collision is used for a similarity assessment. He has given a new approach by adding fuzzy hash function to compute fuzzy fingerprint for a given document to find similarity. The main impact of this approach is the small number of comparisons necessary to conduct a similarity search [4].

Later in 2005 *G.Vert and S.Nasser* used fuzzy logic and graphics point of reference for verifies to minutiae points. Fuzzy set theory used to verify fingerprint images in a database match with the scanned fingerprint of a user attempting to log into system. The idea behind the research that there may be different factor that affect efficiency and reliability of fingerprint verification system such as there may be different template of same fingerprint if it scan from different points. Because two templates obtained from the same fingerprint often can be difficult to match identically, so rules base on fuzzy set theory have been developed to perform matching on imprecise fingerprint templates. These rules allow for certain degree of error in the matching process [10].

In 2006 *Roshini Velamuri* submitted a report in University of Texas El Paso on the topic of Fingerprint recognition using Fuzzy inference techniques. She develops the fingerprint recognition system using fuzzy inference techniques. The two principle components in this recognition system are the administrator, where templates are stored and the fingerprint recognizer. Here do not use the intermittent steps of Image processing techniques like enhancement the image and reduction of the noise. This technique based on extracts true minutiae point by fuzzy rules and then matching fingerprint by these minutiae points. In comparison to other recognition systems designed, this approach has an advantage that is the proposed system is cost effective [22].

In 2007 *A. Montesanto, P. Baldassarri, G. Vallesi, G. Tascini*, given a fuzzy approach for fingerprint recognition by minutiae point extraction. He proposed the fingerprint verification method based on local ridge discontinuities features (minutiae) only using grey scale images. He extracts minutiae using two algorithms those following ridge lines and then recording ridge endings and bifurcations. His approach is based on the fuzzy logic and combines the results obtained using three different methods of minutiae extraction: the sequential method, the reactive agent and the neural classification system. These methods does not guarantee the same performances in the minutiae extraction phase, producing sometimes too many false minutiae due to the noisy images considered in the experimental phase. The sequential method run after the ridge line on the grey scale image of the fingerprint until it does not meet a ridge ending or a point of intersection with another ridge line. In order to establish if the intersection identifies a bifurcation, a fixed threshold is used. During the experiment this method is able to correctly recognize the ridge endings, unlike the bifurcations. In the second method the fingerprint image is very similar to a maze, in which a reactive agent moves avoiding the ridge line that identifies the walls. This approach, using an association between the sensorial profiles and the path is the most efficient in the minutiae extraction. Moreover this method is an interesting and a novel application for this specific problem. Finally the neural classification system consists of a MLP based on a supervised learning that elaborates a grey scale normalized matrix representative of the fingerprint image. The idea of this approach was to resort to the fuzzy logic for joining the results of the three different methods, in order to improve the identification percentage of the fingerprint. For this purpose the minutiae are identified not only by the (x,y) coordinates and the orientation but also by the level of belonging to the OR Fuzzy set produced by the three method [1].

Later In 2007 *Dr.Rosalina Abdul Salam*, presented a project on fingerprint recognition system using neuro-fuzzy set. This system was based on neuro-fuzzy clustering. He proposed that Clustering of fingerprints can help to reduce the complexity of the search process in a database. This can be done by grouping fingerprints with the same characteristic in the same group. The matching algorithm can compare stored fingerprint codes with only one cluster instead of the entire database. In his research, he classified fingerprints into five categories which are arch, left loop, right loop, whorl, and others. The last category is use to categorize fingerprint pattern other then the four categories. Finally, experiments were carried out to show that clustering can reduce the recognition time. Experiments were carried out using neural network classifier, fuzzy logic and Neuro-fuzzy and showed that neural network classifier is the best among the three [21].

Chapter 4
Problem Statement and Methodology

4.1 Problem Statement

In today's world minutia matching is most popular and modern technology for fingerprint matching. If there is enough minutia point in one fingerprint image that are corresponding to other fingerprint image then it is most likely that both image are from same fingerprint. Minutiae are usually matched together by their distance relative to other minutiae around it. If multiple points in one image have similar distances between the multiple points in another image then the points are said to match up. But it is not an easy task to extract minutiae point in fingerprint. Also there may be some false minutiae point that makes the problem more difficult.

Image quality is another problem to find the location and type of minutiae point in fingerprint. The quality of the input fingerprint image affects the performance and fingerprint matching methods. The quality of a fingerprint image measured corresponds to the clarity of the ridge structure in the fingerprint image. A fingerprint that contain high contrast and well defined ridges and valleys, are called as good quality image while a poor quality fingerprint is marked by low contrast and ill-defined boundaries between the ridges. There are several reasons that may degrade the quality of a fingerprint image.

- There may be creases, bruises or wounds present in fingerprint image that may cause of ridge discontinuities.
- Excessively dry fingers lead to fragmented and low contrast ridges.
- Sweat on fingerprints leads to smudge marks and connects parallel ridges.

The main objective of the thesis work is to develop a new idea to extract minutiae point and there location by using fuzzy set and neural network. The objective of the thesis work contains the following steps as described below:

- First understand the basics concept of fingerprint and minutiae point.
- Enhance the poor quality image to good quality image by increase the contrast using the method based on fuzzy logic.

- Find false minutiae point and delete them by making some fuzzy rule.
- Find true minutiae point and there location and then give training by using neural network.

4.2 Methodology

The step-by-step methodology to be followed for finding minutiae point and there location using fuzzy theory and neural network are as follows:

- Different step in fingerprint minutiae point extraction (such as image enhancement, image binarisation, image thinning etc) have be done.
- Based upon above analysis a program is developed for extracting minutiae point in C programming, by using MATLAB 7.0 version.
- Results achieved after the execution of program are compared with the earlier outputs.

Chapter 5
Implementation and Result

5.1 Implementation

This chapter include detail of the process to extract minutiae point in fingerprint and their location. All the work has been implemented in MATLAB 7.0. MATLAB is a software package developed by Maths Work. It integrates computation, visualisation and programming in easy to use environment. MATLAB toolbox allows learning and applying specialized technology. Areas in which toolboxes are available include signal processing, image processing, control systems, neural networks, fuzzy logic, wavelets, simulation, and many others.

The process to extract fingerprint minutiae points contains five steps. First step is image acquisition step to obtained fingerprint image from different sensor. The second step is to enhance the contrast of image by using fuzzy logic. Image binarisation step is done after enhancement of image. Fourth step is to convert the binary image into thinned image and then minutiae point extraction method has been implemented in step five. Neural network training are describes in step six.

Step 1 Image Acquisition

The first step is to acquire fingerprint image. The fingerprint image acquire either by offline or by an online process. The fingerprint acquired by online is called “live-scan” whereas offline fingerprint image are known as “inked” fingerprint. Inked fingerprint are of three types: rolled, dab and latent. In the rolled method of fingerprint acquisition, ink is applied to the finger and then rolled on a paper from one side of the nail to the other to form an impression. This paper is then scanned at 500 *dpi* resolution by a standard grayscale. In the Dab method ink is applied on fingerprint and then presses it onto paper. In the online process there is no need of paper. In this process fingerprint image directly obtain from person. In this thesis offline images are used

First different types of fingerprint image are obtained from different sensor and saved in work space. To use of these image, the command and description are describes below

- **Command**: *>>imread ('finger1.bmp');*
- **Description:** first save the acquire fingerprint image with a name as 'finger1' in workspace and then call this image using function 'imread'

Step 2 Image Enhancements by Fuzzy Set

Fingerprint Image enhancement is to make the image clearer to the further operations. Since the fingerprint images that have been acquired from sensors or other media are not assured with perfect quality, so any enhancement method, for increasing the contrast between ridges and valleys and for connecting the false broken points of ridges due to insufficient amount of ink, is very useful to keep a higher accuracy to fingerprint recognition. For this purpose, fuzzy logic is used to enhance the fingerprint image by increase the contrast between ridge and valleys. Fuzzy logic is not just new method for image enhancement. Many researches have been done on the fuzzy logic and image enhancement [8] [16] [19] [31]. Some of fuzzy rule for image enhancement are such as

- If pixel level is dark then output is darker.
- If pixel level is gray then output is gray.
- If pixel level is bright then output is brighter.

Figure 5.1 and Figure 5.2 shows implementation of these rules in fuzzy inference system:-

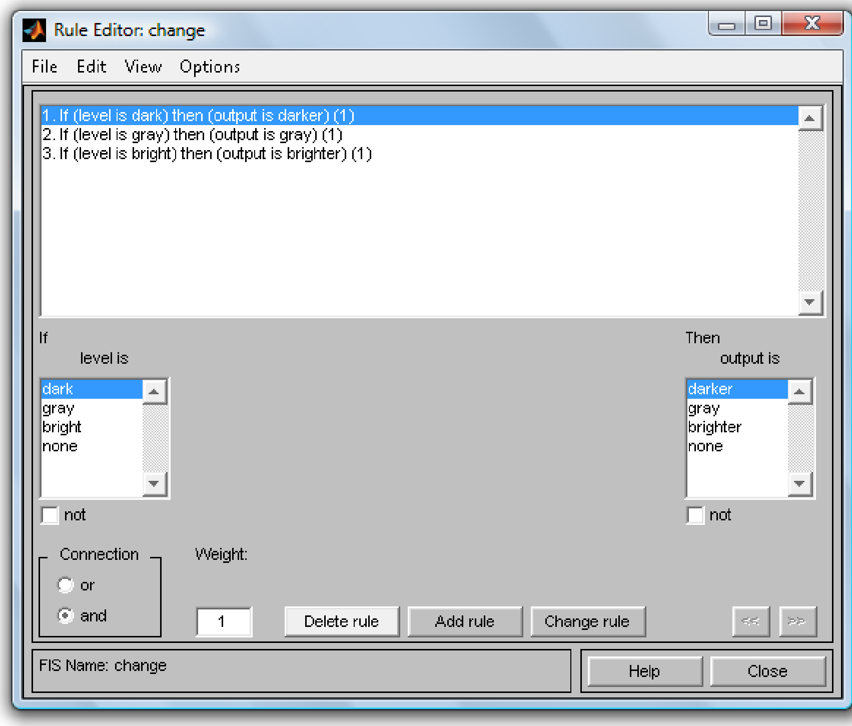

Figure 5.1 FIS rule for image enhancement

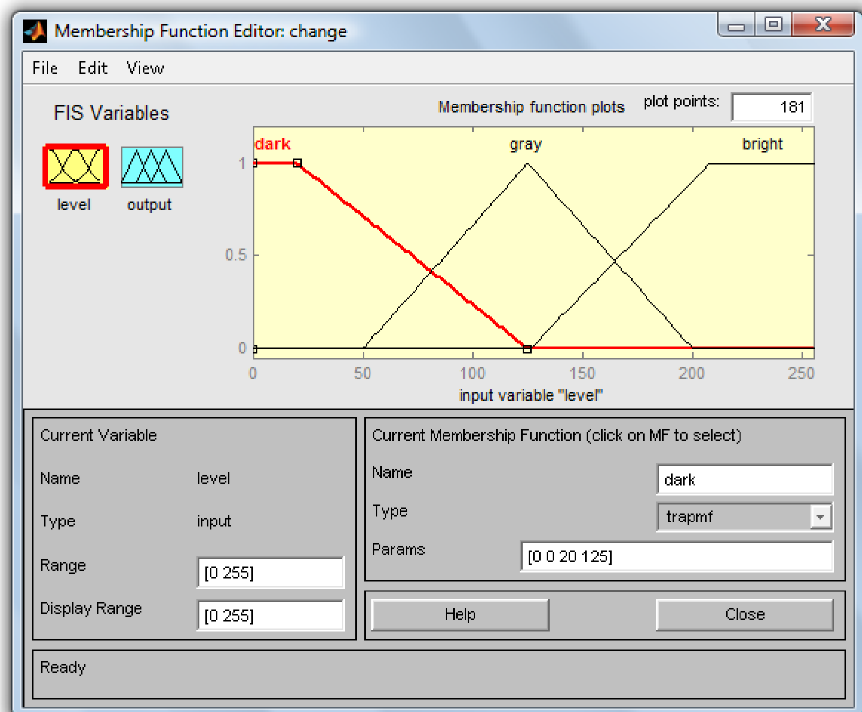

Figure 5.2: FIS Membership Function

On the basis of these rule, a new algorithm based on fuzzy methods on image enhancement have been implemented which have following steps:

- Taking image and set membership function for each gray level as :

$$\mu(x, y) = e^{\frac{-\left(\frac{L-f(x,y)}{s}\right)^2}{2}}$$

  Where L=max gray level

  f (x, y)= any gray level.

  s =variance between gray values

- Get new membership function by putting:

$$v(x,y) = 2*(\mu(x,y))^2 \text{ if } \mu(x, y) <= 0.5$$
$$v(x,y) = 1-2*(1-\mu(x,y))^2 \text{ if } 0.5 <= \mu(x, y) <= 1$$

- Get new image by setting the method as:

$$g(x,y) = L-s\left(\sqrt{-2\log(v(x,y)}\right)$$

To implement this algorithm the command and description are describes as below:

- **Command**: *>>fuzzyenhance*
- **Description:** developed a program based on above algorithm and save as by name "fuzzyenhance" in a M-file and then use command "fuzzyenhance" or press *F5* to run this program. Figure 5.3 shows the result of enhance fingerprint image :

Figure5.3 shows the enhancement result by fuzzy logic

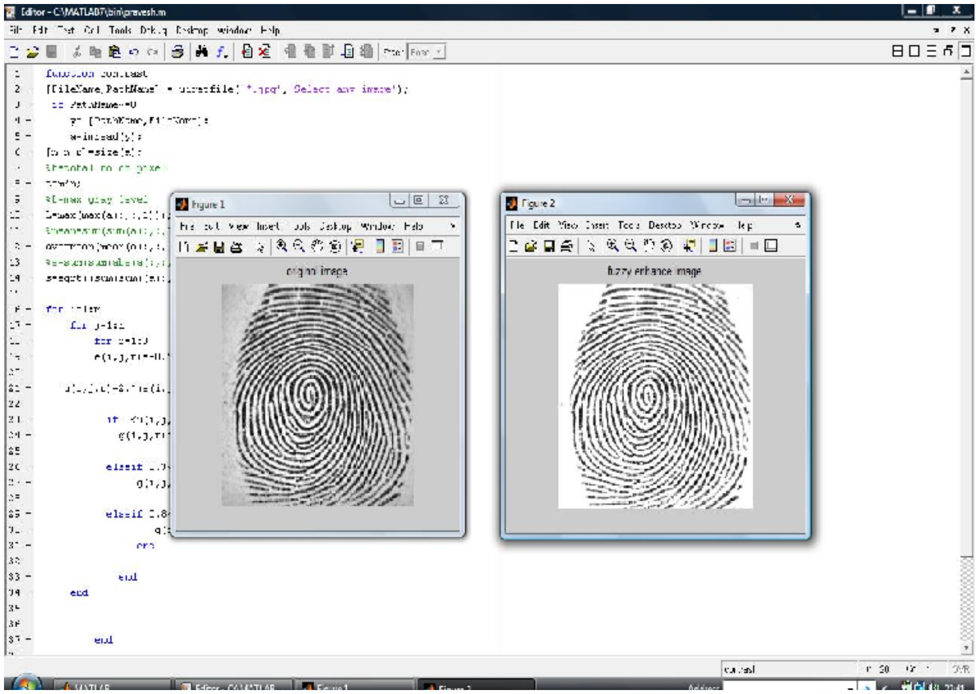

Figure5.3: Fuzzy Image Enhancement

Step 3 Image Binarisation:

Binarisation is the process to convert gray image into binary image. Most minutiae extraction algorithm operates on binary image in which there are only two level of interest 0 (for black level) and 1 (for white level). After the operation, ridges in the fingerprint are highlighted with black colour while valleys are white. The binarisation 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.

Command and description are describes as below:

- **Command:** >>*im2bw('finger1.jpeg', graythresh(finger1))***)**
- **Description:** This command will convert the image into binary image as shown in figure 5.4.

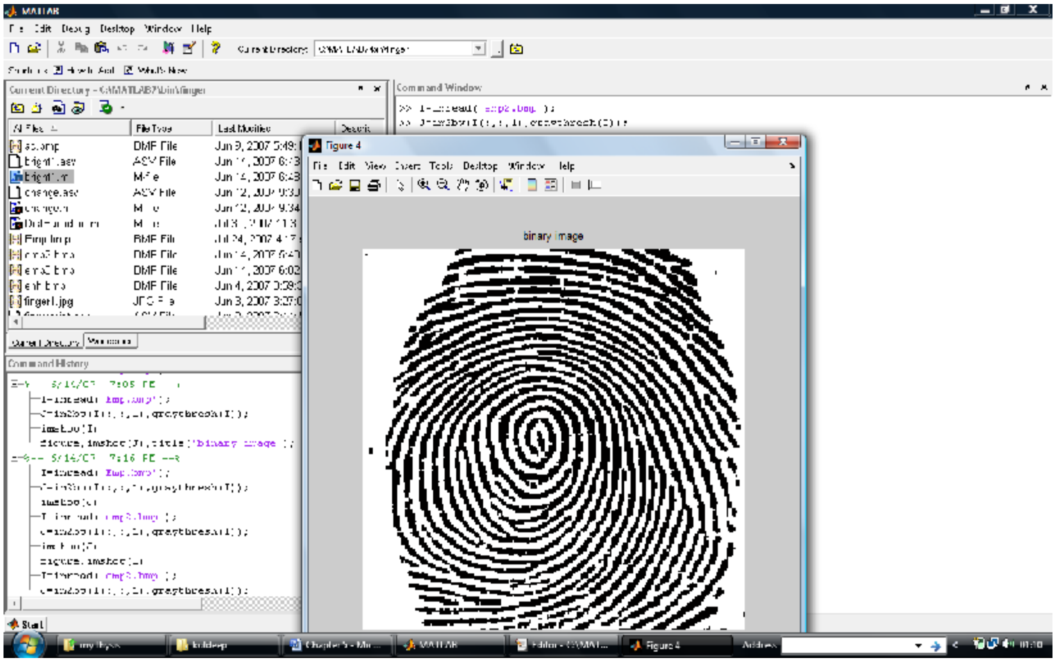

Figure 5.4: Binary Image

Step 4 Image Thinning

Ridge thinning is to eliminate the redundant pixels of ridges till the ridges are just one pixel wide. Thinning is the morphological process to remove the foreground pixel until they are one pixel wide. So in the first step morphological process apply to reduce the width of the ridge. Morphological is a means of changing a stem to adjust its meaning to fit its syntactic and communicational context. Two main morphological processes are:

- **Erosion:** erosion use to thin object in binary image.
- **Dilation:** dilation thins is use to thin the area in valleys in the fingerprint.

A thinning algorithm is given by [11] in which the thinning operation using two iteration. Each sub-iteration begins by examining the neighbourhood 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. By MATLAB it is easy to accessible the operation by *thin* operation under function *bwmorph* . Command and description are describes as below:

- **Command***: >> bwmorph(binaryfinger1,'thin')*

- **Description:** after converting finger1 image into binary image, save as it by name *binaryfinger1*. Then this command convert this binary image to the thin image as shown result in figure 5.5

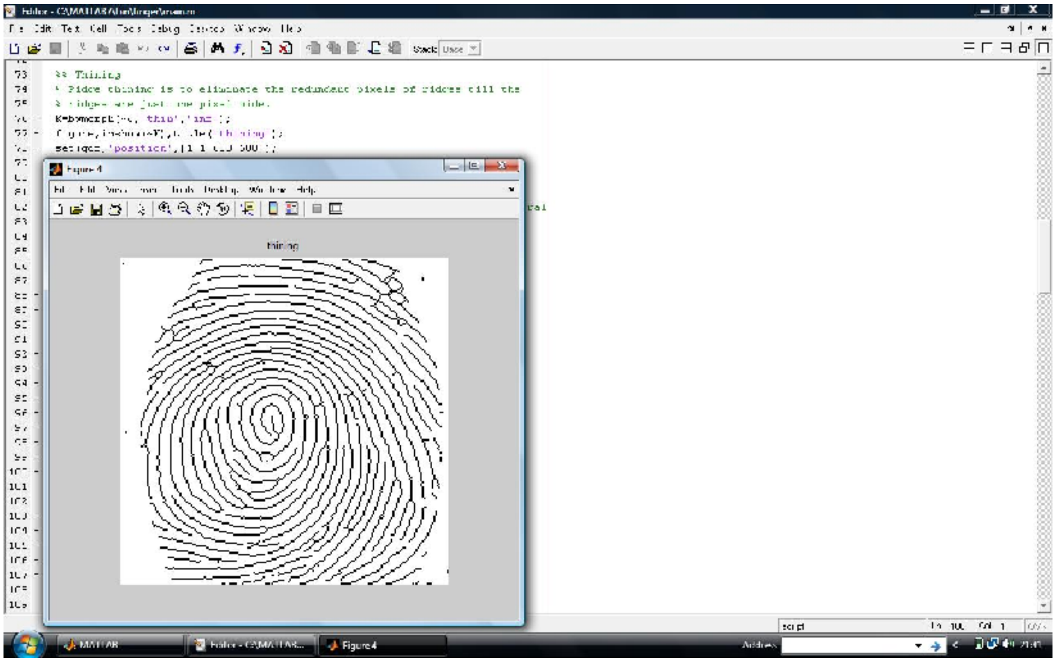

Figure5.5: Image Thinning

Step 5 Minutiae Points Extraction

Extraction minutiae point and their location is an important step of the process. There are many algorithms have been developed for minutiae point extraction. A algorithm based on Tico and Kuosmanen method [27] and Crossing Number (CN) methods is used for extract minutia points. This method extracts the ridge endings and bifurcations from the skeleton image by examining the local neighbourhood of each ridge pixel using a 3x3 window. The CN for a ridge pixel *P* is given by [25]:

$$CN=0.5\sum_{i=1}^{8}|P_i - P_{i+1}|\ , \quad P_9 = P_1$$

Where *Pi* is the pixel value in the neighbourhood of *P*. For a pixel *P*, its eight neighbouring pixels are scanned in an anti-clockwise direction as follows: table 5.1 shows the neighbourhood pixel of *P* in anti-clock direction.

P_1	P_3	P_2
P_5	P	P_1
P_6	P_7	P_8

Table5.1: Neighbourhood Pixel of *P* in Anti-Clock Direction

After the CN for a ridge pixel has been computed, the pixel can then be classified according to the property of its CN value.

- If value of CN is one, then the central pixel is termination.
- If value of CN is two then the central pixel is usual pixel.
- If value of CN is three then the central pixel is buffrication.

Figure 5.6 shows the CN values according to ridge ending and buffrication.

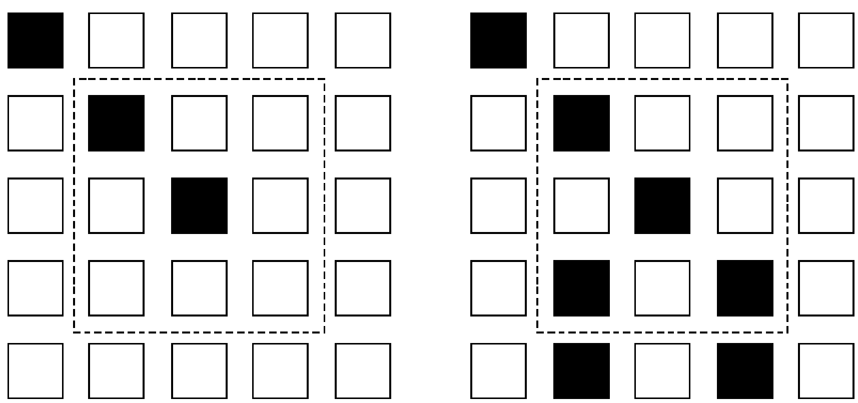

Figure5.6: CN=1 and CN=3

To find termination, first divided image into a image M of size WxW, then label 1 to all pixels in image M (3x3 image) which are eight connected with the termination point. After this step we count in clockwise direction, the number of 0 to 1 transition along to border. If the value of transition is equal to 1 then this minutia point will be consider as termination.

To find buffrication first we examine the eight neighbourhood pixels surrounding the buffrication point in clockwise direction. Now label 1, 2 and 3 to the pixels that are connected to buffrication point. After that, label to rest of ridge pixel that are connected to these three pixels, this labelling is similar to termination labelling. After labelling we count in clockwise the no of 0 and 1, 0 to 2 and 0 to 3 transition along to the border of image. If the value of transition is equal to 1 then minutia point will be consider as buffrication point. Figure5.7 shows these approaches by an image:-

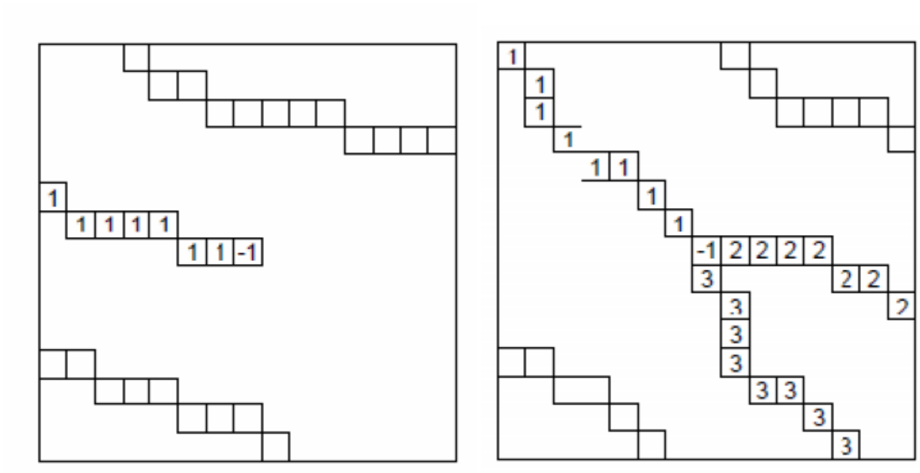

Figure5.7: Termination and Buffrication

In order to find spurious point a new algorithm chooses that is based on some fuzzy rule. This algorithm tests the validity of each minutiae point in thinned image and examines the local neighbourhood around the point. The first step in this algorithm is to find the distance between termination and buffrication. We have use Euclidian method for find distance. After finding distance make some rule for remove these false minutia points. Some of the rules are;-

- If the distance between termination and buffrication is less then D, then remove this minutia.
- If the distance between two buffrication is less then D, then remove this minutia.
- If the distance between two terminations is less then D, then remove this minutia.

After extraction of minutia, find location of these minutiae points. For finding minutiae points' location, the following three points have been obtained:-

- X and Y coordinates
- Orientation angle between these coordinates
- Type of minutiae (ridge ending or buffrication)

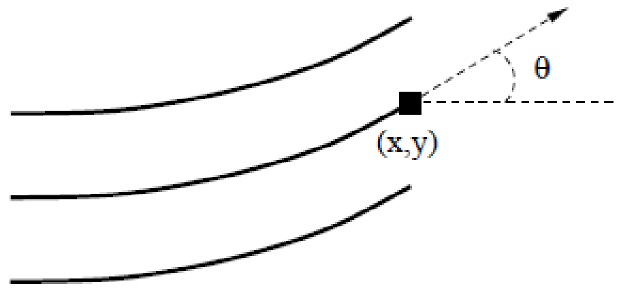

Figure5.8: Orientation of Ridge Pixel

Command and description for extracting minutiae points are given below:

- **Command:** >> main
- **Description:** based on the above algorithm, a program is written and saved as by using name "main" in M-file. To run this program give the command as "main" or press F5. The result is shown in figure 5.9 where Green points represent ridge buffrication while red represent ridge ending.

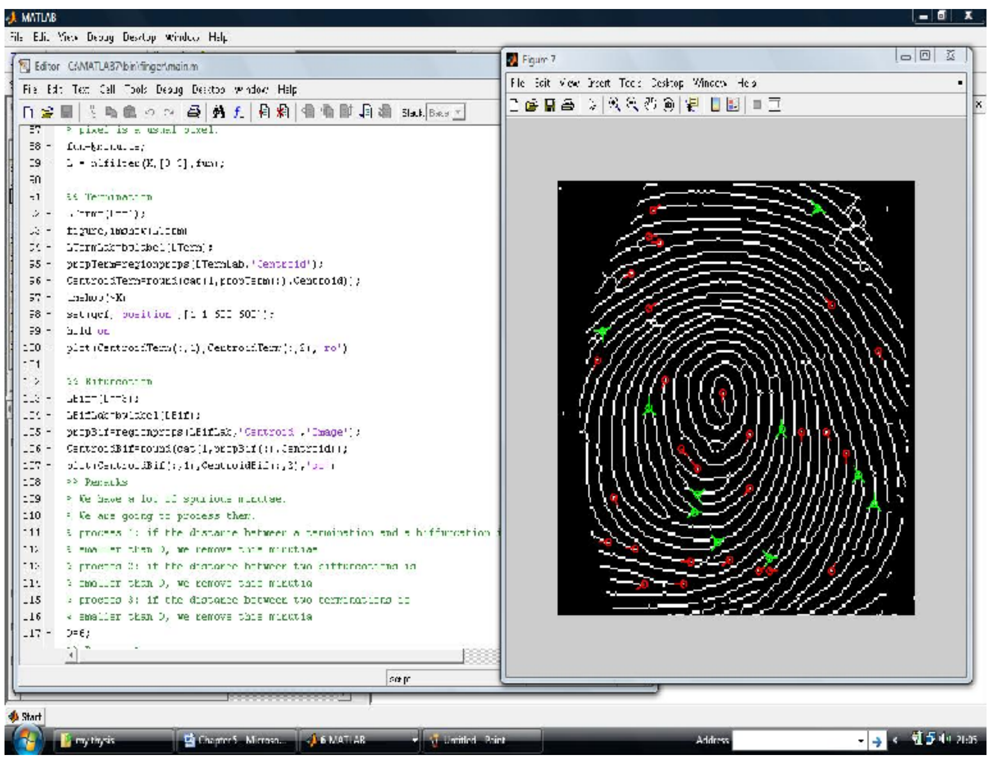

Figure5.9: Minutia Point Extraction

Step 6 Training

The last step is to train minutiae point location by neural network using backpropagation algorithm. All detail about neural network has been given in chapter 2. Backpropagation is most popular supervisor learning algorithm used to training in various systems [2] [5] [7] [13]. With backpropagation, 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 (backpropagated) 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. Figure 5.10 shows that after 80 epochs, the performance is 0.00979445 which increase up to 0.00996026 after taking 170 epochs (Figure5.11). Figure 5.12 show the performance is 1.29503 after 500 epochs.

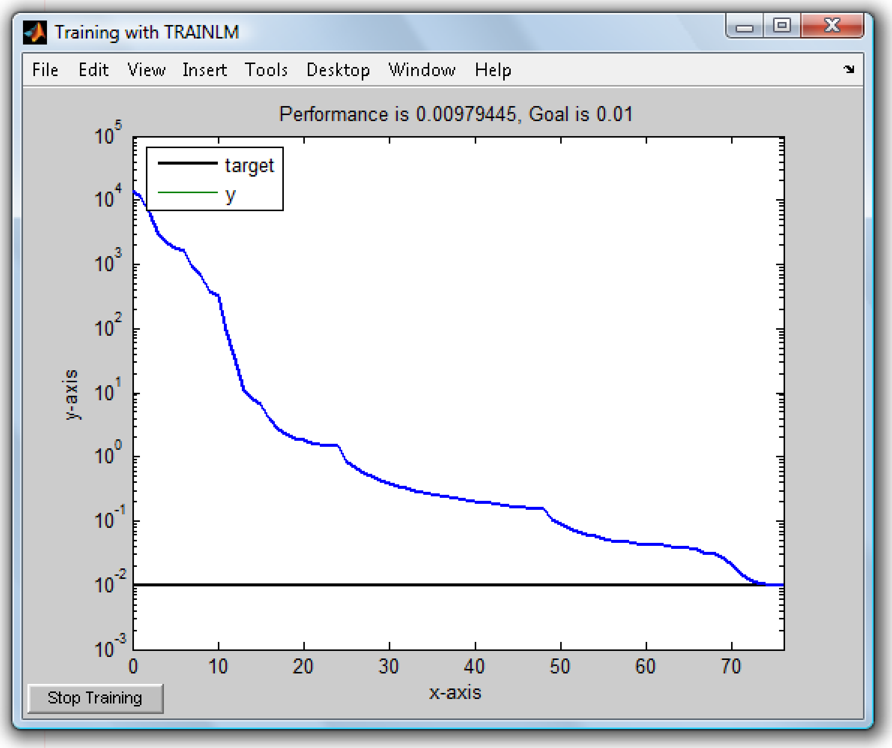

Figure5.10: Training by Neural Network with 80 Epochs

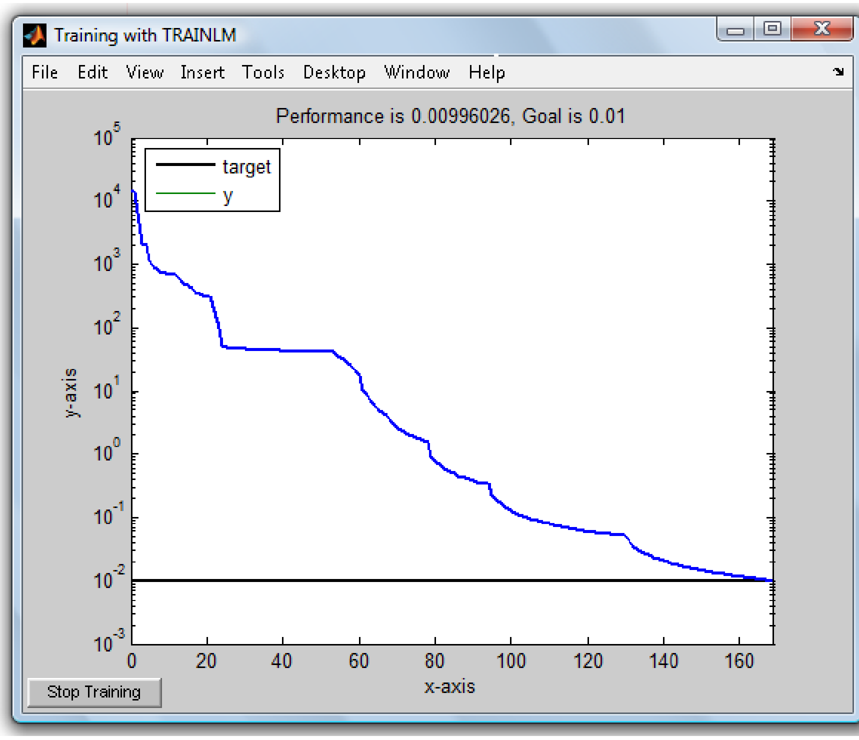

Figure5.11: Training by Neural Network with 170 Epochs

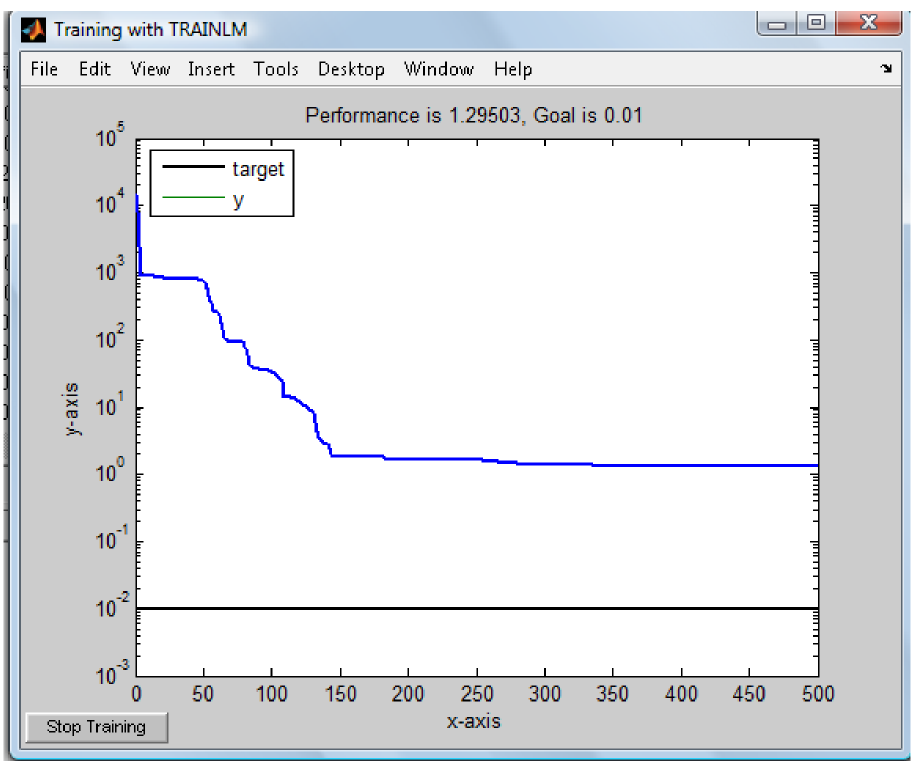

Figure5.12: Training by Neural Network with 500 Epochs

5.2 Result

Experiments were done in around 50 images and some of the results of them being shown below. Result are on the based of two fingerprint image, Finger1 (Figure5.13) and Finger2 (Figure5.16) is shown. Table 5.2 and Table 5.4 are show the buffrication points while Table 5.3 and Table 5.5 shows termination points location Finger1 and Finger2 respectively.

Figure5.13: Finger1

Finger5.14: Enhance Image of Finger1 by Fuzzy Logic

Figure5.15: Minutiae Points in Finger1

Figure5.16: Finger2

Figure5.17: Enhance Image of Finger2 by Fuzzy Logic

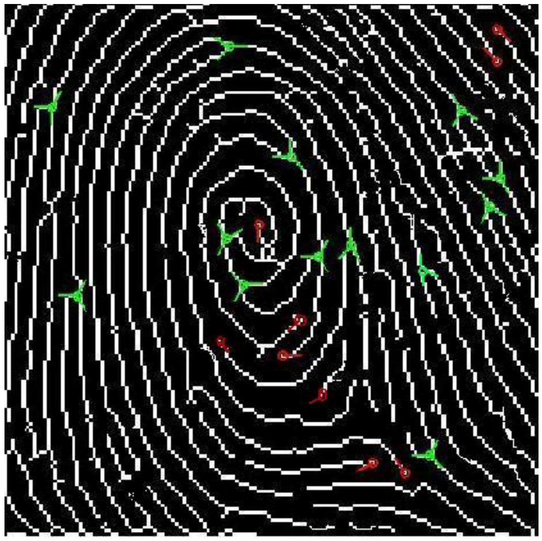

Figure5.18: Minutiae Points in Finger2

Table 5.2 and Table 5.3 show the location of minutiae point corresponding to Finger1.

SNo	X	Y	Angle1	Angle2	Angle3
1	146	13	2.46	-1.57	-0.52
2	26	70	3.14	-1.57	0.52
3	52	105	-2.36	1.57	-0.79
4	126	115	-2.36	1.57	-1.05
5	169	136	-2.36	2.09	-0.79
6	79	145	2.62	-1.05	0.52
7	178	149	-2.62	1.57	-0.79
8	77	153	-2.62	2.09	0
9	90	167	2.36	-2.36	0.52
10	119	174	2.36	-2.09	0

Table 5.2: Buffrication Points Location in Finger1

SNO	X	Y	Angle
1	54	14	0.52
2	52	26	0
3	58	29	3.14
4	42	43	-2.62
5	154	58	2.36
6	52	59	0.52
7	180	79	1.05
8	23	83	-2.09
9	61	92	-2.09
10	93	98	-1.57
11	137	116	-1.57
12	151	116	-1.05
13	108	117	-2.09
14	70	124	-0.79
15	162	126	-1.57
16	79	133	2.36
17	108	142	-2.36
18	32	146	-1.05
19	29	167	3.14
20	44	170	3.14
21	97	175	-2.92
22	75	176	3.14
23	113	180	0.52
24	119	180	0.00
25	154	180	1.05
26	49	186	-2.62
27	71	186	3.14

Table 5.3: Termination Points Location in Finger1

Table 5.4 and Table 5.5 show the location of minutiae point corresponding to Finger2.

SNo	X	Y	Angle1	Angle2	Angle3
1	128	31	2.09	-2.09	-0.52
2	80	44	2.09	-2.09	-0.79
3	135	57	2.09	-1.57	0.00
4	63	66	2.36	-2.36	0.00
5	75	68	3.14	-1.57	-1.05
6	97	69	-2.36	1.57	-1.05
7	88	72	3.14	1.05	-1.05
8	32	120	-2.62	3.14	-1.05
9	36	139	3.14	1.05	-0.79
10	63	13	2.62	-1.05	-1.05
11	13	30	3.14	-1.57	-1.05
12	119	128	3.14	1.05	0.00
13	67	80	2.36	-2.09	0.00
14	20	83	3.14	1.05	-1.05

Table 5.4: Buffrication Points Location in Finger2

SNO	X	Y	Angle
1	138	8	-0.79
2	138	17	2.36
3	55	17	2.36
4	71	63	-1.57
5	55	68	-1.05
6	89	111	1.05
7	55	116	-1.57
8	103	130	-2.62
9	14	132	-1.57
10	37	132	1.57
11	113	132	2.62
12	151	116	-1.05

Table5.5: Termination Points Location in Finger2

Chapter 6

Conclusion and Future scope

The main focus of this research is enhancement of fingerprint image by fuzzy logic and then extraction of fingerprint minutiae points and their location. So before minutia point extraction, first a new fuzzy logic based method for image enhancement was implemented. Experiment was done on poor quality fingerprint image initially and is compared to the result by applying proposed method and without applying method. Experimental results show that enhanced image is obtained by using fuzzy logic gives better result rather then original fingerprint image (poor quality image). The Crossing Number method was then implemented to perform extraction of minutiae. Experiments conducted have shown that this method is able to accurately detect almost valid bifurcations and ridge endings from the thinned image. However, there are cases where the extracted minutiae do not correspond to true minutia points. To improve the performance of methods, training has been done by feedforward neural network.

The work done in the thesis is only up to extracts minutia point. The future work can be extended to matching different fingerprint images. For this purpose first a large database of fingerprint images has to be created and then based on minutiae points, check the images from the database whether they are from same fingerprint or not. Although the method for image enhancement based on fuzzy logic is sufficient but there needs to be develop some more efficient methods for fingerprint image enhancement which can give more accurate result.

Reference

[1] A Montesanto, P. Baldassarri, G. Vallesi, G. Tascini DEIT –Università Politecnica delle Marche, “Fingerprints Recognition Using Minutiae Extraction: a Fuzzy Approach” 14th International Conference on Image Analysis and Processing, 2007

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

[3] "Applications for fuzzy Logic"

http://www.mines.u-nancy.fr/~gueniffe/CoursEMN/I3l1fcourse/fuzzy-appl-10.html, September 2003.

[4] Benno Stein, (Bauhaus University Weimar, Germany) “Fuzzy-Fingerprints for Text-Based Information Retrieval” 5th International Conference on Knowledge Management Journal of Universal Computer Science, pp. 572-579, ISSN 0948-695x, 2005

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

[6] David H. Kemsley, Tony R Martinez, Douglas M. Campbell, "A Survey of Neural Network Research and Fielded Applications", In International Journal of Neural Networks: Research and Applications, vol. 2, No. 2/3/4, pp.123-133, 1992.

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

[8] Faisel Saeed, K. M. George, Huizhu Lu, “Image Contrast Enhancement Using Fuzzy Set Theory”

[9] FuzzyTECH Home Page, http://www.fuzzytech.com. september 2003.

[10] G.Vert S. Nasser, “Minutia Verification using Fuzzy Logic and Geographic Points of Reference” IEEE International Conference on Fuzzy Systems,2005

[11] Guo, Z., and Hall, R. W., “Parallel Thinning with Two-Subiteration Algorithms”. Communications of the ACM 32, 359–373, March 1989

[12] Hamid R. Tizhoosh, “Fuzzy Image Processing**”.**

http://pami.uwaterloo.ca/tizhoosh/wfip.htm

[13] Hassoun, M. H., “Fundamentals of Artificial Neural Networks”. MIT Press, 1995

[14] Huy Huynh Nguyen, “A Neural Fuzzy Approach to Modeling the Thermal Behavior of Power Transformers” 2007

[15] Jang, J. S. R., Sun, C. -T. and Mizutani E. “Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine intelligence”. Upper Saddle River, NJ, Prentice Hall, 1997

[16] M. Hanmandlu , Devendra Jha , Rochak Sharma, “Color image enhancement by fuzzy intensification” 2001

[17] Mamdani, E.H., Assilian, S., “An Experiment in Linguistic Synthesis with a Fuzzy Logic Controller”, International Journal of Man-Machine Studies, Vol. 7, 1975

[18] McChulloch, W. S. and Pitts, W.,”A Logical Calculus of The ideas imminent in nervous activity”. Bulletin of Mathematical Biophysics, Vol. 5. pp. 115 – 133, 1943

[19] Milind Kumar V. Sarode, Dr. S.A.Ladhake, Dr. Prashant R. Deshmukh “Fuzzy system for color image enhancement” proceedings of world academy of science, engineering and technology volume 36 ,Dec 2008

[20] Pete Mc Collum, “An Introduction to Back Propagation Neural Networks”, The Newsletter of the Seattle Robotics Society.

[21] Rosalina Abdul Salam, School of Computer Sciences, University Sains Malaysia “fingerprint recognition system using neuro fuzzy set” 2007

[22] Roshini Velamuri Guide: Sarkodie-Gyan Thompson “Fingerprint recognition using Fuzzy inferencing techniques” April 20, 2006

[23] Salil Prabhakar "Fingerprint Classification and Matching using a Filterbank" Michigan State University, pp. 119-150, 2001.

[24] Takagi, T., Sugeno, M., "Fuzzy Identification of Systems and its Applications to Modelling and Control", IEEE Transactions on Systems, Man and Cybernetics, Vol. 15, pp. 116-132, 1985

[25] Tamura, H. "A Comparison of Line Thinning Algorithms from Digital Geometry viewpoint". In Proc. of the 4th Int. Conf. on Pattern Recognition, pp. 715–719, 1978

[26] The MathWorks, Inc. Fuzzy Logic Toolbox. The MathWorks Inc, 1999

[27] Tico, M., and Kuosmanen, P. "An algorithm for fingerprint image postprocessing." In Proceedings of the Thirty-Fourth Asilomar Conference on Signals, Systems and Computers vol. 2, pp. 1735–1739, Nov 2000

[28] Tsoukalas, L.H. and Uhrig, R.E., "Fuzzy and Neural Approaches in Engineering", John Wiley & Sons Inc, 1997

[29] 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

[30] Vijay Kumar Sagar and Koh .lit Beng. Alex School of Applied Science, "Fingerprint Feature Extraction by Fuzzy Logic and Neural Networks" 1999

[31] Vijayaprasad.P, and Ashraf Gasim Elsid Abdalla "Applying Neuro-Fuzzy Technique to the Enhanced Fingerprint image" 2003

[32] Zadeh, L.A., "Fuzzy Sets", Journal of Information Sciences, Vol. 8, pp. 338-353, 1965

List of Paper Published/Communicated

- Pravesh Kumar and Karun Verma, "Fingerprint Image Enhancement by Fuzzy Set" communicated at International Conference on Advances in Computing, Control and Telecommunication Technology, Trivandrum, Kerela, India Dec 24-25, 2009