

**FINGERPRINT IMAGE IDENTIFICATION USING FUZZY
LOGIC AND NEURAL NETWORKS TECHNIQUE**

**By
Ahmad Hashim Hussein**

**Supervisor
Dr. Ahmad Sharieh**

**This Thesis was Submitted in Partial Fulfillment of the Requirements for the
Master's Degree of Science in
Computer Science**

**Faculty of Graduate Studies
University of Jordan**

August, 2004

This Thesis (**Fingerprint image identification using fuzzy logic and neural networks technique**) was successfully defended and approved on

Examination Committee

Signature

Dr. Ahmad Sharieh, **Chairman**
Assoc. Prof. of Parallel Processing

.....

Dr. Sami Serhan, **Member**
Assoc. Prof. of Compiler Design

.....

Dr. Moussa Habib Abdallah, **Member**
Assoc. Prof. of Image processing

.....

Dr. Ghassan Kanaan, **Member**
Assist. Prof. of Natural language processing
(Al-Yarmouk University)

.....

DEDICATION

I dedicate my thesis to
Who of my thinking

ACKNOWLEDGEMENT

In the beginning, thanks to GOD for the care and help me.

I wish to thank my supervisor Dr. Ahmad Sharieh for his supervision, guidance and suggestion throughout the research.

I would like to thank all members of examination committee. Also, special thanks to my teachers of the department of computer science.

Many thanks go to my teacher and friend Emad Al-huly, my friend Kussay N. Mutter and my friend Braa M. Al-Baker and my friend Ali Al-Atia and my friend Tark who help and support me.

Finally, thanks to my brothers Mohamed Al-Yhyia and Ehsan Al-Yhyia for their support.

LIST OF CONTENTS

Subject	Page
Committee	ii
Dedication	iii
Acknowledgement	iv
List of contents	v
List of Tables	vi
List of Figures	viii
List of Abbreviation.....	x
Abstract in English	xi
1. Introduction	1
2. Literature review	8
3. The Fuzzy Logic and Backpropagation Neural Network Technique for Identification Fingerprint images	45
4. Experimental Results and Discussion	58
5. Conclusions and Future Work	106
References	108
Abstract in Arabic	113

LIST OF TABLES

TABLE NO.	CAPTION	PAGE
1	The energy function for each image, the minimum of energy function, the next close and the difference next close for the input unknown image.	60
2	The energy function for each image, the minimum of energy function, the next close and the difference next close for the input unknown image.	65
3	The energy function for each image and the minimum of energy function for the input unknown image. The input image is identical to one of images of system database, but it contains noise 30%.	69
4	The energy function for each image and the minimum of energy function for the input unknown image. The input image is identical to one of images of system database, but it contains noise 50%.	72
5	The energy function for each image and the minimum of energy function for the input unknown image. The input image is identical to one of images of system database, but it contains noise 70%.	75
6	The energy function for each image and the minimum of energy function for the input unknown image. The input image is identical to one of images of system database, but it contains noise 90%.	78
7	The energy function for each image and the minimum of energy function for the input unknown image. The input image is identical to one of images of system database, but it contains missing 25%.	82
8	The energy function for each image and the minimum of energy function for the input unknown image. The input image is identical to one of images of system database, but it contains missing 50%.	85
9	The unknown input image is identical to two of images of system database.	89
10	The unknown input image is identical to three images of the system database.	90
11	The unknown input image is identical to four of images of system database.	91
12	The unknown input image (2a50x50) is rotated by 2^0 , 4^0 and 6^0 .	92
13	The unknown input image (2a50x50) is rotated by -2^0 , -4^0 and -6^0 .	93
14	The unknown input image (4914a50) is rotated by 30^0 , 45^0 and 90^0 .	95
15	The unknown input image (4914a50) is rotated by -30^0 , -45^0 and -90^0 .	96

TABLE NO.	CAPTION	PAGE
16	The times of learning and converging processes in the standard backpropagation algorithm (SBP) by using images have the sizes 4x4 and 10x10 pixels.	98
17	The times of learning and converging processes in the proposed method by using images have the sizes 4x4, 10x10, 50x50, 128x128 and 256x256 pixels.	98
18	The time of identification and the total time for the proposed system by using database with images in the range 10 to 100. (The used image size is 50x50 pixels).	99
19	The accuracy of matching of standard backpropagation algorithm (SBP) and the proposed method.	100
20	The IRs on input images with noises rates 30%, 50%, 70% and 90% for both standard backpropagation algorithm (SBP) and the proposed method.	101
21	The comparison of input image, which contained missing ratio 25% and 50% for both standard backpropagation algorithm (SBP) and proposed method.	102
22	The IRs of input rotated images by both standard backpropagation algorithm (SBP) and proposed method, using different angles.	103
23	The comparisons of total complexity time of the standard backpropagation algorithm (SBP), and the proposed method.	104
24	The comparisons of total complexity time of the fuzzy logic and neural networks method (Sagar <i>et al.</i> , 1999), and the proposed method.	105

LIST OF FIGURES

FIGURE NO.	CAPTION	PAGE
1	Some of the biometrics: a) ear, b) face, c) facial thermogram, d) hand thermogram, e) hand vein, f) hand geometry, g) fingerprint, h) iris, i) retina, j) signature, and k) voice.	3
2	Various electronic access applications in widespread use that require automatic recognition.	5
3	Six major fingerprint classes. Twin loop images are labeled as whorl in the NIST database.	9
4	Minutiae examples in fingerprint.	10
5	A schematic diagram of a neuron.	13
6	An artificial neuron model.	14
7	Artificial neuron without bias.	15
8	A Single-layer neural net.	16
9	A multilayer neural net.	17
10	Competitive layer.	17
11	Linear transfer function between 0 and 1.	19
12	Linear transfer function between -1 and +1.	20
13	Step function between 0 and 1.	20
14	Linear transfer function between -1 and +1.	21
15	Backpropagation neural network with one hidden layer.	23
16	Calculation of output for backpropagation training algorithm.	25
17	Crisp Membership Function.	28
18	An Example of a Fuzzy Membership Function.	28
19	Trapezoidal Membership Functions.	28
20	Different types of membership functions.	29
21	Components of fuzzy inference system.	31
22	Two-Input Two-Rule Mamdani Fuzzy Inference Model.	34
23	Two-Input Two-Rule Sugeno Fuzzy inference Model.	36

FIGURE NO.	CAPTION	PAGE
24	The used smallest size of the net.	45
25	Creating the fuzzy membership functions for the input fingerprint image.	48
26	The fuzzy graph is drawn, if the middle average is less than the border average.	49
27	The fuzzy graph is drawn, if the border average is less than the middle average.	49
28	The fuzzy graph is drawn depending on data of the example.	49
29	The four states for the vectors.	50
30	The four vectors with their own nets and numbers of nets.	51
31	Operation of identification the nearest image by energy function.	54
32	Samples of fingerprint images which are used in the experiments. (a): The size of fingerprint images is 256 x 256 pixels. (b): The size of fingerprint images is 128 x 128 pixels. (c): The size of fingerprint images is 50 x 50 pixels.	59
33	The minimum energy function for the ten images which exist in the system database.	63
34	(a): The images in system database. (b): The first unknown input image.	63
35	(a): The images in system database. (b): The tenth unknown input image.	64
36	The minimum energy function for ten images which does not exist in the system database.	68
37	(a): The images in system database. (b): The unknown input image does not exist within system database.	68
38	The original images and the same images that contain noise 30%, 50%, 70% and 90%.	81
39	The original images and the same images contain missing 25% and 50%.	88
40	Correct convergence (a) The unknown input image (2a50x50) rotated by 2^0 . (b) The converged image (2a50x50).	94
41	Incorrect convergence (a) The unknown input image (2a50x50) rotated by -6^0 . (b) The converged image (415a50).	94
42	Window of convergence (a) The unknown input image (4914a50 Rotated 45^0). (b) The converged image (4914a50 Rotated 45^0).	97
43	Window of convergence (a) The unknown input image (4914a50 Rotated -90^0). (b) The converged image (4914a50 Rotated -90^0).	97

LIST OF ABBREVIATION

IAFIS	Integrated Automatic Fingerprint Identification Service
NNs	Neural Networks
AN	Artificial Neuron
ANNs	Artificial Neural Networks
FFBP	Feedforward Backpropagation
LMST	Least Mean Squared Error
MSE	Mean Square Error
V	Vector
DNC	Different Next Close
SBP	Standard Backpropagation algorithm
IR	Identification Rate
avmid	average of middle region
avdor	average of border region

Fingerprint Image Identification Using Fuzzy Logic and Neural Networks Technique

By
Ahmad Hashim Hussein

Supervisor
Dr. Ahmad Sharieh

ABSTRACT

The uniqueness of the human fingerprint is considered to be one of the most reliable characteristics for personal identification. Because of the large volume of fingerprints and recent advances in computer technology, there has been increasing interest in automatic identification of fingerprints.

To identify fingerprint images, we used the Fuzzy Logic and Backpropagation neural network techniques. The fuzzy logic is used to convert the digital fingerprint image to the binary image, while backpropagation net is used for the identification process. In this study, the smallest size of the net is used. The learning and convergence processes were done for parts of the fingerprints images and avoiding learning and convergence the similar parts. In addition, the Energy Function is used with the backpropagation algorithm for the identification process.

Using the backpropagation algorithm with Energy Function on parts of an image, we obtained encouraging results in terms of time of learning, time of

convergence, and size of input fingerprint image. The time of learning and convergence for the tested data was 8.2 seconds compared to 1312.7 seconds when using only standard Backpropagation algorithm (SBP). The proposed method can handle images of sizes larger (we tested up 128x128 pixels) than the SBP algorithm (less than 10x10 pixels). Also, the proposed method shows accuracy close to 100% without noise, missing, and rotation. It is better by 20% to 80% than the SBP algorithm in case of noise. It is better by 20% and 40% than the SBP algorithm in case of missing. Also, it is better by 50% than the SBP algorithm in case of rotation.

1. Introduction

1.1 Introduction

The large number of pictures that are collected in science, government, and industry for analysis are continuously increasing the importance and need for an automatic system performing picture analysis. The availability of digital computers and massive amounts of pictorial data in all fields has made picture recognition one of the major topics of current researches (Abdul Abbas, 1996). In this research, fingerprint pictures will be used for the identification process.

All developed algorithms (in fingerprint identification) tried to satisfy one or all the following conditions: makes the method automatic, simple, and fast. The trend today is to design and developing techniques capable of performing pattern recognition in the same way as that of the human beings (AL-Zewary, 1996). Artificial Intelligence systems such as neural networks and fuzzy logic are examples for these techniques.

The combination of neural networks and fuzzy logic seems natural because the two approaches generally attack the design of "intelligent" systems from different points of view. Neural networks provide algorithms for learning, classification, and optimization; whereas fuzzy logic deals with issues such as reasoning on a higher (semantic or linguistic) level. Consequently, the two technologies complement each other (Kulkarni, 2001).

1.2 The problem

In fingerprint the local ridge structures cannot be completely characterized by minutiae. Further, minutiae-based matching has difficulty in efficiently and robustly matching two fingerprint images containing different numbers of unregistered minutiae

points (Prabhakar, 2001). Therefore, we will propose a method to be more accuracy and efficiency for fingerprint identification by using both the fuzzy logic and backpropagation neural network techniques.

Fuzzy logic: By using fuzzy logic, we will propose method to convert digital fingerprint image for the recognition process. This is before learning and convergence processes of backpropagation neural network.

Backpropagation neural network: A backpropagation net (a multilayer, feedforward net trained by backpropagation) can be used to solve problems in many areas (Fausett, 1994). But, the backpropagation algorithm has the limitation of slow convergence (Wu, 1994), lengthy training cycles (Kosko, 1992).

The backpropagation algorithm was failed as a tool for fingerprint classification; the reasons for this failure are (Abdul Abbas, 1996):

1. Convergence time, when the network iterates it takes a long time for adjusting the weights.
2. Array size is restricted using the sequential IBM PC machine. This can be overcome when the multilayer perceptron is applied using the main frame machines.

In this study, the backpropagation algorithm, using the proposed method was successful as a tool for fingerprint classification and identification.

1.3 Importance and applications

Biometrics, which refers to identifying an individual based on his or her physiological or behavioral characteristics, has the capability to reliably distinguish between an authorized person and an imposter. Since biometrics characteristics are distinctive, cannot be forgotten or lost, and the person to be authenticated needs to be

physically present at the point of identification, biometrics is inherently more reliable and more capable than traditional knowledge-based and token-based techniques (Prabhakar, 2001).

Biometrics is the measurement of unique physical characteristics, and it is an ideal solution to the problem of digital identification. Biometrics makes it possible to identify ourselves to digital systems, and through these systems identify ourselves to others (Digital_persona, 2003).

Among all biometrics such as: face, fingerprint, hand geometry, iris, retina, signature, voice print, facial thermogram, hand vein, gait, ear, odor and keystroke dynamics, the fingerprint-based identification is one of the most mature and proven technique (Prabhakar, 2001). Figure 1 shows some of these biometrics.

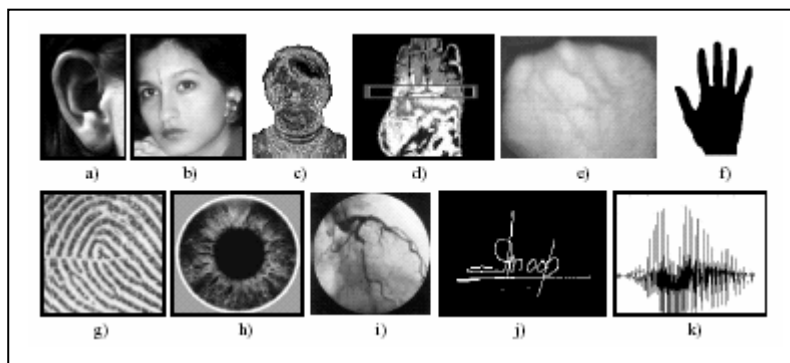


Figure 1: Some of the biometrics: a) ear, b) face, c) facial thermogram, d) hand thermogram, e) hand vein, f) hand geometry, g) fingerprint, h) iris, i) retina, j) signature, and k) voice (Maltoni *et al.*, 2003).

The advantages of fingerprint biometrics for the purpose of personal digital identification include (Digital_persona, 2003):

1. Each one of our ten fingerprints is unique, different from one another and from those of every other person. Even identical twins have unique fingerprints.
2. Unlike passwords, PIN codes, and smartcards that we depend upon today for identification, the fingerprints are impossible to lose or forget, and they can never be stolen.
3. A person has ten fingerprints as opposed to one voice, one face or two eyes.
4. Fingerprints have been used for centuries for identification, and there is substantial body of real world data upon which to base our claim of the uniqueness of each fingerprint.

With the advent of electronic banking, e-commerce, and smartcards and increased emphasis on the privacy and security of identification stored in various database, automatic personal identification has become a very important topic. Accurate automatic personal identification is now needed in a wide range of civilian application involving the use of passports, cellular telephones, automatic teller machines, and driver licenses. Traditional knowledge-based (password or Personal Identification Number (PIN)) and token-based (passport, driver license, and ID card) identifications are prone to fraud because PINs may be forgotten or guessed by an imposter and the tokens may be lost or stolen. Therefore, traditional knowledge-based and token-based approaches are unable to satisfy the security requirements of our electronically interconnected information society (see Figure 2) (Prabhakar, 2001), (Maltoni *et al.*, 2003).

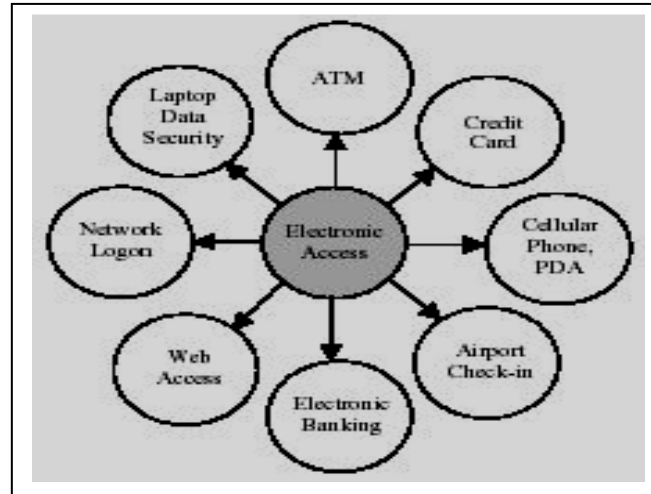


Figure 2: Various electronic access applications in widespread use that require automatic recognition (Maltoni *et al.*, 2003).

1.4 Objectives of the study

The objective of this study is to design and implement fingerprint identification system, which use texture of image. The system is based on the use of feedforward backpropagation neural network and fuzzy logic. The fuzzy logic is used to convert the digit image to binary image before the learning and convergence processes.

This system will be more efficiency and accuracy using the following certain principles:

1. Doing the learning and convergence processes for parts of the image and not all and avoiding learning and convergence the similar parts several times will speed up the learning and convergence processes.
2. Using the smallest size of net will reduce size of learning weights matrices of the learning process.
3. Using the energy function based on Hopfield neural network will help converging to the correct image in high efficiency.

1.5 Pattern Recognition

Pattern recognition can be defined as the categorization of input data into identifiable classes via the extraction of significant features or attributes of the data from a background of irrelevant detail (Radhi, 2003).

Recognition is regarded as a basic attribute of human beings, as well as other living organisms. A pattern is the description of an object. We are performing acts of recognition every instant of our working lives. We recognize the objects around us, and we move and act in relation to them. We can spot a friend in a crowd and recognize what he says. You can recognize the voice of a known individual. You can read handwriting and analyze fingerprints; we can distinguish smiles from gestures of anger. A human being is a very sophisticated information system, partly because he possesses a superior pattern recognition capability (Radhi, 2003).

1.5.1 Neural Networks

A neural net consists of a large number of simple processing elements called neurons, units, cells, or nodes. Each neuron is connected to other neurons by means of directed communication links, each with an associated weight. The weights represent information being used by the net to solve a problem. Neural net can be applied to a wide variety of problems such as: storing and recalling data or patterns, classifying patterns, performing general mappings from input patterns to output patterns, grouping similar patterns, or finding solutions to constrained optimization problems (Fausett, 1994).

The neural networks provide low-level algorithms for learning and optimization that can be combined with other technologies such as fuzzy logic or genetic algorithms to design smart engineering systems. Neural networks are biologically inspired and

provide a powerful tool or designing computer vision systems (Kulkarni, 2001).

1.5.2 Fuzzy Logic

Fuzzy Logic is a departure from classical two-valued sets and logic, that uses "soft" linguistic (e.g. large, hot, tall) system variables and a continuous range of truth values in the interval $[0,1]$, rather than strict binary (True or False) decisions and assignments (Bonde. 2000).

1.6 Structure of the thesis

Chapter One defines the problem and explains its importance and applications. Chapter Two is devoted to a theoretical background for fingerprint classification and identification, neural networks, fuzzy logic. In chapter Three, the implementation of the proposed system is discussed. Chapter Four discusses the experimental results. Chapter Five presents conclusion and future work.

2. LITERATURE REVIEW

2.1 Introduction

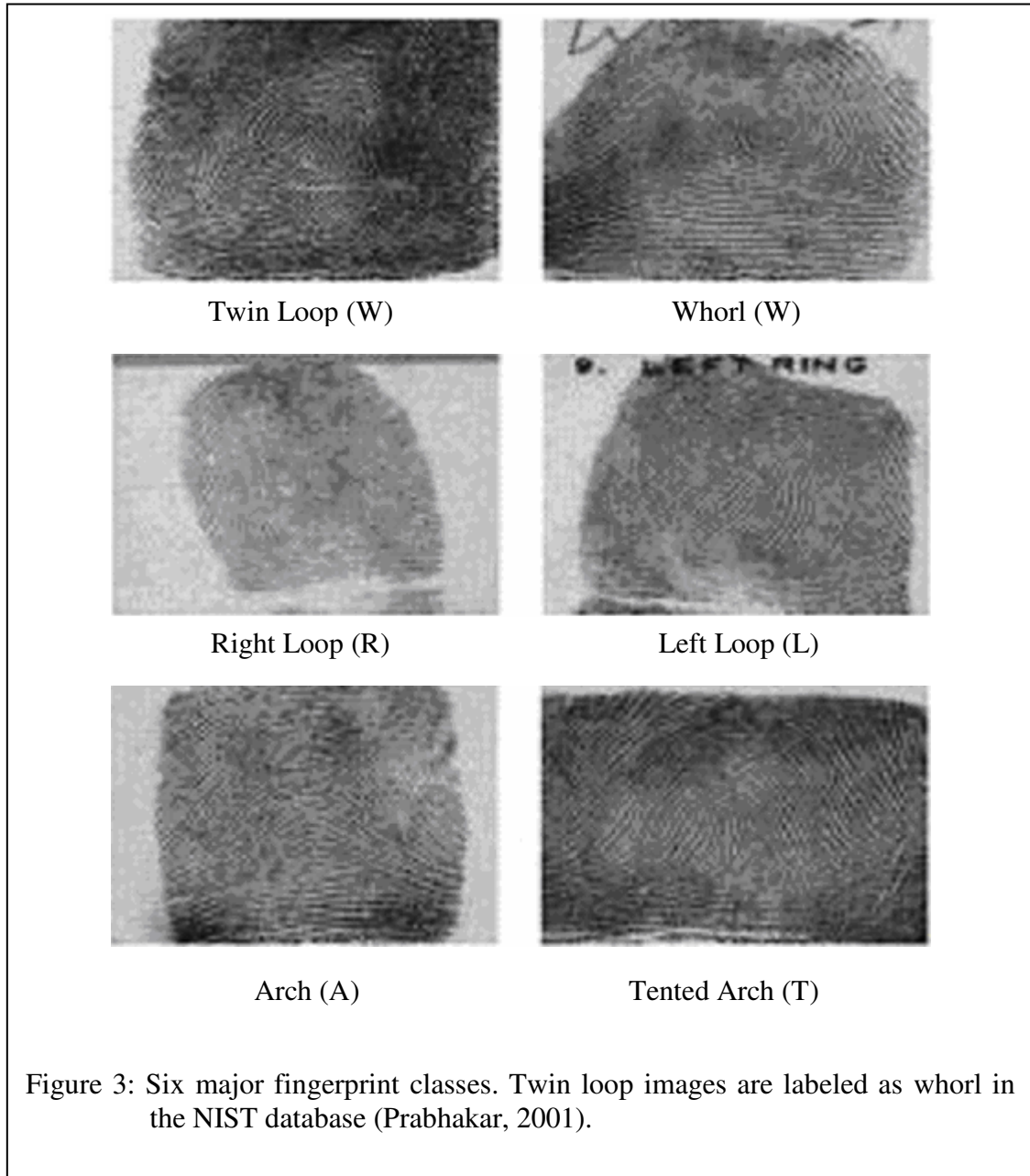
In this chapter, a theoretical background will be introduced on fingerprint classification and identification. Furthermore, we shall introduce Neural Networks and Fuzzy Logic techniques, since these topics will be used and analyzed. The related work will be presented.

2.2 Fingerprint classification

Edward Henry made a more advanced classification of fingerprints, which is known as “*Henry’s classification*”. The real significance of fingerprint patterns is based mainly on the following principles (Abdul Abbas, 1996):

1. Unchangeability: the configuration and details of the patterns are permanent and never change through life until the skin disintegrates after death.
2. Uniqueness: the degree of variation of the ridges is so high that two patterns with the same character never occur even in another finger of the same person or in different individuals.
3. The fingerprint pattern variations are within a limit, which allows a systematic classification of these configurations.

The fingerprints are classified into five distinct classes: *whorl (W)*, *right loop (R)*, *left loop (L)*, *arch (A)*, and *tented arch (T)* (see Figure 3). These classes are chosen based on the classes identified by the National Institute of Standards and Technology (NIST) to benchmark automatic fingerprint classification algorithms (Prabhakar, 2001).



There are two main types of features in a fingerprint: (i) global ridge and furrow structures, which form special patterns in the central region of the fingerprint, and (ii) local ridge and furrow minute details (Figure 4). A fingerprint is classified based on only the first type of features and is uniquely identified based on the second type of features (ridge endings and bifurcations, also known as minutiae) (Prabhakar, 2001).

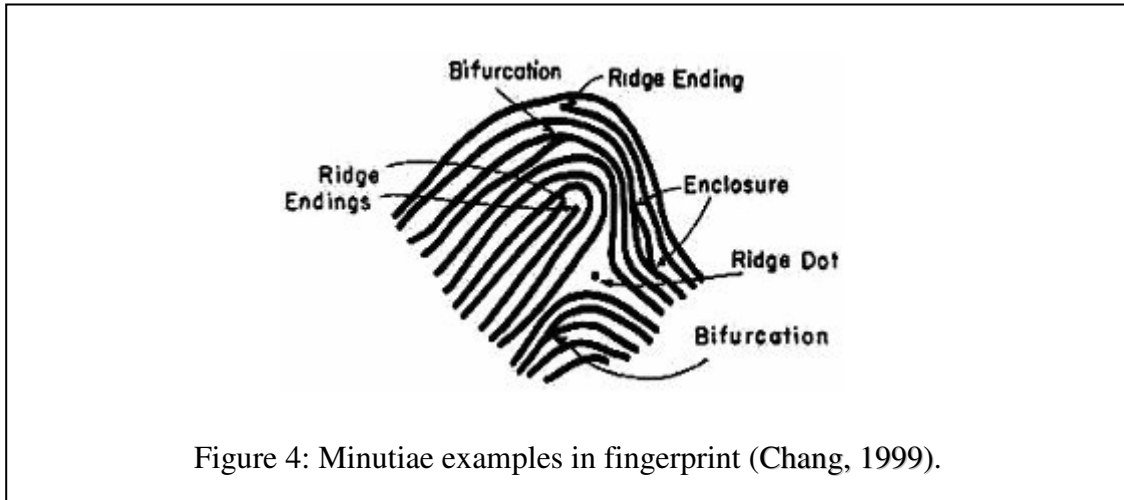


Figure 4: Minutiae examples in fingerprint (Chang, 1999).

2.3 Fingerprint Image Representation

The input data is acquired by digitizing fingerprint images, which are obtained either through a dedicated fingerprint scanner or a standard scanning of inked fingerprint impressions (Chang, 1999).

The digital image is represented as a two-dimensional array of data $I(x, y)$, where each pixel (picture element) value corresponds to the brightness of the image at the point (x, y) . In linear algebra terms, a two-dimensional array is referred to as a matrix, and one row (or column) is called a vector. This image model is for monochrome (one-color, this is what we normally refer to as black and white) image data, but normally we have other types of image data, these are color images. Gray-scale images are referred to as monochrome images; they contain brightness information only, i.e. no color information. The number of bits used for each pixel determines the number of different brightness levels available. Binary images are the simplest type of images and can take only two values, typically black and white, or '0' and '1' (Umbaugh, 1998).

Even though computers have much more powerful, binary vision systems are useful. First of all, the algorithms for computing properties of binary images are well understood. They also tend to be less expensive and faster than vision systems that operate on color or levels. This is due to the significantly smaller memory and

requirements of a gray level system working with 256 gray or color levels will be eight times that of a system working with a binary image of the same size (Eskaf, 1999).

2.3.1 Fingerprint Image Thresholding

Thresholding is a method to convert a gray scale image into a binary image so those objects of interest are separated from the background. For thresholding to be effective in object-background separation, it is necessary that the objects and background have sufficient contrast and that we know the intensity levels of either the objects or the background. In a fixed thresholding scheme, these intensity characteristics determine the value of the threshold. The thresholding illustrate that all pixels of brightness less than that of a selected threshold brightness will be set to black (0), and those equal to or above will be set to white (255) (Baxes, 1994).

2.4 Fingerprint Identification

Fingerprint Identification is one of the most commonly used biometric technologies. Each individual has unique fingerprints. A fingerprint represents the pattern of ridges and valleys (also called furrows) on the surface of the finger (Kulkarni, 2001).

Automatic fingerprint recognition requires that the input fingerprint be matched with a large number of fingerprints in a database of fingerprints. The problem of automatic recognition has been widely studied, but has not been solved completely. The main difficulty is that, in practice, the quality of a fingerprint, which depends upon the acquisition process, is often not very good. Noise and deficiencies in contrast often produce false minutiae and fail to recognition valid minutiae. Fingerprint recognition is the basic task of the Integrated Automatic Fingerprint Identification Service (IAFIS). The IAFIS has grouped fingerprints in ten main categories. Ten-print-based

identification and latent fingerprint recognition are two main concerns (Kulkarni, 2001).

2.5 Neural Networks

Neural networks have been used to model the human vision system. They are biologically inspired and contain a large number of simple processing elements that perform in a manner analogous to the most elementary functions of neurons. The Neural networks learn by experience, generalize from previous experiences to new ones, and can make decisions. The Neural network models mimic the human brain (Kulkarni, 2001).

The neural network is a network of interconnected elements. These elements were inspired from studies of biological nervous systems. The neural networks are an attempt to create machines that work in a similar way to the human brain by building these machines using components that behave like biological neurons (Radhi, 2003).

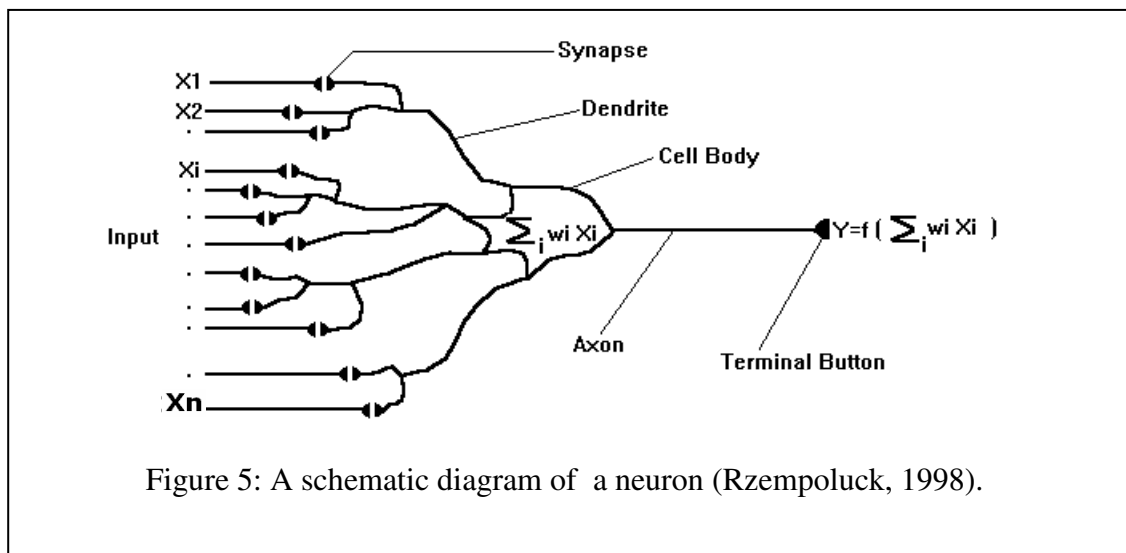
Each neuron has an internal state, called its activation or activity level, which is a function of the inputs it has received. Typically, a neuron sends its activation as a signal to several other neurons. It is important to note that a neuron can send only one signal at a time, although that signal is broadcast to several other neurons (Fausett, 1994).

The function of a neural network is to produce an output pattern when presented with an input pattern. This concept is rather abstract, so one of the operations that a neural network can be made to do - pattern classification -.The Pattern classification is the process of sorting patterns into one group or another (Radhi, 2003).

The Neural networks (NNs) provide a reasonable and powerful alternative to conventional classifiers. Potential benefits of neural networks extend beyond the high computation rates provided by massive parallelism. The NNs provide a greater degree of robustness and fault tolerance (Kulkarni, 2001).

2.5.1 Biological and Artificial Neural Networks

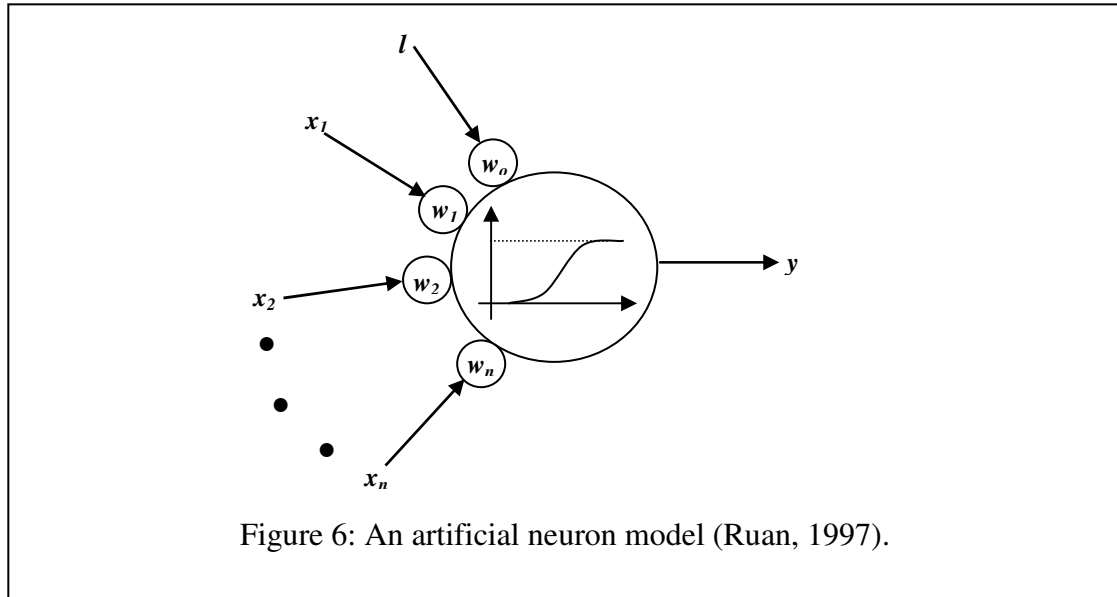
A schematic model of a biological neuron is shown in Figure (5). A typical biological neuron consists of four basic parts: dendrites, synapse, cell body, which is also called the soma, and the axon. Dendrites are branchlike structures that provide inputs to the cell body. Dendrites receive signals from neighboring neurons at connection points called synapses. The cell body essentially sums the membrane potentials provided by the dendrites. When the cumulative excitation in the cell body exceeds the threshold, the cell fires, sending a signal down the axon to other neurons. Axons are long fibers that serve as transmission lines (Kulkarni, 2001).



An artificial neuron (AN) model simulates multiple inputs and one output. The switching (activation) function of input-output relation and the adaptive synaptic weights (see Figure (6)). The first neuron model proposed in 1943 used a step function for the switching function. Several continuous functions, such as sigmoidal or radial functions, are used as a neuron characteristic function, which result in higher performance of NNs (Ruan, 1997).

Several learning algorithms that change the synaptic weights have been proposed.

The combination of the artificial NNs (ANNs) and the learning algorithms have been applied to several engineering purposes (Ruan, 1997).



A schematic diagram representing an artificial neuron (AN) with no bias is shown in Figure (7). The basic operation of an artificial neuron involves summing its weighted input signals and applying an activation function to generate the output. Let $x = (x_1, x_2, \dots, x_n)$ represent the n inputs applied to the artificial neuron (Kulkarni, 2001).

The net input to the neuron at time t is given by Equation (2.1).

$$net(t) = \sum_{i=1}^n x_i w_i \dots\dots\dots (2.1)$$

Where w_i represents the weight for the input x_i .

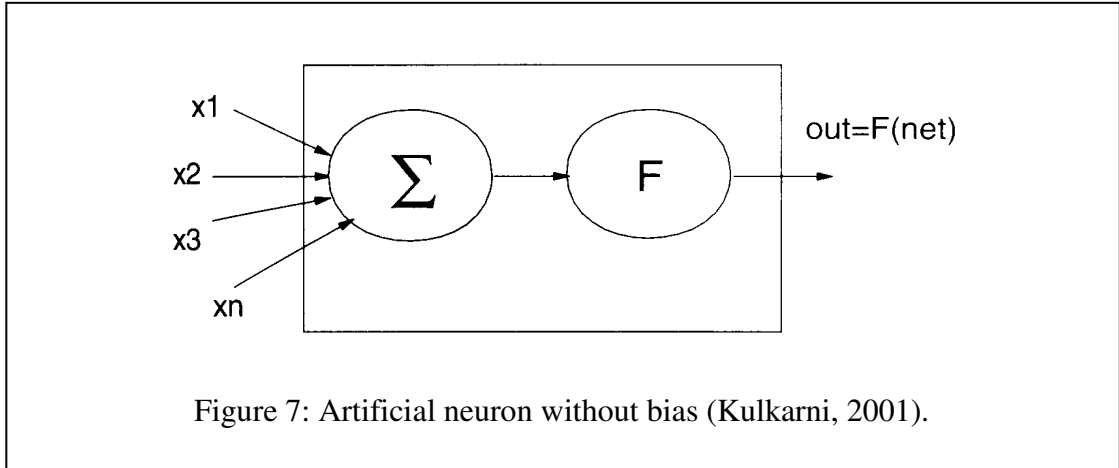
The net input, $net(t)$, is further processed by an activation function F to produce the neuron's output signal $o(t)$. Often, activation values are restricted to the range $[0, 1]$. The most commonly used activation function is the sigmoid function, which is given by Equation (2.2).

$$F(x) = \frac{1}{1 + \exp(-x)} \dots\dots\dots (2.2)$$

The output of the neuron with the sigmoid activation function is then given by

Equation (2.3).

$$o(t) = \frac{1}{1 + \exp(-net(t))} \dots\dots\dots (2.3)$$



An ANN is an information-processing system that has certain performance characteristics in common with biological neural networks. The ANNs have been developed as a generalization of mathematical models of human cognition or neural biology based on the assumptions that (Fausett, 1994):

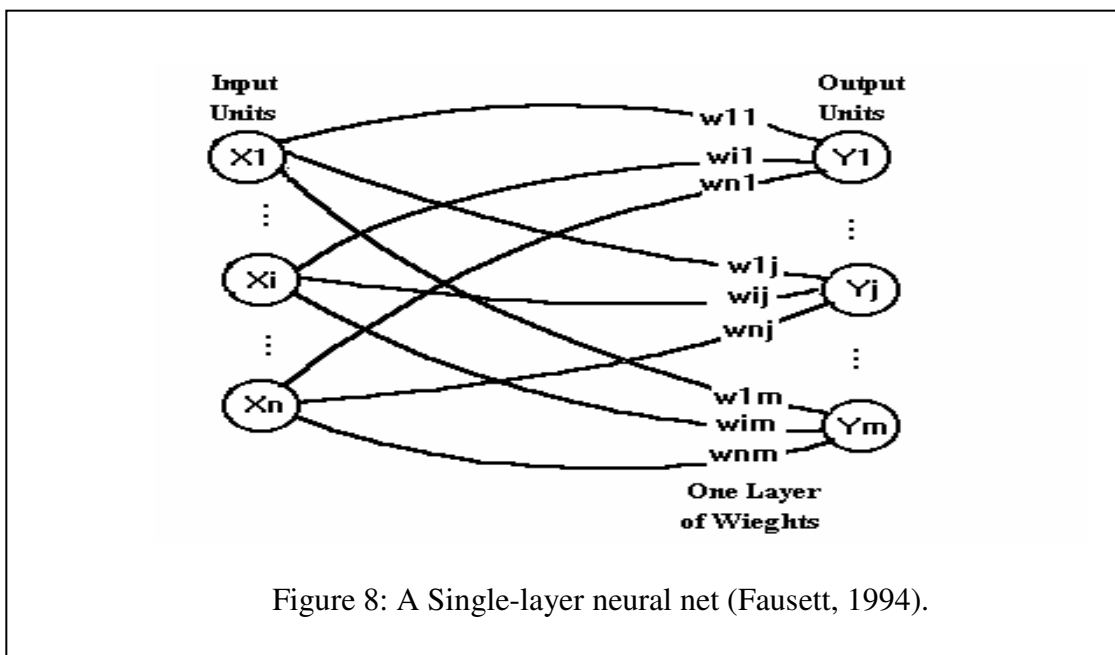
1. Information processing occurs at many simple elements are called neurons.
2. Signals are passed between neurons over connection links.
3. Each connection link has an associated weight, which, in atypical neural net, multiplies the signal transmitted.
4. Each neuron applies an activation function (usually nonlinear) to its net input (sum of weighted input signals) to determine its output signal.

A neural network is characterized by (1) its pattern of connections between the neurons (called its architecture), (2) its method of determining the weights on the connections (called its training, or learning algorithm), and (3) its Activation function (Fausett, 1994).

2.5.2 Architectures

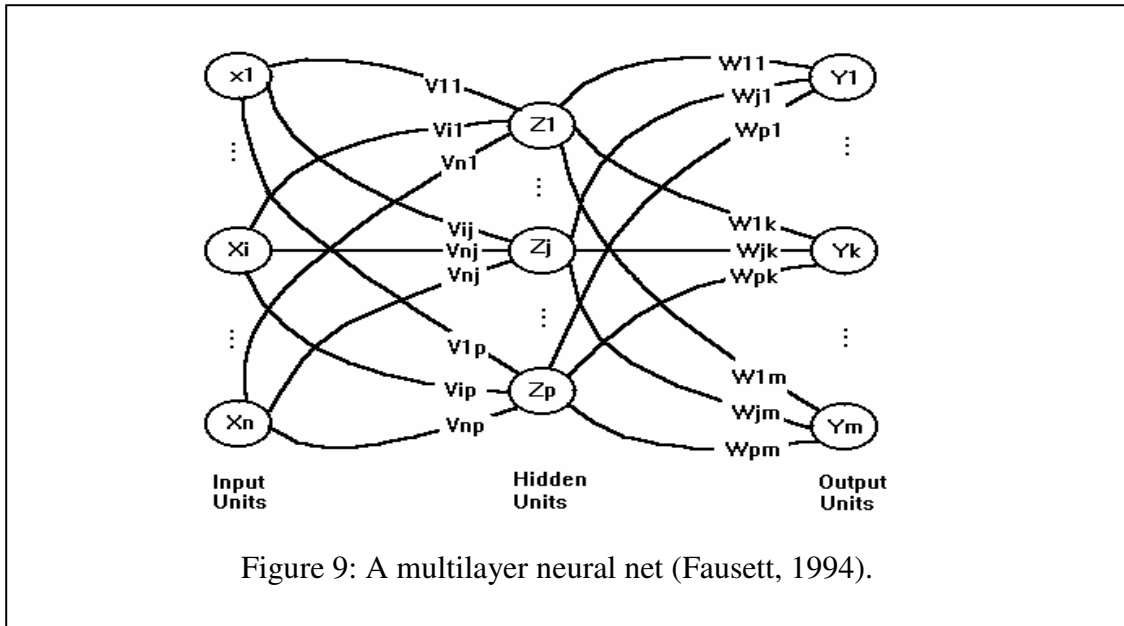
2.5.2.1 Single-Layer Net

A single-layer net has one layer of connection weights. Often, the units can be distinguished as input units, which receive signals from the outside world, and output units, from which the response of the net can be read. In the typical single - layer net shown in Figure (8), the input unit is fully connected to output units but are not connected to other input units (Fausett, 1994).



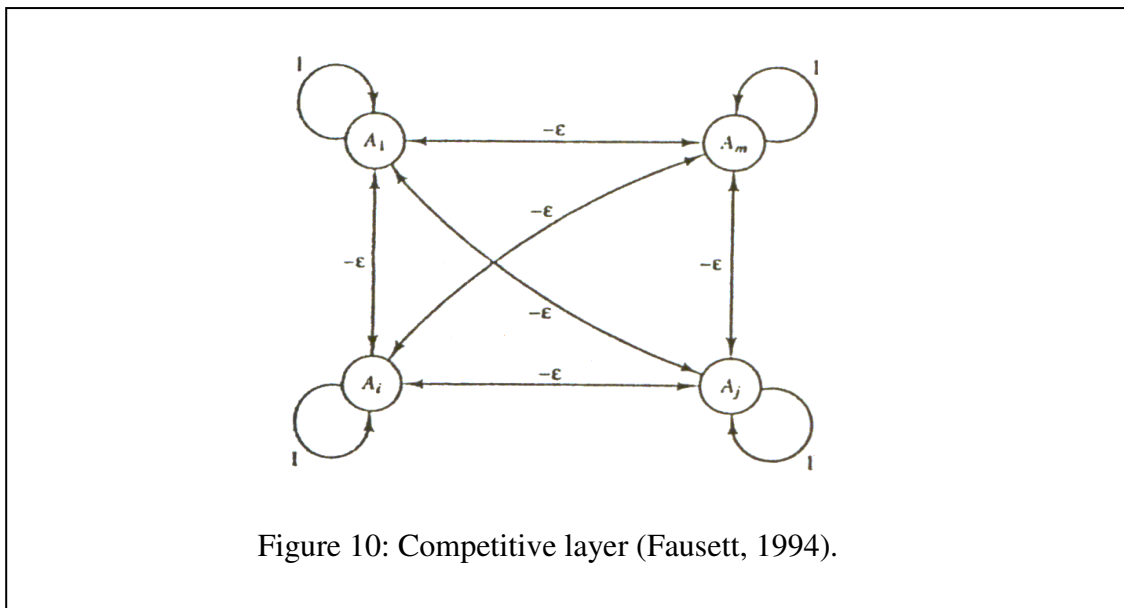
2.5.2.2 Multilayer Nets

A multilayer net is a net with one or more layers (or level) of nodes (so-called hidden units) between the input units and the output units (Figure (9)). Typically, there is a layer of weights between two adjacent levels of units (input, hidden, or output). Multilayer nets can solve more complicated problems than single - layer nets can, but training may be more difficult. However, in some cases, training may be more successful, because it is possible to solve a problem that a single layer net cannot be trained to perform correctly at all, see Figure (9) (Fausett, 1994).



2.5.2.3 Competitive layer

A Competitive layer forms a part of a large number of NN's. Typically, the interconnections between neurons in the Competitive layer are not shown in the architecture diagrams for such nets. An example of the architecture for a competitive layer is given in Figure (10); the competitive interconnections have weight of $-\epsilon$. The operation of a winner-take-all competition (Fausett, 1994).



2.5.3 Setting and weights

In addition to the architecture, the method of setting the values of the weights (training) is an important distinguishing characteristic of different neural nets. For convenience, there are two types of training-supervised and unsupervised- for a neural network; in addition, there are nets whose weights are fixed without an iterative training process (Fausett, 1994):

- Supervised training, in which the training is accomplished by presenting a sequence of training vectors or patterns, each with an associated target output vector. The weights are then adjusted according to a learning algorithm.
- Unsupervised training, that is self-organizing neural nets group similar input vectors together without the use of training data to specify what a typical member of each group looks like or to which group each vector belongs. A sequence of input vectors is provided, but no target vectors are specified. The net modifies the weights so that the most similar input vectors are assigned to the same output unit.
- Fixed-weight nets still other types of neural nets can solve constrained optimization problems. Such nets may work well for problems that can cause difficulty for traditional techniques, such as problems with conflicting constraints (i.e., not all constraints can be simultaneously satisfied). Often, in such cases, a nearly optimal solution (which the net can find) is satisfactory. When these nets are designed, the weights are set to represent the constraints and the quantity to be maximized or minimized.

The weights used on the connections between different layers have much significance in the working of the neural network and the characterization of a network. The following actions are possible in a neural network (Vallurn and Hayoriva, 1996):

Start with one set of weights and run the network (NO TRAINING).

Start with one set of weights, run the network, and modify some or all the weights, and run the network again with the new set of weights. Repeat this process until some predetermined goal is met (TRAINING).

2.5.4 The activation function

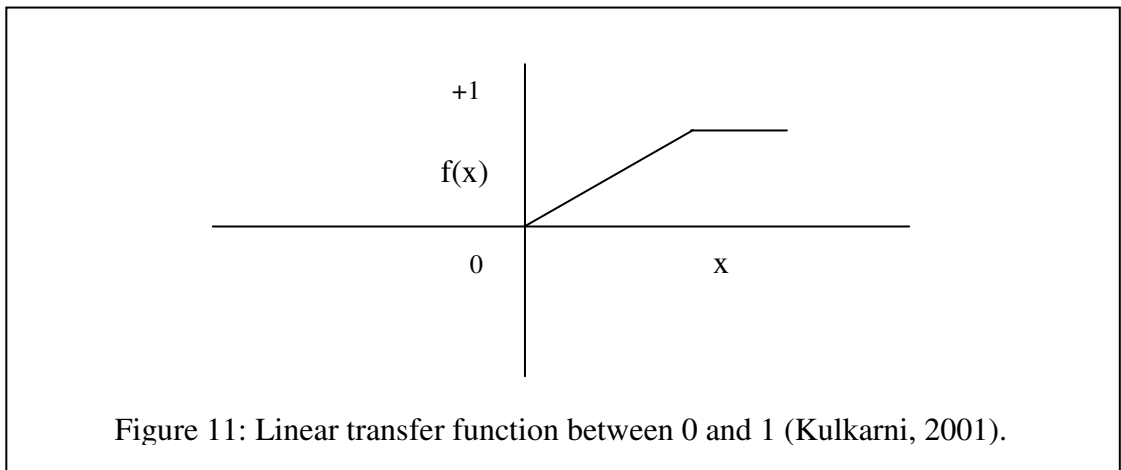
There are many types of activation functions. Often, nonlinear activation functions are used. Some of the activation functions are as follows (Kulkarni, 2001):

(a) Identity function:

$$F(x) = x \quad \forall x \quad \dots\dots\dots (2.4)$$

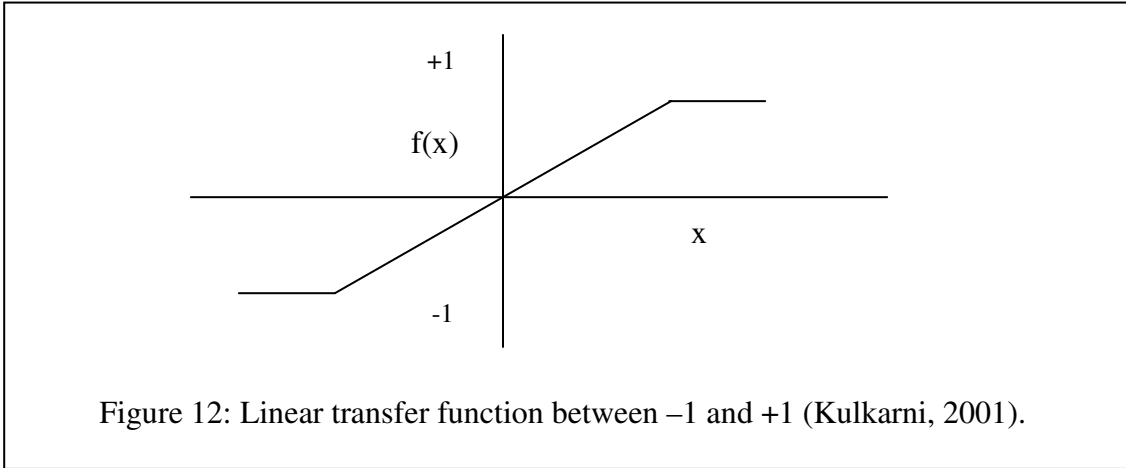
Neurons with this activation function are used in the input layer. These neurons are often single input units, and they just pass input signals to the next layer. The identity function is sometimes referred to as the linear function. It is also possible to limit its output to values in the range [0, 1]. The plot for a saturated linear function that varies between 0 and 1 is shown in Figure (11) and is given by Equation (2.5).

$$F(x) = \begin{cases} 0 & \text{if } x \leq 0 \\ x & \text{if } 0 < x < 1 \\ 1 & \text{if } x \geq 1 \end{cases} \quad \dots\dots\dots (2.5)$$



The plot for a saturated linear function that varies between -1 and +1 is shown in Figure (12) and is given by Equation (2.6).

$$F(x) = \begin{cases} -1 & \text{if } x \leq -1 \\ x & \text{if } -1 < x < 1 \\ 1 & \text{if } x \geq 1 \end{cases} \dots\dots\dots (2.6)$$

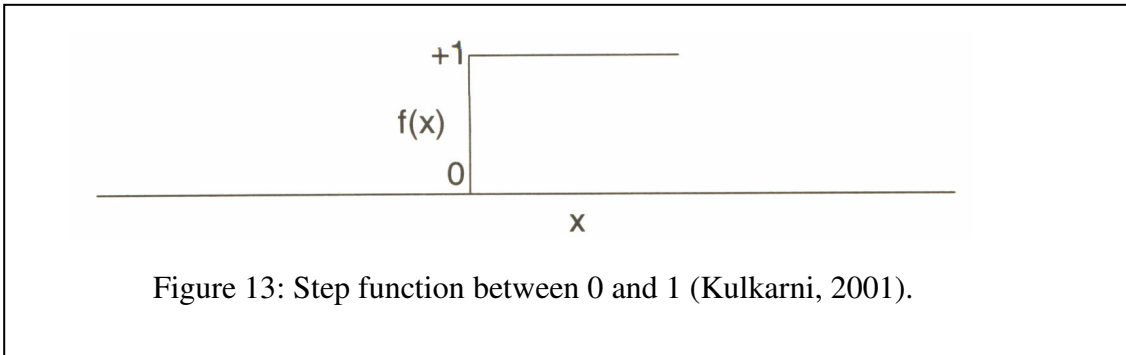


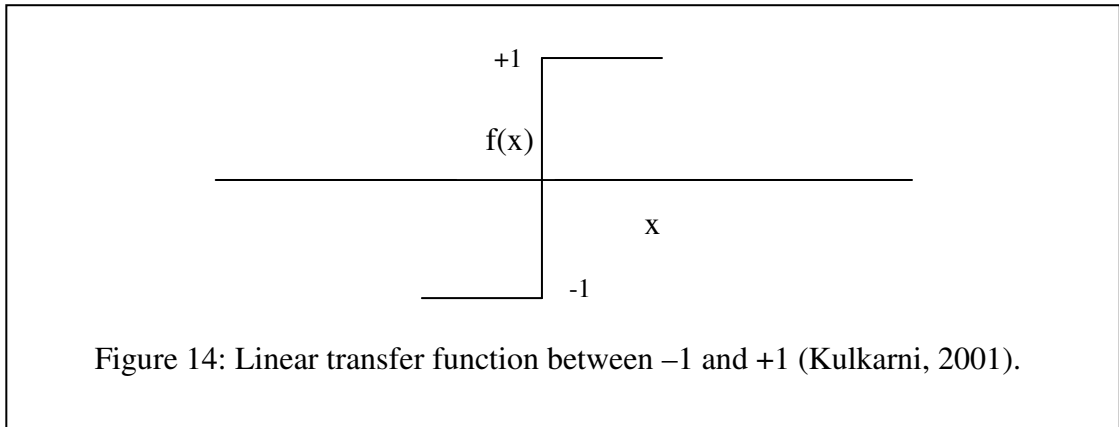
(b) Step function:

In this type of unit, outputs are binary and they can take values as 0 and 1 or -1 and +1. A plot of a step function is shown in Figures (13), (14). Activation functions for these units are given by Equations (2.7) and (2.8).

$$F(x) = \begin{cases} 1 & \text{if } x > 0 \\ 0 & \text{if } x \leq 0 \end{cases} \dots\dots\dots (2.7)$$

$$F(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ -1 & \text{if } x < 0 \end{cases} \dots\dots\dots (2.8)$$





(c) Sigmoid function:

The sigmoid function is the most commonly used activation function. It is preferred because it is smooth and bounded, and it has a simple derivative. The sigmoid function is represented by Equation (2.9).

$$F(x) = \frac{1}{1 + \exp(-\alpha x)} \dots\dots\dots (2.9)$$

Where α is a constant that typically varies between 0.01 and 1.0.

A sigmoid function can be scaled to have a range of values.

(d) Hypertangent function:

A hypertangent function is similar to the sigmoid function and is given by Equation (2.10).

$$F(x) = \frac{1 - \exp(-\alpha x)}{1 + \exp(-\alpha x)} \dots\dots\dots (2.10)$$

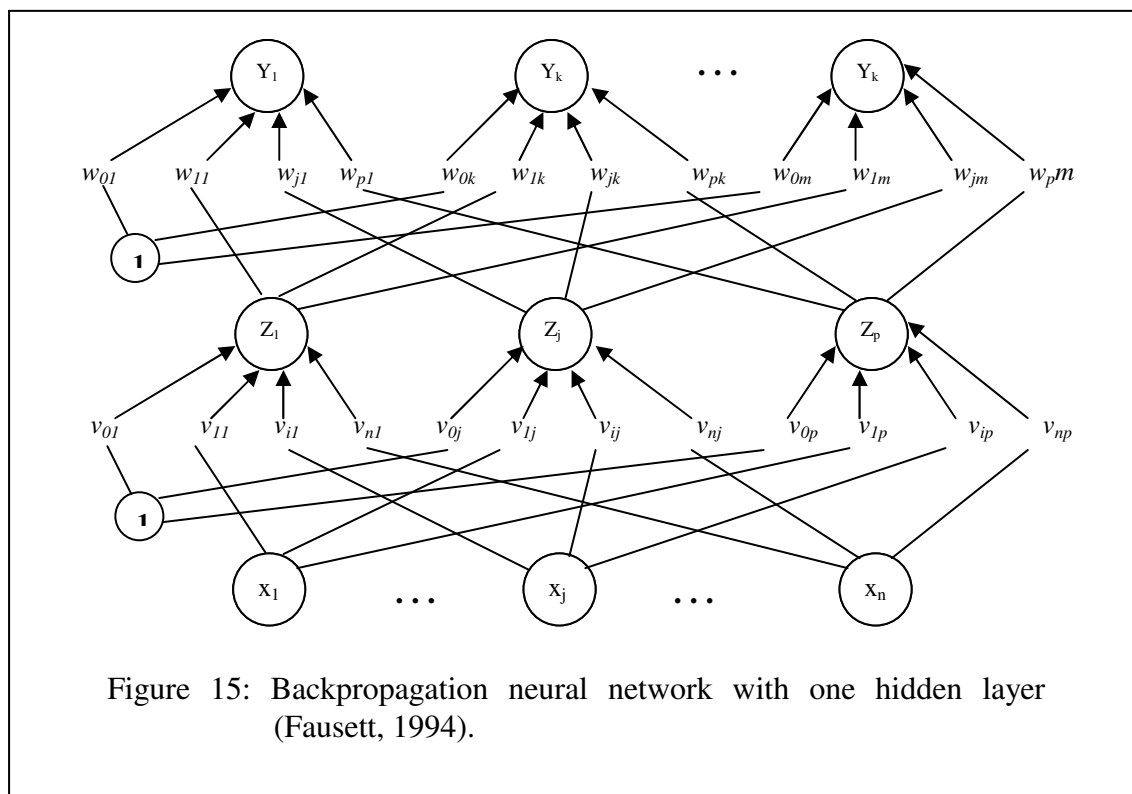
The function varies from -1 to +1.

2.5.5 Feedforward Backpropagation Network

The feedforward backpropagation (FFBP) network is a very popular model in neural networks. It does not have feedback connections, but errors are backpropagated during training. Least mean squared error (LMSE) is used. Many applications can be formulated for using (FFBP) network, and the methodology has been a model for most multilayer neural networks. Errors in the output determine measures of hidden layer output errors, which are used as a basis for adjustment of connection weights between the input and hidden layers. Adjusting the two sets of weights between the pairs of layers and recalculating the outputs is an iterative process that is carried on until the errors fall below a tolerance level. Learning rate parameters scale the adjustments to weights. A momentum parameter can also be used in scaling the adjustments from a previous iteration and adding to the adjustments in the current iteration (Vallurn and Hayoriva, 1996).

2.5.5.1 Architecture

A multilayer neural network with one layer of hidden units (the Z units) is shown in Figure (15). The output units (the Y units) and the hidden units also may have biases as shown in Figure (15). The bias on a typical output unit Y_k is denoted by w_{ok} ; the Z_j is denoted v_{oj} . These bias terms act like weights on connections from units whose output is always 1. (These units are shown in Figure (15) but are usually not displayed explicitly.) Only the direction of information flow for the feedforward phase of operation is shown. During the backpropagation phase of learning, signals are sent in the reverse direction. (Fausett, 1994)



2.5.5.2 Training Algorithm

The backpropagation training algorithm is an iterative gradient algorithm designed to minimize the mean square error (MSE) between the actual output of a multilayer feedforward perceptron and the desired output. It requires continuous differentiable non-linearities. The following assumes a sigmoid logistic nonlinearity is used (Lippmann, 1987).

Step 1. Initialize weights and offsets

Set all weights and node offsets to small random values.

Step 2. Present input and desired outputs

Present a continuous valued input vector x_0, x_1, \dots, x_{N-1} and specify the desired outputs d_0, d_1, \dots, d_{M-1} . If the net is used as a classifier then all desired outputs are typically set to zero except for that corresponding to the class the input is

from. That desired output is 1. The input could be new on each trial or samples from a training set could be presented cyclically until weights stabilize.

Step 3. Calculate actual outputs

Use the sigmoid nonlinearity and formulas as in Figure (16) to calculate outputs

$$y_0, y_1 \dots y_{m-1}$$

Step 4. Adapt weights

Use a recursive algorithm starting at the output nodes and working back to the first hidden layer. Adjust weight by

$$W_{ij}(t+1) = w_{ij}(t) + \eta \delta_j x_i'$$

In this equation $w_{ij}(t)$ is the weight from hidden node j or from an input to node j at time t , x_i' is either the output of node j or is an input, η is a gain term, and

δ_j is an error term for node j . if node j is an output node, then

$$\delta_j = y_j (1 - y_j) (d_j - y_j),$$

where d_j is the desired output of node j and y_j is the actual output.

If node j is an internal hidden node, then

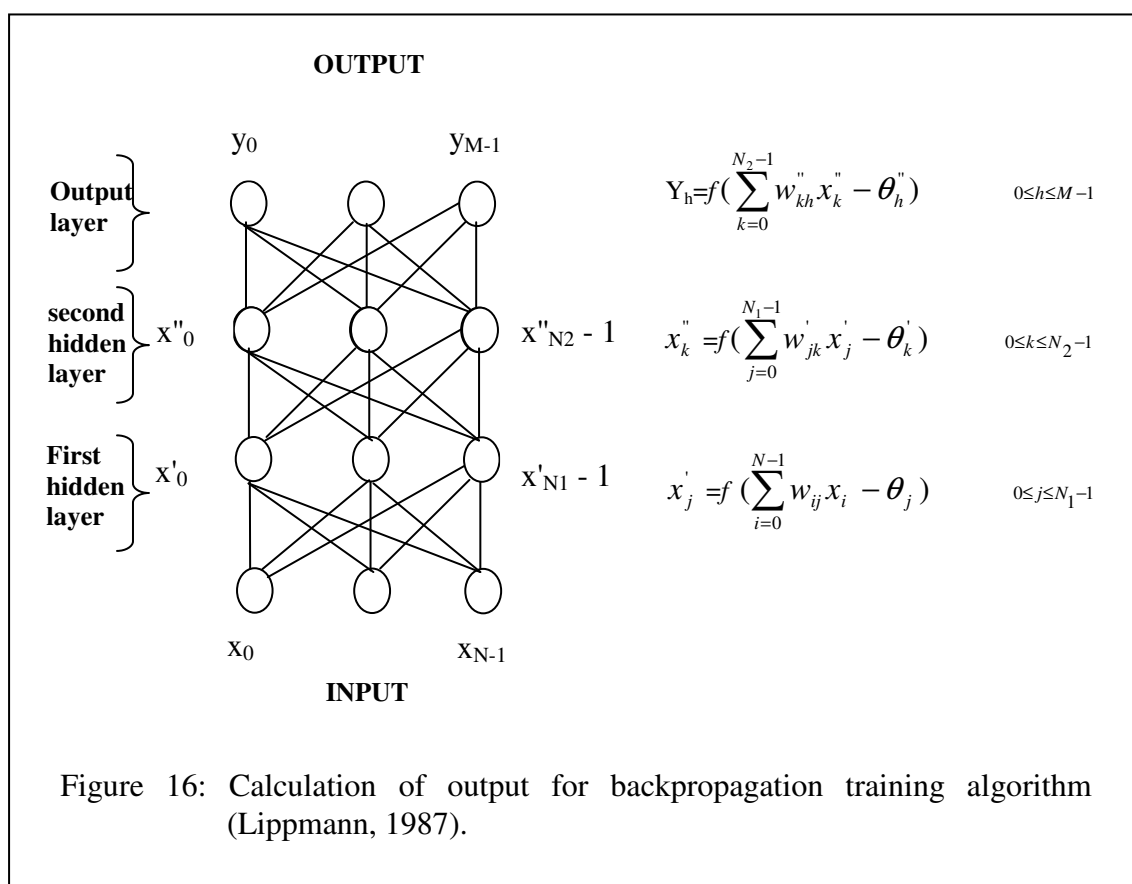
$$\delta_j = x_j' (1 - x_j') \sum_k \delta_k w_{jk},$$

where k is over all nodes in the layers above node j . Interval node thresholds are adapted in a similar manner by assuming they are connection weights on links from auxiliary constant-valued inputs. Convergence is sometimes faster if a momentum term is added and weight changes are smoothed by

$$W_{ij}(t+1) = w_{ij}(t) + \eta \delta_j x_i' + \alpha (w_{ij}(t) - w_{ij}(t-1)),$$

Where $0 < \alpha < 1$.

Step 5. Repeat by going to step 2.



2.6 Fuzzy Logic

Fuzzy Logic is a departure from classical two-valued sets and logic, that uses "soft" linguistic (e.g. large, hot, tall) system variables and a continuous range of truth values in the interval $[0,1]$, rather than strict binary (True or False) decisions and assignments.

Formally, fuzzy logic is a structured, model-free estimator that approximates a function through linguistic input/output associations. Fuzzy rule-based systems apply methods to solve many types of "real-world" problems, especially where a system is difficult to model, is controlled by a human operator or expert, or where ambiguity or vagueness is common. A typical fuzzy system consists of a rule base, membership functions, and an inference procedure (Bonde, 2000).

2.6.1 Classical sets and fuzzy sets

Definition: A set is a collection of objects.

A set may be defined by simply enumerating its elements or by stating the property that all such elements must hold. The former is called an *extensional* definition, whereas the latter is an *intensional* one. Generally, a *universal* set, sometimes denoted by U , is assumed, from which the set elements are chosen; further, a label, A , identifies a given set in compact form (Yen *et al.*, 1995).

EXAMPLE 1:

$$A = \{\text{red, green, blue}\}$$

In this example, the selection is made from a palette of colors (the universal set), which may be finite, or infinite. The colors in the palette, however, are assumed to represent clearly distinct objects. It is only then that the definition as it stands is a precise one; there are elements that clearly belong to the set—namely, *red*, *green*, and *blue*—and others that do not—namely, any other color in the palette.

An alternative approach to defining a set is via a statement that states the property that establishes elementhood, hence an intensional definition as in Example 2.

EXAMPLE 2:

Set of even numbers:

$$B = \{n \mid n \text{ is an integer divisible by two}\}$$

In Example 2, we do not enumerate all elements of B and yet the definition proper and precise (Yen *et al.*, 1995).

2.6.2 Membership functions

The membership function is a measure of the degree to which an element belongs to a fuzzy subset. When the membership function takes only values 0 and 1, the fuzzy set becomes a conventional set. Therefore, the conventional set theory is a special case of fuzzy set theory. The differences between conventional set theory and fuzzy set theory is known that for conventional set theory an element must belong to either a set or its complement set. In the case of a fuzzy subset, an element can simultaneously belong to several fuzzy subsets with some degree of truth (Wu, 1994).

As an example, consider a fuzzy set tall. Let the universe of discourse be heights from 40 inches to 90 inches. With a crisp set, all people with height 72 or more inches are considered tall, and all people with height of less than 72 inches are considered not tall. The crisp set membership function for set tall is shown in Figure 17. The corresponding fuzzy set with a smooth membership function is shown in Figure 18. The curve defines the transition from not tall and shows the degree of membership for a given height. We can extend this concept to multiple sets. If we consider a universe of discourse from 40 inches to 90 inches, then, to describe height, we can use three term values such as short, average, and tall.

In practice, the terms short, medium, and tall are not used in the strict sense. Instead, they imply a smooth transition. Fuzzy membership functions representing these sets are shown in Figure 19. The Figure shows that a person with height 65 inches will have membership value 1 for set medium, whereas a person with height 60 inches may be a member of the set short and also a member of the set medium; only the degree of membership varies with these sets (Kulkarni, 2001). There are many different types of membership function. Some famous ones are illustrated in Figure 20 (Tamimi, 1999).

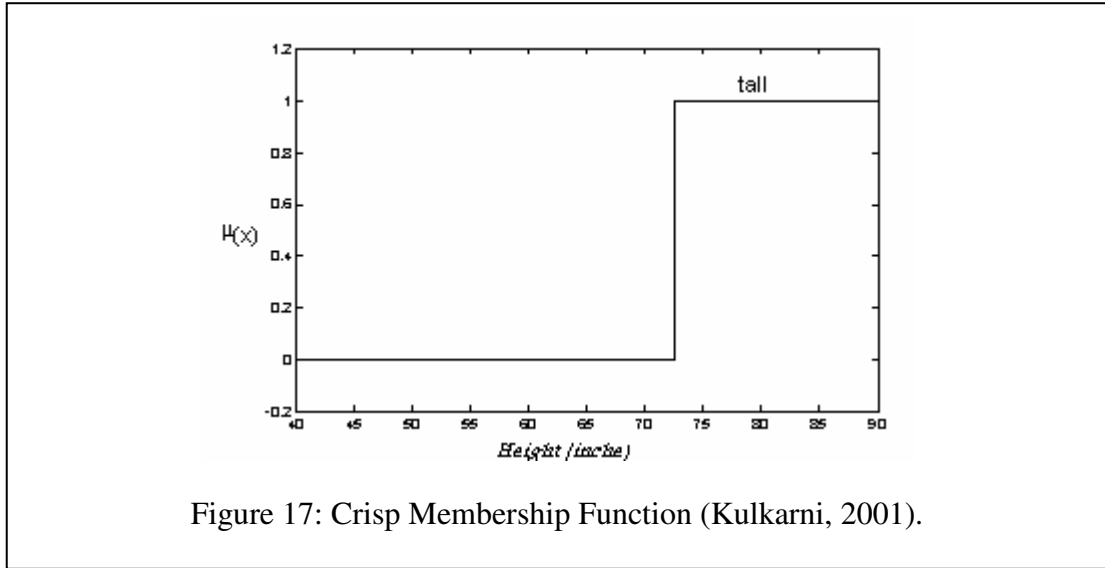


Figure 17: Crisp Membership Function (Kulkarni, 2001).

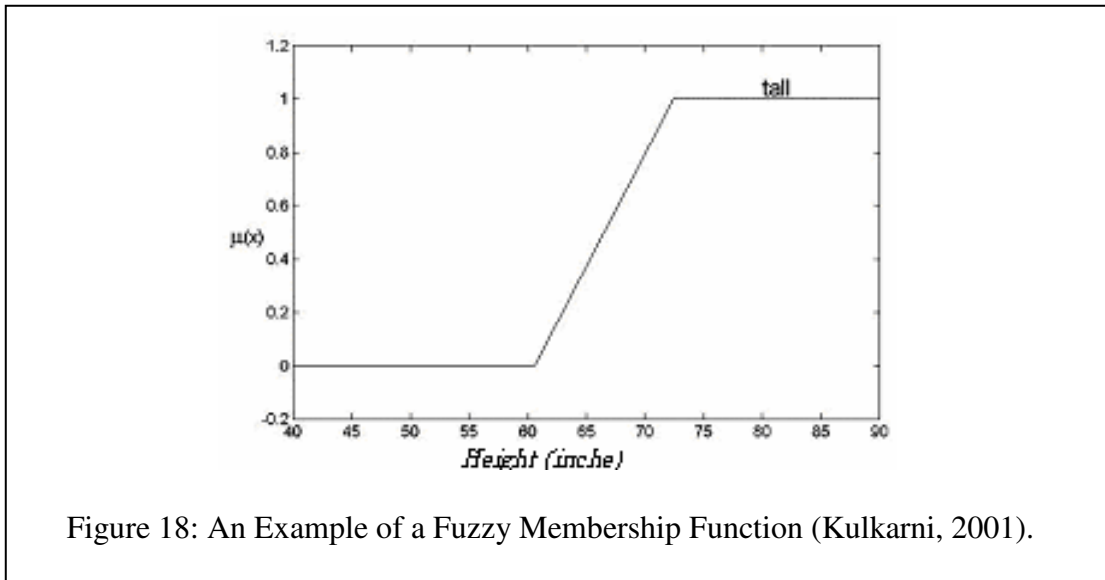


Figure 18: An Example of a Fuzzy Membership Function (Kulkarni, 2001).

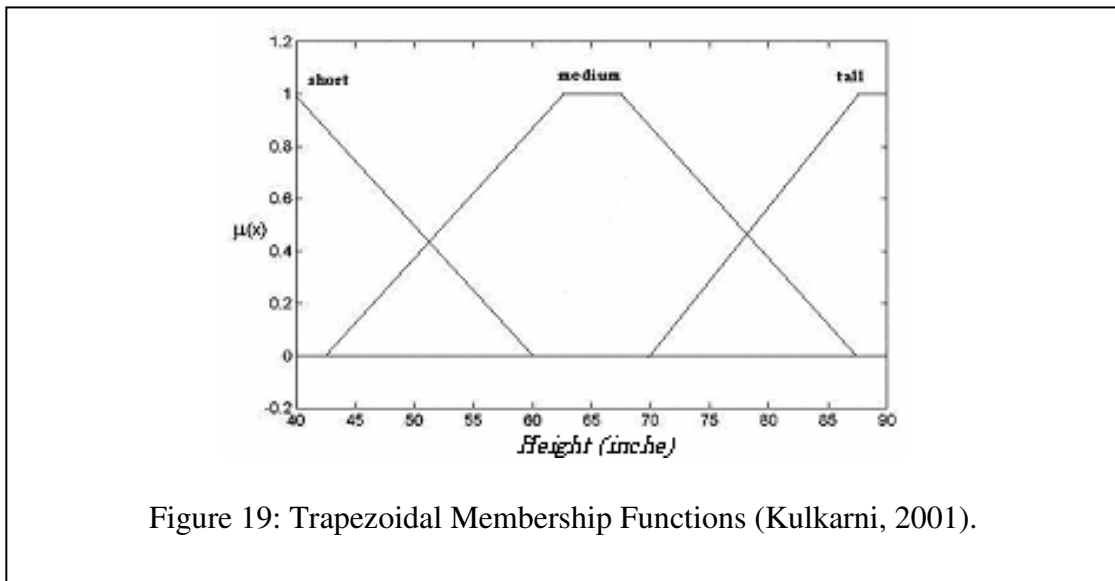
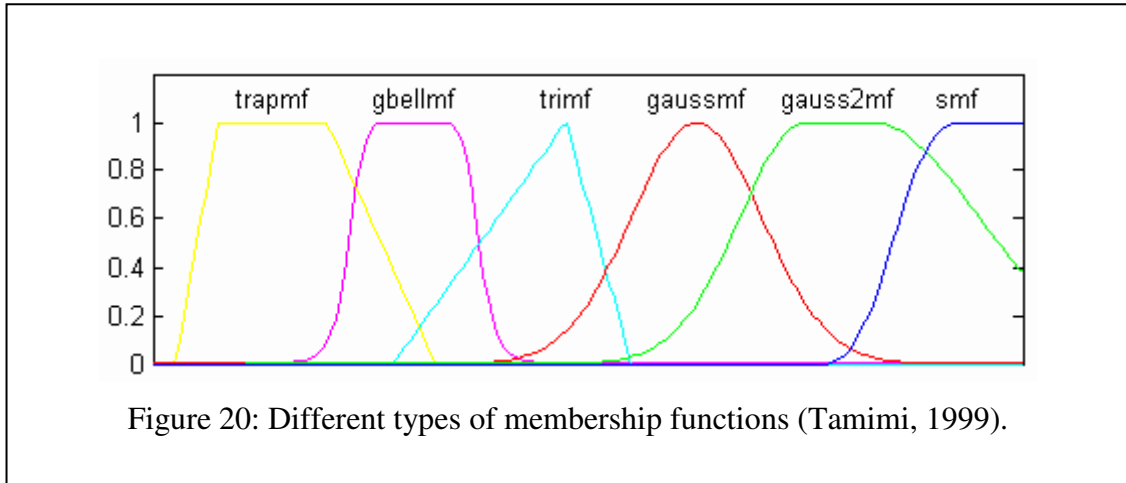


Figure 19: Trapezoidal Membership Functions (Kulkarni, 2001).



2.6.3 Fuzzy Set Operations

In this section, the union, intersection, and complement of fuzzy sets are reviewed. These can be derived by operation of their membership functions as defined in the following (Tanaka, 1997).

Union, intersection, and complement of fuzzy sets:

- Union of fuzzy sets A and B: $A \cup B$ of fuzzy sets A and B is a fuzzy set defined by the membership function:

$$\mu_{A \cup B}(x) = \mu_A(x) \vee \mu_B(x),$$

where

$$\mu_A(x) \vee \mu_B(x) = \begin{cases} \mu_A(x) & \text{for } \mu_A(x) \geq \mu_B(x) \\ \mu_B(x) & \text{for } \mu_A(x) < \mu_B(x); \end{cases}$$

The $\mu_A(x) \vee \mu_B(x)$ can be written as $\max \{ \mu_A(x), \mu_B(x) \}$.

- Intersection of fuzzy sets A and B: intersection $A \cap B$ of fuzzy sets A and B is a fuzzy set defined by the membership function:

$$\mu_{A \cap B}(x) = \mu_A(x) \wedge \mu_B(x),$$

where

$$\mu_A(x) \wedge \mu_B(x) = \begin{cases} \mu_A(x) & \text{for } \mu_A(x) \leq \mu_B(x) \\ \mu_B(x) & \text{for } \mu_A(x) > \mu_B(x); \end{cases}$$

The $\mu_A(x) \wedge \mu_B(x)$ can be written as $\min \{ \mu_A(x), \mu_B(x) \}$.

- Complement of fuzzy set A: complement \bar{A} of fuzzy set A is a fuzzy set defined by the membership function:

$$\mu_{\bar{A}}(x) = 1 - \mu_A(x).$$

2.6.4 Fuzzy System Components

A fuzzy system consists mainly of three main components: fuzzifier, defuzzifier and fuzzy rules.

A fuzzifier converts the inputs from real world values into fuzzy representations by the use of input membership functions.

A defuzzifier converts the output of the fuzzy process logic into a single value in the real world solution by the use of output membership functions.

Fuzzy Rules is a set of rules which determine which output or set of outputs should take account for a given input or set of inputs.

The general form of a fuzzy rule is:

$$\text{If } (INPUT_I \text{ is } VALUE_K) \text{ Then } (ACTION_J \text{ is } VALUE_L)$$

Where the $VALUE_K$ and $VALUE_L$ refer to the respective specific fuzzy sets associated with the particular $INPUT_I$ and $ACTION_J$.

The fuzzy components which had been illustrated in the following Figure (21) (Tamimi, 1999):

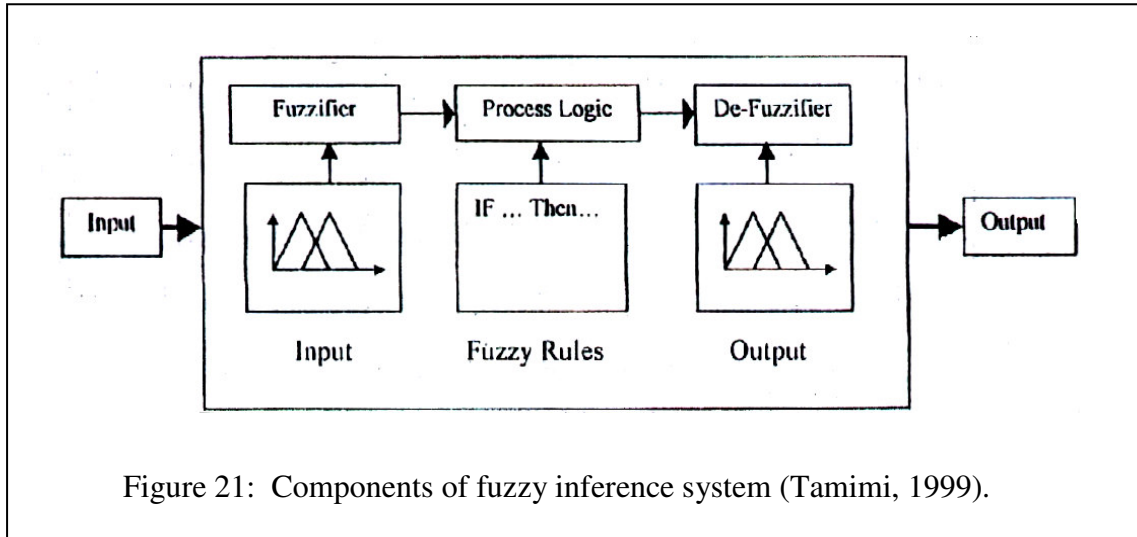


Figure 21: Components of fuzzy inference system (Tamimi, 1999).

2.6.5 Fuzzy If-Then Rules

Fuzzy if-then rules or fuzzy conditional statement are expressions of the form **IF A THEN B**, where **A** and **B** are labels of fuzzy sets characterized by appropriate membership function. Due to their concise form, fuzzy if-then rules are often employed to capture the imprecise modes of reasoning that play an essential role in the human ability to make decisions in an environment of uncertainty and imprecision. An example that describes a simple fact is

$$\underbrace{\text{If pressure is high}}_{\text{antecedent part}}, \text{ then } \underbrace{\text{volume is small}}_{\text{consequent part}}.$$

Where *pressure* and *volume* are *linguistic variables*, *high*, and *small* are *linguistic values* or *label* that are characterized by membership functions (Holec, 1997).

Another form of fuzzy if-then rule, proposed by Takagi and Sugeno (Holec, 1997), has fuzzy sets involved only in the premise part. By using Takagi and Sugeno's fuzzy if-then rule, we can describe the resistant force on a moving object as follow:

If velocity is high, then force = k * (velocity)²

Where, again, *high* in the premise part is a linguistic label characterized by an appropriate membership function. However, the consequent part is described by a nonfuzzy equation of the input variable, velocity.

Both types of fuzzy if-then rules have been used extensively in both modeling and control, through the use of linguistic labels in membership functions. A fuzzy if-then rule can easily capture the spirit of a "rule of thumb" used by humans. From another angle, due to the qualifiers on the premise parts, each fuzzy if-then rule can be viewed as a local description of the system under consideration. Fuzzy if-then rule forms a core part of the fuzzy inference system (Holec, 1997).

2.6.6 Fuzzy Reasoning

The steps for reasoning are as follows:

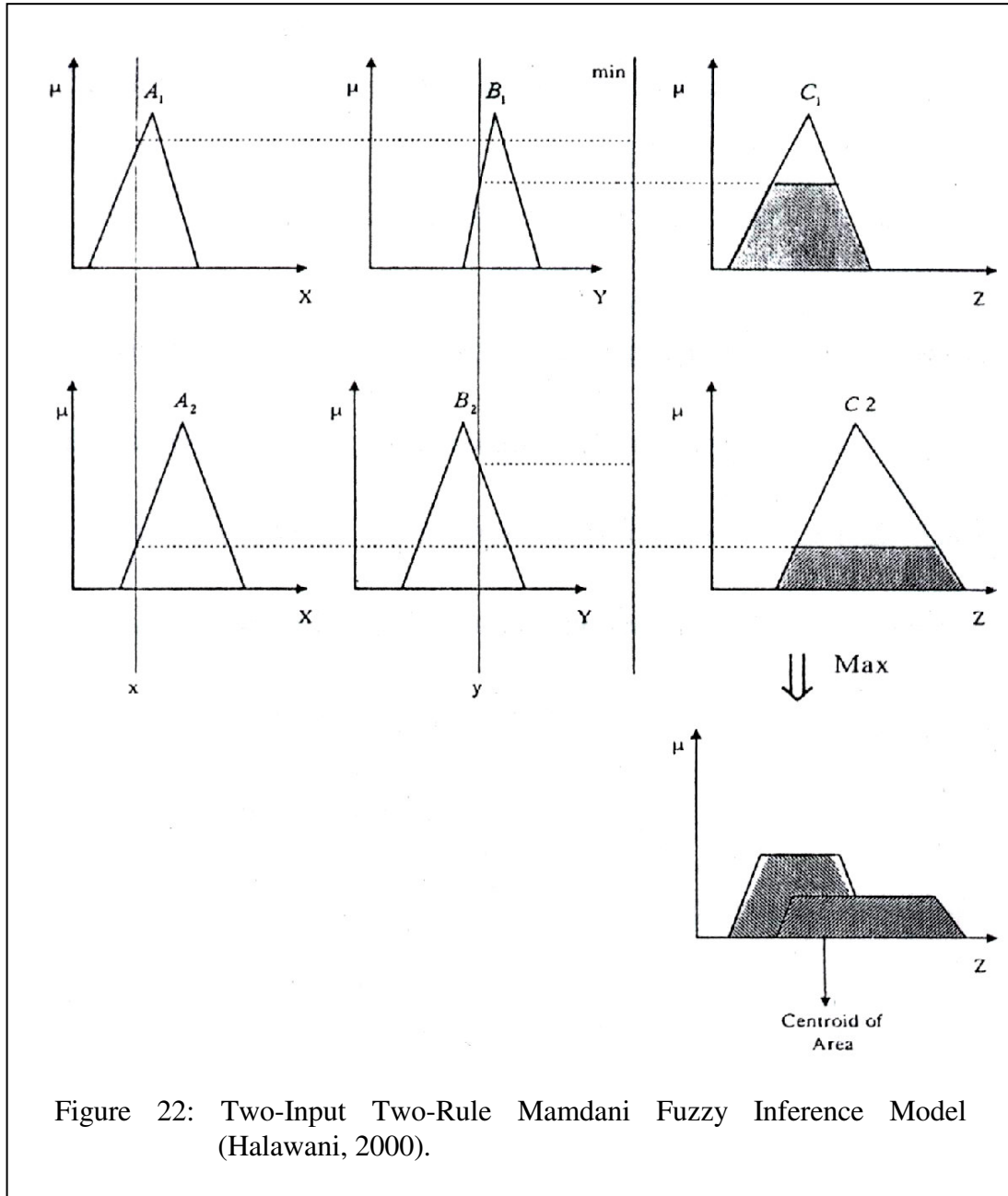
1. Comparing the input variable with the membership functions, so as to obtain a fuzzy value of each linguistic label (this is usually referred to as the fuzzification operation).
2. Combine the membership values to get the weight for each rule.
3. Depending on this weight, the value of the crisp output is evaluated (defuzzification) (Tamimi, 1999).

2.6.7 Type of fuzzy inference systems

In what follows, the three widely used types of fuzzy inference systems are introduced (Mamdani, Sugeno, and Tsukamoto). The basic differences between various models lie in the consequent of their fuzzy rules. Accordingly, the aggregation and defuzzification procedures of the three models are different (Halawani, 2000).

Mamdani Fuzzy Model:

This model was proposed by Mamdani in 1975 to control a steam engine and boiler combination. It was among the very first control systems built based on the theory of fuzzy logic. This model is characterized by the time consuming process of defuzzification, because there is a fuzzy set for each output variable that needs defuzzification. Figure 22 shows a two-input two-rule Mamdani fuzzy model. The two crisp inputs x , and y are defined on the two universes of discourse X and Y , respectively, while z represents the overall output. As mentioned before, the aggregation of the outputs of all the IF-THEN rules yields single fuzzy set (output). The defuzzification process is then performed to extract a crisp value from the output fuzzy set. In Figure 22, the *max* operator is used in the aggregation process (Halawani, 2000).



Sugeno Fuzzy Model:

This model was originally proposed by Takagi, Sugeno, and Kang (also known as TSK fuzzy model) (Halawani, 2000). This model has developed a systematic approach to generate fuzzy rules from the input-output data set. The basic difference between Sugeno fuzzy model and Mamdani fuzzy model is that the output of each rule in Sugeno model is a single crisp value. The over all output in this model is obtained by the

weighted average, explained later. Therefore, the time-consuming process of defuzzification required in Mamdani model is now greatly reduced. By far, for sample-data-based fuzzy modeling, the Sugeno model is the most popular candidate (Halawani, 2000).

The form of a typical fuzzy rule in the Sugeno model is:

$$\text{IF } x \text{ is } A \text{ and } y \text{ is } B \text{ THEN } z = f(x, y)$$

Where A and B are two fuzzy sets in the antecedent of the IF-THEN rule, and z is a crisp value in the consequent computed from the crisp function $f(x,y)$. The function $f(x,y)$ is a polynomial of first, second or higher order degree, or it can be constant. In the last case, the resulting fuzzy inference system is called zero-order model, while when the polynomial is of order one, the model is called first-order model.

The reasoning procedure used in the first order Sugeno fuzzy model is shown in Figure 23 (Halawani, 2000).

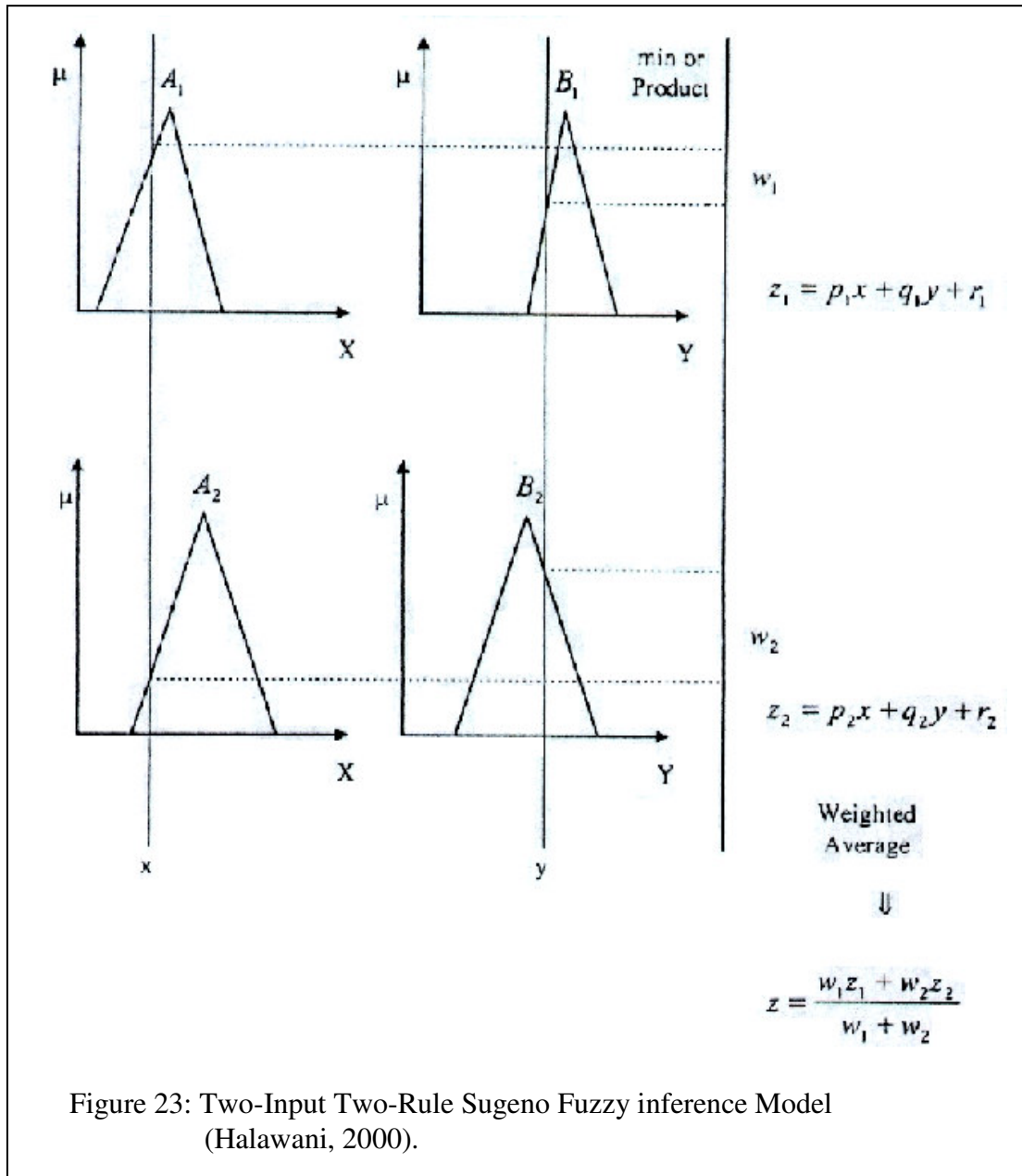
As mentioned above, a crisp value is obtained from each IF-THEN rule. This can be written as:

$$z_i = p_i x + q_i y + r_i$$

Where p_i , q_i and r_i are weights (constants) defined by the user based on the application. The overall output for the Sugeno fuzzy inference model is shown in Figure 23 and calculated using the *weighted average* method as follows:

$$z = \frac{w_1 z_1 + w_2 z_2}{w_1 + w_2}$$

Where w_i is the firing strength of the i^{th} rule (Halawani, 2000).



Tsukamoto Fuzzy Model:

In Tsukamoto fuzzy model, a fuzzy set with a monotonical membership function is used to represent the consequent of the IF-THEN rule.

As in Sugeno fuzzy model, the overall output is obtained as the weighted average of all rules' outputs. This model also avoids the time consuming process of defuzzification by applying the weighted average method to aggregate the outputs of all rules (Halawani, 2000).

2.7 Related Work

Sagar *et al.* (1995) presented approach is based on fuzzy logic techniques. This has the advantage of being simple and less expensive. With the feature extraction process investigated, a verification of fingerprint process is not complete without a proper matching process. The matching algorithm currently under evaluation has the detected minutiae points encoded in a compressed format initially and a fuzzy approximation theorem employed to match this encoded data with the fingerprint under test.

Al-Zewary (1996) designed a system that involved the following: (1) to identify fingerprint images, the system should firstly detect the edges of these images. The five edge detection techniques that are considered in his study were Sobel, Geuen, Prewitt, MarrHeldrith, and Fuzzy. A new edge detection technique was suggested as a result of his study. (2) The second step was to locate the core point, which is considered to be the essential feature in identifying fingerprint images. A new technique was suggested for locating the core point after selecting manually a region that involves the core point to cover all the most inner curves in the fingerprint image.

Sun and Ai (1996) presented a technique for the recognition of fingerprint images. Fingerprint images are preprocessed by using a set of contextual filters first and then thinned to extract the fingerprint “minutiae” features. As the images are noisy, there are many spurious minutiae. These spurious minutiae can be detected efficiently by a proposed method. The method for matching fingerprint images is also given. The usage of the technique allows for fairly high degree of noise tolerance.

Amengual *et al.* (1997) presented a real-time feature extraction method for fingerprint images, which not only takes advantage of adapting some concepts found in the literature but also introduces new ideas and procedures. An adequate tradeoff between the speed and the accuracy of the process can be obtained tuning up the

parameters, which control the operation performed in each of the implemented modules.

Kasaei *et al.* (1997) considered fingerprint as sample images from non-stationary processes with flow patterns, they proposed here a robust technique to extract their features. The unique properties of fingerprint textures are used to enhance the images and improve the fidelity of their features. The ridges are extracted from enhanced foreground area based on local dominant ridge directions. The resulting bit-mapped images are thinned and smoothed to detect structural features. A large number of false features are eliminated in the proposed post-processing stage.

Wilson *et al.* (1997) presented results on direct optical matching of inked and real-time fingerprint images. Direct optical correlations and hybrid optical neural network correlation are used in the matching system for inked fingerprints. Preliminary results on optical matching of real-time fingerprints use optical correlation. Tests of typical cross correlations and auto correlation sensitivity for both binary and 8 bit gray images are presented. When global correlations are tested on a second inked image, results found to be strongly influenced by plastic distortion of the finger.

Maio and Maltoni (1997) proposed a technique based on ridgeline following, where the minutiae are extracted directly from gray-scale images. They choose to extract the features directly from the gray-scale image (without binarization and thinning). The basic idea of their method is to follow the ridgelines on the gray-scale image by “sailing” according to the fingerprint directional image. A set of starting point is determined by superimposing a square-meshed grid on the gray-scale image. For each starting point, the algorithm keeps following the ridgelines until they terminate or intersect other ridgelines (minutiae detection). A labeling strategy is adopted to examine each ridgeline only once and locate the intersections among ridgelines. The results showed that the method performs better results both in terms of robustness and

efficiency.

Jain and Ross (1998) developed a fingerprint-mosaicking scheme that constructs a composite fingerprint template using multiple impressions. A composite template reduces storage, improves matching time and alleviates the problem of template selection. Furthermore, they proposed an algorithm in which two impressions (templates) of a finger are initially aligned using the corresponding minutiae points.

Hong (1998) designed a fingerprint-based biometric system, which is capable of achieving a fully automatic “positive personal identification” with a high level of confidence. He has identified and explored the following issues: (1) feature extraction (2) image enhancement, (3) minutiae matching, (4) integration of multiple biometrics to improve the performance of a biometric system by combining several biometrics (e.g. fingerprint, face, speech, etc.), and (5) fingerprint classification.

Jiang *et al.* (1999) presented an improved approach of minutiae detection that adaptively traces the gray level ridge of the original fingerprint image with piece wise linear lines of different length. While tracing the ridges the fingerprint image is smoothed with an oriented smoothing filter only at selected pixels where smoothing is necessary. After tracing all the ridges, a piece-wise linear skeleton image is obtained. Each ridge in the skeleton is labeled with a number so that each minutiae is associated with one or two ridge numbers. The post-processing is based not only on the skeleton relationship of the minutiae, but also the associated ridge relationship and the certainty level of the minutiae.

Lee and Wang (1999) developed a one-step method using Gabor filters for directly extracting fingerprint features from gray-level images for a small-scale fingerprint recognition system. From experimental results, the use of magnitude Gabor features with eight orientations as fingerprint features lead to good shift-invariant properties and

an accuracy of 97.2% with 3-nearest neighbor classifiers.

Sagar and Beng (1999) proposed the fusion of fuzzy logic and neural network technology in automated fingerprint recognition for the extraction of important fingerprint features, also known as minutiae. The results showed that of average: the fuzzy neural approach is a better alternative. The hybrid fuzzy and neural network model performs the minutiae extraction in two stages, a fuzzy front-end and a neural back-end.

Ammar *et al.* (2000) produced a research in which a web-based test bed is developed to assist the experiments in Fingerprint Image Comparison (FIC). The FIC function, which supported the comparison between submitted fingerprints and candidate fingerprint, is one of the specified functions of the US FBI integrated automated fingerprint identification system (IAFIS). The FIC test bed provides an integrated environment to simplify the testing process and collects information from test results. A 3-tiered architecture is used in the test bed design, which is composed of test client, test bed server and fingerprint image database.

Bhanu *et al.* (2000) developed a novel approach for extraction of minutiae features from fingerprint images. The proposed approach is based on the use of logical templates for minutiae extraction in the presence of data distortion. A logical template is an expression that is applied to the binary ridge (valley) image at selected potential locations to detect the presence of minutiae at these locations.

Huvanandana *et al.* (2000) described an on-line fingerprint identification system consisting of fingerprint image preprocessing, feature extraction and matching. The preprocessing part includes image enhancing steps to acquire binarized and skeletonized ridges. The fast robust matching is proposed, which is a correlation-based method that runs over the 1/8 sized feature points maps.

Liu *et al.* (2000) proposed a new method to extract the minutiae from the gray-level fingerprint image is proposed. The method is based on tracking the relationship of the ridges and furrows in the fingerprint images. The minutiae are detected by finding the relationship change during the tracking. Experimental results demonstrate that the method developed based on the new approach is insensitive to noise and hence extract minutiae better with high accuracy.

Luo *et al.* (2000) introduced an algorithm to deal with the problem of minutiae matching in fingerprint verification. They used a method that is simple and effective to do fingerprint image alignment. In addition, they introduced ridge information into the process of matching of and used a changeable sized box in the matching process, all the above makes the algorithm more able to distinguish images from different fingers and can deal with the nonlinear deformation more robustly.

Espinosa-Duro (2001) describes the design and implementation of a minutiae extraction algorithm for fingerprint identification systems. She presented a high performance minutiae extractor algorithm for fingerprint recognition using the two databases acquired. Anyway, for solving the final identification task, many other problems will have to be handled, and primarily, how to extract a clear and reliable ridge structure from a poor quality image.

Jain *et al.* (2001) produced a hybrid-matching algorithm that uses both minutiae (point) information and texture (region) information for fingerprints matching. Results obtained on the used database show that a combination of the texture-based and minutiae-based matching scores leads to a substantial improvement in the overall matching performance.

Jiangtao and Shaojun (2001) presented a new approach to automatic fingerprint identification system for small sized database, such as personal information security.

Using correlation coefficients is one possible solution for the fingerprint identification for a small size of database. Although the algorithm based on this is not suitable to software implement, it is quite suitable to pure hardware implement.

Jung and Park (2001) implemented a robust fingerprint identification system using hybrid methods. In the feature extraction stage, the average ridge line width is estimated in the ridge line following algorithm, and a new stopping criterion is proposed. A fuzzy technique is applied to the features, and quantitative feature values are evaluated using the weighted feature value window. In the feature matching stage, the patterns of ridge lines in the fingerprint image are analyzed and an adaptive matching boundary is constructed to eliminate false features. The weighted matching score is computed using the quantitative feature value for more reliable matching. A two-step estimation of transformation parameters is also employed to reduce the computation time.

Prabhakar (2001) estimated theoretically the probability of a false corresponding between two fingerprints from different fingers, based on the minutiae representation of fingerprints. He presented a novel filterbank-based for fingerprints, and used this compact representation for fingerprint classification as well as fingerprint verification. Experimental results showed that this algorithm competes well with the state-of-the-art minutiae-based matches. Moreover, he developed a decision level information fusion framework, which improve the fingerprint verification accuracy as well as multiple matches, multiple fingers of the user, or if multiple impressions of the same finger are combined.

Simon-Zorita *et al.* (2001) presented a complete minutiae extraction scheme for automatic fingerprint recognition system. The proposed method uses improving alternatives for the image enhancement process, leading consequently to an increase of the reliability in the minutiae extraction task. Evaluation results are obtained from both

inked and scanned fingerprints. Conclusions in terms of Goodness Index (GI), which compares the results obtained by automatic minutiae extraction with manually extracted ones, are provided in order to test the global performance of this approach.

Jain and Ross (2002) demonstrated that the performance of multibiometric system can be further improved by learning user-specific parameters. Two types of parameters were considered: (1) thresholds that are used to decide whether a matching score indicates a genuine user or an imposter, and (2) weights that are used to indicate the importance of matching scores output by each biometric trait.

Pankanti *et al.* (2002) addressed the problem of fingerprint individuality by quantifying the amount of information available in minutiae features to establish a correspondence between two fingerprint images. They derive an expression, which estimates the probability of a false correspondence between minutiae-based representations from two arbitrary fingerprints belonging to different fingers.

Prabhakar *et al.* (2002) proposed a feature refinement stage followed by a feedforward of the original grayscale image data to a feature verification stage in the context of a minutiae-based fingerprint verification system. They showed that a feature refinement stage that assigns one of two class labels to each detected minutiae (ridge ending and ridge bifurcation), which can improve the matching performance by~1%.

Ross *et al.* (2002) described a hybrid fingerprint matching scheme that uses both minutiae and ridge flow information to represent and match fingerprints. A set of 8 Gabor filters, whose spatial frequencies correspond to the average inter-ridge spacing in fingerprints, is used to capture the ridges strength at equally spaced orientations.

From the above researches, it looks like that the major achievement in fingerprint identification are done by (Sagar et al., 1995) , (Sagar et al. 1999) , (Jung and park, 2001) and (Al-zewary, 1996).

In the presented literature, many preprocessing operations such as binarization, thinning, edge detection and other algorithms were used. These preprocessing operations cause missing some important information, take time for processing, and take space of memory in some it.

They depended mainly on fingerprint identification and recognition on the features (or minutiae) existing in fingerprint images such as ridge ending, bifurcation and core point. Any missing in this features will be incomplete matching process, consequently the fingerprint identification and recognition process will occur incorrectly.

The study is essential because it works in order to be the powerful identification process and it occurs correctly, reducing the time and reducing the size of stored weights matrices.

This study is important because:

1. There is still need for technology for fingerprint recognition.
2. The advantages of both the fuzzy logic and neural networks techniques together.
3. In the fingerprint identification process, it depends on texture of image – no one of above researchers used this technique – not used the features in fingerprint image and preprocessing to avoid the above mentioned disadvantages.

3. THE FUZZY LOGIC AND BACKPROPAGATION NEURAL NETWORK TECHNIQUE FOR IDENTIFICATION FINGERPRINT IMAGES

3.1 Introduction

In this chapter, we will present and explain the proposed method within certain algorithms. The proposed technique uses texture of fingerprint image and it is based on the use of fuzzy logic and backpropagation neural network.

3.2 The proposed method

The proposed method consists of two stages: the learning stage and the convergence and identification stage.

In the proposed method, we will use the smallest size of the net, which contains two nodes of each: input layer, hidden layer and output layer as shown in Figure 24. The size of this net will be the same no matter what size is the used fingerprint image, (16x16, 50x50, 128x128 or 256x256 etc). This representation of the net including the learning and convergence processes of image parts – and not necessary all - will perform the task of learning and convergence in a better way.

In addition, the use of energy function in the convergence and identification stage of proposed method will add traits to the use of backpropagation algorithm.

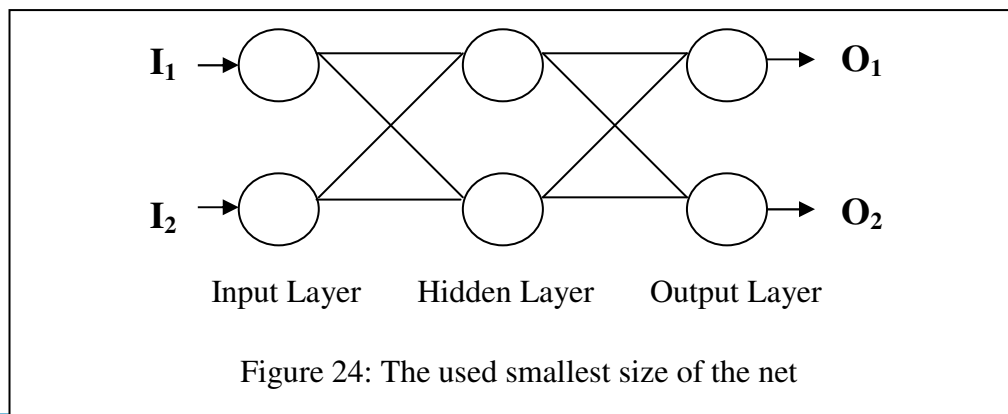


Figure 24: The used smallest size of the net

3.2.1 The learning stage

The learning stage consists of two algorithms, the algorithm of creating fuzzy membership functions and the algorithm of learning.

3.2.1.1 The Algorithm of creating fuzzy membership functions

In this algorithm, we will discuss the extracting average of middle region in the fingerprint image (avmid) and average of border region in the fingerprint image (avbor) as shown in Figure 25.

The smallest value of (avmid) and (avbor) is the first of fuzzy graph. It represents the fuzzy DARK membership function and the largest value is the second of fuzzy graph. It represents the fuzzy BRIGHT membership function, as shown in Figure 25.

The proposed algorithm of creating fuzzy membership function is presented by the following algorithm:

Step 1:

Set nm = 0; {counter for pixels in middle}

Set nb = 0; {counter for pixels in middle}

For x=1 to N {N: length of image}

 For y=1 to M {M: width of image}

 IF I(x,y) is within middle region of image

 Begin

 sm = sm + I(x,y);

 nm = nm + 1;

 End

IF $I(x,y)$ is within border region of image

Begin

$$sb = sb + I(x,y);$$

$$nb = nb + 1;$$

End

End

End

Step 2:

$$avmid = sm / nm; \quad \{avmid: \text{middle average}\}$$

$$avbor = sb / nb; \quad \{avbor: \text{border average}\}$$

Step 3:

$$\text{Min fuzzy} = \min (avmid, avbor)$$

$$\text{Max fuzzy} = \max (avmid, avbor)$$

The smallest value of (avmid) and (avbor) is the first of fuzzy graph. It represents the fuzzy DARK membership function. The largest value is the second of fuzzy graph. It represents fuzzy BRIGHT membership function.

An Example for algorithm of creating fuzzy membership functions:

In this example, we will show the process of creating fuzzy membership functions. First, extract average of middle region and average of border region as shown in Figure 25:

$$sb = 102 + \dots + 174 = 6035;$$

$$nb = 48;$$

$$avbor = sb / nb;$$

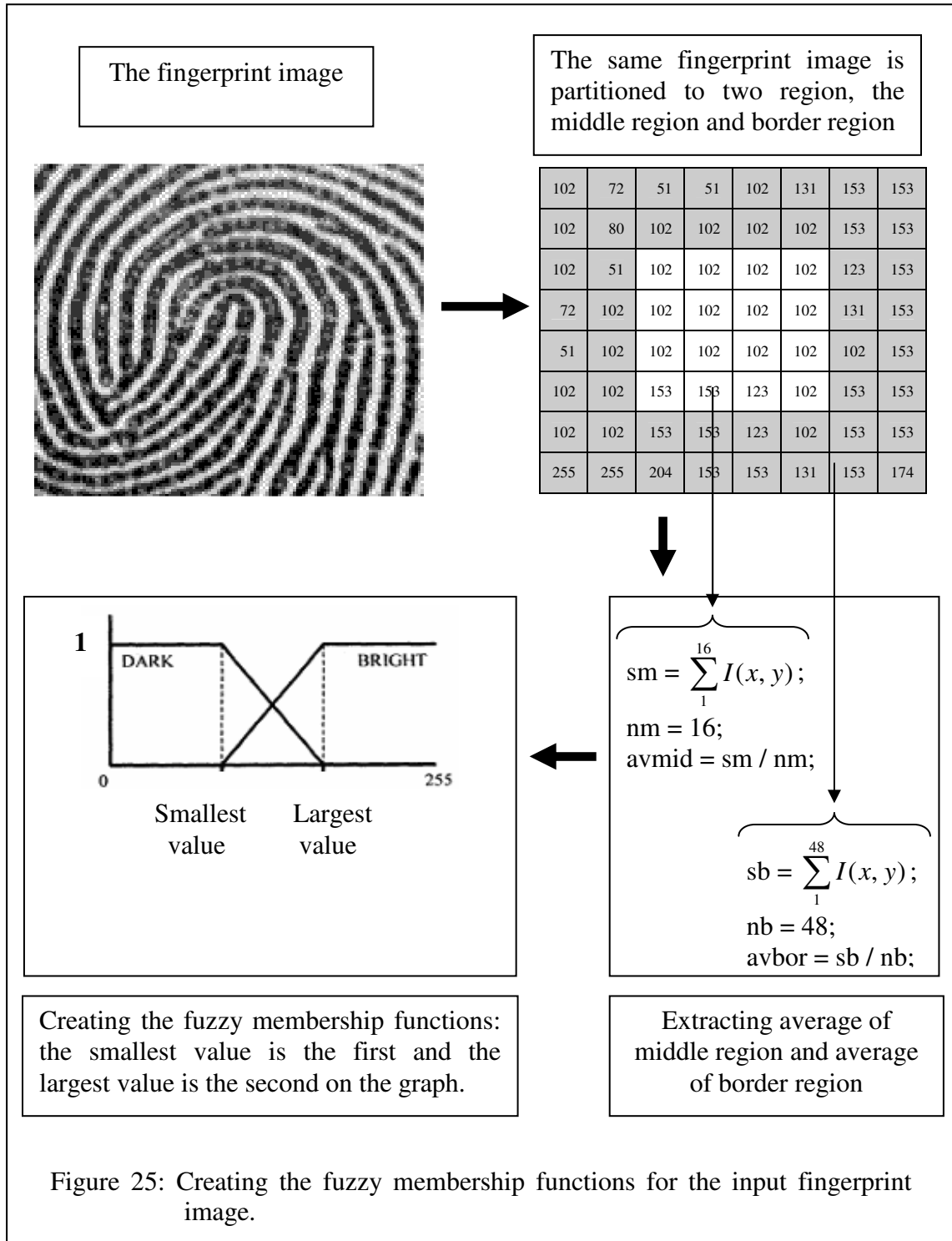
$$avbor = 6035/48 = \mathbf{125.7292};$$

$$sm = 102 + \dots + 102 = 1755;$$

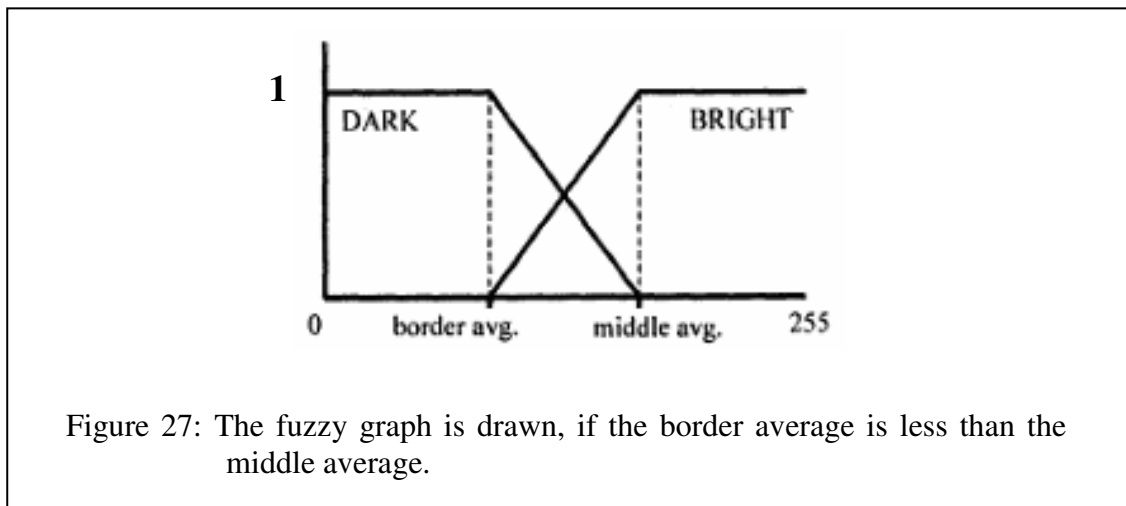
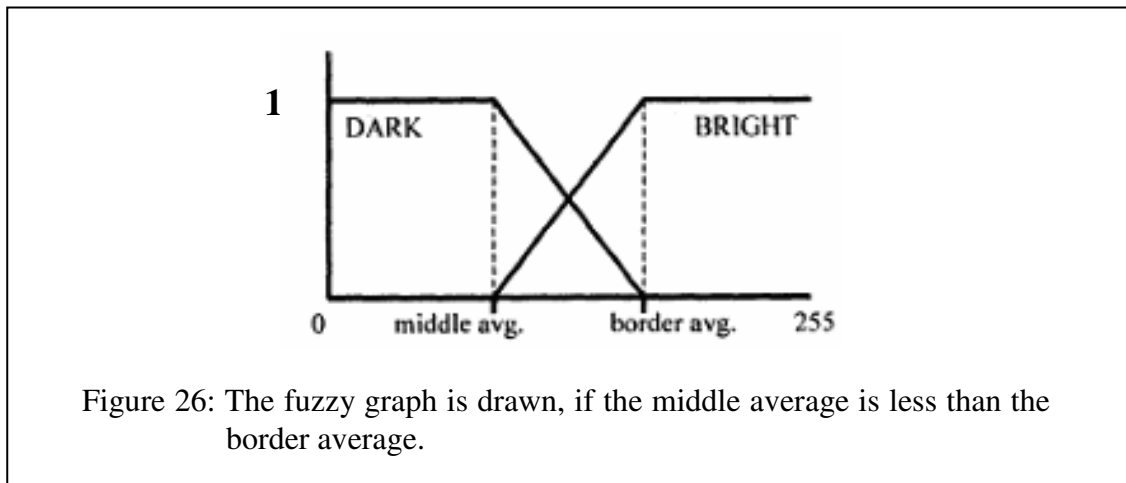
$$nm = 16;$$

$$avmid = sm / nm;$$

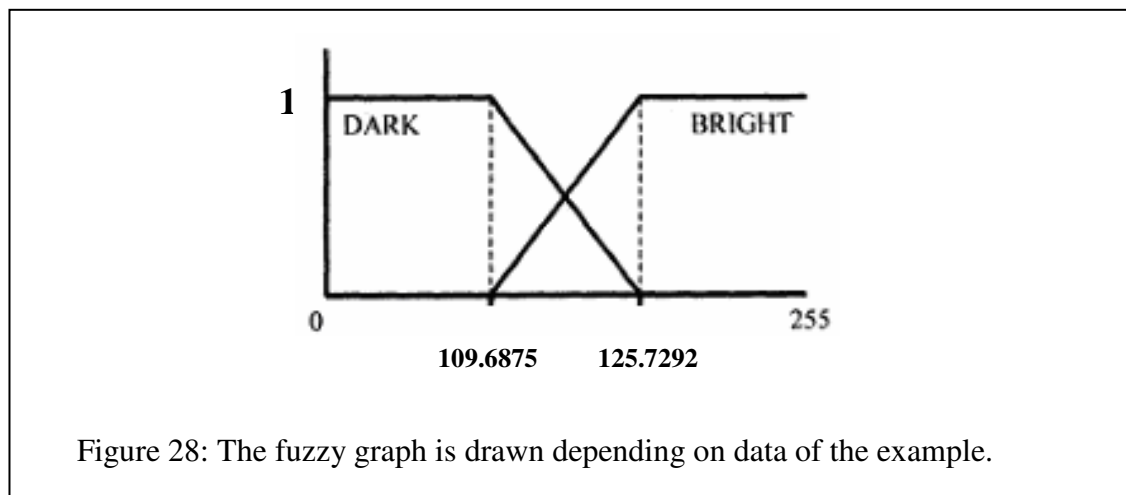
$$avmid = 1755 / 16 = 109.6875;$$



As shown in Figure 25, the fuzzy membership functions are drawn depending on av_{mid} and av_{bor} values, as shown in Figure 26 and Figure 27:



For the above data, $av_{mid} = 109.6875$ and $av_{bor} = 125.7292$, the fuzzy membership functions are drawn, as shown in Figure 28:



3.2.1.2 The Learning algorithm

In this algorithm, we will divide the fingerprint image into rows. Each row is divided into a number of parts of two pixels size, to be considered as a vector (V). This means there are four states for these vectors as shown in Figure 29.

1	2	3	4
0	0	1	1
0	1	0	1

Figure 29: The four states for the vectors.

Therefore, the learning process in backpropagation algorithm for any image will result in four sets of learning weights matrices as a maximum. Each set consists of two matrices: W1 and W2. The W1 represents the weights from input layer to hidden layer, while the W2 represents the weights from hidden layer to output layer. There are four sets of biases arrays as a maximum. Each set consists of two arrays, which are called Wh and Wo. The Wh represents the biases of hidden layer nodes, and Wo represents the biases of output layer nodes. The size of each matrix (W1 and W2) is 2x2, while the size of each array (Wh and Wo) is 2x1.

Hence, each vector V, as shown in Figure 29, will represent its own net. Each net has (W1, W2, Wh and Wo). Therefore, we have only four nets as shown in Figure 30.

We will replace each vector V by a number. This means we will replace all the parts (the vectors) in the image with a number, which represents the number of the net, and by this number we can call the net again.

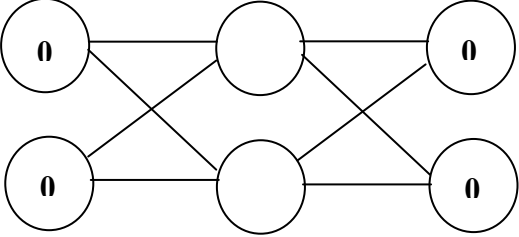
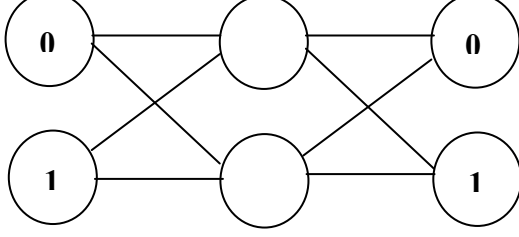
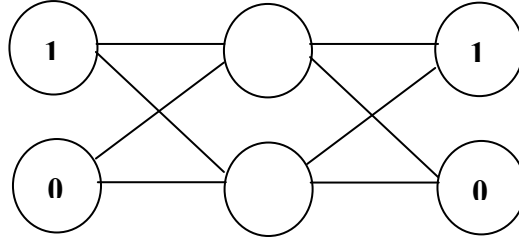
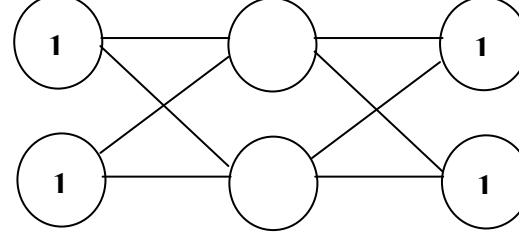
The vector (V)	The net	The number of net
0 0		0
0 1		1
1 0		2
1 1		3

Figure 30: The four vectors with their own nets and numbers of nets.

The learning process of the proposed method for all vectors in the fingerprint image will pass through three steps:

Converting the digit vector to binary vector:

By fuzzy membership functions obtained from the algorithm of creating fuzzy

membership functions and using fuzzification operation, you can convert the digit vector to binary vector, depending on the following fuzzy rules:

IF p1 is Dark **AND** p2 is Dark **THEN** vector = [0,0]

IF p1 is Dark **AND** p2 is Bright **THEN** vector = [0,1]

IF p1 is Bright **AND** p2 is Dark **THEN** vector = [1,0]

IF p1 is Bright **AND** p2 is Bright **THEN** vector = [1,1]

Testing whether the system has learned current vector before or not:

The produced binary vector is examined. If it is saved with its learning weights matrices, then it is learned. Therefore, we will replace this vector by the number of its own net; otherwise the vector is not learned.

Learning the vector:

If the vector is not learned, it is entered to the learning of backpropagation algorithm, then save its learning weights matrices, and save the number of net instead of vector.

The learning algorithm is presented by the following algorithm:

Step1: For x=1 to N {N: length of image}

Step2: For y=1 to M step 2 {M: width of image}

Step3:

Converting the value of pixels $I(x,y)$ & $I(x,y+1)$ to binary values using fuzzification operation and depending on the following fuzzy rules:

IF p1 is Dark **AND** p2 is Dark **THEN** vector = [0,0]

IF p1 is Dark **AND** p2 is Bright **THEN** vector = [0,1]

IF p1 is Bright **AND** p2 is Dark **THEN** vector = [1,0]

IF p1 is Bright **AND** p2 is Bright **THEN** vector = [1,1]

Step4:

IF produced vector is learned

Then Save the number of net for this vector and return to take the other vector.

Else enter the vector to the learning of backpropagation algorithm, save learning weights matrices $W1$ & $W2$, and save number of net for this vector.

3.2.2 The convergence and identification stage

The convergence and identification stage consist of three algorithms. Firstly, the algorithm of creating fuzzy membership functions, secondly the algorithm of identification, and thirdly the algorithm of convergence.

3.2.2.1 The Algorithm of creating fuzzy membership functions

This algorithm is as same as the one mentioned in section 3.2.1.1.

3.2.2.2 The identification algorithm

By using the identification algorithm, we will extract the nearest fingerprint image for the unknown input fingerprint image using energy function (based on Hopfield Neural Network (Kulkarni, 2001)) and depending on the learning weight matrix $W2$ for both vectors of unknown input fingerprint image and vectors of fingerprint images in the database.

Notice that the learning weight matrix $W1$ is similar than some of its parameters. Therefore, we use only the learning weight matrix $W2$. This similarity makes the identification process not active.

The identification process is showed in Figure 31.

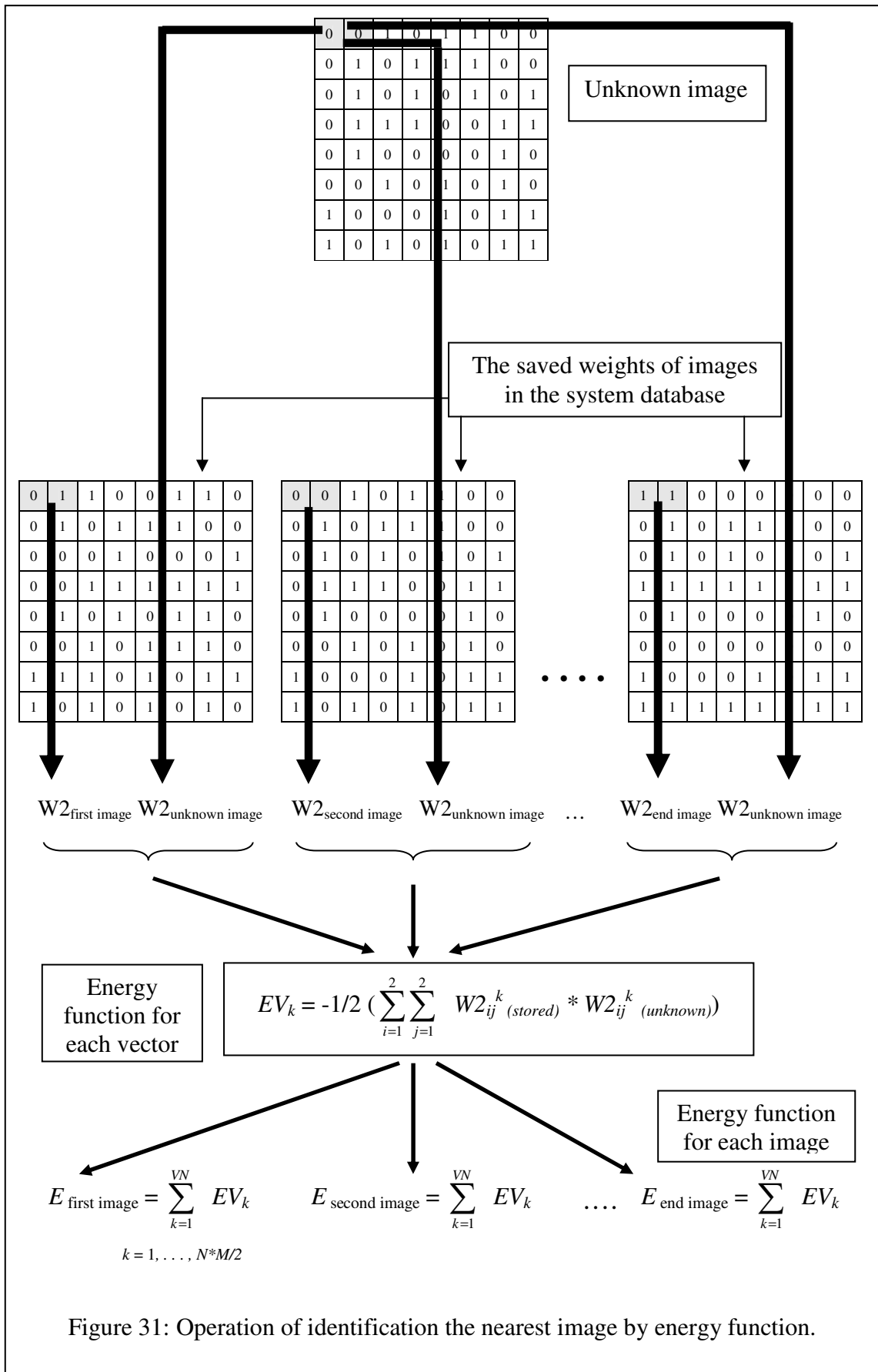


Figure 31: Operation of identification the nearest image by energy function.

The identification process consists of the following steps:

Use fuzzy membership functions obtained from the algorithm of creating fuzzy membership functions and fuzzification operation, to convert digital vector to binary vector depending on the fuzzy rules.

Use the learning weights matrix $W2$ for current vector in the input unknown fingerprint image with each of the learning weights matrix $W2$ for the vectors in the fingerprint images saved in the database, and extract the Energy function between them (EV). Thus, each fingerprint image in the database will have value of its own Energy function (E). Therefore, extract the value of the first minimum small energy function and the value of the second minimum Energy function. Then, extract the value of difference between them (Difference Next Close (DNC)).

Examine whether the system has learned the unknown input fingerprint image before or not using DNC value. If it is less than (-40), then the unknown input fingerprint image is learned. Therefore, the stored image with the minimum energy function representing the nearest image for the unknown input fingerprint image is converged. But If it is more than or equal to (-40), then the unknown input fingerprint image is not learned. Therefore, exit from the system.

The identification algorithm is presented by the following algorithm:

For $x=1$ to N { N : length of image}

For $y=1$ to M { M : width of image}

BEGIN

Convert the value of pixels $I(x,y)$ and $I(x,y+1)$ to binary values using fuzzification operation and depending on the following fuzzy rules:

IF $p1$ is Dark **AND** $p2$ is Dark **THEN** vector = [0,0]

IF p1 is Dark **AND** p2 is Bright **THEN** vector = [0,1]

IF p1 is Bright **AND** p2 is Dark **THEN** vector = [1,0]

IF p1 is Bright **AND** p2 is Bright **THEN** vector = [1,1]

For all stored images do

Begin

For all rows of stored images do

For the vector in the input fingerprint image and learning weight matrix W2 do

Begin

$$EV_k = -1/2 \left(\sum_{i=1}^2 \sum_{j=1}^2 W_{ij}^k (input) * W_{ij}^k (stored) \right) \quad \{Energy\ function\ for\ each\ vector\}$$

End;

$$E = \sum_{k=1}^{VN} EV_k \quad \{Energy\ function\ for\ each\ image\}$$

End;

END

IF DNC < -40

Then unknown input fingerprint image is learned; therefore the stored image with the minimum Energy function (E) representing the nearest image for the unknown input fingerprint image is converged.

IF DNC >= -40

Then unknown input fingerprint image is not learned, therefore exit from the system.

3.2.2.3 The Convergence Algorithm

In this algorithm, if unknown input fingerprint image is learned, then converge the stored image with the minimum energy function (E). This is the nearest one for the unknown input fingerprint image. In the same way of the learning algorithm, we divide

the fingerprint image into rows. Each row is divided into number of parts of two pixels size to be considered as a vector (V). Depending on the number of net and the backpropagation convergence algorithm, the vector (V) is converged. If the vector is converged by the backpropagation convergence algorithm before, it is converged using number of net.

The convergence algorithm is presented by following algorithm:

```

For x=1 to N           {N: length of image}
  For y=1 to M       step 2   {M: width of image}
    For i=1 to NC     {NC: the counter of the array (NoNet) is contained n*m/2
                        of numbers of net saved instead of the vectors}

      IF the current vector is not converged by the backpropagation convergence
        algorithm before,

        Then enter the current vector into the backpropagation convergence algorithm
          and save the produced vector into the array of Converged Image (CI).

           $CI(x,y) = V(1,1)$ 
           $CI(x,y+1) = V(1,2)$ 

        Else the vector is obtained by the number of net. Then, save the produced vector
          into the (CI).

           $CI(x,y) = V(1,1)$ 
           $CI(x,y+1) = V(1,2)$ 

    End;
  End;
End;

```

4. EXPERIMENTAL RESULTS AND DISCUSSION

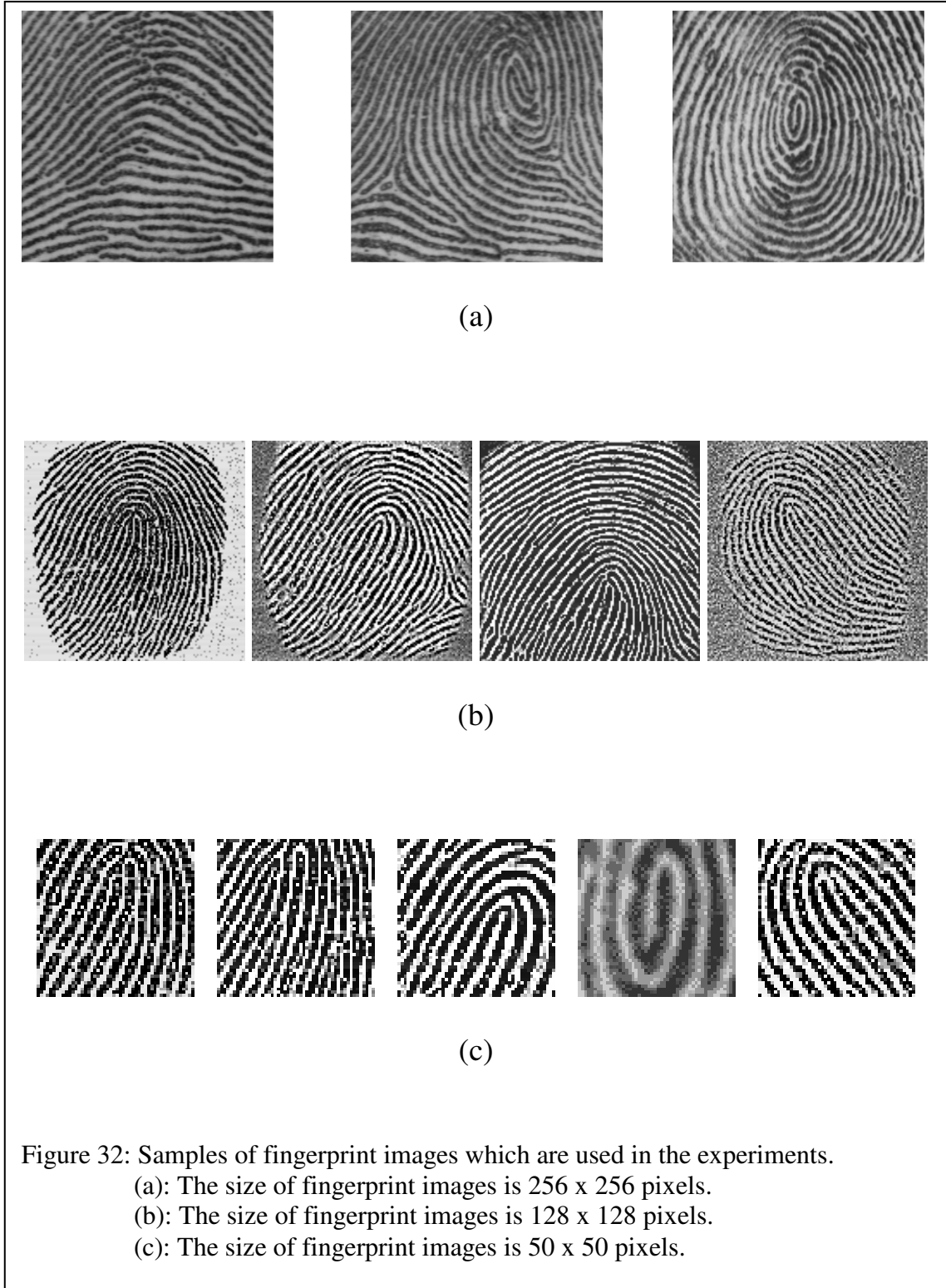
4.1 Introduction

This chapter will be devoted to demonstrate the implementation of the proposed system for fingerprint identification with experimental results and discussions. The proposed system and its experimental results are implemented on a Pentium III 1000 MHz using programs written in MATLAB language. In our experiments, the samples of fingerprint images are captured using database of fingerprint images (Biometrix, 2003), (Neurotechnologija, 2004). The samples were scanned with AF-S2 sensor at 250 dpi resolution, 256-gray scale. The used sizes of these fingerprint images in the experiments are 50x50, 128x128 and 256x256 pixels. These images are of BMP kind. Figure 32 shows some Samples of fingerprint images which are used in the experiments.

4.2 Experimental Results

In this section, we produce experiments on images that match the existed images in the system database and images that are not existed in the system database. The experiments of matching the input images are contained noise 30%, 50%, 70% and 90% with the system database. The experiments of matching the input images are contained missing 25% and 50% with the system database. The experiment of matching the unknown input image with the system database, which is contained more than two, three and four of same input unknown image. The experiment of matching the input images are rotated by 2^0 , 4^0 , 6^0 , -2^0 , -4^0 and -6^0 with the system database, then the experiment of matching the input images are rotated by 30^0 , 45^0 , 90^0 , -30^0 , -45^0 and -90^0 , and the same rotated input images existed in the system database. Furthermore, the experiments of time measurement for learning and converging processes for both

standard backpropagation algorithm and proposed method using different sizes of images were held. The experiment measurement of identification and total time computing for the proposed method were computed.



Experiment 1:

The input of unknown image is identical to one of the images in the system database. Table 1 shows results for the first experiment. Note that the energy function of searched for input image is identical to the one that matches in the system.

Table 1: The Energy function for each image, the minimum of energy function, the next close, and the difference next close for the input unknown image.

Number & name of unknown images	Names of images in the system database	Energy function for each image	The minimum of energy function	The next close	The difference next close DNC
1. 48279a50	48279a50	-340.03	-340.03		-218.72
	410aa50	-117.27			
	411a50	-93			
	413a50	-104.69			
	415a50	-121.3		-121.3	
	417a50	-118.16			
	4910a50	-81			
	4914a50	-106.86			
	48271a50	-75.45			
	2a50x50	-100.26			
2. 410aa50	48279a50	-117.27			-227.16
	410aa50	-411.41	-411.41		
	411a50	-107.59			
	413a50	-148.34			
	415a50	-184.24		-184.24	
	417a50	-172.24			
	4910a50	-124.31			
	4914a50	-134.67			
	48271a50	-115.17			
	2a50x50	-132.85			
3. 411a50	48279a50	-93			-199.53
	410aa50	-107.59			
	411a50	-324.75	-324.75		
	413a50	-125.22		-125.22	
	415a50	-121.56			
	417a50	-100.44			
	4910a50	-87.93			
	4914a50	-91.01			
	48271a50	-81.29			
	2a50x50	-92.35			

Table 1: (continue)

Number & name of unknown images	Names of images in the system database	Energy function for each image	The minimum of energy function	The next close	The difference next close DNC
4. 413a50	48279a50	-104.69			-199.77
	410aa50	-148.34			
	411a50	-125.22			
	413a50	-364.97	-364.97		
	415a50	-165.2		-165.2	
	417a50	-145.74			
	4910a50	-101.5			
	4914a50	-106.21			
	48271a50	-91.92			
2a50x50	-116.68				
5. 415a50	48279a50	-121.3			-230.73
	410aa50	-184.24			
	411a50	-121.56			
	413a50	-165.2			
	415a50	-425.99	-425.99		
	417a50	-195.25		-195.25	
	4910a50	-121.38			
	4914a50	-143.65			
	48271a50	-114.21			
2a50x50	-136.48				
6. 417a50	48279a50	-118.16			-219.96
	410aa50	-172.24			
	411a50	-100.44			
	413a50	-145.74			
	415a50	-195.25		-195.25	
	417a50	-415.22	-415.22		
	4910a50	-119.96			
	4914a50	-146.03			
	48271a50	-117.84			
2a50x50	-131.61				
7. 4910a50	48279a50	-81			-216.07
	410aa50	-124.31		-124.31	
	411a50	-87.93			
	413a50	-101.5			
	415a50	-121.38			
	417a50	-119.96			
	4910a50	-340.39	-340.39		
	4914a50	-119.06			
	48271a50	-87.94			
2a50x50	-102.62				

Table 1: (continue)

Number & name of unknown images		Names of images in the system database	Energy function for each image	The minimum of energy function	The next close	The difference next close DNC
8.	4914a50	48279a50	-106.86			-217.86
		410aa50	-134.67			
		411a50	-91.01			
		413a50	-106.21			
		415a50	-143.65			
		417a50	-146.03		-146.03	
		4910a50	-119.06			
		4914a50	-363.9	-363.9		
		48271a50	-87.79			
		2a50x50	-112.72			
9.	48271a50	48279a50	-75.45			-210.17
		410aa50	-115.17			
		411a50	-81.29			
		413a50	-91.92			
		415a50	-114.21			
		417a50	-117.84		-117.84	
		4910a50	-87.94			
		4914a50	-87.79			
		48271a50	-328.01	-328.01		
		2a50x50	-92.87			
10.	2a50x50	48279a50	-100.26			-214.18
		410aa50	-132.85			
		411a50	-92.35			
		413a50	-116.68			
		415a50	-136.48		-136.48	
		417a50	-131.61			
		4910a50	-102.62			
		4914a50	-112.72			
		48271a50	-92.87			
		2a50x50	-350.66	-350.66		

Figure 33 shows the energy function values for ten different fingerprint images from the database. Figure 34 shows ten different fingerprint images in 34 (a) and the one input image that has a match with image 48279a50 in 34 (b).

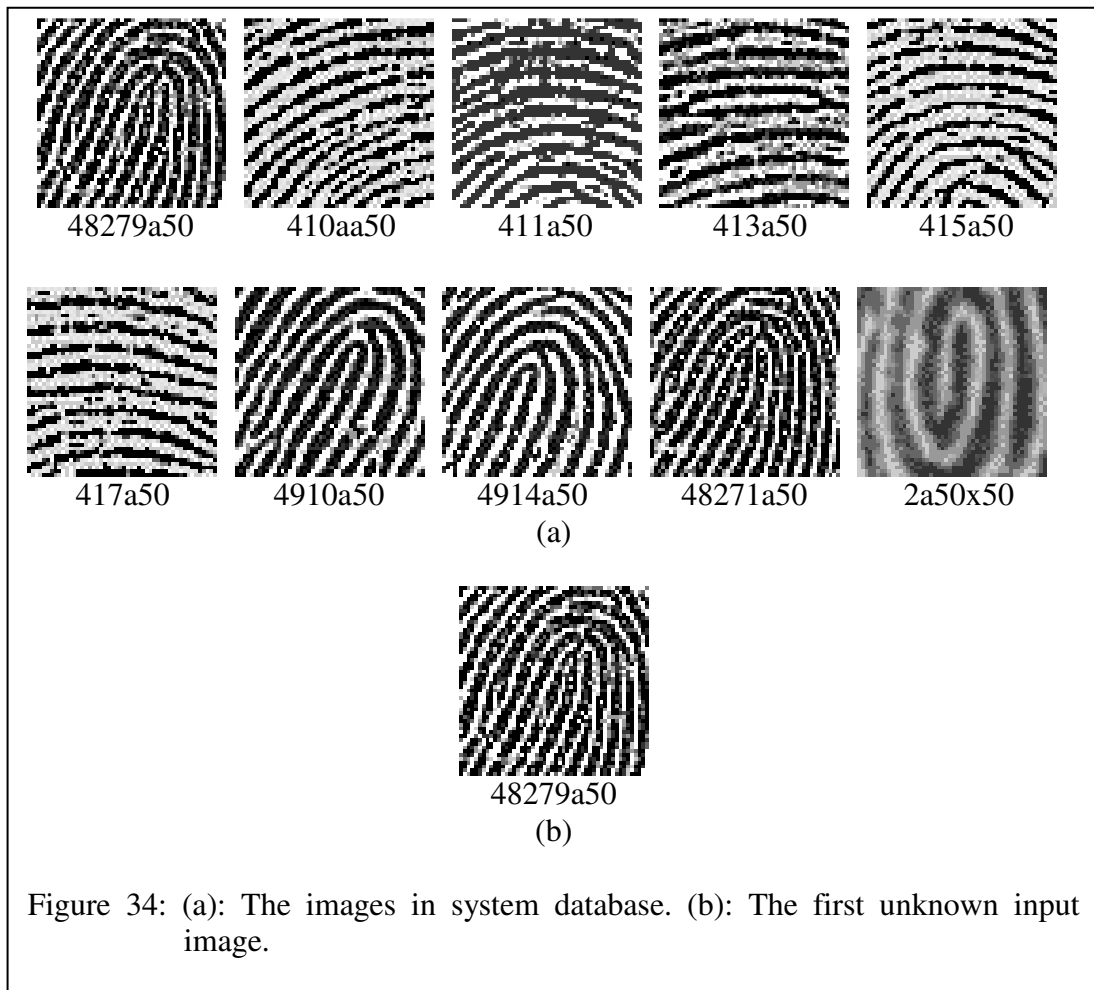
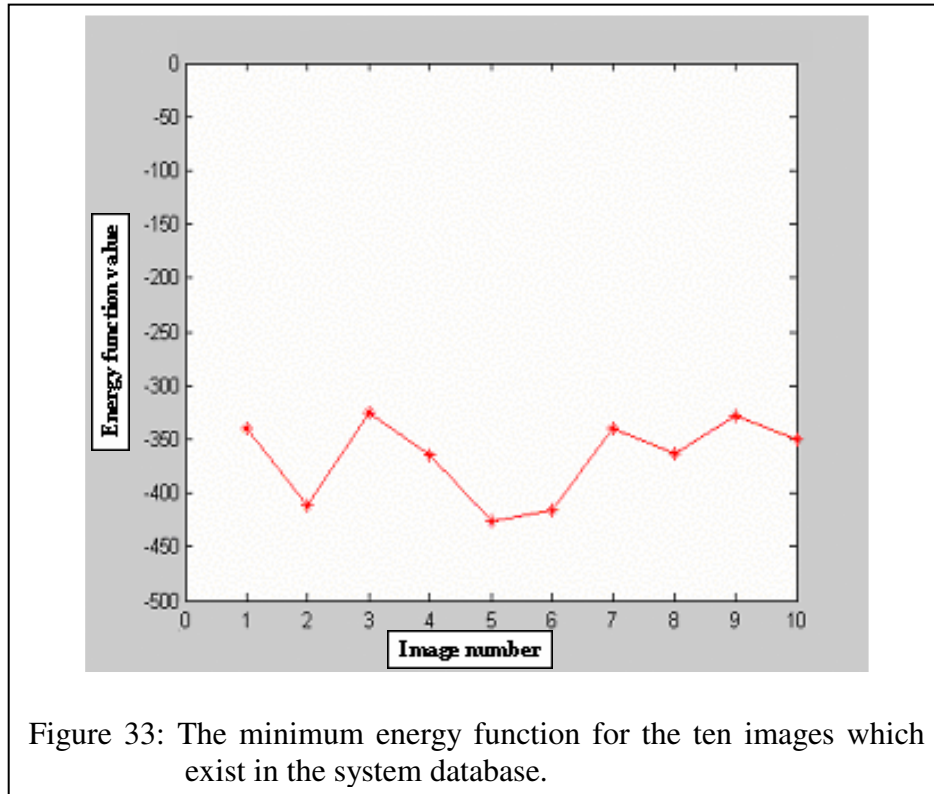


Figure 35 shows ten fingerprint images in the system database (35a) and image 2a50x50 that matches the tenth image in the database (35b).

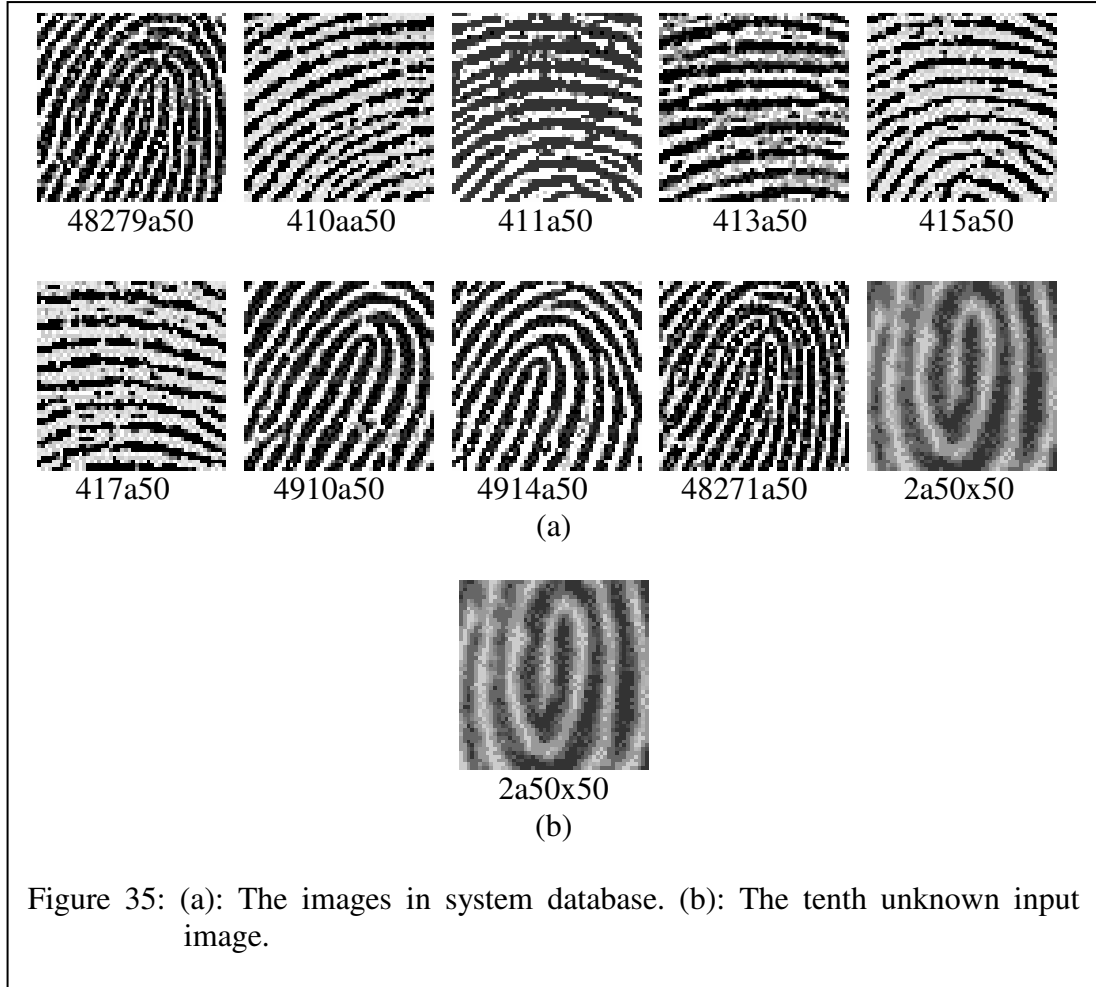


Figure 35: (a): The images in system database. (b): The tenth unknown input image.

Experiment 2:

The unknown input image is not identical to any one of the images of the system database. Table 2 shows the results. Non of the input image matches the ten fingerprint images in the system database.

Table 2: The Energy function for each image, the minimum of energy function, the next close, and the difference next close for the input unknown image.

Number & name of unknown images	Names of images in the system database	Energy function for each image	The minimum of energy function	The next close	The difference next close DNC
1. 48279a50LD	48279a50	-82.82			-11.89
	410aa50	-113.64			
	411a50	-97.43			
	413a50	-112.7			
	415a50	-125.6		-125.6	
	417a50	-137.5	-137.5		
	4910a50	-117.85			
	4914a50	-93.51			
	48271a50	-86.02			
2a50x50	-112.96				
2. 48279a50LT	48279a50	-107.25			-21.61
	410aa50	-119.17			
	411a50	-88.71			
	413a50	-105.34			
	415a50	-140.92	-140.92		
	417a50	-119.3		-119.3	
	4910a50	-98.03			
	4914a50	-91.34			
	48271a50	-86			
2a50x50	-101.25				
3. 412a50	48279a50	-116.36			-9.45
	410aa50	-157.86			
	411a50	-120.66			
	413a50	-127.94			
	415a50	-184.1	-184.1		
	417a50	-174.65		-174.65	
	4910a50	-117.76			
	4914a50	-125.19			
	48271a50	-108.98			
2a50x50	-134.33				

Table 2: (continue)

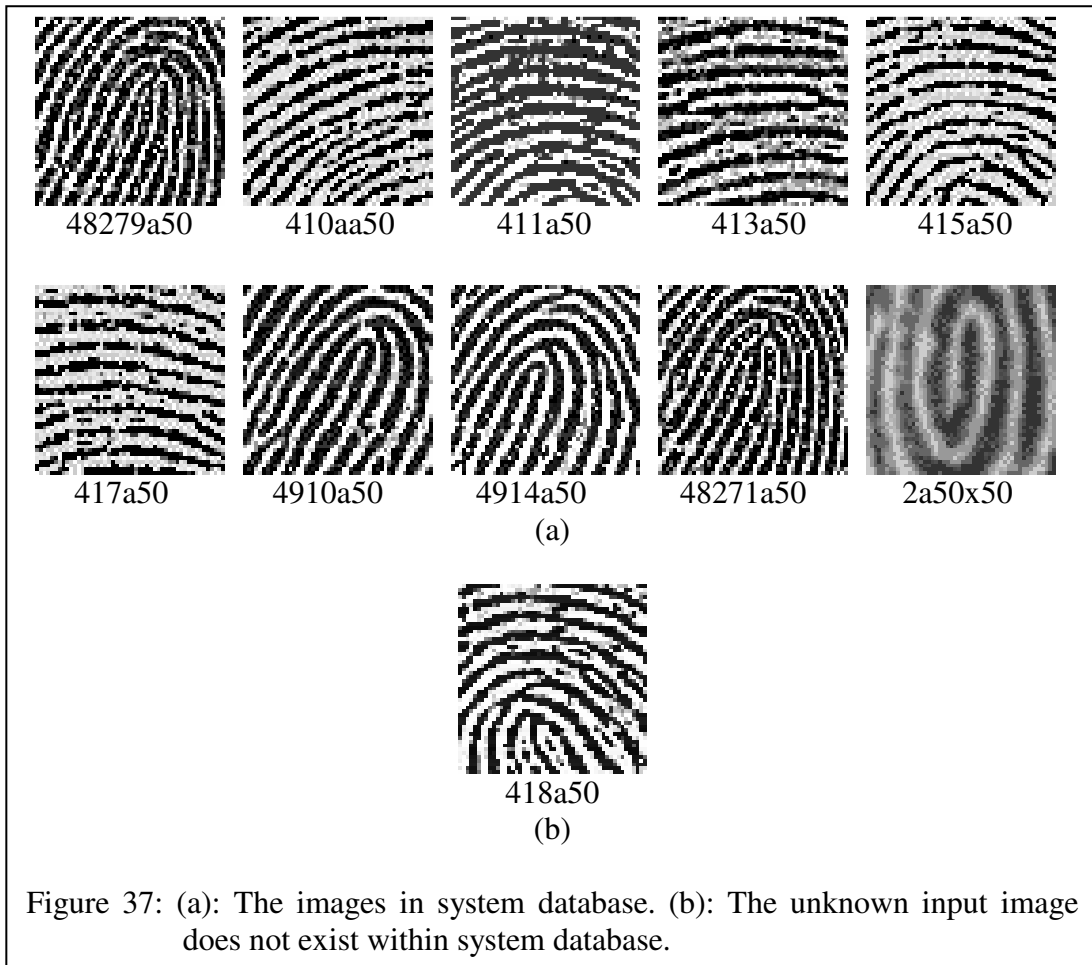
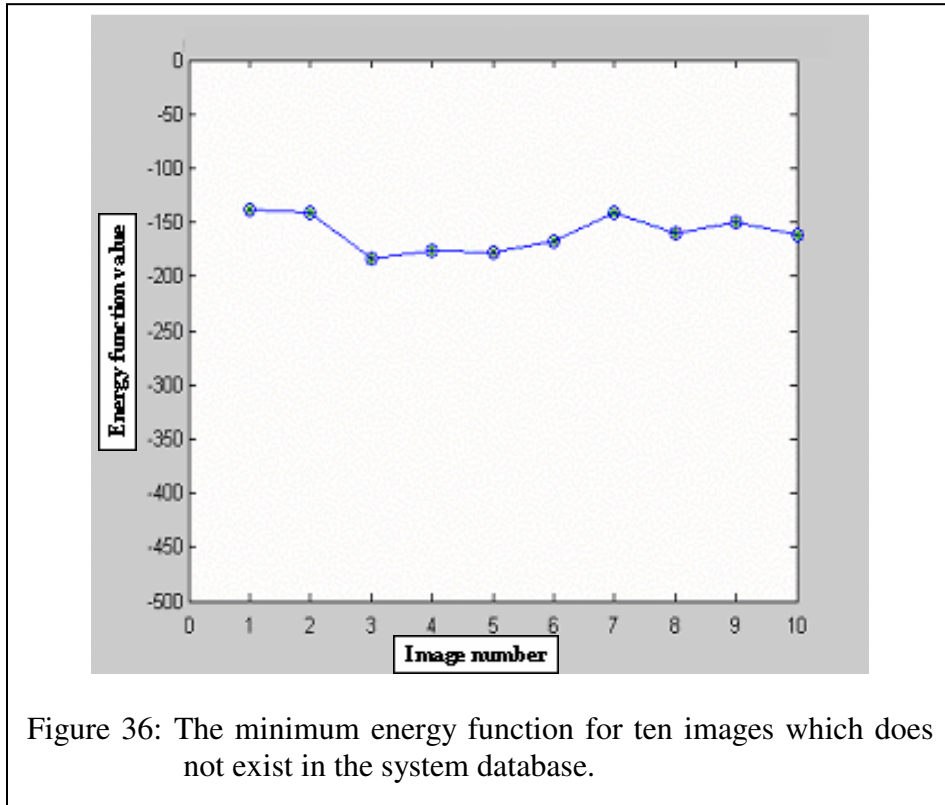
Number & name of unknown images	Names of images in the system database	Energy function for each image	The minimum of energy function	The next close	The difference next close DNC
4. 414a50	48279a50	-108.87			-9.84
	410aa50	-161.74			
	411a50	-110.2			
	413a50	-134.24			
	415a50	-176.57	-176.57		
	417a50	-166.72		-166.72	
	4910a50	-120.6			
	4914a50	-133.26			
	48271a50	-99.94			
2a50x50	-117.81				
5. 416a50	48279a50	-116.45			-4
	410aa50	-177.21	-177.21		
	411a50	-135.15			
	413a50	-139.35			
	415a50	-141.05			
	417a50	-173.21		-173.21	
	4910a50	-122.01			
	4914a50	-140.64			
	48271a50	-107.96			
2a50x50	-131.2				
6. 418a50	48279a50	-122.76			-12.64
	410aa50	-136.13			
	411a50	-100.71			
	413a50	-129.1			
	415a50	-167.35	-167.35		
	417a50	-154.7		-154.7	
	4910a50	-107.61			
	4914a50	-119.65			
	48271a50	-96.94			
2a50x50	-118.79				
7. 419a50	48279a50	-98.72			-6.61
	410aa50	-134.03		-134.03	
	411a50	-106.9			
	413a50	-121.54			
	415a50	-140.65	-140.65		
	417a50	-109.77			
	4910a50	-106.21			
	4914a50	-106.77			
	48271a50	-95.13			
2a50x50	-109.98				

Table 2: (continue)

Number & name of unknown images		Names of images in the system database	Energy function for each image	The minimum of energy function	The next close	The difference next close DNC
8.	4911a50	48279a50	-120.85			-9.58
		410aa50	-151.2		-151.2	
		411a50	-98.09			
		413a50	-129.86			
		415a50	-160.79	-160.79		
		417a50	-144.14			
		4910a50	-139.7			
		4914a50	-130.82			
		48271a50	-119.61			
		2a50x50	-118.27			
9.	4912a50	48279a50	-128.3			-5.64
		410aa50	-144.73		-144.73	
		411a50	-105.53			
		413a50	-120.66			
		415a50	-150.37	-150.37		
		417a50	-143.96			
		4910a50	-81.23			
		4914a50	-88.95			
		48271a50	-94.4			
		2a50x50	-114.95			
10.	4913a50	48279a50	-76.12			-11.24
		410aa50	-144.5			
		411a50	-111.96			
		413a50	-134.23			
		415a50	-161.3	-161.3		
		417a50	-150.05		-150.05	
		4910a50	-99.7			
		4914a50	-113.32			
		48271a50	-79.65			
		2a50x50	-117.35			

Figure 36 shows the relationship between the Energy function and the identification of ten different fingerprint images which are not in the system database. Note that the values of Energy function are smaller than those with match, as shown in Figure 33.

Figure 37 (a) shows ten different fingerprint images and non of them matches with the image in Figure 37 (b).



Experiment 3:

The unknown input image is identical to one of the images in the system database, but it contains noise 30%, 50%, 70% and 90%. Table 3 shows results of the third experiment. Note that the energy function of searched for the input image (contains noise 30%) is identical to the one that matches in the system.

Table 3: The Energy function for each image and the minimum of energy function for the input unknown image. The input image is identical to one of images of system database, but it contains noise 30%.

Number & name of unknown image Noise (30%)	Names of images in the system database	Energy function for each image	The minimum of energy function
1. 48279a50	48279a50	-268.49	-268.49
	410aa50	-128.08	
	411a50	-94.54	
	413a50	-114.7	
	415a50	-138.05	
	417a50	-133.38	
	4910a50	-88.84	
	4914a50	-113.04	
	48271a50	-83	
	2a50x50	-110.59	
2. 410aa50	48279a50	-115.19	
	410aa50	-339.08	-339.08
	411a50	-106.26	
	413a50	-146.37	
	415a50	-181.8	
	417a50	-165.39	
	4910a50	-129.49	
	4914a50	-130.84	
	48271a50	-115.73	
	2a50x50	-130.48	
3. 411a50	48279a50	-100.77	
	410aa50	-117.89	
	411a50	-260.37	-260.37
	413a50	-126.99	
	415a50	-138.57	
	417a50	-119.86	
	4910a50	-97.06	
	4914a50	-106.02	
	48271a50	-84.86	
	2a50x50	-103.43	

Table 3: (continue)

Number & name of unknown image Noise (30%)		Names of images in the system database	Energy function for each image	The minimum of energy function
4.	413a50	48279a50	-104.22	
		410aa50	-147.6	
		411a50	-119.61	
		413a50	-293.72	-293.72
		415a50	-168.89	
		417a50	-148.5	
		4910a50	-102.4	
		4914a50	-108.76	
		48271a50	-95.52	
		2a50x50	-121.76	
5.	415a50	48279a50	-120.66	
		410aa50	-180.42	
		411a50	-119.61	
		413a50	-150.77	
		415a50	-352.3	-352.3
		417a50	-187.27	
		4910a50	-120.18	
		4914a50	-145.53	
		48271a50	-115.45	
		2a50x50	-130.11	
6.	417a50	48279a50	-116.49	
		410aa50	-171.94	
		411a50	-99.55	
		413a50	-146.39	
		415a50	-189.81	
		417a50	-343.23	-343.23
		4910a50	-122.42	
		4914a50	-138.12	
		48271a50	-116.07	
		2a50x50	-128.41	
7.	4910a50	48279a50	-81	
		410aa50	-129.97	
		411a50	-93.46	
		413a50	-112.62	
		415a50	-133.01	
		417a50	-131.19	
		4910a50	-267.52	-267.52
		4914a50	-120.2	
		48271a50	-90.69	
		2a50x50	-110.52	

Table 3: (continue)

Number & name of unknown image Noise (30%)		Names of images in the system database	Energy function for each image	The minimum of energy function
8.	4914a50	48279a50	-103.27	
		410aa50	-144.05	
		411a50	-97.21	
		413a50	-114.86	
		415a50	-147.91	
		417a50	-152.36	
		4910a50	-125.35	
		4914a50	-294.8	-294.8
		48271a50	-96.39	
		2a50x50	-118.34	
9.	48271a50	48279a50	-86.41	
		410aa50	-118	
		411a50	-86.27	
		413a50	-99.15	
		415a50	-124.93	
		417a50	-125.14	
		4910a50	-93.28	
		4914a50	-100.33	
		48271a50	-257.77	-257.77
		2a50x50	-94.22	
10.	2a50x50	48279a50	-102.9	
		410aa50	-135.95	
		411a50	-97.26	
		413a50	-119.24	
		415a50	-142.92	
		417a50	-140.66	
		4910a50	-101.6	
		4914a50	-116.07	
		48271a50	-97.86	
		2a50x50	-288.63	-288.63

Table 4 shows the results for input images (contain noise 50%). Note that the energy function of searched for the input image (contains noise 50%) is identical to the one that matches in the system.

Table 4: The energy function for each image and the minimum of energy function for the input unknown image. The input image is identical to one of images of system database, but it contains noise 50%.

Number & name of Unknown image Noise (50%)	Names of images in the system database	Energy function for each image	The minimum of Energy function
1. 48279a50	48279a50	-222.08	-222.08
	410aa50	-138.88	
	411a50	-97.17	
	413a50	-121.01	
	415a50	-133.86	
	417a50	-126.4	
	4910a50	-104.62	
	4914a50	-108.83	
	48271a50	-88.56	
	2a50x50	-107.12	
2. 410aa50	48279a50	-114.61	
	410aa50	-286.67	-286.67
	411a50	-105.69	
	413a50	-145.12	
	415a50	-177.99	
	417a50	-170.89	
	4910a50	-120.77	
	4914a50	-133.85	
	48271a50	-110.29	
	2a50x50	-125.3	
3. 411a50	48279a50	-98.65	
	410aa50	-124.06	
	411a50	-209.68	-209.68
	413a50	-122.34	
	415a50	-139.35	
	417a50	-131.83	
	4910a50	-98.88	
	4914a50	-105.69	
	48271a50	-94.18	
	2a50x50	-99.98	

Table 4: (continue)

Number & name of unknown image Noise (50%)		Names of images in the system database	Energy function for each image	The minimum of energy function
4.	413a50	48279a50	-106.33	
		410aa50	-154.98	
		411a50	-121.97	
		413a50	-257.02	-257.02
		415a50	-170.02	
		417a50	-158.07	
		4910a50	-101.64	
		4914a50	-121.69	
		48271a50	-96.95	
		2a50x50	-116.41	
5.	415a50	48279a50	-125.42	
		410aa50	-169.81	
		411a50	-116.85	
		413a50	-149.24	
		415a50	-297.98	-297.98
		417a50	-180.33	
		4910a50	-115.18	
		4914a50	-141.17	
		48271a50	-114.11	
		2a50x50	-132.8	
6.	417a50	48279a50	-115.21	
		410aa50	-161.76	
		411a50	-98.85	
		413a50	-141.08	
		415a50	-185.24	
		417a50	-295.49	-295.49
		4910a50	-112.58	
		4914a50	-145.83	
		48271a50	-113.84	
		2a50x50	-136.6	
7.	4910a50	48279a50	-94.92	
		410aa50	-138.13	
		411a50	-98.53	
		413a50	-115.56	
		415a50	-145.52	
		417a50	-137.88	
		4910a50	-225.86	-225.86
		4914a50	-122.83	
		48271a50	-97.93	
		2a50x50	-109.78	

Table 4: (continue)

Number & name of Unknown image Noise (50%)		Names of images in the system database	Energy function for each image	The minimum of Energy function
8.	4914a50	48279a50	-110.91	
		410aa50	-143.57	
		411a50	-89.77	
		413a50	-114.89	
		415a50	-150.38	
		417a50	-145.88	
		4910a50	-115.97	
		4914a50	-239.26	-239.26
		48271a50	-94.17	
		2a50x50	-118.46	
9.	48271a50	48279a50	-90.33	
		410aa50	-128.88	
		411a50	-92.14	
		413a50	-108.01	
		415a50	-134.96	
		417a50	-135.12	
		4910a50	-103.08	
		4914a50	-108.95	
		48271a50	-213.05	-213.05
		2a50x50	-99.21	
10.	2a50x50	48279a50	-98.24	
		410aa50	-139	
		411a50	-100.26	
		413a50	-118.94	
		415a50	-152.8	
		417a50	-138.81	
		4910a50	-106.14	
		4914a50	-113.61	
		48271a50	-93.78	
		2a50x50	-228.55	-228.55

Table 5 shows the results for input images (contain noise 70%). Note that the energy function of searched for the input image (contains noise 70%) is identical to the one that matches in the system.

Table 5: The Energy function for each image and the minimum of energy function for the input unknown image. The input image is identical to one of images of system database, but it contains noise 70%.

Number & name of unknown image Noise (70%)		Names of images in the system database	Energy function for each image	The minimum of energy function
1.	48279a50	48279a50	-178.33	-178.33
		410aa50	-144.64	
		411a50	-107.15	
		413a50	-125.99	
		415a50	-144.77	
		417a50	-150.07	
		4910a50	-103.75	
		4914a50	-120.96	
		48271a50	-89.18	
		2a50x50	-100.68	
2.	410aa50	48279a50	-111.99	
		410aa50	-231.23	-231.23
		411a50	-102.66	
		413a50	-132.91	
		415a50	-174.34	
		417a50	-164.33	
		4910a50	-119.34	
		4914a50	-129.17	
		48271a50	-108.31	
		2a50x50	-125.6	
3.	411a50	48279a50	-108.38	
		410aa50	-143.57	
		411a50	-167.09	-167.09
		413a50	-130.24	
		415a50	-154.9	
		417a50	-146.05	
		4910a50	-101.18	
		4914a50	-126.63	
		48271a50	-103.42	
		2a50x50	-105.7	

Table 5: (continue)

Number & name of unknown image Noise (70%)		Names of images in the system database	Energy function for each image	The minimum of energy function
4.	413a50	48279a50	-108.17	
		410aa50	-148.56	
		411a50	-106.13	
		413a50	-195.37	-195.37
		415a50	-160.78	
		417a50	-159.02	
		4910a50	-106.98	
		4914a50	-120.62	
		48271a50	-108.98	
		2a50x50	-123.37	
5.	415a50	48279a50	-109.56	
		410aa50	-165.54	
		411a50	-115.71	
		413a50	-139.28	
		415a50	-244.13	-244.13
		417a50	-171.8	
		4910a50	-106.87	
		4914a50	-125.92	
		48271a50	-102.88	
		2a50x50	-132.09	
6.	417a50	48279a50	-110.36	
		410aa50	-157.2	
		411a50	-97.61	
		413a50	-129.14	
		415a50	-166.74	
		417a50	-230.46	-230.46
		4910a50	-112.97	
		4914a50	-125.89	
		48271a50	-99.8	
		2a50x50	-118.61	
7.	4910a50	48279a50	-107.25	
		410aa50	-153.3	
		411a50	-103.73	
		413a50	-128.98	
		415a50	-162.66	
		417a50	-151.3	
		4910a50	-194.55	-194.55
		4914a50	-125.97	
		48271a50	-109.02	
		2a50x50	-123.72	

Table 5: (continue)

Number & name of unknown image Noise (70%)		Names of images in the system database	Energy function for each image	The minimum of energy function
8.	4914a50	48279a50	-107.14	
		410aa50	-148.88	
		411a50	-93.38	
		413a50	-111.58	
		415a50	-149.2	
		417a50	-146.97	
		4910a50	-118.15	
		4914a50	-188.99	-188.99
		48271a50	-95.7	
		2a50x50	-107.09	
9.	48271a50	48279a50	-96.4	
		410aa50	-142.41	
		411a50	-100.85	
		413a50	-120.66	
		415a50	-154.68	
		417a50	-143.32	
		4910a50	-106.4	
		4914a50	-112.39	
		48271a50	-157.7	-157.7
		2a50x50	-105.75	
10.	2a50x50	48279a50	-100.88	
		410aa50	-147.04	
		411a50	-105.73	
		413a50	-122.93	
		415a50	-153.86	
		417a50	-150.36	
		4910a50	-108.6	
		4914a50	-119.77	
		48271a50	-96.07	
		2a50x50	-181.74	-181.74

Table 6 shows the results for input images (contain noise 90%). Note that not all the input images are matched in the system database.

Table 6: The Energy function for each image and the minimum of energy function for the input unknown image. The input image is identical to one of images of system database, but it contains noise 90%.

Number & name of unknown image Noise (90%)	Names of images in the system database	Energy function for each image	The minimum of energy function
1. 48279a50	48279a50	-125.42	
	410aa50	-158.69	
	411a50	-100.5	
	413a50	-127.28	
	415a50	-158.74	-158.74
	417a50	-149.92	
	4910a50	-115.04	
	4914a50	-119.57	
	48271a50	-93.13	
	2a50x50	-125.19	
2. 410aa50	48279a50	-111.04	
	410aa50	-169.37	-169.37
	411a50	-102.13	
	413a50	-124.33	
	415a50	-161.04	
	417a50	-155.34	
	4910a50	-113.79	
	4914a50	-124.25	
	48271a50	-106.65	
	2a50x50	-120.8	
3. 411a50	48279a50	-103.51	
	410aa50	-147.38	
	411a50	-136.09	
	413a50	-135.29	
	415a50	-164.62	-164.62
	417a50	-149.96	
	4910a50	-101.47	
	4914a50	-114.63	
	48271a50	-102.21	
	2a50x50	-114.99	

Table 6: (continue)

Number & name of unknown image Noise (90%)		Names of images in the system database	Energy function for each image	The minimum of energy function
4.	413a50	48279a50	-116.65	
		410aa50	-159.51	
		411a50	-116.41	
		413a50	-156.24	
		415a50	-168.99	-168.99
		417a50	-159.68	
		4910a50	-114.89	
		4914a50	-130.7	
		48271a50	-103.13	
		2a50x50	-129.23	
5.	415a50	48279a50	-110.76	
		410aa50	-151.26	
		411a50	-106.44	
		413a50	-131.11	
		415a50	-184.66	-184.66
		417a50	-157.14	
		4910a50	-108.79	
		4914a50	-122.43	
		48271a50	-113.3	
		2a50x50	-118.78	
6.	417a50	48279a50	-110.09	
		410aa50	-150.44	
		411a50	-108.17	
		413a50	-132.74	
		415a50	-163.82	
		417a50	-183.81	-183.81
		4910a50	-108.77	
		4914a50	-121.08	
		48271a50	-102.81	
		2a50x50	-122.07	
7.	4910a50	48279a50	-107.35	
		410aa50	-150.03	
		411a50	-104.4	
		413a50	-124.9	
		415a50	-163.11	-163.11
		417a50	-160.49	
		4910a50	-139.22	
		4914a50	-129.17	
		48271a50	-100.22	
		2a50x50	-116.73	

Table 6: (continue)

Number & name of unknown image Noise (90%)		Names of images in the system database	Energy function for each image	The minimum of energy function
8.	4914a50	48279a50	-119.08	
		410aa50	-158.85	
		411a50	-113.56	
		413a50	-135.93	
		415a50	-165.36	-165.36
		417a50	-161.97	
		4910a50	-108.81	
		4914a50	-142.24	
		48271a50	-94.09	
		2a50x50	-118.83	
9.	48271a50	48279a50	-100.35	
		410aa50	-145.81	
		411a50	-102.31	
		413a50	-136.2	
		415a50	-161.79	-161.79
		417a50	-144.66	
		4910a50	-107.98	
		4914a50	-113.25	
		48271a50	-126.33	
		2a50x50	-112.6	
10.	2a50x50	48279a50	-106.62	
		410aa50	-154.33	-154.33
		411a50	-107.49	
		413a50	-126.63	
		415a50	-149.48	
		417a50	-140.72	
		4910a50	-106.47	
		4914a50	-114.78	
		48271a50	-101.4	
		2a50x50	-146.96	

Figure 38 shows the names of input fingerprint images with their original images and the same images that contain noise 30%, 50%, 70% and 90%.






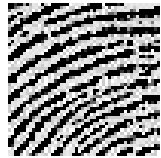




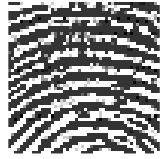









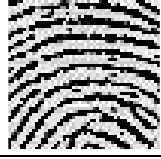
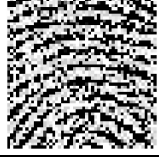



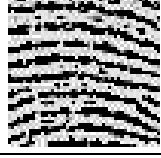
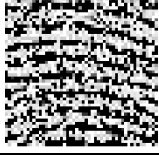



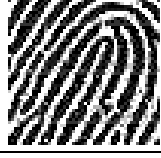
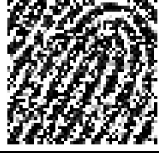








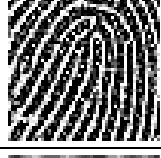




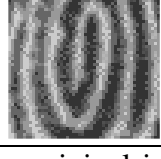
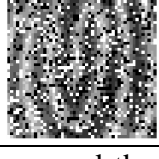



Name of image	The original image	Noise in the image 30%	Noise in the image 50%	Noise in the image 70%	Noise in the image 90%
48279a50					
410aa50					
411a50					
413a50					
415a50					
417a50					
4910a50					
4914a50					
48271a50					
2a50x50					

Figure 38: The original images and the same images that contain noise 30%, 50%, 70% and 90%.

Experiment 4:

The unknown input image is identical to one of these in the system database, but it contains missing 25% and 50% of part. Table 7 shows the results for input images (contain missing 25%). Note that the energy function of searched for the input image (contain missing 25%) is identical to the one that matches in the system.

Table 7: The Energy function for each image and the minimum of energy function for the input unknown image. The input image is identical to one of images of system database, but it contains missing 25%.

Number & name of unknown image Missing (25%)	Names of images in the system database	Energy function for each image	The minimum of energy function
1. 48279a50	48279a50	-300.27	-300.27
	410aa50	-160.94	
	411a50	-107.81	
	413a50	-131.34	
	415a50	-166.58	
	417a50	-166.9	
	4910a50	-100.62	
	4914a50	-128.64	
	48271a50	-95.31	
	2a50x50	-125.95	
2. 410aa50	48279a50	-136.53	
	410aa50	-373.78	-373.78
	411a50	-121.2	
	413a50	-169.65	
	415a50	-222.53	
	417a50	-206.58	
	4910a50	-136.95	
	4914a50	-153.44	
	48271a50	-130.31	
	2a50x50	-162.54	
3. 411a50	48279a50	-111.64	
	410aa50	-151.23	
	411a50	-279.33	-279.33
	413a50	-154.34	
	415a50	-157.83	
	417a50	-142.58	
	4910a50	-114.11	
	4914a50	-123.85	
	48271a50	-104.84	
	2a50x50	-117.08	

Table 7: (continue)

Number & name of unknown image Missing (25%)		Names of images in the system database	Energy function for each image	The minimum of energy function
4.	413a50	48279a50	-122.59	
		410aa50	-187.88	
		411a50	-151.5	
		413a50	-337.43	-337.43
		415a50	-208.59	
		417a50	-181.63	
		4910a50	-127.13	
		4914a50	-132.84	
		48271a50	-111.27	
		2a50x50	-133.39	
5.	415a50	48279a50	-132.48	
		410aa50	-213.07	
		411a50	-134.55	
		413a50	-178.35	
		415a50	-390.25	-390.25
		417a50	-221.91	
		4910a50	-135.3	
		4914a50	-162.63	
		48271a50	-126.36	
		2a50x50	-151.97	
6.	417a50	48279a50	-133.71	
		410aa50	-195.07	
		411a50	-126.35	
		413a50	-172.58	
		415a50	-222.73	
		417a50	-400.26	-400.26
		4910a50	-132.91	
		4914a50	-163.44	
		48271a50	-127.85	
		2a50x50	-143.53	
7.	4910a50	48279a50	-99.2	
		410aa50	-164.64	
		411a50	-107.08	
		413a50	-134.3	
		415a50	-166.1	
		417a50	-164.2	
		4910a50	-307.16	-307.16
		4914a50	-138.53	
		48271a50	-103.54	
		2a50x50	-122.59	

Table 7: (continue)

Number & name of unknown image Missing (25%)		Names of images in the system database	Energy function for each image	The minimum of energy function
8.	4914a50	48279a50	-128.02	
		410aa50	-166.98	
		411a50	-111.03	
		413a50	-142.32	
		415a50	-185.26	
		417a50	-177.68	
		4910a50	-141.91	
		4914a50	-337.12	-337.12
		48271a50	-116.67	
		2a50x50	-133.89	
9.	48271a50	48279a50	-92.89	
		410aa50	-159.99	
		411a50	-110.95	
		413a50	-129.03	
		415a50	-167.76	
		417a50	-172.88	
		4910a50	-120.35	
		4914a50	-119.96	
		48271a50	-285.16	-285.16
		2a50x50	-132.15	
10.	2a50x50	48279a50	-115.88	
		410aa50	-164.53	
		411a50	-110.28	
		413a50	-145.54	
		415a50	-177.12	
		417a50	-173.67	
		4910a50	-121.89	
		4914a50	-130.4	
		48271a50	-107.14	
		2a50x50	-306.69	-306.69

Table 8 shows the results for input images (contain missing 50%). Note that most of input images (contain missing 50%) are matched in the system database.

Table 8: The Energy function for each image and the minimum of energy function for the input unknown image. The input image is identical to one of images of system database, but it contains missing 50%.

Number & name of unknown image Missing (50%)	Names of images in the system database	Energy function for each image	The minimum of energy function
1. 48279a50	48279a50	-230.85	-230.85
	410aa50	-194.98	
	411a50	-143.44	
	413a50	-169.61	
	415a50	-207.96	
	417a50	-186.99	
	4910a50	-146.48	
	4914a50	-169.28	
	48271a50	-146	
2a50x50	-143.34		
2. 410aa50	48279a50	-152.49	
	410aa50	-350.12	-350.12
	411a50	-138.45	
	413a50	-195.65	
	415a50	-264.26	
	417a50	-252.13	
	4910a50	-155.62	
	4914a50	-178.01	
	48271a50	-146.54	
2a50x50	-182		
3. 411a50	48279a50	-142.31	
	410aa50	-206.83	
	411a50	-246.95	-246.95
	413a50	-180.78	
	415a50	-227.37	
	417a50	-206	
	4910a50	-139.57	
	4914a50	-164	
	48271a50	-131.42	
2a50x50	-168.76		

Table 8: (continue)

Number & name of unknown image Missing (50%)		Names of images in the system database	Energy function for each image	The minimum of energy function
4.	413a50	48279a50	-126.36	
		410aa50	-211.63	
		411a50	-137.77	
		413a50	-264.76	-264.76
		415a50	-223.45	
		417a50	-215.33	
		4910a50	-139.56	
		4914a50	-153.87	
		48271a50	-121.11	
		2a50x50	-139.26	
5.	415a50	48279a50	-149.19	
		410aa50	-241.02	
		411a50	-148.27	
		413a50	-198.46	
		415a50	-371.13	-371.13
		417a50	-271.59	
		4910a50	-156.05	
		4914a50	-184.14	
		48271a50	-143.68	
		2a50x50	-182.96	
6.	417a50	48279a50	-146.54	
		410aa50	-226.09	
		411a50	-151.78	
		413a50	-187.49	
		415a50	-240.11	
		417a50	-349.87	-349.87
		4910a50	-157.68	
		4914a50	-187.33	
		48271a50	-139.99	
		2a50x50	-166.23	
7.	4910a50	48279a50	-114.87	
		410aa50	-205.65	
		411a50	-118.36	
		413a50	-168.21	
		415a50	-209.47	
		417a50	-212.13	
		4910a50	-242.23	-242.23
		4914a50	-175.61	
		48271a50	-124.55	
		2a50x50	-148.34	

Table 8: (continue)

Number & name of unknown image Missing (50%)	Names of images in the system database	Energy function for each image	The minimum of energy function
8. 4914a50	48279a50	-149.33	
	410aa50	-217.3	
	411a50	-145.46	
	413a50	-173.5	
	415a50	-235.85	
	417a50	-229.66	
	4910a50	-151.47	
	4914a50	-283.34	-283.34
	48271a50	-117.89	
	2a50x50	-168.15	
9. 48271a50	48279a50	-119.09	
	410aa50	-203.36	
	411a50	-119.02	
	413a50	-163.18	
	415a50	-207.06	
	417a50	-210.31	
	4910a50	-134.13	
	4914a50	-162.85	
	48271a50	-219.46	-219.46
	2a50x50	-144.12	
10. 2a50x50	48279a50	-104.5	
	410aa50	-159.71	
	411a50	-119.64	
	413a50	-138.37	
	415a50	-167.31	-167.31
	417a50	-159.88	
	4910a50	-116.85	
	4914a50	-126.31	
	48271a50	-100.39	
	2a50x50	-132.39	

Figure 39 shows the names of input fingerprint images with their original images and the same images that contain missing 25% and 50%.

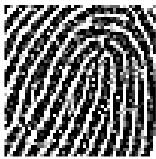
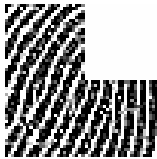

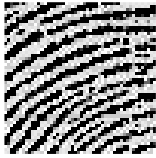
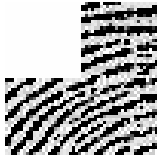

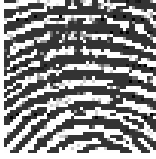





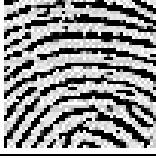
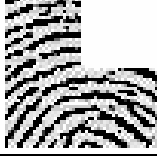
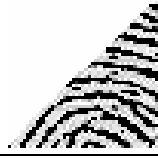
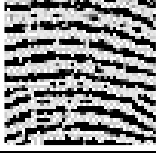
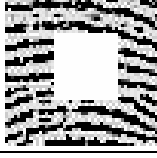

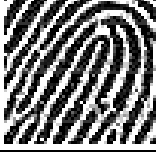
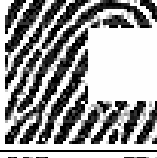
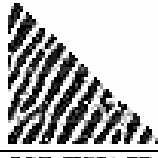

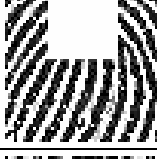

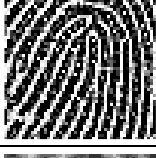
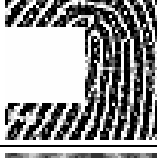
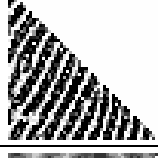
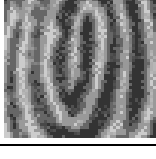
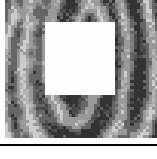
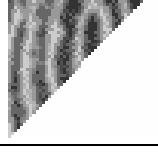
Name of image	The original image	Missing in the image 25%	Missing in the image 50%
48279a50			
410aa50			
411a50			
413a50			
415a50			
417a50			
4910a50			
4914a50			
48271a50			
2a50x50			

Figure 39: The original images and the same images contain missing 25% and 50%.

Experiment 5:

The unknown input image is identical to two, three or four images of the system database. The result is converging one image of them (all have the same minimum of energy function). Table 9 shows the results when the input image is identical to two of images of system database.

Table 9: The unknown input image is identical to two of images of system database.

Number & name of unknown image	Names of images in the system database	Energy function for each image	The minimum of energy function
1. 48279a50	48279a50	-340.03	-340.03
	410aa50	-117.27	
	411a50	-93	
	413a50	-104.69	
	415a50	-121.3	
	48279a50	-340.03	
	4910a50	-81	
	4914a50	-106.86	
	48271a50	-75.45	
	2a50x50	-100.26	
2. 410aa50	48279a50	-117.27	-411.41
	410aa50	-411.41	
	411a50	-107.59	
	413a50	-148.34	
	415a50	-184.24	
	417a50	-172.24	
	4910a50	-124.31	
	410aa50	-411.41	
	48271a50	-115.17	
	2a50x50	-132.85	
3. 411a50	48279a50	-93	-324.75
	410aa50	-107.59	
	411a50	-324.75	
	413a50	-125.22	
	415a50	-121.56	
	411a50	-324.75	
	4910a50	-87.93	
	4914a50	-91.01	
	48271a50	-81.29	
	2a50x50	-92.35	

Table 10 shows the results when the input image is identical to three of images of system database.

Table 10: The unknown input image is identical to three images of the system database.

Number & name of unknown images	Names of images in the system database	Energy function for each image	The minimum of energy function
4. 413a50	48279a50	-104.69	-364.97
	410aa50	-148.34	
	411a50	-125.22	
	413a50	-364.97	
	415a50	-165.2	
	413a50	-364.97	
	4910a50	-101.5	
	4914a50	-106.21	
	48271a50	-91.92	
	413a50	-364.97	
5. 415a50	415a50	-425.99	-425.99
	410aa50	-184.24	
	411a50	-121.56	
	413a50	-165.2	
	415a50	-425.99	
	417a50	-195.25	
	4910a50	-121.38	
	4914a50	-143.65	
	48271a50	-114.21	
	415a50	-425.99	
6. 417a50	417a50	-415.22	-415.22
	410aa50	-172.24	
	417a50	-415.22	
	413a50	-145.74	
	415a50	-195.25	
	417a50	-415.22	
	4910a50	-119.96	
	4914a50	-146.03	
	48271a50	-117.84	
	2a50x50	-131.61	
7. 4910a50	48279a50	-81	-340.39
	4910a50	-340.39	
	411a50	-87.93	
	4910a50	-340.39	
	415a50	-121.38	
	417a50	-119.96	
	4910a50	-340.39	
	4914a50	-119.06	
	48271a50	-87.94	
	2a50x50	-102.62	

Table 11 shows the results when the input image is identical to four of images of the system database.

Table 11: The unknown input image is identical to four of images of system database.

Number & name of unknown images		Names of images in the system database	Energy function for each image	The minimum of energy function
8.	4914a50	48279a50	-106.86	-363.9
		4914a50	-363.90	
		411a50	-91.01	
		413a50	-106.21	
		4914a50	-363.9	
		417a50	-146.03	
		4910a50	-119.06	
		4914a50	-363.9	
		48271a50	-87.79	
		4914a50	-363.9	
9.	48271a50	48271a50	-328.01	-328.01
		48271a50	-328.01	
		411a50	-81.29	
		413a50	-91.92	
		415a50	-114.21	
		417a50	-117.84	
		4910a50	-87.94	
		4914a50	-87.79	
		48271a50	-328.01	
		48271a50	-328.01	
10.	2a50x50	48279a50	-100.26	-350.66
		2a50x50	-350.66	
		411a50	-92.35	
		413a50	-116.68	
		2a50x50	-350.66	
		417a50	-131.61	
		4910a50	-102.62	
		2a50x50	-350.66	
		48271a50	-92.87	
		2a50x50	-350.66	

Experiment 6:

The unknown input image is identical to one images of the system database, but it is rotated by 2^0 , 4^0 , 6^0 , -2^0 , -4^0 and -6^0 . Table 12 shows the results for the input image rotated by 2^0 , 4^0 and 6^0 . Note that most of input rotated images are not matched in the system database.

Table 12: The unknown input image (2a50x50) is rotated by 2^0 , 4^0 and 6^0 .

Number & name of unknown images		Names of images in the system database	Energy function for each image	The minimum of energy function
1.	2a50x50	48279a50	-84.2	
	Rotated by 2^0	410aa50	-131.58	
		411a50	-95.67	
		413a50	-108.81	
		415a50	-132.72	
		417a50	-132.56	
		4910a50	-90.93	
		4914a50	-103.55	
		48271a50	-80.94	
		2a50x50	-233.01	-233.01
2.	2a50x50	48279a50	-150.48	
	Rotated by 4^0	410aa50	-230.16	
		411a50	-148.01	
		413a50	-190.73	
		415a50	-242.25	-242.25
		417a50	-241.64	
		4910a50	-154.95	
		4914a50	-172.52	
		48271a50	-140.75	
		2a50x50	-209.17	
3.	2a50x50	48279a50	-138.09	
	Rotated by 6^0	410aa50	-215.57	
		411a50	-136.04	
		413a50	-176.95	
		415a50	-230.69	-230.69
		417a50	-220.93	
		4910a50	-145.3	
		4914a50	-158.59	
		48271a50	-127.96	
		2a50x50	-164.18	

Table 13 shows the results for an input image rotated by -2° , -4° and -6° . Note that most of input rotated images are not matched in the system database.

Table 13: The unknown input image (2a50x50) is rotated by -2° , -4° and -6° .

Number & name of unknown images	Names of images in the system database	Energy function for each image	The minimum of energy function
1. 2a50x50	48279a50	-82.42	
Rotated by -2°	410aa50	-126.16	
	411a50	-84.72	
	413a50	-108.9	
	415a50	-129.84	
	417a50	-135.01	
	4910a50	-86.88	
	4914a50	-105.2	
	48271a50	-85.08	
	2a50x50	-233.88	-233.88
2. 2a50x50	48279a50	-137.81	
Rotated by -4°	410aa50	-226.91	
	411a50	-138.91	
	413a50	-178.53	
	415a50	-233.58	-233.58
	417a50	-224	
	4910a50	-149.19	
	4914a50	-167.84	
	48271a50	-138.37	
	2a50x50	-211.75	
3. 2a50x50	48279a50	-132.86	
Rotated by -6°	410aa50	-216.32	
	411a50	-136.48	
	413a50	-169.47	
	415a50	-230.02	-230.02
	417a50	-224.94	
	4910a50	-144.55	
	4914a50	-166.42	
	48271a50	-124.19	
	2a50x50	-172.12	

Figure 40 shows the correct convergence by applying experiment 6 on the input image (2a50x50) rotated by 2° in 40 (a) and the converged image that has a match in 40 (b). Figure 41 shows the incorrect convergence by applying experiment 6 on the input image (2a50x50) rotated by -6° in 41 (a) and the converged image that has not a match

with image (415a50) in 41 (b).

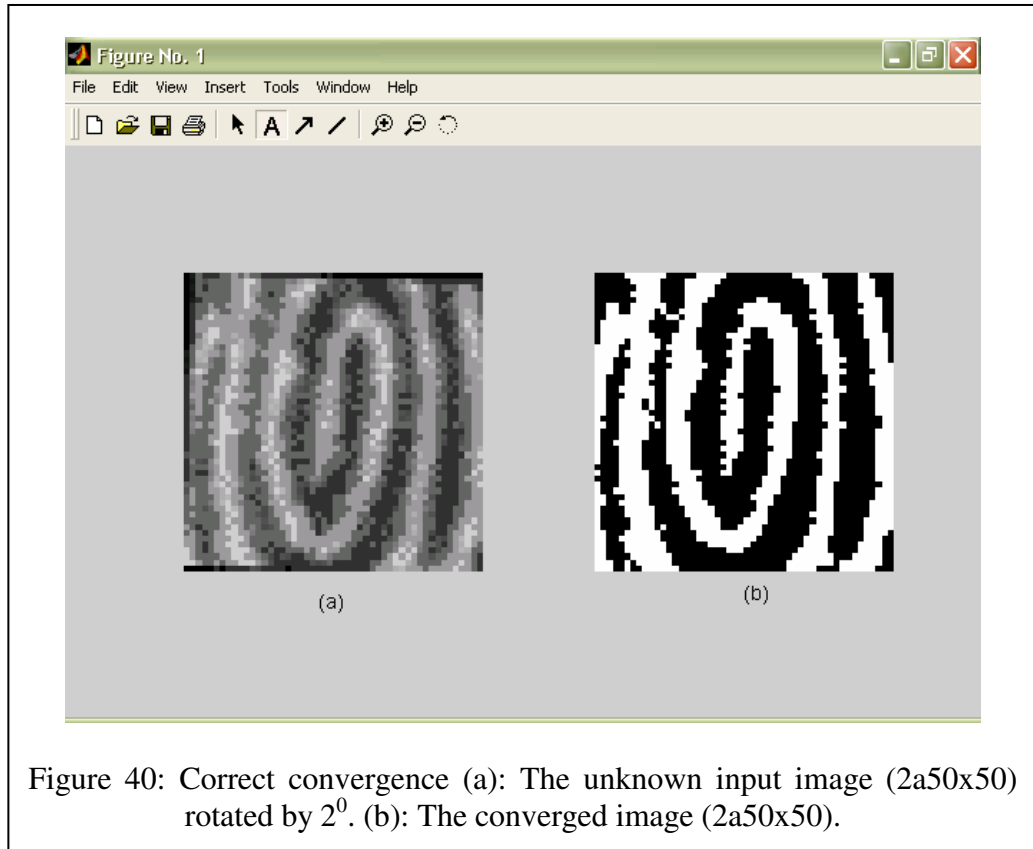


Figure 40: Correct convergence (a): The unknown input image (2a50x50) rotated by 2° . (b): The converged image (2a50x50).

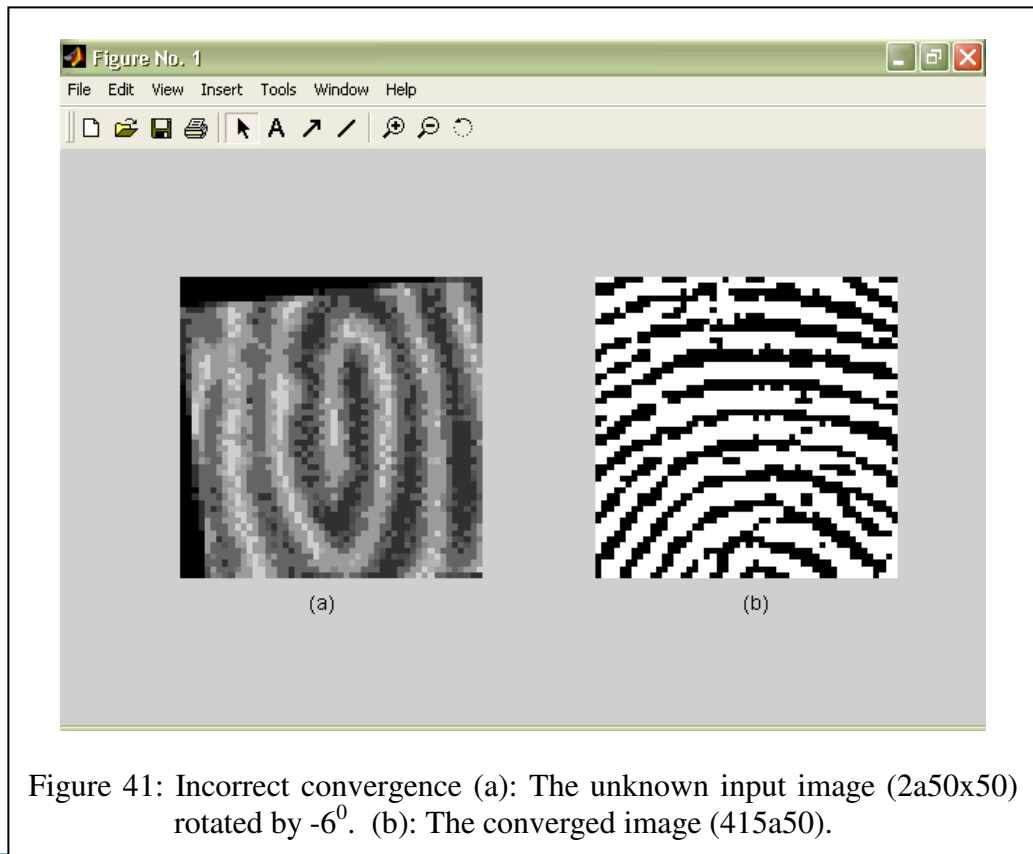


Figure 41: Incorrect convergence (a): The unknown input image (2a50x50) rotated by -6° . (b): The converged image (415a50).

Experiment 7:

The unknown input image is rotated by 30° , 45° , 90° , -30° , -45° and -90° . However, the same rotated image exists in the system database. Table 14 shows the results for the input image rotated by 30° , 45° and 90° . Note that the input rotated images matched in the system database.

Table 14: The unknown input image (4914a50) is rotated by 30° , 45° and 90° .

Number & name of unknown images		Names of images in the system database	Energy function for each image	The minimum of energy function
1.	4914a50 Rotated 30°	48279a50	-103.73	
		410aa50	-153.9	
		411a50	-89.32	
		413a50	-115.96	
		415a50	-142.55	
		417a50	-138.43	
		4910a50	-87.1	
		4914a50 Rotated 30°	-369.3	-369.3
		48271a50	-86.21	
		2a50x50	-100.75	
2.	4914a50 Rotated 45°	48279a50	-100.87	
		410aa50	-152.5	
		411a50	-90.3	
		413a50	-128.94	
		415a50	-124.96	
		417a50	-147.04	
		4910a50	-80.63	
		4914a50 Rotated 45°	-367.38	-367.38
		48271a50	-85.69	
		2a50x50	-102.81	
3.	4914a50 Rotated 90°	48279a50	-91.9	
		410aa50	-139.8	
		411a50	-97.98	
		413a50	-117.56	
		415a50	-143.31	
		417a50	-136.19	
		4910a50	-86.42	
		4914a50 Rotated 90°	-352.6	-352.6
		48271a50	-89.46	
		2a50x50	-105.3	

Table 15 shows the results for an input image rotated by -30° , -45° and -90° . Note that the input rotated images matched in the system database.

Table 15: The unknown input image (4914a50) is rotated by -30° , -45° and -90° .

Number & name of unknown images		Names of images in the system database	Energy function for each image	The minimum of energy function
1.	4914a50	48279a50	-97.62	
	Rotated -30°	410aa50	-155.54	
		411a50	-98.62	
		413a50	-122.77	
		415a50	-157.07	
		417a50	-149.46	
		4910a50	-114.39	
		4914a50 Rotated -30°	-383.41	-383.41
		48271a50	-90.74	
		2a50x50	-122.92	
2.	4914a50	48279a50	-99.52	
	Rotated -45°	410aa50	-149.83	
		411a50	-106.59	
		413a50	-131.44	
		415a50	-165.57	
		417a50	-150.68	
		4910a50	-100.82	
		4914a50 Rotated -45°	-379.91	-379.91
		48271a50	-104.01	
		2a50x50	-111.05	
3.	4914a50	48279a50	-86.48	
	Rotated -90°	410aa50	-131.01	
		411a50	-85.1	
		413a50	-87.82	
		415a50	-146.11	
		417a50	-142.51	
		4910a50	-97.41	
		4914a50 Rotated -90°	-351.95	-351.95
		48271a50	-88.23	
		2a50x50	-104.32	

Figure 42 shows correct convergence by applying experiment 7 on the input image (4914a50 Rotated 45°) in 42 (a) and the correct converged image (4914a50 Rotated 45°) in 42 (b). Figure 43 shows correct convergence by applying experiment 7 on the input image (4914a50 Rotated -90°) in 43 (a) and the correct converged image

(4914a50 Rotated -90^0) in 43 (b).

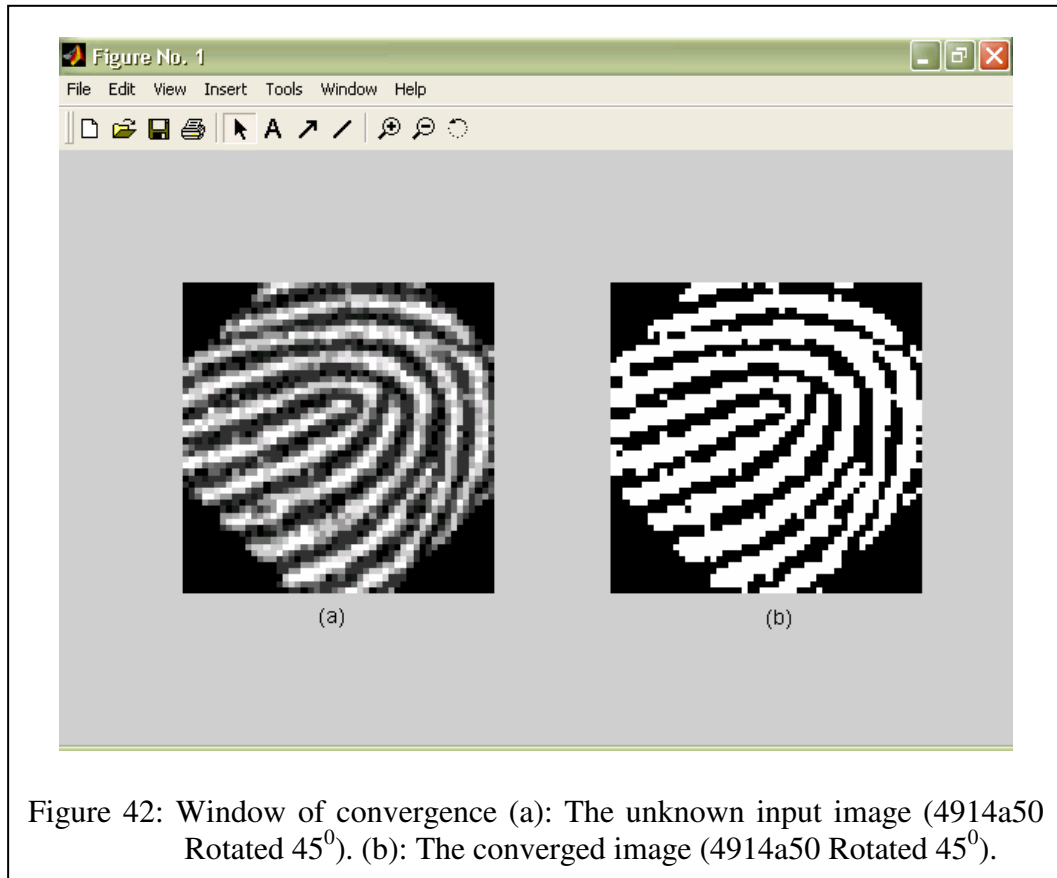


Figure 42: Window of convergence (a): The unknown input image (4914a50 Rotated 45^0). (b): The converged image (4914a50 Rotated 45^0).

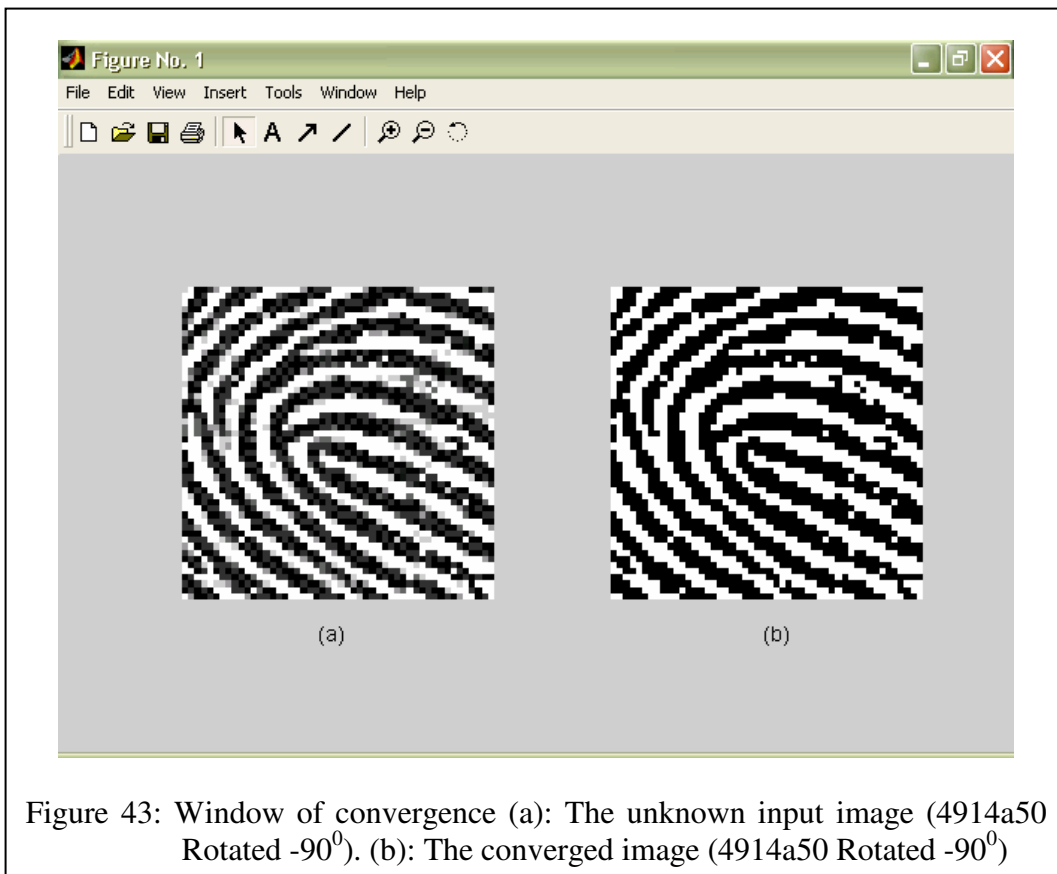


Figure 43: Window of convergence (a): The unknown input image (4914a50 Rotated -90^0). (b): The converged image (4914a50 Rotated -90^0)

Experiment 8:

In this experiment, we measure the time of learning and converging processes in both standard backpropagation algorithm (SBP) and proposed method by using images with different sizes. In this experiment, the net has only learned two images. (The time averages of learning and converging processes are taken for 5 readings). Table 16 shows the times of learning and converging processes in the SBP by using images have the sizes 4x4 and 10x10 pixels. The backpropagation algorithm failed when the input array (input image) size was large (Abdul Abbas, 1996). Therefore, we do not test images that have the sizes 50x50, 128x128 and 256x256 pixels in SBP algorithm.

Table 16: The times of learning and converging processes in the standard backpropagation algorithm (SBP) by using images have the sizes 4x4 and 10x10 pixels.

Size of image (pixel)	The time average of learning process (second)	The time average of converging process (second)	The time of learning + The time of converging (second)
4x4	0.11	0.03	0.14
10x10	1.98	0.04	2.02

Table 17 shows the times of learning and converging processes in the proposed method by using images that have the sizes 4x4, 10x10, 50x50, 128x128 and 256x256 pixels. The results show the times of learning and converging processes in the proposed method are better than the SBP.

Table 17: The times of learning and converging processes in the proposed method by using images have the sizes 4x4, 10x10, 50x50, 128x128 and 256x256 pixels.

Size of image (pixel)	The time average of learning process (second)	The time average of converging process (second)	The time of learning + The time of converging (second)
4x4	0.01	0.004	0.014
10x10	0.04	0.004	0.044
50x50	0.4	0.08	0.48
128x128	3.15	0.56	3.71
256x256	50.98	3.15	54.13

Experiment 9:

In this experiment, we measure the time of identification (the requested time for finding the minimum energy function for the input unknown image) by using database that has the following sizes: 10, 20 30, 40, 50, 60, 70, 80, 90 and 100 images. Table 18 shows the sum time of identification with times of learning and converging which is extracted in experiment 8 for the image that has size 50x50 pixels. The total time for the proposed system (time of learning + time of identification + time of converging) is presented in the last column.

Table 18: The time of identification and the total time for the proposed system by using database with images in the range 10 to 100. (The used image size is 50x50 pixels).

Size of database (image)	The name of input unknown image	The name of converged image	The minimum energy function	The time of identification (second)	The time of learning + The time of converging (second)	The total time of proposed system (second)
10	419	419	-366.52	4.2	0.48	4.68
20	4910	4910	-348.26	8.75	0.48	9.23
30	29956	29956	-342.35	12.61	0.48	13.09
40	48272	48272	-356.58	16.56	0.48	17.04
50	57055	57055	-394.79	21.5	0.48	21.98
60	63341	63341	-305.08	23.19	0.48	23.67
70	99617	99617	-328.38	27.25	0.48	27.73
80	114783	114783	-398.17	32.53	0.48	33.01
90	119428	119428	-368.37	35.38	0.48	35.86
100	157244	157244	-389.6	41.26	0.48	41.74

4.3 Results Analysis and Discussion

In this section, we discuss the previous experimental results of the proposed method compared to the standard backpropagation algorithm (SBP). The SBP is not able to learn large size images. Therefore, the SBP was executed by using ten images size 6x6 pixels. The proposed method was executed on the same images. The comparisons of results is based on the following parameters: the accuracy of matching, the effects of noises, area effects on missing part image, the effects of rotation and the complexity time.

1. The Accuracy of matching

Table 19 compares the Identification Rate (IR) of matching standard backpropagation algorithm (SBP) with the proposed method. Note that IR of the proposed method is 100%, while, the IR of the SBP is lower (90%). The proposed method is better in matching than the SBP. Table 19 lists the results gained from the implementation of both methods on ten images.

Table 19: The accuracy of matching of standard backpropagation algorithm (SBP) and the proposed method.

Input image No.	The SBP Algorithm		The proposed method	
	Identified image No.	IR (%)	Identified image No.	IR (%)
1	1	90%	1	100%
2	2		2	
3	3		3	
4	4		4	
5	5		5	
6	5		6	
7	7		7	
8	8		8	
9	9		9	
10	10		10	

2. The effects of noises

The effect of noises is to degrade the image information. The systems had much difficulty in finding the correct match when fingerprint images contained noises. In this test, different amount of noises added to the image by rates 30%, 50%, 70% and 90%. As it is noticed from Table 20, the results of the IRs for the proposed method are: results (in this experiment) of images of the size 6x6 pixels and the results taken from experiment 3 where the size of image is 50x50 pixels. The table shows that the correct match with existing noises for the proposed method is more efficient than that of SBP.

Table 20: The IRs on input images with noises rates 30%, 50%, 70% and 90% for both standard backpropagation algorithm (SBP) and the proposed method.

Noises rates (%)	Input image No.	The SBP Algorithm		The proposed method		
		Identified image No.	IR (%)	Identified image No.	IR (%)	IR from Exp. 3 (%)
30%	1	5	20%	1	100%	100%
	2	2		2		
	3	5		3		
	4	5		4		
	5	5		5		
50%	1	5	0%	4	60%	100%
	2	5		7		
	3	5		8		
	4	5		9		
	5	2		2		
70%	1	1	60%	2	20%	100%
	2	5		2		
	3	3		2		
	4	3		3		
	5	5		7		
90%	1	5	0%	1	20%	30%
	2	5		7		
	3	5		3		
	4	3		2		
	5	5		3		

From the results in Table 20, the IR of the proposed method taken from experiment 3 is better than other results. Therefore, the proposed method is more efficient for large size images. It has low efficiency for those with small sizes.

3. The effects of area of missing

The missing of information from the fingerprint images affects strongly on the efficiency of the system of the correct recognition. Areas of missing in this test are within ratio 25% and 50%. As it is seen in Table 21, the proposed method has corrected results for IR, which decreased from 100% to 90% and from 80% to 20% as increasing in missing area ratio. While in case of the SBP, the IR is decreased from 40% to 0%. In the proposed method, when missing area ratio is 50%, the second identification rate 90% taken from experiment 4 (The size of image is 50x50 pixels) is better than the 20% of this experiment (The size of image is 6x6 pixels). This means in missing area ratio 50% the proposed method is more efficient for the images that have large sizes.

Table 21: The comparison of input image, which contained missing ratio 25% and 50% for both standard backpropagation algorithm (SBP) and proposed method.

Missing ratio (%)	Input image No.	The SBP Algorithm		The proposed method		
		Identified image No.	IR (%)	Identified image No.	IR (%)	IR from Exp. 4 (%)
25%	1	5	40%	1	80%	100%
	2	4		2		
	3	3		3		
	4	2		4		
	5	5		4		
50%	1	1	0%	1	20%	90%
	2	5		2		
	3	5		8		
	4	10		2		
	5	3		2		

4. The effects of rotation

The results obtained for this test are listed in Table 22. Most of the systems cannot or had much difficulty in finding the correct match when the fingerprint images were rotated by large angles. In this test, the adopted angles is between (-6°) to (6°) for the image with (2°) increment in each case.

It has been noticed that the IR of the SBP is 50%, the result is low. While, in the proposed method the IR is 100%, the result has the higher efficiency. The IR of the proposed method is not affected by the above angles (the size of image is 6×6 pixels), but the IR of the proposed method taken from experiment 6 is affected by the above angles (the size of image is 50×50 pixels). Hence, the proposed method is more efficient for the rotated images that have small sizes and has low efficiency for the large size images.

Table 22: The IRs of input rotated images by both standard backpropagation algorithm (SBP) and proposed method, using different angles.

Rotation angle ($^{\circ}$)	Input image No.	The SBP Algorithm		The proposed method		
		Identified image No.	IR (%)	Identified image No.	IR (%)	IR from Exp. 6
2°	1	1	50%	1	100%	33.33%
4°	2	2		2		
6°	3	5		3		
-2°	4	1		4		
-4°	5	5		5		
-6°	6	5		6		

5. The total time

The backpropagation algorithm was failed as a tool for fingerprint classification; the reasons for this failure are (Abdul Abbas, 1996):

1. Convergence time, when the network iterates, it takes a long time for adjusting the weights.
2. Array size is restricted using the sequential IBM PC machine. This can be overcome when the multilayer perceptron is applied using the main frame machines.

In this study, the backpropagation algorithm, using the proposed method was successful as a tool for fingerprint classification and identification.

Table 23: The comparisons of total complexity time of the standard backpropagation algorithm (SBP), and the proposed method.

The total time of (SBP) Algorithm for 10 images of size 6x6 pixels (Second)	The total time of proposed method for 10 images of size 50x50 pixels (Second)
1312.71	8.28

Table 23 shows the result of the total time obtained by applying the SBP on ten images of size 6x6 pixels. The computed total time (the time of learning for ten images + the time of converging) is 1312.71 second (21.87 minutes). While, the computed total time of the proposed method (the time of learning for ten images + the time of identification + the time of converging) for ten images on image size 50x50 pixels is 8.28 second. The results obtained show that the proposed method is much more

efficiency than the SBP. Hence, the proposed method is better in total time and in large size images.

Table 24: The comparisons of total complexity time of the fuzzy logic and neural networks method (Sagar *et al.*, 1999), and the proposed method.

The time of processing of fuzzy logic and neural networks (SBP) method (Sagar <i>et al.</i> , 1999) (Second)	The total time of proposed method for image of size 50x50 pixels (Second)
6.23	4.68

Table 24 shows the time result of processing obtained by applying the fuzzy logic and neural networks method (Sagar *et al.*, 1999). The time of processing obtained, is 6.23 second. The total time of the proposed method is compared to the time of processing of fuzzy logic and neural networks method, which uses backpropagation algorithm as neural network. The results of time in the two methods show that the proposed method is more efficient.

5. Conclusions and Future Work

5.1 Introduction

In this chapter, we will present the overall view of our conclusions and future work.

5.2 Conclusions

This section summarizes the advantages and disadvantages of applying the fuzzy logic and Backpropagation neural network on fingerprint images recognition.

5.2.1 The Advantages of the proposed method

The major advantages are:

1. The time of learning and convergence for the proposed method is better than SBP (8.2 seconds compared to 1312.7 seconds for the tested samples).
2. The proposed method can handle images of sizes larger than the SBP (up to 128x128 pixels compared to 10x10 pixels, respectively).
3. The proposed method shows accuracy in close to 100% without noise, missing, and rotation while the accuracy of the SBP is 90%. The proposed method is better in 3 scales: by 20% to 80% than the SBP in case of noise, by 20% and 40% than the SBP in case of missing, and by 50% than the SBP in case of rotation.
4. The sizes of the learning weight matrices (W1) and (W2) are constant no matter what size is the input fingerprint image; especially the proposed method is used on net with constant size.
5. Eliminating all the preprocessing and postprocessing (such as the, Thinning, Segmentation and Filtering) lead to the reduction of processing of the fingerprint

images.

5.2.2 The disadvantages of the proposed method

The major disadvantages of the proposed method are:

1. The proposed method has low efficiency for the small size images, for example noise ratios are 30%, 50%, 70%, and 90% for the tested samples.
2. The proposed method has low efficiency for the images that have small sizes (for example at missing area ratio 50%).
3. The proposed method has low efficiency for the rotated images with large sizes.
4. The proposed method failed to recognize the rotated images (with high efficiency); especially the rotation angles are large.

5.3 Future work

The following are the suggested activities for future research:

1. Developing the proposed method for the rotation of images by using large angles, shifting of images and scaling of images to be more efficiency.
2. Combining the methods based on the minutiae in fingerprint images with those based on the texture of images for developing a new method.
3. Comparing the proposed method of converting the digital image to the binary images using the fuzzy logic with the conventional methods to find their strength and weakness.
4. Testing the increment of the vector length, this is used in the proposed method to 3, 4, 5, 6, 7 or 8 pixels to know the effects on the time of learning, identification, and convergence; also for knowing the effects on the size of the learning weight matrices.

REFERENCES

Abdul Abbas, R. E. (1996). **Fingerprint Classification Using Neural Network**. Unpublished master dissertation, University of Technology, Baghdad, Iraq.

AL-Zewary, Mazin Shakir. (1996). **Computer Processing Techniques For Fingerprint Images Identification**. Unpublished master dissertation, University of Baghdad, Baghdad, Iraq.

Amengual, J. C., Juan, A., Perez, J. C., Prat, F., Saez, S. and Vilar, J. M., (1997). **Real-Time minutiae extraction in fingerprint images**. From WWW.cs.unr.edu/~smith/thesis

Ammar, Hany H. Zhen, Xuhui, Nassar, Diao, and Jin, Yingzi. (2000). **Web-Based test bed for fingerprint image comparison**, West Virginia University, Department of computer science and electrical engineering.

Baxes, Gregory A., (1994). **Digital image processing: principles and application**. USA: John wiley & Sons.

Bhanu, Bir, Boshra, Michael and Tan Xuejun, (2000). **Logical templates for feature extraction in Fingerprint images**. From WWW.cs.unr.edu/~smith/thesis

Biometrix ^{8th} **WWW Fp-images**. (n. d.) Retrieved August 10, 2003, from <http://www.biometrix.at/fp-images.zip>

Bonde, Allen. (2000). **Fuzzy Logic Basics**, from [www. Siteterrific.com/fuzzy logic basics.htm](http://www.Siteterrific.com/fuzzy_logic_basics.htm).

Chang, David H., (1999). **Fingerprint Recognition through Circular Sampling**, Center for Imaging Science, Rochester Institute of Technology, Accepted for publication.

Digital-persona ^{1th} **WWW Guide to Fingerprint Identification**. (n. d.) Retrieved January 5, 2003, from [www. rfrancis.com/digital-persona/guidetofingerprintpdf](http://www.rfrancis.com/digital-persona/guidetofingerprintpdf).

Eskaf, K. A. (1999). **Biomedical image recognition neural networks**. Unpublished master dissertation, Arab academy for science and technology and maritime transport.

Espinosa-Duro, Virginia, (2001). **Minutiae detection algorithm for fingerprint recognition.** From WWW.cs.unr.edu/~smith/thesis

Fausett, L., (1994). **Fundamentals of Neural Networks**, (1st ed.). USA: Prentice-Hall.

Halawani, Alaa. H. (2000). **Recognition of Gestures in Arabic Sign Language Using Neuro-Fuzzy Systems.** Unpublished master dissertation, Jordan University of Science and Technology, Irbid, Jordan.

Holecý, Miroslav. (1997). **Application of Neural-Fuzzy Systems in Financial Management.** (master's thesis), <http://neuron-ai.tuke.sk/cig/source/publications/thesis/master-thesis/1997holecy/htm/node26.html>.

Hong, Lin. (1998). **Automatic Personal Identification using Fingerprints.** Unpublished doctoral dissertation, Michigan State University, USA. http://www.bias.csr.unibo.it/research/biolab/lin_thesis.pdf.

Huvanandana, Sanpachai, Kim, Changick and Hwang, Henq-Neng, (2000). **Reliable and fast fingerprint identification for security applications.** From WWW.cs.unr.edu/~smith/thesis

Jain Anil K., and Ross, Arun, (1998). **Fingerprint Mosaicking**, Department of Computer Science, Michigan State University.

Jain Anil K., Ross, Arun and Prabhakar, Salil, (2001). **Fingerprint Matching using Minutiae and Texture Features.** Computer Science & Engineering, Michigan State University. From <http://www.cse.msu.edu/cgi-user/web/tech/document?ID=481>.

Jain, Anil K. and Ross, Arun, (2002) **Learning User-Specific Parameters in a Multibiometric System.** Department of Computer Science & Engineering, Michigan State University.

Jiang, Xudong, Yau, Wei Yun and Ser, Wee, (1999). **Minutiae extraction by adaptive tracing the gray level ridge of the fingerprint image.** From WWW.cs.unr.edu/~smith/thesis

Jiangtao, Meng and Shaojun, Wei. (2001). **Automatic fingerprint identification system for small sized database.** From WWW.cs.unr.edu/~smith/thesis

Jung, D.-W. and Park, R.-H., (2001). **Robust fingerprint identification based on hybrid pattern recognition methods.** From WWW.cs.unr.edu/~smith/thesis

Kasaei, S., Dericge, S. and Boashash, B., (1997). **Fingerprint feature extraction using block-direction on reconstructed images.** From WWW.cs.unr.edu/~smith/thesis

Kosko, Bart, (1992). **Neural networks and fuzzy systems: A dynamical systems approach to machine intelligence,** USA: prentice-Hall international.

Kulkarni. A. D., (2001). **Computer vision and fuzzy-Neural systems,** USA: Prentice Hall.

Lee, Chih-Jen and Wang, Sheng-De, (1999). **Fingerprint feature extraction using Gabor filters.** From WWW.cs.unr.edu/~smith/thesis

Lippmann, Richard P. (1987). An Introduction to Computing with Neural Nets. **IEEE,** April, 4-22.

Liu, Jinxiang, Huang, Zhongyang and Chan, Kap Luk, (2000). **Direct minutiae extraction form gray-level fingerprint image by relationship examination.** From WWW.cs.unr.edu/~smith/thesis

Luo, Xiping, Tian, Jie and Wu, Yan, (2000). **A minutiae algorithm in fingerprint verification.** From WWW.cs.unr.edu/~smith/thesis

Maio, D. and Maltoni, D. (1997) Direct Gray-Scale Minutiae Detection in Fingerprints. **IEEE tPAMI,** vol. 19, No. 1. from <http://www.bias.csr.unibo.it/research/biolab/minextr.html>.

Maltoni, D., Maio, D., Jain, A. K. and Prabhakar, S., (2003). **Handbook of Fingerprint Recognition,** New York: Springer.

Neurotchnologija ^{3th} **WWW Fingercell_Sample_DB.** (n. d.) Retrieved May 25, 2004, from www.neurotchnologija.com/download/fingercell_sample_DB.zip.

Pankanti, Sharath, Prabhakar, Salil and Jain, Anil K., (2002). **On the Individuality of Fingerprints.** Computer Science & Engineering, Michigan State University.

Prabhakar, S. (2001). **Fingerprint classification and matching using a filterbank**. Unpublished master dissertation, Michigan state university, USA.

Prabhakar, Salil, Jain, Anil K. and Pankanti, Sharath, (2002). **Learning Fingerprint Minutiae Location and Type**, Computer Science & Engineering, Michigan State University.

Radhi, Makki Jasim. (2003). **Enhancing of Image Recognition using Neural Network and Genetic Algorithm**. Unpublished master dissertation, University of Jordan, Amman, Jordan.

Ross, Arun, Jain, Anil K. and Reisman, James, (2002). **A Hybrid Fingerprint Matcher**, Computer Science & Engineering, Michigan State University.

Ruan, Da., (1997). **Intelligent hybrid systems: Fuzzy logic, Neural Networks and Genetic Algorithms**, USA: Kluwer Academic publishers.

Rzempoluck, Edward J., (1998). **Neural Network Data Analysis Using Simulnet**, (2nd ed.). New York: Springer-verlag New York.

Sagar, V. K., Ngo, D. B. L. and Foo, K. C. K., (1995). **Fuzzy feature selection for fingerprint identification**. From WWW.cs.unr.edu/~smith/thesis

Sagar, Vijay Kumar and Beng, Koh Jit, (1999). **Fingerprint feature extraction by fuzzy logic and neural networks**. From WWW.cs.unr.edu/~smith/thesis

Simon-Zorita, D., Ortega-Garcia, J., Cruz-Llanas, S., and Gonzalez-Rwdriguez, J., (2001). **Minutiae extraction scheme for fingerprint recognition systems**. From WWW.cs.unr.edu/~smith/thesis

Sun, Xiao and Ai, Zhuming, (1996). **Automatic feature extraction and recognition of fingerprint images**. From WWW.cs.unr.edu/~smith/thesis

Tamimi, H. H. (1999). **Using Artificial Neural Networks and Fuzzy Logic for Controlling the Compression of Digital Images**. Unpublished master dissertation, University of Jordan, Amman, Jordan.

Tanaka, Kazuo, (1997). **An Introduction to Fuzzy Logic for Practical applications**, USA: Rassel.

Umbaugh, Scott E., (1998). **Computer Vision and Image Processing: A Practical Approach Using CVIP Tools**, USA: Prentice Hall PTR.

Vallurn. B. Rao, and Hayoriva, V. Rao., (1996). **C++ Neural Network and Fuzzy Logic**, (2nd ed.). USA: M & T publishing.

Wilson, Charles L., Watson, C. I. and Paek, Eung Gi, (1997). Combined Optical and Neural Network Fingerprint Matching, **proceedings of SPIE VIII**, Vol. 3073.

Wu, Jian-Kang, (1994). **Neural networks and simulation methods**, New York: Marcel Dekker.

Yen, J. Langari R and Zadeh, L. A., (1995). **Industrial Applications of Fuzzy Logic and Intelligent systems**, New York: The Institute of Electrical and Electronics Engineers.

تشخيص صورة بصمة الاصبع باستخدام المنطق الضبابي وتقانة الشبكات العصبية

اعداد
احمد هاشم حسين

المشرف
الدكتور احمد الشرايعه

ملخص

كون بصمة الاصبع البشرية وحيدة، تعتبر من اكثر الميزات الموثوقة لتشخيص الشخصية. وبسبب الحجم الكبير من بصمات الاصابع والتقدم الحديث بتكنولوجيا الحاسوب ، هناك اهتمام متزايد بتشخيص بصمات الاصابع المباشر .

ولتشخيص صور بصمات الاصابع، استخدمنا المنطق الضبابي وتقنيات شبكة الانتشار العكسي العصبية، أستعمل المنطق الضبابي لتحويل صورة بصمة الاصبع الرقمية الى صورة ثنائية، بينما أستعملت شبكة الانتشار العكسي لعملية التشخيص. في هذه الدراسة، يُستعمل حجم الشبكة الاصغر. وعمليات التعليم والاسترجاع تُعمل لاجزاء من صور بصمات الاصابع وتجنب تعليم واسترجاع الاجزاء المتشابهة. بالاضافة الى ذلك، تُستعمل دالة الطاقة مع خوارزمية الانتشار العكسي لعملية الاسترجاع.

باستعمال خوارزمية الانتشار العكسي مع دالة الطاقة على اجزاء من صورة، نحن حصلنا نتائج مشجعة بمصطلحات وقت التعليم، وقت الاسترجاع، وحجم صورة بصمة الاصبع

المدخلة. وكان وقت التعليم والاسترجاع للبيانات المُختبرة 8.2 ثانية مقارنة الى 1312.7 ثانية عند استعمال خوارزمية الانتشار العكسي فقط. والطريقة المقترحة استطاعت ان تعالج صور من احجام اكبر (تم اختبار حجم اعلى من 128×128 نقاط) من خوارزمية الانتشار العكسي (اقل من 10×10 نقاط). ايضاً، الطريقة المقترحة تبين دقة قريبة من 100% بدون تشويه، فقدان، وتدوير. والطريقة المقترحة تكون احسن بـ 20% الى 80% من خوارزمية الانتشار العكسي بحالة التشويه. وهي تكون احسن بـ 20% و 40% من خوارزمية الانتشار العكسي بحالة الفقدان. واحسن بـ 50% من خوارزمية الانتشار العكسي بحالة التدوير للعينات المدروسة.