A NEURAL NETWORK MODEL FOR INVARIANT PATTERN RECOGNITION

BY

MD. MONIRUL ISLAM

A THESIS SUBMITTED TO THE DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING, BUET, IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE OF MASTER OF SCIENCE IN ENGINEERING.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
BANGLADESH UNIVERSITY OF ENGINEERING AND TECHNOLOGY

APRIL 1996.

The thesis titled "A NEURAL NETWORK MODEL FOR INVARIANT PATTERN RECOGNITION" submitted by Md. Monirul Islam, Roll No. 911802F, Session 1989-1990 to the Computer Science and Engineering Department of B.U.E.T has been accepted as satisfactory for partial fulfilment of requirements for the degree of Master of Science in Engineering (Computer Science and Engineering).

Board of Examiners

M. S. Alam.

1. Dr. Md. Shamsul Alam — Chairman and Supervisor 7-4-96
Professor
Department of Computer Science & Engineering
BUET, Dhaka-1000, Bangladesh

J. Kamruzzaman

2. Dr. Joarder Kamruzzaman — Member and Co-supervisor
Assistant Professor
Department of Electrical & Electronic Engineering
BUET, Dhaka-1000, Bangladesh

3. Dr. Mohammad Kaykobad — Member
Associate Professor
Department of Computer Science & Engineering
BUET, Dhaka-1000, Bangladesh

Saiful Islam 7.4.96

4. Dr. Saiful Islam — Member (Ex-officio)
Professor & Head
Department of Computer Science & Engineering
BUET, Dhaka-1000, Bangladesh

M. Q. Ahsan

5. Dr. Quamrul Ahsan — Member (External)
Professor
Department of Electrical & Electronic Engineering
BUET, Dhaka-1000, Bangladesh

Acknowledgements

The author wishes to express his profound gratitude to Dr. Md. Shamsul Alam, Professor, Department of Computer Science & Engineering, BUET, Dhaka, for his constant encouragement, active cooperation and suggestions in every conceivable way.

The author wishes to express his profound gratitude to his thesis co-supervisor, Dr. Joarder Kamruzzaman, Assistant Professor, Department of Electrical & Electronic Engineering, BUET, Dhaka, for his valuable guidance, encouragement, constructive suggestion and kind cooperation throughout all phases of his thesis work.

Thanks are given to Dr. Saiful Islam, Professor and Head, Department of Computer Science & Engineering, BUET, for his encouragement and cooperation.

The author wishes to convey his sincere thanks and deep appreciation to Dr. M. Kaykobad, Associate Professor, Department of Computer Science & Engineering, BUET, Dhaka, for his cooperative comments, valuable suggestions which contributed a lot to his thesis.

The author would like to express his gratitude to Md. Abdus Sattar, Assistant Professor, Department of Computer Science & Engineering, BUET, Dhaka, for his valuable suggestion and encouragement. Thanks are also offer to the microcomputer laboratory personnel of Department of Computer Science & Engineering, BUET, Dhaka.

ABSTRACT

A pattern recognition system using Artificial Neural Network (ANN) classifier is proposed. The system is intended to recognize translated, rotated and scaled versions of the exemplar patterns. The model proposed for this system consists of two parts. The first is a preprocessor and the second is an ANN classifier. Preprocessing is done in two stages. In the first stage, projection from each active bit of the pattern is taken in such a way that for any rotated or scaled version of the exemplar pattern, the projected values become cyclically shifted of those of the exemplar pattern. For translated version of the exemplar pattern, the projected values remain same. The second stage performs Rapid Transform (RT) on the projected values. Thus, for an ideal case when rotated or scaled pattern is assumed to be noise free and distortionless, the outputs of the preprocessor are invariant to rotation, scaling and translation. However, in practical case, rotation and scaling always insert some amount of noise in the input pattern. Therefore, the preprocessed outputs in response to a rotated or scaled pattern appear to be somewhat different from those of exemplar pattern. The second part of the system is an ANN classifier trained by backpropagation algorithm chosen especially for its good ability to deal with variation of inputs within the same category. The outputs of the preprocessor are then fed into the ANN classifier. In spite of the variation present in the preprocessed outputs due to rotation and/or scaling, the ANN classifier is expected to classify the input pattern correctly. The proposed model is tested extensively with ten numeric digits (0 ~ 9). With these patterns, the proposed model achieves considerably good degree of invariance to rotation and scaling. The performance of the system also depends on the number of input and hidden units of ANN classifier. The effect of classifier's size on the performance of the system is also studied.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER-ONE

INTRODUCTION

- GENERAL INTRODUCTION
- LITERATURE REVIEW
- EARLY WORKS ON PATTERN RECOGNITION
- RECENT WORKS
- NEURAL NETWORK BASED PATTERN RECOGNITION SYSTEM
- APPLICATION OF PATTERN RECOGNITION TECHNOLOGY
- OBJECTIVE OF THIS WORK
- INTRODUCTION TO THE THESIS

1.0 General Introduction

A pattern is a quantitative or structural description of an object or some other entity of interest [1]. Recognition is regarded as a basic attribute of human beings and other living organisms. According to the nature of the patterns to be recognized, we may divide our acts of recognition into two major types: the recognition of concrete items and the recognition of abstract items. We recognize characters, pictures, music and object around us. This process may be referred to as sensory recognition which includes visual and aural pattern recognition. On the other hand, we can recognize an old argument or a solution to a problem without restoring to external stimuli. This process involves the recognition of abstract items and can be termed conceptual 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 details. For example, weather prediction can be treated as a pattern recognition problem. The received input data are in the form of weather maps. The system interprets these maps by extracting the significant features and makes a forecast based on these features. Medical diagnosis can also be considered as a pattern recognition problem. The symptoms serve as the input data to the recognition system which identifies the disease by analysis of the input data. The set of patterns to be recognized could be a set of physical objects. A pattern class is a set of patterns that share some common properties. The number of pattern classes is often determined by the particular application in mind. For example, in many information systems we need a machine to recognize various fonts of printed letters and numerals. In this case, there are 62 pattern classes representing 26 upper-case letters, 26 lower-case letters, and 10 numerals. The different fonts and styles of a particular letter or numeral form the patterns in that pattern class. In some problems, the exact number of classes may not be known initially, and it may have to be determined from the observations of many representative patterns. In this case, we would like to detect the possibility of having new classes of patterns as we observe more and more patterns. Human beings perform the task of pattern recognition in almost every instant of their working lives. The subject matter of pattern recognition by machines deals with techniques for assigning patterns to their respective classes automatically.

The study of pattern recognition problems may be logically divided into two major categories :

1. The study of the pattern recognition capability of human beings and other living organisms.
2. The development of underlying theory and practical techniques for machine implementation of a given recognition task.

The first subject area falls in the domain of such disciplines as psychology and biology. The second area is in the domain of engineering, computer science and applied mathematics.

1.1 Literature Review

This section presents brief descriptions of some of the important research works including automatic system design in the area of pattern recognition.

1.1.1 Early Works on Pattern Recognition

A notable early attempt in the area of pattern recognition research is by Grimsdale et al. [2] in 1958. In their method, the input pattern obtained by a flying spot scanner is described in terms of length and slope of straight line segments, and length and curvature of curved segments. This description is compared with that of the prototype stored in the computer in order to reach the proper decision about the identity of the unknown pattern.

Important works in the pattern recognition area have been done by L.P. Horwitz [3] in 1960. The basic idea of his techniques is that of registration invariance, i.e., all patterns are described in such a way that shifting translationally in any direction does not affect any observed parameter. This mode of description has strong implications for the engineering side of the situation. A machine designed to deal with this form of information does not have to search in a character field for the precise location of the character.

S.S. Yau and C.C. Yang in 1966 developed a simple template-matching pattern recognition technique by using any general purpose associative memory [4]. The input patterns for recognitions may have wide variations, provided that the distinct features of individual pattern classes can be extracted. Each pattern class was allowed to have deviations in size, style, orientation, etc. within certain limits.

V.K. Govindan and A.P. Shivaprasad [5] described a system for recognizing members of a sixteen character alphabet (numerals and some miscellaneous printed symbols). The system had been tested via a computer program and has successfully identified samples of printed characters in the presence of background noise. As described in [5], this method does not permit variations in the size and proportion of the input characters.

1.2 Recent Works

The research concerning pattern recognition has always been an attractive field for the scientists. Scientists from a diverse group of disciplines have contributed to this field. Some areas that are directly or indirectly involved in this field are computer science, computer vision, robotics, image analysis and processing, linguistics etc. The background of this remarkable discipline of science has been described in the previous section. Some of the important recent works are mentioned in this section for the purpose of an introduction to the vast number of works that have already been accumulated in this field.

L.N. Kanal [6] selectively surveyed contributions to major topics in pattern recognition during the period 1968 to 1974. This paper also provides a very useful bibliography on representative books and surveys on pattern recognition during the above mentioned period. Theoretical models for automatic pattern recognition are contrasted with practical design methodology. This paper selectively discussed the research contributions to statistical and structural pattern recognition including contributions to error estimation and experimental design of pattern classifiers. This paper concludes with representative set of applications of pattern recognition technology.

Unger [7] observed that for any alphabet there must exist at least one finite set of characteristics that can be used to distinguish among the members. He suggested there must exist a set of yes or no questions such that if these questions are answered with respect to any given figure then there will be only one member of the alphabet to which this figure can belong.

Perotto [8] described a method called morphotopological method for character recognition. In this method characters are described in invariant terms with considerations of topological and morphological type. This method have made it possible to read English numerical characters having strong variations in form, size and position.

A matrix of simple identical intercommunicating cells is the core of a pattern classifier described by Glucksman [9]. A test sequencer tests the matrix, on which the pattern is mapped, by following a binary decision tree. The result of each test determines which of the two tests will follow. Each test eliminates classes of patterns from the total set of classes. The process leads to one final class at the end of a sequence of such tests.

Fu, a pioneer in pattern recognition field, proposed different methods of recognition techniques. In [11], the non-parametric as well as parametric (Bayes) classification methods have been discussed in detail. The paper also includes a discussion on sequential decision mode for pattern classification. In [12], Fu explored the topics on syntactic approach and provided detailed discussions regarding selection on pattern primitives, pattern grammar including special grammar and syntax analysis.

I. Guyon, P. Albrecht, Y. Le Cun, J. Denker and W. Hubbard [15] described a system which can recognize digits and uppercase letters hand printed on a touch terminal. A character is input as a sequence of [x(t),y(t)] coordinates, subjected to a very simple preprocessing, and then classified by a trainable neural network. The network was trained on a set of 12,000 digits and uppercase letters, from approximately 250 different writers, and tested on 2500 such characters from other writers.

S.R. Ramesh [16] described an algorithm for the recognition of English alphabet with a great deal of flexibility. In [16], a scheme is suggested for adapting the syntactic and structural approach to detect characters. The character can be considered as a complex pattern. This pattern can be segmented into intermediate patterns which in turn can be separated into sub-patterns. Each character is represented by certain pattern primitives. These pattern primitives are extracted and analyzed to infer the character.

Trofinn Taxt, Jorunn B. Olafsdottir and Morten Daehlen [17] described statistical methods for recognizing isolated, hand printed symbols directly from raster images of documents such as technical drawings and maps. These methods for recognizing symbols of known or unknown rotation avoid the traditional thinning and vectorization steps in the recognition process. In [17], the outer pixel boundary of an isolated symbol candidate in the binary raster image is considered as a simple closed curve. This curve is then approximated by a parametric spline and coordinates along the spline curve, an elliptic Fourier expansion of the Fourier expansion due to Zahn and Roskies [19]. Curvature values and coordinates along the spline curves or the coefficients of the Fourier expansion are then used as on separate training and test sets consisting of the digits 0 to 9 and all lower-case letters.

In [18], Alireza Khotanzad and Jiin-Her developed a neural network based approach for classification of images represented by translation, scale and rotation invariant features. Two types of features were used: moment invariants derived from geometrical moments of the image, and Zernike moment features. Zernike moments are the mapping of the image onto a set of complex orthogonal polynomials. The utilized network was a multilayer perceptron classifier with one hidden layer. The back propagation learning algorithm was used for its training.

M.A. Sattar and S.M. Rahman [13] discussed different problems of recognition of printed Bangla characters by applying the template matching method. They discussed the optimum number of training set for designing the template and mismatch threshold to be used to avoid the problem of misclassification. However, the problem of invariant recognition was not addressed..

Mia [14] used the same syntactic pattern recognition technique in the case of Bangla printed characters. He constructed numerical codes from the relationship among the strokes representing the structure of the character. There were several groups of characters which showed the same code due to structural similarity among them. These characters were later distinguished by comparing the stroke ratio of a specific stroke for different members of the same group. But the thesis did not discuss the invariance problem.

1.3 Neural Network based recognition system

The most fundamental problem in the area of pattern recognition is associated with scaling, translation and rotation of the patterns, and the design of invariant recognition system possessing high recognition ability but of less complexity and possible minimum size is of immense importance. For invariant recognition of plenary object (binary image), many of the algorithms simplify the problem by reducing it to a shape or contour recognition. Classical approaches are Fourier descriptor [19], moment invariant [20], stochastic models [21] etc. Methods usually adopted for classification by these approaches are nearest neighbor, Bayesian decision theory.

Recent developments in the field of artificial neural networks have provided potential alternatives to the traditional techniques of pattern recognition. Artificial neural nets are inspired from the studies of biological nervous system, and composed of many simple nonlinear computational elements called neurons which are connected by links with variable weights. The inherent parallelism of these networks allow rapid pursuit of many hypothesis in parallel resulting in high computational rates.

The problem of invariant pattern recognition using layered neural networks was addressed by Widrow et al. [22]. Widrow's ADALINE is proposed to build an invariant pattern recognition system which needs large number of slabs of neuron, each slab being invariant to specific degree of translation, rotation or scaling. This makes the resulting network very large and complex since the network must accommodate all the possible degrees of translation, rotation and scales. Neocognitron developed by Fukushima [23] is not so effective for rotation invariance. Higher order neural networks are also proposed to achieve invariance [24] but use of higher order increases the number of connections astronomically and makes its implementation for large scale image planes extremely difficult.

In this work, the aim is to model a neural network based rotation, scaling and translation invariant pattern recognition system which will be of reasonable size, require less complexity of design and possess good recognition ability.

1.4 Applications of Pattern Recognition Technology

Pattern recognition has many practical applications. The followings are some of the applications for which pattern recognition have been used.

1. Use as a telecommunication aid for deaf, in air line reservation, in postal department for postal address reading (both handwritten and printed postal codes) and for medical diagnosis.

2. For use in customer billing as in telephone exchange billing system, order data logging, automatic finger print identification, as an automatic inspection system.

3. In automated cartography, metallurgical industries, computer assisted forensic linguist system, electronic mail, information units and libraries and for facsimile.

4. For direct processing of documents as a multipurpose document reader for large scale data processing, as a micro-film reader data input system, for high speed data entry, for changing text/graphics into a computer readable form, as electronic page reader to handle large volume of mail.

1.5 Objective of this work

Pattern recognition techniques can open a new way of realizing the dream of the natural mode communication between man and machine. Vigorous research work is going on in different fields like speech processing and recognition, finger print identification and character recognition. The objective of the present research is to develop a neural network based model for rotation, scaling and translation invariant pattern recognition system which will also be able to generalize well on the patterns outside the training set.

The proposed model consists of two parts. The first is a preprocessor and the second is a multilayer neural net classifier. The performance of the proposed system depends on the performance of the preprocessor and the neural net classifier. Preprocessing is done in two stages. In the first stage, projection from the input pattern will be taken in such a way that, for any rotated or scaled input pattern, the projected values reduce to cyclically shifted version of those of the standard patterns. The second stage performs Rapid Transform (RT) on the projected values. The outputs of the preprocessor are then fed to a multilayer neural net classifier. The number of input and hidden units of neural net classifier play a significant role in determining its performance. The effect of classifier's size on the performance of the proposed system is also a part of the objective of this work.

1.6 Introduction to the thesis

The remaining part of this thesis is divided into four chapters. In chapter 2, different models of pattern recognition are presented. Chapter 3 deals with proposed model of the pattern recognition. The proposed model consists of two parts. The first is a preprocessor and the second is a neural net classifier. A detailed description of the preprocessor is also given in this chapter. The neural network classifier is presented in chapter 4. The simulation results of the proposed model is given in chapter 5. Conclusions of the obtained results are presented in chapter 6. Some suggestion for future works are also presented in this chapter.

CHAPTER-TWO

MODELS OF PATTERN RECOGNITION

- INTRODUCTION
- FUNDAMENTAL PROBLEMS IN PATTERN RECOGNITION
- DESIGN CONCEPTS AND METHODOLOGIES
- COMMON PROPERTY CONCEPT
- MEMBERSHIP-ROSTER CONCEPT
- CLUSTERING CONCEPT
- METHODOLOGY OF DESIGN
- HEURISTIC METHODS
- MATHEMATICAL METHODS
- LINGUISTIC OR SYNTACTIC METHODS

2.0 Introduction

The goal of a pattern classification is to assign a physical objects, events, or phenomenon to one of the prespecified classes (also called categories). Human beings and animals have performed these tasks since the beginning of their existence. Recognition of concrete patterns by human beings can be considered as a psychophysiological problem which involves a relationship between a person and physical stimulus. When a person perceives a pattern, he makes an inductive interference and associates the perception with some general concepts or clues which he has derived from his past experiences [1].

An obvious but simple-minded solution to a pattern recognition problem is to perform a number of simple tests on the individual input patterns in order to extract the features of each pattern class. Such tests should be sufficient to distinguish between permissible input patterns that belong to different classes. However, there is no general theory to determine which of all possible tests on the real world should be applied to the input patterns. Too few or poorly chosen tests will not characterize the input patterns sufficiently to permit categorization into their respective pattern classes.

2.1. Fundamental Problems in Pattern Recognition System Design

The design of an automatic pattern recognition system generally involves several major problem areas. The first one is concerned with the presentation of the input data which can be measured from the objects to be recognized. This is called the sensing problem. Each measured quantity describes a characteristic of the pattern or object. If, for example, a pattern is a combination of alphanumeric characters, then the following equation can be used for the representation. If the grid has n elements, the measurements can be arranged in the form of a measurement or pattern vector.

$$X = \begin{bmatrix} X_1 \\ X_2 \\ X_3 \\ \cdot \\ \cdot \\ X_n \end{bmatrix} \tag{2.1}$$

This is only one form of representation. There can be numerous presentation schemes which will have their own pros and cons. But the important concept is that a suitable representation scheme chosen before the real processing commences is a key to successful implementation of the scheme.

The second problem in the pattern recognition is concerned with the extraction of characteristic features or attributes from the received input data and the reduction of the dimensionality of the pattern vectors. This is called the feature extraction problem. The features of a pattern class are the characteristic attributes common to all patterns belonging to that class. Such features are called the intraset features. The features, on the other hand, which characterizes the difference between two sets are called intraset features. The elements of intraset features which are common to all pattern classes under consideration carry no discriminatory information and can be ignored. Automatic recognition may be reduced to a simple matching process or a table look up scheme. However, in most pattern recognition schemes which arise in practice, the determination of a complete set of discriminatory features is extremely difficult.

The third problem in pattern recognition system design involves the determination of optional decision procedures, which are needed in the identification and classification process. After the observed data form patterns to be recognized have been expressed in the form of pattern points or measurement vectors in the pattern space, the decision has to be taken about the particular pattern class to which this particular pattern may belong.

If the machine is to be designed to recognize M different pattern classes, denoted by w_1, w_2, w_3, ..., w_M, then the pattern space may be considered to consist of M regions, each of which encloses the pattern points of a class. The recognition problem can now be viewed as that of generating the decision boundaries which separate the M pattern classes on the basis of the observed measurement vectors. If decision boundaries are defined, for example by decision functions, $d_1(x)$, $d_2(x)$, ..., $d_M(x)$, then these functions are scaler and single valued function of pattern x. If $d_i(x)>d_j(x)$ for $j=1, 2, ..., M$, and $j\neq i$ the pattern x belong to pattern class w_i. In other words, if the ith decision function, $d_i(x)$, has the largest value for pattern x, then $x \epsilon w_i$. Such an automatic classifier scheme using a decision making process is illustrated conceptually in block diagram of Fig. 2.1, in which DFG denotes decision function generator.

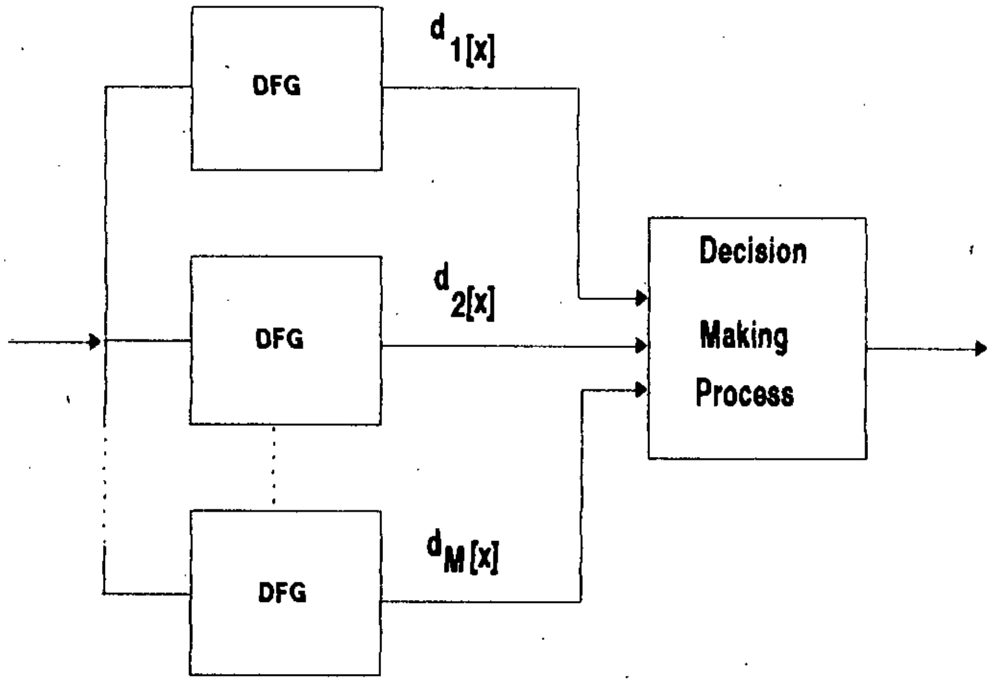


Fig. 2.1 Block diagram of a pattern classifier.

2.2 Design Concepts and Methodologies

The design concepts for automatic pattern recognition are motivated by the ways in which pattern classes are characterized and defined. Depending on the particular target that has to be recognized, the properties or group of properties that are associated with this particular target should serve as the basis of recognition process. There are three basic design concepts. These concepts are discussed here.

2.2.1 Common Property Concept

Characterization of pattern class by common properties shared by all of its members suggests automatic pattern recognition via the detection and processing of similar features. The basic assumption in this method is that pattern belonging to the same class possesses some properties or attributes which reflect similarities among the members. These common properties can be stored in the recognition system. When an unknown pattern is observed by the system, the features are extracted and some times coded and are compared with the stored features. The recognition scheme will classify the new pattern as belonging to the pattern class with similar features.

2.2.2 Membership-roster Concept

Characterization of a pattern class by a roster of its member suggests automatic pattern recognition by template matching. The set of patterns belonging to the same pattern class is stored in the pattern recognition system. When an unknown pattern is shown to the system, it is compared with the stored patterns one by one. The pattern recognition system classifies the input pattern class if it matches one of the stored patterns belonging to that pattern class. Unfortunately, the membership-roster approach will work satisfactorily under the condition of nearly perfect pattern samples.

2.2.3 Clustering Concept

When the patterns of a class are vectors whose components are real numbers, a pattern class can be characterized by its clustering properties in the pattern space. The design of a pattern recognition system based on this general concept is guided by the relative geometrical arrangement of the various pattern clusters [1]. If the classes are characterized by the cluster that are far apart, simple recognition techniques such as minimum distance classifiers can be used. When the cluster overlap, however, it becomes necessary to utilize more sophisticated techniques for partitioning the pattern space.

2.3 Methodology of Design

The basic concepts that have been presented above can be implemented by three different principal categories of methodology: heuristic, mathematical and linguistic or syntactic. It is also possible to devise hybrid models based on these concepts.

2.3.1 Heuristic Methods

The heuristic method is based on human intuition and experience, making use of membership-roster and common property concepts. A system designed using this principle generally consists of a set of ad hoc procedures developed for specialized recognition tasks. However, the structure and performance of the heuristic system will depend to a large degree on the cleverness and the experience of the system engineers.

2.3.2 Mathematical Models

The mathematical approach is based on classification rules that are formulated and derived in a mathematical framework, making use of the common property and clustering concepts. This is in contrast to the heuristic approach, in which decisions are based on ad hoc rules. The mathematical approach may be divided into two subgroups, deterministic and statistical. The deterministic approach is based on the mathematical framework which does not employ explicitly the statistical properties of the pattern classes under consideration.

The statistical approach is based on mathematical classification rules which are formulated and derived in statistical framework. The design of statistical pattern classifiers are generally based on Bayes classification and its variations. This rule yields an optimal classifier when the probability density function of each pattern population and the probability of occurrence of each pattern class are known.

2.3.3 Linguistic or Syntactic Methods

Characterization of patterns by primitive elements or subpatterns and their relationships suggests automatic pattern recognition by the linguistic or syntactic approach, making use of common- property concept. A pattern can be described by a hierarchical structure of subpatterns analogous to the syntactic structure of languages. This permits application of formal language theory to the pattern recognition problem. A pattern grammar is considered as consisting of finite set of elements called variables, primitives, and productions. The rules of production determine the type of grammar. Among the most studied grammars are regular grammars, context free grammars, and context sensitive grammars. The essence of this approach lies in the selection of pattern primitives, the assembling of the primitives and their relationships into pattern grammars.

CHAPTER-THREE

PROPOSED MODEL OF PATTERN RECOGNITION

- **INTRODUCTION**
- **THE PREPROCESSOR**
- **T-BLOCK**
- **S-BLOCK**
- **THE CLASSIFIER**

3.0 Introduction

Pattern recognition systems are expected to automatically classify, describe, or cluster complex patterns or objects based on their measured properties or features. Design of a pattern recognition system essentially involves the following three steps: (i) data acquisition and preprocessing, (ii) representation/feature extraction, and (iii) decision making or clustering.

Pattern recognition deals with the identification or interpretation of the patterns in images. It aims to extract information about the pattern and/or classify its contents. A computer vision system in some way must incorporate pattern recognition capability. Depending on the nature of the patterns and applications, the recognition process can be straightforward or an extremely difficult task. The problem attempted here is a small subclass of the general pattern recognition problem. The pattern to be classified are digitized binary valued 2-D patterns. This representation of the patterns is defined as the pixel-map form of the pattern.

The proposed model shown in Fig. 3.1 presents a pattern classification system which is intended to achieve rotational, scaling, and translational invariance. This system is based on initially preprocessing the input - a binary 2-D pattern and then classifying the preprocessed output by the classifier.

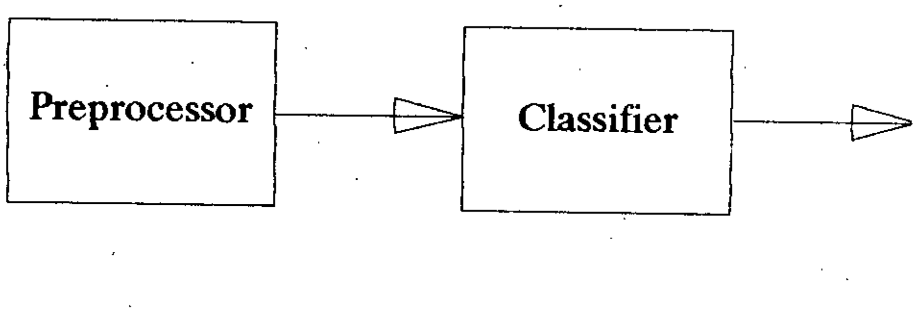


Fig. 3.1 Block diagram of the proposed model.

3.1 The Preprocessor

The purpose of preprocessing is to create an intermediate representation of the input pattern which will later serve as input to the classifier. In order to obtain a high success ratio in classification, the system should provide rotational, scaling and translational invariance before attempting classification. Such invariance can be achieved by a preprocessor with the following properties :

1. The preprocessed output in response to the rotated, scaled and translated version of a pattern should remain unchanged.

2. The computation involved in preprocessing should be reasonably low and easy, i.e., it should not increase the overall complexity of the recognition system.

The preprocessor consists of two blocks as shown in Fig. 3.2. The function of each block is described in the following sub-sections.

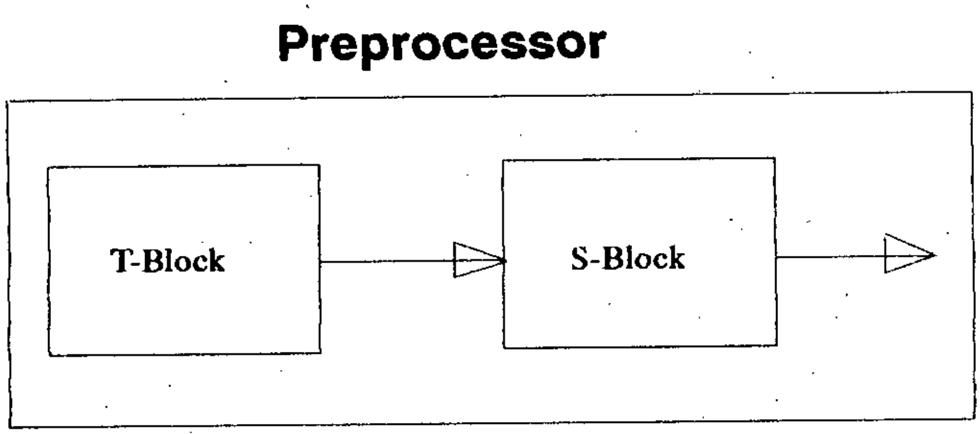


Fig. 3.2 Block diagram of the preprocessor.

3.1.1 T-block

Each pattern is represented by a NxN binary pixel in the pattern grid where '1' represent an on-pixel and '0' off-pixel. The T-block computes the center of gravity of the pattern by averaging the x and y coordinates of the on-pixels. Then it shifts this point to the origin of the grid maintaining invariance. The resulting intermediate pattern is passed to the S-block.

Let the function f(x,y) give the value of the pixel at the coordinate (x, y). For digitized binary-valued 2-D patterns this function will be either '0' or '1'. The formula used in the computation of the center of gravity is:

$$x_{av}=\frac{1}{\sum_{i=1}^{N}\sum_{j=1}^{N} f(x_i,y_j)}\sum_{i=1}^{N}\sum_{j=1}^{N} f(x_i,y_j)\cdot x_i \tag{3.1}$$

$$y_{av}=\frac{1}{\sum_{i=1}^{N}\sum_{j=1}^{N} f(x_i,y_j)}\sum_{i=1}^{N}\sum_{j=1}^{N} f(x_i,y_j)\cdot y_j \tag{3.2}$$

Where x_{av} and y_{av} are the co-ordinates of the center of gravity of the pattern.

The next step is to calculate the radial distance of each pixel with respect to the calculated center of gravity (x_{av}, y_{av}) of that pattern. This radial distance of each pixel is then normalized by the maximum radial distance.

$$r_{ij}(x_i,y_j)=\sqrt{(x_i-x_{av})^2+(y_j-y_{av})^2} \tag{3.3}$$

$$\theta_{ij} = \cos^{-1} \frac{(x_i - x_{av})}{r_{ij}(x_i, y_j)} \tag{3.4}$$

$$r'_{ij}(x_i, y_j) = \frac{r_{ij}(x_i, y_j)}{r^{max}(x_i, y_j)} \tag{3.5}$$

Where $r_{ij}(x_i, y_j)$ is the radial distance of the pixel (x_i, y_j) from (x_{av}, y_{av}) and $r'_{ij}(x_i, y_j)$ is the normalized value of $r_{ij}(x_i, y_j)$ by the maximum radial distance $r^{max}(x_i, y_j)$. θ_{ij} is the angle the pixel (x_i, y_j) makes with the x-axis if the Cartesian coordinate is imagined with (x_{av}, y_{av}) as center.

The T-block basically computes the polar co-ordinate (r_{ij}, θ_{ij}) of each 'ON' pixel with respect to the center of gravity of the pattern. If a pattern is translated within the image grid, each 'ON' pixel and center of gravity will be shifted by the same amount keeping their relative position same. This will keep the output of the T-block exactly the same even if the pattern is translated. Similarly normalizing the radial distance of each pixel by the maximum value will keep the output of the T block unchanged if the pattern is scaled by a certain factor. This is maintained in an ideal case, i.e., it is assumed that scaling does not introduce any error in the scaled version of the pattern. Following the same argument, if a pattern is rotated by an angle δ then, in an ideal case, each (r_{ij}, θ_{ij}) of the original pattern will be shifted by δ, i.e.,

$$r^r_{ij}(x_i, y_j) = r_{ij}(x_i, y_j) \tag{3.6}$$

$$\theta^r_{ij} = \theta_{ij} + \delta \tag{3.7}$$

Where $(r^r_{ij}, \theta^r_{ij})$ is the polar co-ordinate of the rotated pattern. Thus the output of the

T- block will be cyclically shifted by the angle of rotation of the pattern. Therefore, to make the preprocessed output invariant, the output of the T block is needed to be passed through a cyclic shift invariant transformation. This transformation realized by the S block is described in the following subsection.

3.1.2. S- block

The transformation used in this block is called Rapid Transformation (RT) [29] and requires that the number of inputs be 2^N. So the output of the T-block is first graphed into 2^N which are used as the inputs of RT transformation. All the r_{ij}'s with in the angle-slot of L degree ($L = 360°/2^N$) are summed up in the following manner

$$X_i' = \sum_{i \times L \geq \theta_{ij} < (i+1)L} r_{ij}'(x_i, y_j) \qquad 0 \leq i < 2^N \tag{3.8}$$

To tackle the scaling problem the summation of r_{ij}'s in a certain angle-slot i.e. X_i' is divided by the number of active bit in that angle-slot to determine the inputs X_i to RT transformation.

$$X_i = \frac{X_i'}{\sum_{i \times L \geq \theta_{ij} < (i+1)L} \text{no of active bit}} \qquad 0 \leq i < 2^N \tag{3.9}$$

The principle of the transform for an one dimensional array of eight input variables, representing the components of an S dimensional pattern vector X is shown in Fig .3. 3. For an array of $N=2^M$ input variables (numbered from 0 to N-1), M transformation steps are required. In the first transformation step the input variables are divided into two groups, numbered from 0 to (N/2-1) and from N/2 to (N-1). The variables of the first transform column (1) are then calculated by

$$X_i^1 = X_i + X_{i+N/2} \qquad \Big|_{i=0}^{N/2-1}$$

$$X_{i+N/2}^1 = |X_i - X_{i+N/2}| \qquad \Big|_{i=0}^{N/2-1} \tag{3.10}$$

In the second transformation step the two groups of the variables in column (1) are again divided into two subgroups each, giving the variables of column (2) by

$$X_i^2 = X_i^1 + X_{i+N/4}^1 \qquad \Big|_{i=0}^{N/4-1}$$

$$X_{i+N/4}^2 = |X_i^1 - X_{i+N/4}^1| \qquad \Big|_{i=0}^{N/4-1}$$

$$X_{i+N/2}^2 = X_{i+N/2}^1 + X_{i+3N/4}^1 \qquad \Big|_{i=0}^{N/4-1} \tag{3.11}$$

$$X_{i+3N/4}^2 = |X_{i+N/2}^1 - X_{i+3N/4}^1| \qquad \Big|_{i=0}^{N/4-1}$$

This procedure is repeated $M(=\log_2 N)$ times.

In general, the variables in any column (R) are calculated from the variables in the preceding column (R-1) by

$$X_{m+2ns}^R = |X_{m+2ns}^{R-1} + X_{m+(2n+1)s}^{R-1}| \qquad \Big|_{m=0}^{s-1} \Big|_{n=0}^{t-1} \tag{3.12}$$

$$X_{m+(2n+1)s}^R = |X_{m+2ns}^{R-1} - X_{m+(2n+1)s}^{R-1}| \qquad \Big|_{m=0}^{s-1} \Big|_{n=0}^{t-1} \tag{3.13}$$

With $s=2^{M-R}$ and $t=2^{R-1}$.

These operations are illustrated in Fig. 3.3 for N=8. The dotted arrows indicate subtraction, while the solid arrows indicate addition. The absolute value is then taken at each cell.

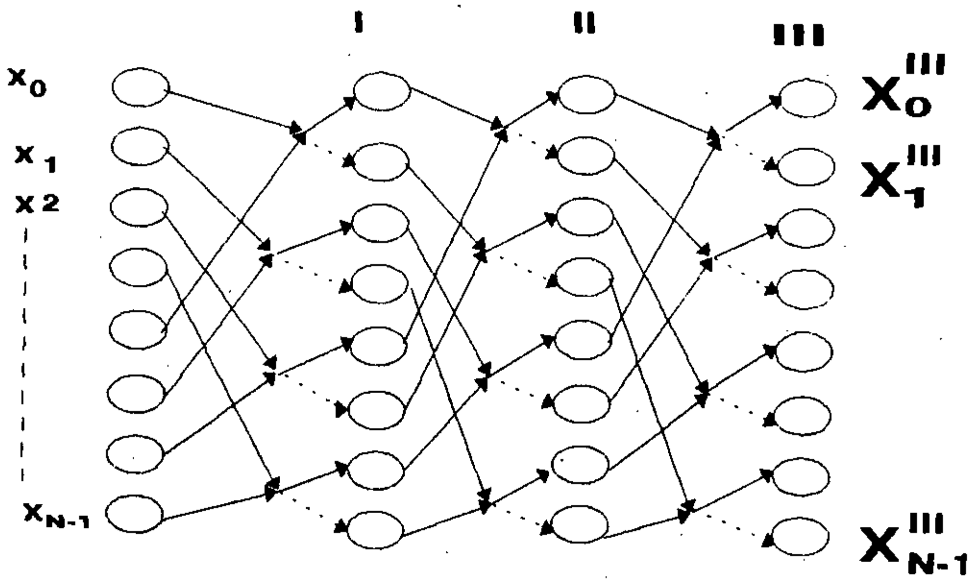


Fig.3.3 Tree graph of the R-transform, for 8 input variables.

3.2 The Classifier

As stated in the previous section, the preprocessed output will be invariant to translation, scaling and rotation in an ideal case, i.e., on the assumption that scaling and rotation do not introduce any noise in the pattern. However, in practical cases, some amount of noise is always introduced in case of scaling and rotation. The preprocessed output will therefore be some what changed when a pattern is either rotated or scaled. The final classification stage of the proposed model must be robust enough to deal with the noise present in the input pattern in order to yield a good overall performance of the recognition system. The classifier in the proposed model is an artificial neural net classifier which has the desired capability of classifying patterns even with some amount of noise present. The outputs of the preprocessor in response to an exemplar pattern are fed to the network as inputs and a locally represented binary vector is applied at the output layer as target signal. A detailed description of this classifier is presented in the next chapter.

CHAPTER-FOUR

THE ARITIFITIAL NEURAL NETWORK CLASSIFIER

- **INTRODUCTION**
- **BIOLOGICAL NEURONS**
- **ARITIFIAL NEURAL NETWORKS**
- **BACK-PROPAGATION LEARNING ALGORITHM**
- **UPDATE OF OUTPUT-LAYER WEIGHTS**
- **UPDATE OF HIDDEN-LAYER WEIGHTS**

4.0 Introduction

Use of Artificial Neural Networks (ANN) for pattern classification purposes has become very popular in recent years. Especially, feed-forward type networks are frequently applied to many recognition problems. Such networks learn the desired input-output relationship using certain training procedure. The network has input and output layers as well as one or multiple intermediate layers called hidden layers. Given the input and output layer size, one should decide on the number of hidden layers and also the size of each hidden layer. In fact the decision is critical since it determines the performance of the classifier. The decision depends on the type of the problem to be dealt with, as well as on the size and variety of input-outputs. The size of the hidden layer must be large enough so that the network can distinguish the pattern belonging to different classes, while as compact as possible to avoid memorization of the exemplars.

The classifier used in the proposed pattern recognition system is a multilayer feed-forward network. The training algorithm is the widely used back-propagation algorithm. The number of nodes in the input layer is fixed and equal to the number of outputs of the preprocessor. Output layer is configured so that each class is represented by an output node.

In the following sections, the basic functioning of biological nervous system is succinctly described and back propagation learning algorithm to train artificial neural network classifier is represented in details.

4.1 Biological Neurons

In quest to build intelligent machines in the hope of achieving human-like performance in the field of speech and image recognition, natural language processing etc., we have but one naturally occurring model: the human brain, a highly powerful computing device. It follows that one natural idea is to simulate the functioning of brain directly on a computer. The general conjecture is that thinking about computation in terms of brain metaphor rather than digital computer will lead to insights into the nature of intelligent behavior. This conjecture is strongly supported by the following facts.

Present day digital computers are capable of storing vast quantities of information. Their circuits operate in nanoseconds and can perform extensive arithmetic calculations without error. Human being cannot approach these capabilities. On the other hand, human being routinely performs simple tasks such as walking, talking, and common sense reasoning. Biological neurons which operate in millisecond range are extremely slow devices when compared to their counterpart in digital computers. Yet human can perform extremely complex tasks, like interpreting visual scene or understanding a sentence, in just one tenth of a second. In other words, we do in about a hundred steps what current computers cannot do in ten million steps [35]. The underlying reason is that, unlike conventional computers, the brain contains a huge numbers of neurons, information processing elements of biological nervous system, acting in parallel [35]. This suggests that in our search for solutions to intelligent problems, we need machines built on large number of processing elements computing in parallel.

The massive parallelism in human brain serves a strong motivation for the idea of building an intelligent machine modelled after biological neurons, now known as artificial neural networks. Artificial neural networks are based on our present understanding of biological nervous system. Though uncertainty concerns the details of biological mechanisms, the organization and information-processing characteristics of several nerve cells are fairly well understood.

Fig. 4.1 shows a systematic diagram of two biological neurons in synaptic contact. Neurons are electrically-active cells, they interact with one another through the flow of local electrical (ionic) currents between them. These ionic currents are driven by a voltage difference across the neuron's cell membrane. This voltage is normally maintained at a level called the membrane resting potential which is about 60 millivolts. A neuron impulse consists of a rapid voltage change which occurs in a localized sections of neuron's membrane. This voltage change is produced by a flow of local ionic currents through the membrane. Once initiated, these flows propagate to the adjacent neurons along the length of nerve fibre. An impulse lasts a few milliseconds.

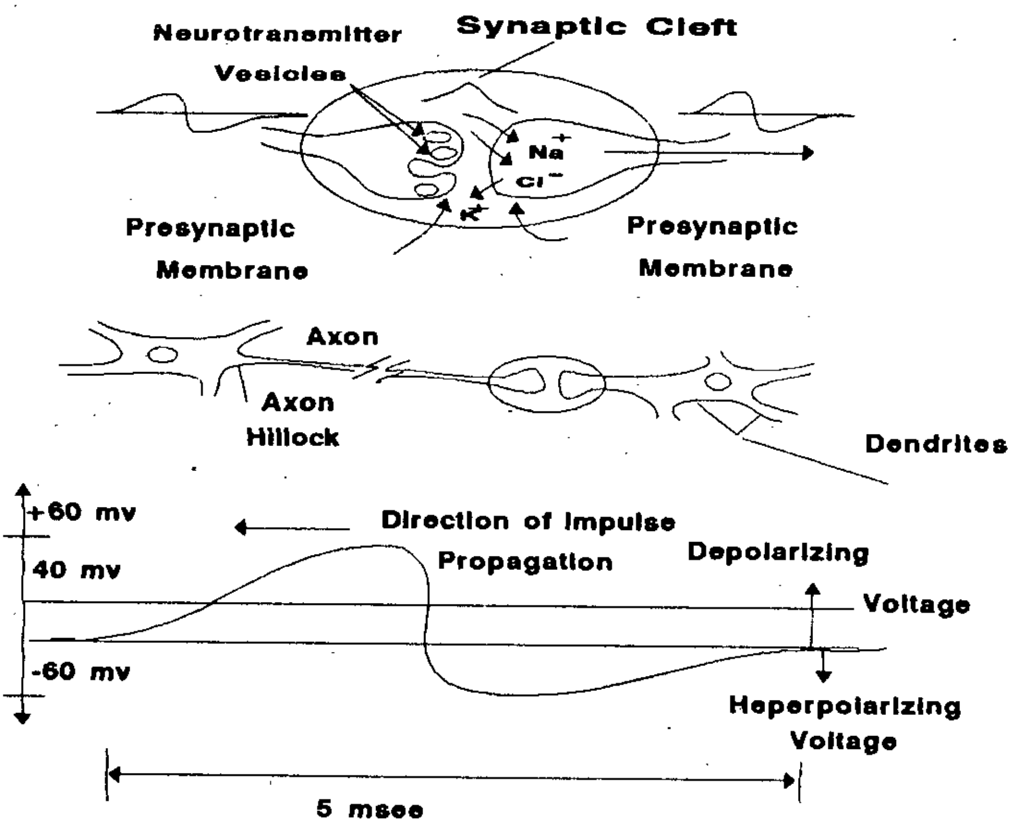


Fig. 4.1. Biological neurons in synaptic contact.

The main elements of the biological neural systems are:

The soma or nerve cell, the large round central body of the neuron from 5 to 1000 microns in diameter.

- The output fibre called an axon attached to the soma electrically active producing the pulse which is emitted by the neuron.

- The output fibre leading to the soma called dendrite, receives inputs from the other neurons by means of specialized contact.

Each such fibre transmits pulses of current (nerve impulses) in one direction, so that neurons fed currents to one another in definite directions. The axon of each cell feeds the dendrites of its sink-cells through synapses. Each such synapses consists of a pre-synaptic axon-end butted against a post synaptic dendrite-end, usually with a small intervening gap called synaptic cleft. As a nerve impulse reaches a axon end it causes release of a chemical called neurotransmitter which diffuses across the fluid filled synaptic gap. When this chemical reaches the dendrite end in a few msec, it induces a voltage change across the post-synaptic membrane of the dendrite. It can cause the membrane potential to become more positive (depolarizing the membrane) or it can cause the membrane potential to become more negative (hypolarizing the membrane).

If the membrane of the receipt cell's axon hillock is sufficiently depolarized by the arriving impulses, the receipt cell may fire an output impulse down its axon. The soma combines the currents that reach it from the cells and if the combined current exceeds the cell firing threshold it may produce one or more output pulses.

These simple mechanisms are surprisingly powerful. First, each neuron is basically an independent processor. Each neuron receives, combines and produces impulses as an autonomous information processing unit, operating in parallel with other neurons. The processing which occurs at each neural element is comparatively simple, but at any given instant, a tremendous number of such operations are in progress.

4.2 Artificial Neural Networks

Artificial Neural Network is defined as massively parallel interconnected networks of simple (usually adaptive) elements and their hierarchical organizations which are intended to interact with the objects of the real world in the same way as biological nervous systems do [10]. They constitute an alternative knowledge representation paradigm for Artificial intelligence and cognitive science. Artificial Neural Networks are made up of processing units called neurons. The neuron structure given in Fig. 4.2 is the basic building block of most networks.

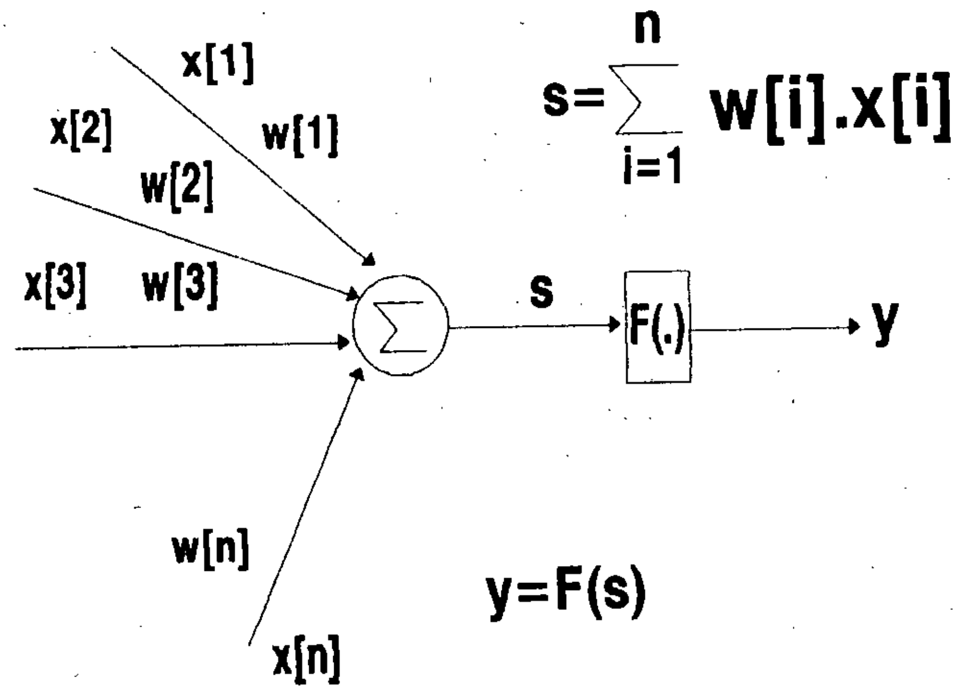


Fig. 4.2 Structure and function of a neuron.

This neuron performs a weighted summation over the outputs of the neurons that are connected to its inputs. Then an activation function also called as the limiting or squashing function is applied on the resulting sum to determine the neuron's output value. Widely used activation function types are various sigmoids, and hard-limiting (thresholding) functions, some of which are shown in Fig. 4.3. The weight associated with the internode connections represent the influence of a neuron's output over activation of the subsequent neurons. Depending on the sign of the weight, a neuron may excite or inhibit other neurons. There is a variety of interconnection topologies between the neurons. The most common one is the class of multilayer feed-forward networks - a subclass of multilayer perceptrons.

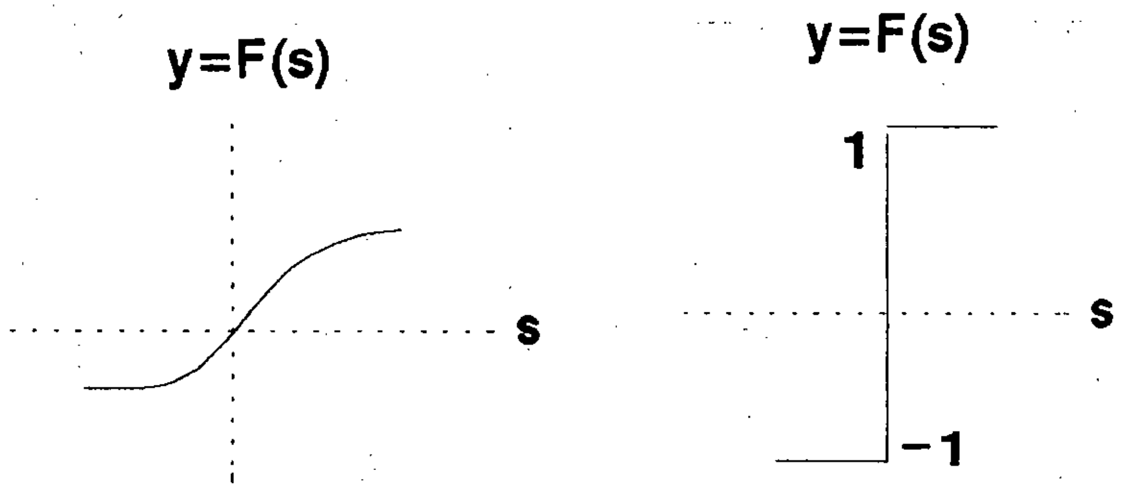


Fig. 4.3 Some activation functions.

4.3 Back-Propagation Learning Algorithm

Back- propagation learning algorithm is intuitively appealing because it is based on a relatively simple concept : if the network gives the wrong answer, then the weight are corrected so that the error is lessened and as a result future result of the network are more likely to be correct.

The back-propagation learning algorithm involves two phases: During the first phase the input is presented and propagated forward through the network to compute the output value o_{pk} for each unit. This output is then compared with the targets, resulting in an error signal δ_{pk} for each output unit. The second phase involves a backward pass through the network (analogous to the initial forward pass) during which the error signal is passed to each unit in the network and the appropriate weight changes are made. This second backward pass allows the recursive computation of δ as indicated above. The first step is to compute δ for each of the output units. This is simply the difference between the actual and desired output values times the derivative of the squashing function. Then the weight changes for all connections that feed into the final layer can be computed. After this is done, then compute δ's for all units in the penultimate layer. This propagates errors back one layer, and the same process can be repeated for every layer.

The significance of the process is that, as the network trains, the nodes in the intermediate layers organize themselves such that different nodes learn to recognize different features of the total input space. After training, when presented with an arbitrary input pattern that is noisy or incomplete, the units in the hidden layers of the network will respond with an active output if the new input contains a pattern that resembles the feature the individual unit learn to recognize during training. Conversely hidden-layer units have a tendency to inhabit their outputs if the input pattern does not contain the feature that they were trained to recognize.

The backpropagation network as shown in Fig. 4.4 is a layered feed forward network that is fully interconnected by layers. Thus, there are no feedback connections and no connections that bypass one layer to go directly to a later layer. Although only three layers are used in discussion, more than one hidden layer is permissible.

Suppose a set of P vector-pairs, $(\mathbf{X}_1, \mathbf{Y}_1)$, $(\mathbf{X}_2, \mathbf{Y}_2)$, ..., $(\mathbf{X}_p, \mathbf{Y}_p)$ which are examples of functional mapping $\mathbf{Y}=\phi(\mathbf{X})$: $\mathbf{X}\epsilon\mathbf{R}^N$, $\mathbf{Y}\epsilon\mathbf{R}^M$ [35]. The network will learn an approximation $O=Y'=\phi'(X)$ for its training. To derive a method of doing this training that usually works, provides the training-vector pairs have been chosen properly and there is sufficient number of them. Learning of a neural network means finding an appropriate set of weights. The learning technique described here resembles the problem of finding the equation of a line that best fits a number of known points.

Let us consider an input vector, $X_p=(x_1, x_2, \ldots, x_{pN})^t$, is applied to the input layer of the network. The "p" subscript refers to the p training vector.The input units distribute the values to the hidden-layer units. The net input to the jth hidden unit is

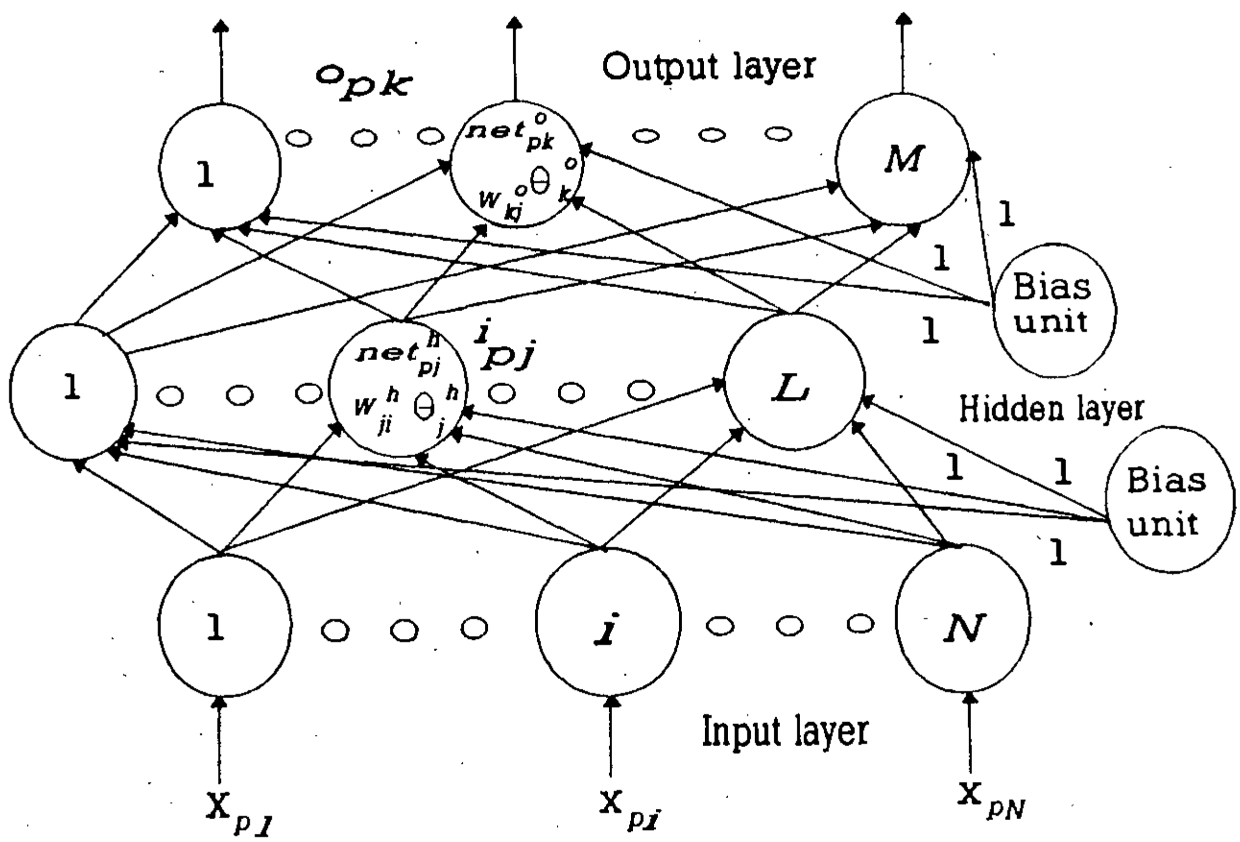


Fig.4.4 Three-layer backpropagation network.

$$net^h_{pj} = \sum_{i=1}^{N} w^h_{ji} x_{pi} + \theta^h_j \tag{4.1}$$

Where w^h_{ji} is the weight on the connection from the ith input unit to jth hidden unit, and θ^h_j is the bias term. The "h" superscript refers to quantities on the hidden layer. Assuming that the activation of this node is equal to the net input; then, the output of this node is

$$i_{pj} = f^h_j(net^h_{pj}) \tag{4.2}$$

Where the function $f^h_j(net^h_{pj})$ is referred to as an activation function. Its domain is the set of activation values, net, of the neuron model.

The equations for the output nodes are

$$net^o_{pk} = \sum_{j=1}^{L} w^o_{kj} i_{pj} + \theta^o_k \tag{4.3}$$

$$o_{pk} = f^o_k(net^o_{pk}) \tag{4.4}$$

Where the "o" superscript refers to quantities on the output layer.

4.3.1 Update of Output-Layer Weights

The error at a single output unit is defined $\delta_{pk} = (y_{pk} - o_{pk})$, where the subscript "p" refers to the p training vector, and "k" refers to the kth output unit. In this case y_{pk} is the is the desired output and o_{pk} is the actual output from the kth unit. The error to be minimized is the sum of the squares of the errors for all output units:

$$E_p=\frac{1}{2}\sum_{k=1}^{M}\delta_{pk}^{2} \tag{4.5}$$

To determine the direction in which to change the weights, the negative of the gradient of E_p, ∂E_p , with respect to weights, w_{kj} is calculated. Then the values of the weights can adjust such that the total error is reduced. It is often usual to think of E_p as a surface in a weight space.

From Eq.(4.5) and the definition of δ_{pk}

$$E_p=\frac{1}{2}\sum_{k}(y_{pk}-o_{pk})^2 \tag{4.6}$$

$$\frac{\partial E_p}{\partial w_{kj}^{o}}=-(y_{pk}-o_{pk})\frac{\partial f_k^{o}}{\partial(net_{pk}^{o})}\frac{\partial(net_{pk}^{o})}{\partial w_{kj}^{o}} \tag{4.7}$$

Where Eq. (4.4) is used for the output value, o_{pk} and the chain rule for partial derivatives. The last factor of Eq.(4.7) is

$$\frac{\partial(net_{pk}^{o})}{\partial w_{kj}^{o}}=\left(\frac{\partial}{\partial w_{kj}^{o}}\sum_{j=1}^{L}w_{kj}^{o}i_{pj}+\theta_k^{o}\right)=i_{pj} \tag{4.8}$$

Combining Eqs (4.7) and (4.8), the negative gradient

$$-\frac{\partial E_p}{\partial w_{kj}^{o}}=(y_{pk}-o_{pk})f_k^{o\prime}(net_{pk}^{o})i_{pj} \tag{4.9}$$

As far as the magnitude of the weight change is concerned, it has been taken to be proportional to the negative gradient. Thus the weights on the output layer are updated according to

$$w_{kj}^{o}(t+1)=w_{kj}^{o}(t)+\Delta_p w_{kj}^{o}(t) \tag{4.10}$$

Where

$$\Delta_p w_{kj}^{o}=\eta(y_{pk}-o_{pk})f_k^{o'}(net_{pk}^{o})i_{pj} \tag{4.11}$$

The factor η is called the learning-rate parameter. If sigmoid function is used then weight update equation for output unit is

$$w_{kj}^{o}(t+1)=w_{kj}^{o}(t)+\eta(y_{pk}-o_{pk})o_{pk}(1-o_{pk})i_{pj} \tag{4.12}$$

By defining output layer error term

$$\begin{aligned}\delta_{pk}^{o}&=(y_{pk}-o_{pk})f_k^{o'}(net_{pk}^{o})\\&=\delta_{pk}f_k^{o'}(net_{pk}^{o})\end{aligned} \tag{4.13}$$

By combining the equation (4.12) and (4.13) the weight update equation becomes

$$w_{kj}^{o}(t+1)=w_{kj}^{o}(t)+\eta\delta_{pk}^{o}i_{pj} \tag{4.14}$$

4.3.2 Update of Hidden-Layer Weights

The error of the hidden layer is given by

$$E_p=\frac{1}{2}\sum_k (y_{pk}-o_{pk})^2$$
$$=\frac{1}{2}\sum_k (y_{pk}-f_k^o(net_{pk}^o))^2$$
$$=\frac{1}{2}\sum_k (y_{pk}-f_k^o(\sum_j w_{kj}^o i_{pj}+\theta_k^o))^2 \tag{4.15}$$

The gradient of E_p with respect to the hidden-layer weights

$$\frac{\partial E_p}{\partial w_{ji}^h}=\frac{1}{2}\sum_k \frac{\partial}{\partial w_{ji}^h}(y_{pk}-o_{pk})^2$$

$$=-\sum_k (y_{pk}-o_{pk})\frac{\partial o_{pk}}{\partial(net_{pk}^o)}\frac{\partial(net_{pk}^o)}{\partial i_{pj}}\frac{\partial i_{pj}}{\partial(net_{pj}^h)}\frac{\partial(net_{pj}^h)}{\partial w_{ji}^h} \tag{4.16}$$

Each of the factors in **Eq.** (4.16) can be calculated explicitly from previous equations. The result is

$$\frac{\partial E_p}{\partial w_{ji}^h}=-\sum_k (y_{pk}-o_{pk})f_k^{o'}(net_{pk}^o)w_{kj}^o f_j^{h'}(net_{pj}^h)x_{pi} \tag{4.17}$$

The hidden-layer weights update in proportion to negative of the **Eq.** (4.17) :

$$\Delta_p w_{ji}^h=\eta f_j^{h'}(net_{pj}^h)x_{pi}\sum_k (y_{pk}-o_{pk})f_k^{o'}(net_{pk}^o)w_{kj}^o$$

By using **Eq.**(4.13)

$$\Delta_p w_{ji}^h=\eta f_j^{h'}(net_{pj}^h)x_{pi}\sum_k \delta_{pk}^o w_{kj}^o \tag{4.18}$$

Every weight update on the hidden layer depends on all the error terms, δ^{o}_{pk}, on the output layer. The known errors on the output layer are propagated back to hidden layer to determine the appropriate weight changes on that layer. By defining hidden layer error term

$$\delta^{h}_{pj}=f^{h'}_{j}(net^{h}_{pj})\sum_{k}\delta^{o}_{pk}w^{o}_{kj} \tag{4.19}$$

So the weight update equation becomes analogous to those for the output layer:

$$w^{h}_{ji}(t+1)=w^{h}_{ji}(t)+\eta\delta^{h}_{pj}x_{pi} \tag{4.20}$$

The amount of weight adjustment depends on three factors : δ, η, x. The size of the weight adjustment is proportional to δ the error value of that unit . Thus a larger error value for that unit results in the larger adjustments to its incoming weights.

The weight adjustment is also proportional to x , the output value for that originating unit. If this output value is small , then the weight adjustment is small. If this output value is large, then the weight adjustment is large. Thus a higher activation value for incoming unit results in a larger adjustment to its outgoing weight.

The variable η in the weight adjustment equation is the learning rate. Its value-commonly between 0.25 and 0.75- is chosen by the neural network user, and usually reflects the rate of learning of the network [35].

CHAPTER-FIVE

SIMULATION RESULTS

- INTRODUCTION
- PATTERNS USED IN SIMULATION
- INVARIANCE ACHIEVED BY THE PREPROCESSOR
- CLASSIFICATION PERFORMANCE
- SUMMARY

5.0 Introduction

Based on the pattern recognition model proposed in the previous chapters, an investigation was carried out to classify and recognize the English numeric digits (0 ~ 9). In this chapter, the invariance achieved by the preprocessor and the recognition ability of proposed model as a whole when applied to this particular 10-class problem are reported. As regard to neural net classifier, 9 different network architectures with different number of hidden (5,7,9) and input (16,32,64) units have been studied to investigate the effect of network size on overall recognition performance. The classification performance for these different networks are studied in this chapter.

5.1 Patterns used in simulation

The data base consists of 64x64 binary images of all the 10 English numeric digits (0 to 9) as shown in Fig. 5.1. Each pattern has 4096 pixels (64x64), each pixel takes value '1' when black and '0' when white. These patterns are used as exemplar for the preprocessor. Here one exemplar pattern in each category was considered. The neural network classifier was trained with preprocessed output of these exemplar patterns as inputs and corresponding locally represented binary vectors as outputs.

Forty one different images of each pattern are used in the classification phase. These test patterns per pattern-class consists of 18 rotated images, 15 scaled images, and 8 translated images. The rotated and scaled images are taken from 5° to 90° at 5° interval and scaling factor 0.5 to 2.0 at 0.1 interval respectivcly. As a sample of test patterns, Fig. 5.2 shows the 6 rotated images of exemplar pattern '3', rotation angles are 15°, 30°, 45°, 60°, 75°, and 90°. Three scaled images of exemplar pattern '1' are shown in Fig. 5.3, scaling factors are 0.5, 1.25, and 2.0. Fig. 5.4 shows pattern '7' shifted within the image grid at three different positions.

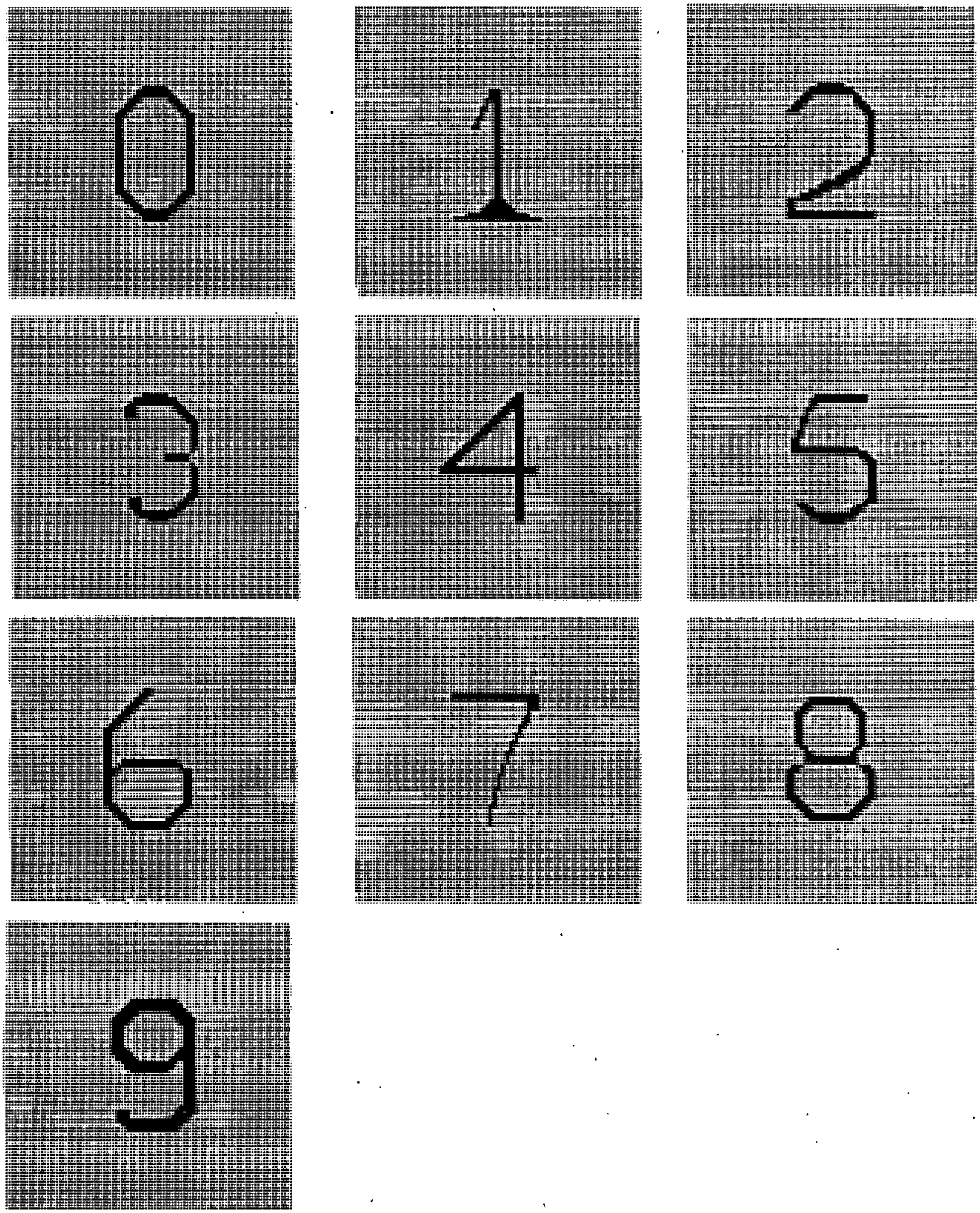


Fig. 5.1 Exemplar patterns used in simulation.

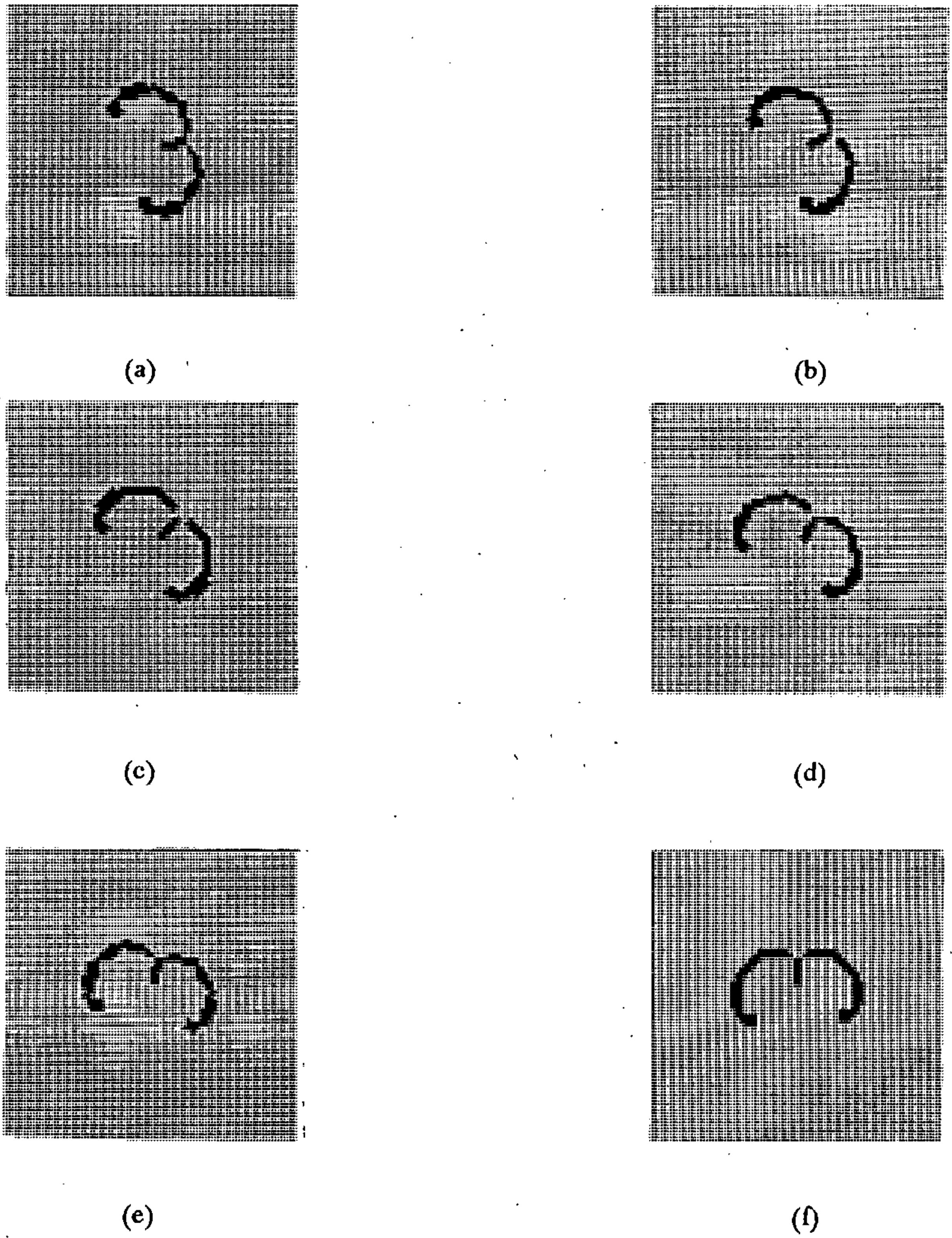
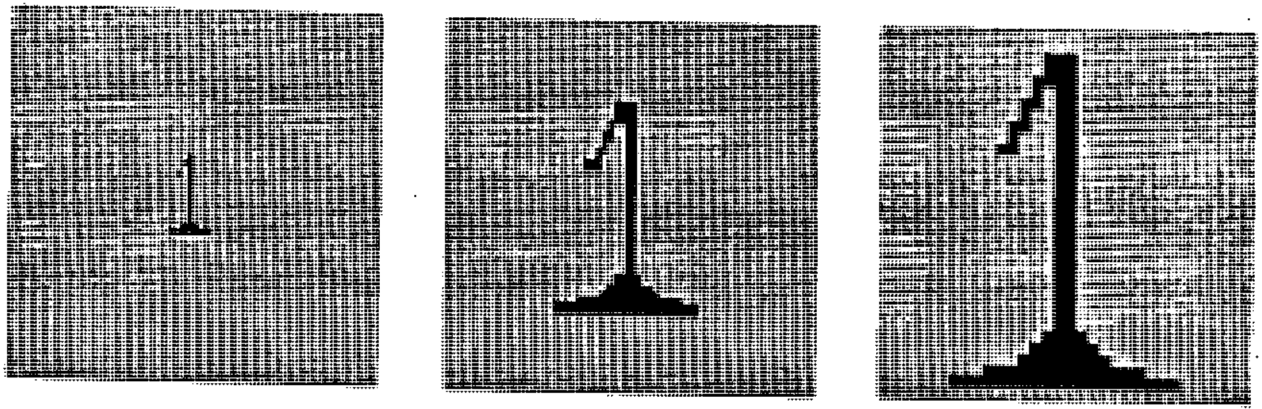
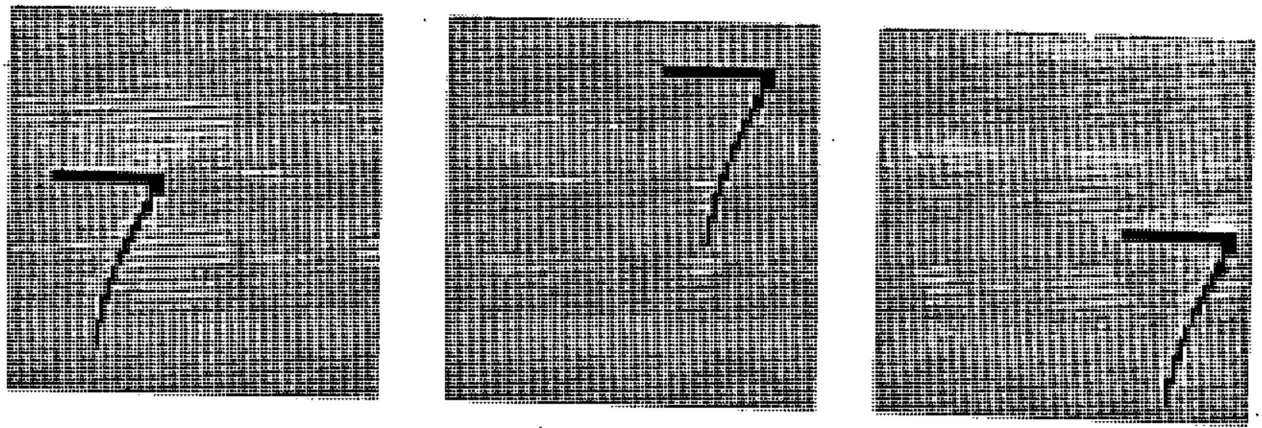


Fig. 5.2 Test patterns of exemplar pattern '3'. (a) 15° , (b) 30° , (c) 45° , (d) 60° , (e) 75° and (f) 90° rotated version of the exemplar pattern '3'.



(a) (b) (c)

Fig. 5.3 Test patterns of exemplar pattern '1'. Scaling factor (a) 0.5, (b) 1.25, and (c) 2.0 of the exemplar.



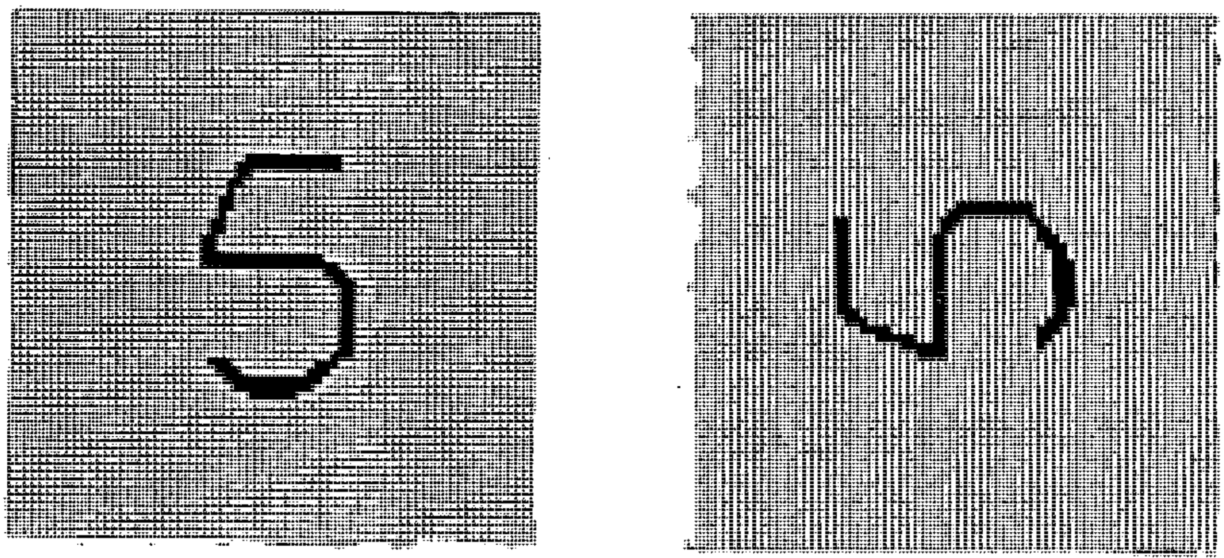
(a) (b) (c)

Fig. 5.4 Test patterns of exemplar pattern '7' shifted within the image grid at three different positions. (a) shifted to left by 15 pixels and downward by 10 pixels with respect to the original position, (b) shifted to right by 15 pixels and upward by 10 pixels, and (c) shifted to right by 20 pixels and downward by 15 pixels.

5.2 Invariance achieved by the preprocessor

The invariance of the preprocessor means the output of the S-block as described in chapter 3 is same for any rotated, scaled, or translated version of exemplar pattern.

The exemplar pattern '5' and its 90° rotated version are shown in Fig. 5.5. The inputs of the S-block for exemplar pattern '5' are shown in Fig. 5.6 and for 90° rotated version in Fig. 5.7. From the observation of Fig. 5.6 and Fig. 5.7 it is clear that the magnitude of the corresponding input number is not same but it is cyclically shifted by an angle of 90° since the test pattern is a 90° -rotated version of the exemplar pattern. But the output of the S-block as shown in Fig. 5.8 is same for these two inputs. The same output generated in these two cases show the invariance achieved by the preprocessor. However, it is worth mentioning that rotation of any exemplar pattern by an angle other than 90° or its integer multiplicative introduces some amount of noise in the pattern. Due to this noise, the outputs of the S-block for any rotated version of exemplar pattern will not exactly be the same as those of the exemplar pattern. Even with the noise present, if the output of the S-block in response to a test pattern is close enough to the corresponding exemplar's outputs, then the neural net classifier is expected to deal with this deviation and classify the test pattern correctly.



(a) (b)

Fig. 5.5 (a) Exemplar pattern '5', (b) its 90° rotated version.

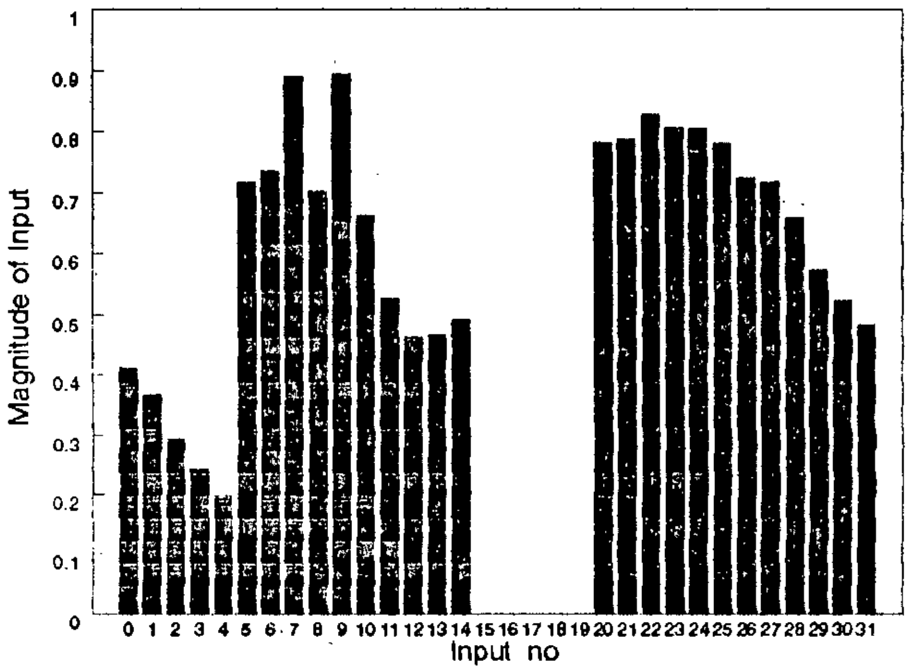


Fig. 5.6. Inputs of S-block for exemplar pattern '5'.

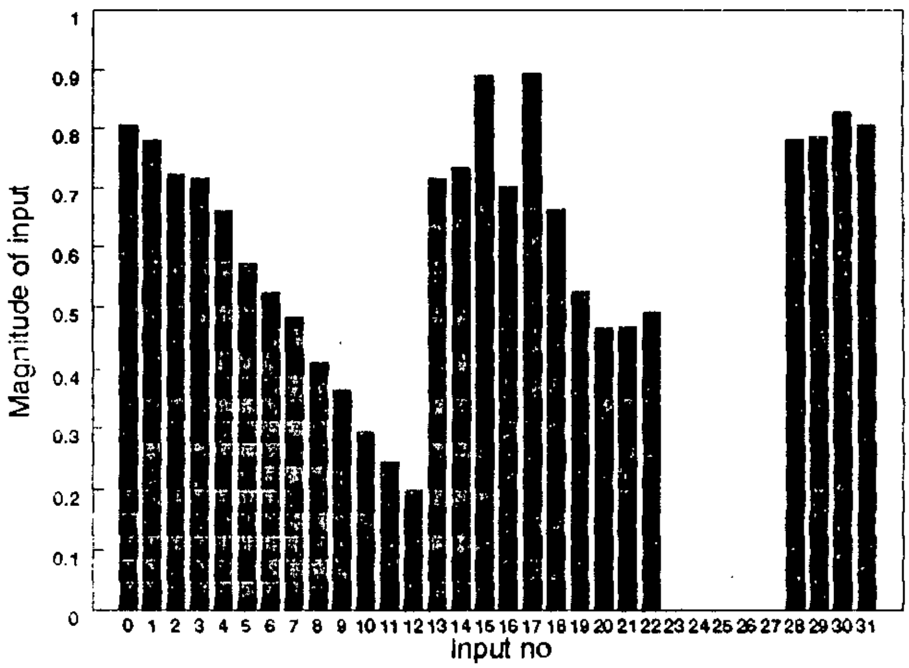


Fig. 5.7. Inputs of S-block for 90° rotated version of exemplar pattern '5'.

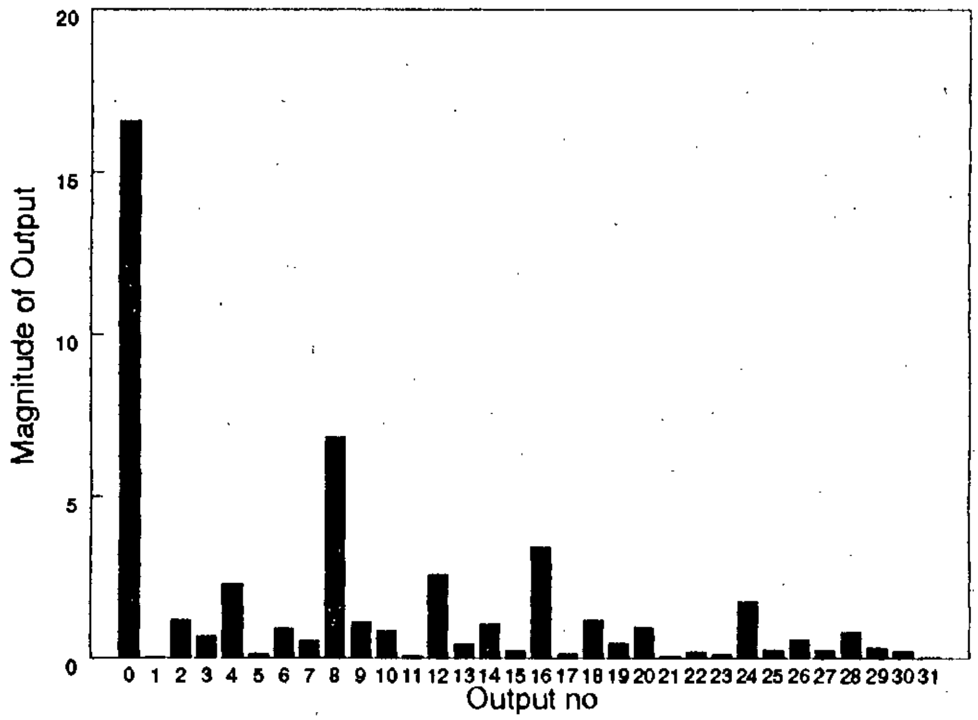


Fig. 5.8 Output of S-block for exemplar pattern '5' and its 90° rotated version. Output of S-block remained same.

5.3 Classification Performance

An important problem in any pattern classification is the estimate of the classification error [18]. The estimation of the error rate is done by finding the ratio of the number of misclassified test samples to the total number of tested samples.

To compute it, the available sample must be divided into two sets: one is the standard pattern and other one is the sample to be recognized. The training set is composed of distortionless pattern as shown in Fig. 5.1. The test set is composed of rotated, scaled, and translated versions of these patterns.

The neural net classifier was presented with the S-block outputs of an exemplar pattern at the input layer and corresponding target outputs at the output layer. Presentation of the exemplar patterns and weight modification, described in chapter 4, were repeated until the sum-squared-error measured reduced to 0.28. The network was then tested on the test set. The performance of the network for 32 input units and 9 hidden units (32-9-10 network) is given in Table 5.1 for shifted pattern, Table 5.2 for rotated pattern, and Table 5.3 for scaled pattern. The other networks performance, (64-5-10, 64-7-10, 64-9-10, 32-5-10, 32-7-10, 16-5-10, 16-7-10, and 16-9-10 network), are given in appendix A.

A 32-9-10 network was trained 10 times using different learning rate parameters and initial weights. Each time performance of the network was investigated with test patterns. Since the performance of the neural network varies with the initial weight setting and learning parameters, the recognition rate for each pattern was calculated by averaging the results along with the standard deviation. This recognition rate and standard deviation gives a better idea about the classifier's performance. The following tables, Table 5.1 for shifted pattern, Table 5.2 for rotated pattern, and Table 5.3 for scaled pattern, give the average recognition rate and standard deviation . The first entry of each box in the table indicates the average recognition rate and the second entry gives the standard deviation. When a pattern is translated there is no distortion in the translated pattern so the overall recognition of the individual pattern is always 100% in the proposed model as it is shown in Table 5.1.

A rotated pattern may contain some amounts of distortion in comparison to that of the exemplar pattern. So, the overall recognition may or may not be 100% which is shown in Table 5.2. From Table 5.2 it is seen that the overall recognition rate of exemplar pattern '2' is poor in comparison to other exemplar patterns in case of rotation. Due to distortion and similarity in construction with other exemplar pattern '3', the output of the preprocessor in case of rotated exemplar pattern '2' is almost the same as the output of the rotated exemplar pattern '3'. This is illustrated in Fig. 5.9.

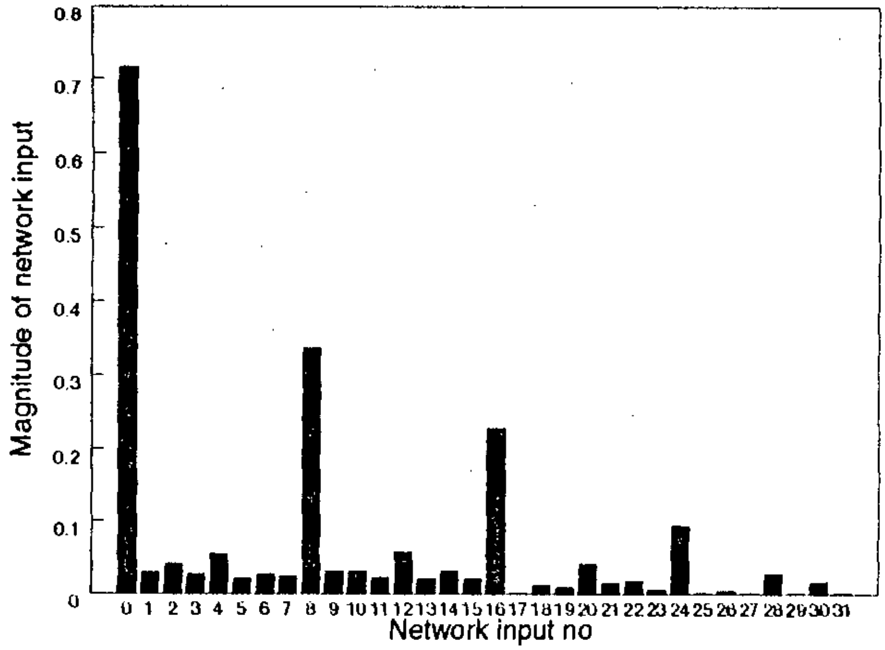
This figure shows that the outputs of the preprocessor in response to 15° rotated version of patterns '2' and '3' are very much similar. This leads to the misclassification of '2' as '3' in most of the misclassified cases. Rotated version of all other exemplar patterns were correctly classified, i.e., 100% recognition rate and the overall recognition rate is 95.17%.

Table 5.1 Average Recognition rate, standard deviation and overall recognition rate for shifted pattern. A total of 80 test patterns, 8 in each class, are used. The results presented in this table are the average on ten trials with ten different 32-9-10 networks. The first entry of each box in the table indicates the average recognition rate, and the second entry gives the standard deviation.

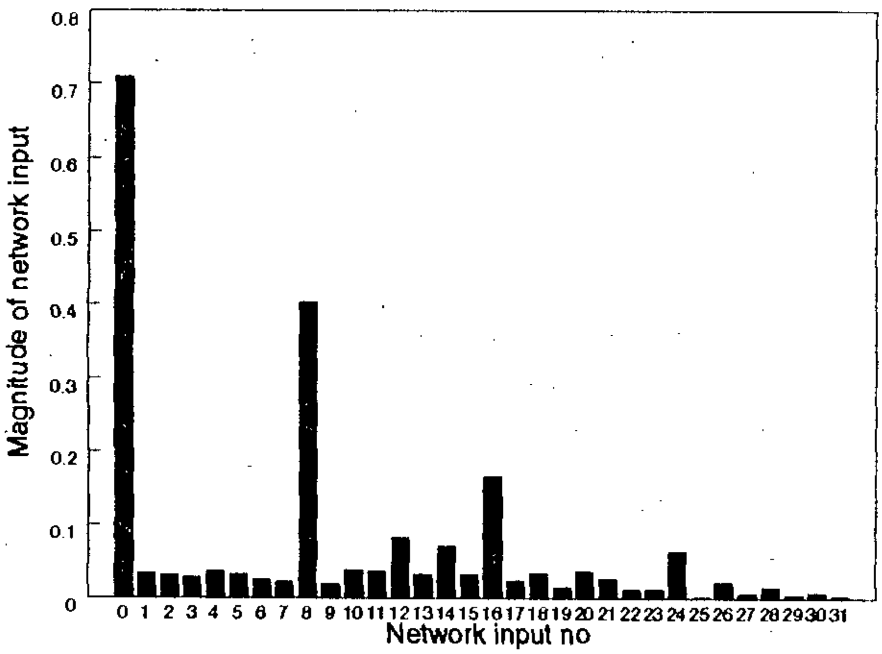
Number of input units=32, Number of hidden units=9										
Shift	Pattern 0	Pattern 1	Pattern 2	Pattern 3	Pattern 4	Pattern 5	Pattern 6	Pattern 7	Pattern 8	Pattern 9
x=5, y=5	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=10, y=10	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=15, y=15	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=20, y=20	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=25, y=25	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=30, y=30	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=40, y=40	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=45, y=45	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%,0	100%, 0	100%, 0	100%, 0	100%, 0
Recognition rate	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%
Overall Recognition = 100%										

Table 5.2 Average Recognition rate, standard deviation and overall recognition rate for rotated pattern. A total of 180 test patterns, 18 in each class, are used. The results presented in this table are the average on ten trials with ten different 32-9-10 networks. The first entry of each box in the table indicates the average recognition rate, and the second entry gives the standard deviation.

Number of input units=32, Number of hidden units=9										
Angle of Rotation	Pattern 0	Pattern 1	Pattern 2	Pattern 3	Pattern 4	Pattern 5	Pattern 6	Pattern 7	Pattern 8	Pattern 9
5°	100%, 0	100%, 0	20%, .16	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
10°	100%, 0	100%, 0	20%, .16	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
15°	100%, 0	100%, 0	20%, .16	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
20°	100%, 0	100%, 0	20%, .16	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
25°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
30°	100%, 0	100%, 0	20%, .16	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
35°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
40°	100%, 0	100%, 0	20%, .16	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
45°	100%, 0	100%, 0	20%, .16	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
50°	100%, 0	100%, 0	20%, .16	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
55°	100%, 0	100%, 0	20%, .16	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
60°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
65°	100%, 0	100%, 0	20%, .16	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
70°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
75°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
80°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
85°	100%, 0	100%, 0	30%, .21	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
90°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
Recognition rate	100%	100%	51.67%	100%	100%	100%	100%	100%	100%	100%
Overall Recognition = 95.17%										



(a)



(b)

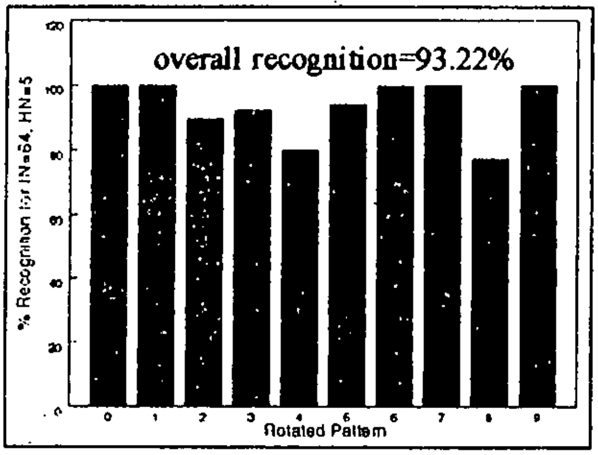
Fig. 5.9 Network input of test pattern. (a) 15° rotated version of exemplar pattern '2'. (b) 15° rotated version of exemplar pattern '3'.

Scaling a pattern by some factor introduces some amount of noise in the pattern and also makes the pattern thick/thin. The amount of noise and the degree of thickness depends on the scaling factor. All these makes recognition of a scaled pattern difficult for the recognizer. Table 5.3 shows the simulation results when the recognition test was done on scaled patterns. The results show that the system is more prone to misclassification when the pattern is scaled down to a very small size, especially, when the scaling factor is 0.5 ~ 0.6. With a scaling factor 0.5, patterns '0', '2' were misclassified in all the ten trials, and pattern '4', '6' were correctly classified only in few trials. This is because, with this scaling factor, patterns becomes extremely thin. Although the problem of thickness has been taken into account in the preprocessing step, the combined effect of noise and thickness can sometimes change the preprocessor's output in such a way that it become closer to another category leading to misclassification. Recognition of patterns scaled by a factor greater than 1.0 is almost 100%. Overall recognition achieved by the system when tested on 150 scaled pattern is 95.6%.

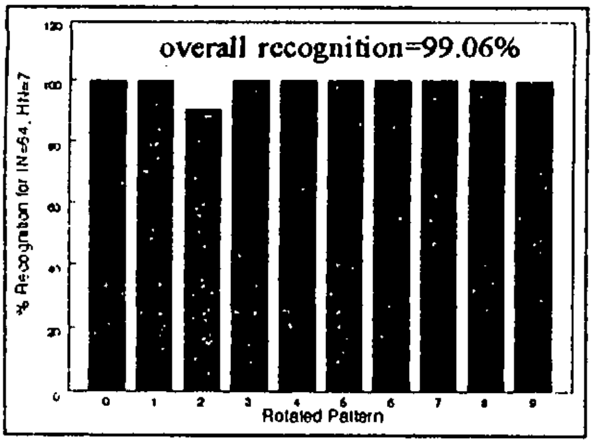
The classification performance of different ANN classifier for rotated version of each exemplar pattern is shown in Fig. 5.10. The number of input units of the classifier actually depends on the number of inputs and outputs of Rapid Transform. For example, when the number of inputs and outputs of Rapid Transform is 64, the input size of the ANN classifier is 64. This can be interpreted as extracting the feature of the input pattern within a window of 5.625°angle (360.0°/64.0=5.625°) looking through the center of gravity of the image. Similarly 11.25° and 22.5° angle for 32 and 16 inputs respectively. Thus, as the Fig. 5.10 (a), (d) and (g) show, varying the number of input units does vary the performance of the recognizer given the number of hidden units kept unchanged. Observation of the results also reveals increasing the hidden units does not always guarantee much improvement. In the case of the networks with 32 and 16 input units, varying the number of hidden units brought little change in overall performance. Especially pattern '2' is poorly classified networks having input units of 32 and 16. Pattern '2' is better classified (recognition rate 89.44% for hidden unit of 5, 90.56% for hidden unit of 7, and 90.56% for hidden unit of 9) when the number of input units of ANN classifier is 64. With these networks overall performance also increases with increasing number of hidden units.

Table 5.3 Average Recognition rate, standard deviation and overall recognition rate for scaled pattern. A total of 150 test patterns, 15 in each class, are used. The results presented in this table are the average on ten trials with ten different 32-9-10 networks. The first entry of each box in the table indicates the average recognition rate, and the second entry gives the standard deviation.

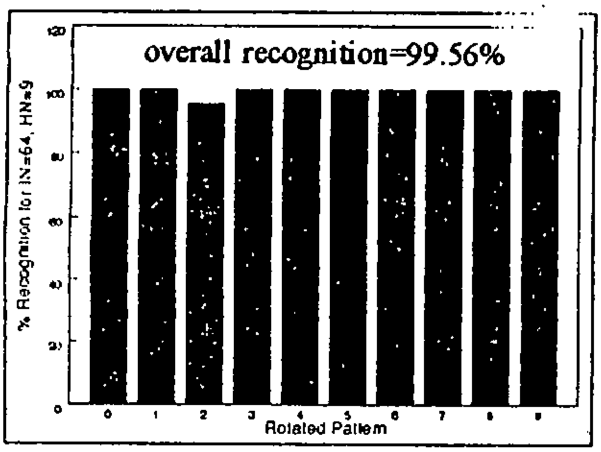
Number of input units=32, Number of hidden units=9										
scale factor	Pattern 0	Pattern 1	Pattern 2	Pattern 3	Pattern 4	Pattern 5	Pattern 6	Pattern 7	Pattern 8	Pattern 9
0.5	0%, 0	100% 0	0%, 0	100%, 0	40%,.24	100%, 0	20%,.16	100%, 0	100%, 0	100%, 0
0.6	100%, 0	100% 0	0%, 0	100%, 0	100%, 0	100%, 0	90%,.09	100%, 0	100%, 0	100%, 0
0.7	100%, 0	100% 0	0%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
0.8	100%, 0	100% 0	0%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
0.9	100%, 0	100%,0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.0	100%, 0	100%,0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.1	100%, 0	100% 0	0%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.2	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%,0	100%, 0	100%, 0	100%, 0	100%, 0
1.3	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.4	100%, 0	100%,0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.5	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.6	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.7	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.8	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.9	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
2.0	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
Recognition rate	93.33%	100%	66.66%	100%	96%	100%	100%	100%	100%	100%
Overall Recognition = 95.6%										



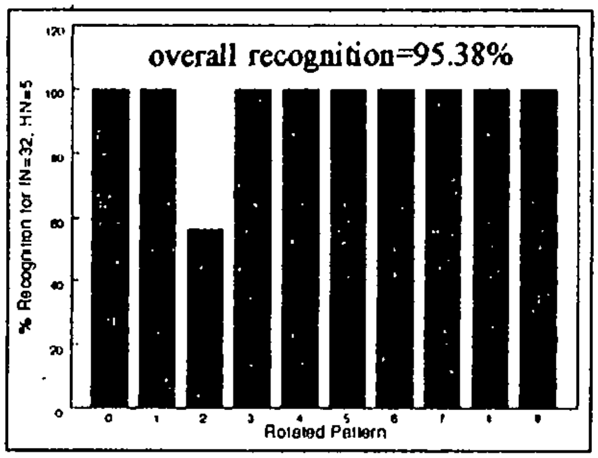
(a) 64-5-10 network



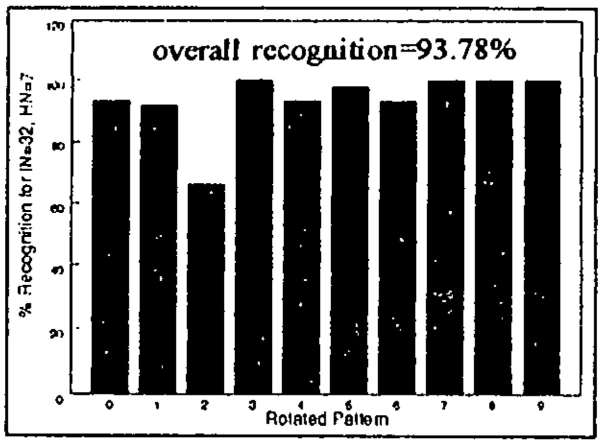
(b) 64-7-10 network



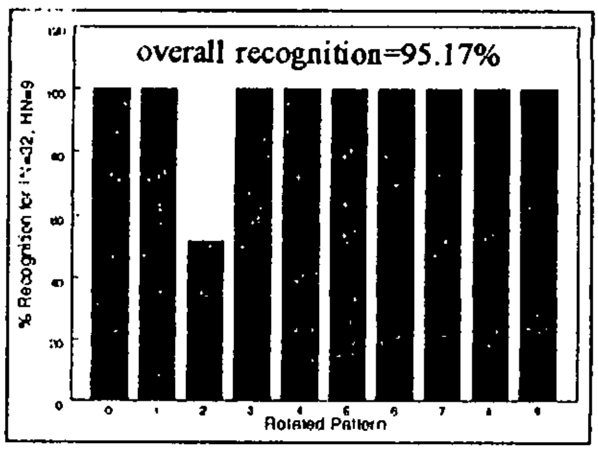
(c) 64-9-10 network



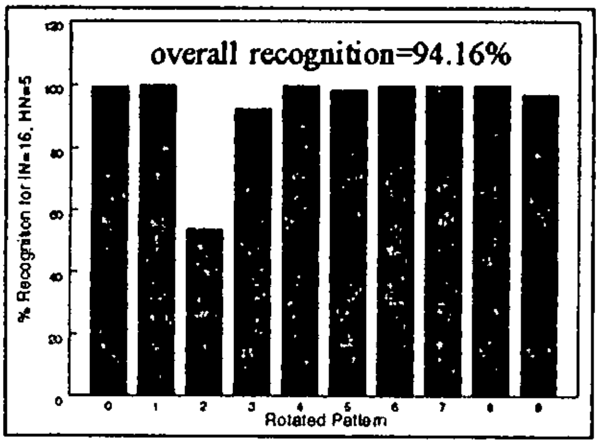
(d) 32-5-10 network



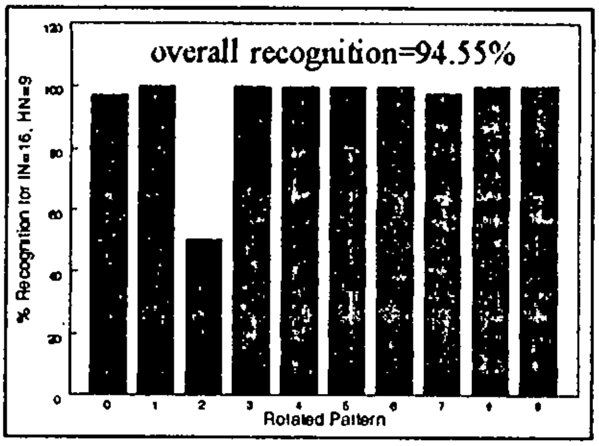
(e) 32-7-10 network



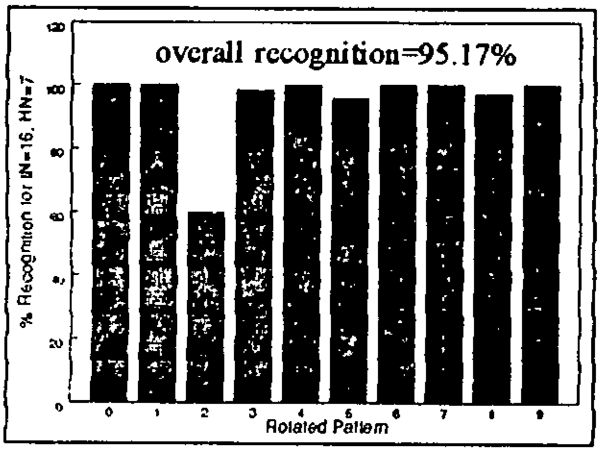
(f) 32-9-10 network



(g) 16-5-10 network

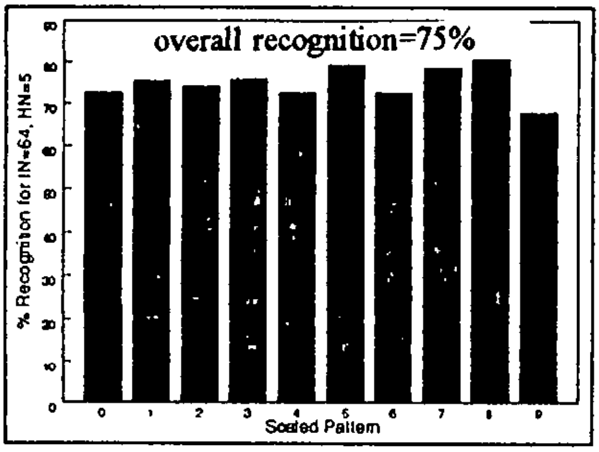


(i) 16-9-10 network

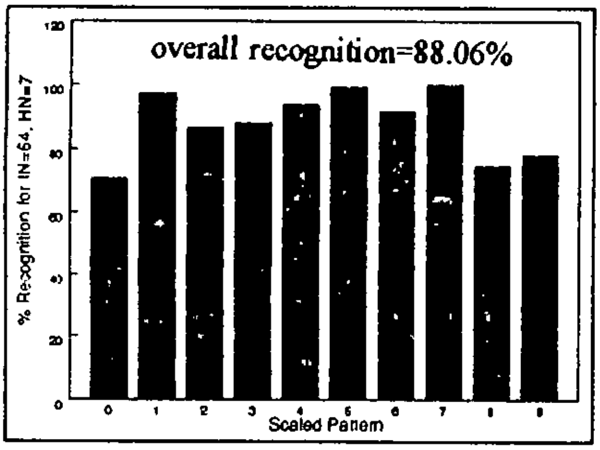


(h) 16-7-10 network

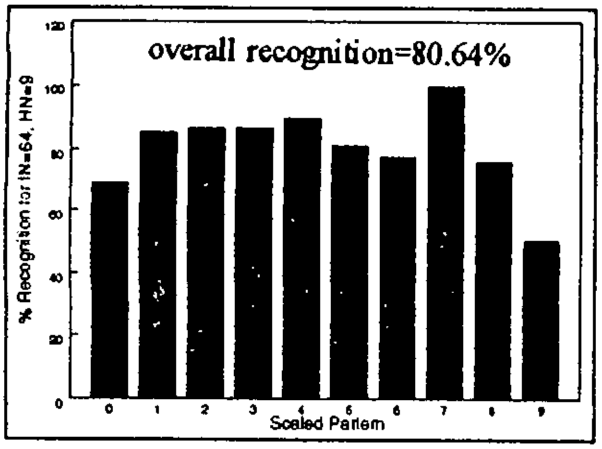
Fig. 5.10 Individual recognition rate of rotated patterns using (a) 64-5-10 network, (b) 64-7-10 network, (c) 64-9-10 network, (d) 32-5-10 network, (e) 32-7-10 network, (f) 32-9-10 network, (g) 16-5-10 network, (h) 16-7-10 network, and (i) 16-9-10 network.



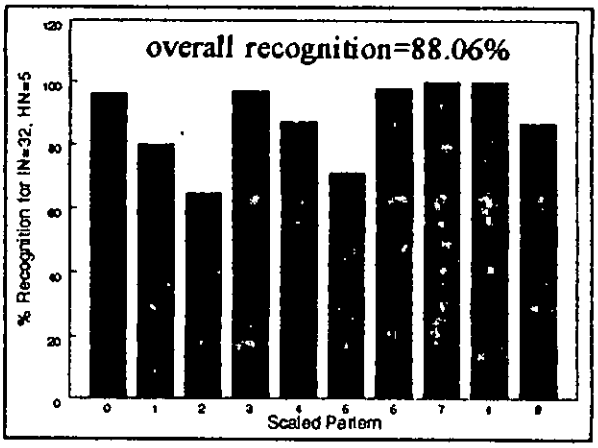
(a) 64-5-10 network



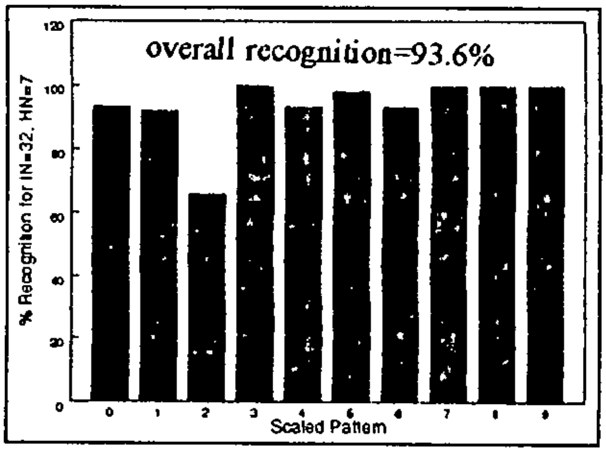
(b) 64-7-10 network



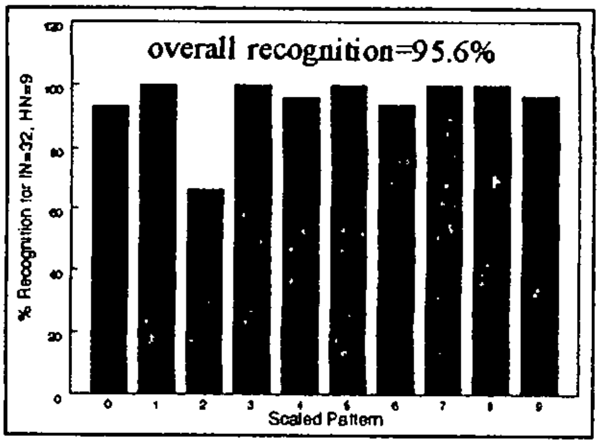
(c) 64-9-10 network



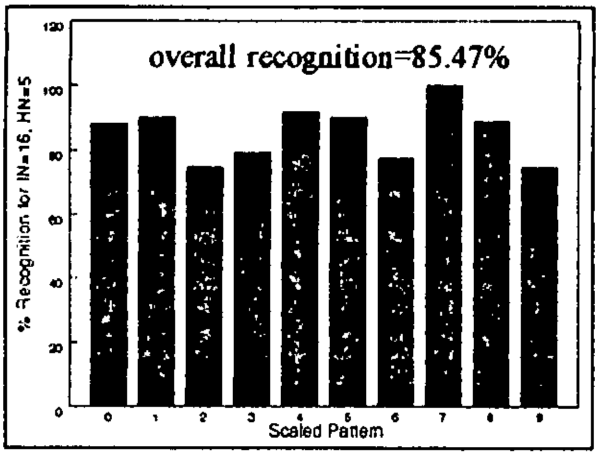
(d) 32-5-10 network



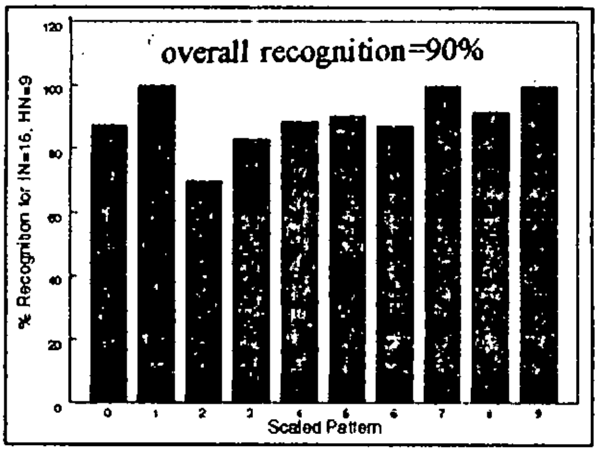
(e) 32-7-10 network



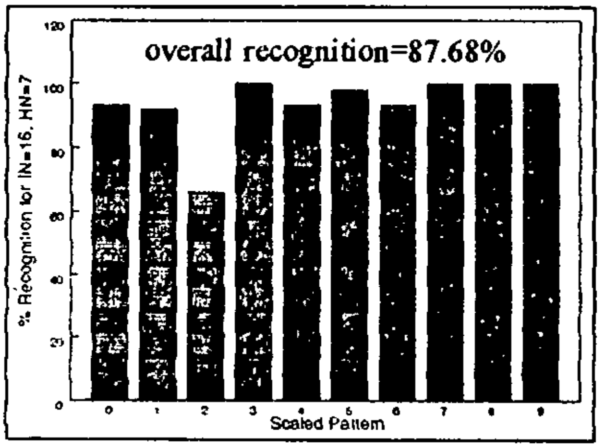
(f) 32-9-10 network



(g) 16-5-10 network



(i) 16-9-10 network



(h) 16-7-10 network

Fig. 5.11 Individual recognition rate of scaled patterns using (a) 64-5-10 network, (b) 64-7-10 network, (c) 64-9-10 network, (d) 32-5-10 network, (e) 32-7-10 network, (f) 32-9-10 network, (g) 16-5-10 network, (h) 16-7-10 network, and (i) 16-9-10 network.

Fig. 5.11 shows the classification performance of different ANN classifier for scaled version of each exemplar pattern. Here also it has been observed that the number of input units affect the recognition performance of the system (refer to Fig. 5.11 (a), (d), (g)). However, the network (64-9-10) which performs best with rotated pattern does poorly with scaled patterns. With 64 input units decreasing the number of hidden units yields better performance. For example, 64-5-10 network performs better than 64-9-10 network. With 32 and 16 input units, increasing the number of hidden units slightly improves the overall performance.

The performance of each network described earlier is summarized in Table 5.4. The table shows that the rotated version and the scaled version of patterns are best recognized by the 64-9-10 network and the 32-9-10 network, respectively. However, in the proposed recognition system, a single classifier is used for classification. Therefore, the

Table 5.4 . Overall Recognition rate of rotated and scaled pattern of different networks.

Network size	Overall recognition rate of rotated pattern	Overall recognition rate of scaled pattern
64-5-10	93.22%	75%
64-7-10	99.06%	88.06%
64-9-10	99.56%	80.64%
32-5-10	95.38%	88.06%
32-7-10	93.78%	93.6%
32-9-10	95.17%	95.6%
16-5-10	94.16%	85.47%
16-7-10	95.17%	87.68%
16-9-10	94.55%	90%

same classifier must be used for classification of both rotated and scaled pattern. Considering both rotated and scaled pattern, a 32-9-10 network yields the best recognition performance.

5.5 Summary

In this chapter, simulation results with ten English numeric digits (0 ~ 9) have been represented. Results demonstrate a good degree of invarience achieved by this simple preprocessor. In case of translated patterns, the preprocessed outputs are the same as those of the exemplar one. In case of rotation and scaling, the preprocessed outputs are somewhat different from those of the exemplar one. This is due to the inclusion of some noise when a pattern is rotated and/or scaled. This deviation is dealt by the ANN classifier. The overall recognition rate of the system is quite satisfactory. The system is capable of recognizing all the translated patterns. In rotation and scaling, the recognition rate is above 95%. However, the recognition rate of the system is highly dependent on the classifier's size. Size of the classifier which yields best recognition performance has to be determined by trial and this size is problem specific. A network of a given size might perform best for rotation and another for scaling. For the particular problem used in simulation a 32-9-10 network size gives the best performance. It may be mentioned here that in this simulation one exemplar per category is used. If more than one exemplar per category were used, a higher recognition rate could be achieved.

The good recognition rate of this system suggests the use of this system in handwritten character recognition. If large database of handwritten character is used for training the ANN along with the preprocessor, the recognition system can be expected to perform reasonably well.

CHAPTER-SIX

CONCLUSIONS

- CONCLUSIONS
- FUTURE WORKS

6.1 Conclusions

In this thesis, a pattern recognition system has been proposed which is simple, computationally inexpensive, and recognizes the unknown pattern in real time. In this system a preprocessor is cascaded with an Artificial Neural Network (ANN) classifier. The preprocessor is design to produce invariant or near invariant outputs even if a pattern is translated, rotated, or scaled. ANN classifier is used because of its growing application in pattern recognition related problems and its better ability to generalize on the patterns that has not been trained by the network. Backpropagation learning algorithm is used to train the classifier. Even though it might take long time to train an ANN classifier, the advantage is that it recognizes an input pattern in real time once it has been trained.

The proposed system was tested with ten numeric digits (0 ~ 9). Simulation results with these patterns show quite encouraging results. For rotated and scaled patterns, the recognition rate is above 95%. The case where a pattern is both rotated and scaled has not been simulated. The number of input/output of Rapid Transform (which is also the number of inputs to ANN classifier) and hidden units of classifier play significant role in determining the recognition capability of the system. A given size of classifier might yield best recognition rate for rotation while another for scaling. There is no analytical method for determining the optimum size of an ANN classifier for a given task. This problem exists with all types of neural networks and the only way to determine the best one is by trials. Thus several classifiers of different sizes have to be trained and the one that gives best recognition performance has to be selected. Use of more than one exemplar per category may enhance the recognition rate.

However, the true ability of proposed system needs to be tested with real-world application, such as, handwritten character recognition which is much more complicated. Handwritten characters may appear separated, e.g., postal zip code or continuous, e.g., handwritten text. If the characters appear continuous, special algorithm must be used before the preprocessor to separate them. In either case, a pattern may appear both rotated and scaled which has not been simulated in the present study. For real world application, a reasonably large database of handwritten character has to be selected, passed through the preprocessor and its outputs has to be trained by the ANN classifier. A complicated architecture for the ANN classifier can be designed rather than the simple three-layer backpropagation network considered in the present work in order to achieve better generalization ability of the classifier and hence increased overall recognition rate.

6.2 Future Works

1) The Rapid Transform (RT) used in the preprocessor appears to be very much sensitive to the noise present in the input pattern. Also, the computation of RT is such that the first output becomes the largest and the most dominating one. Most of the other outputs become very insignificant (refer to Fig. 5.8). To make the preprocessor work better a modified form of RT or any other cyclic shift invariant transform can be used so that most of the outputs becomes significant. Then learning the preprocessed output by ANN classifier will be much more easier.

2) Recognition rate is relatively poor when scaling factor is 0.5 ~ 0.6. Use thicking/thinning algorithm in conjunction with the preprocessor could help increase the recognition performance. The proposed preprocessor can be modified incorporating such as algorithm.

3) The ANN classifier was trained by backpropagation learning algorithm. Recently, some other multilayer feedforward learning algorithm, e.g., double backpropagation [32], OLL (Optimization Layer by Layer) -learning algorithm [33], weight smoothing network [34] have been reported which claimed to possess better generalization ability than backpropagation. It will be worth to investigate with ANN classifier trained with those algorithm and see whether that improves the overall

recognition rate.

4) In the present work only one ANN classifier has been used. This classifier may not perform best for rotation or scaling alone. Instead of using one classifier, multiple classifier can be used. Multiple classifiers can be trained independently, either with different learning algorithm or with the same algorithm but different set of learning parameters. Each classifier will classify the unknown input into certain category and a voting scheme among the classifiers will decide the assigned category. This might improve the overall performance.

5) The proposed recognition system can be used for Bangla handwritten character recognition. Because of the similarity between Bangla characters, similar ones can be grouped. Thus all the characters can be divided into a number of groups. An ANN network will be trained to assign an input pattern to a particular group by giving a single high output at the output layer. Each output of this network will be connected to another subnet at the upper layer. Thus there will be as many subnets as the number of groups. An unknown pattern will first be assigned to a group by the lower layer network. This will in turn activate a subnet which will classify the input to the correct category within the assigned group.

REFERENCES

[1] J. T. Tou and R. C. Gonzalez, "Syntactic Pattern Recognition, A Introduction," 1982, Addision-Wesley Publishing Company, Reading, Massachusetts, U. S. A, pp. 7-21.

[2] R. L. Grimsdale, "A System for the automation recognition of patterns," ***Proc. of the IRE 106***, vol. 7, 1959, pp. 210-221.

[3] L. P. Horwitz and G. L. Shelton, JR, "Pattern Recognition Using Autocorrelation," ***Proc. of the IRE***, vol. 13, January 1961, pp. 175-185 .

[4] S. S. Yau and C. C. Yang, "Pattern Recognition by Using an Associative Memory," ***IEEE Transaction On Electronic Computers***, vol. 15, no.12, Dec. 1966, pp. 944-947.

[5] V. K. Govindan and A. P. Shivaprasad, "Character Recognition-A Review," ***Pattern Recognition***, vol. 23, no.7, 1990, pp. 671-683.

[6] L. N. Kanal, "Patterns in Pattern Recognition," ***IEEE Trans. On Information Theory***, vol. IT-20, Nov. 1974, pp. 697-722.

[7] S. H. Unger, "Pattern Detection and Recognition," ***Proc. of the IRE***, vol-47, Oct. 1959, pp. 1737-1752.

[8] P. G. Perotoo, "A New Method for Automatic Character Recognition," ***IEEE Trans. On Electronic Computers***, Oct. 1963, pp. 521-526.

[9] H. A. Glucksman, "A Propagation Pattern Classifier," ***IEEE Trans. On Electronic Computers***, June 1965, pp. 434-443.

[10] H. Ritter and K. Sculten, "Kohonen's Self-Organizing Maps: exploring their

computational capabilities," *Proc. IEEE IJCNN-88*, San Diego, CA, vol. 1, 1988, pp. 3-16.

[11] K. S. Fu, "Recent Developments in Pattern Recognition," *IEEE Trans. On Computers*, vol. C-29, Oct. 1980, pp. 410-430

[12] K. S. Fu, "Introduction, Digital Pattern Recognition", Edited by K. S. Fu, Springer-Verlag, New York, 1976, pp. 95-134.

[13] M. A. Sattar and S. M. Rahman, "An Experimental Investigation On Bangla Character Recognition System," *Bangladesh Computer Society Journal*, vol. 4, no.1, Dec. 1989, pp. 1-4.

[14] Md. Abul Kashem Mia, "Recognition of Printed Bangla Characters by Syntactic Method of Pattern Recognition," *M. Sc. Engg. Thesis*, Department of Computer Science and Engineering, Bangladesh University of Engineering and Technology, Aug. 1992, pp.1-10.

[15] I. Guyon, P. Albrecht, Y. Le. Cun, J. Denker, W. Hubbard, "Design of A Neural Network Character Recognizer For A Touch Terminal," *Pattern Recognition*, vol.24, no. 2, 1991, pp. 105-119.

[16] S. R. Ramesh, "A Generalized Character Recognition Algorithm: A Graphical Approach," *Pattern Recognition*, vol.22, no.4, 1989, pp.347- 350.

[17] Torfinn Taxt, Jorunn B. Olafsdottir and Morten Dehlen, "Recognition of Handwritten Symbols," *Pattern Recognition*, vol. 23, no. 11, 1990, pp.1155-1166.

[18] Alireza Khotanzad, Jinn-Her Lu, "Classification of Invariant Image Representation Using A Neural Network," *IEEE Trans. on Acoustics, Speech, And Signal*

Processing, vol. 38, no. 6, June 1990, pp 570-580.

[19] C. T. Zahn and R. Z. Roskies, "Fourier Description Of Plane Closed Curves," *IEEE Transaction On Computers*, vol. C-21, 1972, pp. 269-281.

[20] S.A. Dudari, K. J. Breediry and R. B. McGhee, "Aircraft Identification By moments Invariants," *IEEE Transaction On Computers*, vol. C-26, 1977, pp. 39-45.

[21] R. L. Kashyag and R. Chellapa, "Stochastic Models For Close Boundary Analysis: Representation and Reconstruction," *IEEE Transaction On Information Theory*, vol. IT-27, no. 5, 1981, pp. 627-637.

[22] B. Windrow, R. G. Winter and R. A. Bayter, "Layered Neural Nets For Pattern Recognition," *IEEE Trans. Acoustics, Speech and Signal Processing*, vol. ASSP-36, no.7, 1988, pp. 1109-1118.

[23] K. Fukushima, S. Miyake and T. Ito, "Neocognitron: A Neural Network Model For A Mechanism Of Visual Pattern Recognition," *IEEE Trans. Syst., Man and Cyber.*, vol. SMC-13, 1983, pp. 826- 834.

[24] L. Spirokovsha and M. B. Reid, "Connectivity Strategies For Higher-Order Neural Networks," *in Proc. International Conference On Neural Networks*, IJCNN 1990, San Diego, vol.1, 1990, pp. 21-26.

[25] A. P. Reeves, R. J. Prokop, S. E. Andrews, and F. Kuhl, "Three-dimensional shape analysis using moments and Fourier descriptors," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 10, no. 6 , Nov. 1988, pp. 937-943.

[26] A. Kryzak, S. Y. Leung, C. Y. Suen, "Reconstruction of two-dimensional patterns by Fourier descriptors," *Proceedings of 9th International Conference On Pattern*

Recognition, Rome, Italy, Nov. 1988, pp. 555-558.

[27] E. Persoon, K. S. Fu, "Shape discrimination using Fourier descriptors," ***IEEE Transactions on Systems, Man, and Cybernetics***, vol. SMC-7, Mar. 1977, pp. 170-179.

[28] K. S. Fu, "Syntactic Pattern Recognition and Application," Englewood Cliffs, NJ. Prentice-Hall, 1982.

[29] D. Casasent, D. Psaltis, "Position, rotation, and scale invariant correlation," ***Applied Optics***, vol. 15, 1976, pp. 1795-1799.

[30] H. Reitboeck and T. P. Brody, "A transformation with invariance under cyclic permutation for application in pattern recognition," ***Information and Control***, vol. 15, 1969, pp. 130-154.

11968

[31] Y. N. Hsu, H. H. Arsenault, G. April, "Rotational invariant digital pattern recognition using circular harmonic expansion," ***Applied optics***, vol. 15, Jan. 1991, pp. 161-165.

[32] Harris Durcker, Yann Le Cun, "***Improving generalization performance using double backpropagation***," IEEE Trans. On Neural Networks, vol. 3, No. 6, January 1995, pp.31-41.

[33] S. Ergezinger, E. Thomsen, "***An accelerated learning algorithm for multilayer perceptrons: optimization layer by layer***," IEEE Trans. On Neural Networks, vol. 6, No. 1, January 1995, pp.31-41.

[34] Jack S. N. Jean, Jin Wang, "***Weight Smoothing To Improve Network Generalization***," IEEE Trans. On Neural Networks, vol. 5, No.5, September 1995, pp.31-41.

[35] Jacek M. Zurada, "Introduction to Artificial Neural Systems," West Publishing Company, 1992, pp. 1-200.

APPENDIX-A

SIMULATION RESULT TABLES

Table A.1 Average Recognition rate, standard deviation and overall recognition rate for shifted pattern. A total of 80 test patterns, 8 in each class, are used. The results presented in this table are the average on ten trials, with ten different 32-5-10 networks. The first entry of each box in the table indicates the average recognition rate, and the second entry gives the standard deviation.

Number of input units=32, Number of hiddenunits=5										
Shift	Pattern 0	Pattern 1	Pattern 2	Pattern 3	Pattern 4	Pattern 5	Pattern 6	Pattern 7	Pattern 8	Pattern 9
x=5, y=5	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=10, y=10	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=15, y=15	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=20, y=20	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=25, y=25	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=30, y=30	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=40, y=40	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=45, y=45	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%,0	100%, 0	100%, 0	100%, 0	100%, 0
Recognition rate	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%
Overall Recognition= 100%										

Table A.2 Average Recognition rate, standard deviation and overall recognition rate for rotated pattern. A total of 180 test patterns, 18 in each class, are used. The results presented in this table are the average on ten trials, with ten different 32-5-10 networks. The first entry of each box in the table indicates the average recognition rate, and the second entry gives the standard deviation.

Number of input units=32, Number of hidden units=5										
Angle of Rotation	Pattern 0	Pattern 1	Pattern 2	Pattern 3	Pattern 4	Pattern 5	Pattern 6	Pattern 7	Pattern 8	Pattern 9
5°	100%, 0	100%, 0	20%, .16	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	90%, .09
10°	100%, 0	100%, 0	40%, .24	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
15°	100%, 0	100%, 0	20%, .16	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
20°	100%, 0	100%, 0	30%, .21	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
25°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	90%, .09
30°	100%, 0	100%, 0	30%, .21	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	90%, .09
35°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
40°	100%, 0	100%, 0	30%, .21	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
45°	100%, 0	100%, 0	30%, .21	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
50°	100%, 0	100%, 0	30%, .21	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	90%, .09
55°	100%, 0	100%, 0	30%, .21	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
60°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	90%, .09
65°	100%, 0	100%, 0	30%, .21	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
70°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
75°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
80°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
85°	100%, 0	100%, 0	30%, .21	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
90°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
Recognition rate	100%	100%	56.67%	100%	100%	100%	100%	100%	100%	97.22%
Overall Recognition = 95.38%										

Table A.3

Average Recognition rate, standard deviation and overall recognition rate for scaled pattern. A total of 150 test patterns, 15 in each class, are used. The results presented in this table are the average on ten trials, with ten different 32-5-10 networks. The first entry of each box in the table indicates the average recognition rate, and the second entry gives the standard deviation.

Number of input units=32, Number of hidden units=5										
scale factor	Pattern 0	Pattern 1	Pattern 2	Pattern 3	Pattern 4	Pattern 5	Pattern 6	Pattern 7	Pattern 8	Pattern 9
0.5	50%, .25	70%, .21	0%, 0	70%, .21	0%, 0	90%, .09	80%, .16	100%, 0	100%, 0	0%, 0
0.6	100%, 0	100%, 0	20%, .16	80%, .16	10%, .09	80%, .16	90%, .09	100%, 0	100%, 0	0%, 0
0.7	100%, 0	100%, 0	30%,.21	100%, 0	100%, 0	20%, .16	100%, 0	100%, 0	100%, 0	100%, 0
0.8	90%, .09	100% 0	0%, 0	100%, 0	100%, 0	10%, .09	100%, 0	100%, 0	100%, 0	100%, 0
0.9	100%, 0	80%, .16	0%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.0	100%, 0	100%,0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.1	100%, 0	100% 0	0%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.2	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%,0	100%, 0	100%, 0	100%, 0	100%, 0
1.3	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	90%, .09	100%, 0	100%, 0	100%, 0	100%, 0
1.4	100%, 0	100%,0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.5	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.6	100%, 0	90%, .09	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.7	100%, 0	90%, .09	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.8	100%, 0	90%, .09	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	80%, .16
1.9	100%, 0	90%, .09	100%, 0	100%, 0	100%, 0	90%, .09	100%, 0	100%, 0	100%, 0	80%, .16
2.0	100%, 0	90%, .09	100%, 0	100%, 0	100%, 0	90%, .09	100%, 0	100%, 0	100%, 0	80%, .16
Recognition rate	96%	80%	64.67%	96.67%	87.33%	71.3%	98%	100%	100%	50.67%
Overall Recognition= 88.06%										

Table A.4 Average Recognition rate, standard deviation and overall recognition rate for shifted pattern. A total of 80 test patterns, 8 in each class, are used. The results presented in this table are the average on ten trials, with ten different 32-7-10 networks. The first entry of each box in the table indicates the average recognition rate, and the second entry gives the standard deviation.

Number of input units=32, Number of hidden units=7										
Shift	Pattern 0	Pattern 1	Pattern 2	Pattern 3	Pattern 4	Pattern 5	Pattern 6	Pattern 7	Pattern 8	Pattern 9
x=5, y=5	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=10, y=10	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=15, y=15	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=20, y=20	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=25, y=25	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=30, y=30	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=40, y=40	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=45, y=45	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%,0	100%, 0	100%, 0	100%, 0	100%, 0
Recognition rate	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%
Overall Recognition= 100%										

Table A.5 Average Recognition rate, standard deviation and overall recognition rate for rotated pattern. A total of 180 test patterns, 18 in each class, are used. The results presented in this table are the average on ten trials, with ten different 32-7-10 networks. The first entry of each box in the table indicates the average recognition rate, and the second entry gives the standard deviation.

Number of input units =32, Number of hidden units =7										
Angle of Rotation	Pattern 0	Pattern 1	Pattern 2	Pattern 3	Pattern 4	Pattern 5	Pattern 6	Pattern 7	Pattern 8	Pattern 9
5°	100%, 0	100%, 0	30%, .21	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	70%, .21
10°	100%, 0	100%, 0	30%, .21	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
15°	100%, 0	100%, 0	20%, .16	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
20°	100%, 0	100%, 0	30%, .21	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
25°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	90%, .09
30°	100%, 0	100%, 0	20%, .16	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
35°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
40°	100%, 0	100%, 0	20%, .16	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
45°	100%, 0	100%, 0	20%, .16	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
50°	100%, 0	100%, 0	20%, .16	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
55°	100%, 0	100%, 0	20%, .16	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
60°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
65°	100%, 0	100%, 0	20%, .16	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
70°	100%, 0	100%, 0	20%, .16	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
75°	100%, 0	100%, 0	20%, .16	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
80°	100%, 0	100%, 0	20%, .16	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
85°	100%, 0	100%, 0	20%, .16	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
90°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
Recognition rate	100%	100%	39.44%	100%	100%	100%	100%	100%	100%	98.38%
Overall Recognition = 93.78%										

Table A.6 Average Recognition rate, standard deviation and overall recognition rate for scaled pattern. A total of 150 test patterns, 15 in each class, are used. The results presented in this table are the average on ten trials, with ten different 32-7-10 networks. The first entry of each box in the table indicates the average recognition rate, and the second entry gives the standard deviation.

Number of input units =32, Number of hidden units =7										
scale factor	Pattern 0	Pattern 1	Pattern 2	Pattern 3	Pattern 4	Pattern 5	Pattern 6	Pattern 7	Pattern 8	Pattern 9
0.5	0%, 0	70%, .21	0%, 0	100%, 0	0%, 0	70%, .21	10%, .09	100%, 0	100%, 0	100%, 0
0.6	100%, 0	100%, 0	0%, 0	100%, 0	100%, 0	100%, 0	90%, .09	100%, 0	100%, 0	100%, 0
0.7	100%, 0	90%, .09	0%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
0.8	90%, .09	100%, 0	0%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
0.9	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.1	100%, 0	60%, .24	0%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.2	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.3	100%, 0	60%, .24	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.4	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.5	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.6	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.7	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.8	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.9	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
2.0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
Recognition rate	93.33%	92%	66%	100%	93.33%	98%	93.33%	100%	100%	100%
Overall Recognition=93.6%										

Table A.7 Average Recognition rate, standard deviation and overall recognition rate for shifted pattern. A total of 80 test patterns, 8 in each class, are used. The results presented in this table are the average on ten trials, with ten different 64-5-10 networks. The first entry of each box in the table indicates the average recognition rate, and the second entry gives the standard deviation.

Number of input units=64, Number of hidden units=5										
Shift	Pattern 0	Pattern 1	Pattern 2	Pattern 3	Pattern 4	Pattern 5	Pattern 6	Pattern 7	Pattern 8	Pattern 9
x=5, y=5	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=10, y=10	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=15, y=15	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=20, y=20	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=25, y=25	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=30, y=30	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=40, y=40	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=45, y=45	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%,0	100%, 0	100%, 0	100%, 0	100%, 0
Recognition rate	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%
Overall Recognition= 100%										

Table A.8 Average Recognition rate, standard deviation and overall recognition rate for rotated pattern. A total of 180 test patterns, 18 in each class, are used. The results presented in this table are the average on ten trials, with ten different 64-5-10 networks. The first entry of each box in the table indicates the average recognition rate, and the second entry gives the standard deviation.

Number of input units=64, Number of hidden units=5										
Angle of Rotation	Pattern 0	Pattern 1	Pattern 2	Pattern 3	Pattern 4	Pattern 5	Pattern 6	Pattern 7	Pattern 8	Pattern 9
5°	100%, 0	100%, 0	60%, .24	100%, 0	90%, .09	90%, .09	100%, 0	100%, 0	100%, 0	100%, 0
10°	100%, 0	100%, 0	50%, .25	100%, 0	90%, .09	90%, .09	100%, 0	100%, 0	100%, 0	100%, 0
15°	100%, 0	100%, 0	20%, .16	100%, 0	80%, .16	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
20°	100%, 0	100%, 0	90%, .09	100%, 0	80%, .16	100%, 0	100%, 0	100%, 0	80%, .16	100%, 0
25°	100%, 0	100%, 0	100%, 0	100%, 0	80%, .16	90%, .09	100%, 0	100%, 0	30%, .21	100%, 0
30°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	90%, .09	100%, 0	100%, 0	100%, 0	100%, 0
35°	100%, 0	100%, 0	100%, 0	100%, 0	90%, .09	90%, .09	100%, 0	100%, 0	90%, .09	100%, 0
40°	100%, 0	100%, 0	100%, 0	90%, .90	80%, .16	90%, .09	100%, 0	100%, 0	40%, .24	100%, 0
45°	100%, 0	100%, 0	100%, 0	90%, .90	80%, .16	100%, 0	100%, 0	100%, 0	30%, .21	100%, 0
50°	100%, 0	100%, 0	100%, 0	90%, .90	80%, .16	100%, 0	100%, 0	100%, 0	50%, .25	100%, 0
55°	100%, 0	100%, 0	100%, 0	90%, .90	100%, 0	90%, .09	100%, 0	100%, 0	100%, 0	100%, 0
60°	100%, 0	100%, 0	100%, 0	100%, 0	90%, .09	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
65°	100%, 0	100%, 0	100%, 0	90%, .09	100%, 0	90%, .09	100%, 0	100%, 0	40%, .24	100%, 0
70°	100%, 0	100%, 0	100%, 0	100%, 0	80%, .16	90%, .09	90%, .09	100%, 0	30%, .21	100%, 0
75°	100%, 0	100%, 0	100%, 0	100%, 0	80%, .16	90%, .09	100%, 0	100%, 0	100%, 0	100%, 0
80°	100%, 0	100%, 0	90%; .09	100%, 0	80%, .16	90%, .09	100%, 0	100%, 0	100%, 0	100%, 0
85°	100%, 0	100%, 0	90%, .09	100%, 0	80%, .16	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
90°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
Recognition rate	100%	100%	89.44%	92.22%	80%	93.88%	99.44%	100%	77.22%	100%
Overall Recognition = 93.22%										

Table A.9 Average Recognition rate, standard deviation and overall recognition rate for scaled pattern. A total of 150 test patterns, 15 in each class, are used. The results presented in this table are the average on ten trials, with ten different 64-5-10 networks. The first entry of each box in the table indicates the average recognition rate, and the second entry gives the standard deviation.

Number of input units=64, Number of hidden units=5										
scale factor	Pattern 0	Pattern 1	Pattern 2	Pattern 3	Pattern 4	Pattern 5	Pattern 6	Pattern 7	Pattern 8	Pattern 9
0.5	50%, .25	30%, .21	10%, .09	20%, .16	30%, .21	20%, .16	30%, .21	30%, .21	30%,.21	30%,.2 1
0.6	40%, .24	50%, .25	20%, .16	30%, .21	40%, .24	30%, .21	40%, .24	60%, .24	40%,.24	50%, .25
0.7	30%, .21	50%, .25	30%, .21	60%, .24	50% .25	40%, .24	50%, .25	40%, .24	30%,.21	50%, .25
0.8	60%, .24	70%, .21	60%, .24	100%, 0	70%, .21	70%, .21	90%, .09	90%, .09	90%,.09	70%, .21
0.9	90%, .09	100%, 0	90%, .09	80%, .16	70%, .21	80%, .16	80%, .16	70%, .21	80%,.16	60%, .24
1.0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.1	90%, .09	90%, .09	100%, 0	100%, 0	70%, .21	90%, .09	100%, 0	90%, .09	90%,.09	80%, .16
1.2	90%, .09	90%, .09	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	90%, .09
1.3	60%, .24	90%, .09	90%, .09	70%, .21	70%, .21	90%, .09	90%, .09	90%, .09	90%,.09	80%, .16
1.4	90%, .09	100%, 0	100%, 0	90%, .09	90%, .09	90%, .09	90%, .09	100%, 0	100%, 0	80%, .16
1.5	80%, .16	90%, .09	100%, 0	100%, 0	100%, 0	100%, 0	90%, .09	100%, 0	100%, 0	90%, .09
1.6	80%, .16	80%, .16	90%, .09	90%, .09	90%, .09	90%, .09	100%, 0	90%, .09	100%, 0	100%, 0
1.7	80%, .16	90%, .09	100%, 0	90%, .09	90%, .09	80%, .16	100%, 0	80%, .16	100%, 0	90%, .09
1.8	80%, .16	100%, 0	90%, .09	90%, .09	80%, .16	100%, 0	80%, .16	80%, .16	100%, 0	80%, .16
1.9	80%, .16	80%, .16	80%, .16	80%, 16	70%, .21	90%, .09	80%, .16	80%, .16	80%,.16	70%, .21
2.0	80%, .16	80%, .16	80%, .16	80%, .16	80%, .16	90%, .09	80%, .16	80%, .16	80%,.16	70%, .21
Recognition rate	72.67%	75.33%	74%	76%	72.66%	79.33%	72.66%	78.66%	80.67%	68%
Overall Recognition= 75%										

Table A.10 Average Recognition rate, standard deviation and overall recognition rate for shifted pattern. A total of 80 test patterns, 8 in each class, are used. The results presented in this table are the average on ten trials, with ten different 64-7-10 networks. The first entry of each box in the table indicates the average recognition rate, and the second entry gives the standard deviation.

Number of input units=64, Number of hidden units=7										
Shift	Pattern 0	Pattern 1	Pattern 2	Pattern 3	Pattern 4	Pattern 5	Pattern 6	Pattern 7	Pattern 8	Pattern 9
x=5, y=5	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=10, y=10	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=15, y=15	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=20, y=20	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=25, y=25	100%, 0	100%; 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=30, y=30	100%, 0	100%, 0	100%, 0	100%; 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=40, y=40	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=45, y=45	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%,0	100%, 0	100%, 0	100%, 0	100%, 0
Recognition rate	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%
Overall Recognition= 100%										

Table A.11 Average Recognition rate, standard deviation and overall recognition rate for rotated pattern. A total of 180 test patterns, 18 in each class, are used. The results presented in this table are the average on ten trials, with ten different 64-7-10 networks. The first entry of each box in the table indicates the average recognition rate, and the second entry gives the standard deviation.

Number of input units=64, Number of hidden units=7										
Angle of Rotation	Pattern 0	Pattern 1	Pattern 2	Pattern 3	Pattern 4	Pattern 5	Pattern 6	Pattern 7	Pattern 8	Pattern 9
5°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
10°	100%, 0	100%, 0	60%, .24	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
15°	100%, 0	100%, 0	10%, .09	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
20°	100%, 0	100%, 0	60%, .24	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
25°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
30°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
35°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
40°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
45°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
50°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
55°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
60°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
65°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
70°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
75°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
80°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
85°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
90°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
Recognition rate	100%	100%	90.56%	100%	100%	100%	100%	100%	100%	100%
Overall Recognition = 99.06%										

Table A.12 Average Recognition rate, standard deviation and overall recognition rate for scaled pattern. A total of 150 test patterns, 15 in each class, are used. The results presented in this table are the average on ten trials, with ten different 64-7-10 networks. The first entry of each box in the table indicates the average recognition rate, and the second entry gives the standard deviation.

Number of input units=64, Number of hidden units=7										
scale factor	Pattern 0	Pattern 1	Pattern 2	Pattern 3	Pattern 4	Pattern 5	Pattern 6	Pattern 7	Pattern 8	Pattern 9
0.5	0%, 0	60%, .24	80%, .16	10%, .09	100%, 0	100%, 0	10%, .09	100%, 0	10%, .09	10%,.09
0.6	0%, 0	100%, 0	20%, .16	10%, .09	40%, .24	90%, .09	20%, .16	100%, 0	10%, .09	10%,.09
0.7	0%, 0	100%, 0	30%, .21	100%, 0	100%, 0	100%, 0	50%, .25	100%, 0	10%, .09	100%,0
0.8	0%, 0	100%, 0	70%, .21	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	10%, .09	100%,0
0.9	60%, .24	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%,0
1.0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.1	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%,0
1.2	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%,0
1.3	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	80%,.16	100%,0
1.4	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%,0
1.5	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%,0
1.6	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.7	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%,0
1.8	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	10%,.09
1.9	100%, 0	80%, .16	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	10%,.09
2.0	100%, 0	80%, .16	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%,0
Recognition rate	70.67%	97.33%	86.67%	88%	94%	99.33%	92%	100%	74.67%	78%
Overall Recognition= 88.06%										

Table A.13 Average Recognition rate, standard deviation and overall recognition rate for shifted pattern. A total of 80 test patterns, 8 in each class, are used. The results presented in this table are the average on ten trials, with ten different 64-9-10 networks. The first entry of each box in the table indicates the average recognition rate, and the second entry gives the standard deviation.

Number of input units=64, Number of hidden units=9										
Shift	Pattern 0	Pattern 1	Pattern 2	Pattern 3	Pattern 4	Pattern 5	Pattern 6	Pattern 7	Pattern 8	Pattern 9
x=5, y=5	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=10, y=10	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=15, y=15	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=20, y=20	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=25, y=25	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=30, y=30	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=40, y=40	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=45, y=45	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%,0	100%, 0	100%, 0	100%, 0	100%, 0
Recognition rate	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%
Overall Recognition= 100%										

Table A.14 Average Recognition rate, standard deviation and overall recognition rate for rotated pattern. A total of 180 test patterns, 18 in each class, are used. The results presented in this table are the average on ten trials, with ten different 64-9-10 networks. The first entry of each box in the table indicates the average recognition rate, and the second entry gives the standard deviation.

Number of input units=64, Number of hidden units=9										
Angle of Rotation	Pattern 0	Pattern 1	Pattern 2	Pattern 3	Pattern 4	Pattern 5	Pattern 6	Pattern 7	Pattern 8	Pattern 9
5°	100%, 0	100%, 0	20%, .16	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
10°	100%, 0	100%, 0	20%, .16	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
15°	100%, 0	100%, 0	20%, .16	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
20°	100%, 0	100%, 0	20%, .16	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
25°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
30°	100%, 0	100%, 0	20%, .16	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
35°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
40°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
45°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
50°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
55°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
60°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
65°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
70°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
75°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
80°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
85°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
90°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
Recognition rate	100%	100%	95.56%	100%	100%	100%	100%	100%	100%	100%
Overall Recognition=99.56%										

Table A.15 Average Recognition rate, standard deviation and overall recognition rate for scaled pattern. A total of 150 test patterns, 15 in each class, are used. The results presented in this table are the average on ten trials, with ten different 64-9-10 networks. The first entry of each box in the table indicates the average recognition rate, and the second entry gives the standard deviation.

Number of input units=64, Number of hidden units=9										
scale factor	Pattern 0	Pattern 1	Pattern 2	Pattern 3	Pattern 4	Pattern 5	Pattern 6	Pattern 7	Pattern 8	Pattern 9
0.5	0%, 0	0%, 0	50%,.25	0%, 0	0%, 0	60%, .24	0%, 0	100%, 0	0%, 0	0%, 0
0.6	0%, 0	80%, .16	40%, .24	0%, 0	50%,.25	60%, .24	0%, 0	100%, 0	0%, 0	0%, 0
0.7	0%, 0	40%, .24	10%,.09	100%, 0	100%, 0	100%, 0	0%, 0	100%, 0	0%, 0	0%, 0
0.8	0%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	70%, .21	100%, 0	100%, 0	0%, 0
0.9	40%, .24	60%, .24	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	0%, 0
1.0	100%, 0	100%,0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	0%, 0
1.1	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	20%, .16	100%, 0	100%, 0	100%, 0	0%, 0
1.2	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%,0	100%, 0	100%, 0	100%, 0	100%, 0
1.3	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	50%, .25	100%, 0
1.4	100%, 0	100%,0	100%, 0	100%, 0	100%, 0	100%; 0	100%, 0	100%, 0	100%, 0	80%, .16
1.5	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	80%, .16
1.6	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.7	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.8	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.9	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
2.0	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
Recognition rate	69.33%	85.33%	86.67%	86.67%	90%	81.33%	78%	100%	76.67%	50.67%
Overall Recognition=80.64%										

Table A.16 Average Recognition rate, standard deviation and overall recognition rate for shifted pattern. A total of 80 test patterns, 8 in each class, are used. The results presented in this table are the average on ten trials, with ten different 16-5-10 networks. The first entry of each box in the table indicates the average recognition rate, and the second entry gives the standard deviation.

Number of input units=16, Number of hidden units=5										
Shift	Pattern 0	Pattern 1	Pattern 2	Pattern 3	Pattern 4	Pattern 5	Pattern 6	Pattern 7	Pattern 8	Pattern 9
x=5, y=5	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=10, y=10	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=15, y=15	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=20, y=20	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=25, y=25	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=30, y=30	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=40, y=40	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=45, y=45	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%,0	100%, 0	100%, 0	100%, 0	100%, 0
Recognition rate	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%
Overall Recognition= 100%										

Table A.17 Average Recognition rate, standard deviation and overall recognition rate for rotated pattern. A total of 180 test patterns, 18 in each class, are used. The results presented in this table are the average on ten trials, with ten different 16-5-10 networks. The first entry of each box in the table indicates the average recognition rate, and the second entry gives the standard deviation.

Number of input units=16, Number of hidden units=5										
Angle of Rotation	Pattern 0	Pattern 1	Pattern 2	Pattern 3	Pattern 4	Pattern 5	Pattern 6	Pattern 7	Pattern 8	Pattern 9
5°	100%, 0	100%, 0	50%, .25	100%, 0	100%, 0	80%, .16	100%, 0	100%, 0	100%, 0	100%, 0
10°	100%, 0	100%, 0	10%, .09	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
15°	100%, 0	100%, 0	20%, .16	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
20°	100%, 0	100%, 0	20%, .16	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
25°	100%, 0	100%, 0	80%, .16	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
30°	100%, 0	100%, 0	20%, .16	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
35°	100%, 0	100%, 0	30%, .09	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
40°	100%, 0	100%, 0	40%, .24	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
45°	90%, .09	100%, 0	20%, .16	60%, .24	100%, 0	90%, .09	100%, 0	100%, 0	100%, 0	60%, .24
50°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
55°	100%, 0	100%, 0	100%, 0	50%, .25	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
60°	100%, 0	100%, 0	50%, .25	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
65°	100%, 0	100%, 0	30%, .21	70%, .21	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
70°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
75°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
80°	100%, 0	100%, 0	60%,.24	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
85°	100%, 0	100%, 0	40%, .24	90%, .09	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	90%, .09
90°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
Recognition rate	99.44%	100%	53.89%	92.77%	100%	98.33%	100%	100%	100%	97.22%
Overall Recognition= 94.16%										

Table A.18 Average Recognition rate, standard deviation and overall recognition rate for scaled pattern. A total of 150 test patterns, 15 in each class, are used. The results presented in this table are the average on ten trials, with ten different 16-5-10 networks. The first entry of each box in the table indicates the average recognition rate, and the second entry gives the standard deviation.

Number of input units=16, Number of hidden units=5										
scale factor	Pattern 0	Pattern 1	Pattern 2	Pattern 3	Pattern 4	Pattern 5	Pattern 6	Pattern 7	Pattern 8	Pattern 9
0.5	60%, .24	50%, .25	30%, .21	0%, 0	30%, .21	60%, .24	10%, .09	100%, 0	50%, .25	20%, .16
0.6	80%, .16	80%, .16	10%, .09	0%, 0	50%,.25	60%, .24	10%, .09	100%, 0	50%, .25	20%, .16
0.7	80%, .16	60%, .24	40%, .24	40%, .24	100%, 0	100%, 0	0%, 0	100%, 0	60%, .24	50%, .25
0.8	100%, 0	100% 0	30%, .21	50%, .25	100%, 0	100%, 0	70%, .21	100%, 0	100%, 0	0%, 0
0.9	100%, 0	100%, 0	50%, .25	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	60%, 0
1.0	100%, 0	100%,0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.1	100%, 0	60% .24	60%, .24	50%, .25	100%, 0	30%, .21	70%, .21	100%, 0	100%, 0	100%, 0
1.2	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%,0	100%, 0	100%, 0	100%, 0	100%, 0
1.3	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	70%, .21	100%, 0
1.4	100%, 0	100%,0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	80%, .16
1.5	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	80%, .16
1.6	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.7	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.8	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.9	100%, 0	100% 0	100%, 0	100%, 0	100%; 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
2.0	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
Recognition rate	88%	90%	74.66%	79.33%	92%	90%	77.33%	100%	88.67%	74.67%
Overall Recognition= 85.47%										

Table A.19 Average Recognition rate, standard deviation and overall recognition rate for shifted pattern. A total of 80 test patterns, 8 in each class, are used. The results presented in this table are the average on ten trials, with ten different 16-7-10 networks. The first entry of each box in the table indicates the average recognition rate, and the second entry gives the standard deviation.

Number of input units=16, Number of hidden units=7										
Shift	Pattern 0	Pattern 1	Pattern 2	Pattern 3	Pattern 4	Pattern 5	Pattern 6	Pattern 7	Pattern 8	Pattern 9
x=5, y=5	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=10, y=10	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=15, y=15	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=20, y=20	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=25, y=25	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=30, y=30	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=40, y=40	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=45, y=45	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%,0	100%, 0	100%, 0	100%, 0	100%, 0
Recognition rate	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%
Overall Recognition= 100%										

Table A.20 Average Recognition rate, standard deviation and overall recognition rate for rotated pattern. A total of 180 test patterns, 18 in each class, are used. The results presented in this table are the average on ten trials, with ten different 16-7-10 networks. The first entry of each box in the table indicates the average recognition rate, and the second entry gives the standard deviation.

Number of input units=16, Number of hidden units=7										
Angle of Rotation	Pattern 0	Pattern 1	Pattern 2	Pattern 3	Pattern 4	Pattern 5	Pattern 6	Pattern 7	Pattern 8	Pattern 9
5°	100%, 0	100%, 0	20%, .16	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
10°	100%, 0	100%, 0	30%, .21	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
15°	100%, 0	100%, 0	10%, .09	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
20°	100%, 0	100%, 0	20%, .16	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
25°	100%, 0	100%, 0	50%, .25	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
30°	100%, 0	100%, 0	20%, .16	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
35°	100%, 0	100%, 0	60%, .24	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
40°	100%, 0	100%, 0	50%, .25	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
45°	100%, 0	100%, 0	30%, .21	70%, .21	80%, .16	60%, .24	100%, 0	100%, 0	60%, .24	100%, 0
50°	100%, 0	100%, 0	50%, .25	100%, 0	100%, 0	80%, .16	100%, 0	100%, 0	100%, 0	100%, 0
55°	100%, 0	100%, 0	70%, .21	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
60°	100%, 0	100%, 0	80%, .16	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
65°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
70°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
75°	100%, 0	100%, 0	90%, .09	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
80°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	90%, .09	100%, 0
85°	100%, 0	100%, 0	90%, .09	100%, 0	100%, 0	90%, .09	100%, 0	100%, 0	100%, 0	100%, 0
90°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
Recognition rate	100%	100%	60%	98.330%	100%	96.11%	100%	100%	97.22%	100%
Overall Recognition=95.17%										

Table A.21 Average Recognition rate, standard deviation and overall recognition rate for scaled pattern. A total of 150 test patterns, 15 in each class, are used. The results presented in this table are the average on ten trials, with ten different 16-7-10 networks. The first entry of each box in the table indicates the average recognition rate, and the second entry gives the standard deviation.

Number of input units=16, Number of hidden units=7										
scale factor	Pattern 0	Pattern 1	Pattern 2	Pattern 3	Pattern 4	Pattern 5	Pattern 6	Pattern 7	Pattern 8	Pattern 9
0.5	10%, .09	0%, 0	50%,.25	0%, 0	0%, 0	60%, .24	0%, 0	100%, 0	100%, 0	100%, 0
0.6	20%, .16	50%, .25	30%, .24	20%, .16	40%, .16	40%, .24	60%, .24	100%, 0	100%, 0	100%, 0
0.7	0%, 0	60%, .24	20%,.16	0%, 0	100%, 0	100%, 0	50%, .25	100%, 0	100%, 0	100%, 0
0.8	10%, .09	70%, .21	100%, 0	50%, 0	100%, 0	100%, 0	80%, .16	100%, 0	100%, 0	100%, 0
0.9	40%, .24	60%, .24	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.0	100%, 0	100%,0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.1	100%, 0	100% 0	50%, .25	100%, 0	100%, 0	20%, .16	100%, 0	100%, 0	100%, 0	100%, 0
1.2	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%,0	100%, 0	100%, 0	100%, 0	100%, 0
1.3	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.4	100%, 0	100%,0	100%, 0	100%, 0	90%, .09	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.5	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.6	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.7	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.8	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.9	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
2.0	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
Recognition rate	72.1%	82.66%	81.33%	78%	88.66%	88%	86%	100%	100%	1007%
Overall Recognition=87.68%										

Table A.22 Average Recognition rate, standard deviation and overall recognition rate for shifted pattern. A total of 80 test patterns, 8 in each class, are used. The results presented in this table are the average on ten trials, with ten different 16-9-10 networks. The first entry of each box in the table indicates the average recognition rate, and the second entry gives the standard deviation.

Number of input units=16, Number of hidden units=9										
Shift	Pattern 0	Pattern 1	Pattern 2	Pattern 3	Pattern 4	Pattern 5	Pattern 6	Pattern 7	Pattern 8	Pattern 9
x=5, y=5	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=10, y=10	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=15, y=15	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=20, y=20	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=25, y=25	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=30, y=30	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=40, y=40	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
x=45, y=45	100%, 0	100% 0	100%, 0	100%, 0	100%, 0	100%,0	100%, 0	100%, 0	100%, 0	100%, 0
Recognition rate	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%
Overall Recognition= 100%										

Table A.23 Average Recognition rate, standard deviation and overall recognition rate for rotated pattern. A total of 180 test patterns, 18 in each class, are used. The results presented in this table are the average on ten trials, with ten different 32-9-10 networks. The first entry of each box in the table indicates the average recognition rate, and the second entry gives the standard deviation.

Number of input units=16, Number of hidden units=9										
Angle of Rotation	Pattern 0	Pattern 1	Pattern 2	Pattern 3	Pattern 4	Pattern 5	Pattern 6	Pattern 7	Pattern 8	Pattern 9
5°	100%, 0	100%, 0	20%, .16	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
10°	100%, 0	100%, 0	20%, .16	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
15°	100%, 0	100%, 0	20%, .16	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
20°	100%, 0	100%, 0	20%, .16	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
25°	100%, 0	100%, 0	50%, .25	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
30°	100%, 0	100%, 0	20%, .16	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
35°	100%, 0	100%, 0	80%, .16	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
40°	90%, .09	100%, 0	60%, .24	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
45°	60%, .24	100%, 0	30%, .09	100%, 0	100%, 0	100%, 0	100%, 0	70%,.21	100%, 0	100%, 0
50°	100%, 0	100%, 0	60%, .24	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
55°	100%, 0	100%, 0	50%, .25	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
60°	100%, 0	100%, 0	70%, .21	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
65°	100%, 0	100%, 0	40%, .24	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
70°	100%, 0	100%, 0	60%, .24	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
75°	100%, 0	100%, 0	80%, .16	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
80°	100%, 0	100%, 0	70%, .21	100%, 0	100%, 0	100%, 0	100%, 0	90%, .09	100%, 0	100%, 0
85°	100%, 0	100%, 0	60%, .24	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
90°	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
Recognition rate	97.22%	100%	50.55%	100%	100%	100%	100%	97.77%	100%	100%
Overall Recognition= 94.55%										

Table A.24 Average Recognition rate, standard deviation and overall recognition rate for scaled pattern. A total of 150 test patterns, 15 in each class, are used. The results presented in this table are the average on ten trials, with ten different 16-9-10 networks. The first entry of each box in the table indicates the average recognition rate, and the second entry gives the standard deviation.

Number of input units=16, Number of hidden units=9										
scale factor	Pattern 0	Pattern 1	Pattern 2	Pattern 3	Pattern 4	Pattern 5	Pattern 6	Pattern 7	Pattern 8	Pattern 9
0.5	40%, .24	100%, 0	0%, 0	0%, 0	10%, .09	60%, .24	30%, .21	100%, 0	50%, .25	100%, 0
0.6	40%, .24	100%, 0	0%, 0	0%, 0	50%, .25	60%, .24	40%, .24	100%, 0	60%, .24	100%, 0
0.7	70%, .21	100%, 0	0%, 0	50%, .25	70%, .21	100%, 0	70%, .21	100%, 0	70%, .21	100%, 0
0.8	100% 0	100%, 0	60%, .16	100%, 0	100%, 0	100%, 0	70%, .21	100%, 0	100%, 0	100%, 0
0.9	100% 0	100%, 0	40%, .24	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.1	100%, 0	100%, 0	60%, .24	100%, 0	100%, 0	40%, .24	100%, 0	100%, 0	100%, 0	100%, 0
1.2	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%,0	100%, 0	100%, 0	100%, 0	100%, 0
1.3	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.4	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.5	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.6	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.7	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.8	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
1.9	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
2.0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0	100%, 0
Recognition rate	87.33%	100%	70.66%	83.33%	88.66%	90.66%	87.33%	100%	92%	100%
Overall Recognition=90%										