ن الرحيم

Sudan ... e and

T...

Colle... Studies

Title of the Thesis:

Use of two Stage Neural Networks for Recognition of Isolated Arabic Optical Characters

استخدام شبكات عصبية ذات مستويين في ال قراءة الآلية للحروف العربية غير المتصلة

A Thesis Submitted in Fulfillment of the Requirements for the Degree of Master in Computer Science

Name of Candidate: Omer Balola Ali Ahmed

Supervised by: Dr.Mohammed ElHafiz Mustfa

بسم الله الرحمن الرحيم

قال الله سبحانه و تعالي :

(هُوَ الَّذِي جَعَلَ الشَّمْسَ ضِيَاءً وَالْقَمَرَ نُوراً وَقَدَّرَهُ مَنَازِلَ لِتَعْلَمُوا عَدَدَ السِّنِينَ وَالْحِسَابَ مَا خَلَقَ اللَّهُ ذَلِكَ إِلاَّ بِالْحَقِّ يُفَصِّلُ الآيَاتِ لِقَوْمٍ يَعْلَمُونَ)

الآية (5) سورة يونس

DEDICATION

I am dedicates this effort

To hold dear to my mother

And to my brother Mr. Adlan Balola

To all my teachers in all phases of the noble course.

To my dear wife

To my children

ACKNOWLEDGEMENTS

First and foremost, thanks to God, the Creator of human beings and control of all things,

I, therefore, express my deepest gratitude to all those who contributed directly or indirectly, because I would never have been able, by myself, to achieve this

My greatest gratitude and thanks to my supervisor,

Dr. Mohammed ElHafiz Mustafa

Who gave me the idea, support and guidance me for his continuous encouragement and carefulness until I successfully complete this research.

My thanks extended to all staff member of Sudan university of Science and Technology Faculty of Computer Science and Information Technology

And my thanks to my wife Mashaer Yasin

ABSTRACT

The research in Handwritten Arabic Optical Character Recognition area by artificial intelligence scientists continuing until this moment. This thesis, manipulates thirty four forms of Arabic letters, twenty eight forms of the basic letters and extra six forms for some letters. The data set used in this research is an Isolated Handwritten Arabic Characters (IHAC) dataset, which collected by Arabic Language Technology Research Group at Sudan University of Science and Technology. To solve the problem of strong similarity between some Arabic letters, this thesis proposed a two stages classification method. The first stage contains a classifier that classifies the input letter to one of fifteen subgroups. The second stage contains number of classifiers, one classifier for each subgroup (for instance the group ب ت ن ث has a classifier which output only one of these four letters).

The BackPropagation Neural Network (BPNN) is used to design and to train the classifiers. This system achieved **78.77%** recognition rate for testing dataset and **99.4%** for training dataset in the group stage. One classifier for the character stage has been tested and achieved **92.77%** recognition rate for testing dataset.

To address overfitting problem, which reflected by the difference between testing and training results, some overfitting solutions have been testing and their results are encouraged.

المستخلص

الأبحاث في مجال القراءة الآلية للحروف العربية المكتوبة بخط اليد من قبل علماء الذكاء الاصطناعي ما زالت مستمرة حتى هذه اللحظة. هذه الإطروحة تعالج التعرف على 34 شكل للحرف العربية منها 28 شكل لحرف عربي أساسي، و 6 لأشكال أخرى لبعض الحروف. مجموعة البيانات المستخدمة في هذا البحث هي للحروف العربية المنفصلة المكتوبة بخط اليد، والتي تم جمعها من قبل مجموعة بحوث لتكنولوجيا اللغة العربية في جامعة السودان للعلوم والتكنولوجيا.

إقترحت هذه الإطروحة التعامل مع مسألة التعرف الآلي للحروف العربية على مرحلتين: المرحلة الأولى يتم فيها إستخدام مصنف المجموعات الذي يتعرف على المجموعة التي ينتمي لها الحرف المعطى. المرحلة الثانية تتكون من مجموعة من المصنفات، كل مصنف يتعرف على حروف مجموعة واحدة (مثلاً مصنف مجموعة حروف ب ت ن ث ينسب الحرف المعطى لأحد هذه الحروف).

تم إستخدام الشبكة العصبية للإنتشار العكسي لبناء و تدريب المصنفات.

طبق هذا النظام على عدد 34 شكل من أشكال الحروف العربية ، التي وزعت على 15 مجموعة وحقق معدل الدقة 78.77 ٪ لمجموعة بيانات الإختبار ونسبة 99.4% لمجموعة بيانات التدريب. كما تم إختبار مصنف حرف لمجموعة واحدة وحقق معدل الدقة 92.77٪ لمجموعة بيانات الاختبار. و لمعالجة مشكلة التدريب الزائد الواضحة من الفرق بين نتيجة بيانات التدريب ونتيجة بيانات الإختبار، تم إستخدام ثلاث من تقنيات معالجة التدريب الزائد وكانت نتائجها المبدئية مشجعة.

TABLE OF CONTENTS

Chapter Three: 3. Handwritten Arabic Characters Recognition (HACR)

Chapter Four: 4. Backpropagation Neural Network (BPNN)

TABLE OF FIGURES

TABLE OF TABLES

ABBREVIATIONS

1. **OCR**: Optical Character Recognition.
2. **BPNN**: Back-Propagation Neural Network.
3. **HACR**: Handwritten Arabic Character Recognition.
4. **PR**: Pattern recognition.
5. **ANNs**: Artificial Neural Networks.
6. **MCR:** Magnetic Character Recognition.
7. **SUST-ARG:** Sudan University for Sciences and Technology Arabic Recognition Group.

CHAPTER ONE

INTRODUCTION

CHAPTER ONE

INTRODUCTION

1.1 Overview

Character recognition systems can contribute tremendously to the advancement of the automation process and can improve the interaction between man and machine in many applications, including office automation, check verification and a large variety of banking, business and data entry applications. Very little research has gone into character recognition in Arabic due to the difficulty of the task and lack of researchers interested in this field [1]. As the Arab world becomes increasingly computerized and mobile, and technology becomes increasingly ubiquitous, the need for a natural interface becomes apparent. Typing is not a natural user-friendly interface, leaving handwriting recognition as a viable alternative [2-3].

For the above reasons and because of the benefits of Optical Character Recognition (OCR) and after carefully studying the problems for recognized Arabic characters, has been constructing recognition classifier.

This classifier design with a technique for recognizing handwritten Arabic characters using PCA feature extraction and neural networks (BackPropagation Neural network) pattern classification methods.

1.2 Research Scope

The research in Handwritten Arabic Character Recognition (HACR) usually includes features extraction stage, which is consist of presented images of character, scanned these images, preprocessing and extracted features. Last stage includes training, classification and output results. Concentrating of training and classification.

1.3 Research Problem

Neural networks algorithms need a long time for training, the large number of multi parameters learning and that need to repeat the great experiments, also the backpropagation algorithm facing the problem of overtraining, and there are some algorithms that solve the problem of overtraining need high space of memory.

1.4 Thesis Objective

The aim of this thesis is to build a network classifier to recognize Handwritten Arabic character using SUST-ARG dataset with back-propagation neural network.

1.5 Thesis Contents

Chapter one is an introduction to the research. Chapter two explains the concept of pattern recognition and OCR structure. Chapter three is of the handwriting Arabic character recognition. Chapter four explain introduction of the neural network and the backpropagation neural network and its algorithm. Chapter five explains introduction of digital images, chapter six interested the SUST-ADG, implementation processes, and show the experiments results and discuss the result. Chapter eight provide conclusion, recommendation and further work.

CHAPTER TWO
OPTICAL CHARACTER RECOGNITION

CHAPTER TWO

OPTICAL CHARACTER RECOGNITION

2.1 PATTERN RECOGNITION

2.1.1Overview

Pattern Recognition is the research area that studies the operation and design of systems that recognize patterns in data.

Also it one of branches of Artificial Intelligence concerned with the classification or description of observations. Pattern recognition aims to classify data.

Pattern recognition (PR) has a long history, but before the 1960s it was mostly the output of theoretical research in the area of statistics. As with everything else, the advent of computers increased the demand for practical applications of pattern recognition, which in turn set new demands for further theoretical developments [4].

2.1.2 What is Pattern Recognition?

(Duda and Hart) defined the pattern recognition is a field concerned with machine recognition of meaning regularities in noisy of complex environments. [5] (Pavlidis) defined pattern recognition in his book: “the word pattern is derived from the same root as the word patron and, in his original use, means something which is set up as a perfect example to be imitated. Thus pattern recognition means the identification of the ideal which a given object was made after.” [6] (Gonzalez,Thomas) defined pattern recognition as a classification of input data via extraction important features from a lot of noisy data. [7] (Watanabe) said that pattern recognition can be looked as categorization problem, as inductive process, as structure analysis, as discrimination method and so on. [8] (Fukunaga) defined pattern recognition as” A problem of estimating density functions in a high-dimensional space and dividing the space into the regions of categories of classes.”[9] (Robert P.W. Duin) described the nature of pattern recognition is engineering; the final aim of Pattern recognition is to design machines to solve the gap between application and theory. [10] (Sergios Theodoridis,) Pattern recognition is a scientific discipline whose aim is the classification of the objects into a lot of categories or classes. Pattern recognition is also an integral part in most machine intelligence system built for decision making [11].

2.1.3 Pattern Recognition Applications

- Optical character recognition.
- Biomedical (Neuroscience, ECG monitoring, drug development, DNA sequences).
- Speech recognition.
- Industrial inspection.
- Biometric (face recognition, fingerprint, iris recognition).
- Military applications(Automated Target Recognition).

2.1.4 Methods of Pattern Recognition

2.1.4.1 Statistical Methods

Statistical classification methods, are based on the Bayes decision theory, which aims to minimize the loss of classification with given loss matrix and estimated probabilities. According to the class-conditional probability density estimation approach, statistical classification methods are divided into parametric and nonparametric ones [12].

2.1.4.2 Structural Methods

Structural methods represent a pattern as a structure (string, tree, or graph) of flexible size. The structural representation records the stroke sequence or topological shape of the character pattern, and hence resembles well to the mechanism of human perception. In recognition, each class is represented as one or more structural templates, the structure of the input pattern is matched with the templates and is classified to the class of the template of minimum distance or maximum similarity. The structural matching procedure not only provides an overall similarity but also interprets the structure of the input pattern and indicates the similarities of the components [12].

2.1.4.3 Artificial Neural Networks

Artificial Neural Networks (ANNs) were initially studied with the hope of making intelligent perception and cognition machines by simulating the physical structure of human brains. The principles and algorithms of ANNs have found numerous applications in diverse fields including pattern recognition and signal processing. A neural network is composed of a number of interconnected neurons, and the manner of interconnection differentiates the network models into feedforward networks, recurrent networks, self-organizing networks [12], and so.

2.1.4.4 Fuzzy Logic

Fuzzy logic is an approach to computing based on "degrees of truth" rather than the usual "true or false" (1 or 0). The idea of fuzzy logic was first advanced by

Dr. Lotfi Zadeh of the University of California at Berkeley in the 1960s. Dr. Zadeh was working on the problem of computer understanding of natural language. Natural language (like most other activities in life and indeed the universe) is not easily translated into the absolute terms of 0 and 1. (Whether everything is ultimately describable in binary terms is a philosophical question worth pursuing, but in practice much data we might want to feed a computer is in some state in between and so, frequently, are the results of computing.) [13].

2.1.5 Pattern Recognition Stages

Pattern recognition system involves the following processes:

2.1.5.1 Segmentation

Segmentation is the partition of the whole data into single objects.

2.1.5.2 Feature Extraction

The traditional goal of the feature extractor is to characterize an object by making numerical measurements. Good features are those whose values are similar for objects belonging to the same category and distinct for objects in different categories. Feature extraction is very problem dependent.

Usually one feature is not enough to differentiate between objects from different categories. Multiple features representing the same object are organized into feature vectors. The set of all possible feature vectors is called the feature space. Invariant features are such that remain the same if something (irrelevant) is done to the sensed input [14].

2.1.5.3 Classification

The task of the classifier component is to use the feature vector provided by the feature extractor to assign the object to a category.

The abstraction provided by the feature vector representation of the input data enables the development of a largely domain-independent theory of classification. Essentially the classifier divides the feature space into regions corresponding to different categories. The degree of difficulty of the classification problem depends on the variability in the feature values for objects in the same category relative to the feature value variation between the categories.

Variability is natural or is due to noise. Variability can be described through statistics leading to statistical pattern recognition [14].

2.1.5.4 Post processing

The post-processor uses output of the classifier to decide on the recommended action. However, sometimes-different actions can have different costs. These costs can be taken into account when designing an optimal classification system. The post-processor can also exploit the context or combine results of several classifiers.

Pattern recognition is one of the attractive and growing up research domain. Magic applications come true as a result of it [14].

2.2 Optical Character Recognition

2.2.1 Overview

Character Recognition or Optical Character Optical Character Recognition (OCR) is the process of converting scanned images of machine printed or handwritten text (numerals, letters, and symbols), into a computer process able format (such as ASCII). This is the technology long used by libraries and government agencies to make lengthy documents quickly available electronically. Advances in OCR technology have spurred its increasing use by enterprises. For many document-input tasks, OCR is the most cost-effective and speedy method available. And each year, the technology frees acres of storage space once given over to file cabinets and boxes full of paper documents. Before OCR can be used, the source material must be scanned using an optical scanner (and sometimes a specialized circuit board in the PC) to read in the page as a bitmap (a pattern of dots). Software to recognize the images is also required. The

OCR software then processes these scans to differentiate between images and text and determine what letters are represented in the light and dark areas. Older OCR systems matched these images against stored bitmaps based on specific fonts. Today's OCR engines add the multiple algorithms of neural network technology to analyze the stroke edge, the line of discontinuity between the text characters, and the background. Allowing for irregularities of printed ink on paper, each algorithm averages the light and dark along the side of a stroke, matches it to known characters and makes a best guess as to which character it is. The OCR software then averages or polls the results from all the algorithms to obtain a single reading.

(OCR) is an important research area in pattern recognition. The objective of an OCR system is to recognize alphabetic letters, numbers, or other characters, which are in the form of digital images, without any human intervention [15].

2.2.2 Different Families of Character Recognition

The different areas covered under the general term "character recognition" are either the on-line or the off-line CR, each having its dedicated hardware and recognition methods (see figure 2.1).

2.2.2.1 On-line Character Recognition

In on-line character recognition applications, the computer recognizes the symbols as they are drawn. The typical hardware for data acquisition is the digitizing tablet, which can be electromagnetic, electrostatic, pressure sensitive, and so on; a light pen can also be used. As the character is drawn, the successive positions of the pen are memorized (the usual sampling frequencies lay between 100Hz and 200Hz) and are used by the recognition algorithm [16].

2.2.2.2 Offline Character Recognition

Off-line character recognition is performed after the writing or printing is completed.

Off-line handwriting recognition refers to the process of recognizing characters that have been scanned from a surface (such as a sheet of paper) and are stored digitally in gray scale format. After being stored, it is conventional to perform further processing to allow superior recognition. In case of online handwritten character recognition, the handwriting is captured and stored in digital form via different means. Usually, a special pen is used in conjunction with an electronic surface. As the pen moves across the surface, the two-dimensional coordinates of successive points are represented as a function of time and are stored in order [16]. Two families are usually distinguished: magnetic and optical character recognition.

1. Magnetic Character Recognition (MCR)

In MCR, the characters are printed with magnetic ink. The reading device can recognize the characters according to the unique magnetic field of each character. MCR is mostly used in banks for check authentication [16].

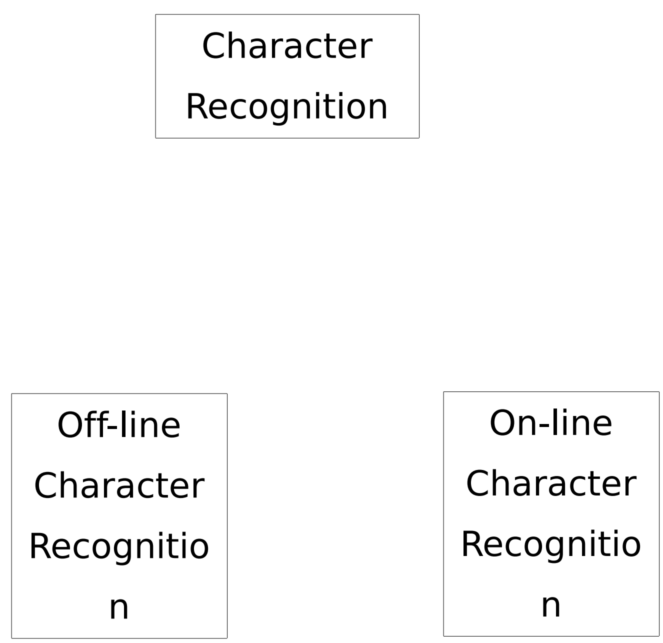
2. Optical Character Recognition (OCR)

OCR deals with the recognition of characters acquiring by optical means, typically a scanner or a camera. The characters are in the form of pixelized images, and can be either printed or handwritten, of any size, shape, or orientation. The OCR can be subdivided into handwritten character recognition and printed character recognition. Handwritten Character Recognition is more difficult to implement than printed character recognition due to diverse human handwriting styles and customs. In printed character recognition, the images to be processed are in the forms of standard fonts like Times New Roman, Arial, Courier, etc [16].

3. Comparison between Offline and Online Approaches

– Online recognition system: the system accepts the movement of pen from the hardware such as graphic tablet, light pen; and there is a lot of information during the input process available such as: current position, movement's direction, stopping points, starting points, strokes order. Al-Taani In [17] proposed online Arabic digital recognition system.

- Offline recognition system: the system accept image as input from scanner, offline recognition in more difficult than online character recognition: because of not availability of contextual information and prior knowledge like text position, size of text, order of strokes, stop points, and start points, furthermore there are a noises in image while the noises in online recognition near to be absent. Comprehensive papers [18, 19, and 20] discussed the scientific progress in the Offline word recognition.



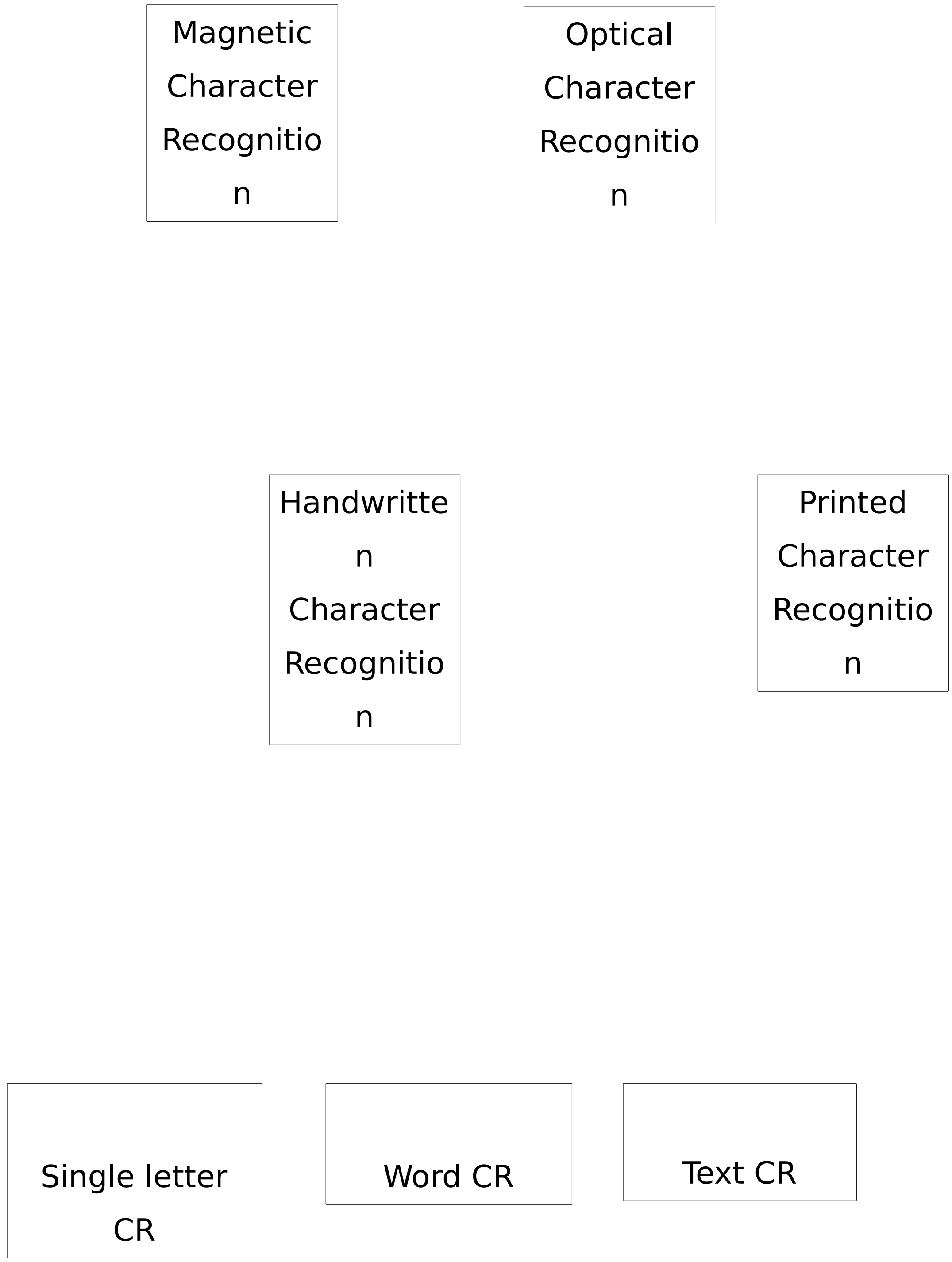


Figure 2.1: The different families of character recognition.

CHAPTER THREE
HANDWRITTEN ARABIC CHARACTERS RECOGNITION (HACR)

CHAPTER THREE

HANDWRITTEN ARABIC CHARACTERS RECOGNITION

Arabic Characters 3.1

3.1.1 Overview

Arabic is a language spoken by Arabs in over 22 countries, and roughly associated with the geographic region of the Middle East and North Africa, but is also spoken as a second language by several Asian countries in which Islam is the principle religion (e.g. Indonesia). However, non-Semitic languages such as Farsi, Urdu, Malay, and some West African languages such as Hausa have adopted the Arabic alphabet for writing [21]. Arabic, one of the six United Nations official languages, is the mother tongue of more than 300 million people. during the 7th century new Arabic letters were created by adding dots to existing letters to avoid ambiguities with Alhajaj Bin Yousif how Bani Umiha country president [22].

3.1.2 Main Types of Written Arabic Text

3.1.2.1 Classical Arabic

The language of the Qur'an and classical literature, it is pure Arabic and differs from Modern Standard Arabic mainly in style and vocabulary, some of which remains undefined, unknown, and implicit [22].

3.1.2.2 Modern Standard Arabic

The universal language of the Arabic-speaking world understood by all its speakers. In addition to pure Arabic, it includes new foreign arabized words, scientific and technological terms. It is the Academic Language and the language of the vast majority of written material and formal TV shows, lectures, etc. MSA has a clear, well-determined vocabulary as well as explicit, well-established morphological rules and grammar rules [22].

Due to the cursive nature of the script, there are several characteristics that make recognition of Arabic distinct from the recognition of Latin scripts or Chinese (see Figure 3.1).

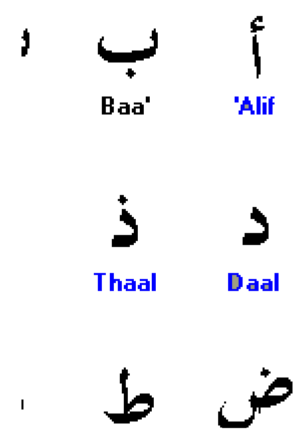


Figure (3.1): Letters of the Isolated Arabic Alphabet

3.1.3 Arabic Alphabet

Arabic has 28 letters in the alphabet. It is based on 18 distinct shapes that vary according to their connection to preceding or following letters. Using a combination of dots and symbols above and below these shapes, the full complement of 28 consonants can be constructed. The Arabic character set is composed of 28 basic characters. Fifteen of them have dots and 13 are without dots. Dots above and below the characters, play a major role in distinguishing some characters that differ only by the number or location of dots.

Arabic is a cursive language. There are no capital letters and some letters are not connected to the letters that follow them (letters in Figure 3.2).

Arabic is written from right to left. Arabic has four forms for each letter depending on the position of the letter in each word. These are beginning, middle, end and isolated (see Figure 3.2)[23]. Arabic contains dots and other small marks that can change the meaning of a word.

Letter Name	*Isolated Form*	*Final Form*	*Medial Form*	*Initial Form*
Alef	ا	ـا		
Ba	ب	ـب	ـبـ	بـ
Ta	ت	ـت	ـتـ	تـ
Tha	ث	ـث	ـثـ	ثـ
Jeem	ج	ـج	ـجـ	جـ
Ha	ح	ـح	ـحـ	حـ
Kha	خ	ـخ	ـخـ	خـ
Dal	د	ـد		

Figure. 3.2: Samples of Various Arabic Letter Forms.

3.1.4 Characteristics of Arabic Characters

This section provides a comprehensive list of characteristics of the Arabic characters with figures to illustrate the concepts.

Arabic is written from right to left in printed forms (Figure 3.3). Arabic has 28 basic characters Table (3.1). No upper or lower cases exist in Arabic.

جامعة السودان للعلوم و التكنلوجيا
هي أكثر الجامعات السودانية إهتماماً
بالتخصصات التطبي قية والعملية،
حيث أنها تضم كثيراً من المعامل
التطبي قية والورش الهندسية .

Fig. 3.3: A sample of printed Arabic text.

2. The shape of the character varies according to its position in the word (Table 3.1). Each character has either two or four different forms. Off course this will increase the number of classes to be recognized from 28 to 100 (other shapes like ء , ى and their forms are not included, they will be discussed shortly).

3. Arabic is always written cursively. Words are separated by spaces. However, there are 6 characters can be connected only from the right, these are: ا , د , ذ , ر , ز , و.

4. Arabic characters are ‘normally’ connected on an imaginary line called baseline as shown in Figure 3.4.

5. Clearly, the above six characters (ا , د , ذ , ر , ز , و), if appeared in a word, will cause the word to be divided into blocks of connected components called subwords. Thus a word can have one or more subwords. Subwords are also separated by spaces, but usually shorter than the one between words. So, this issue need to be considered to avoid segmenting a word into two part. Examples of words in which all characters are connected: فسيكفيكهم , مجيد , خليل . Examples of words consist of subwords: . طريق , شهادة , ورود

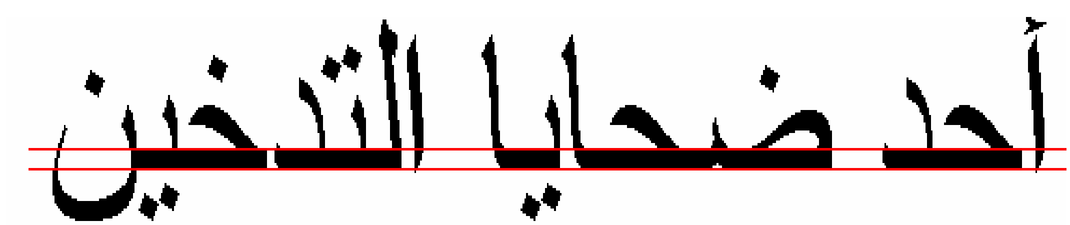


figure. 3.4: A figure shows the base-line.

6. Character width differs from one character to another, see for example ک and ا They differ also in height, for instance: د and ل . In addition to that, the width and
height vary across the different shapes of the same character in different position in the word. For example ب and ب . Another example is the difference between ح and ح in terms of height.

7. To justify an English paragraph, we used to increase the spaces between the words. To justify an Arabic paragraph, the common practice by many systems is to elongate the baselines of the words of a line, not by inserting inter-word spaces [20]. Figure (2.3) shows the same paragraph presented in Figure (2.1) but justified from both sides; notice the elongation of the baseline.

Table (3.1): illustrates position of Arabic letters

Character Name		Isolated	Begin	Middle	End	Others character shapes
Alef	الف	ا	ا	ـا	ـا	إ أ
Ba'	باء	ـب	بـ	ـبـ	ـب	ئ
Ta'	تاء	ـت	تـ	ـتـ	ـت	ؤ ء
Tha'	ثاء	ـث	ثـ	ـثـ	ـث	لا
Jeem	جيم	ج	جـ	ـجـ	ـج	
H'a'	حاء	ح	حـ	ـحـ	ـح	
Kha'	خاء	خ	خـ	ـخـ	ـخ	
Dal	دال	د	د	ـد	ـد	
Thal	ذال	ذ	ذ	ـذ	ـذ	
Raa'	راء	ر	ر	ـر	ـر	
Zain	زين	ز	ز	ـز	ـز	
Seen	سين	س	سـ	ـسـ	ـس	
Sheen	شين	ش	شـ	ـشـ	ـش	
Sad	صاد	ص	صـ	ـصـ	ـص	
Dad	ضاد	ض	ضـ	ـضـ	ـض	
Taa	طاء	ط	طـ	ـطـ	ـط	
Zain	ظين	ظ	ظـ	ـظـ	ـظ	
Ain	عين	ع	عـ	ـعـ	ـع	
Gain	غين	غ	غـ	ـغـ	ـغ	
Fa	فاء	ف	فـ	ـفـ	ـف	
Gaf	قاف	ق	قـ	ـقـ	ـق	
Kaf	كاف	ك	كـ	ـكـ	ـك	
Lam	لام	ـل	لـ	ـلـ	ـل	
Mim	ميم	م	مـ	ـمـ	ـم	

Non	نون	ـن	نـ	ـنـ	ـن	
Ha	هاء	هـ	هـ	ـهـ	ـه	
Waw	واو	و	و	ـو	ـو	
ya	ياء	ي	يـ	ـيـ	ـي	

قام الدكتور محمد الحافظ عميد كلية
علوم الحاسوب وتقنية المعلومات
بجولة حول أقسام الكلية ووجة
بالتركيز على تكريس إستخدام الجودة
الشاملة في التدريس **والإدارة .**

Fig. 3.5 : The text presented in Figure (3.4) but justified from both sides.

8. Fifteen characters have dots associated with the character, they can be above or below the primary part (refer to Table 3.1).
9. Some characters share the same primary part [21] and distinguished from each other by the secondary part (the dots), see Table (3.1).
10. Some characters contain closed loop (refer to Table 3.1). Loop is an important feature to describe a character. Character هـ contains two loops. 11. Alif-Maqsora (ى), shares the same primary part of character ي but without dots. This character appears only at the end of the word.
12. Hamza (ء) zigzag shape, is not really a letter, it is a complementary shape appears in the following cases:

a) Always: with character ک in the separated or final forms. Here it is used to distinguish it from letter ل.

b) Separated: May appear at the beginning, in the middle or at the end of a word. This is the only case in which the character can't be connected from both sides. Examples: .ءامن , ال قراءة , السماء

c) Occasionally: to indicate a pause (short stop) in the pronunciation of the vowel. In most instances you will see it with a "hamza carrier", that is either ا , و or ي with a hamza floating above or bellow one of them. Examples: . سبأ, فإنهم, مؤمن, يومئذ , شيء

13. There are only three characters that represent vowels, ا , و or ي . However, there are other shorter vowels represented by diacritics in the form of overscores or underscores (Table 3.1).

15. Sometimes, the dots are omitted from characters ق , ن , ش when they appear at the end of a word. The short strokes of س and ش sometimes omitted in handwriting.

The small stroke of character ص , when it is followed by م -for instance-, is omitted. See Figure (3.1) for a sample of Arabic handwritten text that shows these examples and others.

16. Dots may appear as two separated dots, touched dots, hat or as a stroke.

3.2 Handwritten Arabic Character recognition

3.2.1 Introduction

Handwritten Arabic Character Recognition (HACR) is the system attempts to recognize a text that has been written by a person (not a machine).

Handwriting recognition attracted the attention of researchers since the origin of computers. Nowadays, the technological progress made in the field of computer architectures and peripheral devices as well as the advances of scientific research make possible the development of new systems for handwriting recognition. In addition, the series of Workshops on Frontiers in Handwriting Recognition and the International Conferences on Document Analysis and Recognition provide continuously new stimuli to researchers, and point out those industrial applications of handwriting recognition can become a large business [24].

3.2.2 Architecture of HACR

HACR is a system which loads a letter image, preprocesses the image, converts image to vectors matrix by matlab software extracts proper image features, classify the characters based on the extracted image features package PCA (Principle Common Analysis) and the known features are stored in the image model library, and recognizes the image according to the degree of similarity between the loaded image and the image models. In preprocessing noise is removed from image so that recognition algorithms can be applied on image easily. Some feature extraction algorithms only deal with the contours of the image while some algorithms calculate every pixel of the image [25]. The preprocessing stage, which includes binarization, edge detection, gap filling, and segmentation and so on, can make the initial image more suitable for later computation.

3.2.3 Application of Offline Handwritten Recognition

Some of the more important applications of offline handwritten recognition are discussed in the following section.

1- Cheque Reading: Offline handwritten recognition is basically used for cheque reading in banks. Cheque reading is the very important commercial application of offline handwritten recognition. Handwritten recognition system plays very important role in banks for signature verification and for recognition of amount filled by user [26].

2- Postcode Recognition: Handwritten recognition system can be used for reading the handwritten postal address on letters. Offline handwritten recognition system used for recognition handwritten digits of postcode. HCR can be read this code and can sort mail automatically.

3- Form Processing HCR can be also used for form processing. Forms are normally used for collecting the public information. Replies of public information can be handwritten in the space provided [26].

4- Signature Verification: HCR can be also used for identify the person by signature verification. Signature identification is the specific field of handwritten identification in which the writer is verified by some specific handwritten text. Handwritten recognition system can be used for identify the person by handwriting, because handwriting may be vary from person to person [26].

5- **Newspapers:** can be used in newspaper and journals to recognition columnists for writers.

3.2.4 Handwritten Character Recognition Position

The Arabic alphabet contains of 28 letters. Each has between two and four shapes, depends on position of the letter with in its word or sub-word. The shape corresponds to the four position beginning, middle, end, and isolated as shown in figure.3.6.

1- **Beginning:** Is the character at the beginning of the word and the first letter is written of the word.

2- **Middle:** Character is located between two characters in word.

3- **End:** Is the character at the bottom of the word.

4- **Isolated:** Is the character which is located individually in the first word or the middle or the last like أ.ا.

3.2.5 Arabic Handwritten Character Difficulties

The following difficulties are Arabic handwritten recognition:

- Arabic characters can have more than one shape according to their position in a word whether at the beginning, middle, final, or stand alone, as shown in Figure 3.6.
- Different writers or even the same writer under different conditions will write some Arabic characters in completely different ways.
- Other characters have very similar contours and are difficult to recognize especially when non-character and external objects are present in the scanned image. Figure 3.6 shows a list of such characters.

Handwritten Arabic characters depend largely on contextual information

Optical Character Recognition

Handwritten Character Recognition

Printed Character Recognition

Word CR

Single letter CR

Text CR

Begin letter

Middle letter

End letter

Isolated letter

Figure 3.6: The different families of Arabic character recognition

CHAPTER FOUR
BACKPROPAGATION NEURAL NETWORK (BPNN)

CHAPTER FOUR

BACKPROPAGATION NEURAL NETWORK (BPNN)

4.1 Neural Networks

4.1.1 Introduction

An Artificial Neural Network (ANN) is an information processing paradigm that is inspired by the way biological nervous systems, such as the brain, process information. The key element of this paradigm is the novel structure of the information processing system. It is composed of a large number of highly interconnected processing elements (neurones) working in unison to solve specific problems. ANNs, like people, learn by example. An ANN is configured for a specific application, such as pattern recognition or data classification, through a learning process. Learning in biological systems involves adjustments to the synaptic connections that exist between the neurons [27].

4.1.2 The Neuron

The key word to understand the brain structural organization and function is the neuron. The idea of the neurons was introduced by Ramon y Cajal in 1911 and refers to the fundamental logical units that the whole Central Nervous System is consisted of. It is indicative that the neuron lies somewhere in the middle of the structural organization of the brain. A neuron is a nerve cell with all of its processes. Neurons are one of the main distinguishing features of animals (plants do not have nerve cells). Neurons come in a wide variety of shapes, sizes and functionality in different parts of the brain. The number of different classes of neurons that have been identified in humans lies between seven and a hundred (the observed wide variety in that estimation n is related to how restrictively a class of neurons [28].

However, the behavior of a neuron can be captured by a simple model as
show in fig 4.1.

f(x)

Σ

W1

W2

W3
B
I1

I2

I3
1
x
a
Inputs
Summation of weighted inputs
Thresholding output

fig.4.1: simple model of an artificial neuron

Where **I1**, **I2**, and **I3** are the inputs, **W1**, **W2**, and **W3** are the weights, **B** is the bias, **x** is an intermediate output, and **a** is final output. The equation for **a** is given by $a = f(W_1 I_1 + W_2 I_2 + W_3 I_3 + B)$ where *f* could be any function. Most often, *f* is the sign of the argument (i.e. 1 if the argument is positive and -1 if the argument is negative), linear (i.e. the output is simply the input times some constant factor), or some complex curve used in function matching (not needed here). For this model we will use the first case where *f* is the sign of the argument for two reasons: it closely matches the 'all or nothing' property seen in biological neurons and it is fairly easy to implement.

4.1.3 Advantages and Disadvantages of using Neural Network

1. The advantages are

1. Perform tasks with multivariate non-linear problems.

2. Network will be able to continue running, even with minor error, pertaining to their parallel nature.

3. Implemented in any application without any problem.

4. Learns from the training data given; create its' own organization or representation of any information it receives.

2. The disadvantages are

1. Large sample size to obtain accuracy, thus more processing power and time are required.

1 2. Requires training to operate, with trial and error of training parameters to achieve optimal results.

4.1.4 Neural Network Types Comparison

A neural network is made up of layers of nodes and each node has a link called weight. The main objective of the networks are to find the suitable weight to associate with each node, thus the desired output will be achieved. Table 4.1 shows the comparison of the various networks [30- 33].

4.1.5 Neural Networks Architectures

An artificial neural network is defined as a data processing system consisting of a large number of simple highly interconnected elements (artificial neurons) in an architecture inspired by the structure of the cerebral cortex of the brain (Tsoukalas and Uhrig ,1997) .there are several classes of NN ,classified according to their learning mechanisms. However, follow are fundamentally different classes of networks. All the three classes employ the digraph structure for their representation [29].

Table 4.1: Comparison of the various neural networks.

Network	**Applications**	**Advantages**	**Disadvantages**
Back-propagation	Pattern recognition Noise removal Classification Mapping Signal/image segmentation	Good at forming internal representations of features in input data or classification and other tasks. Easy implementation.	Long training time Can get stuck in local minima resulting in sub-optimal solutions.
Adaptive Resonance Theory	Pattern recognition	Explicit rules and templates not required Retain learned categories	Sensitive to noise in their input. Not suitable for continuous, incremental on-line training. Category representations are localized instead of being distributed over multiple nodes.
Adaline\ Madaline	Adaptive signal filtering. Adaptive equalization	Fast and easy to implement.	Linear relationship between input and output assume.

Boltzmann Machine	Pattern recognition. Optimization	Able to form optimal representation of pattern features. Follows energy surface to obtain optimization minima.	Long learning time.

Network	**Applications**	**Advantages**	**Disadvantages**
Hopfield	Autoassociative recall. Optimization	Simple concept. Proven dynamic stability. Easy to implement in VLSI.	Unable to learn new state. Poor memory storage. Many spurious states returned.
Learning Vector Quantization	Autoassociative recall. Data compression.	Able to self organize vector representations of probability distributions in data. Rapid execution after training is complete.	Unresolved issues in selecting numbers of vectors to use and length of time for training. Slow training.
Recurrent	Robotics control. Speech recognition. Sequence element prediction	Good for classification. Mapping time varying information.	Complex network. May be difficult to train and optimise.

4.1.5.1 Single Layer Feedforward Network

This type of network comprises of tow layers, namely the input layer and the output layer. The input layer neurons receive the input signals and the output layer neurons receive the output signals. The synaptic links carrying the weights (w) connect every input neuron to the output neuron but not vice-versa. The input layer merely transmits the signals to the output layer. Figure 4.2 illustrates an example network.

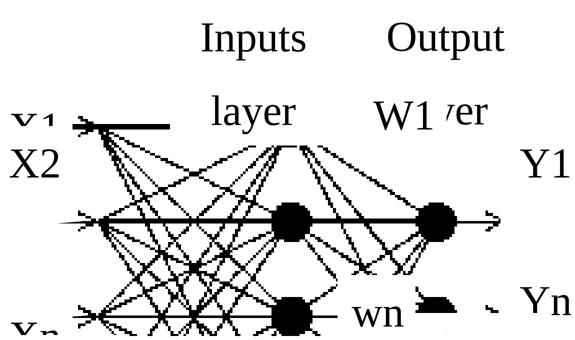
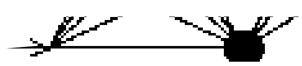


Fig.4.2 single layer feedforward network



4.1.5.2 Multilayer Layer Feedforward Network

This network, as its name indicates is made up of multiple layers. Thus, architecture of this class besides possessing an input and output layers also have one or more intermediary layers called hidden layers. The computation units of the hidden are known as the hidden neurons or hidden units. The hidden layer aids in performing useful intermediary computation before directing the input to the output layer. The input layer neurons are linked to the hidden layer neurons and the weights on these linked are referred to as input hidden layer weights. Again, the hidden layer neurons are linked to the output layer neurons and the corresponding weights are referred to as hidden output layer weights. Multilayer feedforward network with I input neurons, m1 neurons in the hidden layer, m2 neurons in the second hidden layer and n output neurons in the output layer is written as 1 – m1 – m2 –n. Figure 4.3 illustrates a multilayer feedforward network with configuration 1– m – n.

X1
X2
Xi
y1
V11
V1m
V2m
V3m
yj
Z2
Zk
Z1
V21
V31
W11
W12
W1n
W22
W21
W2n
Inputs layer
Hidden layer
Output layer

Fig.4.3:A multilayer feedforward network

Xi : input neurons, yj : hidden neurons, zk : output neurons, vij : input hidden layer weight, wjk : output hidden layer.

4.1.5.3 Recurrent Networks

These networks differ from feedforward network architecture in the sense that there is at least one feedback loop. Thus, in these networks, for example, there could exist one layer with feedback connection as shown in fig. 4.4. There could also be neurons with self-feedback links, i.e. the output of a neuron is fed back into itself as input.

Inputs layer
Hidden layer
Output layer

Feedback link

Fig.4.4: A recurrent neural network.

4.1.6 Types of neural networks according to nutrition

4.1.6.1 Feedforward Network

Feedforward network is a network where its' data move in one direction only from the inputs to the hidden nodes, with output of no feedback paths. In this architecture, the lowest path is the input layer; highest path is the output layer. The output of any layer is only allowed to proceed to a higher path and any input should only be from a lower path. Examples of a feedforward network can be found from Figure 4.3[29].

4.1.6.2 Feedback Network

Unlike feedforward networks, a feedback network allows output to be input to preceding layers. It is a process whereby the modifications to the weights are controlled by the activity of some of the products of the outputs. This type of network is also known as the recurrent network as shown in Figure 4.5 [29].

X1 Y1

X2 Y2

X3 Y3

.**Figure 4.5:** illustrates feedback network

4.1.6.3 Comparison between Feedforward and Feedback Network

The processing speed of feedforward network is faster than feedback as data flows only in one direction. Feedforward networks are guaranteed to reach stability, whereas the feedback networks must iterate over many cycles until the system stabilized, without guarantee that the system to be stable.

However, feedback networks can perform many functions, which other networks cannot, such as automatic gain or selecting a maximum in complex systems.

4.1.7 Characteristics OF ANNs

In the recall process, an ANN accepts a signal presented at the input buffer, then produces at the output buffer a response that has been determined by the "training" of the network. The simplest form of recall occurs when there are no feedback connections between layers or within a layer (i.e., the signals flow from the input buffer to the output buffer in a process called "feed forward" information flow). In this type of network the response is produced in one computer cycle. When ANNs do have feedback connections, the signal reverberates around the network, across or within layers, until some convergence criteria have been met and a steady-state signal can be presented to the output buffers.

The characteristics that make ANN systems different from traditional computing and artificial intelligence are: 1) learning by example, 2) distributed associative memory, 3) fault tolerance, and 4) pattern recognition.

4.1.8 Learning Methods

There are a lot of learning methods for neural network. These methods will adjust the network weights to let it gradually converge to a value such that the input is the desired output. Learning methods can be broadly classified into three basic types supervised, unsupervised, and reinforced [29].

4.1.8.1 Supervised Learning

Which incorporates an external teacher, so that each output unit is told what its desired response to input signals ought to be. During the learning process global information may be required. Paradigms of supervised learning include error-correction learning, reinforcement learning and stochastic learning.

An important issue conserving supervised learning is the problem of errorconvergence, ie the minimization of error between the desired and computed unit values. The aim is to determine a set of weights which minimizes the error. One well-known method, which is common to many learning paradigms, is the least mean square (LMS) convergence [29].

4.1.8.2 Unsupervised Learning

Uses no external teacher and is based upon only local information. It is also referred to as self-organization, in the sense that it self-organizes data presented to the network and detects their emergent collective properties. Paradigms of unsupervised learning are Hebbian learning and competitive learning.

Ano2.2 From Human Neurons to Artificial Neuronesther aspect of learning concerns the distinction or not of a separate phase, during which the network is trained, and a subsequent operation phase. We say that a neural network learns off-line if the learning phase and the operation phase are distinct. A neural network learns on-line if it learns and operates at the same time. Usually, supervised learning is performed off-line, whereas unsupervised learning is performed on-line [29].

4.1.8.3 Reinforced learning

In this method, a teacher though available, dose not present the expected answer but only indicates if the computed output is correct or incorrect. The information provided helps the network in its learning process. A reward is given for a correct answer computed and penalty for a wrong answer. But, reinforced learning is not one of popular forms of learning [29].

5.1.9 Applications of Neural Networks

Neural networks are applicable in virtually every situation in which a relationship between the predictor variables (independents, inputs) and predicted variables (dependents, outputs) exists, even when that relationship is very complex and not easy to articulate in the usual terms of "correlations" or "differences between groups." A few representative examples of problems to which neural network analysis has been applied successfully are:

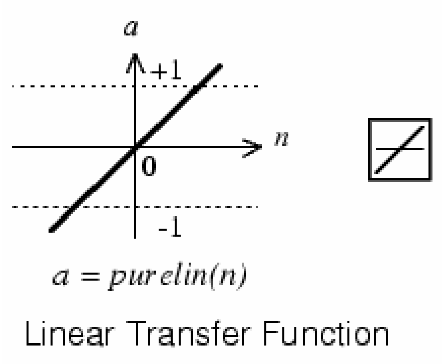
- **Detection of medical phenomena.** A variety of health-related indices (e.g., a combination of heart rate, levels of various substances in the blood, respiration rate) can be monitored. The onset of a particular medical condition could be associated with a very complex (e.g., nonlinear and interactive) combination of changes on a subset of the variables being monitored. Neural networks have been used to recognize this predictive pattern so that the appropriate treatment can be prescribed [34].

- **Stock market prediction.** Fluctuations of stock prices and stock indices are another example of a complex, multidimensional, but in some circumstances at least partially-deterministic phenomenon. Neural networks are being used by many technical analysts to make predictions about stock prices based upon a large number of factors such as past performance of other stocks and various economic indicators [34].
- **Credit assignment.** A variety of pieces of information are usually known about an applicant for a loan. For instance, the applicant's age, education, occupation, and many other facts may be available. After training a neural network on historical data, neural network analysis can identify the most relevant characteristics and use those to classify applicants as good or bad credit risks [34].
- **Monitoring the condition of machinery.** Neural networks can be instrumental in cutting costs by bringing additional expertise to scheduling the preventive maintenance of machines. A neural network can be trained to distinguish between the sounds a machine makes when it is running normally ("false alarms") versus when it is on the verge of a problem. After this training period, the expertise of the network can be used to warn a technician of an upcoming breakdown, before it occurs and causes costly unforeseen "downtime." [34].
- **Engine management.** Neural networks have been used to analyze the input of sensors from an engine. The neural network controls the various parameters within which the engine functions, in order to achieve a particular goal, such as minimizing fuel consumption [34].

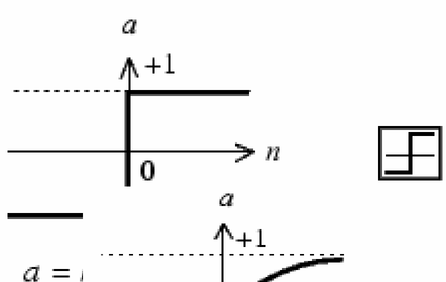
5.1.10 Transfer Function

The behavior of an ANN (Artificial Neural Network) depends on both the weights and the input-output function (transfer function) that is specified for the units. This function typically falls into one of three categories: [29]

1- Linear (or ramp)



2- Threshold



3- tan-Sigmoid

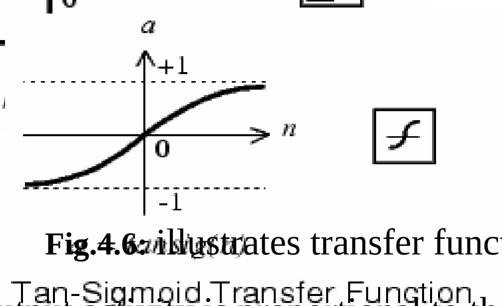


Fig.4.6: illustrates transfer function

- Linear units, the output activity is proportional to the total weighted output.
- threshold units, the output is set at one of two levels, depending on whether the total input is greater than or less than some threshold value.
- Sigmoid units, the output varies continuously but not linearly as the input changes. Sigmoid units bear a greater resemblance to real neurons than do linear or threshold units, but all three must be considered rough approximations.

To make a neural network that performs some specific task, we must choose how the units are connected to one another (see figure 4.7), and we must set the weights on the connections appropriately. The connections determine whether it is possible for one unit to influence another. The weights specify the strength of the influence.

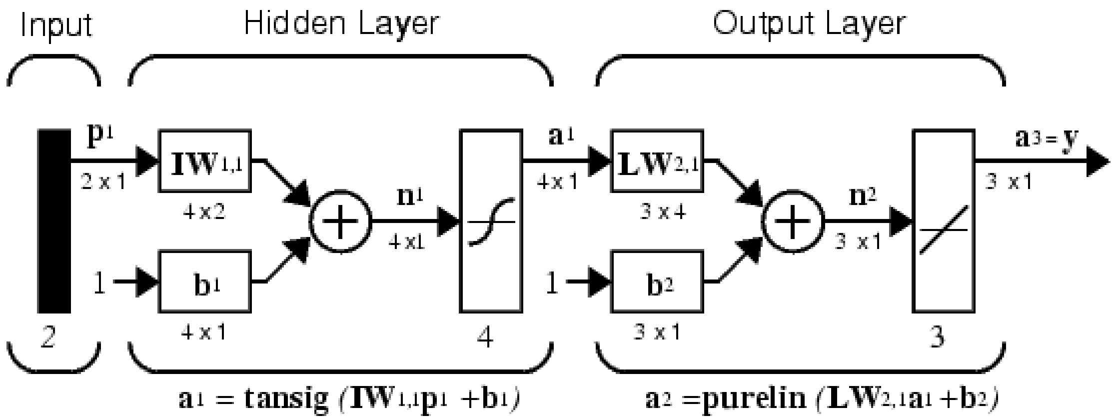


Figure.4.7: the three layer network

We can teach a three-layer network to perform a particular task by using the following procedure:

1. The network is present the network with training examples, which consist of a pattern of activities for the input units together with the desired pattern of activities for the output units.
2. It is determine how closely the actual output of the network matches the desired output.
3. The weight of each connection can be change so that the network produces a better approximation of the desired output.

4.2 The Back-Propagation Neural Network (BPNN)

4.2.1 Introduction

The backpropagation mechanism most often used with these neural networks is based on the gradient descent for learning parameters of these surfaces; therefore we will call this concept “gradient learning of discrimination surfaces.” Backpropagation was designed for supervised training and is not capable of self-learning within the internal neural dynamics. However, the analysis below concentrates on more fundamental limitations of this type of network, limitations related to the nominalistic concept in general.

A standard neuron is shown to partition the classification space with a hyperplane into two halves and, therefore, to be capable of representing a linear classifier. The multineuron, single-layer network is capable of representing multiple linear classifiers, and a nonlinear transformation of the neuronal output does not modify the nature of this neural network, remaining a collection of linear classifiers [35].

4.2.2 Back Propagation Operations

Back propagated networks, start as a network of nodes arranged in three layers--the input, hidden, and output layers. The input and output layers serve as nodes to buffer input and output for the model, respectively, and the hidden layer serves to provide a means for input relations to be represented in the output. Before any data has been run through the network, the weights for the nodes are random, which has the effect of making the network much like a newborn's brain--developed but without knowledge [36].

When presented with an input pattern, each input node takes the value of the corresponding attribute in the input pattern. These values are then transported, at which time each node in the hidden layer multiplies each attribute value by a weight and adds them together. If this is above the node's threshold value, it “fires” a value of 1; otherwise it “fires” a value of 0. The same process is repeated in the output layer with the values from the hidden layer, and if the threshold value is exceeded, the input pattern is given the classification. When training the network, once a classification has been given, it is compared to the actual classification. This is then back propagated through the network, which causes the hidden and output layer nodes to adjust their weights in response to any error in classification, if it occurs. The modification of the weights is done according to the slope of the error curve, which points in the direction to the local minimum near the instance. Unfortunately, the local minimum is not always the global minimum, which causes the network to settle in a non-optimal configuration. The network can sometimes be deterred from settling in local minima by increasing or decreasing the number of hidden layer nodes or even by rerunning the algorithm [36].

4.2.3 Backpropagation Mathematical Algorithm

$$\boldsymbol{x}_p = (x_{p1}, x_{p2}, \cdots, x_{pN})$$

- Consider input vector

Hidden layer:

Net input to the *j*th hidden unit

$$net_{pj}^h = \sum_{i=1}^{N} w_{ji}^h x_{pi} + \theta_j^h$$

hidden layer

*p*th input vector unit, *j*th hidden unit, *i*th input with *j*th unit.

Output of the *j*th hidden unit

$$i_{pj} = f_j^h(net_{pj}^h)$$

Transfer function

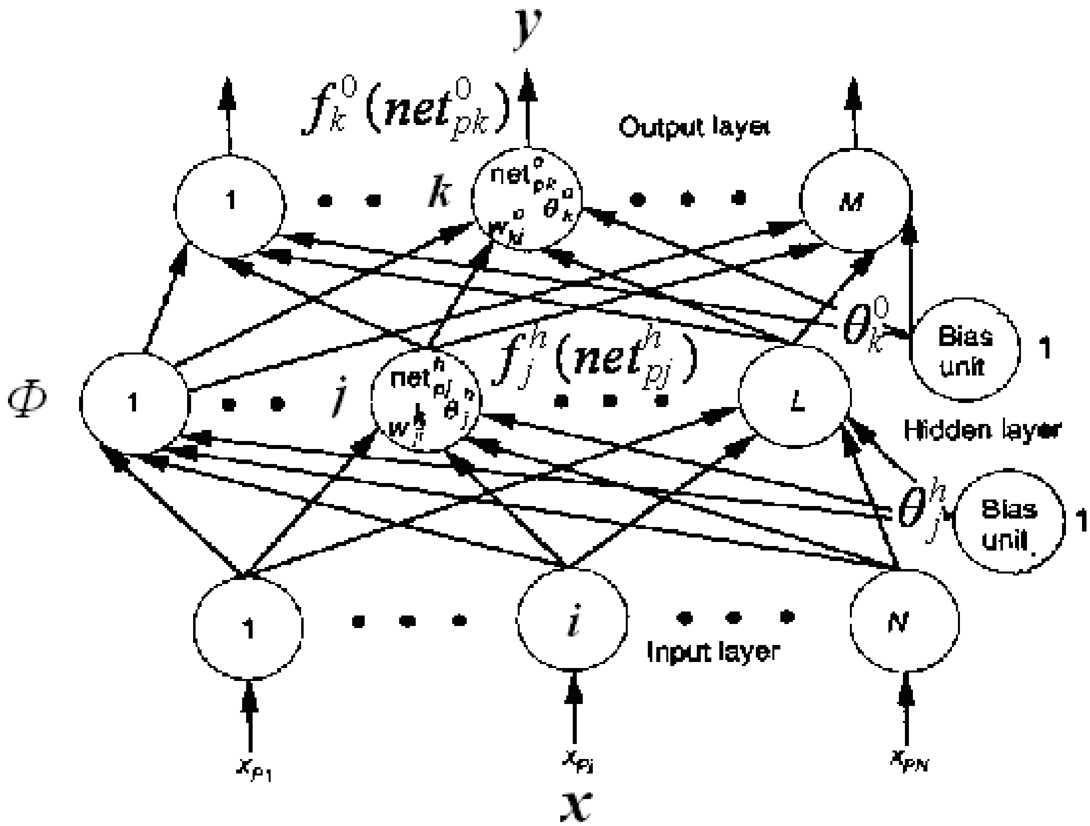


Fig.4.8: Multilayer feedforward backpropagation network.

Output layer:

$$net^0_{pk} = \sum_{j=1}^{L} w^0_{kj} i_{pj} + \theta^0_k$$

Input

$$o_{pk} = f^0_k(net^0_{pk})$$

Output

- Update of output layer weights

The error at a single output unit k

$$\delta_{pk} = (y_{pk} - o_{pk})$$

The error to be minimized

$$E_p(\boldsymbol{w}^o) = \frac{1}{2}\sum_{k=1}^{M} \delta^2_{pk}$$

, M: # output units

The descent direction $-\nabla E_p$

The learning rule

$$\mathbf{w}^o(t+1) = \mathbf{w}^o(t) + \triangle \mathbf{w}^o = \mathbf{w}^o(t) - \eta \nabla E_p$$

Where η : learning rate

Determine ∇E_p

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

∵

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

∵ $o_{pk} = f_k^o(net_{pk}^o)$,

$$\frac{\partial o_{pk}}{\partial w_{kj}^o} = \frac{\partial f_k^o}{\partial (net_{pk}^o)} \frac{\partial (net_{pk}^0)}{\partial w_{kj}^o}$$

∴

∵ $net_{pk}^o = \sum_{j=1}^{L} w_{kj}^o i_{pj} + \theta_j^o$, L: # hidden units

$$\frac{\partial (net_{pk}^0)}{\partial w_{kj}^o} = \frac{\partial}{\partial w_{kj}^o}\left(\sum_{j=1}^{L} w_{kj}^o i_{pj} + \theta_j^o\right) = i_{pj}$$

∴

$$-\frac{\partial E_p}{\partial w_{kj}^o} = (y_{pk} - o_{pk}) f_k'^o i_{pj} = -\nabla_{w_{kj}^o} E_p$$

∴

The weights on the output layer are updated as

$$w_{kj}^{o}(t+1) = w_{kj}^{o}(t) - \eta \nabla_{w_{kj}^{o}} E_p$$

$$= w_{kj}^{o}(t) + \eta (y_{pk} - o_{pk}) f_k'^{o} i_{pj} \quad \text{--- (A)}$$

$$f_k'^{o}(net_{pk}^{o})$$

。Consider

Two forms for the output functions f_k^o

i, Linear $f_k^o(x) = x$

ii, Sigmoid $f_k^o(x) = (1+e^{-\lambda x})^{-1}$

or $f_k^o(x) = \frac{1}{2}[1 - \tanh(\lambda x)]$

For linear function, $f_k'^{o} = 1$,

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

(A)

Let $\lambda = 1$. (A) $\Rightarrow$

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

4.2.4 Advantage and Disadvantage of BPNN

1. Advantages:

1. Good for prediction and classification.

2. Learn a tremendous variety of pattern mapping prior knowledge of a mathematical function that maps the input patterns to the output patterns.

3. Flexible in determining the number of layers, interconnections, processing units and learning constant [37].

2. Disadvantages

1. Slow compared with other algorithm as it has slow learning speed.

2. Training of the network requires hundreds or thousands of iteration which may take hours to complete

3. Local minimum not always the global minimum, due to the premature saturation, the error of the neural network remains significantly high for some period of time during training [36].

6.2.5 BPNN Network Architecture

The number of inputs and outputs is determined by your problem definition and your knowledge representation decisions. An additional parameter is the number of hidden units. It is a common practice to keep this number small in order to improve the neural network's ability to generalize. However, if you select a number that is too small, the network may not converge on the training set, or training may take a very long time. Setting this parameter is part of the art of trying to specify and use back propagation networks in applications.

The number of hidden layers is usually set to one. In some cases, you may not need any hidden layers, resulting in improved training time. If the function you are trying to learn is very complex, you may have to use two or three hidden layers, with a corresponding degradation in training speed [43].

Input Layer - Neurons in this layer receive inputs from external source and passes it to the system for processing.

Hidden Layer – Neurons in this layer processes the data sent to them from the input layer and passes the processed data to the output.

Output Layer – Receive processed data from the hidden layer and outputs it to the system.

The following addition parameters relate to a back propagation network:

4.2.5.1 Activation Functions

The activation function can be any function. A linear activation function essentially results in a neural network capable of generalized linear regression. Non-linear activation functions introduce non-linearity into the network, resulting in a key feature of ANNs; approximation of non-linear functions. Additionally, differentiable activation functions are required since weight adjustments made during training are determined using gradient descent techniques [38].

4.2.5.2 Error Functions (MSE)

The error function is a measure of a network's predictive accuracy for a particular dataset. All error functions are based on the error of the prediction, i.e. the differences between the actual output values and the predicted output values of thedataset. Common error functions include the root mean square of the prediction error, the mean absolute (MA) error and the root relative squared error [38].

4.2.5.3 Learning Rate (lr)

Learning rate is the control parameter of some training algorithms which controls the step size when weights are changed. The learning rate affects the time taken for a network to learn. The greater the learning rate, the faster the network will learn. This may also result to an unstable system if the rate is too small. Time taken to learn by network will be too long. Adaptive learning rate is being adopted in this project as it allows the learning rate to be as huge as possible but in a stable state [43].

4.2.5.4 Momentum (mc)

Momentum is used to enhance the speed of the learning process where a portion of the last change is also incorporated as an addictive factor to the current weights. Acting like a low pass filter, momentum allows the network to ignore small features in error surface. Without momentum, a network may get stuck in a shallow local minimum [43].

4.2.6 Network Training

The network is trained to produce the correct steering direction using the back-propagation learning algorithm. In backpropagation, the network is first presented with an input and activation is propagated forward through the network to determine the network's response. The network's response is then compared with the known correct response. If the network's actual response does not match the correct response, the weights between connections in the network are modified slightly to produce a response more closely matching the correct response. Autonomous driving has the potential to be an ideal domain for a supervised learning algorithm like backpropagation since there is a readily available teaching signal or "correct response" in the form of the human driver's current steering direction. In theory it should be possible to teach a network to imitate a personas they drive using the current image as input and the person's current steering direction as the desired output. Training on real images would dramatically reduce the human effort required to develop networks for new situations, by eliminating the need for a hand- programmed training example generator. On-the-fly training should also allow the system to adapt quickly to new situations.[39].

The following is Train function with matlab program:

```
[net,tr]=train(net, inputdata, outdata);
```

Input training data = inputdata

Output data = outdata

Function train( ): Once the network weights and biases are initialized, the network is ready for training. The network can be trained for function approximation (nonlinear regression), pattern association, or pattern classification.

4.2.7 Network Simulation

Function sim( ) The `sim` command causes the specified Simulink model to be executed. The model is executed with the data passed to the sim command, which may include parameter values specified in an options structure. The values in the structure override the values shown for block diagram parameters in the Configuration Parameters dialog box, and the structure may set additional parameters that are not otherwise available (such as DstWorkSpace). The parameters in an options structure are useful for setting conditions for a specific simulation run.

Anewn = sim(modeldata, inputdata);

test_error=length(find((round(Anewn)- outdata)~=0))/length(outdata);

test_error = (test_error/2)*100;

input testdataset= inputdata

Output data = outdata

Network = modeldata

Confusion matrix 4.2.8

A confusion matrix contains information about actual and predicted classifications done by a classification system.

Overfitting 4.2.9

When facing a classification problem, we need to define a model that fits our pattern and defines its nature well. If the model doesn't make a good characterization of our pattern, the classification will not be accurate. On the other hand, a high complex model will give us a perfect classification for this one pattern, but if we try to use this model to classify other pattern of the same problem, we will probably obtain a not so good classification. This is called overfitting. The overfitting occurs when a model for a given pattern has been defined so tightly that, although for this one pattern a perfect classification is obtained, when applied to another pattern of the same type, it is unlikely to obtain a good classification [46]. The importance of this issue has made it one of the most important areas of research in statistical pattern classification, i.e., how to adjust the complexity of the model. A model should be able to achieve the right classification for the given patterns, without being too complex in a way that it performs poorly on novel patterns [46]. The problem of finding the right complexity for a classifier can be defined as the balance between over fitting and pour characterization.

Overfitting Solutions with Neural Network 4.2.10

1. Early Stopping

The default method for improving generalization is called early stopping. This technique is automatically provided for all of the supervised network creation functions, including the backpropagation network creation functions such as newff.

In this technique the available data is divided into three subsets. The first subset is the training set, which is used for computing the gradient and updating the network weights and biases. The second subset is the validation set. The error on the validation set is monitored during the training process. The validation error normally decreases during the initial phase of training, as does the training set error. However, when the network begins to overfit the data, the error on the validation set typically begins to rise. When the validation error increases for a specified number of iterations (net.trainParam.max_fail), the training is stopped, and the weights and biases at the minimum of the validation error are returned. The test set error is not used during training, but it is used to compare different models. It is also useful to plot the test set error during the training process. If the error in the test set reaches a minimum at a significantly different iteration number than the validation set error, this might indicate a poor division of the data set.

2. Regularization

Another method for improving generalization is called regularization. This involves modifying the performance function, which is normally chosen to be the sum of squares of the network errors on the training set.

2.1Modified Performance Function(trainbgf)

The typical performance function used for training feedforward neural networks is the mean sum of squares of the network errors.

$$F = mse = \frac{1}{N}\sum_{i=1}^{N}(e_i)^2 = \frac{1}{N}\sum_{i=1}^{N}(t_i - a_i)^2$$

It is possible to improve generalization if you modify the performance function by adding a term that consists of the mean of the sum of squares of the network weights and biases

$$msereg = \gamma mse + (1-\gamma)msw$$

Where γ is the performance ratio, and

$$msw = \frac{1}{n}\sum_{j=1}^{n} w_j^2$$

Using this performance function causes the network to have smaller weights and biases, and these forces the network response to be smoother and less likely to overfit.

2.2 Automated Regularization (trainbr)

It is desirable to determine the optimal regularization parameters in an automated fashion. One approach to this process is the Bayesian framework of David MacKay [MacK92]. In this framework, the weights and biases of the network are assumed to be random variables with specified distributions. The regularization parameters are related to the unknown variances associated with these distributions. You can then estimate these parameters using statistical techniques.

A detailed discussion of Bayesian regularization is beyond the scope of this user guide.

4. Training with Noise

There is a third approach to controlling the trade-off bias against variance, which involves the addition of random noise to the input data during training. This is generally done by adding a random vector onto each input pattern before it is presented to the network, so that, if the patterns are being recycled, a different random vector is added each time.

It is discovered that the addition of training patterns artificially generated by injecting noises to the existed ones improves the performance of generalization[42]

CHAPTER FIVE
DIGITAL IMAGE PROCESSING

CHAPTER FIVE

DIGITAL IMAGE

5.1 Overview for Images

Human beings are predominantly visual creatures; we rely heavily on our vision to make sense of the world around us. We not only look at things to identify and classify them, but we can scan for deference's, and obtain an overall rough "feeling" for a scene with a quick glance.

Humans have evolved very precise visual skills: we can identify a face in an instant; we can differentiate colors, we can process a large amount of visual information very quickly.

However, the world is in constant motion: stare at something for long enough and it will change in some way. Even a large solid structure, like a building or a mountain, will change its appearance depending on the time of day (day or night); amount of sunlight (clear or cloudy), or various shadows falling upon it [40].

The term image is refer to an picture defined as two dimensional and as a finite set of digital values called picture elements or pixels each of which has a particular location and value. A single pixel represents a value of either light intensity or color. Images are processed to obtain information beyond what is apparent given the image's initial pixel values.

5.2 Digital image processing

Image processing is processing image information be using a computer to improve the quality of image by extracting some features that describe the relevant aspects of the images, unlike the original images, the features can be compared and obtain a good image quality.

Digital image defined as two dimensional function *f (x,y)* where x and *y* are spatial (plane) coordinated, and the amplitude of *f* at any pair of coordinates *(x,y)* is called the intensity of the image at that point (figure 5.1). The term gray level is used often to refer to the intensity of monochrome images. Color image formed by combination of individual two dimensional images. For example in the RGB color system a color image consists of three (red, green, blue) individual component images (figure 5.1).

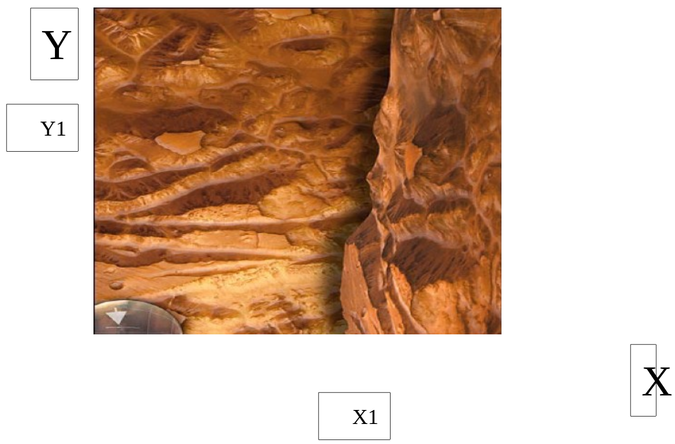


Figure 5.1: digital image

An image changed to digital form to store in computer memory or in an other media. The process of change image to digital form done with many devices such as scanner, digital camera and handycam. When change image to digital form there are many techniques applying to processing image.

5. 2.1 Digital Image Processing Techniques

1. Image acquisition: is the first process of Digital Image Processing and involves preprocessing such as scaling.

2. Image enhancement: a process to improve quality of given image.

Is among the simplest and most appealing areas of Digital Image Processing. The idea behind enhancement techniques is to bring out details that are obscured, or to simply highlight certain features of interest in the image.

Blue B(x,y) Green G(x,y) Red $R(x,y)$

Figure.5.2: Images Color Components

3. Image segmentation: a process of division or subdividing an object in image

4. Image restoration: a process to improve a given image.

Is an area that deals with improvement of appearance of image of the system. The restoration techniques can be based on mathematical or probabilistic models of image degradation.

5. Color image processing: a process relating to color an image.

6. Image compression: a process of compression image data.

As the name implies, deals with the techniques for reducing the storage required saving an image, or the bandwidth required to transmit it.

5.2.2 Types of Digital Images

There are some basic types of images.

5.2.2.1 Indexed Image

An indexed image consists of a data matrix, X, and a colormap matrix, map. The colormap is an *m*-by-3 array containing floating-point values in the range [0, 1]. Each row of map specifies the red, green, and blue components of a single color. The color of each image pixel is determined by using the corresponding value of X as an index into map. The value 1 points to the first row in map, the value 2 points to the second row [41].

5.2.2.2 Binary Images

Each pixel is just black or white. Since there are only two possible values for each pixel, we only need one bit per pixel. Such images can therefore be very efficient in terms of storage. Images for which a binary representation may be suitable include text (printed or handwriting), fingerprints, or architectural plans [44].

5.2.2.3 Grayscale Images

Each pixel is a shade of grey, normally from (black) to (white). This range means that each pixel can be represented by eight bits, or exactly one byte. This is a very natural range for image file handling. Other grayscale ranges are used, but generally they are a power of 2. Such images arise in medicine (X-rays), images of printed works, and indeed 256 different grey levels is sufficient for the recognition of most natural objects [44].

5.2.2.4 True color Images

A true color image, sometimes called an RGB image, is an *m*-by-*n*-by-3 data array that defines red, green, and blue color components for each individual pixel. Each color component is a value between 0 and 1. A pixel whose color components are (0, 0, 0) displays as black, and a pixel whose color components are (1, 1, 1) displays as white. The three color components for each pixel are stored along the third dimension of the data array. For example, if RGB is a true color image, then the red, green, and blue color components of the pixel (10,5) are stored in RGB(10,5,1), RGB(10,5,2), and RGB(10,5,3),respectively[41].

CHAPTER SIX
EXPERIMENTS, RESULTS AND DISCUSSIONS

CHAPTER SIX
EXPERIMENTS, RESULTS AND DISCUSSIONS

6.1 Introduction

The first main objective of this thesis was to design and train a back-propagation neural network to recognize isolated handwritten Arabic letters use SUST-ARG dataset.

The first set of experiments showed high confusion between similar letters this give the imposition that a classifier that classify letter into group of similar letters, followed by a set specialized classifiers. Each specialized classifier recognizes letters in a group of similar letters. This idea was tested in second round of experiments. The accuracy of the second round is much better than the first one. However, in this round the difference of accuracy rate between testing dataset and training dataset is very big, which indicates that the classifier is suffering from overfitting problem.

In third experiments, are applied many methods to address over-fitting problems. Figure .6.1 explains these experiments stages.

This chapter is organizes as followings, section 1, describes SUST-ARG dataset and the techniques used to reduce the dimensionality of the dataset. Section 2, describes the first experiments, which conducted on dataset of 34 classes. Section 2, describes the second experiments conducted on grouped dataset. Section 4, describes the experiments conducted to solve the overfitting problem.

First experiments training a BPNN .Arabic character using 34 classes

Experiments result: high error .between similar letters

Suggest: 2 stage classifier (group (and character

Second experiments training a BPNN Arabic character using 15 .classes

Experiments result: 1. better accuracy, 2. big different between training and testing

Suggest: overfitting solution

Experiments with three techniques for solving overfitting problem

Figure.6.1 illustrates the dataset training processes.

6.2 SUST-ARG

6.2.1 Overview

SUST-ARG stands Sudan University for Sciences and Technology Arabic Recognition Group (**SUST-ARG**).

This research group has designed collected and prepared many dataset for developing recognizers of the Arabic language. One of these data sets is SUST-ARG letters, which contains isolated Arabic characters. The following subsection describes this dataset building stages.

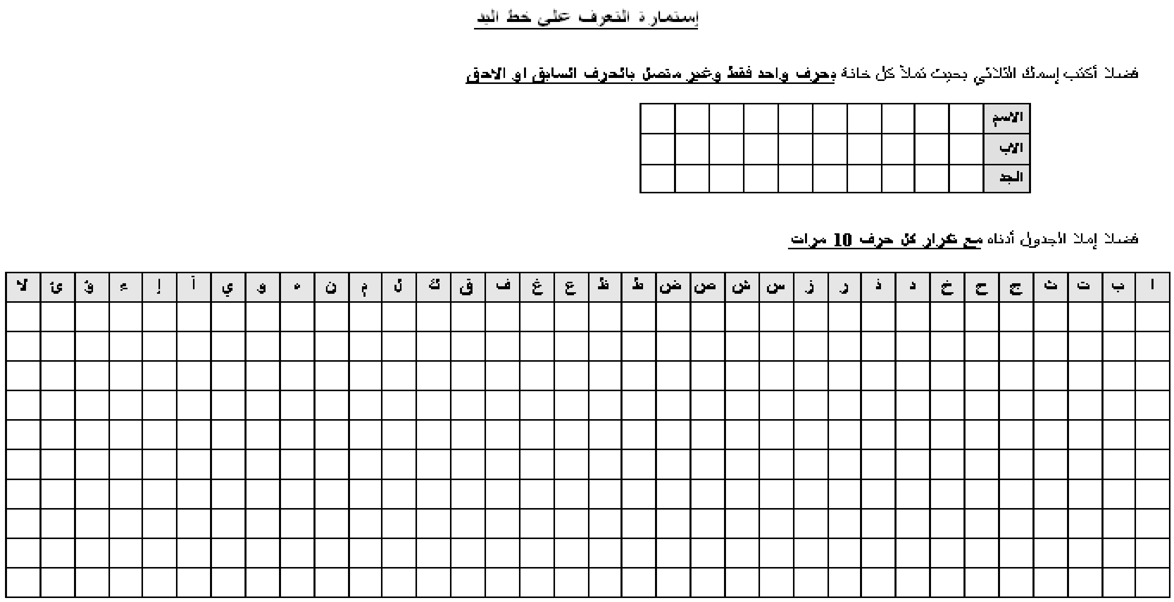
6.2.2 Collection and Preparation

The form as shown in Figure.6.2 has been designed to collect the required handwritten letters.

One hundred and forty one forms have filled by different subjects. These forms scanned by scanner device accuracy of 300 dpi, saved as color images, Figure.6.3 shows letters extraction process. This process extracts each specific letter from all form and put it in a separate folder as grayscale image, any letter composed from 1410 images written by hand.

The characters prepared as is shown below in figure.6.4 are scanned using a scanner and these characters will be segregated according to their own character group.

A set of Arabic handwritten characters was selected concentrating on the basic Arabic letters, which consist of 28 characters and the related characters as (أ، إ ئ ء، لا، ؤ).



إستمارة التعرف على خط اليد

فضلا أكتب إسمك الثلاثي بحيث تملأ كل خانة بحرف واحد فقط وغير متصل بالحرف السابق او اللاحق

الاسم										
الاب										
الجد										

فضلا إملا الجدول أدناه مع تكرار كل حرف 10 مرات

ا	ب	ت	ث	ج	ح	خ	د	ذ	ر	ز	س	ش	ص	ض	ط	ظ	ع	غ	ف	ق	ك	ل	م	ن	ه	و	ي	آ	إ	ء	ؤ	ئ	لا

Figure. 6.2: illustrates alphabet Arabic letters document

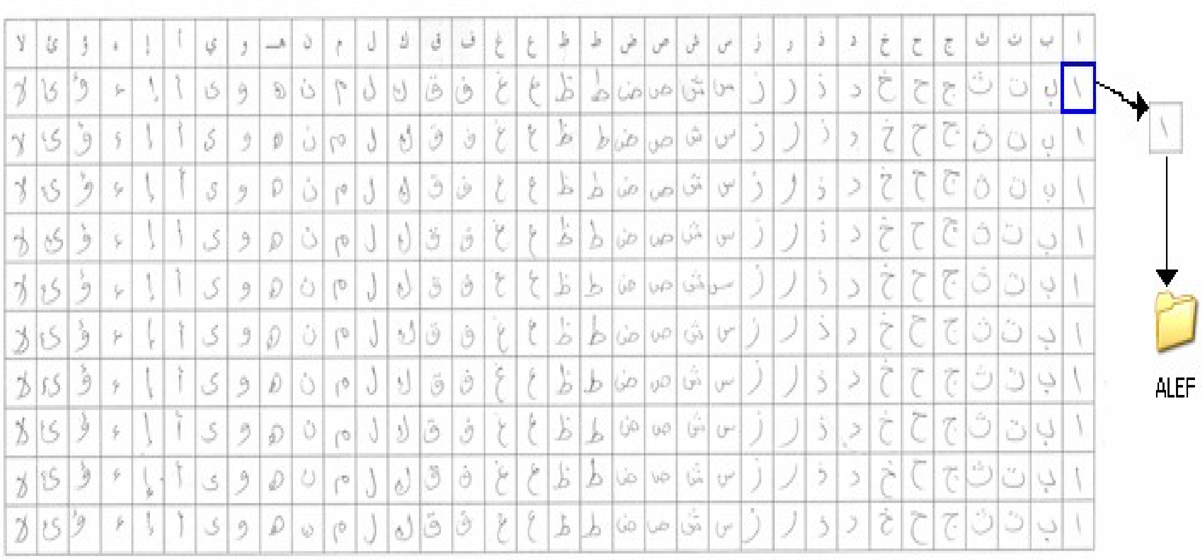


Figure. 6.3: illustrates letters extraction

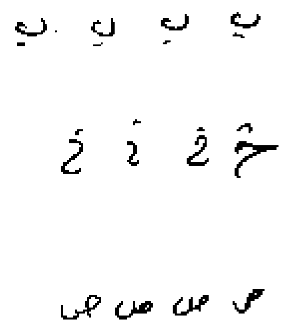


Figure.6.4 illustrates Sample of character (ba), (kha) and (sad).

Figures.6.5 and figure 6.6 illustrate the different processes carried to prepare the dataset to be use for training and testing the classifier.

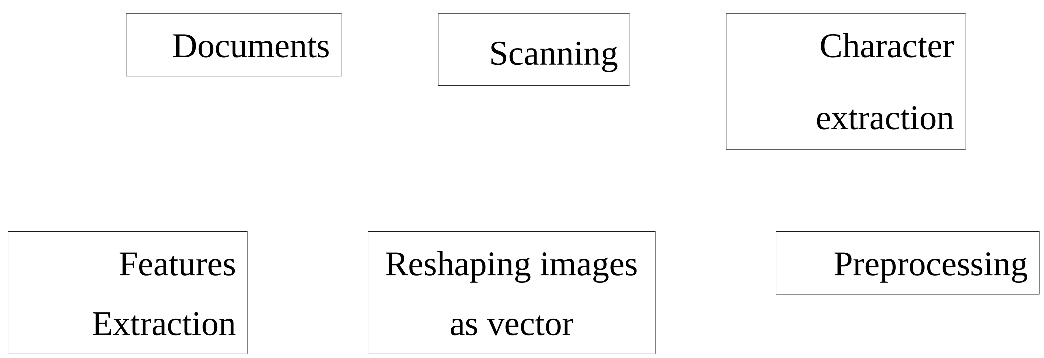


Figure.6.5: illustrates implementation thesis stages.

6.2.3 Preprocessing Operations

It is primary to be held at the images until it is purified of the impurities and to adjust its size for each sample one-size-fits, which attempt to improve the performance of the eigen method of letter recognition. The preprocessing techniques which were used it is resized, noise removal, median filter and grayscale as shown in figure.6.7.

6.2.3.1 Resize

The purpose of this technique is to put images in one size, which is 30x30 pixels matrix.

Designed document with table contains 34 arabic letter and given it to student to fill the table with Arabic letters by the hand

Scanned handwritten document on scanner with (300 dpi) to create image document (as bmp file

Cutting each of letter and store it in separated image file with .matlab program

Preprocess the images by resize it in the range 33x33 pixel, mediate filter, removals noise and grayscale

Converts the image to image vectors and saved as matlab file contains matrices represent the .dataset of letters images

Extract features from train and test dataset file using principle common analysis (PCA) and save the new matrices .in data file

Training datafile feature matrix

Testing datafile Feature Matrix

Figure.6.6: illustrates the stages details of Handwritten Arabic Letters Recognition system.

Input dataset as bitmap image

Preprocessing stages

Preprocessed images

Grayscales

Resized image

Noise removal

Figure. 6.7. illustrates the stages of the preprocessing.

Figure.6.8: illustrates the adjust letter image size to 30x30

6.2.3.2. Grayscale

The objective of this process to converts images to grayscale by eliminating the hue and saturation information while retaining the luminance.

6.2.3.4 Noise Removal

The major objective of noise removal is to remove any unwanted bit-patterns, which do not have any significance in the output. The techniques used namely:

- Median filtering

Is a nonlinear operation often used in image processing to reduce "salt and pepper" noise. A median filter is more effective than convolution when the goal is to simultaneously reduce noise and preserve edges.

- Adjusting Image Intensity

This step increases contrast by adjusting image intensity values. It suppresses eventually outlier's pixels to improve contrast stretching. It then stretches contrast to use the whole range of intensity channel and if necessary it saturates some low or high values [1]. Figure.6.9 shows image adjusting.

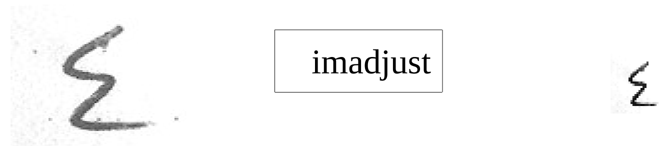


Figure.6.9 illustrates the adjust letter image intensity.

6.2.4 Converting Images to Vectors

In this process converted the image into a set of vectors describing each foreground pixel that makes up the character and collected all images vectors in a large matrix.

6.2.5 Features Extraction

This process extracts the features of the characters that are most relevant for classifying at recognition stage. This is an important stage as it can help avoid misclassification, thus increasing recognition rate.

Extracted features from the images matrix to reduce the matrix dimension from (900) to (350, 150, 100) using the classical implementation of PCA with matlab software.

6.3 First Experiments

The proposed network topology is shown in figure.6.10. Many experiments have been conducted to find best values for the number input units (M) and hidden layer units (H). The result of these experiments is shown in table

(6.1). The highest recognition rate for testing dataset is 54.30% occurred with 350 features as inputs, 100 units in hidden layer and 2500 epoch as shown in table (6.1). The confusion matrix as shown in table (6.2), reflect that fact that the error occurred between similar-shaped letters. For example, to the letter “ت” out of 200 samples only 53 samples are correctly recognized. About many of number of characters out of 147 were recognized as other characters, as shown in table (6.3). 26% of the character “د”, were recognized as “ء”, which is another indicative instance for this similar shaped letters confusion.

We conclude that the low recognition for the letter “ت” is due to strong similarity with character “ث” and character “ن”, And more reliability rate in a letter “ ا “ for the clear difference between it and all the characters except for some near writers in the drawing with the letter “أ”.

Input layer Hidden layer Output layer

M H O

Figure.6.10. Network topology. M= 50, 100 or 350, H=10 - 150, O= 34.

Table (6.3).illustrates Ta letter confusion.

character	ت	ن	ث	ش	ئ
ت	53	19	48	11	10

Table (6.1). Network Performance for Different Hidden Layer nodes for 34 classes refers to the Recognition Rate.

Recognition rate		time	Hidden units	Epoch	features
testing	training				
29	35.43	5,30	10	500	350
41.48	59.40	9,42	40		
43.42	68.13	14,19	70		
36.50	55.23	19,20	100		
24.81	27.49	26,1	150		
35	44.87	18,6	40	1500	
37.17	50.73	43,5	70		
36	46	55,43	100		
41.33	67.03	1,11,38	70	2500	
54.30	**83.87**	**1,32,9**	**100**		
36.06	44.09	27,52	70	2500	100
51.33	77	59,27	100		
30.11	30	15,23	40	2500	50

37	41.89	24,46	70
51.36	68.85	54,49	100

6.4. Second Experiments

In this second set of experiments we have grouped all similar characters in sub class, as shown in table (6.4). The same network topology was only the out put layer was changed to eleven units. We concluded more number of groups may perform better. For these reason we regrouped the letters in fifteen groups as shown in table (6.4).

Two stages of classifier are proposed as shown in Figure.6.11. It is a multistage classifier that has tow stages. The first stage is based on features extracted from groups of similar characters. The last stage has sub-classifiers. Sub-classifier is a multiple neural network (BPNN) classifier to recognize character of only one group.

إ أ ا

Second stage Group-1 Group-2 Group-n

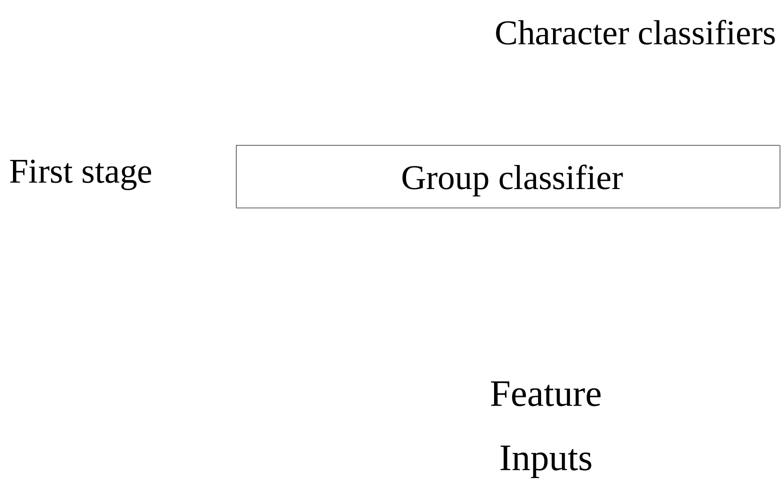


Figure.6.11.illustrates multi- classifiers system

6.4.1 Group Classifier

Some Arabic letters which similar in its shape, when written by hand are similar in dots and rings as shown in figure (6.12). The system accepts the features of the letter that we need to recognize and classify it to its group, at this stage, were conducted many experiments are applied in two dataset, which are different in number of classes; namely:

Table (6.4) the classes of grouped characters.

Eleven Character Group	Fifteen Character Group
ا إ أ	ا إ أ
ب ت ث ن ش	ب ت ث ن
ج ح خ م	ج ح خ
د ذ ر ز ء	د ذ ر ز ء
س ص ض	س ص ض
ط ظ	ش
ع غ	ط ظ
ف ق ؤ	ع غ
لا ل ک	ف ق ؤ

و هـ
ي ئ

ل ك
م
هـ
و
ي ئ
لا

A. **dataset with 11 classes**

Were gathered together the letters that has similar feature in eleven groups as shown in table (6.3). And conducted many experiments with different features and hidden layer units and trained again, the result of experiments shown in table (6.5). The highest recognition rate is **71.73%** for testing dataset.

Taa Tha letter etter of Na letter Ya hamza letter

Figure.6.12: example similarly characters.

Table (6.5): Network Performance for Different Hidden Layer nodes for 11 classes refers to the Recognition Rate.

Feature	Hidden units	Time Minutes	Train accuracy %	Test accuracy %
50	70	14,3	87.83	61.23

	100	18,11	91.56	63.5
	200	35,26	97.02	67
100	100	22,29	94.76	62.37
	200	40,38	98.08	68.28
	600	**2,13,1**	**89.77**	**71.73**
350	70	28,34	93.52	59.73
	100	39,39	95.10	62.55
	200	**1,12,37**	**94.93**	**64.64**

B. Dataset with 15 classes

In the confusion matrix as shown in table (6.6), for the experiment which is highest recognition rate for testing dataset as shown in table (6.5). Were found some letter groups has low rate of classifying percent. Example the letters of the second group classified 17 times as eighth group and 17 times read as ninth group, the letter that lead to similar between groups was letter " ش", so were set it in separate class, and other groups are applied this condition. Then the new grouped dataset are trained again with 15 class contains letters grouped and other letters in separate classes. Many experiments were conducted upon it with various different number of features set as shown in table (6.7). The highest recognition rate is **78.77%** for testing dataset of fifteen groups of character as shown in table (6.7). Figure.6.14 explains the experiment performance, which highest recognition rate.

The output resulted in this stage is not final output, but it fed as input pattern, to the final stage.

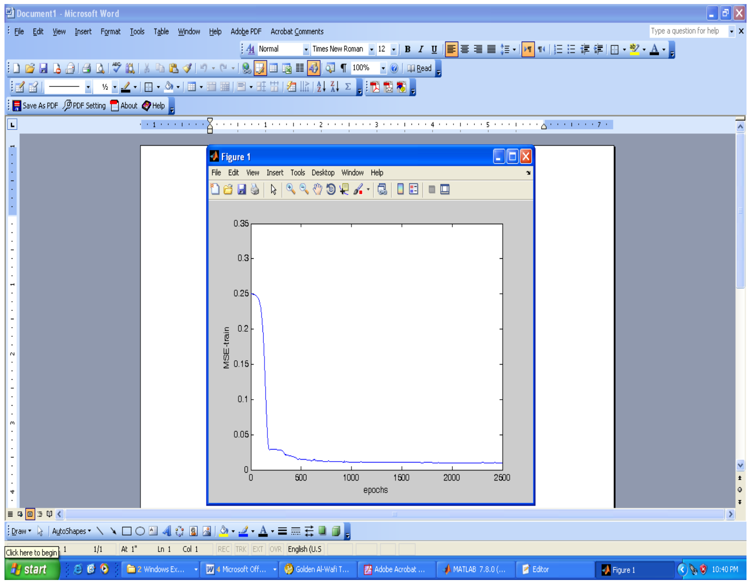


Figure.6.13 illustrates MSE with 100 hidden units and 2500 epoch for dataset with 11 classes.

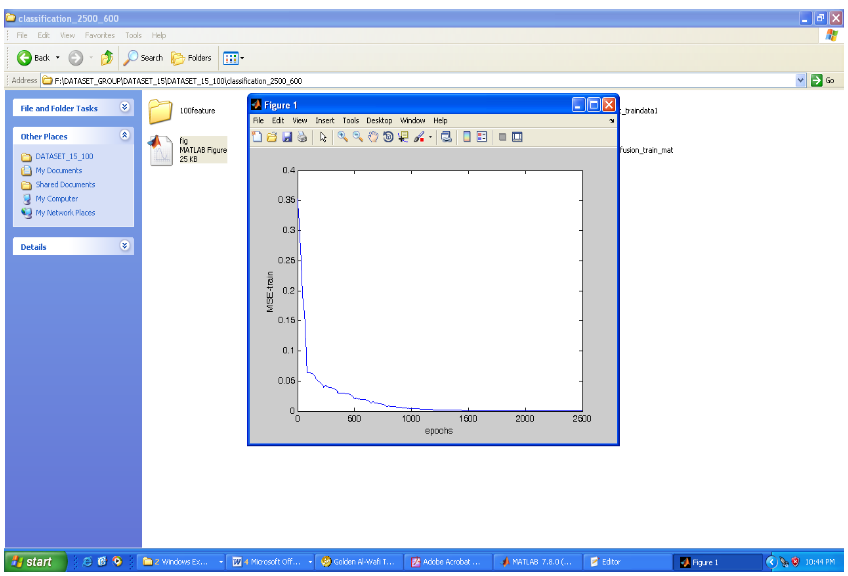


Figure.6.14 illustrates MSE with 100 hidden units and 2500 epoch for dataset with 15 classes.

.

acc ئ وهـ لا ف ع ط ظ س د ذ ج ب ا إ
ي ل ق ؤ غ ص ز ر ح ث أ

		ث ن ش	خ م	ء	ض				ک			
ا إ أ	172	3	0	3	0	7	9	2	3	0	1	86
ب ت ث ن ش	0	81	9	13	20	5	9	17	17	3	26	40.5
ج ح خ م	7	1	99	5.	12	7	33	7	11	11	7	49.5
د ذ ر ز ء	4	2	2	149	0	9	2	9	3	17	3	74.5
س ص ض	0	1	5	0	174	3	0	5	1	8	3	87
ط ظ	1	2	1	3	6	132	0	22	10	17	6	66
ع غ	5	1	30	2	0	5	149	2	1	2	3	74.5
ف ق ؤ	0	14	2	5	12	11	5	133	5	9	4	66.5
لا ل ک	1	24	7	1	1	8	1	7	129	14	7	64.5
و هـ	0	0	3	9	5	16	1	3	4	157	2	78.5
ئ ي	2	16	5	2	12	4	0	17	10	5	127	63.5

Table (6.6): confusion matrix on testing dataset with 200 hidden nodes and 2500 epoch for **11class.**

6.4.2 Character Classifier

In this classifier after determining that character to its group with group classifier, the character input to the classifier of group and match it with one of its elements. The experiments applied on one selected from group of 15 dataset classes, the letter group are “إ، أ، ا” as shown in table (6.3). The highest recognition rate is **92.77%** to testing dataset as shown in table (6.8). Some groups in the group classifier contain one letter, for that the final output can be taken in the first stage.

Table (6.7): Network Performance for Different Hidden Layer nodes for 15 classes refers to the Recognition Rate.

% Recognition rate		Time	Hidden	Feature
Testing dataset	Training dataset	Minutes	units	
62.50	84.78	19,16	70	50
65.14	90.16	27,20	100	
70.1	96	52,29	200	
52.48	**72.38**	**13,07**	**30**	100
61.60	88.31	23,10	70	
65	**92.51**	**31,3**	**100**	
71.04	97.75	59,5	200	
74.27	98.18	1,23,21	300	
76.64	98.63	1,55,23	400	
77.10	99	2,38,21	500	
78.77	**99.4**	**2,49,39**	**600**	
74	92.47	3,11,17	700	
74.40	98.73	1,44,42	300	150
73.04	92.20	1,57,47	400	
77.04	98.86	2,35,49	500	
77.37	98.94	3,0,10	600	
72.10	92	2,33,10	400	200
77,70	98.68	3,10,02	500	
73.34	98.40	2,36,33	400	250
71.90	92.12	3,13,5	500	

Table(6.8): recognition accuracy testing dataset for 3 classes

Test % accuracy	Train % accuracy	Time Minutes	Hidden units	Feature
91.50	99.84	11,41	200	100
92.77	**99.76**	**32,10**	**600**	
90.5	**99.94**	**21,30**	**200**	350

87.67	98.70	28,40	300	

6.4.3 Discussion

The Experiments have shown that the grouped characters method is an effective technique for improving character recognition accuracy.

The results presented in previous experiments shows that 100, 200 and 600 number of hidden units gives the best performance on testing dataset as shown in table (6.9).

Table (6.9): Illustrates the highest recognition accuracy in all experiments

Testing accuracy %rate	**Training % accuracy rate**	**Hidden nodes**	**Dataset size**
54.34	**83.87**	**100**	(Dataset (34 classes
78.77	**96**	**600**	grouping dataset (classes 15)
92.77	**99.76**	**200**	(One group (3classes

6.5 Overfitting Solution

Table (6.9) shows the best result of the previous experiments, these results shows that there is a big difference between percentage of recognition rate of training dataset and testing dataset by an average 24%, which there is an overfitting problem.

In the literature there are many techniques to remedy overfitting problem. We have tested three of them as shown in figure.6.15, which are reported in the following sections.

(Features (inputs

Overfitting solution algorithms

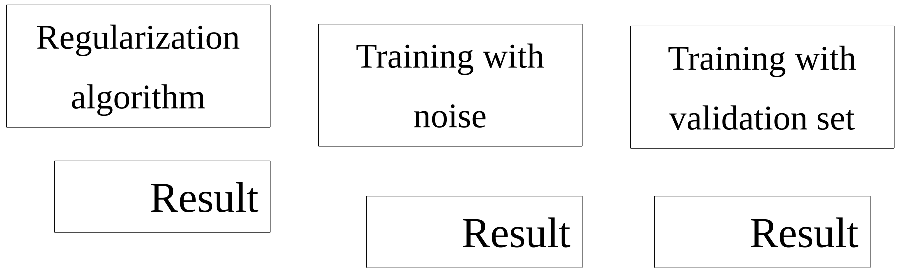


Figure.6.15: Overfitting solution techniques.

6.5.1 Early Stopping of Training

With using validation dataset, when stopped the training, when the operator maximum failure (max_fail) to a specific number of times as shown in the table (6.10). The highest recognition rate for the test dataset is 68% and for the training data equal to 83.16%. We find that the use of this method contributed to solving the problem of over-training compared with previous results, which appeared in table (6.7) for the experiment that used 100 features as inputs and 100 units of the hidden layer.

Table(6.10): recognition accuracy testing dataset with different Max_fail

Recognition rate		time	Max_fail	Epoch	Hidden	Features
testing	training					
64.44	81.52	5,11	146	500	100	100
69	86.83	10,22	673	1000		
68	83.16	27,44	2246	2500		

6.5.2 Network Training with Noise

In this method we added random data to the training dataset and repeat the process of add five times in several experiments with different epoch as shown in the table (6.11).The highest recognition rate for the test dataset is

71.14% and for the training data in the same experiment is equal to 82.54%, we find that the use of this method contributed to solving the problem of over-training compared with previous results, which appeared in table (6.7) for the experiment that used 100 features as inputs and 100 units of the hidden layer.

Table(6.11): recognition accuracy testing dataset with noise

Recognition rate		time	Epoch	hidden	features
testing	training				
70.17	80.83	3,46,11	500	100	100
70.47	82.20	4,36,26	1000		
71.14	82.54	6,45,21	2500		

6.5.3 Regularization with Modified Method

This method is used with the training function (trainbfg). This technique requires more computation in each iteration and more storage, hence has been used of a smaller number of hidden layer units as shown in table (6.12), The highest recognition rate to the test dataset is 60% and to the training data in the same experiment is equal to 62.77%, we find that the use of this method contributed to solving the problem of over-training compared with previous results, which appeared in table (6.7) for the experiment that used 100 features as inputs and 30 units of the hidden layer.

.**Table(6.12):** recognition accuracy testing dataset with regularization

Recognition rate		time	Epoch	hidden	features
testing	training				
41.64	43.24	2,19,15	1659	10	100
48.80	53.44	4,22,53	1003	20	
60	62.77	7,39,13	1481	30	

6.5.4 Discussion

Using validation sets, training with noise and regularization methods, testing results is better by 3%, 6.14% and 8% respectively.

Our results for third experiment as shown in each table (6.10) and table (6.11) show the techniques which has been used alleviate the overfitting. However the result of third experiment in table (6.12), indicate that the recognition is more promised, provided of have request more memory and time.

CHAPTER SEVEN

CONCLUSION AND RECOMMENDATIONS

CHAPTER SEVEN

7. Conclusion and Recommendations

7.1 Conclusion

In this thesis, we have presented a system for recognizing handwritten Isolated Arabic characters. The dataset that is used in these experiments is from isolated Arabic character set. It collected from 141 writers with each person written each letter ten times and scanned with 300 dpi. The total of all characters are (30600 characters) are divided into a training (23800 characters), and testing sets (6800 characters). The grayscale of 900 pixels of each image of characters dataset were used and reduced to different features set size as the inputs. Features were extracted from images using PCA.

In the classification, back-propagation neural network classifier has been used to training and simulating dataset.

The proposed classifier works into two stages, in the first the proposed classifier classify the input character into subgroup. In the second level the classifier classify the character of a given subgroup to specific character. In the first stage, training dataset contains fifteen classes, each class consists of 1000 samples of image. The recognition rate for testing dataset was **78.77%** and **99.4%** for training dataset. In the second level we have tested only one subgroup (أ ,ا and إ), the recognition rate attained by this group was **92.77%.**

The results presented in previous experiments shows that 600 hidden units give the best performance on testing set. As the attained result reflect high degree of overfitting. Three techniques of overfitting solution have been tested to remedy this problem.

7.2 Recommendations and Future Work

The training dataset used in our experiments contains 700 samples for each character, and each character is represented by 100 features. This indicates clearly that increasing the dataset size would help in improving the classifier recognition accuracy.

This research recommends using grid computers to test more neural network model, by increasing number of hidden layer units and epoch.

Training results are excellent, 99.4% for training dataset. However, the difference between training and testing dataset is 20.83%, is very big. Overfitting remedy tests done are limited, more work on overfitting need to be done. Specially finds in more memory for regularization with modified method. Also testing other overfitting method is recommended.

REFERENCES

[1] A. Amin, Arabic character recognition, handbook of character recognition and document image analysis, World Scientific Publishing Company, 1997.

[2] A. Amin. and A. NATO, the state of the art on Arabic character recognition, School on Fundamentals in Handwriting Recognition, Springer-Verlag 1993a, pp. 1-150.

[3] L. M. Al-Zoubaidy. Arabic printed /handwritten characters recognition using ANN, Ph.D thesis, Mosul University, Iraq, 2002.

[4] S. Theodoridis, pattern recognition second edition, acid-free paper.2003.

[5]Editorial, Advances in Pattern Recognition, Pattern Recognition Letters 2005, 26,395-398,.

[6]T. Pavlidis, Structural Pattern Recognition. Springer Verlag, Berlin Heidelberg New York, 1977.

[7] R. Gonzalez,R.C.Thomas,M.G, Syntatic Pattern Recognition:an Introduction ,Addison Wesley,Reading,MA,1978.

[8]Watanabe, Pattern Recognition: Human and Mechanical . Wiley, New York, 1985.

[9] K. Fukunaga. Introduction to statistical pattern recognition (2nd ed). Academic Press, Boston 1990.

[10] R. P.W. Duin,Structural, Syntactic, and Statistical and Pattern Recogition : Joint Iapr International Workshops Sspr 2002 and Spr 2002, Windsor, Ontario, Canada, August 6-9, 2002 Proceedings.

[11] S. Theodoridis, Konstantinos Koutroumbas , pattern recognition , Pattern Recognition ,Elsevier(USA)) ,1982.

[12]. C. Y. Suen. Character recognition by computer and applications. In T. Y. Young and K. S. Fu, editors, Handbook of Pattern Recognition and Image Processing. Academic Press, Inc., Orlando, FL, 1986, pp. 569–586.

[14] J. Tohka, Introduction to Pattern Recognition, Institute of Signal Processing Tampere University of Technology, www.tut.fi/ wohtopr.pdf, downloaded 12April 2005.

[15] Rejea and S. N. Srihari, On-Line and Off-Line Handwriting Recognition: A Comprehensive Survey, IEEE Transactions on Pattern Analysis and Machine Intelligence, v. 22, no. 1, January 2000.

1 [16] R. Plamondon and S. N. Srihari, “On-line and off-line handwritten recognition: a comprehensive survey”, IEEE Transactions on PAMI, Vol, 2000, 22(1), pp. 63–84.

[17] Ahmad T. Al-Taani, “An Efficient Feature Extraction Algorithm for the Recognition of Handwritten Arabic Digits,” INTERNATIONAL JOURNAL OF COMPUTATIONAL INTELLIGENCE, VOL. 2, 2005, ISSN: 1304–4508.

[18] Koerich, A. L., R. Sabourin, C. Y. Suen, “Large Vocabulary Off-Line Handwriting Recognition: A Survey,” Pattern Analysis and Applications, v. 6, no. 2, July 2003, pp. 97–121.

[19] Tal Steinherz, Ehud Rivlin, Nathan Intrator, “Offline cursive script word recognition.

[20] T. Reiss, Recognizing Planar Objects Using Invariant Image Features, Springer- Verlag, 1993.

[21] Amin, A. “Arabic Character Recognition”, Handbook of Character Recognition and Document Image Analysis, World Scientific Publishing Company, 1997, pp. 398.

[22] 1. Cheriet, M.: Strategies for Visual Arabic Handwriting Recognition. Issues and Case Study. In: IEEE Int. Conference on Signal Processing and Applications, Sharjah (UAE) (2007) (invited paper).

[23] http://www.speech.sri.com/people/anand/771/html/node37.html

[24] . S. Impedovo, "Frontiers in Handwriting Recognition", in "Fundamentals in Handwriting Recognition", S. Impedovo (ed.), NATO-ASI Series, Springer- Verlag Publ., Berlin, 1994, G. Dimauro, S. Impedovo, G. Pirlo, A. Salzo Handwriting Recognition: State of the Art and Future Trends.

1 [25] S. Mori, C. Y. Suen and K. Yamamoto, “Historical review of OCR research and development”, Proceedings of the IEEE, Vol. 80(7), 1992, pp. 1029-1058.

[26] Abuhaiba, I.S.I., Mahmoud, S.A., Green, R.J.: Recognition of Handwritten Cursive ArabicCharacters. IEEE Trans. PAMI 16(6), 664–672 (1994), Mohamed Cheriet Mohamed Cheriet, Visual Recognition of Arabic Handwriting: Challenges and New Directions.

[27] D.Bersekas, and R. Gallager, Data Network, Prentice-Hall,1987.

[28] K. Adamopoulos, Application OF Back propagation Learning Algorithms ON Multilayer Perceptrons, 8 May 2000.

[29] S. Rajasekaran, G.A. Vijayalakshmi PAI. Neural network, fuzzy logic, and genetic algorithm synthesis and application, 2004.

[30]http://www.doc.ic.ac.uk/~nd/surprise_96/journal/vol4/cs11/report.htl

[31] http://web.ift.uib.no/Fysisk/Teori/NEURO/neurons.html

[32] D. T. Pham and A. B. Chan, Unsupervised Adaptive Resonance Theory

Neural Networks for Control Chart Pattern Recognition, Proc Instn Mech

Engrs Vol. 215 Part B.

[33] N.B. Karayiannos, A.N. Venetsanopoulos, Artificial Neural Networks

[34] Wilmslow,Introducing Neural Networks. UK: Sigma Press. 1992.

[35] L. I. Perlovsky, Neural Networks and Intellect: Using Model-Based Concepts, Oxford university press, New York, 2001, pages (63-64)

Gamma Ray Burst ToolShed Help. Website: [36]

http://grb.mnsu.edu/grbts/doc/manual/Gamma_Ray_Burst_ToolShed_He.html

L. Zong Ren , Recognition of Handwriting on Postal Letters and [37]
.Parcels,2009
[38] D. J. Scott1, The Discovery of New Functional Oxides Using Combinatorial Techniques and Advanced Data Mining Algorithms. 2008.
[39] A.i Sukhdeve, Mitra Theja K., Nikhil Challa, P. Avinash Rao, Vinay G. and Vivek Chaturvedi Nagpur-4400112007-2008.
[40] A. McAndrew, An Introduction to Digital Image Processing with Matlab Notes for SCM2511 Image Processing 1, Victoria University of Technology, Semester 1, 2004, page 1.
[41] Matlab the language of technical computing [matab help].
[42]An. ,The effect of adding noise during BP training on a generalization performance, neural computation Vol 7, 1996, pp643-674.
[43]
http://www.research.ibm.com/able/doc/reference/com/ibm/able/beans/doc-files/BackPropagation.html accessed date 22/1/2011.
[44] A. McAndrew, An Introduction to Digital Image Processing with Matlab, Notes for SCM2511 Image Processing 1, Semester 1, 2004, pp 17