Off-Line Arabic Handwriting Recognition Using Neural Network

التعرف الضوئي علي الكتابة اليدوية العربية باستخدام الشبكات العصبونية

By

Osama Nayel Al- Sayaydeh

Supervisor

Dr.Venus W. Samawi

Al al-Bayt University

Off-Line Arabic Handwriting Recognition Using Neural Network

التعرف الضوئي على الكتابة اليدوية العربية باستخدام الشبكات العصبونية

By

Osama Nayel Al-Sayaydeh
0720901009

Supervisor

Dr. Venus W. Samawi

A Thesis Submitted to the Scientific Research and Graduate Faculty in partial fulfillment of the Requirements for the Degree of Master of Computer Science

Members	Approved
Dr. Venus W. Samawi (Supervisor)	
Dr. Saad Bani Mohammad	
Dr. Jehad Odeh	
Dr. Waleed A.Jabbar M. Ali Rasheed (Isra Univ)	

Al al-Bayt University

Mafraq, Jordan

2011

Dedication

My greatest appreciation is awarded to my beloved parents who taught me the significance of education.

Acknowledgment

It gives me great pleasure to express my gratitude to all people who have made this study a reality. I am especially most grateful to Dr. Venus Samawi, the supervisor, who always supports and encourages me in different but equally important ways. This support was really behind making this study appear as reality after it was a dream. I thank her very much for her kindness and insightful suggestions. Without her help and contributions, this work could not have been accomplished.

My heartfelt gratitude is offered to the members of the discussion committee: Dr. Waleed A.Jabbar, Dr. Saad Bani Mohammad and Dr. Jehad Odeh for their valuable and important scientific comments which contributed to the improvement of this thesis.

Appreciation is extended to all professors at computer science department. I also thank the staff members of department of computer science.

My greatest thanks are due to my father and my mother for the help, sacrifice and encouragement they offered and the patience they exhibited throughout all the stages of working on thesis. Finally, I owe special thanks to my brothers: Mohammad, Ehab and Anas and my sisters: Ansam, Ashwaq and Wasan for their support.

Osama Nayel Al-Sayaydeh

Tables of contents

List of Tables

List of Figures

List of Abbreviations

Abbreviations	Elaboration
ANN	Artificial Neural Network
Bp	Backprogation
CC	Connected Components
MLP	Multi Layer Perceptron
NN	Nearest Neighbor
OCR	Optical Character Recognition

Abstract

This work aims to produce a system capable of segmenting handwritten Arabic documents to characters or fragments as well as recognizing them using neural networks. It is clear that the success in the process of recognizing Arabic documents will lead to better communication between man and computer, which makes the computer a more effective tool. In order to create a system that is able to recognize Arabic documents correctly, it must be accompanied by a strong method capable of segmenting the documents properly.

Our system mainly consists of four stages: scanning, preprocessing, segmentation and recognition.

- Scanning: is inputting the paper document in the computer by using a scanner.
- Preprocessing aims to improve the image through the use of the smoothing that works to remove noise and fills the gaps, then uses the Binarization for transforming the scanned image into binary image.
- The third stage proposes a new segmentation method. The suggested segmentation stage consists of three steps: labeling connected parts, extracting features form labeled connected parts, segmenting the labeled image into lines and then into fragments or characters.
- The fourth stage is recognition. This stage consists of two levels: the first level consists of two neural networks: one for recognizing the character and the other for fragments. Each segmented part from the third stage is fed to the two neural networks, which are working in parallel (at the same time). The second level is decision level which is used to determine the class of the entered character depending on the highest recognition rate in both neural networks.

In this work, I have been concerned with two things. First, I have developed an algorithm able to improve the accuracy of segmenting Arabic documents to characters or fragments. Second, I have established a neural network to recognize the fragment or characters which were segmented. I have tested this system on a number of handwritten documents of people. Then, these documents were subjected to preprocessing that has in turn improved the image, segmentation and finally recognition.

The developed segmentation method recorded an overall segmentation rate 74.5%, while the recognition rate was 74.77%.

Chapter One
Introduction

1.1 Background

In spite of the great development in computers, in terms of speed in completing the business and access of information, our primary ambition is to search for computers able to communicate with human beings in an actual way and also have the ability to distinguish patterns such as voices and handwritings effectively.

Handwriting has received an attention by many researchers and is considered not only one of the most important challenges, but it is also one of the oldest in the field of computer. The reason of this great interest is to improve the communication between human and computer, which makes the computer friendlier to use [Cha02]. Understanding the handwritings by computer is due to the importance of optical character recognition (OCR) for business, office, mailing address, and reading in addition to check processing etc [Ker07, Cha04, Pet04].

A very serious difficulty facing the researchers in recognizing handwriting is the variety and uncertainty of human writing, not only because of the great variety in the shape, but also because of the overlapping and the cursive of the character. The cursive problem not only exists in English but also in Arabic language. In English, the problem is only in handwriting style, but in Arabic the problem is in handwriting and printed styles. For this reason, Arabic is named fully cursive language. Figure (1.1) shows the cursive problem in Arabic and English languages. For example, figures (1.1.a, 1.1.b, 1.1.d) show the cursive among the characters, but in the figure (1.1.c) there are no cursive characters.

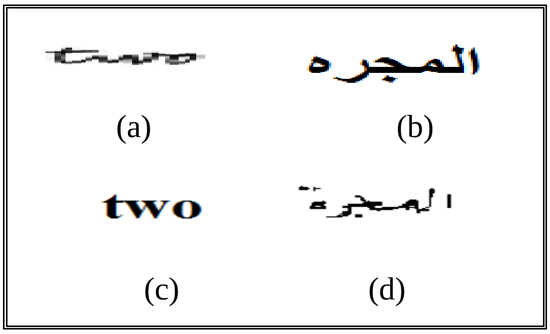


Figure (1.1) cursive problem in Arabic and English

(a) English word "two" with cursive (b) Arabic word "المجره"

(c) English word "two" without cursive (d) Arabic word "المجره"

There are two kinds of handwriting recognition: *online recognition* and *offline recognition*. In online recognition, the text is entered in the computer using special devices, such as light pen, tablet PC, or PDA, where a sensor traces the pen movements as well as pen-up/pen-down switching and the text is recognized in real time [Kla01, Bai04]. Offline recognition deals with the text in the form of an image that is incorporated into a computer using a scanner or camera and can be converted to a text that can be modified. By comparing the two methods, offline and online, it was found out that the online way is easier than offline one. This could be due to the fact that in online way we have the Preliminary information about the entered text like the movement of pen and (x, y) coordinate pairs. But in the offline way, we don't have any information about the entered text beforehand, because we are dealing with the form of image [Alm08, Pet04]. For this reason, the results obtained by using offline are still low when compared with the results of online. In this work, we have used the offline method for recognizing Arabic handwriting.

1.2 Arabic Script

The offline method has been used in several languages such as Japanese, Latin, Chinese, as well as Arabic. As a result, good findings were obtained in recognizing these languages [Vas06, Tay02]. Although the Arabic characters are used in more than one language such as Persian and Urdu, the researches published in Arabic are few when compared with other languages (Chinese, English) and the recognition accuracy is still low. This is due to the nature of Arabic language.

More than 350 million people in Africa and Asia use Arabic language in writing and speaking, but there are also some people using only Arabic characters in writing without pronouncing them such as Parisian and Urdu. In addition, Arabic language has many features and characteristics that distinguish it from other languages and this makes the process of recognition very difficult [Ali08]. In fact, the main Arabic language characteristics are:

1. In contrast to English text, Arabic is written from right to left, rather than left to right.
2. Arabic language consists of 28 letters distributed as follows: 15 characters having points, whereas 13 characters without points. Table (1.1) shows Arabic characters.

Table (1.1) Arabic alphabet

Characters With points	Character Without points
ب	أ
ت	ح
ج	د
خ	ر
ذ	س
ز	ص
ش	ط
ض	ع
ظ	ك
غ	ل
ف	م
ق	ه
ن	و
ي	

3. There are diacritical marks in Arabic languages which are used in order to distinguish the meanings of words. Diacritical marks are either above or below the characters. Figure (1.2) shows two words with the same shape but with different diacritical marks. In figure (1.2.a) the word means "engagement asking for marriage", but in figure (1.2.b) the word means "giving a speech".

خِطّبة	خُطّبة
(a)	(b)

Figure (1.2) different diacritics in same Arabic word
(a) Arabic word "خِطبة" (b) Arabic word "خُطبة"

4. There are characters that have the same shape, but they are different in the location of the point above the character or under the character as shown in table (1.2).
5. Each character in Arabic language has more than one form (shape). The forms of a character change according to its position in the word. There are 22

characters, each character has four forms. There are 6 characters, each of which has two forms. Table (1.2) shows Arabic characters in all forms.

Table (1.2): Arabic character in all forms

Isolated	Initial	Medial	End
أ	-	-	ـا
ب	بـ	ـبـ	ـب
ت	تـ	ـتـ	ـت
ث	ثـ	ـثـ	ـث
ج	جـ	ـجـ	ـج
ح	حـ	ـحـ	ـح
خ	خـ	ـخـ	ـخ
د	-	-	ـد
ذ	-	-	ـذ
ر	-	-	ـر
ز	-	-	ـز
س	سـ	ـسـ	ـس
ش	شـ	ـشـ	ـش
ص	صـ	ـصـ	ـص
ض	ضـ	ـضـ	ـض
ط	طـ	ـطـ	ـط
ظ	ظـ	ـظـ	ـظ
ع	عـ	ـعـ	ـع
غ	غـ	ـغـ	ـغ
ف	فـ	ـفـ	ـف
ق	قـ	ـقـ	ـق
ك	كـ	ـكـ	ـك
م	مـ	ـمـ	ـم
ن	نـ	ـنـ	ـن
ه	هـ	ـهـ	ـه
و	-	-	ـو
ي	يـ	ـيـ	ـي

6. Arabic is written cursively in printed or handwritten styles. Figure (1.3) shows the cursive problem in handwriting and printed style.

(a) (b)

Figure (1.3) cursive in Arabic in handwritten and typewritten

(a) typewritten word "عمان" (b) handwritten word "عمان"

7. A word in Arabic language may be one sub word or more. See the figure (1.4) which shows the sub words in one Arabic word. In the figure (1.4.a) the word consists of five sub words, each sub word is underlined. In figure (1.4.b) the word consists of two sub words, each sub word is underlined.

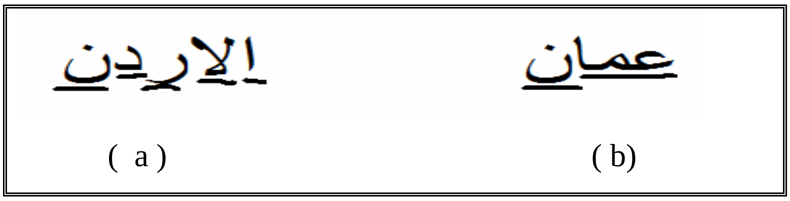


Figure (1.4) two Arabic words consist of sub words.
(a) Word "أردن" (b) word "عُمان"

8. Ligature means merging two or three characters into one form. Figure (1.5) shows ligature in some Arabic words. For example, in figure (1.5.a) there is one ligature in word "بحر"; the ligature is between two characters "ب"and "ح", whereas in figure (1.5.b) there is ligature between three characters "ل","م" and "ح". In fact, this characteristic not only depends on the character itself but it also depends on the selected Arabic font. Figure (1.6) shows lists of ligatures found in Arabic font.

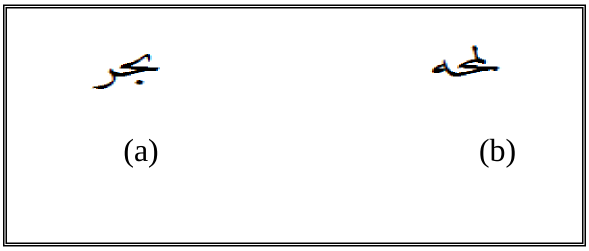


Figure (1.5) Ligatures in Arabic
(a) Word "بحر" (b) word "لمحه"

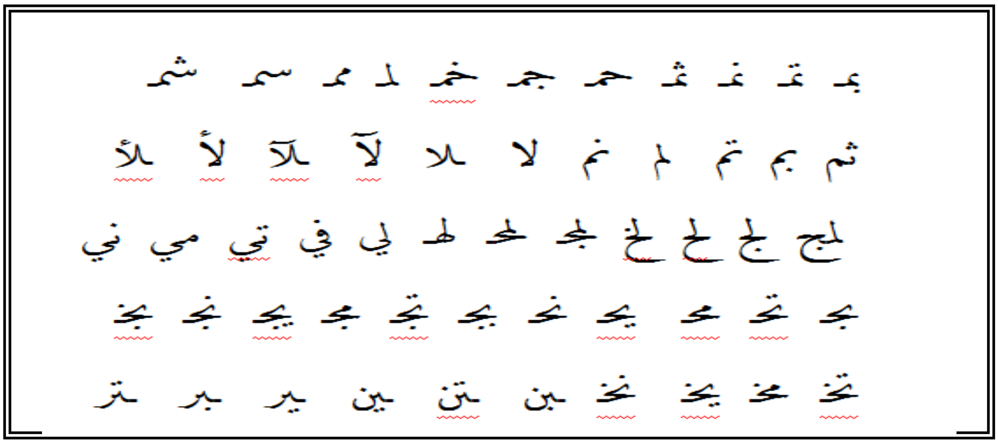


Figure (1.6) ligature found in Naskah fonts

9. There are other features of Arabic language, which make the recognition process more difficult, such as the existence of different types of fonts.

1.3 Problem Statement

Recognition in Arabic handwriting is very important, especially in the fields of office automation, processing bank checks and the mailing addresses. The recognition of text means to transform the human writings into machine readable text, but the most difficult problem in Arabic handwritten text recognition is the cursive of handwriting, which makes the segmentation process very challenging. Another major problem is the shape difference of the same character if it is written by different persons as in "سـ" and "  ". This could be due to the variability of human writings in Arabic. Therefore, it is important to develop a system capable of segmenting Arabic word to characters accurately and improve the character recognition ability.

1.4 Literature Review

The interest in recognition of Arabic writing started lately compared with other languages, such as Japanese, English and Chinese. The first recognition of Arabic writings came into sight in 1975. However, this recognition process was concerned with Arabic printed text. Afterwards, the concern in Arabic has increased through publishing various researches which are not only concerned with printed texts but also interested in handwritten ones. Moreover, there are also researches dealing with the process of dividing Arabic writing (printed or handwritten) into characters or connected parts.

In this work, we will cite some of the important researches concerning the recognition of Arabic language characters, indicating the used methods, the weaknesses, and the conclusions of these researches.

In (2003), Mohammad Sarfraz, Syed Nazin Nawan and Abdulaziz Al-Khuraidly [Sar03] proposed a technique for recognizing Arabic *printing* text using neural network. They used four stages for recognizing text: the first stage is pre-processing which includes Binarization, smoothing and normalization. The second stage is segmentation of text into line, word and then into characters. The third stage is features extraction using moment invariant (moments of an image can be thought as the decomposition of image into a series of numbers that describe the distribution of the image function) as features vector. The last stage is neural network which is used for recognition. The system showed recognition rate of about 73%. It is clear that this study did not deal with handwritten Arabic text.

Labiba Souici and Mokhatar Sellami in (2004) [Sou04], developed an Arabic literal amount recognition system that uses a neuron-symbolic classifier. This system consists of four main stages: the first stage is concerned with structural features extracted from words contained in the amount vocabulary such as (desenders, ascender, loop, dots). The second stage is to build symbolic knowledge base that reflects a classification of words according to their features. The third stage is to use translation algorithm to convert symbolic representation into neural network to determine the neural network architecture and initialize its connection with specific value. The fourth stage, is empirical learning used to recognize new handwritten amounts. This study achieved recognition rate 93%. This study was limited to the words literal Arabic amounts such as "سبعون" ،"ثمانون".

In (2006), Somay Alm'addeed [Alm 06] proposed a system for recognizing handwritten Arabic words using neural network. The first stage is pre-processing which contains normalization, which attempts to remove some of the variation in the images which do not affect the identity of the word. The second stage is feature extraction. This paper deals with global features (descenders, ascenders, number of loops, number of segments, lower dots, upper dots,) which are extracted from word. The neural network used global feature as input to recognize Arabic word. The final system produced had an overall recognition rate 63%.

A novel holistic technique for classifying and retrieving Arabic handwritten text document proposed by Salma Brook, Zaher Al-Aghbari in (2008) [Bro08]. The system consists of three main steps: first is segmenting the Arabic text images to words; then, the words images are segmented to connected parts. Second, several features are extracted from these connected parts and then combined to represent a word with one consolidated feature vector. Finally, classification step using feed-forward Neural Network is used to classify the connected parts.

In the same year, Z.shaaban [Sha08], suggested a new approach to tackle the problem of recognizing machine printed Arabic texts. The first step in the system is normalization of scanned image for reducing the size of image, and then segmented the normalized image into characters, finally, sent the segmented image to classifier. The suggested scheme depends on multiple parallel neural networks classifier. It consists of two phases. First phase categorizes the input character into one of eight groups. Second phase classifies the character into one of the Arabic character classes in the group. The system has been tested on more than 100 Arabic text images. The system achieved high recognition rate 98%. This study dealt with the printed Arabic text only.

In (2009), Jawad H.ALkhateeb, Jianmin Jiang, Jinchang Ren and Stan S Ipson [ALk09] proposed a machine learning approach for classifying handwritten Arabic word. The proposed approach consists of three stages. Pre-processing stage, consists of word segmentation and normalization to remove as much as possible of the variations in handwritten images for consistent analysis and robust recognition. Feature extraction stage used three different feature extraction methods for each segmented word namely the Discrete Cosine Transform (DCT), Moment Invariants, and Absolute Mean Value of overlapping blocks. Finally, classification stage used the features that are extracted in second stage to train NN to recognize words. The system used IFN/ENIT database for testing purposes which consist of 32492 Arabic words. The recognition rate from using DCT features is 80.74% and from using Moment Invariants is 75.74.

1.5 Thesis Objectives

In this work I have been concerned with two aims: first, I have developed an algorithm able to segment Arabic documents to characters or connected parts. It is clear that the segmentation stage is significant and complicated in Arabic language. This can be attributed to font shapes, types, and cursive nature. Therefore, it is important to segment the Arabic words into characters in an accurate way since segmentation stage is very important and affects the recognition accuracy. The handwritten Arabic document will be segmented into lines, then connected-parts or characters.

The second aim is to suggest a robust recognizer based on feed-forward neural networks. To improve the recognition ability, the suggestion is to construct two different recognizers, one for recognizing isolated Arabic characters, and the other for recognizing the fragments. Finally, the suggested system is tested using 15 different handwritten documents.

1.6 Thesis Organization

This research consists of five chapters organized as follows:

- ***Chapter two*** (Theoretical Background): consists of three parts: part one presents introduction. Part two describes the OCR system components, scanning, pre-processing, segmentation, feature extraction and classification. Part three describes the artificial neural network, types of neural network; learning and feed-forward Neural network Backprogation (BP).
- ***Chapter three (***Design and Implementation) consists of two parts: part one presents introduction. Part two describes the proposed system, scanning, pre-processing, segmentation, classification and decision level.
- ***Chapter four (***Assessment Result) consists of three parts: introduction. Part two describes the result of segmentation algorithm and presents the data set. Part three presents the recognition accuracy, neural for characters and fragments.
- ***Chapter five*** (Conclusion and future works)***:*** concentrates on conclusions with recommendations for future works.

Chapter Two

Theoretical Background

2.1 Introduction

When considering the previous studies and published researches in the field of recognizing writing (handwritten or printed), it was found that five main phases are needed in order to achieve writing recognition. These phases are: scanning, preprocessing, segmentation, feature extraction, and classification. Figure (2.1) shows the general layout model of writing recognition.

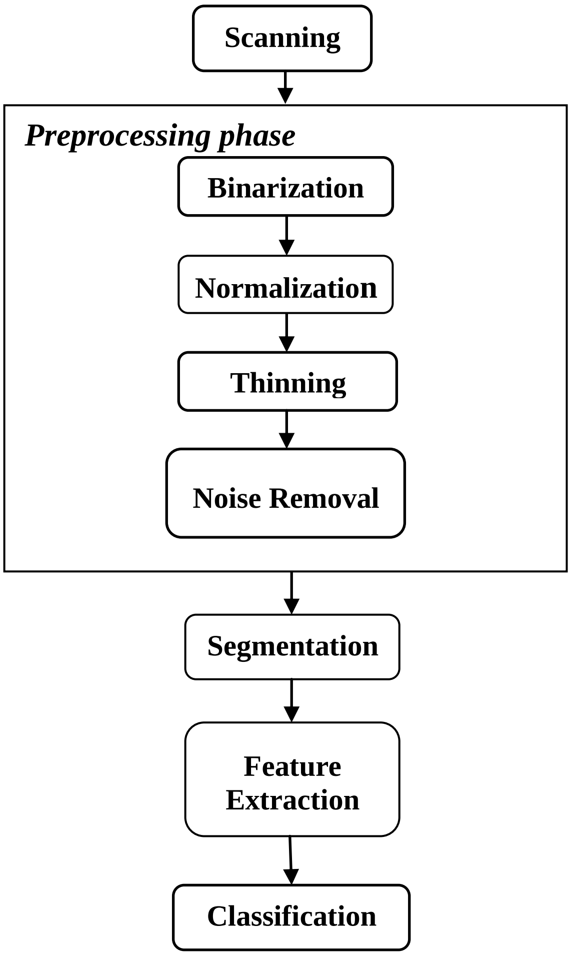


Figure (2.1) step involved in the OCR system [Abd07]

In this work, the concentration is on both segmentation and classification phases. Therefore, an algorithm is suggested for improving the Optical Character Recognition (OCR) document segmentation and Neural Networks have been used as recognizer.

This chapter is concerned with exploring OCR system components in general, such as preprocessing, segmentation, feature extraction and classification in addition to illustration of neural network.

2.2 OCR System Components

Optical character recognition (OCR) is one of the most successful applications of automatic pattern recognition. In general, any OCR system consists of five phases (scanning, preprocessing, segmentation, feature extraction, and classification) as shown in figure (2.1).

2.2.1 Scanning Phase

Scanning is the first step on OCR system; this step is carried out by using a scanner device or digital camera which converts the paper into an image stored in computer. Resolution of scanned image affects the OCR accuracy. It has become obvious when scanning resolution increases, the OCR accuracy and file size are increased [Alm08]. In our work we have used scanner devices as input unit.

2.2.2 Preprocessing Phase

Preprocessing is a set of operations applied to scanned image to enhance the quality of image. This step is very important in handwritten text to reduce the variation in different writing styles and noise. This step is also very significant because the recognition accuracy depends on the quality of the input image. This step is carried out by using a set of operations such as [Gar09, Ari98]: Binarization (thresholding), noise removal, normalization and thinning.

A) Binarization

Binarization is a process that is used to convert a gray scale image format or RGB (Red, Green, and Blue) image into binary image. A binary image consists of two levels: 0's or 1's. 1's is a black pixel that indicates the characters or any object. 0's is a white pixel that indicates the background pixels [Cha 02, Tri 95]. See figure (2.2) which illustrates the result of Binarization process. The main aim of using this process is to reduce storage requirements and to increase processing speed. In this work, I have dealt with different formats of scanned images.

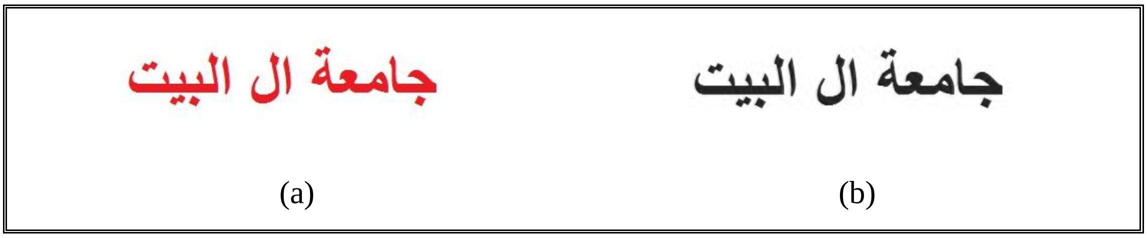


Figure (2.2) Binarization process of the word "ال البيت"

(a) Original image (b) the result of Binarization

B) Normalization

Normalization is applied in order to reduce the variability in different handwriting styles. *Skew detection* and *slant correction* are also used to perform normalization.

After the Binarization step, *skew detection* is performed. The aim of this process is to align the image text into right direction. The skew occurs during document scanning or copying because the paper or pages may not be fed straight into scanner device. Figure (2.3) shows the skew problem. It is important to get rid of this problem since it will reduce the accuracy of segmentation and classification processes. To eliminate the problem, at first, detect the skew angle (skew detection) [Hul98]. Then, use skew correction which rotates the scanned page into correct angle. Skew detection and correction are done by using different methods such as [Bou06, Hul98]: *Hough Transform*, *Projection Profile*, *Fourier transform,* and *Nearest Neighbor* (NN).

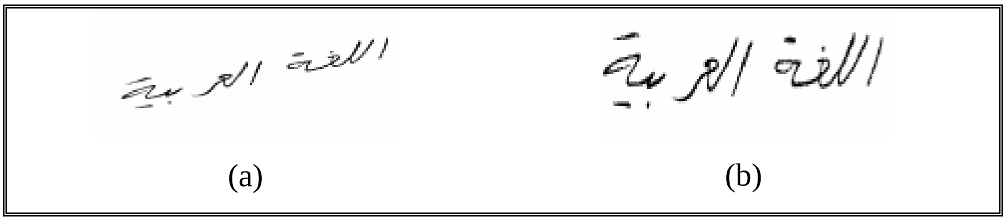


Figure (2.3) skew correction of word "اللغة العربية"

(a) Original image (b) the result of skew correction

Slant correction is used to eliminate the slope in the character or connected parts found in cursive writing as shown in figure (2.4). The main purpose of slant correction is to reduce the variations of different handwriting styles. For this reason, slope should be eliminated because it dramatically reduces the accuracy of segmentation and classification. Slant correction could be implemented using different methods such as [Kul05,]: *Projection Profile,* and *Hough Transform*.

(a) (b)

Figure (2.4) slant correction of word "اللغة العربية"
(a) Original image (b) the image after apply slant correction

C) Thinning

The main aim of thinning is to minimize an object until it becomes one pixel wide as shown in figure (2.5). Thinning process makes the recognition process easier since the width of outline is reduced to a single pixel. As a result, the memory and time consumption in recognition stage are reduced.

(a) (b)

Figure (2.5) thinning of word "محمد"
(a) Original image (b) the result of thinning process

D) Noise Removal

Noise Removal is used to remove unwanted objects (noise objects) from scanned image. Noise occurs in scanning process or in the writing instrument. It could be: *disconnected lines segment, gaps and bumps in lines and rounding of corners*. Many techniques are used to solve this problem, such as [Upp07]:

- ***Filtering***: the main aim of this technique is to remove noise and reduce unwanted points. Filtering can be used for smoothing, sharpening, thresholding, removing slightly texture or colored background and contrast adjustment purpose [Kav06].
- ***Morphology***: A broad set of image processing operations that affects the image depending on shapes is so called *morphology*. Morphological operation applies a structure element to the image. The output resulted image will be of the same size. Each pixel's value in the output image is based on a comparison of the corresponding pixel in the input image with its neighbors. By choosing the size and shape of the neighborhood, you can construct a morphological operation that is sensitive to specific shapes in the input image. The basic morphological operations are “Opening” and “Closing”. Morphological opening and closing

changes the definition of pixel set depending on the neighborhood pixels and structure element. The most basic operations in morphology are *dilation* and *erosion* [Lei01,Blo91].

Dilation means adding pixels to the boundaries of objects in an image, while erosion means removing pixels on object boundaries. The number of pixels added or removed from the objects in an image depends on the size and shape of the *structuring element* (which is a matrix consisting of only 1's and 0's that can have any arbitrary shape and size) used to process the image. In the morphological dilation and erosion operations, the state of any given pixel in the output image is determined by applying a rule to the corresponding pixel and its neighbors in the input image. The rule used to process the pixels defines the operation as dilation or erosion. Figure (2.6) shows the dilation process on a binary image.

The morphological dilation increases the size of objects, filling in holes and broken areas and connecting areas that are separated by space smaller than the size of structure element. In this work, morphological operations are used to fill the gaps between the objects in image using first technique Dilation.

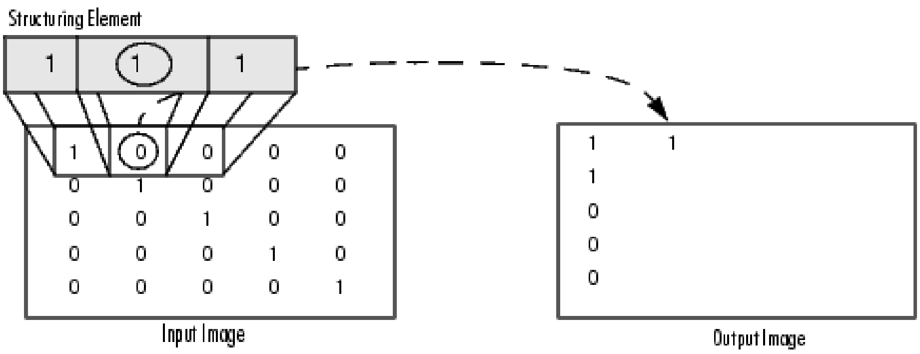


Figure (2.6) the dilation process on a binary image

2.2.3 Segmentation

Segmentation generally refers to the process of subdividing image into its constituent regions or objects. The aim of segmenting an image is to make it more meaningful and easier to be analyzed. The segmentation process is used in many practical applications such as *medical imaging*, *locate objects in satellite images*, *face recognition*, *fingerprint recognition*, and *text segmentation* (the concern of this research).

Text segmentation is defined as the process of dividing written text into meaningful units (words, sub-words, or characters) [See 02, Cha 02]. Segmentation process is a significant stage in text recognition systems, especially in Arabic language (the Arabic language contains cursives and ligatures). The recognition stage is mainly based on the accuracy of segmentation stage. Inaccurate segmentation will cause unexpected behavior of the recognition stage. The segmentation process is divided into two main approaches:

1- ***Holistic approach***: in this approach, the features are extracted from the scanned image without image segmentation. Take the word image from the input device, calculate number of features (Ascender, number of dots, Descender, loop, connected components, etc..) which contain useful information about the word image. Finally, the word image is compiled according to the features in the dictionary. Figure (2.7) shows some global features used in this approach. At the recognition phase, each word input (features) will be compared with word features in the dictionary. We noticed that there is no segmentation in this method but a number of features were taken from the image itself. Afterwards, only the features of the image were entered in the recognizer. On one hand, this method is regarded not only quick but also effective when used with printed fonts. On the other hand, this method is not successful when used with handwritten texts because the extraction of global features from texts resulted in big variety in writing.

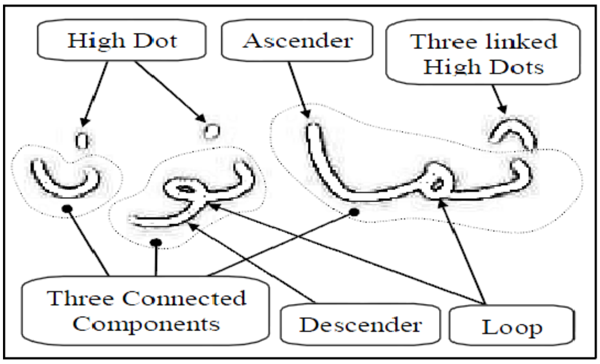


Figure (2.7) global features in the word "ثمانون"

2- ***Analytical approach***: in this approach, the image is segmented into words then to characters or connected parts. In this work, analytical approach of segmentation is used.

In the analytical approach, many methods are applied to segmenting the text image into lines, then word, then into connected parts and characters. In general, this approach is very difficult and considered a critical stage because any error in segmentation will affect the accuracy of the classification phase. There are many methods used in the analytical approach such as:

A) Projection Profile

Projection profile counts the number of pixels values in each column and row for text image. The projection profile method has two types: horizontal and vertical [Bro08, Sha08, Elg01, Hul98].

- Horizontal projection profile counts the number of ink pixels (black pixels in binary images) along every row in the text image. Horizontal projection is used to segment the text into lines; white space between the lines is used to segment the image text into lines. Figure (2.8) shows the horizontal projection [Bro08].

(a) Original image (b) Horizontal projection

Figure (2.8) horizontal projection for determing baseline

- Vertical projection counts the number of black pixels along every image. This method is used to segment the line into word, or segment the word into connected parts. The spacing between the words shows the gaps (zero pixels) between the words or intra word. Figure (2.9) shows the vertical projection [Elg01, Hul98].

(a) Original image (b) Vertical projection

Figure (2.9) vertical projection example

B) Labeling Connected components

This method is used for identifying the isolated character. It is simple and effective method for segmenting binary image by examining the connectivity of pixels with their neighbors and then labeling the connected sets. The connected components may be 4 or 8 connected in 2D image. In this work, we depend on this approach for segmenting the text image [Vel 10, Ros66]. Next chapter shows this method in details.

2.2.4 Feature Extraction

Feature extraction is the process of extracting useful information from an image to pass them to the classifier. The features are a measurement applied to image and combined together to generate vector called feature vector. The purpose of extracting features is to find out the significant characteristics of the image of the character which we use for describing the character. In fact, the process of extracting features comes before recognition stage. The feature extraction is divided into two categories as follows [Bro08, Alm06, Sou04, Moz05]:

- *Structural features* are the most common features. These techniques examine the geometry and topology of character image e.g strokes, endpoints, intersections of line segments, loop, dots and their position (above/below) the baseline[Sou04][Alm06][Ami06].

  Structural features are significant for solving the variety problems in using multi fonts, but structure features are difficult to extract from the Arabic document image and many errors occur because of the small difference among Arabic characters.
- *Statistical features* describe a character image by extracting number of characteristics measurements from the character image. These features include Zoning and Moments. Zoning feature is based on segmenting the text to zones in which pixels density is used in these zones as features. Moments describe numerical quantities at some distance from a reference point or axis, such as: Hu moments and Zernike moments [Baz99, Par97, Sar03].

2.2.5 Classification

The main aim of this process is to make decisions about the class membership of a pattern. The classifier attempts to identify the pattern (character image) that represents the input features. There are three approaches for recognizing a character: *statistical*, *structure* and *artificial neural network (ANN).*

A) Statistical classifier

In this type of classification, the character is carried out by choosing the character class which is most probable, or has maximum measure of expected classification error. There are many of statistical methods applied on OCR system such as: Hidden Markov Modeling (HMM) which is the most efficient method in the statistical classification and widely used in handwritten and printed text.

B) Structural classifier

This method depends on the shape of character. The structural features of a character are the strokes, holes or other features such as concavities.

C) Artificial neural network (ANN)

Artificial Neural networks (ANN) deal with processing data in a way similar to human brain. Like mental human thinking, neural networks process information to obtain the best solution. In fact, these ANN use a set of learning algorithms to learn. Neural networks have been widely used in the field of pattern recognition (OCR, face recognition, voice recognition).

2.3 Artificial Neural Network

The term of the artificial neural network (ANN) came from the idea of the work of human brain. The main objective of the establishment of neural networks is to simulate the thinking of the human mind and to make the computer more interactive and smart tool. Together, these will make the computer more capable of solving different kinds of problems, such as recognition of patterns, prediction, ..etc.

Artificial Neural Network (ANN) is information processing inspired in natural neurons. Like human brain, neural network obtains knowledge through learning. Neural network knowledge is stored within inter neurons connection. The basic unit of neural network is neuron which simulates the basic function of biological neuron. Figure (2.10) shows the biological neuron [Fau94].

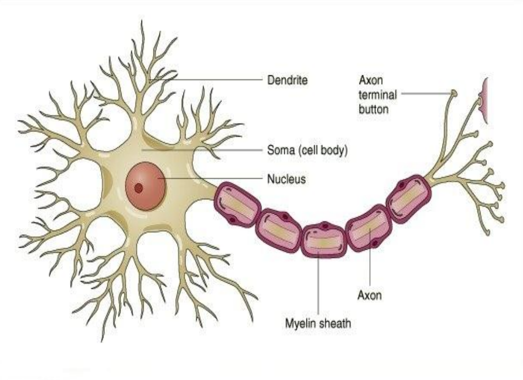


Figure (2.10) biological neuron

Artificial neuron consists of:

- A set of links which describes neuron inputs, with weights $W_1, W_2 \ldots, W_n$.
- An adder function for computing the weighted sum of the inputs (p is number of inputs) and an activation function for limiting the amplitude of the neuron output [Zur96]:

$$F(x_i) = \sum_{i=1}^{p} W_{ji} X_i \qquad (2.1)$$

$$\text{Log- Sigmoid function} = \frac{1}{1+e^{-n}} \qquad (2.2)$$

- The neuron includes special fixed input known as bias. The bias is an external parameter of the neuron. Figure (2.11) show the neuron and bias.

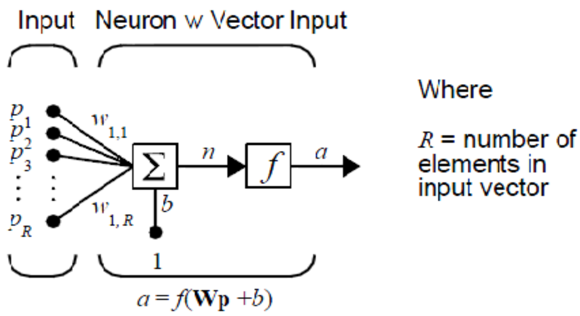


Figure (2.11) The artificial neuron [Math07]

2.3.1 Types of Neural Network

Neural network is divided into dynamic and static networks. Static networks have no feed-forward elements and contain no delay. The output is calculated directly from the input through feedfoward connection. In dynamic networks, the output depends not only on current input to the network, but also on the current or previous inputs, outputs, or states of the network.

Neural network can be divided in terms of architecture into three main classes, Single layer Perceptron (SLP), Multi-layer Perceptron (MLP), and Recurrent (feedback) [Fau94].

A) Single layer Perceptron (SLP).

Single layer network is the oldest structure of neural network, which consists of a single layer of output nodes. The inputs are fed directly to output by series of weight. The input layer is fully connected to the output layer but is not connected to other input units. Figure (2.12) shows the single layer network.

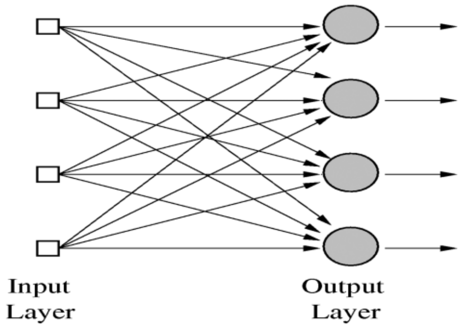


Figure (2.12) A single layer model [Fau94]

B) Multi-layer Perceptron (MLP)

The MLP is the most popular type of NN architecture. In MLP there is one or more layers of neurons called hidden units between input units and output units. The input nodes pass the information to the units in the first hidden layer, then the outputs from the first hidden layer are passed to the next layer, and so on. This structure is called feed-forward because the connection lines between input, hidden, and output is unidirectional link (on way). Figure (2.13) shows the multilayer networks [Fau94]:

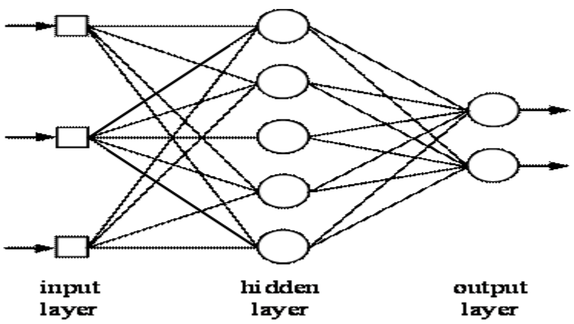


Figure (2.13) A multi layer model [Fau94]

C) Recurrent (feedback)

This type of network has connections that go backward from the output nodes to the input nodes. Recurrent NN has arbitrary connections between any nodes. The next state of a network depends on the connections weights, currently input and the previous state of the network. Recurrent takes long time to compute stable output and it is difficult in learning. This network is used to solve problem that depends on previous inputs, such as predict the weather tomorrow. Figure (2.14) shows recurrent networks.

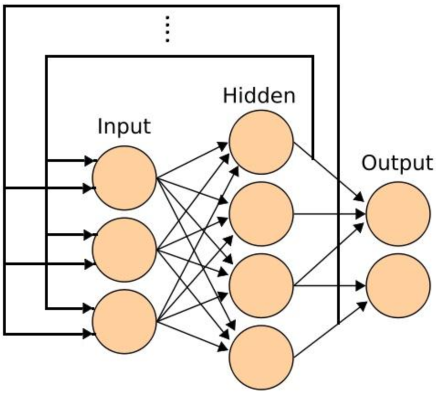


Figure (2.14) A recurrent network with one hidden layer [Fau94]

2.3.2 Learning

Neural network behaves like human brain because it learns from experiences by changing the connection weight. The learning process of neural networks is the basis to guide the network to perform various operations such as recognition. Learning or training the network is done by giving the network a set of examples that should be selected very carefully because it will contribute to accelerate the process of learning network. The set of training examples is called training set. The learning ability of a neural network is determined by the architecture and the algorithm method chosen for training.

Neural Network training methods are generally divided into two types: supervised and unsupervised. In supervised learning, the input and target output are provided to neural network. While in unsupervised learning, it is not possible to determine what the result of the learning process because of no target output.

A) Supervised Learning

In supervised training, training set of examples and desired output are provided. The computation model applies each entry in the training set and learns by examples. Supervised training that precedes the neural network is taken through several iterations until the actual output matches the desired output. Many problems are solved such as recognition and regression [Fau94].

B) Unsupervised Learning

In unsupervised training, no desired outputs are provided to the network. The net modifies the weights so that most similar input vectors are assigned to the same output unit. It is also referred to as self-organizing [Fau94].

2.3.3 Feed-forward Back-propagation Neural Network

Back-propagation is the most popular supervised training algorithm for neural network [Rum86]. The first actual use of back-propagation algorithm was in 1986 by Rumelhart, Hinton and Williams. The NN explained here consists of three layers: input layer, hidden layer, and output layer.

The first part of feed-forward describes how this neural network processes the pattern and recalls it. In feed-forward NN, each layer has a connection to the next layer; this connection feeds to next layer only (i.e. no connection back).

The second part back-propagation describes how neural is trained. The neural network is presented with training data. The actual results of neural network are compared with the desired results. The difference between actual and desired results produces an error. The calculated error is used to adjust the weights of the various layers backwards from the output layer all the way back to the input layer. The basic back-propagation algorithm consists of three steps:

- The training data is fed to input layer, then propagated to hidden layer and to the output layer.
- The actual result is compared with the desired output by subtracting the actual output from desired output to produce the error signal.
- The error signal for each desired output value is then back-propagated from the output to the inputs in order to appropriately adjust the weights in each layer of the network. The main goal of error signal is to update the weights. The weight is updated by using the following formula:

$$W_{new} = W_{old} + \partial\,(\text{desired} - \text{actual}) \times \text{input} \qquad (2.3)$$

∂ = the learning rate.

W= the weight.

***Algorithm*: *Back-Prop agation Algorithm for one hidden layer (using Delta Rule)* [Fau94]**

Training a network by backpropagation involves three stages: *feedforward of input pattern*, the *backpropagation of the associated error*, and *weight adjustment*.

Step0: Let V be the weights between the input layer and the hidden layer

Let W be the weights between the hidden layer and the output layer

Let v_{0j} be the bias on hidden layer j.

Let w_{0k} be the bias on output k.

Initialize weight vectors (set them to small random values (between -0.5 to +0.5 (or between -1 and 1 or other suitable interval) higher initial values tend to result in saturation region after activation, where as small initial values result in values close to zero)).

Initialize α learning rate.

(Possible choices for weight initialization will be discussed later).

Step1: While stopping condition is false, do steps 2-11.

Step2: E=0

Step3: For each training pair do steps 4-10

Feedforward (steps 4-5)

Step4: For each hidden unit, (z_j, j=1,..,h), find

$$z\text{-}in_j = v_{0j} + \sum_{i=1}^{n} x_i v_{ij} \qquad (2.4)$$

$$z_j = f(z\text{-}in_j) \qquad (2.5)$$

where $f(.)$ is an activation function

Step5: For each output unit, (y_k, k=1,..,m), find

$$y\text{-}in_k = w_{0k} + \sum_{j=1}^{h} z_j w_{jk} \qquad (2.6)$$

$$y_k = f(y\text{-}in_k) \qquad (2.7)$$

where $f(.)$ is an activation function

Backpropagation of error (steps 6-7)

Step6: For each output unit (y_k, k=1,..,m), receive a target pattern corresponding to the input training pattern, compute its error information term.

$\delta_k = (d_k - y_k) f'(y\text{-}in_k)$ (2.8)

where d is the desired output

calculate its weight correction term (used to update w_{jk} later)

$\Delta w_{jk} = \alpha \delta_k z_j$ (2.9)

calculate its bias correction term (used to update w_{0k} later)

$\Delta w_{0k} = \alpha \delta_k$ (2.10)

and sends δ_k to units in the layer below.

Step7: For each output unit (z_j, j=1,..,h), sum its Delta inputs (from units in the layer above).

$$\delta\_in_j = \sum_{k=1}^{n} \delta_k \, w_{jk} \quad (2.11)$$

multiply by the derivative of its activation function to calculate its error information term. $\delta_j = \delta\_in_j \, f'(z\text{-}in_j)$ (2.12)

calculate its weight correction term (used to update v_{ij} later)

$\Delta v_{ij} = \alpha \delta_j x_i$ (2.13)

calculate its bias correction term (used to update w_{0k} later)

$\Delta v_{0j} = \alpha \delta_j$ (2.14)

Update weights and bias (steps 8-9)

Step8: For each output unit (y_k, k=1,..,m) update its bias and weights ($j=0,\ldots,h$)

$$w_{jk}^{t+1} = w_{jk}^{t} + \Delta w_{jk}^{t} \quad (2.15)$$

Step9: For each hidden unit (z_j, j=1,..,h) update its bias and weights ($i=0,\ldots,n$)

$$v_{ij}^{t+1} = v_{ij}^{t} + \Delta v_{ij}^{t} \quad (2.16)$$

Step10: $E = E + \frac{1}{2} \sum_{k=1}^{n} (d_k - y_k)^2$ (2.17)

Step11: test stopping condition ($E < Emax$) Goto step 2

Chapter Three

Design and Implementation

3.1 Introduction

To attain the aims of the study, segmentation algorithm has been developed to partition handwritten text into characters, or fragments. Two neural networks have been suggested to recognize the segmented characters, and fragments respectively.

This chapter discusses the phases and approaches that have been used for designing a complete system capable of segmenting and recognizing Arabic handwriting text.

3.2 The Proposed System

The proposed approach consists of four primary phases: Scanning, preprocessing, segmentation, recognition. Figure (3.1) shows the proposed system phases.

In the first phase, an image has been taken and saved in computer by scanner. Then, preprocessing has been conducted so as to get rid of noise available in the image. In segmentation phase, a text has been divided into characters, or fragments. Finally, two levels of recognizer are suggested. Level one consists of two neural nets, and decision level will be made (represents level two). In decision level, a comparison of the output of characters of neural network with the output of fragments of neural network has been made. The decision is based on suggested rule assumption (will be illustrated in the recognition phase). In order to get acquainted with the system in detail, the system phases will be illustrated in details.

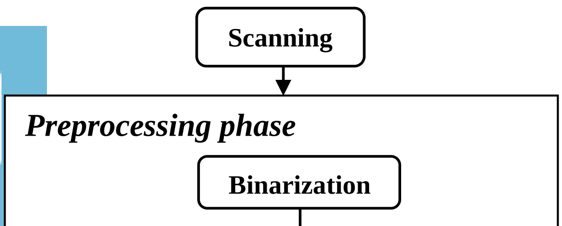


Figure (3.1) The suggested system

3.2.1 Scanning

At this stage, the document image is produced using the scanner device and saved in computer. The image of the scanned document is saved in the computer in the form of bmp, jpg, or png. In this work, the resolution of the scanned document is 300 dpi.

3.2.2 Preprocessing Phase

This phase is very important because it affects the accuracy of the next phases (segmentation and recognition phases). In fact, the main goal of preprocessing phase is enhancing the image quality by removing the noise from the image. Preprocessing phase includes many steps. In this work, we use two steps: Binarization (Thresholding), and smoothing.

A) Binarization

The first prepossessing step used in this work is Binarization. It is used to convert the gray scale image or color image into binary image based on threshold value. In thresholding, a threshold value T is selected (T value is 0.5 and it is midway between black and white). Set all pixels ≥ T to 1, which means it is black pixel. Set all pixels < T to 0 value which means it is white pixel. Thus, thresholding is transformation of an input image ***f*** to a binary image ***g*** depending on threshold T as follows:

$$g(x,y)=\begin{cases}1 \text{ } if \text{ } f(x,y)>T\\0 \text{ } if \text{ } f(x,y)\leq T\end{cases} \qquad (3.1)$$

Here, g(x,y)=1, for the foreground pixels; and g(x,y)=0, for the background pixels. Figure (3.2) illustrates some results of the Binarization process.

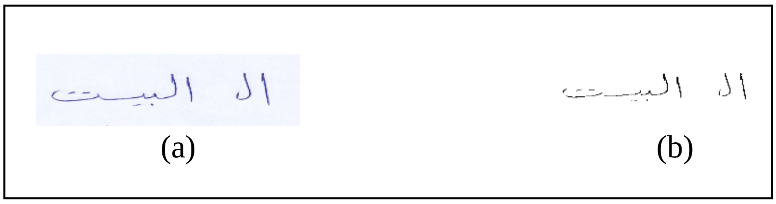


Figure (3.2) Binarization of word "ال البيت"

(a) Original image (b) the result of Binarization

B) Smoothing

The main goal of smoothing process is to reduce the noise (noise may be bumps in edge, or small gaps) from the scanned document images. Smoothing can fill the small gaps, or remove the small bumps in edges. In this work, the morphological operations have been used. The morphology consists of two main operations: dilation and erosion [Lei01]. Generally, dilation operation adds pixels to the bounders of object in an image, while the erosion operates reversibly. The number of pixel added or removed depends on the size of structure elements. We use the first operation dilation, which is represented in the following formula (3.2) [Hei98, Par97]:

$$A \oplus B = \left\{ z \,\middle|\, (\hat{B}_z) \cap A \neq \phi \right\} \qquad (3.2)$$

A by B is set of structure elements where ϕ is the empty set and B is structure elements. In other words, the dilation of A by B is the set consisting of all structure elements origin locations where the reflected and translated B overlaps at least one some of portion of A. Two types of dilation are applied: horizontal structure zones, and vertical structure zones. The result of applying the two types of dilation is shown in figure (3.3).

(a) (b)

Figure (3.3) smoothing word "ال البيت"

(a) The original image (b) the result of dilation operation

3.2.3 Segmentation

After the image is preprocessed, which yields smoothed image (noise free). The resulted image will pass through segmentation phase. In this section, we will segment the image into a set of meaningful regions.

The process of segmentation of Arabic handwritten image into homogenous and meaningful regions is considered one of the main obstacles due to the nature of Arabic language which differs from other languages in ligature, cursive, and other characteristics illustrated in chapter one. The segmentation phase is very important process. Any error committed in this process will directly lead to an error in the recognition phase.

In this section, we will show a developed segmentation process to divide the text into lines, then into words and then into fragments or characters. The suggested segmentation approach is composed of three main procedures, as shown in figure (3.4). The first step is labeling the connected components (CCs), where the connected parts in text image will be determined and refer to each of CCs with a specific number. The second step is feature extraction, where a number of features will be extracted (Area, Centroid, bounding box) from each CCs. The final step is segmenting the text which involves segmenting labeling components to characters or fragments and then order the segmented parts.

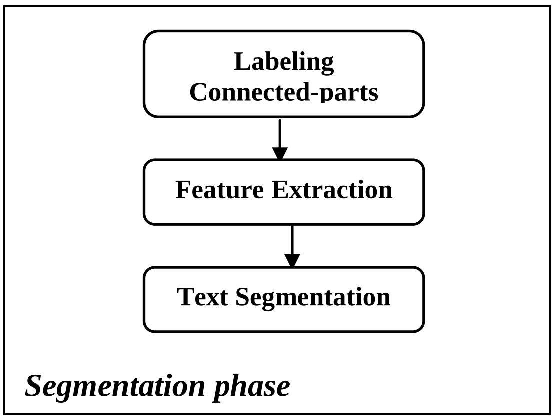


Figure (3.4) segmentation phase

A) Labeling connected Components (CCs)

Labeling CCs is an effective method of segmenting binary image by examining the connectivity of pixel with their neighbors and labeling the CCs. There are two practical methods which can be used for labeling CCs: recursive algorithm and sequential algorithm. In the suggested system, the sequential algorithm is used.

Sequential algorithm was designed by Rosefined and Platz in 1966 [Sha01]. Sequential algorithm is relatively simple to implement, and efficient in detecting the connected components. It is the most popular algorithm used with binary image. This algorithm is implemented in two steps.

- The first step is used to test connectivity through looking for foreground pixels. If it finds 8 connected neighboring pixels, it assigns them as a label. The output of the first phase is stored in equivalence table which contains the pixels connectivity (pixels are connected to other pixels) and the label number [Mar00]. The following conditions are used to determine the value of the label to be assigned to the current pixel:
    1. p(x, y) >1
    2. p(x-1, y) =1
    3. p(x-1, y+1) =1
    4. p(x-1, y-1) =1
    5. p(x, y+1) =1
    6. p (x, y-1) =1
    7. p(x+1, y) =1
    8. p(x+1, y-1) =1
    9. p (x+1, y+1)=1

Where the p(x, y) is scanned pixel, p(x-1, y) is the west pixel, p(x-1, y-1) is the south west pixel, p(x-1, y+1) is the north west, see figure (3.5) which shows the 8 connected neighbors.

P(x-1,y+1)	P(x,y+1)	P(x+1,y+1)
P(x-1,y)	P(x, y)	P(x+1,y)
P(x-1,y-1)	P(x,y-1)	P(x+1,y-1)

Figure (3.5) eight connected neighbors

- The second step is used to analyze the equivalence table to determine the final label of each provisional label. The main goal of this phase is to minimize the size of table. Figure (3.6) shows the implementation of the algorithm.

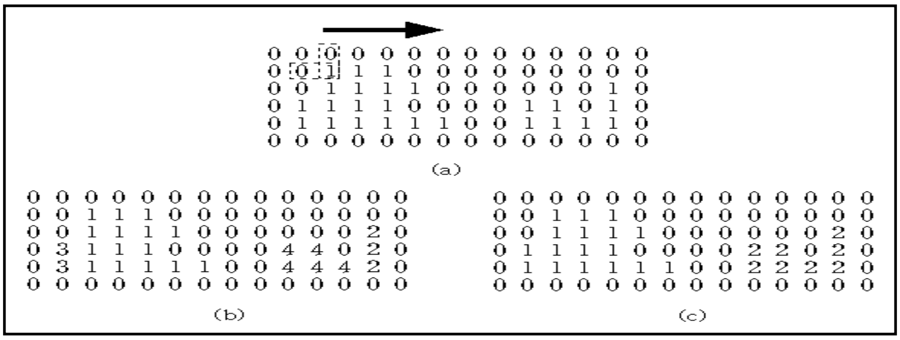


Figure (3.6) the result of applying labeling connected parts algorithm

Figures (3.6 a, b) show the first phase of the sequential algorithm. Figure (3.6.a) shows the scanning of image which is done row by row to determine the pixels which are connectivity while figure (3.6.b) shows the initial equivalence table and also shows there are 4 regions. Figure (3.6.c) shows the second phase where the equivalence is rebuilt to minimize the size of table. The final output of this algorithm is array containing labels for connected objects in image and the total number of connected objects found in the image.

B) Feature Extraction

Feature extraction is the process of extracting useful information from image. In this section, a number of properties of the labeled image will be extracted. These properties will be used for segmenting labeled image into characters or connected-parts. In order to a void making errors or losing the segmented parts, the segmented text will be rearranged using suggested sorting algorithm based on Bubble sort algorithm. I used the bubble sort algorithm because it has several advantages. First, the data is sorted in place so there is little memory overhead and, once sorted, the data is in memory, ready for processing. Second, it is simple to write and easy to understand. The algorithm is illustrated in figure (3.9). In this way, the text is divided sequentially. The extracted properties are Centroid, Area, and Bounding Box.

- Area: is the number of pixels in a region. The area of each region within image is calculated by counting the number of pixels in the labeled region. This is done by the following equation (3.3) [Ach05]:

$$Area = \sum_{i} \sum_{j} f\,[i\,,j\,] \qquad 3.3$$

  Where f [i,j] represents the pixels of labeled region. As an example for 3×3 pixels, the region area is 9 by applying the previous equation.

- Centroid: Centroid or (center of mass) is the point that is used to specify its position. The Centroid can be found by equation (3.4, 3.5) [Ach05]:

$$\bar{X} = \frac{1}{A} \sum_{i=1} x_i \qquad 3.4$$

$$\bar{Y} = \frac{1}{A} \sum_{i=1} y_i \qquad 3.5$$

  Where x_i and y_i is the coordinate of points in the region, $\bar{X}$ and $\bar{Y}$ are the region's Centroid, A is the area of the region.

- Bounding box: the bounding box is the smallest rectangle enclosing the object; the bounding box size is suitable to contain all object pixels. It is a very important feature because it can be used for the extraction of the object pixels for further reference purpose. The bounding box is defined by its coordinate's x-low, x-high, y-low and y- high, which are computed for each region [Pra07].

The properties (Centroid, area, bounding box) can be extracted by the following matlab function:

M = *regionpruops* (*L*, '*Basic*');

The *regionpruops* function is used for extracting properties from the labeled image. The string *Basic is* used to compute only Area, Centroid and bounding box [Math07]. *L* is denoting the labeled region. How does function work? For each labeled region in labeled matrix *L*, the basic properties are computed and stored it in the matrix *M*. The final output of this section is the matrix that contains each label with its properties.

C) Text Segmentation

In this section I will use an algorithm capable of segmenting the text to lines and then to characters or fragments. In fact, this algorithm is based on the output of the sections A and B. After segmenting the text, Bubble sort algorithm will be used to arrange the segmented parts in order to get a well-segmented text going with the original one.

.

The developed segmentation algorithm is shown in figure (3.7). The algorithm takes as input, L (number of labeled image), n (total number of label in image), and array I (which contain the extracted features). The algorithm from line (1-9) is used to segment the image to fragments or characters. Line (4) checks if the area of labeled part is greater than T_0 (Threshold value = 150, this value was chosen by trial), then increase K (number of segment) by 1. In line (6), `imcrop function is` used for segmenting the labeled part. This function takes the feature Bounding box of image and then cuts segment image and stores it in *Ic array*. In line (7), calculate the Centroid for each segmented image and store it in *cent.* Lines (8-12) are used to determine the line in segmented image. In line (8), we check the k value. In line (7), Centroid for each k is calculated, and then in this line (9) calculates the *variance* between two Centroid for k previous and current and store it in the *potential* variable. In lines (9-12), check the *potential* value if it is > T_1 (T_1 = 200, this value was chosen by trial), this means that k and k-1 are in different lines. Otherwise, it is in the same line.

```
Input: [L, n], L the labeled connected parts in image,
        n total number of labled connected parts.
        [I]: contain the features (Area, Centroid, boundinbox).
```

```
 K=0; //the number of segment in image.
Line =1 ; //number of line in image.
for s = 1:n
if((I(s).Area > T0))
  k = k+1;
  Ic = imcrop(I,I(s).boundingbox); //cut the image
  cent=I(s).Centroid;//store Centroid for segment image.
       if(k>1)
        potential = var(centerx(k-1:k));
      if(potential > T1)
          line = line + 1;
          k = 0;
      End
       End
            End
               End
```

Figure (3.7) segmentation algorithm

Here is an example of the application of the developed segmentation algorithm on the handwritten word محمد as shown in figure (3.8). Figure (3.8.a) shows the handwritten word محمد, whereas figure(3.8.b) shows the word محمد segmented to fragments and characters. As a result, we have got one fragment "محـ" and two characters "مـ ، د".

(a) (b)

Figure (3.8) word "محمد" : (a) original image word (b) segment word

By using the pervious algorithm, the image has been segmented to parts. Then the segmented parts will be arranged in order to get a well segmented text going with the original one. For this reason, the Bubble sort algorithm has been used to achieve this task. This algorithm depends on comparing the Centroid for currents segmented parts with the Centroid for previous segmented parts to accomplish text sorting, provided that the two parts should be available in the same line. Figure (3.9) shows the algorithm responsible for arranges segmented parts.

```
1- for i = 1:r
2-     for j = 1:c
3-         for l = 1:c-1

4-                 Ish = Isegments{i,l+1,1};
5-               if(~isequal(Ish,[]))
6-                     cent = Isegments{i,l,2};
7-                     cent2 = Isegments{i,l+1,2};
8-                     if(cent(2) > cent2(2))
9-                         Itemp = Isegments(i,l,:);
10-                          Isegments(i,l,:)=Isegments(i,l+1,:);
11-                          Isegments(i,l+1,:) = Itemp;
                       end
                 end
           end
     end
end
```

Figure (3.9) Bubble sort algorithm

In figure (3.8), the array *Isegments* contains three variables (i, l, 1) which represent the (line, column, Centroid). In line (4), store the array *Isegemnts* in Ish. Line (5), check if the array is not empty. Line (6, 7), store the Centroid, for current column and Centroid previous in the same line. Line (8, 11), the Bubble sort is comparing between two Centroid and swapping between them. The output is text arranged by ascending.

3.2.4 Recognition Phase

The recognition phase consists of two levels. The first level contains two neural networks to recognize the fragments and characters respectively. The second level is a decision maker.

Neural networks will be used for carrying out the task of recognition because of their efficiency and high ability to recognize the patterns. In this phase, the characters, and connected-parts, which have been segmented previously, will be recognized using two suggested neural networks. First, the neural network is trained for recognizing the Arabic characters only, whereas the second neural network is trained to recognize the fragments that are difficult to be segmented to characters due to the quality of ligature.

Once an image has been segmented, it will be fed to both neural networks at the same time in order to be recognized. Then, a decision will be made (if it should be recognized as character or as fragment) according to a declared rule.

3.2.4.1 Neural Network Level

Concerning the type of neural network used in the suggested system, two MLP are trained using Backpropagation algorithm, which has been illustrated in chapter two. The two suggested neural nets have different architectures (input, hidden, and output nodes), each is trained with its own parameters, and different training set.

The main point is that the image resulted from the segmentation phase is 40×70 pixels. The two nets have different numbers of input nodes (the character recognizer net has 36 input node, while the fragments recognizer net has 2800 node in the input layer). Since the resulted image is fed to both neural nets, the segmented image will be resized to 6×6 pixels before feeding it to the character recognizer neural net.

A) Artificial Neural Network (ANN) for characters

This neural net is designed to recognize Arabic handwritten characters. A multilayer neural network (MLP), which consists of three layers input, hidden and output layer is suggested.

The suggested feed-forward neural network consists of three layers:

- Input layer: constitutes of 36 neurons which receive pixels form binary image of size 6×6.
- Hidden layer: is the technique used in this work for choosing the number of hidden layers and the number of neurons which depends on trial and error basis. In this work, different networks were examined to reach good performance. It was found that the best number of hidden neurons is 12.
- Output layer: consists of 1 neuron, which indicates the character number (range 1-28 + unknown character represented by 29).

The NN consists of number of parameters that are needed to train the network. These are:

- TrainParam.epoches=600; maximum number of iteration for training procedures. The training stops if max number of iterations reached.
- TrainParam.lr = 0.2; learn rate, value to update weight.
- TrainParam.goal = 10e-10; the quality of network in terms of recognition defined as mean squared error between real output and desired output.

In this work, back-propagation algorithm (discussed in chapter two) is used to train the network. The back-propagation algorithm uses the transfer function. In this work, the used transfer function is Log-sigmoid (logsig) from input to first hidden layer, and Liner transfer function (purline) is used for output layer. Figure (3.10) shows the activation function.

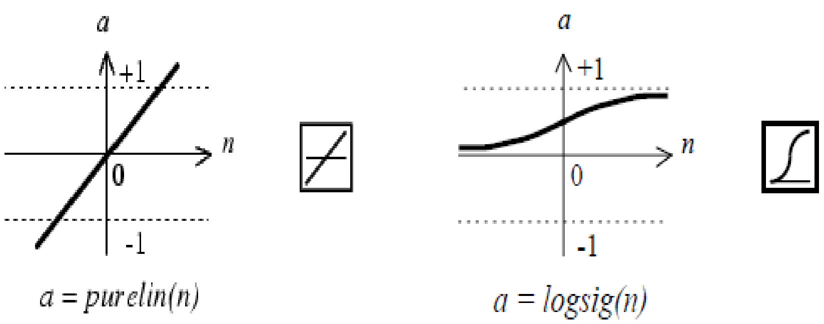


Figure (3.10) Transfer function [Math07]

(a) Pureline function (b) Log-sigmoid function

Figure (3.11) shows the structure of the suggested neural network used for recognizing the characters.

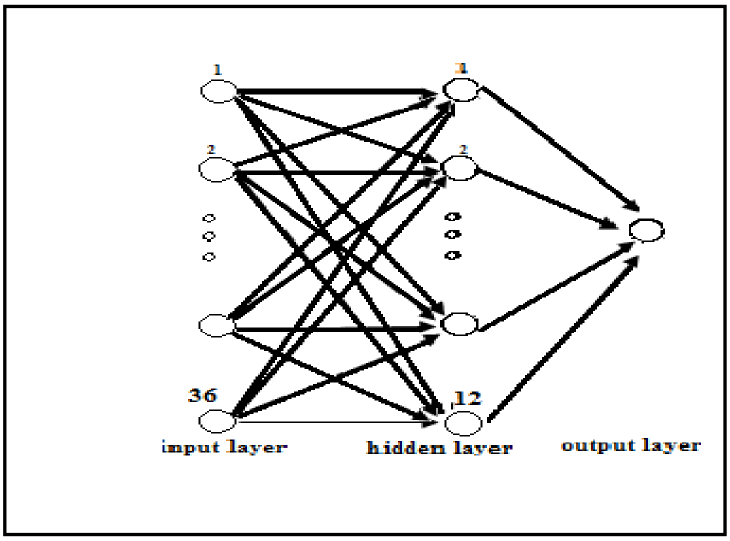


Figure (3.11) Structure of neural network for character recognition

B) Artificial Neural Network (ANN) for fragments

MLP neural network type is used for recognizing the fragments which are segmented previously. The segmented text (characters or fragments) will be input into MLP network for recognition. The image size input into the network is 40×70 pixels. The suggested neural network consists of four layers: input, two hidden layers and output layer.

- Input layer: consists of 2800 (40×70) neurons that represent the number of pixels in the input image (image segment of the fragments) that has been segmented in section (3.2.3).
- Two hidden layers: consists of 50 neurons, and 25 neurons respectively. The number of neurons in both the hidden layers has taken arbitrarily by trail and error method. Taking into account the factors of both accuracy and computation time.
- The output layer: consists of 4 neurons since we need to classify 4 fragments (في، ـمحـ، ـلح، ـلحـ) each of which corresponds to one of the possible fragments that might be considered.

In addition, a number of parameters are used to train the suggested neural network (NN), these are:

-TrainParam.epochs = 600; maximum number of iteration for training procedures (condition stop).

-TrainParam.goal = 0.1; condition stop.

-TarinParam.lr = 0.000001; learning rate.

The MLP network used here is trained in a supervised manner (the input and desired output are provided). In order to train the MLP network, we have used the backprogation algorithm which uses activation function. In this work, Log-sigmoid (logsig), and Tan-sigmoid (tansig) transfer functions have been used. In other words, activation function from input to first hidden layer is (logsig); from first hidden to second hidden is (tansig) and from second to output layer is (logsig). The difference between the Tan-sigmoid and Log-sigmoid is the output range. The Tan range is between (-1, +1) ,whereas the Log range is between (0, 1). Figure (3.12) shows the two activation functions (Log-sigmoid and Tan sigmoid function).

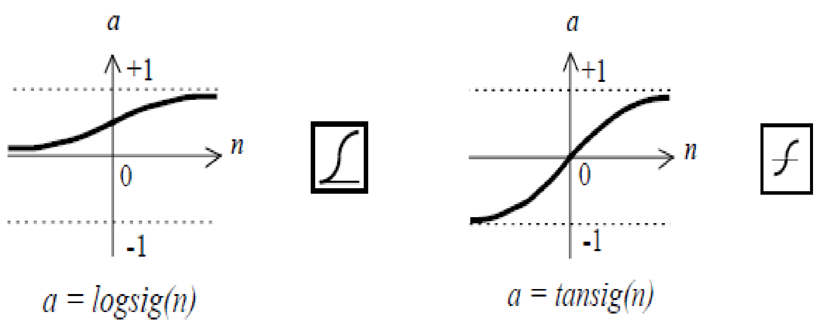


Figure (3.12) Activation function [Math07].

(a) Log-sigmoid transfer function (b) Tan sigmoid transfer function

Figure (3.13) shows the structure of feed-forward neural network used as fragments recognizer.

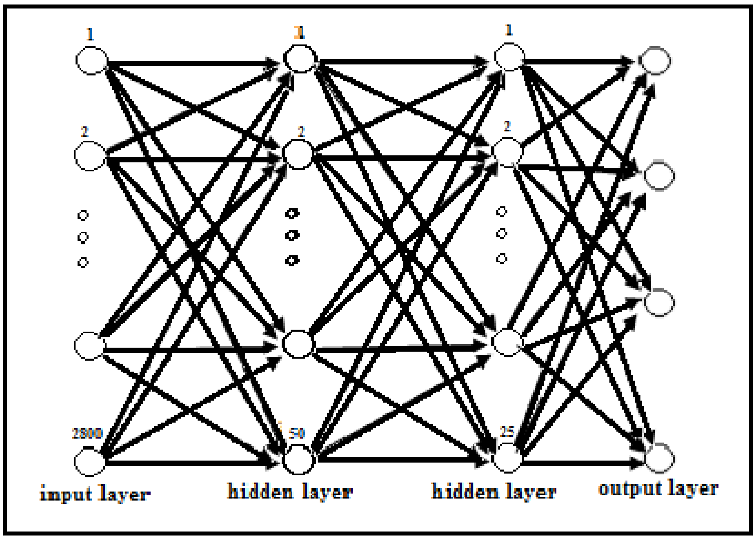


Figure (3.13) Structure of neural network for fragments

3.2.4.2 Decision Making Level

This level is concerned with decision making, in which the output of the two neural networks is considered. The decision is made to be based on the following rules:

- Step1: If the output of the character recognition neural net is unknown, then check the fragments recognizer.

  The segment is recognized based on neuron with the maximum value. (i.e. it is recognized as the section corresponding the maximum neuron).

- Step2: If the output of the character recognition neural net is a number *n* between (1-28), also check the output of the fragments recognizer:

  -If the maximum value output neuron is < T_{n2} (T_{n2} = 0.2, form the fragments neural) then, the decision is the character corresponding to the character that n represents.

  Else, the segment is recognized as unknown.

Chapter Four

Experimental Result

4.1 Introduction

In this chapter, we are going to present the experimental results of the used system. In other words, the results obtained by the algorithms used in segmentation will be pointed out. In addition, I will point out the results obtained by using the neural networks for recognizing both Arabic characters and connected-parts. To study the behavior of the suggested system, three types of data have been collected.

- The first type of data has been used for two purposes: testing the segmentation algorithm, and obtaining results concerning strengths and weaknesses of segmentation process.
- The second type of data has been only assigned to training and testing the neural network associated with characters, and to investigate the aspects of strength and weakness of recognition process.
- The third type is used to train the neural network concerning recognizing fragments to investigate the strengths and weaknesses of the proposed system. Here the data has been divided into training data and testing data.

This chapter is composed of two main parts: the result of segmenting algorithm, and recognition of Arabic characters and fragments.

4.2 Segmentation

In the previous chapter, the developed algorithm has been presented with the purpose of segmenting the handwritten Arabic document to lines and then to connected parts or characters. In fact, the previously mentioned algorithm consisted of three basic stages: Labeling connected parts, features extraction and segmentation.

The process of segmentation has been dependent on the concept of labeling connected components because this method has several advantages which have made it different from other methods: this method has the ability of dealing easily with the text containing (skew, slope) as it does not take into account the slope of the text or any other factors in the process of segmentation. In fact, this method is concerned with the existence of pixels and in turn helps to get rid of the skew and slope problem resulted from the wrong scanning of an image. Another advantage is that this method is fast in the process of segmentation. Moreover, it depends on testing the neighbors.

4.2.1 Data set

The data was collected from 15 people. Each person wrote 15 Arabic documents in handwriting. Each handwritten document is different from other handwritten documents in size and in the way of handwriting. Moreover, the words of handwritten documents are different. Each document consists of 5 lines. Each line has about 10 or 12 words. We have got about 50-60 words in each document. But the total number of the handwritten words collected from the subjects is nearly 800 words. These words have been experimented by algorithm in order to be segmented. Figure (4.1) illustrates the sample of the collected data.

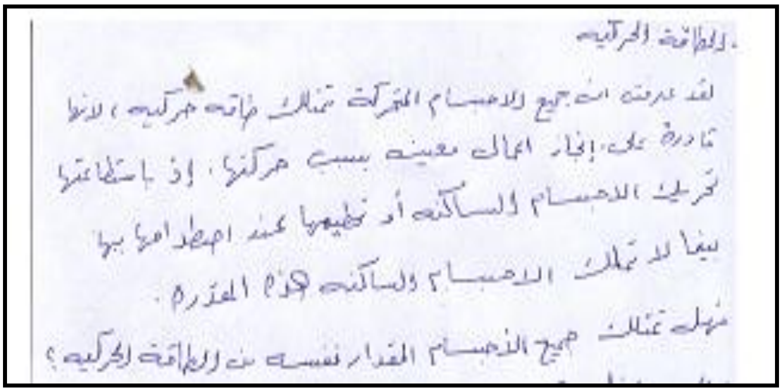


Figure (4.1) sample of collected data

4.2.2 Document Segmentation to Lines

This section discusses the obtained results through applying the algorithm to the target samples. It also indicates the ability of this algorithm to segment the Arabic text handwritten to lines perfectly well. The result obtained after applying it to all the collected documents is 100%. This means there was no problem in segmentation process of documents to lines. In turn, this ratio indicated the extent of efficiency of the applied algorithm in segmenting the text to lines. The result gained from this algorithm is better than the results of Zahour and et-al [Zah01], which have total accuracy 97% in segmenting the document to line; also it is better than Tripathy and Pal [Tri04] which have total result 60%. One of the problems that [Zha01, Tri04] faced, is the existence of skew in the text. In general, the algorithm developed in this work did not face problems because it dealt with labeling connected parts rather than projection profile. In projection profile, any skew in the document lines will highly affect the accuracy of the segmentation process, while in the developed connected part approach, any skew in the document lines will not affect the segmentation process since it depends on pixel connectivity.

4.2.3 Segmenting Line into Connected-Parts or Characters

In the previous section, I illustrated the use of the developed algorithm in segmenting a text into lines, whereas this section presents the results obtained from segmenting the line to connected-parts or isolated characters. Table (4.1) shows the results of segmenting lines to isolated characters or connected-parts. Having a look at table (4.1) indicates that the total ratio obtained is 97.4%.

Table (4.1) Result of segmenting line into connected parts or isolated characters

Number of document	Result %
Doc1	90
Doc2	98
Doc3	96
Doc4	98
Doc5	99
Doc6	99
Doc7	99
Doc8	99
Doc9	96
Doc10	100
Doc11	96
Doc12	97
Doc13	96
Doc14	99
Doc15	99
Ratio	97.4%

In fact, document (10) has got the best results, while document (1) got the worst result. The weakness of document (1) lies in the existence of touch point between the isolated characters and connected parts. For example, figure (4.2) shows the touch point between isolated and connected parts.

Figure (4.2) the word "الله"

We also noticed that there was an existence of pixel connected between the character (أ) and the character (ل).

4.2.4 Segmenting Connected-Parts into Characters or Fragments

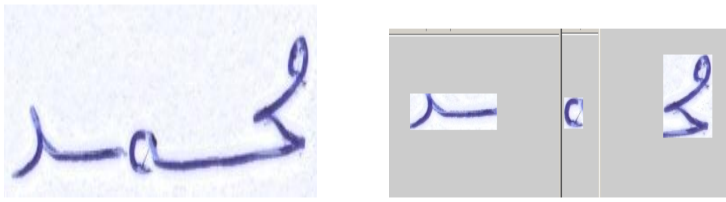
After having presented the results of segmentation lines to characters or connected-parts, we will discuss the results of segmenting connected parts into characters or fragments. In the process of segmentation, I took into consideration the existence of any disconnect in the connected parts even if it is one pixel. In this case, the connected parts will be segmented to characters. Table (4.2) shows the results of this stage. When looking at the table (4.2), it is clearly seen that the total ratio reached about 26.7%. This can be due to the nature of Arabic handwriting which is difficult to be divided into characters. Another reason is the existence of cursive.

Table (4.2) Result of segmenting connected parts into characters or fragments

Number of document	Result %
Doc1	40
Doc2	18
Doc3	20
Doc4	15
Doc5	23
Doc6	15
Doc7	25
Doc8	41
Doc9	30
Doc10	10
Doc11	20
Doc12	42
Doc13	38
Doc14	39

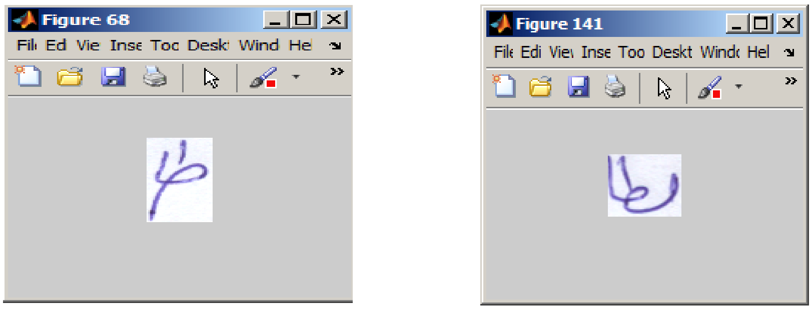
Doc15	25
Ratio	26.7%.

Figure (4.3) illustrates the best result obtained from segmenting the connected parts to characters. The figure also shows that the word (محمد) is divided into characters, whereas figure (4.4) shows the failure in the process of segmenting connected parts to characters. This can be due to the absence of any disconnection event if it is one pixel in the connected word.



(a) (b)

Figure (4.3) segmenting connected parts into characters



(a) (b)

Figure (4.4) failure of segmentation character.
(a) connected part("طا"), (b) connected part("لطا")

In this section we illustrated the result obtained through experimenting the collected data set when applying the developed algorithm. It is noticed that the process of segmenting the text to lines got the best result while the process of segmenting the connected parts to characters got the worst result. Based on this, the applied algorithm achieved a total ratio of 74.5% as shown in the previously mentioned tables.

4.3 Recognition Accuracy

In this section, I will discuss the results obtained from using neural networks for recognizing Arabic text handwritten either it is characters or parts. As mentioned in chapter three, two different neural nets are used to recognize isolated character and fragments respectively. The next two sections will discuss the recognition behavior of the two nets.

4.3.1 Character Recognition Neural Networks

In the previous chapter, the structure of the suggested character recognition neural net was illustrated. In this part, the dataset used to train and test the suggested neural net will be discussed. It also presents the results obtained from the application of the neural networks.

The data set was collected from 15 people who were asked to write characters. Each of them wrote 28 Arabic characters four times. As a result, each character has 60 images and the total number of the images of characters to which the neural networks will be applied has become 1680 images. These images represent all Arabic characters used for training and testing the net. The data set is divided into:

- Training part: 90% of data set was assigned to training the neural network on different shapes of characters. The reason for this is that Arabic characters have different shapes in handwriting; each person wrote characters differently and even the same person wrote the characters differently. Therefore, the total number of images has become 1512 images for training network.
- Testing part: 10% of the data set was assigned to testing the ability of neural network to recognize the characters. As a result, the total number of images has become 168 images. Each image represents a character in different shapes.

After having discussed the shapes of the data set used for training and testing neural network, I will present the ability of neural network to recognize each character of handwritten Arabic characters. Table (4.3) illustrates the recognition ratio of each character.

It is noticed that the highest recognition ratio was for (أ). This can be attributed to the non-existence of a character similar to (أ) shape. This helps the neural networks to recognize it easily. As for the remaining of Arabic characters, the ratio ranged between 68% - 80%. This is due to the existence of similarity among these characters in spite of the difference in the position of dot. An example of that is the characters (ن،ب،ط،ظ). Therefore, the neural network faced a difficulty in recognizing these characters.

It is also clear that the lowest ratio was the recognition of the character (ع). It reached 50%. This could be attributed to the overlapping of (ع) with characters (غ،أ). The evidence appeared in the process of testing where the character (ع) was recognized as (أ) because of the existence of (ء) above the character (أ).

Table (4.3) Result of the Recognition of Arabic characters

Characters	Recognition ratio %
أ	88
ب	74
ت	66
ث	70
ج	82
ح	68
خ	78
د	70
ذ	76
ر	64
ز	68
س	84
ش	76
ص	78
ض	74
ط	73
ظ	70
ع	50
غ	66
ف	77
ق	73
ك	62
ل	80
م	77
ن	74
هـ	78
و	68
ي	74
Ratio	72.8%.

After illustrating the recognition ratio of each Arabic character shown in table above, it can be said the total ratio of recognizing Arabic characters reaches 72.8%.

4.3.2 Fragment Recognition Neural Network

In this part the data set used for training and testing the neural network will be presented. In addition, the results obtained from testing the data set will be discussed.

The data was collected from 15 writers. Each writer wrote 4 Arabic fragments 15 times. Each connected part has 225 images. Training the neural network was limited to four fragments (لحـ،في،لج،محـ). Therefore, the total number of data set was 900 images. Every set of 225 images represents a fragment of these four fragments. The researcher divided the data into two parts:

- Training part: 90% of the data set was assigned for this part. As a result, the total number of the images of fragments to be trained by the neural network is (810) images.
- Testing part: 10% of the data set was assigned for this part. As a result, the number of images to be tested is 90 images.

The ratio of recognizing each fragment of these four fragments is illustrated in table (4.4). It is clearly seen that the highest ratio of recognition was for the fragment (في) because of the non-existence of fragment similar to this fragment. On the other hand, the lowest ratio was between the two fragments (محـ،لحـ). This is because of the existence of similarity between these two fragments.

Table (4.4) Result of the Recognition of fragments

Connected parts	Recognition rate%
لج	80
محـ	72
لحـ	70
في	85
Ratio	76.75%

After having declared the recognition ratio of each fragment, the total ratio of recognizing fragment is 76.75%

The total ratio of both neural nets for recognizing both isolated characters and fragments reached about 74.77%. The resulted recognition ratio is good. Generally, when comparing the ratio of recognition that the applied system has reached with those of other systems it can be said that it achieved the best ratio in some cases. For instance, when compared with Muhammad Sarfraz and et-al [Sar 03] who had total recognition ratio 73%, it can be concluded that our system has got better results. Moreover, when compared our system with Somay Alm'addeed [Alm 06] who developed a system for recognizing handwritten words and got about ratio 63% it has become obvious that our results exceed the results of her study by about 11%. This is due to the weakness of her system which was based on the *global features* for recognizing Arabic handwritten words. In fact, this is very difficult since we can't rely on these *global features* for recognizing the handwritten words because handwriting varies from one person to another. On the contrary, if these features were applied to one type of printed fonts, the results would be better.

Chapter five

Conclusion and future works

5.1 Introduction

In this work, we have worked to develop an algorithm capable of segmenting the Arabic documents into lines, words, then to connected-parts or characters. After document segmentation, neural network was developed to recognize the resulted connected-parts and characters. Two neural networks were developed to improve the recognition accuracy: network designed for characters recognition, and a network designed for recognizing connected-parts. To evaluate the performance of the developed system, 15 handwritten documents were used for both training and testing. This chapter illustrates the main points observed by me as well as the recommended future works.

5.2 Conclusion

I faced several difficulties in both document segmentation phase and recognition phase. Some of these difficulties are:

1. I have tried to solve noise problem available in the scanned image by using preprocessing which helped to reduce the noise by depending on Morphology method which filled the small gaps or removed the small bumps in edge by dilation and erosion. This, in turn, made both of the segmentation and recognizing processes easy
2. It is very hard to separate the words into characters. As well known, segmenting the words into characters depends on the silent area (straight) separating the characters. In Arabic, especially in handwriting, sometimes 2 or 3 characters are written without any silent area between them. Therefore, the output of the word segmentation is either character or connected word. This problem forces me to use two different recognizers, one for fragments, and one for characters.
3. The document segmentation process is divided into 3 segmentation processes: line segmentation, word segmentation, characters or fragments segmentation. The developed system could:
    - Perfectly perform line segmentation because our segmentation process depended on the pixel existence rather than the pixel density.

- Word segmentation reached 97.4%. This good result can be attributed to the fact that I depend on the pixel connectivity for segmenting the word rather than the projection profile which depends on the pixel density.
- Segmenting words to characters or fragments recorded the worst results. It reached 26.7%, because of the nature of Arabic language (which is called full cursive language) and the dependence on the existence of any cut in a word (even it is one pixel).

4. The big difference in handwriting is that a writer does not leave enough space between words and isolated characters. I have dealt with this problem through depending on the features (bounding-box, area, Centroid) in the process of segmentation which contributed somewhat in solving this problem.
5. A very serious problem lies in the fact that an Arabic character can be written differently by the same person. For instance, the character alef (أ) can be written like (أ،ا) by the same person. Furthermore, this problem is considered one of the biggest dilemmas faced by the neural network for recognizing characters. Therefore, I did my best to solve this problem to some extent by increasing the size of the study sample, and by training the neural network using different forms of the same character as well as using noise characters during the training phase.
6. One of the dilemmas is the problem of ligature. This problem is the result of Arabic language features from other languages. Therefore, I used another particular neural network for recognizing the fragments because of the difficulty of dividing the ligature into characters. The recognition rate for fragments is 76.75% on average, while the recognition rate for characters is 72.8. This difference in accuracy is due to the big conflict between the 28 Arabic characters, especially the characters with dots above and below the letter which makes the recognition process very difficult. An example of this is ((,(خ, ح, ج) , بـ , يـ ,نـ) (ش, س) (أ,عـ) etc).
7. What characterizes the segmentation method, used in this work, is that it is not affected by the presence of skew in the scanned image. This skew could be attributed to either wrong scanning of document or a way of handwriting. It is clear that the used method here does not rely on the so-called pixel density which is used in projection profile method, but it depends on the existence of

pixels regardless the errors resulted from input method. This, in turn, facilitated preprocessing the scanned document in order to correct the skew angle.

5.3 Future works

1. In this work, I have worked to recognize four fragments only, in future I will work to increase the number of these fragments to include all Arabic fragments, where I will take all the Arabic fragments that don’t contain dots such as (لح،محـ،بحـ،،لمحـ،بمـ،تم...etc). If the user enters a fragment that contains a dot he determines if the dot is above or below then, he will enter the fragment to the system which recognizes it without dots. After the process of recognition, the system will ask if the dots are above or below to be recognized correctly.
2. In this work, I have inserted the image pixels as input to the neural network. I will work in future to use a number of features (Hu moments Invariants, Zernike moments, and Legendre moments) rather than insert the image pixel in order to recognize them, which probably will help to increase the rate of recognition of neural network of characters and fragments.
3. In this work, I have relied on label connected components algorithm to segment the document to characters and fragments but the percentage was low. In future, I will integrate label connected components algorithm with projection profile in order to segment the document into characters and fragments to improve the percentage of segmentation.

References

[Abd 07] Shubari A.Abdullah, **"Off-line Handwritten Arabic Characters Segmentation Using Slant-Tolerant Segment Features (STSF)",** Master Thesis, University Sains Malaysia, April 2007.

[Ach 05] Tinku Acharya, Ajoy K.Ray, **"Image Processing: Principles and Application"**, John Wiley & Sons Inc., 2005.

[Ali 08] Abdurazzag Ali Aburas and Mohamed E.Gumah, "**Arabic Handwriting Recognition: challenge and solutions**", International on Symposium on Information Technology(ITSim),Kuala Lumpur Malaysia , August 2008, pp. 1-6.

[Alk 09] Jawad H AlKhateeb, Jinchang Ren, Jianmin Jiang, Stan Ipson,"**A Machine Learning Approach For Classifying Offline Handwritten Arabic Words**", 09. International conference on CyberWorlds,2009, pp.219-223.

[Alm 06] Somaya AlMa'adeed , " **Recognition of off-line handwritten Arabic words using neural network**" , proceeding of the geometric modeling and imaging- new Trends, August 2006, pp.141-144.

[Alm 08] Somaya Al-Ma'adeed , Eman Mohammed , Dori Al Kassis , Fatma al Muslih ,"**Writer Identification Using Edge-Based Directional Probability Distribution Feature For Arabic Words**", IEEE/ACS International Conference on computer systems and application, 2009,pp. 582-590.

[Ami 06] A.Amin, N.Al-Darwish," **Structural Description To Recognizing Had-Printed Arabic Character Using Decision Tree**", International Journal of computer and Applications, Volume 28 Issue 2, April 2006. pp. 129-134.

[Ari 98] Nafiz Arica, "**An offline character Recognition System For free Style Handwriting**" , Master Thesis, the middle East Technical university, September 1998.

[Bai 04] A.Rauf Baig, "**Spatial- Temporal Artificial Neurons Applied To Online Cursive Handwritten Character Recognition**", ESANN-Europen Symposium on Artificial Neural Networks, Bruges (Belgium), Aprial 2004, pp. 561-566.

[Baz 99] Issam Bazzi, Richard M.Schwartz, John Makhoul, "**An Omni font Open-Vocabulary OCR system for English and Arabic pattern Analysis And Machine Intelligence**", IEEE Transaction on Pattern Analysis and Machine Intelligence, volume 21 Issue 6, Jun 1999, pp. 495-504.

[Blo 91] Dan S.Bloomberg, "**Multiresolution Morphological Approach to document Image Analysis**", presented in international conference on document analysis and recognition Sanit-Malo France, October 1991, pp.963-971.

[Bou 06] Faouzi Bouchareb, Mouldi Bedda and Salim Ouchetati, "**New Preprocessing Methods For Handwritten Arabic Word**", Asian Journal of information Tecnology, volume 5 Issue 6, 2006, pp. 609-613.

[Bro 08] Salama Brook, Zaher Al Aghbari, "**Classification of Persona; Arabic Handwritten Document**", WSEAS, Trums action on information science & application, Volume 5 Issue 6, Jan 2008, pp. 1021-1030.

[Cha 02] Watchara Chatwiriga, "**Offline Thi Handwriting Recognition In Legal Amount**", PhD thesis , West Virginia university , 2002.

[Cha 04] Abdolah Chalecale, Goshah Naghdy, Prashan Premaratne,Alfred Mertins, "**Cursive Signture Extraction And Verfication**", In proceeding 2nd international workshop on information technology & its disciplines (WITID 2004), 2004, pp. 109-113.

[Elg 01] Ahmed M.Elgmmal, Mohmaed A.Ismail, "**Techniques for Language Identification for Hybrid Arabic-English Document Images",** proceeding of the sixth International Conference on Document Analysis and Recognition,2001, pp. 1100-1104.

[Fau 94] L.Fausett, "**Fundamental of Neural Network**", Printice-Hall, International Inc, 1994.

[Gar 09] Naveen Gary, "**Handwriting Gurumukhi character Recognition Using Neural Networks**", Master Thesis, THAPAR university, June 2009.

[Hei 98] Henk J.A.M Heijmans, Jos B.T.M Roerdink**, "Mathematical Morphology and Its Applications to Image and Signal Processing",** Kluwer publishing, 1998.

[Hul 98] Jonathan J.Hull, "**Document Image Skew Detection Survey Annotated Bibliography**", Appeared in Document Analysis System II, Ricoh California Research Center what conference, pp.40-64, 1998

[Kav 06] Ergina Kavallerdctou and Efstathios Stamatatos, "**Improving the Qulity of Degraded Document Images**", proceeding of the second international conference on Document Image Analysis for Libraries (DIAL 06) ,Lyon, 2006, pp. 10-349

[Ker 07] Zsolt Kertesz, Bence Kovari, "**Offline Signature Verification Using Feature Based Image Registration**", IEEE, 11th International conference on intelligent engineering system (INES), 2007, pp.93-94.

[Kla 01] Tim Klassen, "**Towards Neural Network Recognition Of Handwritten Arabic Letters**", Master thesis, Dalhousie University, 2001.

[Kul 05] Adrian Kuha, "**Using Local Slant Correction to Normalize Handwritten Text Samples**", http://scg.unibe.ch/archive/projects/Kuhn03a.pdf, university of Bern, December 2005.

[Lei 01] Neucimer J.Leite, Silvio J.Gumaraes, "**Morphological Residues and a general framework for Image filtering**", EURASIP Journal on Applied Signal Processing- Nonlinear signal and image processing, Volume 2001 Issue 4, December 2001, pp. 219-229

[Mar 00] Stephane Marchand-Maillet, Yazid M. Sharaiha, **" Binary Digital Image Processing: A Discrete Approach",** Academic Pr, 2000.

[Math 07] Matlab online support, www.mathworks.com acess helpdesk techoc Matlab.shtml.

[Moz 05] Saeed Mozoffari, Karim Faez, Majid Ziaratban, "**A Hybrid Structural/ Statistical for Handwritten Farsi/ Arabic Numeral Recognition**", MVA2005 IAPR conference on machine vision application, Tsukuba science city Japan, 2005, pp.218-221.

[Pra 07] William K. Pratt, "**Digital Image Processing**", PIKS Scientific Inside,4th edition, 2007.

[Par 97] J.R.Parker, "**Algorithm for Image processing and computer vision**", Jhon Wiley and sons Inc , 1997.

[Pet 04] Peter Burrow, "**Arabic handwriting recognition**", Master thesis, University of Edinburgh, 2004

[Rum86] D.E. Rumelhart, G.E Hinton, and R.J Williams, Leaning Internet. Representation by Error Propagation in Rumelhart,D.E. and McCelland, J. L, **"Parallel Distributed Processing":** Explorations in the Microstructure of cognition. MIT Press, Cambridge Massachusette, 1986.

[Ros 66] Azriel Rosenteld and John L.Pfaltz, "**Sequential operation in digital picture processing**", Journal of the ACM (JACM), volume 13Issue 4 .October 1966, pp. 471-494.

[Sar 03] Muhammad Sarfraz, Syed Nazim Nawaz , Abdulaziz Alkuraidly ,"**Offline Arabic text recognition system**", International conference on geometric modeling and graphics (GMAG'03), 2003, pp.30-35.

[See 02] Torsten Seemann, "**Digital image Processing using Local Segmentation**", phd thesis, in Monash university, April 2002.

[Sha 01] Linda Shapiro, George Stockman, **" Computer Vision",** Prentice Hall, 2001.

[Sha 08] Z.Shaaban," **A new recognition scheme for machine printed Arabic texts based on neural network**", PWASET, volume 31, July, 2008 ,pp.707-710.

[Sou 04] Labiba Souici Meslati , Mokhtar Sellami ,"**A hybrid approach for Arabic literal amount recognition** ", Arabian journal for science and engineering, volume 29, October, 2004, pp. 177-194.

[Tay 02] Yong Haur Tay, "**Offline Handwriting Recognition using Artificial Neural Network and Hidden Markov Model**", PhD, Teknologi Malaysia, 2002.

[Tri 95] Oivind Due Trier, Anil K.Jain, "**Goal- directed Evaluation of Binarization methods**", IEEE Transaction on pattern Analysis and machine Intelligent, volume 17 Issue 12, December 1995, pp.1191-1201.

[Tri 04] N. Tripathy and U. Pal. ,"**Handwriting Segmentation of Unconstrained Oriya Text**" IWFHR Ninth International Workshop on Frontiers in Handwriting Recognition, 2004, pp. 306–311.

[Upp 07] Deepthi Uppalapati, "**Integration of offline and online signature verification systems**", Master thesis, Indian Institute of technology, Kanpur, July 2007.

[Vas 06] Ravinder Kumar Vashisht ,"**Recognition Pre-Segmented Handwritten Character in Gurmukhi Script using Features Based on Zernike Moments and KNN Classifier** ",Master thesis ,PUNJABI University,2006.

[Vel 10] C.M.Velu, P.Vivekanandan, "**Automatic letter sorting Indian Postal Address Recognition System based on PIN codes**", Journal of internet and information system, volume 1 Issue 1, June 2010, pp.6-15.

[Zah 01] A.Zahour, B.Taconet, P.Mercy, and S.Ramdan, "**Arabic Hand-written Text-line extraction**", in proceeding of the Sixth International. Conference on document Analysis and recognition, ICDAR 2001 , Seattle, USA, September 10-13 2001, pp. 281–285.

[Zur96] J.M.Zurada, "**Introduction to Artificial Neural System**",JAICO publishing House, 3rd edition,1996.

ملخص البحث

تهدف هذه الدراسة بصورة رئيسة لإنتاج نظام قادر على تقسيم النصوص العربية المكتوبة بخط اليد إلى أحرف أو إلى أجزاء مترابطة بالإضافة إلى تمييزها باستخدام الشبكات العصبونية. ومن الواضح إن النجاح في عملية تمييز النصوص العربية سيؤدي إلى تحسين التواصل ما بين الإنسان والحاسوب وجعل الحاسوب أداة أكثر تفاعلية. ولإيجاد نظام قادر على تمييز النصوص العربية بشكل دقيق، لا بد من إيجاد طريقة قوية قادرة على تقسيم النص بشكل صحيح.

ويتكون نظامنا بشكل رئيسي من أربع مراحل: إدخال الصورة، المعالجة الأولية (التنعيم، الترميز)، التقسيم، والتمييز.

- إدخال الصورة : يتم إدخال النص المكتوب بخط اليد إلى الحاسوب باستخدام الماسح الضوئي.
- المعالجة الأولية : تهدف إلى تحسين الصورة من خلال استخدام التنعيم الذي يعمل على إزالة الشوائب و ملىء الفراغات ومن ثم استخدام الترميز الذي يعمل على تحويل الصورة الممسوحة إلى صوره ثنائيه.
- المرحلة الثالثة التقسيم حيث تم تطوير طريقة جديدة للتقسيم، تتكون مرحلة التقسيم من ثلاث خطوات: تحديد الأجزاء المترابطة، استخراج عدد من الخصائص من الأجزاء التي تم تحدديها، و تقسيم الأجزاء المحددة إلى أحرف أو أجزاء.
- المرحلة الرابعة التمييز، وتتألف هذه المرحلة من مستويين : يتكون المستوى الأول من شبكتين عصبونيتين: واحده لتمييز الأحرف وأخرى لتمييز الأجزاء . أي جزء يتم تقسيمه يتم إدخاله إلى الشبكتين اللتان تعملان بشكل متزامن من اجل تمييزه. المستوى الثاني وهو اتخاذ القرار حيث يعمل هذا الجزء على تحديد مخرجات الشبكتين العصبونيتين ومن ثم اخذ أعلى نسبة تمييز بين الشبكتين.

في هذه الدراسة لقد ركزت على موضوعين، الأول أنه عملت على تطوير خوارزمية قادرة على تقسيم النصوص العربية إلى أحرف أو أجزاء. والأمر الثاني أنه قامت ببناء شبكتين عصبونيتين الأولى لتمييز الأحرف وأخرى لتمييز الأجزاء التي يتم تقسيمهما من قبل خوارزمية التقسيم. لقد قمت باختبار هذا النظام على عدد من الأشخاص الذين طلب منهم كتابة عدد من النصوص العربية بخط أيديهم. وخضعت هذه النصوص للمعالجة الأولية التي عملت على تحسين الصورة ومن ثم التقسيم وأخيرا

التمييز. ولقد سجلت طريقة التقسيم التي طورتها نسبة تقسيم بلغت (74.5%). في حين وصلت نسبة التمييز إلى (74.77%).