Democratic and Popular Republic of Algeria

Ministry of Higher Education and Scientific Research

University Ahmed Draia - Adrar

Faculty of Sciences and Technology

Department of Mathematics and Computer Science

جامعة أحمد دراية.أدرار-الجزائر
Université Ahmed Draia.Adrar -Algérie

Thesis Presented to Fulfill the Partial Requirement for Master's Degree in Computer Science

Option: Networks and Intelligent Systems

Title

Recognition of Printed Arabic Script Using Support Vector Machine

Prepared by

Ms. Yamina Ouled jaafri

The Jury Members:

Mr. CHGUEUR Djilali	President
Mr. Mamouni El mamoun	Supervisor
Mr. DEMRI Mohammed	Examiner1
Mr. Kaddi Mohammed	Examiner2

Academic Year 2016/2017

Abstract

Arabic Optical Character Recognition (AOCR) is the science of conversion Arabic text image documents of type printed, or handwritten into editable text. OCR role is to help or replace humans in computerizing paperwork in order to accelerate, improve and reduce cost as well as time and effort. It provide although the ability to electronically editing, storing more compactly and searching documents. It is not a recent research field; it had started about 40 years ago.

The need for it has become increasingly urgent due to overcrowding paperwork in our societies. So a lot of research conducted on AOCR as the Arabic script language is the mother language of over quarter of the world population despite this fact, robust and reliable performance AOCR system is still challenge.

In this thesis, a proposed Arabic Character dataset generated for evaluating and testing feature extraction systems purpose. a combination of statistical features has been proposed to increase the recognition accuracy, using SVM classifier.

For text recognition, a novel segmentation approach for machine Arabic printed text for different segmentation stages; line segmentation, word and sub-word segmentation, and character segmentation based on profile projection techniques are proposed.

Keywords: AOCR, Segmentation, Classification, SVM, Statistical features.

Dedication

To my father and my mother, who showered me with their support and showed me unconditional love.

To my brothers and sisters who have not ceased to support me and be examples of perseverance, courage and generosity.

To my family, my nieces, my nephews, and my friends.

I dedicate this research to the University of Ahmed Draia, to the MI department, my professors, my classmates and the working agents.

Acknowledgements

First and foremost, I give my deep thanks to Allah for giving me the opportunity and the strength to accomplish this work.

I would like to thank my supervisor Mr. Mamouni El Mamoun for his help and support during my work for creating a research environment that has been very inspiring. This work would not have been possible without his guidance, encouragement and motivation. He deserves my respect and thanks.

My respect and gratitude to the members of the jury who made me the honor of judging this work and their availability, observations and reports that have enabled me to enrich my work.

Thanks to all the teachers of the faculty of Sciences and Engineering Sciences.

I thank all my colleagues' promotion 2016/2017.

All those who participated directly or indirectly to the realization of this work.

Table of Contents

Chapter 1: Definitions and basics

Chapter 2: Stat of art of segmentation

Chapter 3: Support vector machines

Chapter 4: Results and analyses

List of Figures

List of Tables

List of Abbreviation

AOCR:	Arabic Optical Character Recognition.
CRR:	Character Recognition Rate.

General introduction

Optical Character Recognition (OCR) science has been created to help people editing their scanned documents. Recently, it has been used in different applications like automatic data entry (for scanned documents), Automatic number plate recognition, Assistive technology for visually impaired people, and many other fields. OCR has been started with the invention of retina scanner in 1870, unfortunately this can't be said about machine printed Arabic text recognition.

Some of the reasons may be taken to the lack of research comparing to Latin, Chinese and other languages, but, at the same time the nature of the Arabic script even in its printed form poses real challenges to researchers.

This thesis aimed to enhance the optical printed Arabic characters recognition accuracy across using a combination of statistical features, accuracy is evaluated using CRR (Character Recognition Rate) metric. The results show that the proposed statistical features achieved average CRR of **99.08%**, and for text recognition, a novel segmentation approach for machine Arabic printed text for different segmentation stages; the CRR for test recognition using the combination of statistical features, Support Vector Machines as classifier and the proposed segmentation approach achieved **91.90%.**

This report consists of four chapters. Chapter one includes a brief introduction to the AOCR as well as Arabic OCR System. Chapter two gives detailed working principle of some method of segmentation. In chapter three we present the Support Vector Machines classifier used in our AOCR system, we give principle general of working support vector machine and their mathematical basis and also we have exposed the two cases of the Separable and non-separable data. Chapter four presents our application using the Matlab language programming, as well as the results of the proposed system. Finally, we present a conclusion of our complete work study with a clear future plan, which we aim to help future researchers to continue with this subject.

1. Introduction:

This chapter presents general short introduction about Arabic language, Arabic language properties and also introduction about Arabic optical character recognition this will include detailed information about AOCR types as well as its general system processes.

2. Arabic Language:

Arabic characters are used in many languages as it is the first language for more than 400 million people in the world. It is also used by more than triple the previous number of Muslims all over the world as a second language, as it is the language in which the Holy Qur'an was revealed.

Arabic was added to the official languages of the United Nations in 1973 as the sixth language. The other five official languages (Chinese, English, French, Russian and Spanish) were chosen when the United Nations was founded. Also, as has been reported by National Geographic, Arabic is expected to be one of the 5 major languages by 2050.

Arabic is one of the Semitic languages. Languages like Arabic, Urdu, Persian (Farsi), Sorani and Luri dialects of Kurdish, Jawi, Pashto, Sindhi, Hausa, Kashmiri, Kazak, Kyrghyz, Malay, Morisco, Pashto, Punjabi, Tatar, Turkish, and Uyghur use Arabic characters with some little differences see [1].

2.1. Arabic Language Properties:

Arabic language properties can be listed as [2] [3]:

a. Arabic language consists of 28 characters (see Table 1.1).

b. **Multiple grapheme cases:** As mentioned before, Arabic language script is cursive (connected), this characteristic cause the characters to be context sensitive and change its form and have multiple variants according to its position (Start, Middle, End, Isolated) (see Table 1.1).

Table 1. 1. Arabic language primitives.

Character name		**Arabic Language Primitives**			
		Isolated	Connected		
			End	Middle	Start
Alif	**ألف**	أ	ـا	-	-
Baa	**باء**	ب	ـب	ـبـ	بـ
Taa	**تاء**	ت	ـت	ـتـ	تـ
Thaa	**ثاء**	ث	ـث	ـثـ	ثـ
Jeem	**جيم**	ج	ـج	ـجـ	جـ
Haa	**حاء**	ح	ـح	ـحـ	حـ
Khaa	**خاء**	خ	ـخ	ـخـ	خـ
Daal	**دال**	د	ـد	-	-
Thaal	**ذال**	ذ	ـذ	-	-
Raa	**راء**	ر	ـر	-	-
Zaay	**زاي**	ز	ـز	-	-
Seen	**سين**	س	ـس	ـسـ	سـ
Sheen	**شين**	ش	ـش	ـشـ	شـ
Saad	**صاد**	ص	ـص	ـصـ	صـ
Dhaad	**ضاد**	ض	ـض	ـضـ	ضـ
Ttaa	**طاء**	ط	ـط	ـطـ	ط

Dthaa	**ظاء**	ظ	ـظ	ـظـ	ظـ
Ain	**عين**	ع	ـع	ـعـ	عـ
Ghen	**غين**	غ	ـغ	ـغـ	غـ
Faa	**فاء**	ف	ـف	ـفـ	فـ
Qaf	**قاف**	ق	ـق	ـقـ	قـ
Kaf	**كاف**	ك	ـك	ـكـ	كـ
Lam	**لام**	لا	ـل	ـلـ	لـ
Mem	**ميم**	م	ـم	ـمـ	مـ
Noon	**نون**	ن	ـن	ـنـ	نـ
Haa	**هاء**	ه	ـه	ـهـ	هـ
Wow	**واو**	و	ـو	-	-
Yaa	**ياء**	ي	ـي	ـيـ	يـ

c. The Arabic language is written from right to left.

Figure 1.1 shows the Arabic printed sentence: "Alhoma Sali Ala Sidina Mohammed (اللهم صلي وسلم على سيدنا محمد) ".

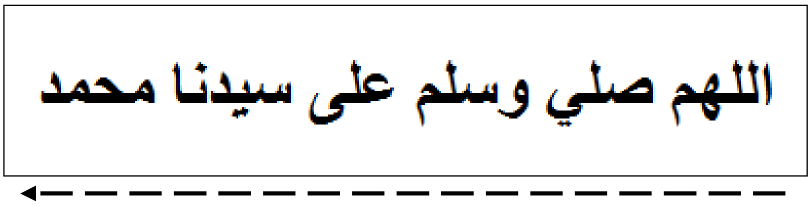


Figure 1. 1. Arabic Read/Write direction.

d. Dots:

Arabic characters consist of two parts:

- **Grapheme:** it is the main structure of the character. Multiple characters share the same grapheme (see Figure 1.2).

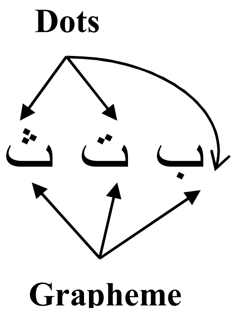


Figure 1. 2. Dots characteristic in Arabic characters.

- **Dots:** many characters can share the same grapheme but not the same number of dots. There are 15 characters in the language have dots, (10) characters have (1) dot, (3) characters have (2) dots and (2) characters have (3) dots, as it showing Table 1.2.

Table 1. 2. Dots count and existence in Arabic characters.

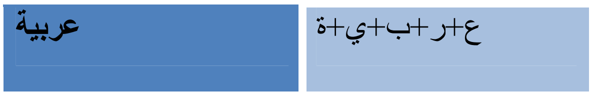
Number of dots	Character name	
1	Baa	ب
	Jeem	ج
	Khaa	خ
	Thaal	ذ
	Zaay	ز
	Dhaad	ض
	Dthaa	ظ

	Ghen	غ
	Faa	ف
	Noon	ن
2	Taa	ت
	Qaf	ق
	Yaa	ي
3	Thaa	ث
	Sheen	ش

e. **Connectivity:** Unlike Latin language, Arabic characters is connected (cursive) within the same word, this connection can be interrupted at the middle of the word at few certain characters (أ، د، ذ، ر، ز، و) see Table 1.3.

Table 1. 3. Characters connectivity and interrupting.

Word	Primitives
متصلة	م+ت+ص+ل+ة
أصوات	أ+ص+و+ا+ت
بذل	ب+ذ+ل
إزاء	إ+ز+ا+ء
تاريخ	ت+ا+ر+ي+خ
أبجدية	أ+ب+ج+د+ي+ة



f. **Ligatures:** Depending on font type, many characters can be compound together at certain positions in the word, and represented by a single atomic grapheme which is called Ligatures. e.g. lamalif (لأ) is a combination of lam (ل) and alif (أ) see Table 1.4 below.

Table 1. 4. Ligatures.

Ligature	Characters
لم	ل+م
لا	ل+ا
لحـ	ل+حـ
لجـ	ل+جـ
ممـ	م+مـ

g. **Diacritics:** Diacritics used for correct and standard pronunciation and called (Tashkyl), as it's appear in Figure 1.3.

بِسْمِ اللّهِ الرَّحْمَنِ الرَّحِيمِ

Figure 1. 3. Diacritics.

h. **Sub word(s):** some Arabic words are composed of sub-word(s) or pws (piece of words). Exists if the middle of the word contains one of following characters **(ذ، أ, د**

ر، ز، و). As example see Figure 1.4, the word (فأسقيناكموه) contains four sub-words or pws: (فأ,سقينا,كمو,ه).

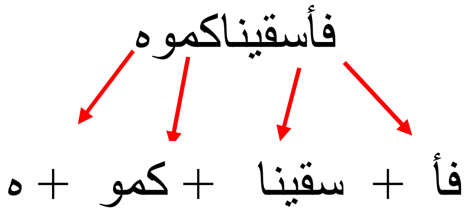


Figure 1. 4. Sub words or PWS.

3. Optical Character Recognition (OCR):

Character recognition is a process of identification of printed, typewritten or handwritten characters, which converts image of printed, scanned text into a text understandable by machine.

The characters are optically scanned and converted into machine editable form. It is an important field of image processing and artificial intelligence. It is widely used as a form of data entry from books, research articles, bank forms, office papers or any other form of printed records. It is also used to store the old and decaying written materials like handwritten books, research documents, old manuscripts etc. [5].

4. Arabic OCR type:

The character recognition process generally falls under two categories (Figure 1. 6):

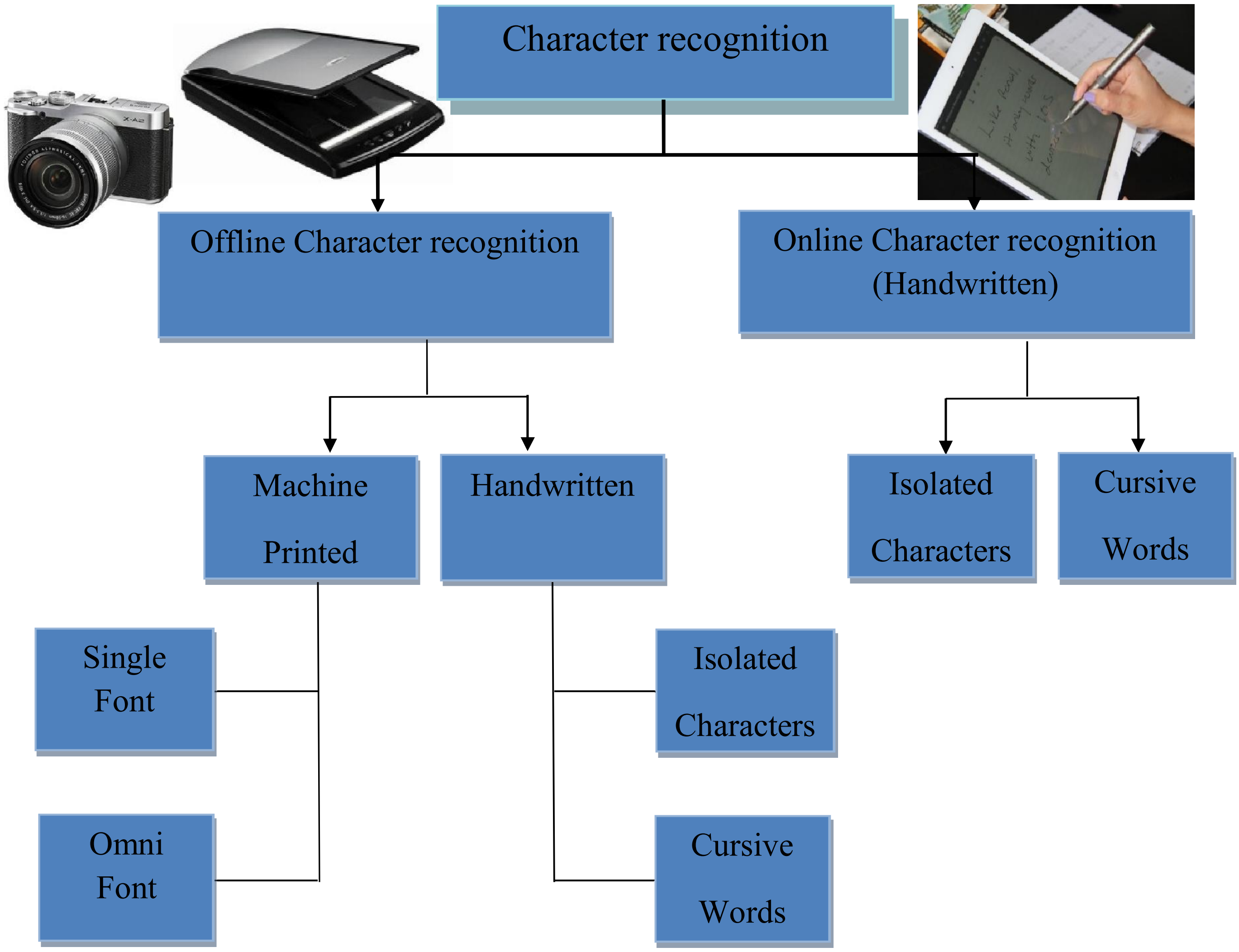


Figure 1. 5. Types of character recognition [16].

4.1. Online Character Recognition:

Online character recognition refers to the real time acquisition and recognition of characters. Usually an optical pen is used for writing characters. The characters are recognized at the same time we write with the optical pen and then displayed on computer.

Online character recognition is also called handwriting recognition because it recognizes the characters written by hand through optical pen [4].

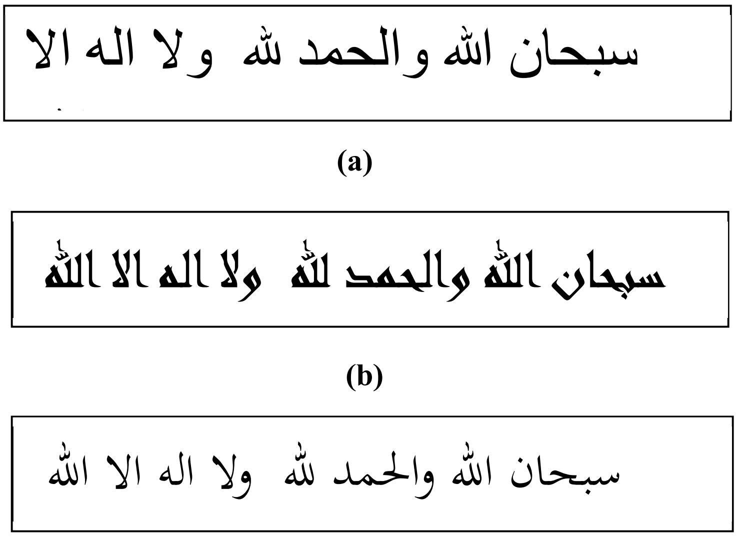
4.2. Offline Character Recognition:

Offline character recognition refers to the recognition of characters which are present on the sheet of paper. The paper is scanned through a digital device like scanner or camera and the image is stored in the computer. The scanned image is then used in the recognition process from which specified characters are recognized [4].

Offline character recognition is further divided into two major categories: Machine Printed and handwritten.

4.2.1. Machine Printed (Type Written): where, the characters or words are written by a machine like computers under well-known font types. It's classified into two categories:

- **Single font:** The document is written only in one type of font.
- **Omni-font:** The document is written in more than one type of font.



سبحان الله و الحمد لله و لا اله الا

(a)

سبحان الله والحمد لله ولا اله الا الله

(b)

سبحان الله والحمد لله ولا اله الا الله

(c)

Figure 1. 6. Example of printed Arabic script (a): Times New Roman Font, (b): Andalus Font, (c): Traditional Arabic Font.

4.2.2. Handwritten: where, the text is written manually by a human. This type is assumed to be the most difficult style because of the variations in character shape even if it is rewritten by the same person.

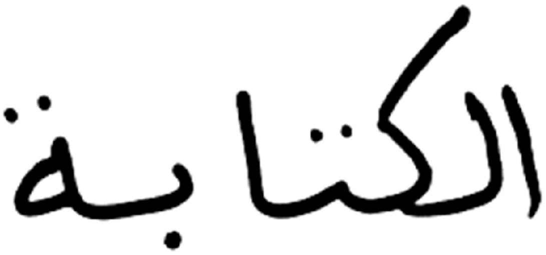


Figure 1. 7. Example of Arabic handwriting.

5. Arabic OCR System:

The general AOCR system processes according to recent researches like [7] and others, can be listed as shown in below to:

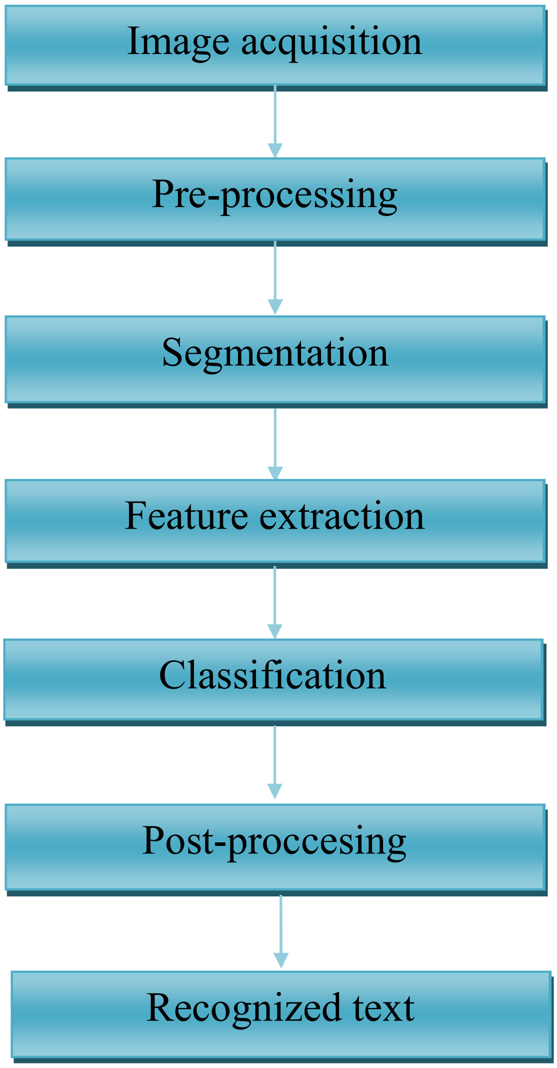


Figure 1. 8 . General AOCR System processes.

5.1. Image Acquisition:

The process of obtaining the images through some specialized devices is known as the image acquisition. There are two most common and relatively inexpensive devices, digital camera and scanner, are used for the image acquisition [6].

5.2. Image Pre-processing:

The preprocessing is the most important task to transform input image into most appropriate and suitable format to achieve the noise free and clean image.

The goal of image preprocessing is to remove some imperfections and noise for the image enhancement, and image improvement quality can be obtained by applying possible image processing algorithms [6].

5.2.1. Binarization:

Refers to transform the RGB color of the text image in black/white color. It is the initial step of most document image analysis and understanding systems. Usually, it distinguishes text areas from background areas.

Binarization plays a key role in document processing since its performance affects quite critically the degree of success in a subsequent character segmentation and recognition.

The resultant binary images has values of 0 each for all the foreground black pixels and 1 each for all the background white pixels [18].

5.2.2. Filtering:

Noise may occur during writing and scanning processes. There are different types of noise like salt and pepper noise, which affects the recognition rates. So it is very important to deal with such noise. Filtering is a technique for modifying or enhancing an image. Median, mean filters are example for removing noise in scanned documents [17].

5.2.3. Normalization:

Normalization is the process where the image size reduces without change in structure or shape of an image [17].

5.3. Segmentation:

Systems differ in requiring this process as a main process, according to that segmentation can be classified into two types:

a. Global Segmentation Approach which is also called segmentation free or holistic approach in which the segmentation just required for dividing the text into words,

b. Analytical approach, in which the page is divided into lines, the lines is divided into words and the words is divided into its primitive characters [8].

5.4. Feature Extraction:

The recognition of characters depends on the differences between its features.

Which can be classified to [8]:

- Structural or topological features: these features depend on the geometrical information of the characters. Some of these features are (convexities, concavity, end points, number of holes, etc.).
- Statistical features: obtained from the arrangement of points constituting the character matrix. In contrast to topological features it is less affected by distortions or noise. These features can be (zoning, moments, projection histograms, n-tuples, distances, outlines and crossings, etc.).
- Global transformations: depend on the transformation schemes which converts the pixel representation of the pattern to another representation which enable to discriminate between characters e.g. (Direction codes, Hough transform, Walsh transform, Fourier descriptors, etc.).

5.5. Classification:

This process and Extracting features are considered the pivot elements in the recognition processes. In this process, the extracted features compared with other features of previously known characters (model) to find the closest match.

This process is a major task after feature extraction to classify the object into one of several categories.

There are a number of various classification techniques applied in text recognition:

5.5.1. K-Nearest Neighbors:

The algorithm k-nearest neighbors [9] (KNN) is among the simplest artificial learning algorithms based on similarities. The basic idea when classifying a given observation is to vote its nearest neighbors in the sense of a predefined distance. The class of the new observation is then determined by the majority among the k nearest neighbors.

5.5.2. Support Vector Machine:

The foundations of SVM originated from early concepts developed by Cortes and Vapnik [10], this method has proven to be very robust for general classification and regression, [11, 12].

Support vector machines are based on two key ideas: the notion of maximum margin and the concept of kernel function. In linear classification, they create a hyper plane that separates the data into two sets with the maximum margin [13]. For the cases where the data are not linearly separable, they map the data representation space into an area of larger dimension in which it is probable that there is a linear separator.

5.5.3. Neural Network Classifier:

The conception of this method is very schematically inspired from the function of biological neurons. They receive the signals (electrical impulses) through highly branched extensions of their cellular body (dendrites) and they send the information through long extensions (axons) [14].

The algorithm ANN learns a model by means of a feed forward neural network trained by a back propagation algorithm. A neuron is primarily a mathematical operator. It performs a weighted sum, followed by a nonlinear function. This function must be bounded, continuous and differentiable, the most frequently used ones are sigmoid functions [15].

5.5.4. Template Matching:

This approach [16] this is one of the simplest approaches to patter recognition. In this approach, a prototype of the pattern that is to be recognized is available. Now the given pattern that is to be recognized is compared with the stored patterns. The size and style of the patterns is ignored while matching.

5.6. Post-Processing:

OCR post-processing goal is to increase the probability that OCR hypotheses are correct, and they are compatible with the language constraints imposed by the task. The Language Model conform from these constraints and can be as complex as an unconstrained sentence like the natural language or as simple as a small set of valid words.

6. Conclusion:

OCR has attracted an immense research interest not only because of the very challenging nature of this problem to shorten the reading capabilities gap between machines and humans but also because it improves human machine interaction in many applications.

In this chapter we have presented the Arabic language properties and also general system processes. The general system processes consist of Image Acquisition, Image Pre-processing, Segmentation, Feature extraction and finally Classification.

In the next chapters, we will discuss the detailed description of the system processes applied to the Arabic printed recognition system.

1. Introduction:

Optical character recognition has been the subject of considerable research activity for many years, segmentation is a very important step in the process of character recognition, and has a major impact on the quality of the recognition system.

An error produced in this step can be very serious, since it affects the performance of the recognition systems carrying the character to be either misrecognized or in most cases rejected completely.

The human being can easily segment the Arabic word into characters; however, it is not easy to segment it directly into perfect characters by the computer.

We present in this chapter the segmentation methods based on the techniques used, the most frequently cited in the literature.

2. Levels of Text Segmentation:

Text image segmentation can be achieved at three levels [19] [20] [22], Segmentation at any of these levels directly depends on the nature of the application. The various levels in the hierarchy are as shown in Figure 2.1.

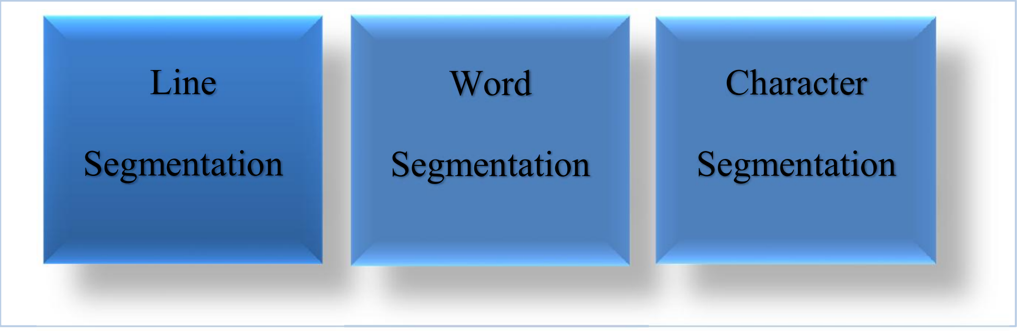


Figure 2. 1. Levels of segmentation.

2.1. Line Segmentation:

Line segmentation is the first and a preliminary step for text based image segmentation, to separate the text lines, from the document image, the horizontal projection profile of the document image is found. The horizontal projection profile is the histogram of the number of black pixels along every row of the image. White space between the text lines is used to segment the text lines [21].

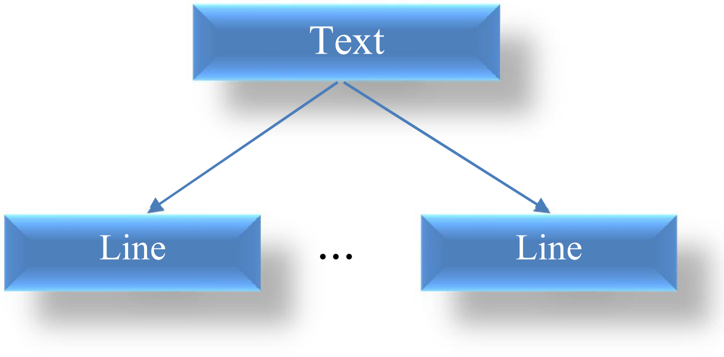


Figure 2. 2. Segmentation of text into lines [23].

2.2. Word Segmentation:

Word segmentation is the next level of segmentation, the spacing between the words is used for word segmentation. The segmentation of the words is found by taking vertical projection profile of an input text line. Vertical projection profile is the sum of black pixels along every column of an image [21].

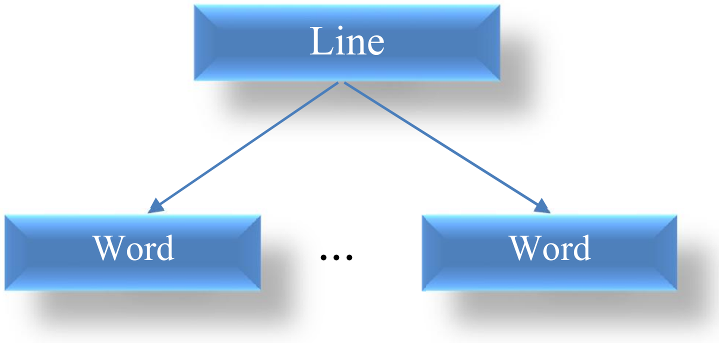


Figure 2. 3. Segmentation of lines into words [23].

2.3. Character Segmentation:

Character segmentation is the final level for text based image segmentation, character segmentation is the decomposition of an image into sub images, which only contain a single character. It is a critical step in most OCR systems, and typically the cause of a high proportion of OCR errors.

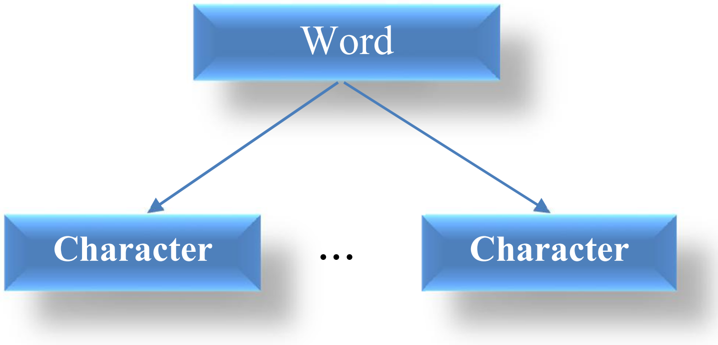


Figure 2. 4. Segmentation of word into characters [24].

3. Segmentation:

The OCR systems classified into according to the existence of the segmentation into two strategies which have been applied to printed and handwritten Arabic character recognition global and analytical segmentation approach [25]:

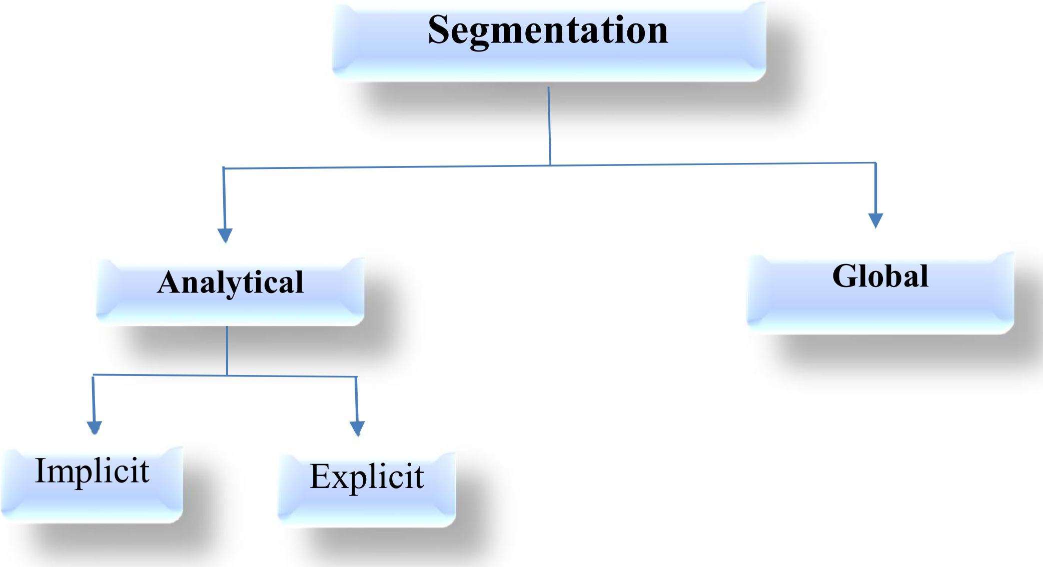


Figure 2. 5. Segmentation strategies.

3.1. Global Segmentation Approach:

Global Segmentation Approach which is also called segmentation free or holistic approach, in this approach, the text is segmented only into lines and words, no need to extract the characters or strokes, the recognition step is applied directly. This method usually uses a lookup dictionary contains all sub-words possibilities to do the recognition so it suitable for limited number of words that perform an enumeration like the number and cities' names, however, written text must be segmented using more advanced methods such as [2] Hidden Markov Models (HMMs), Artificial Neural Networks (ANNs) methods.

3.2. Analytical Approach:

In this approach, the text is segmented into small parts that perform characters and strokes that the words or sub-words in the text consist of. This method requires more processing but it gives better results since it can cover most of characters possibilities, this makes the extraction feature and recognition steps easier.

Analytical segmentation approaches are divided into two categories: implicit segmentation and explicit based segmentation [26]:

3.2.1. Implicit Based Segmentation Approach:

Generally, in this approach segmentation and recognition of characters are achieved at the same time so the characters are segmented while being recognized

This type of segmentation is usually designed with rules that attempt to identify all the character's segmentation points. The basic principle of recognition-based character segmentation is to use a mobile window of variable width to provide the tentative segmentations which are confirmed (or not) by the classification.

3.2.2. Explicit Based Segmentation Approach:

In the explicit segmentation (or dissection segmentation), words are explicitly or externally segmented into characters which are then recognized individually.

4. Segmentation Methods:

The segmentation of the Arabic word into individual characters is a crucial step in recognizing printed Arabic text. Most of the recognition errors arise from segmentation errors.

Many methods are proposed for Arabic OCR character segmentation and they are classified into: projection profile methods, character skeleton based methods, contour tracing based methods, template matching based methods, morphological operations based methods.

4.1. Segmentation Methods Based On Vertical Horizontal Projection:

Methods based on projection profiles are usually used for lines and words, sub-words segmentation, when a clear gap is found between them, horizontal projection is used for line segmentation and vertical projection is usually used for word and sub -word segmentation.

The aim of the projection method is to simplify drastically a system of character recognition by reducing two-dimensional information into one dimension, these methods are based on the fact that the connection stroke is always of less thickness than other parts of the words [27] [28].

- **The horizontal projection:**

The horizontal projection is defined as:

$$h(i) = \sum_i P(i,j) \ldots\ldots\ldots\ldots\ldots\ldots\ldots (1)$$

- **The vertical projection:**

The vertical projection is defined as:

$$h(j) = \sum_j P(i,j) \ldots\ldots\ldots\ldots\ldots\ldots\ldots (2)$$

Where $P(i,j)$ is the pixel value which is either zero (white or background) or one (black), i,j refer to rows and columns respectively.

The horizontal projection is useful in separating the lines and finding the text baseline, while the vertical one helps in segmenting the words, sub words and characters.

Figure 2.6 shows the vertical projection profile of sentences, while Figure 2.7 shows the horizontal projection profile the longest spike represents the baseline.

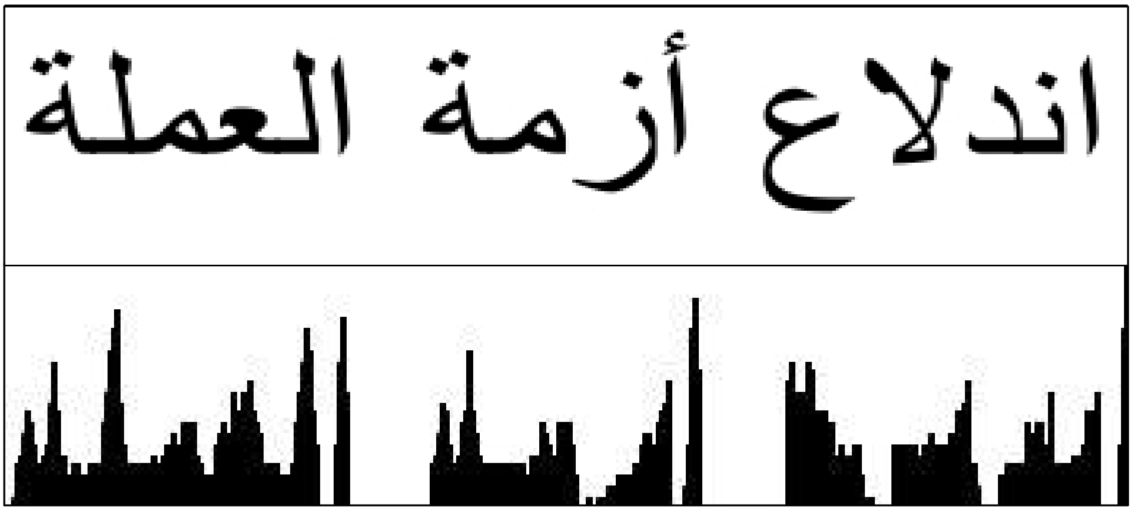


Figure 2. 6. Example of vertical projection profile.

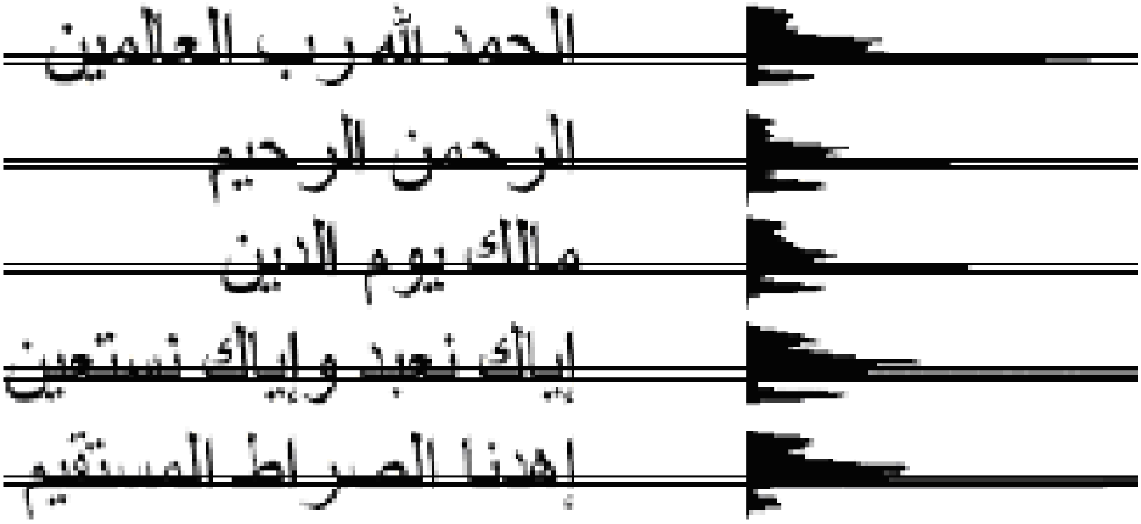


Figure 2. 7. Example of horizontal projection profile.

4.2. Segmentation Methods Based On the Thinned Characters (character skeleton):

The thinning operation means producing the skeleton of the image. A skeleton is a one pixel width produced by highlighting the centerline of the word. It helps in restoring the essential information about the word. This method is highly noise-influenced. And in many cases the shape of the character differs from the original one and this makes the segmentation process more difficult [26] [29].

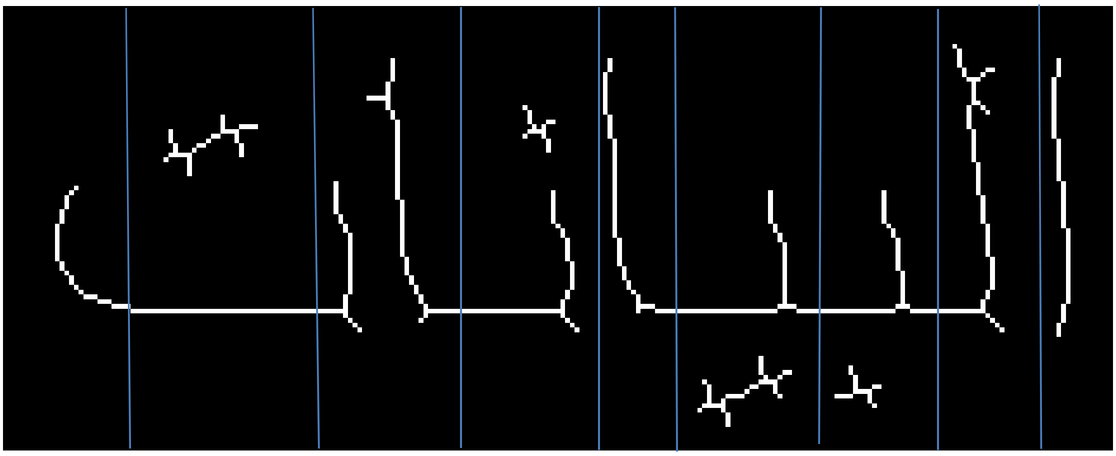


Figure 2. 8. Binary image skeleton extraction.

4.3. Segmentation Methods Based On Contour Tracing:

In this method, the pixels that form the outer shape of the character or word are extracted, researchers used many ways to determine the cutting points on the contour, in general contour based methods avoid the problems appear in the thinning because it depends on extracting the structure of the word which gives a clear description for it, but they are affected by the noise, so some enhancements also should be applied [29].

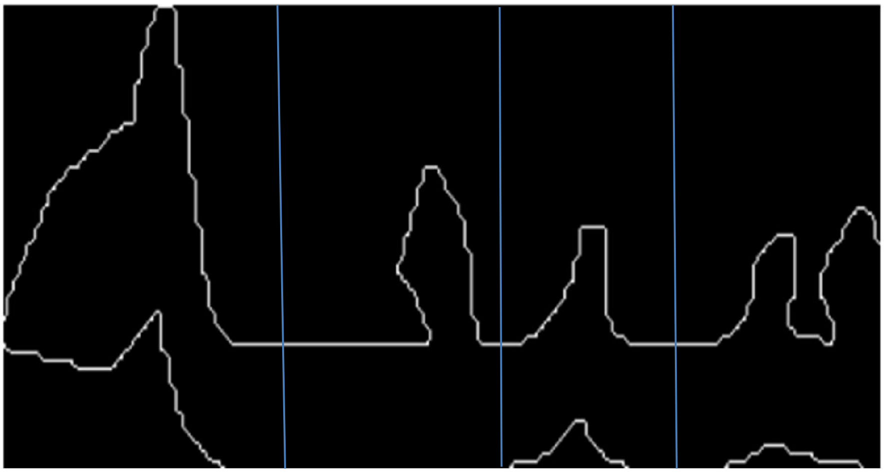


Figure 2. 9. Binary image contour extraction.

4.4. Segmentation Methods Based On Morphological Operations:

In this method, morphological operations are used for segmentation, usually closing followed by opening operations are applied, this method is not an independent method because other techniques should be used beside for segmentation, little number of researchers used this method [26].

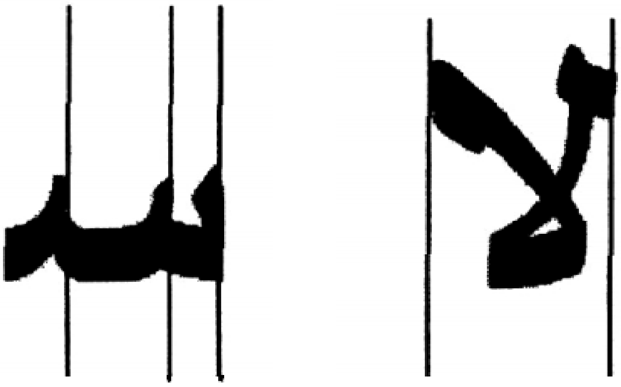
4.5. Methods Based On Template Matching:

In this method, usually a sliding window slides over the baseline is used, if any match is noticed then the center pixel in the sliding window is considered as the cutting point, the problem in this method is if the cutting point locates under the baseline, a segmentation failure will be occurred [26].

5. Various Challenges During Segmentation[21]:

- There can be variation in shapes and writing styles of different writers.
- Cursive nature of Arabic writing i.e. the characters in a sub word written connected to each other.
- Characters can have more than one shape according to their position inside the word image.

- Some characters can have similar contours.
- Most characters share similar shape with others. The position or number of dots in the character makes the only difference.



Over-Segmentation Under-Segmentation

Figure 2. 10. Common Problems in Arabic Character Segmentation.

6. Conclusion:

In any Optical Character Recognition (OCR) system the segmentation step is usually the essential stage in which an extensive portion of processing is devoted and a considerable share of recognition errors is attributed.

Segmentation is one of the most important steps in any recognition system; there exist two approach of segmentations, Global and Analytical Segmentation Approach**.**

In this chapter we have tried to expose the different methods used in the segmentation, it aims at decomposing the text into lines, words, sub-words, and characters.

Character segmentation process introduces the most serious problem in the development of off-line Arabic character recognition system.

1. Introduction:

SVM was first proposed by V.Vapnik, which is a pattern recognition method developed from statistical learning theory.

SVM is a popular machine learning method for classification, regression, and other learning tasks. Originally, SVM was a technique for building an optimal binary (2-class) classifier. Later the technique was extended to regression and clustering problems. SVM is a partial case of kernel-based method.

It can map feature vectors into a higher-dimensional space using a kernel function and build an optimal linear discriminating function in this space or an optimal hyper-plane that fits into the training data.

The purpose of this chapter is to provide an introductory tutorial on the basic ideas behind Support Vector Machines (SVMs).

2. History of Support Vector Machine:

Data classification is one of the main data mining tools used to automatically classify data records into a certain number of categories. Machine learning has proven to be a solid ground for building data classification models [31].

Support vector machines (SVMs), a leading tool in machine learning, are especially suitable for classification. SVMs were first introduced by Vapnik et al. [30] in 1975 more than two decades ago, and have been intensively studied ever since. They have gained the attraction of many researchers and scientists in the machine learning community due to the desired properties they exhibit such as guaranteed convergence, sound theoretical basis, good generalization capabilities, and efficient handling of high-dimensional data.

3. Support Vector Machines:

3.1. Optimal Hyperplane for Linearly Separable Patterns case:

Support Vector Machine is used for classification and Regression. It is a strategy of separating the samples by just drawing a decision boundary known as hyper plane in case of linear classification.

For two classes of given examples, the goal of SVM is to find a classifier that will separate the data and maximize the distance between these two classes. With SVM, this classifier is linear called "hyperplane".

Now here in the below Figure 3.1 we can see that for classification we have many decision boundaries, which are capable of classifying the dataset, but the question is that which hyper plane should be selected such that it will be optimal?

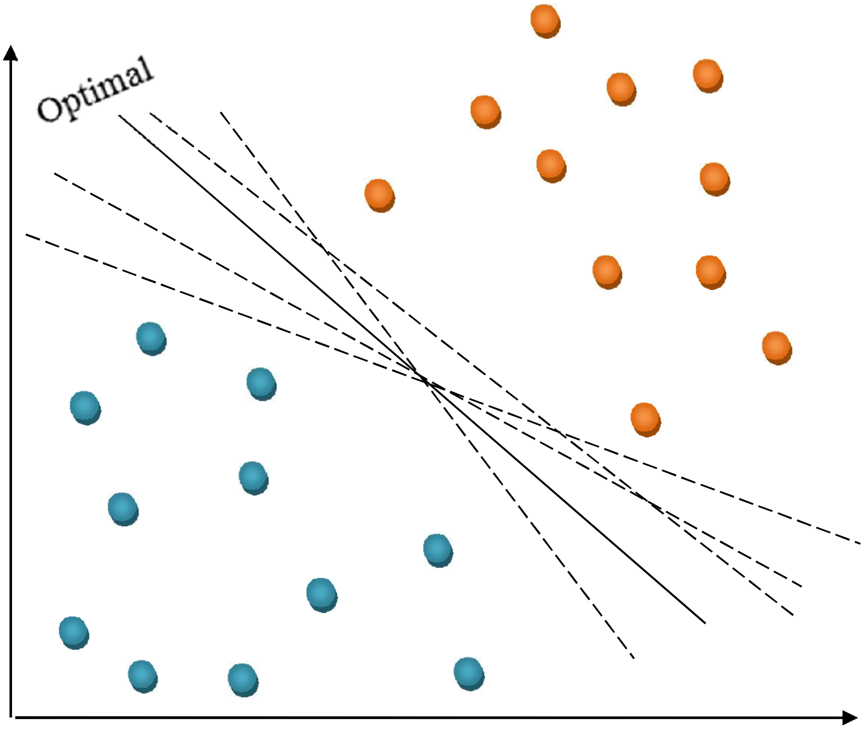


Figure 3. 1. Hyperplanes.

We assume the data are linearly separable, given a set of labeled training vectors:

$$(x_1, y_1) ,... (x_i, y_i), x_i \epsilon\, R^d ,y_i \epsilon \{ -1, +1\},\ \ i = 1, 2... n,$$

Denote the i^{th} training vector with its corresponding class label, and let

$$h(x): w.x + b = 0 \qquad\qquad (1)$$

Where each vector belongs to one of two classes according to its label, support vector machines look for the hyperplane with maximal margin that separates the data, the closest points, which alone are used to determine the hyperplane, are called support vectors.

The class is given by the sign of $h(x): f(x) = sign(h(x)$. if $h(x) \geq 0$ Then x is class 1 otherwise x is of class -1. The separator is then a hyperplane of equation

$h(x): w.x + b = 0$

Si (x_i, y_i) is p elements of the learning base noted A_P, we want to find the classifier h such that: $y_i(w.x_i + b) \geq 0\ i\ \epsilon\ [1, P]$

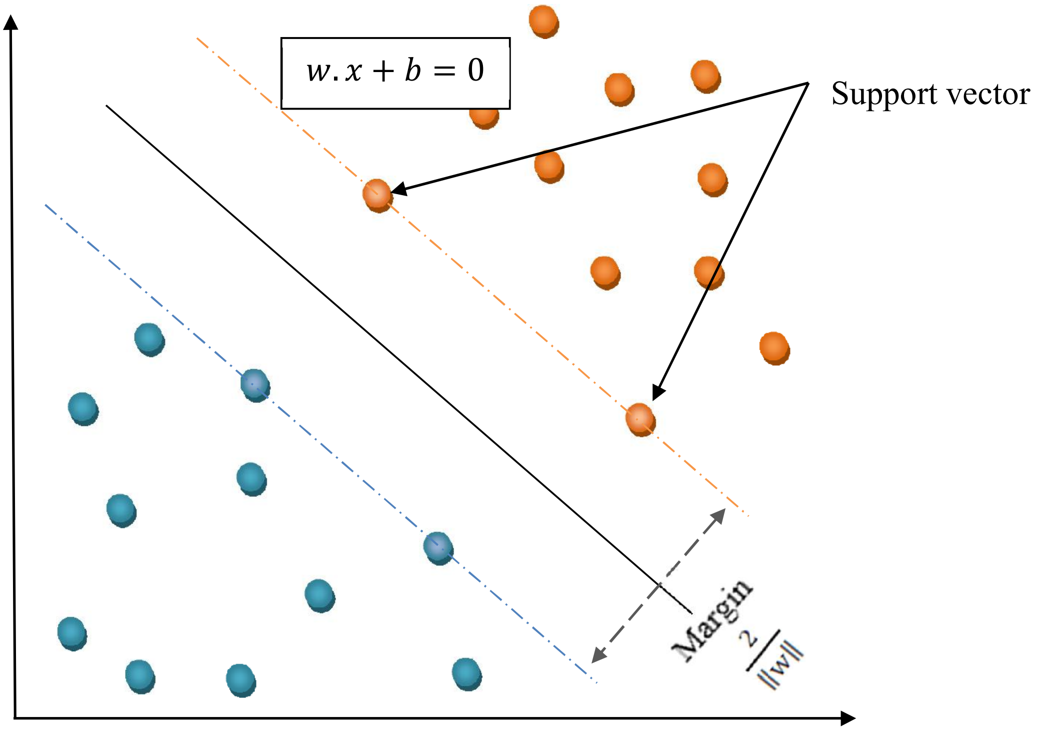


Figure 3. 2. The Optimal Hyper-plane.

In the context of SVMs, the margin Figure 3.2 is defined to be twice the distance from the closest point in the positively (negatively) labeled data set to the hyperplane.

The point x_0 closest to h, the unsigned distance from x_0 to h is defined by:

$$\frac{1}{||w||} \tag{2}$$

The margin between the two classes is defined to be twice the distance of equation (2), is Equals to $\frac{2}{||w||}$. The dual of the primal optimization problem to find the hyperplane with maximal margin, the following constrained optimization problem is solved, where rather than maximizing $\frac{2}{||w||}$, for mathematical convenience, $\frac{1}{2}\,||w||^2$ is minimized, and further, the constraint inequalities $\mid w.x_i + b \mid \geq 1$ are equivalently expressed as $y_i\,(w.x_i + b) \geq 1$ (y_i being 1 if x_i is on the positive side of h that is along the direction of w, and -1 otherwise, does the job of the absolute value):

$$\begin{cases} min(\frac{1}{2}\,||w||^2) \\ y_i\,(w.x_i + b) \geq 1\,, \;\; \forall(x_i, y_i) \in A_P \end{cases} \tag{3}$$

3.2. Dual Formulation:

The dual of the primal optimization problem in (3) can be transformed into a dual formulation using the Lagrange multipliers. The equation is then written in the following form:

$$L(w, b, \alpha) = \frac{1}{2}||w||^2 - \sum_{i=1}^{P} \alpha_i(y_i(w.x_i + b) - 1) \tag{4}$$

The formulation of Lagrange allows to find the extremums by canceling the partial derivatives of the function $L(w + b + a)$. The Lagrange L should be minimized in relation to w and b and maximized in relation to a, this new problem is solved by calculating the partial derivatives:

$$\frac{\partial L}{\partial w} = w - \sum_{i=1}^{P} \alpha_i y_i x_i = 0 \qquad (5)$$

$$\frac{\partial L}{\partial b} = \sum_{i=1}^{P} \alpha_i y_i = 0 \qquad (6)$$

By re-injecting the first two partial derivatives 5 and 6 into equation 2 we obtain:

$$L(\alpha) = \frac{1}{2}\sum_{i=1}^{P} \alpha_i y_i \sum_{j=1}^{P} \alpha_j y_j x_i . x_j - \sum_{i=1}^{P} \alpha_i y_i \sum_{j=1}^{P} \alpha_j y_j x_i . x_j - \sum_{i=1}^{P} \alpha_i y_i b + \sum_{i=1}^{P} \alpha_i$$

The following dual formulation is extracted (depending on α_i):

$$L(a) = \sum_{i=1}^{p} \alpha_i - \frac{1}{2}\sum_{i,j} y_i \, y_j \, \alpha_i \;\; \alpha_j x_i . x_j \qquad (7)$$

The aim is therefore to maximize $L(a)$ under the constraints $\alpha_i \geq 0$ and $\sum_i \alpha_i y_i = 0$.

At the optimal α^* The **Karush–Kuhn–Tucker** optimality condition (conditions KTT) are satisfied and make it possible to write the following equality:

$$\alpha_i [\, y_i \, (w . x_i + b) - 1] = 0, \forall_i = \epsilon \; [1, P] \qquad (8)$$

This gives us α_i =0 or $y_i \, (w . x_i + b) - 1$= 0. This Two possibilities imply that only the α_i

Associated with examples on the margin may be non-zero. In other words, these examples on the margin constitute the support vectors, which alone help to define the optimal hyperplane.

This maximization is a quadratic programming problem of dimension equal to the number of examples. Equation 5 gives us the optimum value for w note w^*:

$$w = \sum_i \alpha^* y_i x_i = 0$$

With α^* the optimal Lagrange coefficients. Using the equation of hyperplane 1 we obtain the maximum margin hyperplane:

$$h(x) = \sum_{i=1}^{p} \alpha^* y_i x . x_i + b \tag{9}$$

3.3. Non Separable Patterns Case:

For linear data, a separating hyperplane can be used for classifying it. However not all the time have linear data sometimes nonlinear data has to be classified where is separating hyperplane won't work easily hence we need a special function known as the kernel function to map the nonlinear data to high dimensional feature space. Hence the new mapping is now linearly separable see figure below:

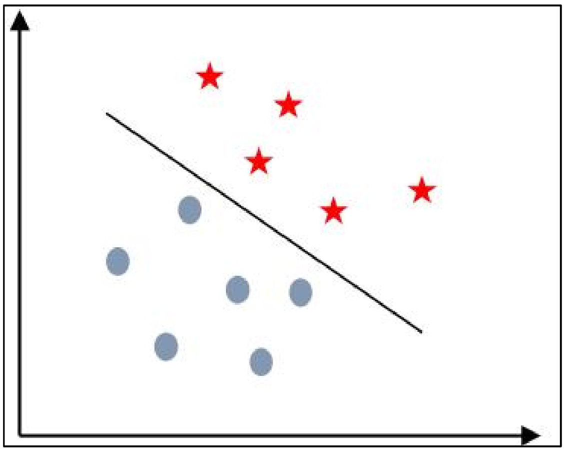


Figure 3. 3. linearly separable case.

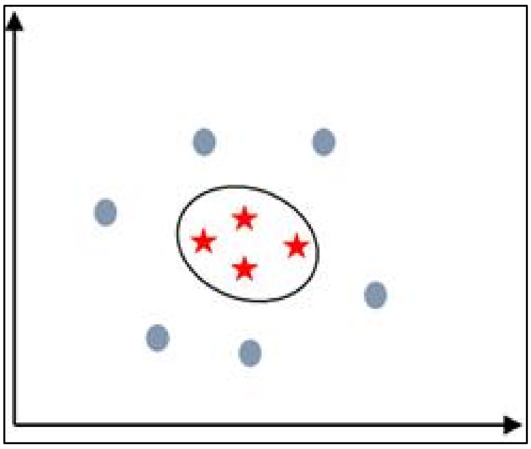


Figure 3. 4. Non linearly separable case.

The Mapping function defined by kernel is:

$$k(x, y) = \phi(x).\phi(x)$$

Transformation of data into feature space makes it easy to classify nonlinear data and define a similarity measure based on the dot product. In non-linear classification the data sets are present anywhere which cannot be classified using hyperplane, hence the data has to be transformed into high dimensional feature space.

$$< x_1.x_2 > k(x_1, x_2) =< \phi(x_1).\phi(x_2) >$$

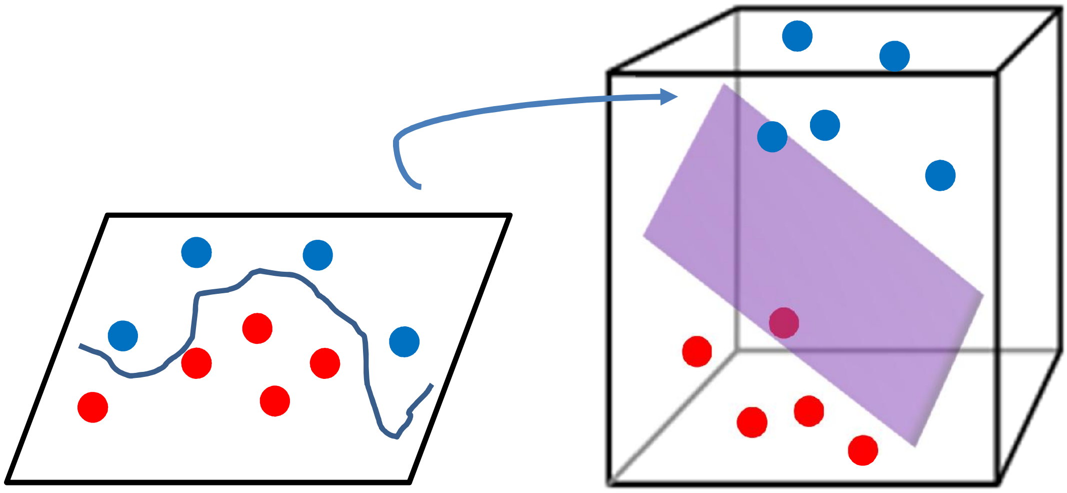


Figure 3. 5. Feature Space Representation.

Therefore introducing non-negative slack variables $\xi = \ \xi_1 \ldots \xi_p$ to the soft-margin SVMs can be derived, which make it possible to supple the constraint for each example. The new primal form described in 3 then becomes:

$$\begin{cases} min\left(\frac{1}{2}\,\|w\|^2 + \sum_{i=1}^{p}\ \xi_i\right) \\ y_i\,(w.x_i + b) \geq 1 - \xi_i\,,\ \ \forall(x_i, y_i) \in R^d \end{cases} \qquad (10)$$

As for the primal form we obtain a new dual formulation which is then similar to that described previously. If we also use the kernel function K in the dual 7 formulation by applying the Lagrange multiplier method, we then try to maximize the new function $L(a)$:

$$L(w,b,\xi,\alpha) = \frac{1}{2}w^2 + C\sum_{i=1}^{P}\xi_i - \sum_{i=1}^{P}\alpha_i[y_i(w.x_i + b) - 1 + \xi_i]$$

Applying the same method we obtain $L(a)$ from the expression of the preceding Lagrangian

$$L(a) = \sum_{i=1}^{p}\alpha_i - \frac{1}{2}\sum_{i,j}^{p}\alpha_i\ \alpha_j y_i\, y_j\, k(x_i.x_j) \qquad (14)$$

$$\forall(x_i.x_j) \geq 0 \in A^P, 0 \leq \alpha_i \leq C, and \sum_i \alpha_i y_i = 0$$

The only change is the additional constraint on the coefficients α_i, which results in the upper bound C. So the separator hyperplane is rewritten with the kernel function in the following form:

$$h(x) = \sum_{i=1}^{p}\ \alpha_i{}^{*} y_i k(x.x_i) + b$$

3.3.1. Kernel Function:

Several SVM kernel functions that were used to map the input space to feature space and gave a good classification accuracy when classifying a new example. The most common kernel functions are [32]:

- **Linear Core:** $K(x, y) = x.y$.
- **Sigmoid:** $K(x, y) = tanh(ax.y + b)$.
- **Radial Basis Function (RBF):** $K(x, y) = exp\left(-\frac{||x-y||^2}{\sigma^2}\right)$.
- **Polynomial:** $K(x, y) = (ax.y + b)\hat{}d$.

4. Multi Class Support Vector Machines Based On Binary Classification:

Since SVMs were originally designed for binary problems, several methods were proposed to extend binary SVMs to solve multi classification problems. Multi-class pattern recognition problems are commonly solved using a combination of binary SVMs and a decision strategy to decide the class of the input pattern. Each SVM is independently trained. In this section, we introduce OAA, OAO, and multi-class SVM methods which are based on binary classifiers.

We assume hereafter that training data set (x_i, y_i), consists of N examples belonging to M classes, where C_iє { 1 ... M} is the class of x_i and ϕ is the mapping function.

4.1. One-Against-All:

The One against All approach is considered as the earliest extension of binary SVM. In OAA approach, M binary SVM models are constructed where M is the number of classes. An SVM is constructed to discriminate each class against the others $(M\text{-}1)$ classes [33].

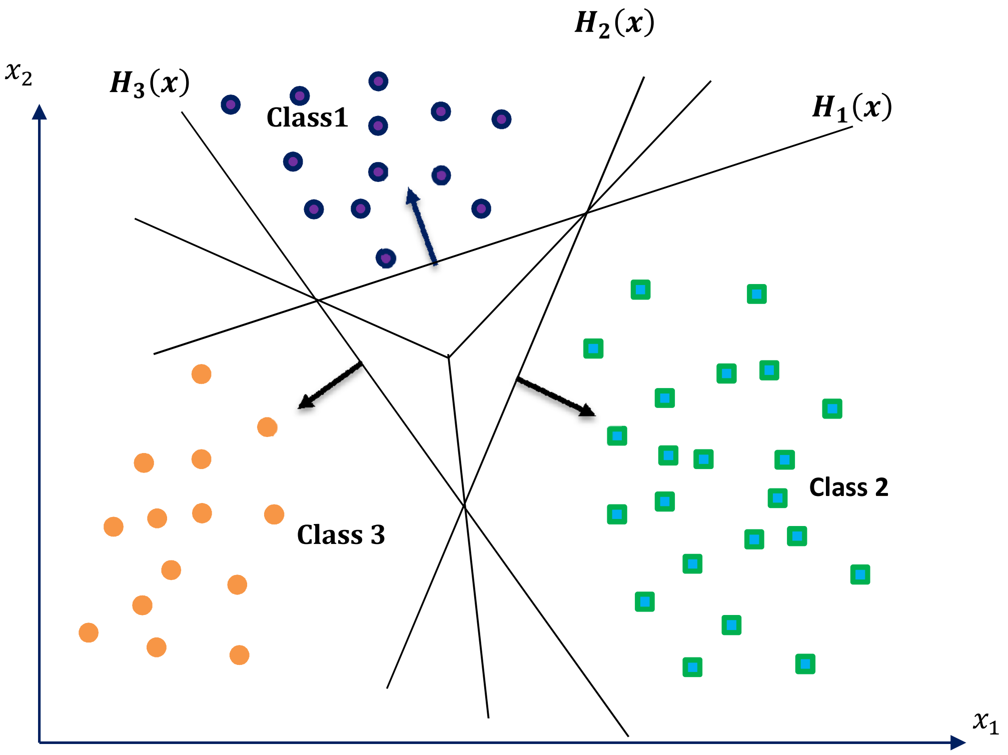


Figure 3. 6. One-vs-all.

4.2. One-Against-One:

One-against-one. This method requires $\frac{1}{2}m(m\text{-}1)$ Classifiers for All pairs of possible classes.

During the test, the method requires the combination of all classifier outputs for a decision to be issued [33].

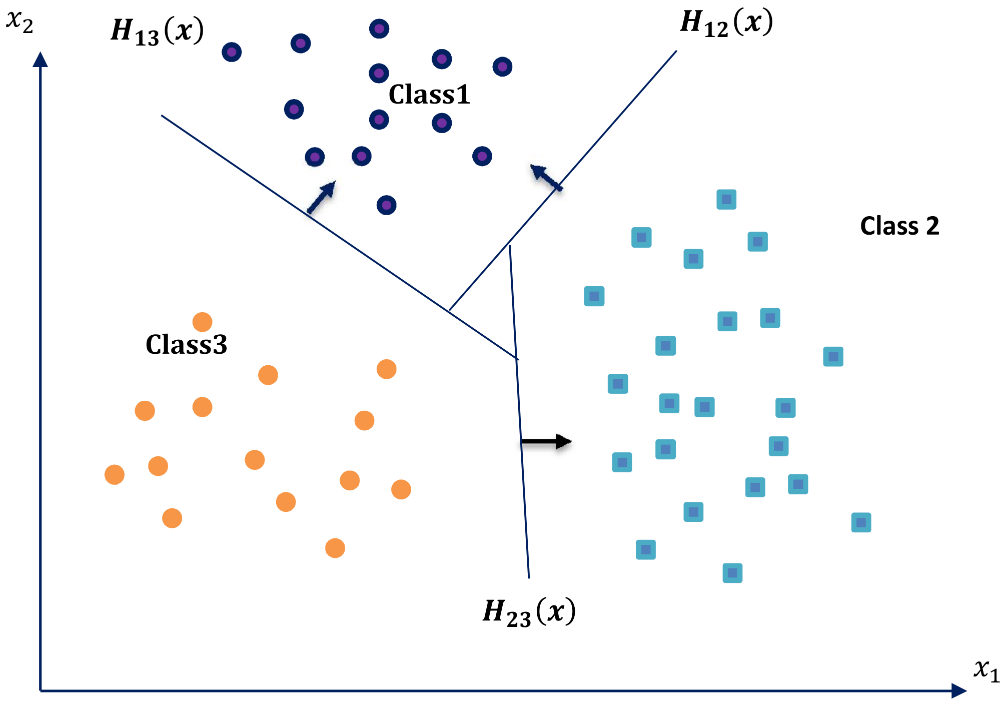


Figure 3. 7. One-Vs-one approach.

5. Domains of Application of SVM's:

SVM is a classification method that shows good performance in solving various problems. This method has shown its effectiveness in many fields of applications such as [34]:

- Image processing;
- Texts categorization;
- Medical Diagnostics;
- Very large data sets;
- E-learning;
- Speech Recognition.

6. Conclusion:

Support vector machines are one of the widely used machine learning algorithms for data classification. In this chapter we principle general of working support vector machine and their mathematical basis. We have exposed the two cases of the data separable and non-separable.

In the simplest form, SVM uses a linear hyperplane to create a classifier with a maximal margin, in other cases, where the data is not linearly separable, the data is mapped into a higher dimension feature space. This task is achieved using various nonlinear mapping functions: polynomial, sigmoid and Radial Basis Functions (RBF).

4. Introduction:

In this chapter, we have presented the database generated, the working Environment and the different steps of a complete system of printed Arabic recognition were implemented with the programming language MATLAB 2013, because it made it is easy to create graphical interface using its GUI (graphical user interface).

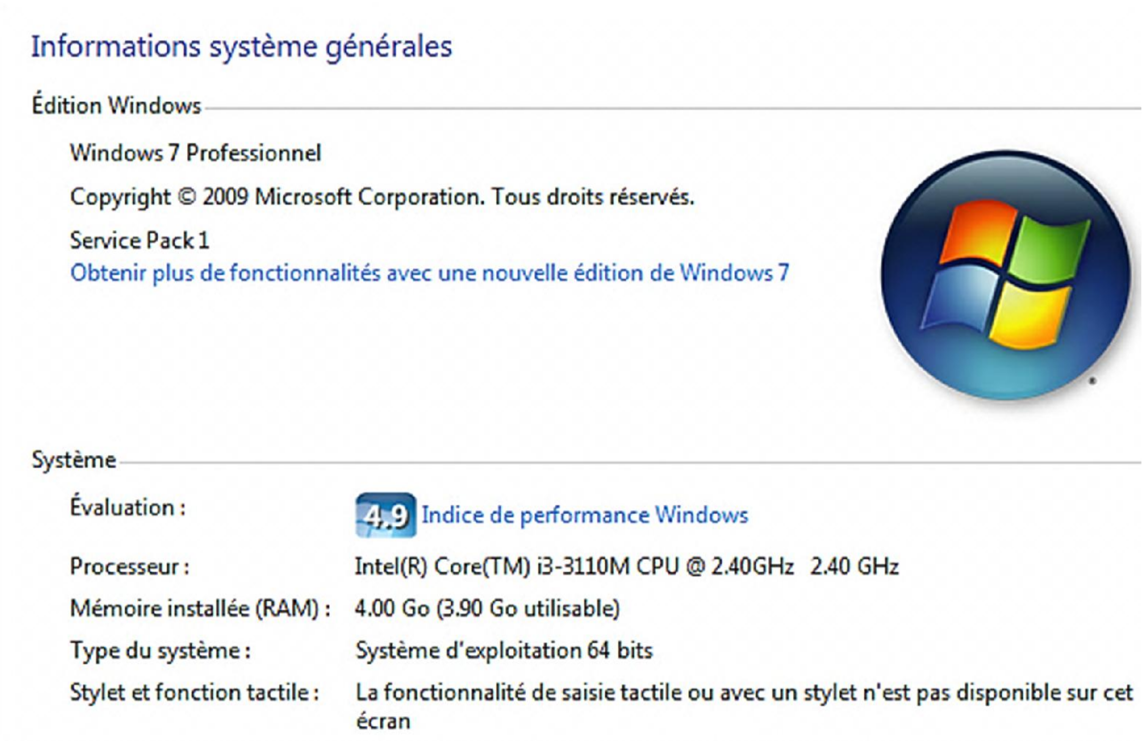
The goal of this chapter is to present our off-line printed Arabic system and evaluate it performance, which was implemented according to the implemented segmentation method and the use of RBF kernel of the SVM.

The proposed system for printed Arabic recognition has several major steps. Each of the recognition step affect the accuracy and the performance of the recognition, so we focus on enhancing the character recognition accuracy of AOCR through using effective features as well the segmentation that are capable of increasing the recognition accuracy through its ability to extract information related to the statistical shape of the characters.

5. Working Environment:

2.1. Hardware Environment:

Hard drive: 500 GB;



Informations système générales

Édition Windows

Windows 7 Professionnel

Copyright © 2009 Microsoft Corporation. Tous droits réservés.

Service Pack 1

Obtenir plus de fonctionnalités avec une nouvelle édition de Windows 7

Système

Évaluation :	4,9 Indice de performance Windows
Processeur :	Intel(R) Core(TM) i3-3110M CPU @ 2.40GHz 2.40 GHz
Mémoire installée (RAM) :	4.00 Go (3.90 Go utilisable)
Type du système :	Système d'exploitation 64 bits
Stylet et fonction tactile :	La fonctionnalité de saisie tactile ou avec un stylet n'est pas disponible sur cet écran

Figure 4. 1. Screenshot describing the characteristics of the machine.

2.2. Programming Language

a. MATLAB

The name MATLAB stands for MATRIX LABRATORY.

MATLAB is a high-performance language for technical computing. It integrates computation, visualization, and programming in an easy-to-use environment where problems and solutions are expressed in familiar mathematical notation. Typical uses include:

- Math and computation;
- Algorithm development;
- Modeling, simulation, and prototyping;
- Data analysis, exploration, and visualization;

- Scientific and engineering graphics;
- Application development, including graphical user interface building.

MATLAB is an interactive system whose basic data element is an array that does not require dimensioning. This allows you to solve many technical computing problems, especially those with matrix and vector formulations.

b. Why are we to choose Matlab?

MATLAB allows interactive work is in command mode or program mode, while still having the ability to make graphical visualizations. It has the following features:

- Computing power;
- Math library;
- The extent of the range of numbers and their details;
- The inclusion of GUI functions and utilities within the graphical tool;
- The ability to liaise with other conventional programming languages;

For the GUI, scientific and even artistic representations of objects can be created on the screen using mathematical expressions or directly using a graphical tool. Indeed, the design of our application we chose the toolbox in MATLAB GUIDE.

c. Introducing the tool box GUIDE

GUIDE or Graphical User Interface Development Environment is a graphical tool in MATLAB that provides a set of tools for creating graphical interfaces intuitively. The user has several predefined graphical tools such as buttons, menus ... that allow it to create a graphical interface that communicates with a computer program.

6. The Data Base Used:

3.1. Dataset Generation:

According to our literature survey, there is no standard primitive dataset for evaluating the different features and classifiers utilized by others, at least doesn't contain many font type.

So we have proposed a dataset for evaluate our features to be the base for our basic AOCR system. Our database contain 28 characters from 'Alif' to 'Yaa' in addition to a composite character 'LamAlif' (لأ), which is composed of 'Lam' (ل) and 'Alif' (أ) characters, yielding in 29 characters, It is contains the primitives of the Arabic characters in all of its four positions (Isolated-

Start-Middle-Final); e.g. the “Baa” character has the following primitive positions: ‘ ب’ in the Isolated position, “ بـ” in Start position, “ـبـ”in the Middle, and ـب in the Final, which yielded 100 primitive shapes. Each one is written in 61 different sizes (from the size 12 to 72). Which cause us a 6100 sample.

This database contain 5 different Arabic font types (Ariel, Times New Roman, Simplified Arabic, Traditional Arabic and Andalus), which increase the database to be contain 30500 (6100 * 5) samples in whole dataset.

Generation process:

a. It has been generated using Microsoft office word software.

b. Generated using (Ariel, Times New Roman, Simplified Arabic, Traditional Arabic and Andalus) type.

c. The dataset consists of 28 characters from ‘Alif’ to ‘Yaa’ and the composite character ‘LamAlif’ (لأ), which is composed of ‘Lam’ (ل) and ‘Alif’ (أ) characters, yielding in 29 characters,

d. (ـ)(Shift +ت) has been used to generate characters in positions (Start, Middle and End) as shown in Table 4.1 this resulted in 100 primitive.

e. Each one of the 100 primitives is written in (61) different sizes (12-72) for addressing the different scattering of pixels through different sizes.

f. After that, the characters were exported from Word software as a pdf file.

g. The pdf file was converted to images using (PDFill_Free_PDF_Tools[1]) software at 110 dpi (dot per inch) to address small resolution scanned images.

h. Then, each images been segmented into single characters.

i. This Model contains (61*100) = 6100 character sample in one font.

[1] https://pdfill-free-pdf-tools.en.softonic.com/post-download

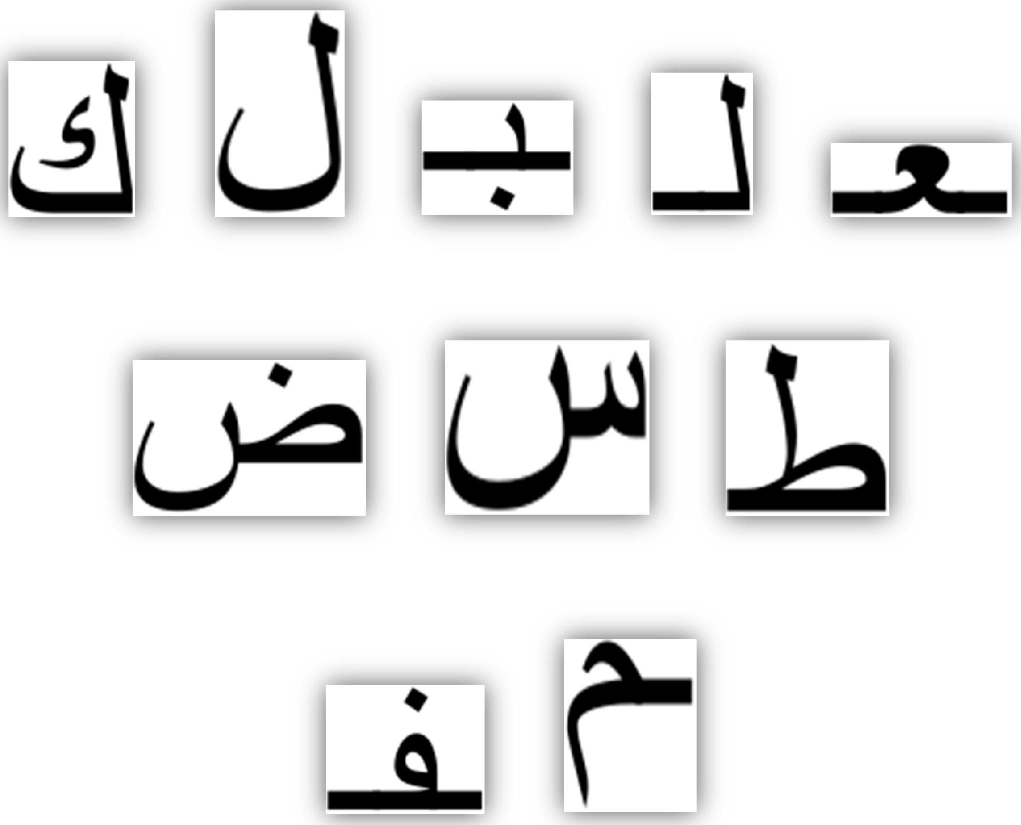


Figure 4. 2. Example images of the 10 classes used in our database.

Table 4. 1. The Arabic letters used with its different forms.

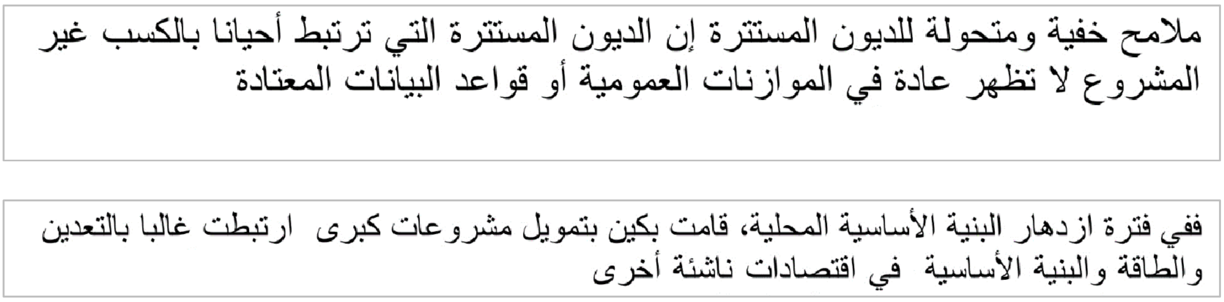
Character name		**Arabic Language Primitives**			
		Isolated	Connected		
			End	Middle	Start
Alif	**ألف**	ء \ؤ \ أ	ـئ \ـؤ \ـا	ـئـ	ئـ
Baa	**باء**	ب	ـب	ـبـ	بـ
Taa	**تاء**	ة \ت	ـة \ـت	ـتـ	تـ

Thaa	**ثاء**	ث	ـث	ـثـ	ثـ
Jeem	**جيم**	ج	ـج	ـجـ	جـ
Haa	**حاء**	ح	ـح	ـحـ	حـ
Khaa	**خاء**	خ	ـخ	ـخـ	خـ
Daal	**دال**	د	ـد	-	-
Thaal	**ذال**	ذ	ـذ	-	-
Raa	**راء**	ر	ـر	-	-
Zaay	**زاي**	ز	ـز	-	-
Seen	**سين**	س	ـس	ـسـ	سـ
Sheen	**شين**	ش	ـش	ـشـ	شـ
Saad	**صاد**	ص	ـص	ـصـ	صـ
Dhaad	**ضاد**	ض	ـض	ـضـ	ضـ
Ttaa	**طاء**	ط	ـط	ـطـ	ط
Dthaa	**ظاء**	ظ	ـظ	ـظـ	ظـ
Ain	**عين**	ع	ـع	ـعـ	عـ
Ghen	**غين**	غ	ـغ	ـغـ	غـ
Faa	**فاء**	ف	ـف	ـفـ	فـ
Qaf	**قاف**	ق	ـق	ـقـ	قـ
Kaf	**كاف**	ك	ـك	ـكـ	كـ

Lam	لام	لا	ـل	ـلـ	لـ
Mem	ميم	م	ـم	ـمـ	مـ
Noon	نون	ن	ـن	ـنـ	نـ
Haa	هاء	ه	ـه	ـهـ	هـ
Wow	واو	و	ـو	-	-
Yaa	ياء	ي	ـي	ـيـ	يـ

3.2. Text Database Description:

The used dataset contains different resources of texts as images, these resources include magazines, books, and papers, these documents are images with 300 dpi resolution, which generated using both Times New Roman and Arial font types and sizes 14-16.



ملامح خفية ومتحولة للديون المستترة إن الديون المستترة التي ترتبط أحيانا بالكسب غير
المشروع لا تظهر عادة في الموازنات العمومية أو قواعد البيانات المعتادة

ففي فترة ازدهار البنية الأساسية المحلية، قامت بكين بتمويل مشروعات كبرى ارتبطت غالبا بالتعدين
والطاقة والبنية الأساسية في اقتصادات ناشئة أخرى

Figure 4. 3. Samples of text database.

4. Architecture of the Proposed System:

The following figure shows the different steps of our recognition system, step by step:

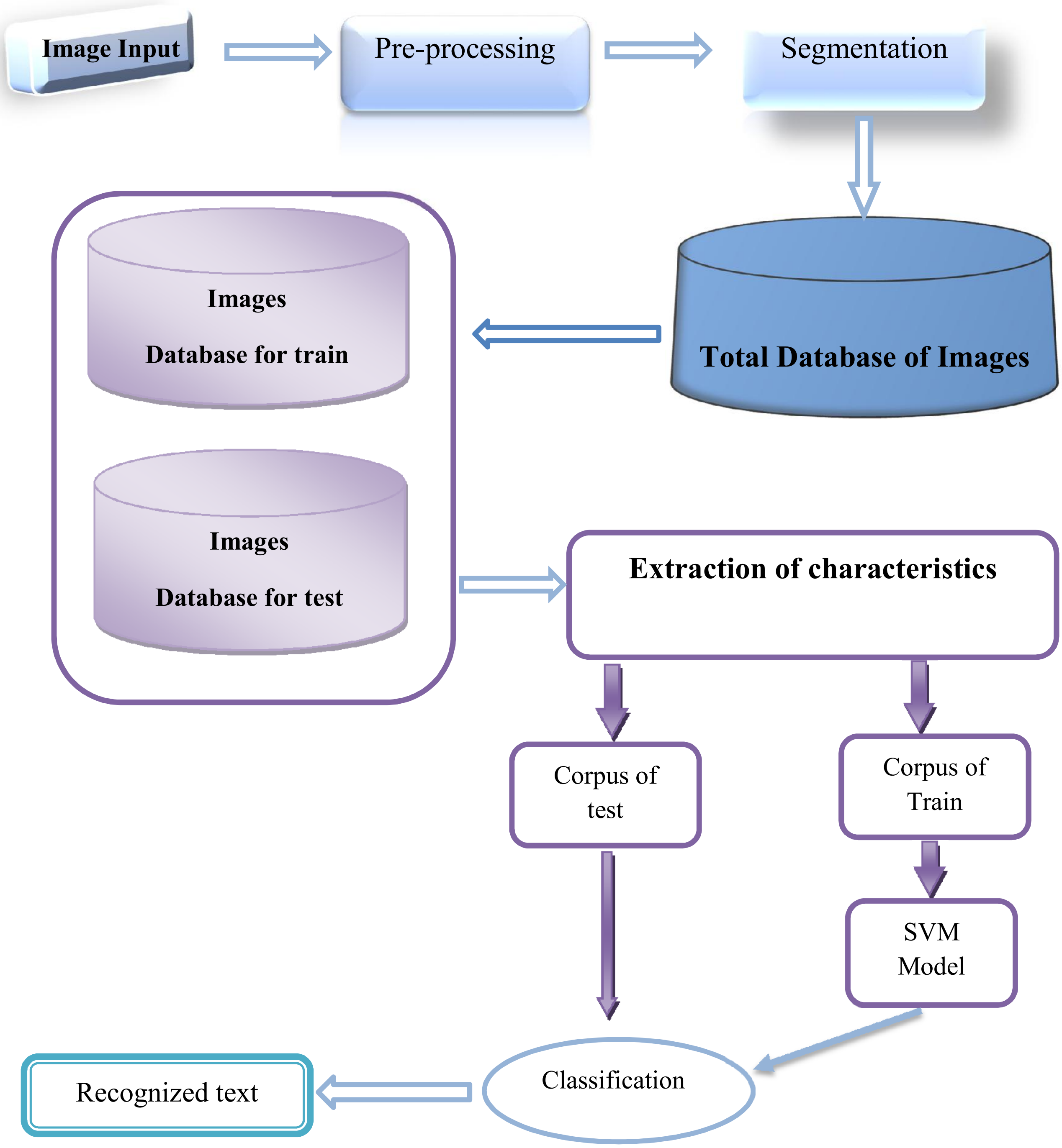
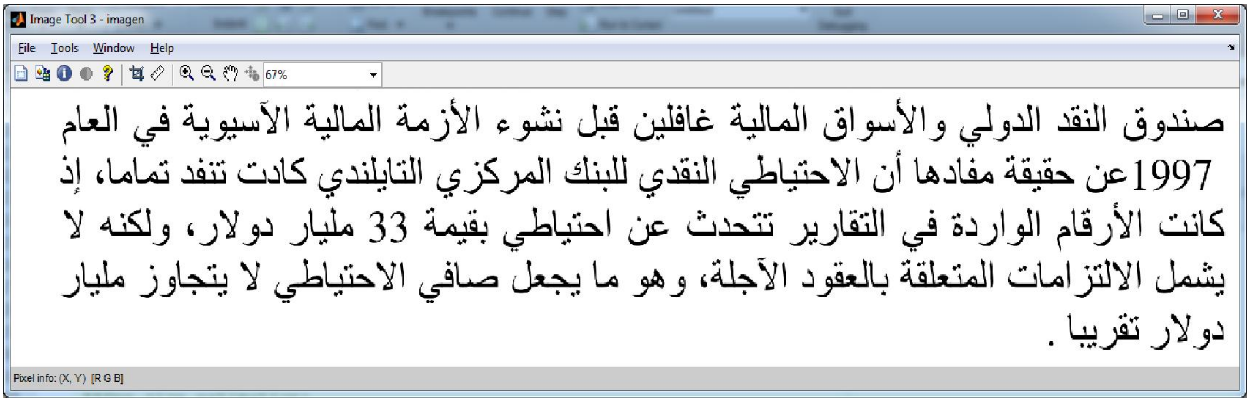


Figure 4. 4. General scheme of our recognition system.

4.1. Pre-processing:

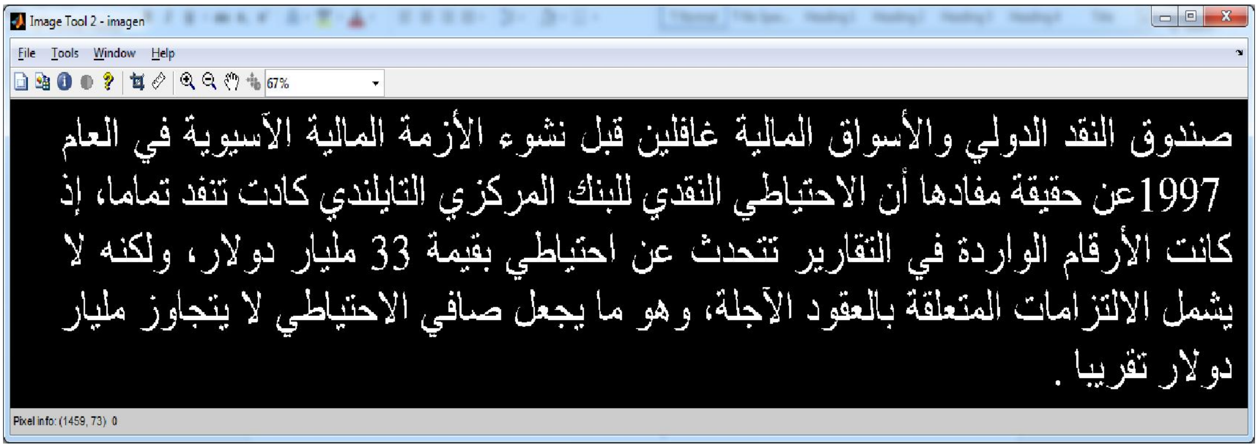
4.1.1. Binarization:

The first step of the preprocessing stage is image thresholding which converts the grayscale image into binary.



صندوق النقد الدولي والأسواق المالية غافلين قبل نشوء الأزمة المالية الآسيوية في العام
1997عن حقيقة مفادها أن الاحتياطي النقدي للبنك المركزي التايلندي كادت تنفد تماما، إذ
كانت الأرقام الواردة في التقارير تتحدث عن احتياطي بقيمة 33 مليار دولار، ولكنه لا
يشمل الالتزامات المتعلقة بالعقود الآجلة، وهو ما يجعل صافي الاحتياطي لا يتجاوز مليار
دولار تقريبا .

(a)



صندوق النقد الدولي والأسواق المالية غافلين قبل نشوء الأزمة المالية الآسيوية في العام
1997عن حقيقة مفادها أن الاحتياطي النقدي للبنك المركزي التايلندي كادت تنفد تماما، إذ
كانت الأرقام الواردة في التقارير تتحدث عن احتياطي بقيمة 33 مليار دولار، ولكنه لا
يشمل الالتزامات المتعلقة بالعقود الآجلة، وهو ما يجعل صافي الاحتياطي لا يتجاوز مليار
دولار تقريبا .

(b)

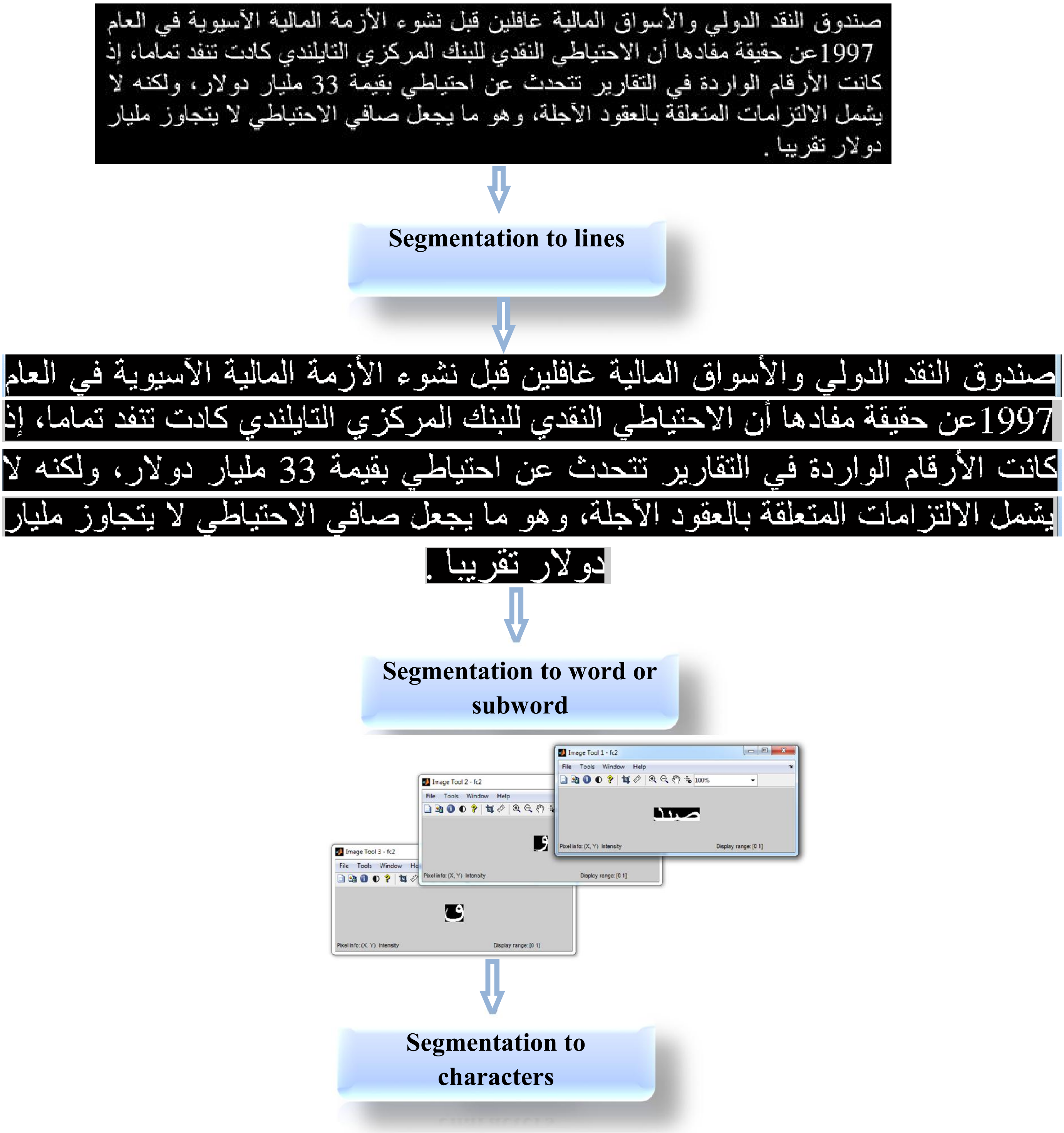
Figure 4. 5. (a) Original image (b) image after Binarization.

4.2. Segmentation:

Generally, segmentation process try to split a document into regions in the order pages into multi lines or single line, lines into words, and words into characters.

There are several classical segmentation methods that commonly used such as Projection Analysis, Connected Component Processing, etc.

In the proposed system the segmentation algorithm used based on the projection profile we chose it because it the most common method that used since it is simple and fast, it consists of three levels of segmentation: lines segmentation, sub-words segmentation and characters segmentation:



صندوق النقد الدولي والأسواق المالية غافلين قبل نشوء الأزمة المالية الآسيوية في العام
1997عن حقيقة مفادها أن الاحتياطي النقدي للبنك المركزي التايلندي كادت تنفد تماما، إذ
كانت الأرقام الواردة في التقارير تتحدث عن احتياطي بقيمة 33 مليار دولار، ولكنه لا
يشمل الالتزامات المتعلقة بالعقود الآجلة، وهو ما يجعل صافي الاحتياطي لا يتجاوز مليار
دولار تقريبا .

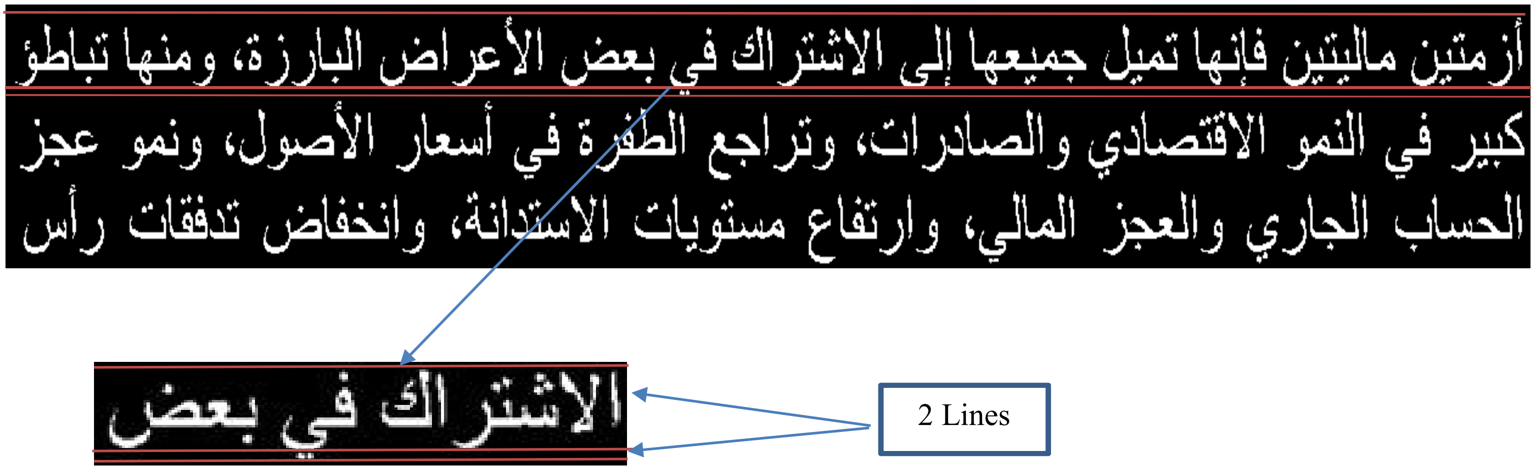
صندوق النقد الدولي والأسواق المالية غافلين قبل نشوء الأزمة المالية الآسيوية في العام
1997عن حقيقة مفادها أن الاحتياطي النقدي للبنك المركزي التايلندي كادت تنفد تماما، إذ
كانت الأرقام الواردة في التقارير تتحدث عن احتياطي بقيمة 33 مليار دولار، ولكنه لا
يشمل الالتزامات المتعلقة بالعقود الآجلة، وهو ما يجعل صافي الاحتياطي لا يتجاوز مليار
دولار تقريبا .

Figure 4. 6. General scheme of segmentation.

a. **Segmentation to line:** involves horizontal projection of the image rows to find the empty rows between rows that contain text as in Table below, Lines segmentation algorithm steps:

Step 0	**If** the current row index ***i*** is smaller than the max rows index build up the horizontal projection of this row. **If** its value equals 0 Go to step1 **Else** Go to step 2 **Else** End
Step 1	Cut the corresponding row.
Step 2	Go to the next raw. Go to step 0.

- **over segmentation problem case:**



أزمتين ماليتين فإنها تميل جميعها إلى الاشتراك في بعض الأعراض البارزة، ومنها تباطؤ
كبير في النمو الاقتصادي والصادرات، وتراجع الطفرة في أسعار الأصول، ونمو عجز
الحساب الجاري والعجز المالي، وارتفاع مستويات الاستدانة، وانخفاض تدفقات رأس

الاشتراك في بعض

➢ Over segmentation problem solving:

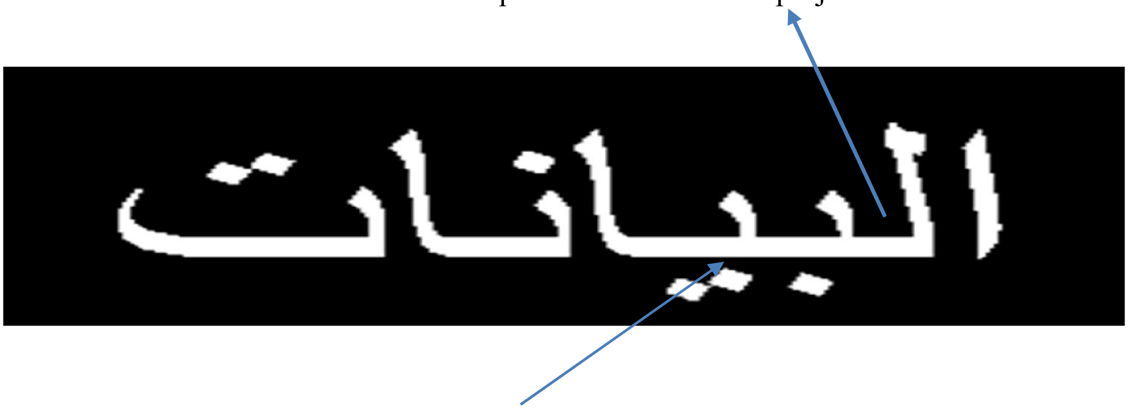
To handle this issue, first we compute the pin size which is the pen thickness used for writing, the pen size can handle by taking the most frequent value in the vertical projection, figure 6 shows an example of pen size calculations:

We notice that the space between the first cut line and the dot is equal to:

$$\textbf{Dot location} = \textbf{Pen thinkness} \ /2$$

So before cutting we must verifier the space between the first cut and dot location, to ensure that the cut location is not above the dots.

Most frequent value in vertical projection =6



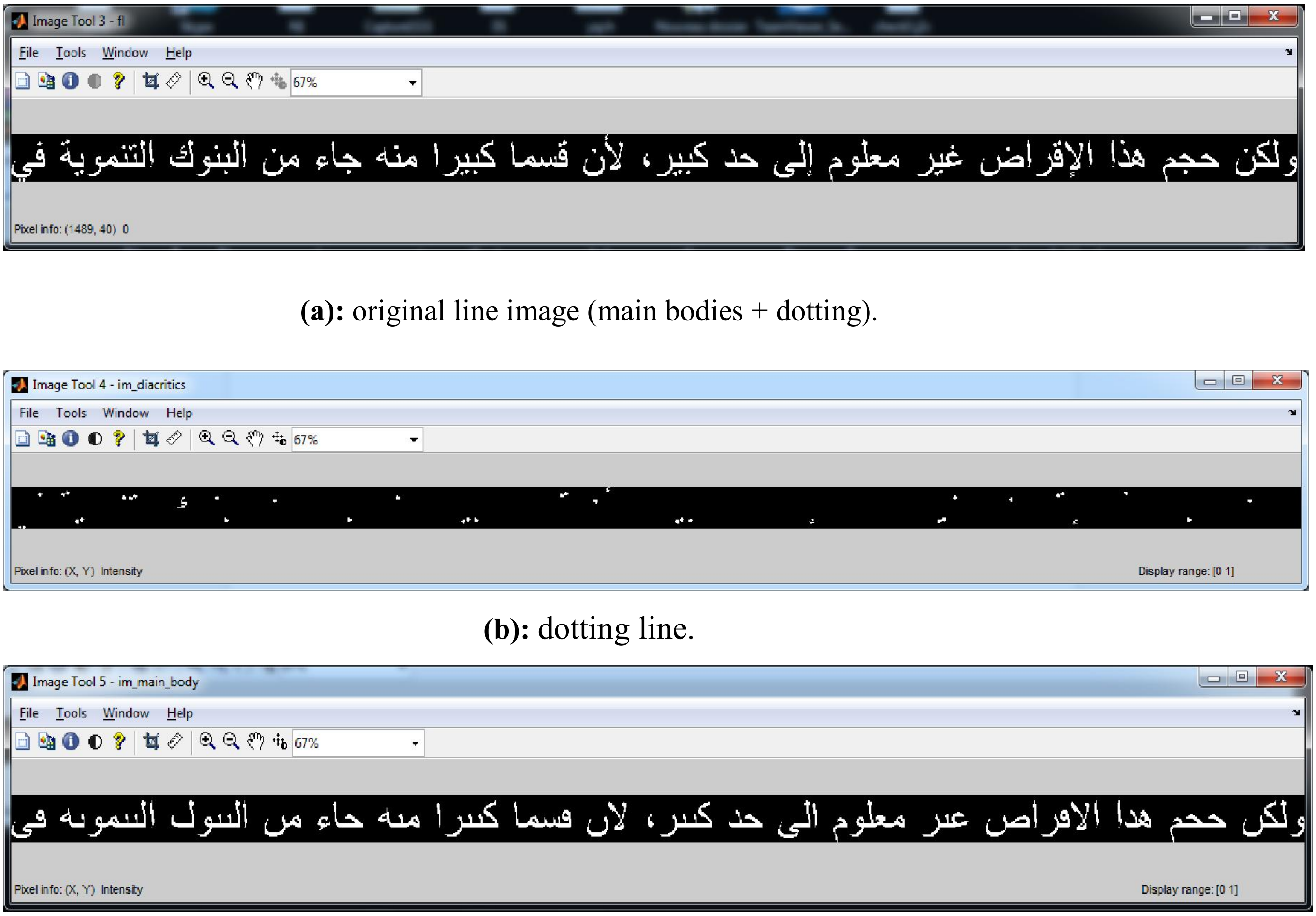
Space between the cut line and Dot =3

Figure 4. 7. The relation between the pen thinkness and the cut place.

➢ Text Line Image Processing:

- **Dot Elimination:**

In this stage, to facilitate the word and character segmentation stages extracting the main body of the text without dotting and base line are generated:



ولكن حجم هذا الإقراض غير معلوم إلى حد كبير، لأن قسما كبيرا منه جاء من البنوك التنموية في

(a): original line image (main bodies + dotting).

(b): dotting line.

ولكں حجم هدا الاٯراص عىر معلوم الى حد كىىر، لاں ٯسما كىىرا مىه حاء مں الىںوك الىںموىه ڡى

(c): Main body of line.

Figure 4. 8. Text Line processing stages.

- **Determine the Threshold Value:**

```
Vertical_Projection of the current line = sum_ column (Main body of line);
Pen size of the current line = the most frequency value of vertical_Projection;

threshold = Pen size of the current line;
```

a. Segmentation To Word/ Sub Word:

Words segmentation is the next level to follow after the lines segmentation. Vertical projection profile of images columns is performed to each text line in-order to divide it into sub-words. Word/Sub word segmentation algorithm steps:

```
Step 0    If the current column index i is smaller than the max columns
              index build up the vertical projection of this column.
                  If its value equals 0
                  Go to step1
                  Else
                  Go to step 2
              Else End
Step 1    Cut the corresponding column.
Step 2    Go to the next column.
              Go to step 0.
```

The challenge after this step is to know that the text between two gaps is a word or a sub-word, and if it is a sub-word to which word belong. Moreover, the gap between two consecutive words or sub-words is not fixed and depends on the font type, and size.

To handle this issue, the pin size is compared with the separation space. Thus, if the separation space between two consecutive words/sub-words is larger than the mean of the pen size of these two consecutive words/sub-words, then the separation region performs a separation between two different words else the separation region is between two sub-words in the same word, defined formally as:

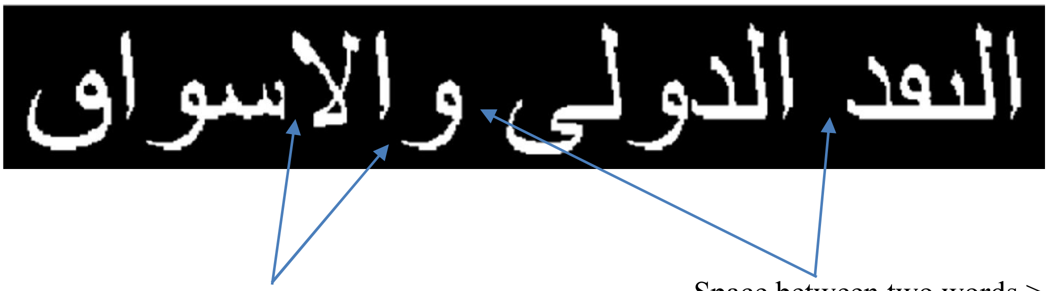
$$\boldsymbol{If\quad (length\ \ (separation\ space)\ >\ pen\ size}$$

$$\boldsymbol{then}$$

$$\boldsymbol{the\ separation\ is\ bet\ ween\ two\ different\ words}$$

$$\boldsymbol{Else}$$

$$\boldsymbol{the\ separation\ is\ between\ two\ sub\ words}$$



Space between two sub words<=threshold

Space between two words > threshold

Figure 4. 9. Distance between two sub-words in the same word and different words.

b. Segmentation to Characters:

The proposed algorithm for character segmentation based on vertical projection following a sequence of steps:

Step0 **Input**	Read sub word binary image " sub_img" go to step 1
Step 1	vertical projection V[i] is applied all over the input "sub_img" go to step 2
Step 2	Searching for the number of values equal to the threshold defined in line segmentation all over the vertical projection vector. s=0 **For** I =1:Length(V) **If** V[i] == threshold s=s+1 **End** **End** **If** s = 0 the sub word input is a letter (ام -أ -0- 9 - 5 -ح -ج -ة...) go to step 6 **Else** go to the next step
Step 3	Searching all over the vertical projection vector for values equal to the given threshold and sequent repeated more than threshold over two time (threshold/2). If no value is found than the input sub word is detected as one letter (2- لا -لآ -4 حـ...), Else go to Step 4.

Step 4	The middle of this sequence of repeated values considered as initial cutting points `N[i]`.
Step 5	**For i** =1: Number of cutting points Check the length between cutting points N and N-1 **If** length **(**cutting points N-1: cutting points N**)** **< 12** Ignore the cutting points N. **End** **End**
Step 6 **Output**	Segmented Characters

4.3. Feature Extraction :

The objective of the feature extraction stage is to capture the most relevant and discriminate characteristics of the text image to recognize. The selection of good features can strongly affect the classification performance.

In our system, two types of feature extraction methods has been used which are:

4.3.1. Statistical Features:

The used features to enhance the AOCR accuracy is the 14 statistical features used by [35] as well as two statistical features used in [36] these features can be listed as following:

There are 14 statistical features extracted from each character, four of them are for the whole image as listed below:

- ✓ Height / Width.
- ✓ Number of black pixels / number of white pixels.
- ✓ Number of horizontal transitions: Used to detect the curvature of each character and found to be effective for this purpose. The procedure runs a horizontal scanning through the

character box and finds the number of times that the pixel value changes state from 0 to 1 or from 1 to 0 as shown in Figure 4. 10. The total number of times that the pixel status changes, is its horizontal transition value.

- ✓ Number of vertical transitions:
- ✓ Similar process of horizontal transition is used to find the vertical transition value.

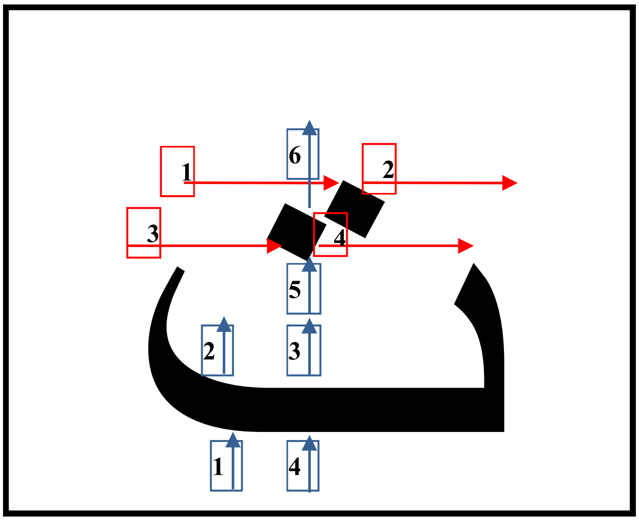
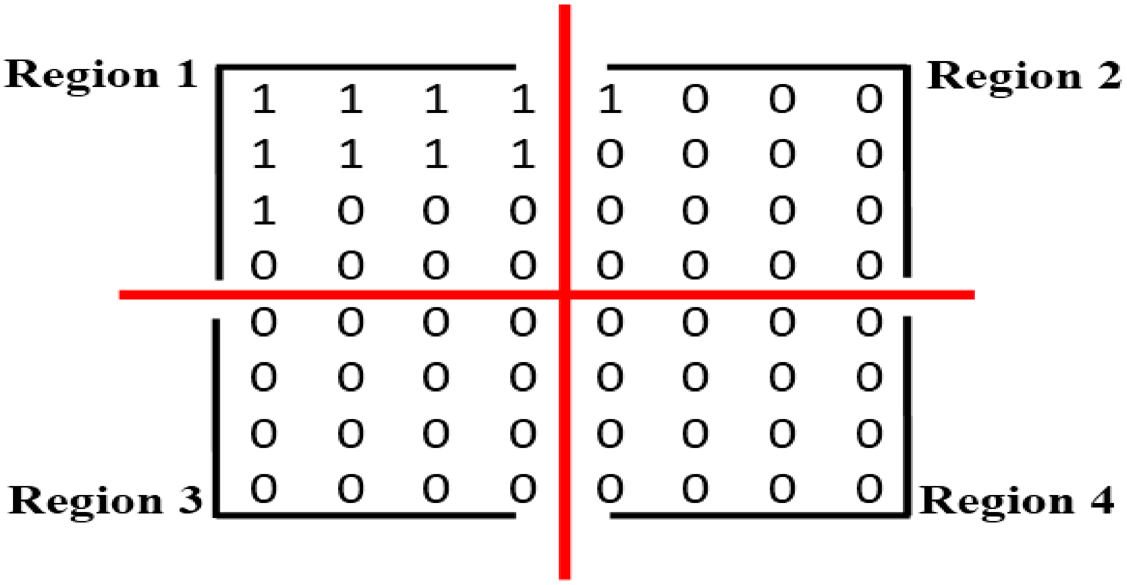


Figure 4. 10. Horizontal and Vertical transitions.

The other 10 features are extracted after dividing the image of the character into four regions to get the following ratios as shown in Figure 4.11. While black pixels is (0) and white pixels is (1):



1	1	1	1	1	0	0	0
1	1	1	1	0	0	0	0
1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

Figure 4. 11. Dividing the image into four regions.

- ✓ Black Pixels in Region 1/ White Pixels in Region 1.
- ✓ Black Pixels in Region 2/ White Pixels in Region 2.
- ✓ Black Pixels in Region 3/ White Pixels in Region 3.
- ✓ Black Pixels in Region 4/ White Pixels in Region 4.
- ✓ Black Pixels in Region 1/ Black Pixels in Region 2.
- ✓ Black Pixels in Region 3/ Black Pixels in Region 4.
- ✓ Black Pixels in Region 1/ Black Pixels in Region 3.
- ✓ Black Pixels in Region 2/ Black Pixels in Region 4.
- ✓ Black Pixels in Region 1/ Black Pixels in Region 4.
- ✓ Black Pixels in Region 2/ Black Pixels in Region 3.

Additional feature extraction:

- ✓ **Center Of Mass Feature:** Is the relative location (relative to the height and width of the image) of the center of mass of the Black Ink. The center of mass of the letter jeem is shown in Figure 4.12.

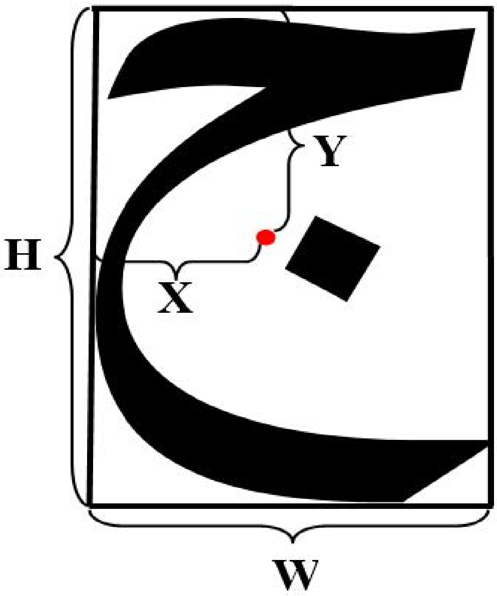


Figure 4. 12. The center of mass of the letter Jeem is marked by the (red) dot.

✓ Black Ink Histogram Features:

Each image has a horizontal black ink histogram feature and a vertical one. The horizontal black ink histogram feature $f_h = (h_1, \ldots, h_H)$, where H is the height of the bounding box of the black ink is calculated as follows:

For $i = 1 \ldots$ H, let bi be the number of black ink pixels in row i.

For 1…H, let h_i be $b_i / max\{b_i\}$.

Finally the f_h is normalized to a feature vector of 20 values.

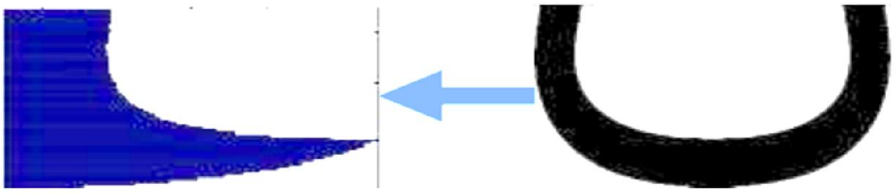


Figure 4. 13. Horizontal black ink histogram feature.

The vertical black ink histogram feature (f_v) is calculated in a similar manner.

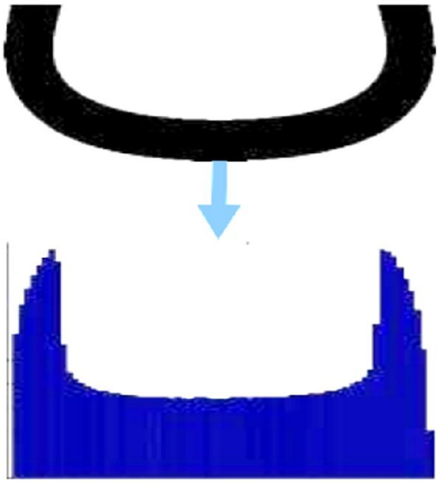


Figure 4. 14. Vertical black ink histogram feature.

4.3.2. Corpus:

In our recognition system the feature vector is of size 56, represent a statistical feature.

The corpus is a text file that has a particular structure to be used directly by the LIBSVM application. Each line of the corpus represents the characteristic vector of an image, the first value of the line is the number of the class, and then each characteristic vector value is preceded by an index (Index: Value).

4.4. Classification:

SVM is used to make the decision by assigning each input character images into its desired writer, a MATLAB toolbox implementing SVM is freely available for academic purposes:

- Download it from: the official website[2].
- Extract the tar file svm.tar under the matlab toolbox directory;
- Add .../matlab/toolbox/svm to your MATLAB path;
- Type help svm at the MATLAB prompt for help.

[2] https://www.csie.ntu.edu.tw/~cjlin/libsvm/

5. Presentation of the Application:

5.1. The name of the application:

We chose the name **AOCR** (Arabic optical character recognition) name of our application.

5.2. Interface of the application:

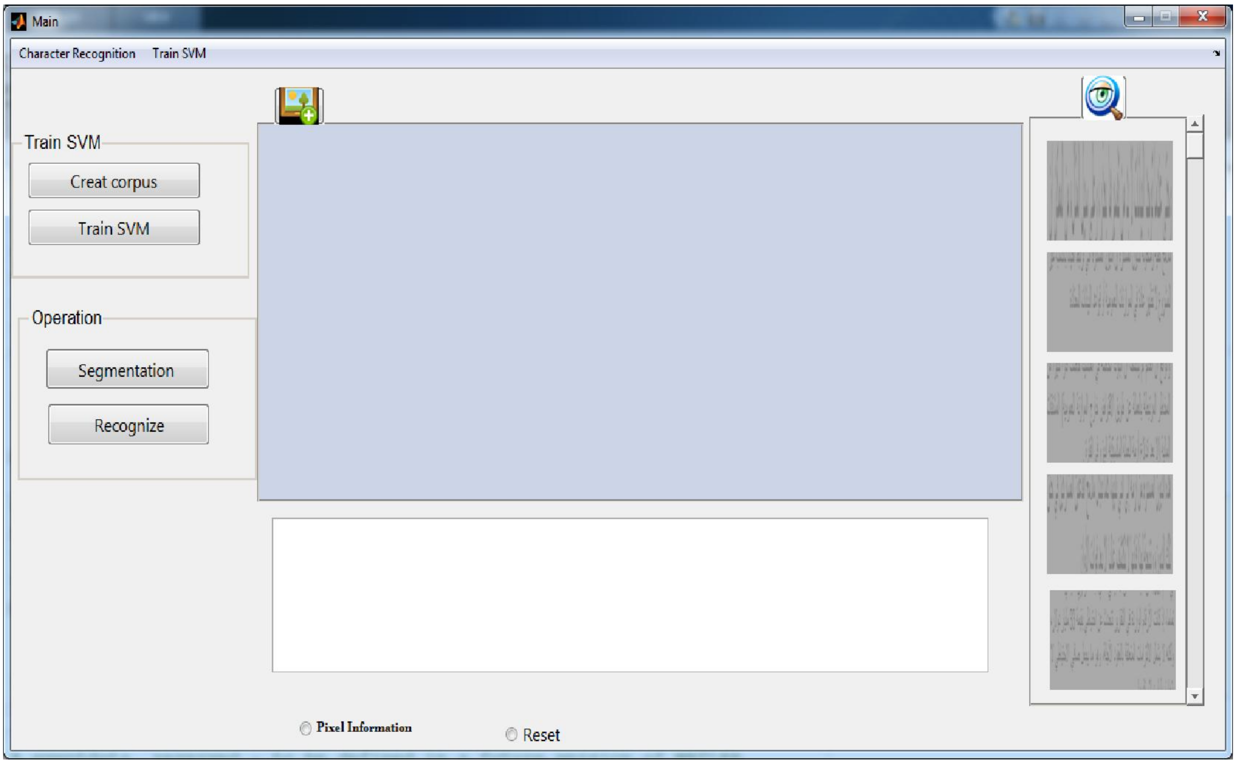


Figure 4. 15. Interface of The application.

6. Experimental and Results:

The phase of recognition is accomplished by two main parts. The first is performed in letters while the second is performed in text using **SVM** Kernel RBF Function classifier with parameters one against all, σ =0.025, and C =1000.

6.1. Character Recognition:

The used Primitive Arabic Characters in our data base contain five different font types, each one contain all 28 Arabic characters as well as lam-alif ligature in all of its possible forms

(Isolated, Start, Middle and End). In each shape we use 43 shape for train and 18 shape for test. Each font type in the dataset contains 100 primitive shapes.

We tested the combined features on the dataset using 70-30 strategy, which means 70% of dataset used for training and 30% used for testing, That means in each font there is 4300 sample for train and 1800 sample for test. For each class we use 43 shape for train and 18 shape for test.

The used evaluation metric is the Character Recognition Rate (CRR):

$$\boldsymbol{CRR} = \frac{\boldsymbol{Number\ Of\ Tested - \ Number\ \ Of\ Failure}}{\boldsymbol{Number\ Of\ Tested}} \times \mathbf{100\%}$$

A. Experience one:

In the first experimental we use 4300 sample for train and 1800 sample for test of the Andalus Font. For each class we use 43 shape for train and 18 shape for test:

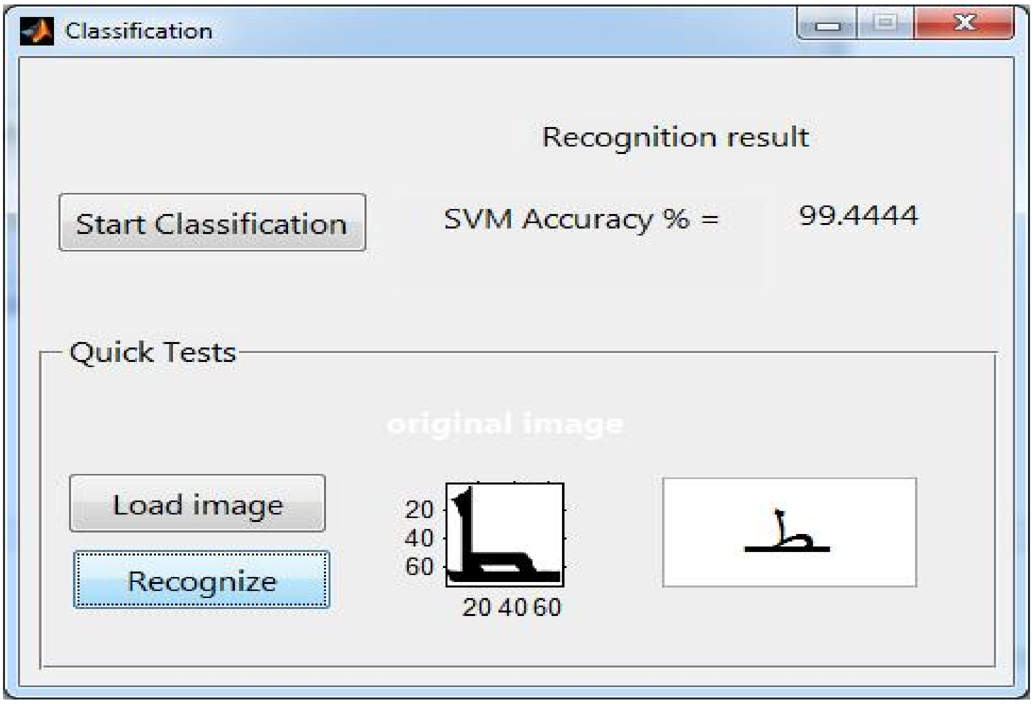


Figure 4. 16. Results of Classification for characters recognition using Andalus font.

- Results:

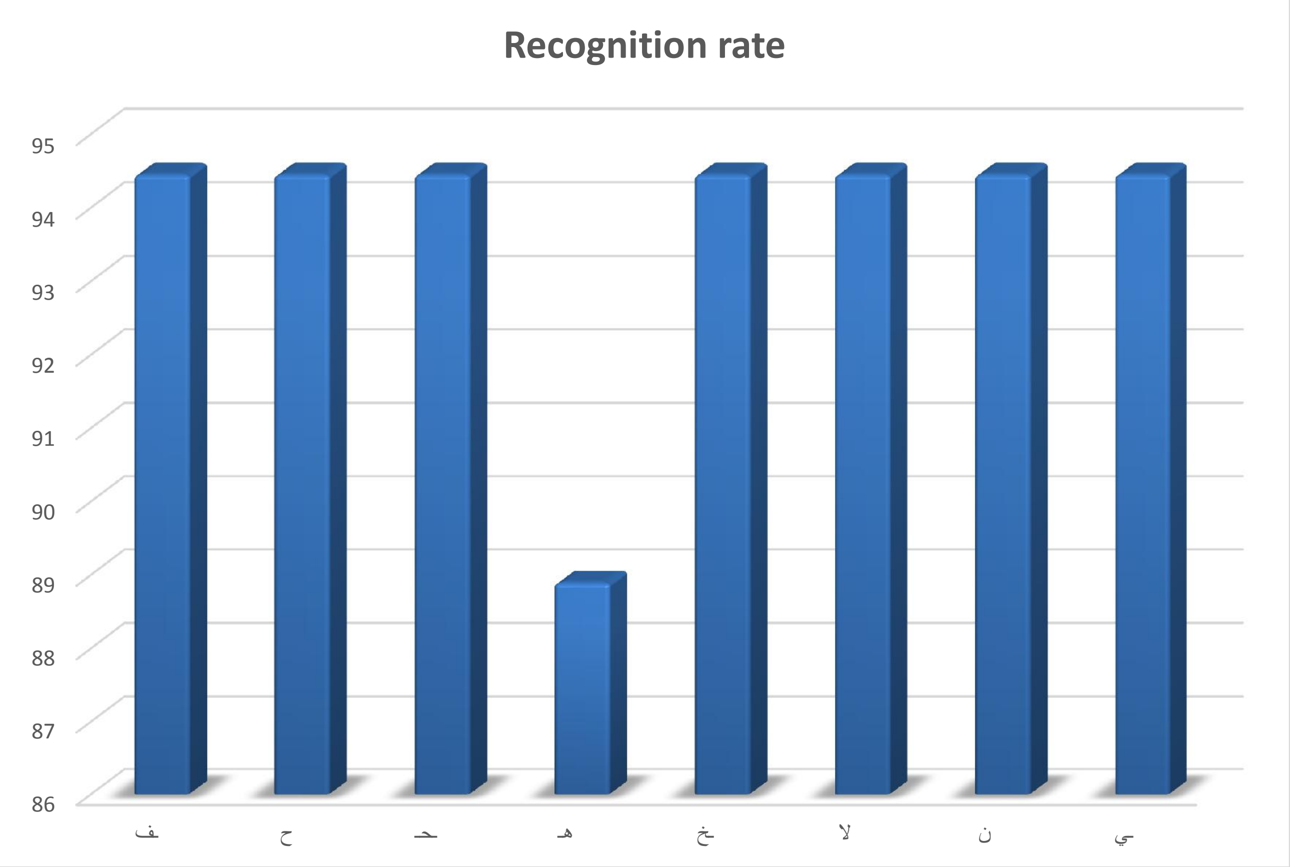


Figure 4. 17 . Histogram of recognition rates of some class in Andalus font type.

The results of classification of Andalus font was perfect (CRR =100%), except some class which are very good recognized as it shown in the Figure 4.17.

B. Experience two:

In this experience we use 4300 sample for train and 1800 sample for test of the Traditional Arabic Font. For each class we use 43 shape for train and 18 shape for test:

➢ Results:

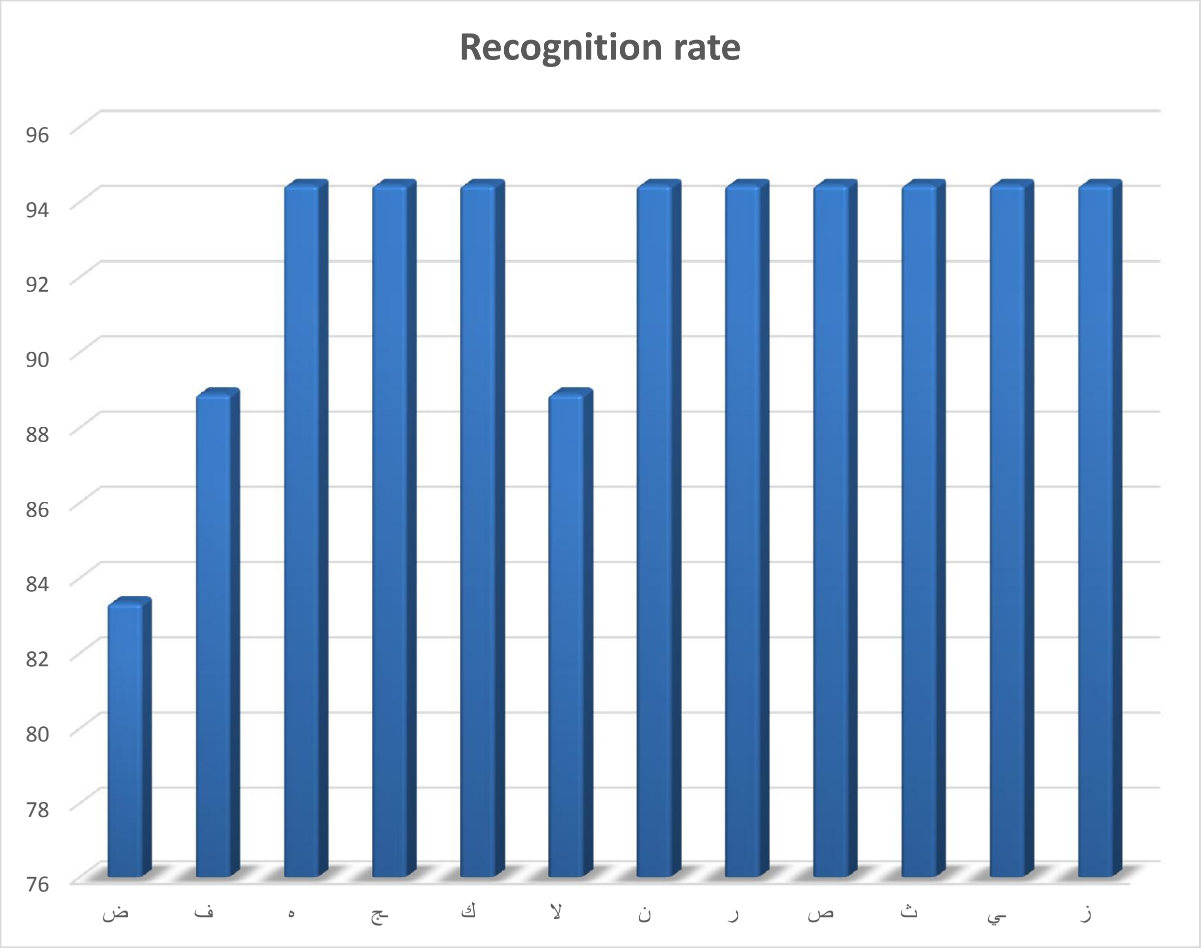


Figure 4. 18. Histogram of recognition rates of some class in Traditional Arabic font type.

The Figure 4.18 illustrate, the class that not achieve (CRR =100%) of classification of Traditional Arabic font.

C. Experience three:

In this experience we use 4300 sample for train and 1800 sample for test of the Arial Font. For each class we use 43 shape for train and 18 shape for test:

- **Results:**

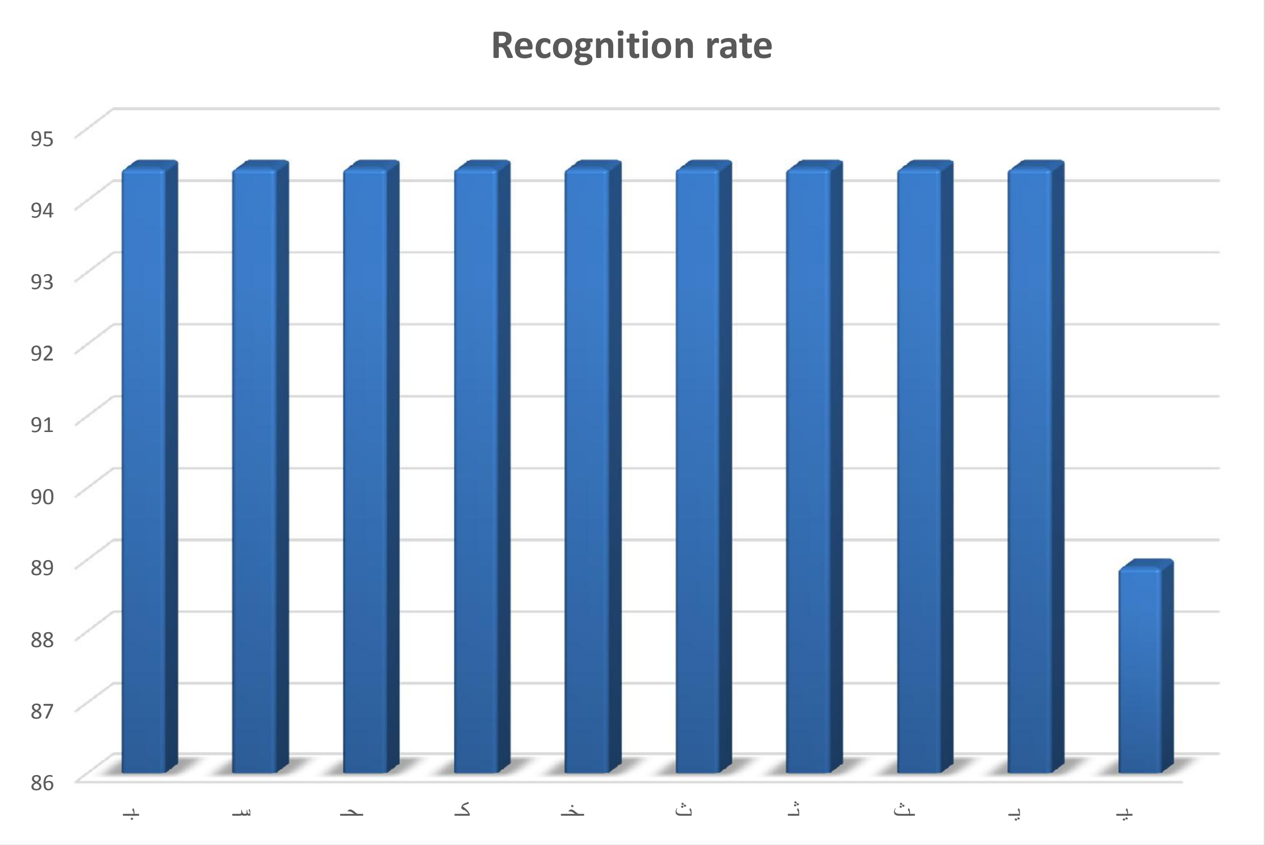


Figure 4. 19. Histogram of recognition rates of class in Arial font type.

The results of classification of in Arial font the results was excellent (CRR =100%) except some class presented in the Figure 4.19 which are very good recognized.

D. Experience four:

In this experience we use 4300 sample for train and 1800 sample for test of the Times New Roman Font. For each class we use 43 shape for train and 18 shape for test:

- **Results:**

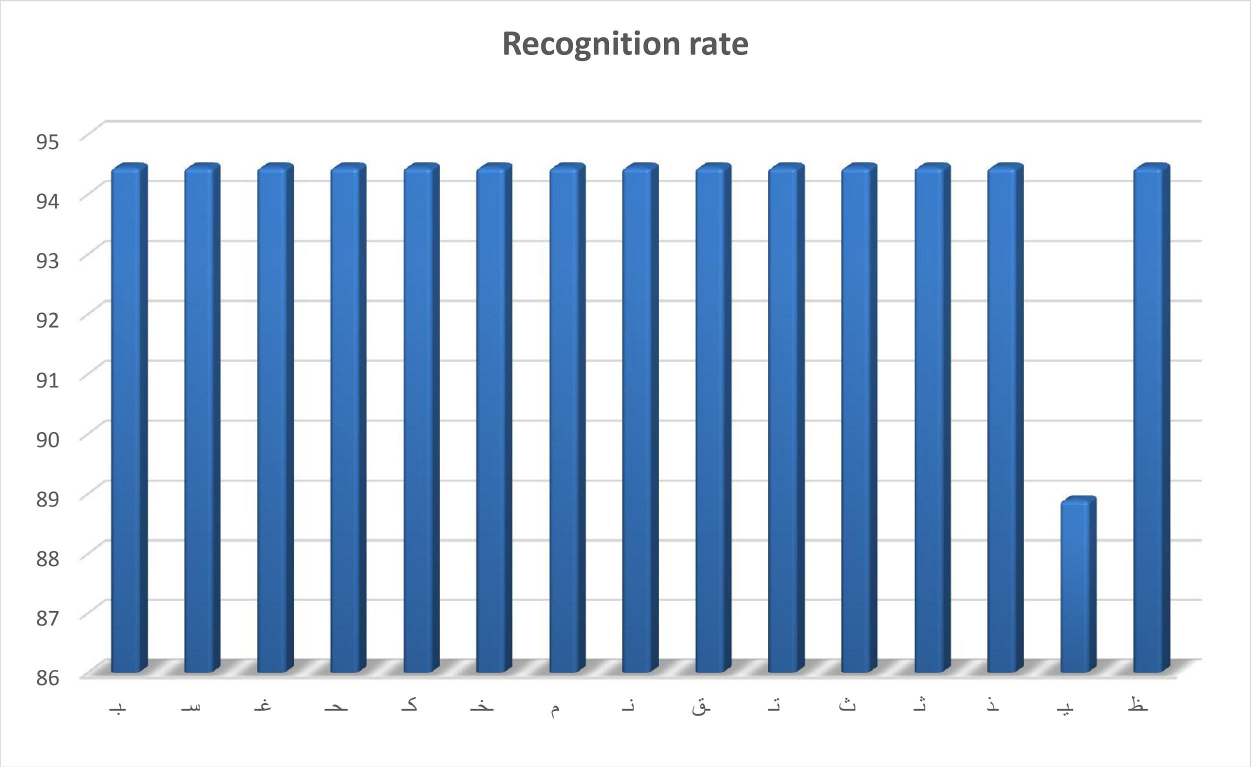


Figure 4. 20. Histogram of class recognition rates in Times New Roman font type.

The Figure 4.20 illustrate, the results of class that not achieve 100% CRR of classification of in Times New Roman font. As compared with the results of the three font classification presented before the results was very similar. Perfect results are obtained except some class presented in the Figure which are very good recognized.

E. Experience five:

In this experience we use 4300 sample for train and 1800 sample for test of the Simplified Arabic Font. For each class we use 43 shape for train and 18 shape for test:

- **Results:**

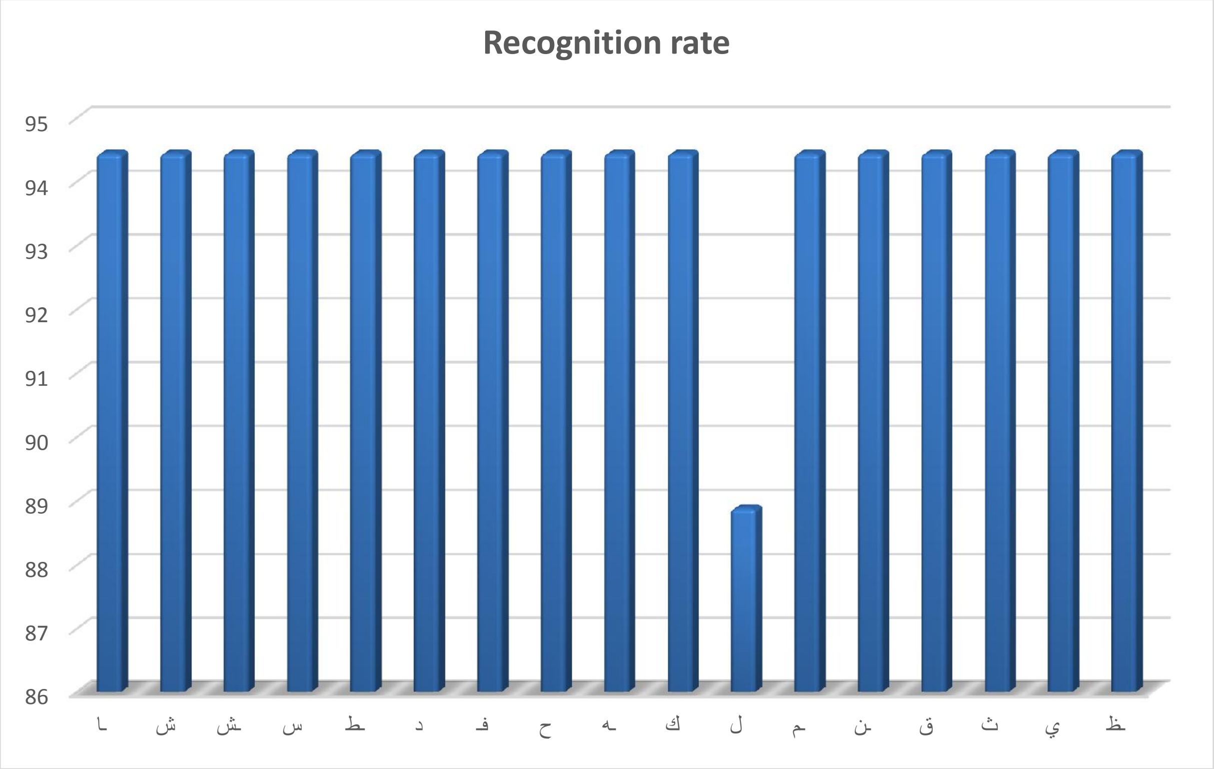


Figure 4. 21. Histogram of recognition rates of some class in Simplified Arabic font type.

The results of classification of in Simplified Arabic font the results was similar to the four font classification presented before. Figure 4.21 illustrate the class that not achieve 100 % of the recognition rate.

➢ Results of testing:

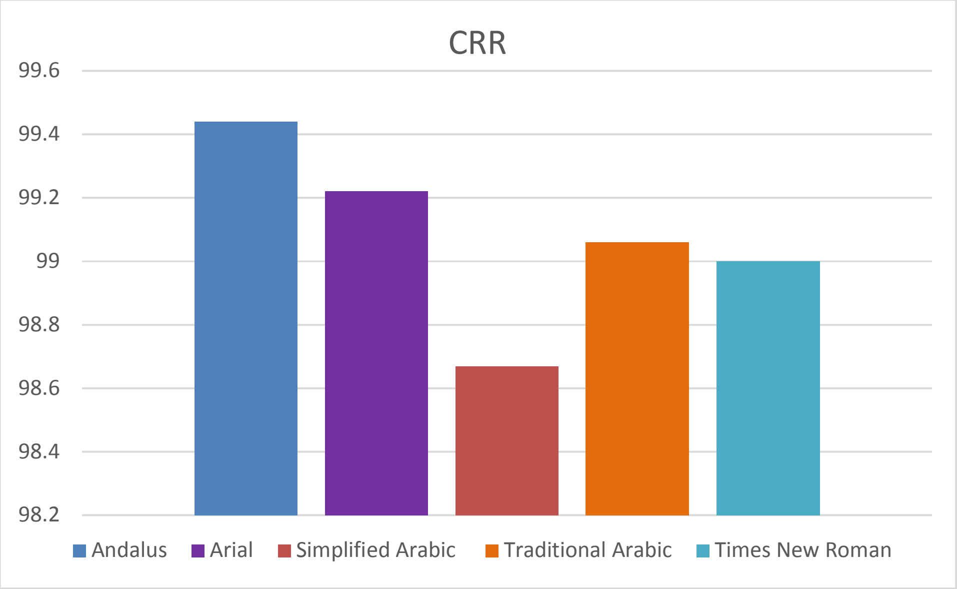


Figure 4. 22. Character recognition rate evaluation using *5* different font types.

6.1.1. Discussion:

The obtained results in CRR metric using the statistical features is shown in Figure 4.22. The figures illustrates the results of Character recognition rate using 5 different font types.

The statistical features achieved CRR of 99.44 % in Andalus type, 99.22% in Arial type, 98.667% in Simplified Arabic, 99% in Times New Roman and 99.05% in Traditional Arabic. So the average recognition ratio for the 5 tested font types is **99.08%** using **SVM** Kernal RBF Function classifier with parameters one against all, σ =0.025, and C =1000.

The most errors occurred when the written size is very small size between 12-14. Also the unrecognized letters differ based on the used font type; meaning, the misrecognized letters using one font type are not identical to the misrecognized letters using anther one font type, which cause huge difficult to define a set of misrecognized characters. From the obtained results of using feature vector of 56 characteristic a very good results were achieved.

6.2. Text Recognition:

6.2.1. Segmentation:

a. Results Of Line Segmentation

Table 4. 2. Line segmentation results for regular style with different font and sizes.

Font Name	Total number of lines	number of lines correctly segmented	Accuracy
Times New Roman	*71*	*71*	*100%*
Arial	*68*	*68*	*100%*
TOTALS	*139*	*139*	***100%***

As it's appear in the previous table the results of line segmentation it is perfect since we achieved the accuracy 100%, after it was less due to the over segmentation problem, where the second component (dots) was considered as an independent line; which cause as two lines, the first one contain the a sequence of words some of them contain letters without dots, and a second line contain only the lost dots, instead of one line gathering the both.

b. Results of Sub Word Segmentation:

Table 4. 3. Sub Word segmentation results for regular style with different font and sizes.

Font Name	Total number of sub word	number of sub word correctly segmented	Accuracy
Times New Roman	*2301*	*2269*	*98.61%*
Arial	*2192*	*2133*	*97.31%*
TOTALS	*4493*	*4402*	***97.97%***

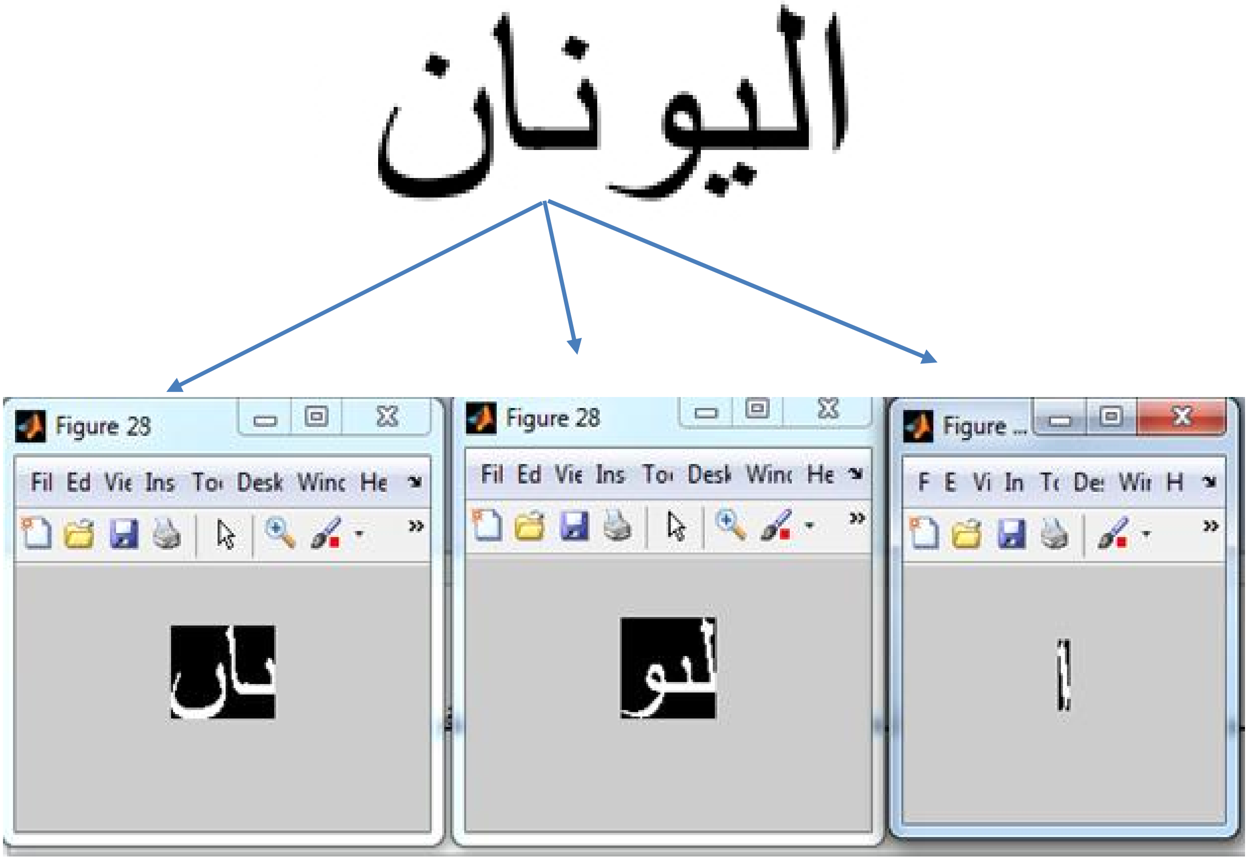


Figure 4. 23. Example of under Segmentation problem in sub word Segmentation.

In this phase we chose to apply the segmentation over sub word instead of word to handle less possible number of connected letters at a time in character segmentation;

The results it's good by achieving the accuracy of 97.97%. Thought. The accuracy of 2.03% of non-segmented sub word, and that due to one main reason which is the overlap between two sub words as it's shown in the figure 4.23 where the vertical projection between the sub words is differ from the value 0, which violates the condition of sub word segmentation. By observation, this issue appear mostly when the letter " أ" is involved and in the middle of the word followed by certain characters such as **"ت", " د"," صـ"**.

c. Results of Character Segmentation:

Table 4. 4. Character segmentation results for regular style with different font and sizes:

Font Name	**Total number of Character**	**number of Character correctly segmented**	**Under segmentation**	**Over Segmentation**	**Accuracy**
Times New Roman	*3927*	*3653*	*4.7%*	*2.28%*	*93.02%*
Arial	*3651*	*3393*	*4.46%*	*2.61%*	*92.93%*
TOTALS	*7623*	*7046*	*4.58%*	*2.44 %*	***92.97%***

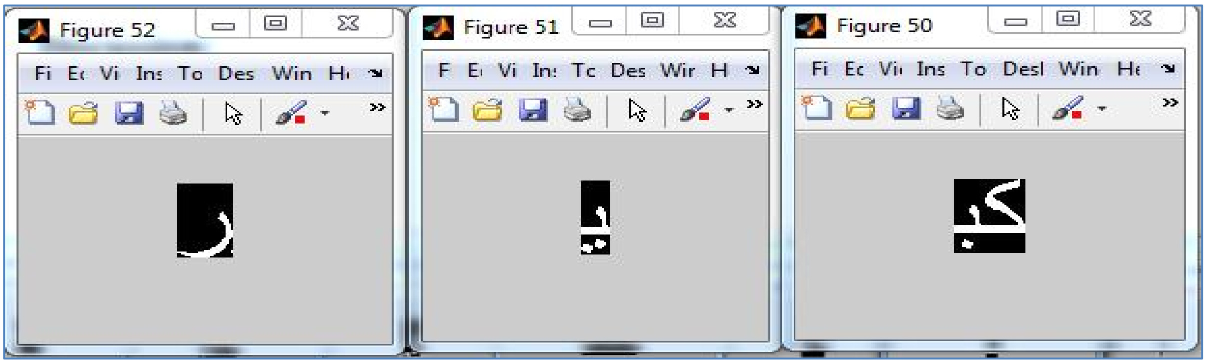


Figure 4. 24. Example of under Segmentation problem in Character Segmentation.

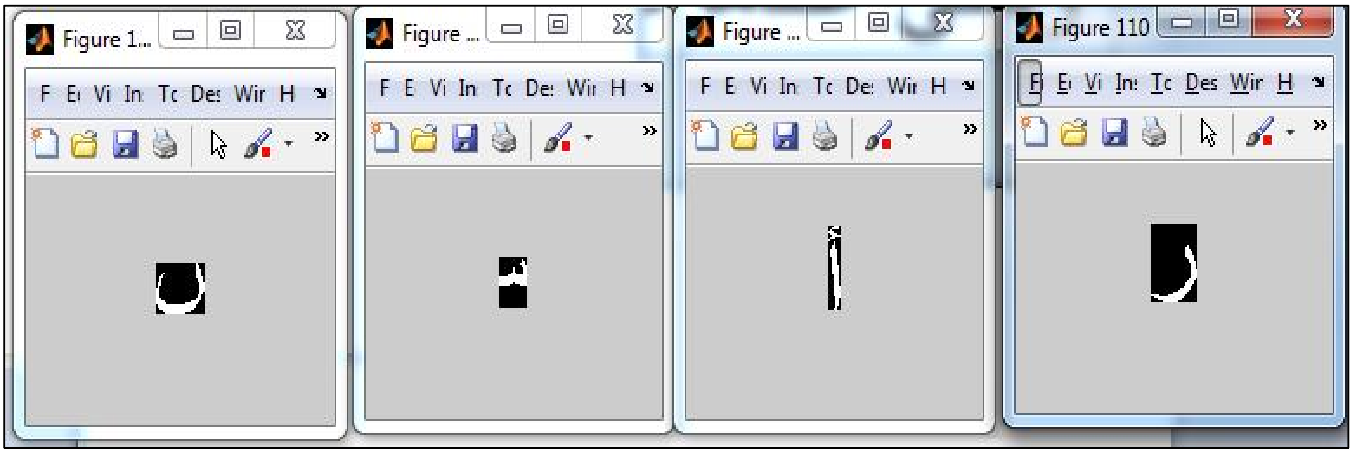


Figure 4. 25. Example of Over Segmentation problem in Character Segmentation.

After the sub word segmentation each sub word passed to next level, which is known as the character segmentation. However, in this level the sub word need to be divided to set of isolated characters. This level considered as the most difficult level in the entire segmentation process as well as the hard operation to be applied due to Arabic writing been cursive.

The study shows that the results of the proposed algorithm of segmentation characters is good, where we achieve the accuracy of 92.97% for both Times New Roman and Arial font. Even thought, the miss segmentation problems are appears with accuracy 7.02%, where it is divided to main problem, under and over segmentation problems:

➢ The under segmentation appear with an accuracy equal to 4.58%, the main reason of this issue back to overlap problem, where the shape of letters interfere with each other vertically, which makes the separate point over the junction line hard to defined (see Figure 4.24).

The under segmentation problem appear the most when the character "كـ " is involved in both start and the middle followed by certain characters such as " تـ ", " سـ "," بـ ". Furthermore, the other reason of the under segmentation issue back to the miss segmented sub word in the previous level.

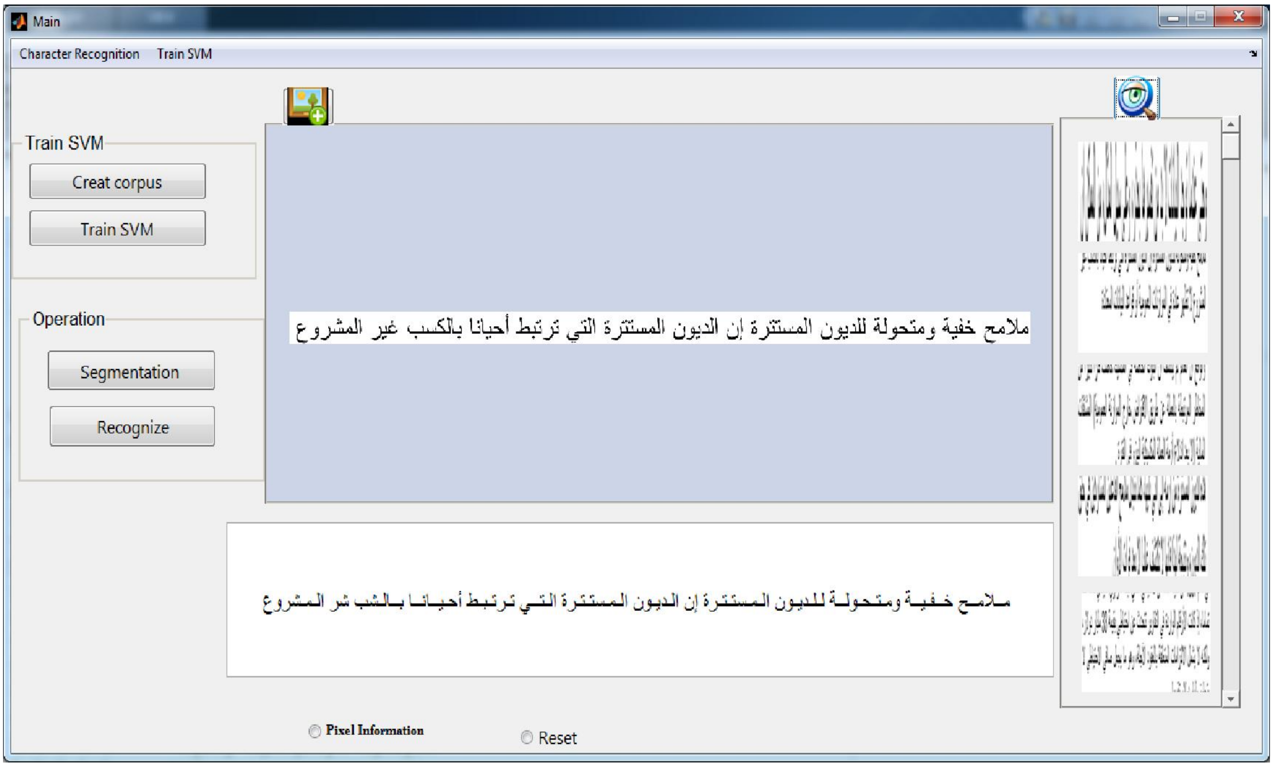
➢ The over segmentation problem appear with an accuracy equal to 2.44 %, the main reason of this issue back to the shape of characters itself, where the character shape appear in certain way that's look like it contain a junction line which cause us two part of the same character instead of a completed one.

6.2.2. Text recognition:

a. Results of recognition:

Table 4. 5. Results of text recognition for Times New Roman and Arial.

Font Name	Total number of Character	number of Character correctly classified	Accuracy
Times New Roman and Arial	*7623*	*7006*	***91.90%***



ملامح خفية ومتحولة للديون المستترة إن الديون المستترة التي ترتبط أحيانا بالكسب غير المشروع

ملامح خفية ومتحولة للديون المستترة إن الديون المستترة التي ترتبط أحيانا بالشب شر المشروع

Figure 4. 266. Example of text recognition.

The table 4.5 above represent the results of recognition characters after segmentation, which achieve the accuracy 91.90%, as we notice this results is very similar to the accuracy obtained from characters segmentation due to the strong relation between the two phases.

Meaning, the text recognition is directly affected by the correctness and incorrectness of sub word and character segmentation.

7. Conclusion :

In this chapter we have presented a detailed description of our Arabic Printed recognition system with the performance evaluation of the segmentation phase and recognition.

We develop Arabic Optical Character Recognition (AOCR) system that has four stages: preprocessing, segmentation, feature extraction, and classification.

In the proposed system the segmentation algorithm used based on the projection profile which consists of three levels of segmentation: lines segmentation, sub-words segmentation and characters segmentation:

For line segmentation stage the over segmentation problems are addressed and solved, the sub word segmentation is performed by applying vertical projection, the proposed method also determines if the sub-words are related to the same word or to different words regardless to the font type or size by estimating the pen size for each line.

In the feature extraction stage, 56 features are extracted, in the classification stage the input features that obtained from feature extraction stage and classified using SVM classifier.

We presented the results of experiments for the two phases (segmentation and recognition) using:

- Segmentation algorithm based on projection;
- SVM multiclass (One-against-all) with Gaussian Kernel (RBF).

Conclusion and Future Work

Optical Character Recognition (OCR) is a technique which converts image of printed, scanned text into a text understandable by machine. Character recognition of off-line Arabic machine printed script remains a challenging problem in pattern recognition

In our work we present our off-line printed Arabic Optical Character Recognition system and evaluate it.

In Character Recognition, the problem of the recognition accuracy of Arabic optical character recognition is addressed. The proposed approach focus on enhancing recognition accuracy of AOCR through using effective features that are capable of increasing the recognition accuracy through its ability to extract information of the characters.

A new Primitive of Arabic Characters dataset is proposed using deferent font and size for evaluation and testing purpose. Experimental results showed that the combined feature achieved excellent results up to **99.08%** using **CRR** metric.

In text Recognition, a new segmentation method based on projection profile is proposed. Results using the proposed segmentation method show encouraging results, The major problem making the task crucial is the character segmentation process, because the nature of the Arabic writing. The segmentation method used achieve **100%, 97.97%, 92.97%** for line, word and character segmentation respectively.

Segmenting Arabic manuscripts into text lines, words and characters is an important step to make recognition systems more efficient and accurate. Correct character segmentation leads to the correct character recognition, thus segmentation stage is very important process. So the segmentation of Arabic text is error-prone. It is the stage where most of the errors occur and where the error in segmentation will result in classification errors.

For phase of classification we use LIBSVM Kernel RBF Function classifier with parameters one against all, σ =0.025, and C =1000.

There are different defining problems are still need to resolve for the high accuracy of Optical Character Recognition (OCR) of printed Arabic script Therefore as a future work we suggest to:

- Solve the problem of overlapping sub word segmentation;
- Solve the problem of overlapping and under segmentation between Characters;
- Use post processing of image by the use of dictionary, the unrecognized letters produces spelling mistake of word, this words are further passed to dictionary for the correction of words.

Bibliography

[1] S.Naz and al, “The optical character recognition of Urdu-like cursive scripts”, Pattern Recognition, vol. 47, no. 3, 2014, pp. 1229-1248.

[2] M.S.Khorsheed, “Off-line Arabic character recognition--a review” , Pattern analysis and applications, vol. 5, no. 1, pp. 31-45, 2002.

[3] A.M.AL-Shatnawi, S.AL-Salaimeh, F.AL-Zawaideh, and K.Omar, “Offline arabic text recognition--an overview ” , World of Computer Science and Information Technology Journal (WCSIT), vol. 1, no. 5, pp. 184-192, 2011.

[4] Honey Mehta et al, Sanjay Singla and Aarti Mahajan, “ Optical Character Recognition (OCR) System for Roman Script & English Language using Artificial Neural Network (ANN) Classifier ” , International Conference on research advances in integrated Navigation Systems(RAINS-2016), April 06-07, 2016, R.L. Jalappa Institue of Technoloy,8 Doddaballapur, Bangalore, India.

[5] Vivek Kumar Verma, Pradeep Kumar Tiwari, “ Removal of Obstacles in Devanagari Script for Efficient Optical Character Recognition”, International Conference on Computational Intelligence and Communication Networks, 2015.

[6] Anwar Ali Sanjrani et al, “Handwritten Optical Character Recognition System for Sindhi Numeral”, 2016.

[7] Mohamed Dahi, Noura A. Semary, Mohiy M. Hadhoud , “A Comparative Study of Different Approaches of Primitive Printed Arabic Optical Character Recognition”,2015.

[8] Mohamed Dahi, Noura A. Semary, and Mohiy M. Hadhoud, “ Primitive Printed Arabic Optical Character Recognition using Statistical Features” ,2015.

[9] Duda, R. O., & Hart, P. E. (1973). “Pattern classification and scene analysis” Vol. New York: Wiley.

[10] Vapnik V, “The nature of statistical learning theory”. Springer Science & Business Media, 2000.

[11] Duwairi RM, Marji R, Sha'ban N, Rushaidat S , " Sentiment Analysis in Arabic tweets ", In: the 5th International Conference on Information and Communication Systems (ICICS), 2014.

[12] Abdulla1 NA, Al-Ayyoub M, Al-Kabi MN. "An extended analytical study of Arabic sentiments", International Journal of Big Data Intelligence, 2014.

[13] Smola AJ, Schölkopf B. "A tutorial on support vector regression", Statistics and computing, 2004.

[14] Wang, S. C, "Artificial neural network", In Interdisciplinary Computing in Java Programming Springer U. 2003.

[15] Walid Cherif. Abdellah Madani and Mohamed Kissi, "A combination of Low-level light stemming and Support Vector Machines for the classification of Arabic opinions".

[16] Nafiz Arica and Fatos T. Yarman-Vural, "An Overview of Character Recognition Focused on Off-Line Handwriting" , IEEE transactions on systems, man, and cybernetics, vol. 31, no. 2, may 2001.

[17] Swital J. Macwan and Archana N. Vyas. "Classification of Offline Gujarati Handwritten Characters", 2015 International Conference on Advances in Computing, Communications and Informatics (ICACCI).

[18] Amit Choudhary, Rahul Rishi, Savita Ahlawat, "A New Character Segmentation Approach for Off-Line Cursive Handwritten Words", 2013 International Conference on Information Technology and Quantitative Management.

[19] [1] Rodolfo P. dos Santos, Gabriela S. Clemente, "Text Line Segmentation Based on Morphology and Histogram Projection" , in 10th International Conference on Document Analysis and Recognition, 2009.

[20] M. Maloo and K. V. Kale, " Gujarati Script Recognition: A Review ", in International Journal of Computer Science Issues, Vol 8, July 2011.

[21] M. Thungamani and P. Ramakhanth Kumar, "A Survey of Methods and Strategies in Handwritten Kannada Character Segmentation", in International Journal of Science Research, Vol 01, issue 01, June 2012, pp. 18-23.

[22] Gupta Mehula, Patel Ankita, Dave Namrata, Goradia Rahul and Saurin Sheth, “Text Based Image Segmentation Methodology” , in 2nd International Conference on Innovations in Automation and Mechatronics Engineering, ICIAME 2014.

[23] S. Kumar and C. Singh, “A study of Zernike moments and its use in Devangari handwritten character recognition”, Proceedings of the International Conference on Cognition and Recognition, pp. 514-520, 2005.

[24] L.I. Kuncheva, “Combining Pattern Classifiers: Methods and Algorithms”, John Wiley Sons, New Jersey-United States of America (USA), 2004.

[25] Dzulkifli Mohamad, Ghazali Sulong, “Implicit Vs Explicit based Script Segmentation and Recognition: A Performance Comparison on Benchmark Database”, Int. J. Open Problems Compt. Math., Vol. 2, No. 3, September 2009.

[26] A. M. Zeki, M. S. Zakaria, and C.-Y. Liong, “ Segmentation of Arabic characters: A Comprehensive Survey ”, International Journal of Technology Diffusion, vol. 2, no. 4, pp. 48 –82, 2011.

[27] M. Shafii, “ Optical Character Recognition of Printed Persian/ Arabic Documents” , Ph.D. dissertation, department of Electrical and Computer Engineering, University of Windsor, 2014.

[28] S. Naz, A. I. Umar, S. H. Shirazi, S. B. Ahmed, M. I. Razzak, and I. Siddiqi, “Segmentation techniques for recognition of arabic-like scripts: A comprehensive survey” , Education and Information Technologies, Feb. 2015.

[29] M. Al-A’ali and J. Ahmad, “ Optical character recognition system for arabic text using Cursive multi-directional approach”, Journal of Computer Science, vol. 3, no. 7, pp. 549–555, Jul. 2007.

[30] B. E. Boser, I. M. Guyon, and V. N. Vapnik, “A training algorithm for optimal margin classifiers”, in Proceedings of the fifth annual workshop on Computational learning theory. ACM, 1992, pp. 144–152.

[31] Wissam Aoudi and Aziz M. Barbar, “Support Vector Machines: A Distance-Based Approach to Multi-Class Classification”, IEEE International Multidisciplinary Conference on Engineering Technology (IMCET), 2016.

[32] Madan Somvanshi and al. “A Review of Machine Learning Techniques using Decision Tree and Support Vector Machine”, IEEE International Multidisciplinary Conference on Engineering Technology (IMCET), 2016.

[33] Marwa AMARA and Kamel ZIDI, “A comparative study of multi-class support vector machine methods for Arabic characters recognition”, IEEE International Conference, 2016.

[34] T.M. Wahbi, M.E.M. Musa, and I.M. Osman, “ On finding the best number of states for a HMM-based off-line Arabic word recognition systems ”, The International Arab Conference on Information Technology (ACIT), Riyadh- Saudi Arabia, 2011.

[35] M. Rashad and N. Semary, “Isolated Printed Arabic Character Recognition Using KNN and Random Forest Tree Classifiers”, in Advanced Machine Learning Technologies and Applications, Springer, 2014, pp. 11-17.

[36] A. Rosenberg and N. Dershowitz, “Using SIFT Descriptors for OCR of Printed Arabic”, Tel Aviv University, 2012.