Jurnal Teknologi, 36(E) Jun 2002: 1–18

A SURVEY AND COMPARATIVE EVALUATION OF SELECTED OFF-LINE ARABIC HANDWRITTEN CHARACTER RECOGNITION SYSTEMS

KASMIRAN JUMARI[1] & MOHAMED A. ALI[2]

Abstract. In this study we tried to cover the Optical Character Recognition (OCR) systems used for off-line Arabic Optical Text Recognition (AOTR). We cast some light on the characteristics of Arabic writing. This paper will also present a general concept on the different stages normally followed for handwritten character recognition: preprocessing, segmentation, feature extraction, classification, and post-processing. In addition, evaluation methods for different AOTR systems are presented.

Keywords: Off-line character recognition, handwritten recognition, Arabic Optical Text Recognition (AOTR), OCR Evaluation

Abstrak. Kajian ini akan melihat Sistem Pencaman Aksara Optikal bagi Pencaman Teks Optikal Arabik (Arabic Optical Text Recognition (AOTR)) secara luar talian. Ia dilakukan dengan meneliti ciri-ciri tulisan Arabik. Kertas ini juga akan memberikan konsep am bagi langkah-langkah yang biasanya dijalankan dalam pencaman aksara tulisan tangan seperti prapemprosesan, pensegmenan, perolehan ciri-ciri tersendiri, pengkelasan dan pemprosesan lanjut. Sebagai tambahan, beberapa kaedah lain bagi sistem AOTR juga akan diberikan.

Kata Kunci: Pencaman aksara luar-talian, pencaman tulisan tangan, pencaman Teks Optikal Arabik, penilaian Pencaman Aksara Optikal (OCR)

1.0 INTRODUCTION

Optical Character Recognition, in general, is becoming more intensive than before, in particular Arabic Character Recognition. Commercial systems for Arabic Optical Text Recognition (AOTR) are becoming more available. Paper prevailed as a medium for writing since the advent of writing as a form of communication. Only recently, electronic media has started to replace paper, by which time and space are conserved.

The oldest familiar application of optical character recognition (OCR) was using this technique for checks sorting in banks. The vast applications of automatic reading are numerous (1993). Applications like: zip code reading, mail sorting, providing

[1] Fakulti Kejuruteraan Universiti Kebangsaan Malaysia, Bangi 43600, Malaysia, (kbj@pkrisc.cc.ukm.my)

[2] Fakulti Kejuruteraan Universiti Kebangsaan Malaysia, Bangi 43600, Malaysia, (fadeel1@eng.ukm.my)

assistance to blind people to read, reading of customer filled forms, automating office archiving and retrieving text, and improving human-computer interfaces (*e.g.* pen-based computers (Ullmann. J, 1982; Govindan. V and A. P. Shivaprasad, 1990)) require optical character recognition. The interesting nature of the optical character recognition, as well as, its importance have created a lot of research orientation in different aspects and gained numerous advances.

Most of the published researches on OCR have been on Latin characters, with some researches on Chinese and Japanese character recognition emerging in the mid-1960s. A survey on on-line handwritten recognition is carried out by Wakahara *et al.* 1990; while Bozinovic and Srihari 1989; Simon 1992 made a survey on off-line cursive word recognition. Mori *et al.* (1984) and Stallings (1976) provided survey on machine recognition of hand-printed characters and Chinese character recognition, respectively. Even though there is about half a billion people worldwide in several different languages using Arabic alphabet for writing (like Arab, Persian, Urdu and Jawi), yet, Arabic character recognition has not been covered as perfectly as Latin, Chinese, or Japanese. Nazif (1975) has apparently published the first work on Arabic Optical Text Recognition (AOTR). In the late 1970s and early 1980s, Badi and Shimura (1979) and Nouh (1980) dealt with printed Arabic characters where as Amin (1980) dealt with handwritten Arabic characters. A considerable number of researches on AOTR have been carried out during 1980s, a trend that continues in the 1990s.

2.0 ARABIC OPTICAL TEXT RECOGNITION (AOTR) SYSTEM

Arabic Optical Text Recognition (AOTR) is a branch of the main trunk, Optical Character Recognition (OCR). It has been reported that the first published work on AOTR is back in 1975 by Nazif. It was a Master's thesis in which he developed a system for recognizing printed Arabic characters based on extracting strokes, which he called radicals, and their position. There are reasons why there are not enough researches on AOTR , these reasons are introduced in the following section.

2.1 General Characteristics of the Arabic Text

Arabic writing style can be, in general, classified into; typewritten (Naskh), hand-written (Ruqq'a) and artistic, for decorative calligraphy, (Kufi, Thuluth and Diwani). Arabic writing is similar to English i.e. it uses letters, numerals, space and special punctuation and symbols. However, it is unlike English as far as representation of vowels is concerned.

There are a number of characteristics which make Arabic cursive writing unique compared to Latin, Chinese and Japanese. In addition, these characteristics give clear reason for why there are not enough researches on AOTR. These characteristics can be summarized as follow:

(i) Arabic text is written from right to left, as compared to Latin and Japanese, as shown in Figure 1 and it is cursive even if it is printed, which means each character, in a word, has a left and/or right connection point that normally lies on an imaginary line called base-line, upon which other characters of the same word and other words lie. Moreover, Arabic alphabet has no capital or small shape like in Latin. Similar to The American Standard Code for Information Interchange (ASCII) code each Arabic character has a single code in ASMO code (Arabic Standard and Metrology Organization).

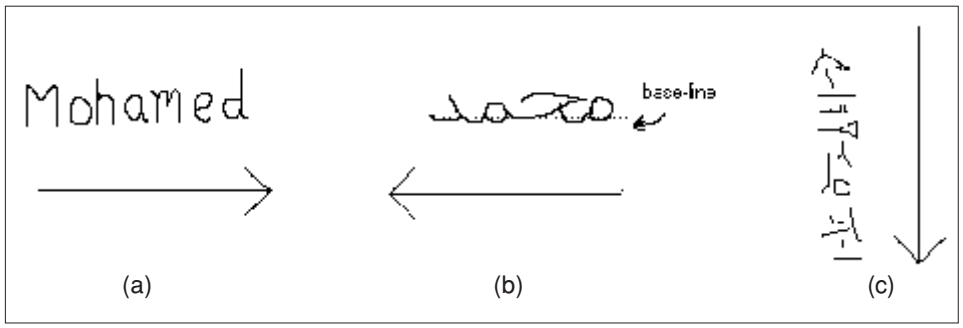


Figure 1 Direction of writing (a) English (b) Arabic and (c) Chinese

(ii) The Arabic alphabet has 28 basic characters as shown in Table 1. However, some researchers (Abohaiba *et al.*, 1993, El-Wakil, M *et al.*, 1989 and Amin A., 1997) considered Lam'Alif (ﻻ), is a combination of two characters Lam (ﻟ) and Alif (ﺍ) as another character which make the number of Arabic alphabets as 29 characters.

The shape of a character depends on its position in a word (whether the character is isolated, at the beginning, at the center or at the end of the word) hence each character may have up to four different shapes as shown in Table 2. Fifteen out of 28 basic Arabic characters have from one to three

Table 1 Arabic basic alphabet

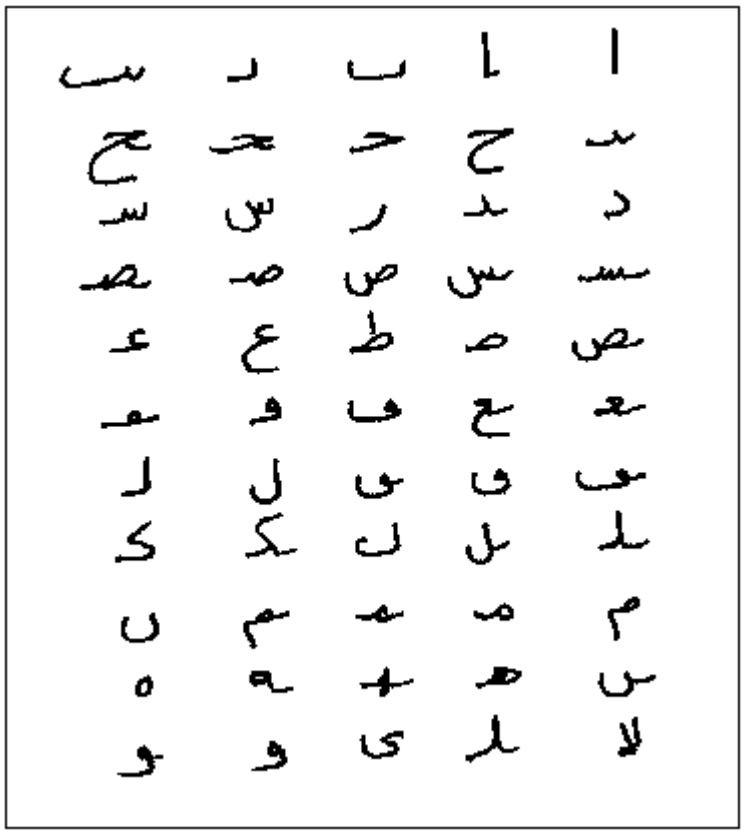
خ	ح	ج	ث	ت	ب	ا
ص	ش	س	ز	ر	ذ	د
ق	ف	غ	ع	ظ	ط	ض
ي	و	ه	ن	م	ل	ك

Table 2 Different Arabic character shapes depending on the position of character in the word

Pronunciation	As a end letter	As a middle letter	As a start letter	As isolated letter
Alif	ـا	ـا	ا	ا
Ba	ـب	ـبـ	بـ	ب
Ta	ـت	ـتـ	تـ	ت
Tha	ـث	ـثـ	ثـ	ث
Jeem	ـج	ـجـ	جـ	ج
Hha	ـح	ـحـ	حـ	ح
Kha	ـخ	ـخـ	خـ	خ
Dal	ـد	ـد	د	د
Thal	ـذ	ـذ	ذ	ذ
Ra	ـر	ـر	ر	ر
Zayn	ـز	ـز	ز	ز
Seen	ـس	ـسـ	سـ	س
Sheen	ـش	ـشـ	شـ	ش
Saad	ـص	ـصـ	صـ	ص
Dhad	ـض	ـضـ	ضـ	ض
Tta	ـط	ـطـ	طـ	ط
Ttha	ـظ	ـظـ	ظـ	ظ
Aain	ـع	ـعـ	عـ	ع
Ghain	ـغ	ـغـ	غـ	غ
Fa	ـف	ـفـ	فـ	ف
Quaf	ـق	ـقـ	قـ	ق
Kaf	ـك	ـكـ	كـ	ك
Lam	ـل	ـلـ	لـ	ل
Meem	ـم	ـمـ	مـ	م
Noon	ـن	ـنـ	نـ	ن
Ha	ـه	ـهـ	هـ	ه
Waw	ـو	ـو	و	و
Ya	ـي	ـيـ	يـ	ي

dots. These dots distinguish one character from another of the same shape. In addition three characters may have a zigzag stroke called "Hamza" (ء). For the sake of character recognition, if we omit all most similar shapes regardless of dots and Hamza we will end up with the most basic shape of Arabic characters as illustrated in Table 3.

Table 3 Basic shapes of Arabic character



(iii) Most of the 28 Arabic characters are consonant and only three are vowels; أ،و،ي (Alif, Waw and Ya respectively). Arabic language, however, utilize some other short vowels called Tashkeel (vowel diacritics). These Tashkeel namely; Fatha, Dhamma, Kasra, Sukun, Madda, Shadda and Tanween. Figure 2 shows the position of these vowels over some characters.

The presence and absence of vowel diacritics may vary the meaning of a specific word. For instance, كتب is the Arabic word for both "books", when we put "Dhamma" over letters (ت,ك and ب), and "wrote", when we put "fatha" over letters (ت,ك and ب). These vowels should be con-

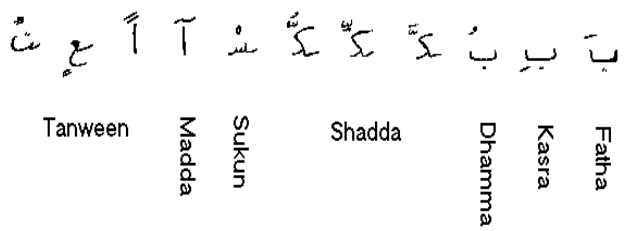


Figure 2 Vowel diacritics "Tashkeel" and their position over Arabic text

sidered in the case where a word needed to be written in a separated form. However, vowels are not always necessary in case of a full sentence where the inherent contextual information in the sentence can be used to assign the appropriate meaning.

(iv) The height and size (width) of Arabic characters are not fixed. Height and size varies across various characters and across the different shapes of the same character in different position in the word.

(v) While most Arabic characters (handwritten or printed) can be joined either from left-side, right-side or from both sides, six of Arabic characters are not connectable with succeeding character (أ د ذ ر ز و). Therefore, if one of these characters exists in a word it divides the word into two or more sub-words. Figure 3 illustrates how, one or more of those characters can divide a word into two or more sub-words.

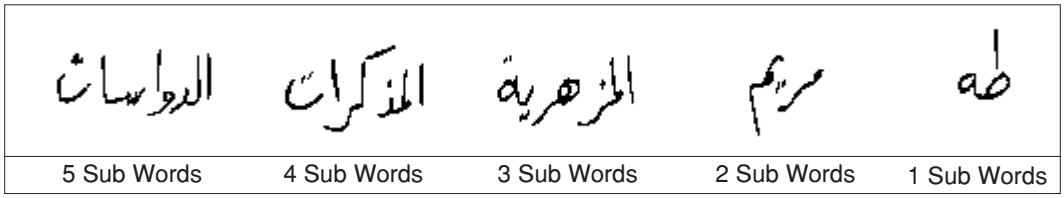


Figure 3 Arabic words may consist of one or more of sub-words

(vi) Two or three characters, specially in Arabic handwritten text, can be combined vertically forming the so called "ligature" as shown in Figure 4. The word محمد gives a good example of ligature characteristic, where three letters; meem (مـ), ha (حـ) and meem (مـ) are combined vertically forming ligature. In addition, Figure 4 shows a vertical over-lap without touching like in the word (رسول) where two letters, Ra (ر) and Seen (س) are vertically over lapped.

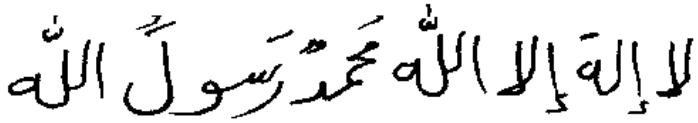


Figure 4 Arabic handwritten with ligature and over-lap

2.2 Character Recognition Systems

Character recognition system vary widely in the way they acquire their input data (text image), the mode of text being acquired (whether it is a printed or handwritten text), the characters connectivity (isolated characters or cursive word), and finally the text font (single or omnifont). In general, handwritten character recognition systems are divided according to raw data (image) acquisition into two main systems; Off-line and On-line systems.

(i) *Off-line Character Recognition System*

Off-line Character Recognition is performed after the writing or printing is complete. In typical off- line OCR system, input characters are read and digitized by an optical scanner or digital camera. However, scanner with higher resolution (200-600 dots/ inch) is recommended.

Many researchers have carried out researches concerning Off-line Character Recognition. Hyder & Koujah (1988), Habib and Bouzerdoum (1994) have developed an off-line system for handwritten isolated characters recognition. Recognition system for Arabic (Hindi) numerals has been introduced by A. Goneid *et al.*, (1992), Abdulazim and Hashish (1989), and S. H. Abbas *et al.*, (1986). Off-line character recognition systems for typewritten cursive words are introduced in Simon (1992), Saadallah and Yacu (1985) and El-sheikh (1990) whereas, off-line systems introduced in Amin (1991), El-Sheikh (1989), El-Wakil (1989) and Alimi (1995) are dealing with recognition of only isolated Arabic characters. Abdulazim and Hashish (1989) have developed an off-line bilingual (Arabic/Latin) typewritten text recognition system.

(ii) *On-line Character Recognition Systems*

On-line (sometimes called real-time or dynamic) systems have the ability, unlike the off-line system, to recognize a text in real-time using special device called tablet. Writing device captures the temporal or dynamic information which includes the number, duration, and the order of each stroke.

On-line OCR systems developed for handwritten text are limited. Nevertheless, there are some systems designed for recognition of isolated characters (Simon, 1991 and Saadallah and Yacu, 1985) and handwritten mathematical formulas (El-Sheikh, 1990 and Amin, 1991). Of course some other systems for other on-line handwritten text recognition are also developed (El-Sheikh, 1989, El-Wakil, 1989, Alimi, 1995, Al-Fakhri, 1997, Alimi and O. Ghorbo, 1995 and Bin Omar, 2000). The on-line recognition systems however is beyond the scope of this paper.

2.3 Systematic Procedures in AOTR Systems

AOTR systems for the last two decades have gone through successive steps for character recognition. Researchers have to go through those steps to recognize a handwritten text, starting with a handwriting on a sheet of paper and ending with a text file of what has been recognized, so that it can be manipulated (e.g. editing, copying, printing). In general, the five steps that researchers have used in performing character recognition, can be summarized as follow:

(i) Preprocessing
(ii) Segmentation
(iii) Feature extraction
(iv) Classification
(v) Post-processing

What follows is a discussion of the recent work carried out by different researchers using the above mentioned steps. The discussion will emphasize on handwritten Arabic character recognition.

(i) Preprocessing

Preprocessing step includes operations on the digitized image, of a raw image, intended to minimize noise and increase capability of extracting features, by cleaning and thinning the image. Those operations are namely; binarization, smoothing, thinning, alignment, normalization, and base-line detection. The accuracy of OCR systems greatly depends on the quality of the input image of the text to be recognized. Brown and Ganapathy (1983) assume that a smart preprocessing system may lead to characters uniformity which, in turn, lead to easy recognition.

Researches have been carried out concerning characterizing noise that affects input images and analyzing the effect of noise on recognition rate for OCR. Baird (1993) has determined the optical distortions that affect input document images, such as skew, speckle, and blur. Kunango *et al*. (1993) have suggested a document degradation model which model the distortion that occurs during thick bound documents photocopying. The most popular steps normally followed in preprocessing stages are:

(a) Binarization

For text recognition, binarization process converts a gray scale image into bi-level image. An effective binarization method is by computing the histogram of the gray values of the image and then finding a cut-off point. However, researchers like Pavlidis (1993) have started recognizing gray scale images without prior binarization.

(b) Filtering and Smoothing

Filtering and smoothing are conditioning steps that remove unwanted variations in the input image. Amin *et al*., (1986), Amin (1982) and Amin and Masini (1982) have studied conditioning steps for removing stray ending strokes joining two strokes when they are very close to each other. Margner (1992) has used a low-pass filter as another smoothing method for on-line text recognition.

On the other hand, off-line OCR system incorporate 3x3 window to traverse pixel-by-pixel 2-D image as a method for noise reduction (Mahmoud, 1994). Mathematical morphology is an alternative technique for smoothing. The closing operation can eliminate small holes and fill gaps on the contours, where as the opening operation can break narrow isthmuses and vanish small islands, sharp peaks, and capes (Al-Badr and Haralick, 1992). Bin Omar *et al*., (2000) have presented a way to remove the secondaries of Jawi (Arabic) characters.

(c) Thinning

Thinning is a very important preprocessing step for the analysis and recognition of different types of images. Thinning is the process of minimizing the width of a line, in an image, from many pixels wide to just one pixel (Lam *et al*., 1992). Thinning algorithms can be classified into two types; sequential algorithms (Zainodin *et. al*, 1994), and parallel algorithms (Jang and Chin, 1992).

The main difference between these two types is that sequential algorithm operates on one pixel at a time, and the operation depends on preceding processed results, while parallel algorithm operates on all the pixels simultaneously. In many applications, the latter is preferred mainly due to its simplicity and hardware feasibility (Kim *et al*., 1992). The most common thinning algorithms are based on an edge erosion technique where a window is moved over the image and a set of rules applied to the contents of the window. Abuhaiba *et al*., (1991) have used clustering based skeletonization algorithm for thinning of Arabic character. Ahmad and Ward (1998) have developed a thinning algorithm with just 16 rules. These rules aim at deleting the pixel which are north-west corner pixels or pixels that lie in the east or south boundaries of the designated pixel. The 16 rules are transformed by rotating them by 90°, 180°, and 270° to produce 64 rules which, then in one pass, are applied on every pixel. Jang and Chin (1990) have used Mathematical Morphology to design a fast thinning Algorithm. Altuwaijri and Bayoumi (1998) have proposed a thinning algorithm based on clustering the data image employing ART2, Adaptive Resonance Theory, network which is a self-organizing neural network for the clustering of Arabic characters.

Thinning algorithms are sensitive to noise, even for small variations in the input pattern. Many algorithms modify their functions so that the T-junction shape is modified to Y-junction instead (Abuhaiba *et al*., 1991). Most of the proposed algorithms perform an iterative transformation on the original pattern, however, Olszewski (1994)

used a non-iterative thinning algorithm.

A new thinning algorithm based on line sweep operation has been proposed by Chang *et al.*, (1995). A line sweep is a process where the plane figure is divided into parallel slabs by lines passing through certain "events".

(d) Normalization

Arabic characters sizes vary enormously. Therefore, a normalization method should be followed to scale characters to a fixed size and to centre the character before recognition. Fakir and Sodeyama (1993) have applied 2-D independent expansion factors to fit a character in a box of a certain size. Impedovo *et al.*, (1991) have pointed out that normalization is the procedure which adjust the character size to a standard form according to the respective steps of processing. Researchers (Hassan, 1985 and Jambi, (1992) have measured the average height of a character from the horizontal projection to approximate the average width of a character for segmentation. Blumenstein and Verma (1999) used a percentage of the average word height to approximate character width.

(e) Slant Correction

One of the most obvious measurable factors of different handwriting styles is the angle between most longer strokes in a word and the vertical direction referred to as the word slant (1989). The aim of this stage is to detect any slanted strokes. It can be achieved in two steps; slope detection and slant correction. Bozinovice and Srihari (1980) detected the slope by creating windows for every left stroke, calculated the centre of gravity for its upper and lower halves, and then they calculated the average slope. On the other hand, Motawa *et al.*, (1997) calculated the slant angle merely by a single erosion run, which only yield all the slanted strokes of the image, and claim that their algorithm proved to be faster than other algorithms. New method for evaluating and improving the linearity and accuracy of the slant estimation has been introduced by Y. Ding *et al.*, (1999).

(f) Base-line and Skew Detection

In Arabic script, base-line is defined as the line on which all letters lie. It contains useful information about the orientation of the text and provides the connection points between characters. Horizontal projection histogram is considered as one of the methods for fixing the base-line.

Base-line, in several AOTR systems, is used to detect skew alignment of that text, to perform lines separation in a text block, and to segment words into characters (Yousefi *et al.*, 1990 and Tolba and Shaddad, 1990). Fakir and Sodeyama (1993) used Hough transform to calculate the slope of the base-line and accordingly realign the text. Chin *et al.*, (1997) made a comparison study on different skew detection algorithm like Hough Transform, Projection Histogram, Method of least Squares, and

Word Centroid Least Squares in detecting skew. A fast algorithm for Skew Detection is developed by Amin *et al.*, (1995).

(ii) Segmentation

One of the main problems in Arabic text, especially its handwriting, is its cursiveness. Therefore, segmentation is a crucial step in almost all AOTR systems. After the preprocessing stage, the majority of character recognition systems perform breaking up operation, on the text to be recognized, and end up with individual character or stroke before recognition stage. Segmenting a page of text can be categorized into two levels; page decomposition and word segmentation.

Several techniques have been adopted to solve the problem of page decomposition, and they can be classified into three basic categories: top-down, bottom-up and hybrid solutions. Concerning Arabic document, page decomposition was limited to separating the different lines of a text block and extracting the sub-word. Horizontal projection histogram is a technique used for text image. In histogram, each gap (minimum value) corresponds to line-break in the text. Haj-Hassan (1985) employed the vertical projection to directly separate words from each other. The histogram technique is not suitable in cases where characters are over lapping especially in Arabic handwritten. The alternative way to solve this problem is to identify the different sub-words, by tracing their contours (1997), and then shifts them apart and inserts a blank column between them, or by tracing their skeleton (Bushofa and Spann, 1997). In Fischer *et al.*, (1995) have developed an algorithm to extract the important data from a digitized gray level image of a page of the Australian Telecom Yellow Pages.

For word segmentation, on the other hand, quite a number of researches have been conducted. Most of them are on Latin handwriting, yet a fair amount of work has been done for Arabic, Chinese and Japanese handwriting. Although some algorithms designed for Latin cursive word segmentation may carry over to Arabic handwriting, they are, generally, not adequate for that task. Shoukry (1991) has employed a sequential algorithm for the segmentation of typewritten Arabic digitized text. Al-Badr and Haralick (1992) used mathematical morphology to recognize printed Arabic without real segmentation. In another research, they emphasised their idea about character recognition without segmentation (Badr and Haralick, 1994, 1995). Romeo *et al.*, (1995) proposed two methods of character segmentation for Arabic handwritten characters and cursive Latin characters. These methods are Classical horizontal and vertical projections which detect the lowercase writing area in lines. They used contour-following algorithm to resolve the problem of overlapping lower or upper strokes. Motawa, Amin, and Sabourin applied their segmentation algorithm, based on Mathematical Morphology, on Arabic cursive script Motawa *et al.*, (1997). They carried out the segmentation in three steps; (i) filtering to reduce noise, suppress small islands, narrow channels, and block connectors. (ii) obtaining singularities (iii) subtracting the

singularities from the original image to get the regularities. Researchers proposed a new segmentation algorithm based on neural network (Blumenstein and Verma, 1999).

(iii) Feature Extraction

The next step after segmentation stage is the feature extraction in which the character produced in segmentation stage is used to extract some features which in turn passed to the next stage, the classifier. Features can be categorized into; global transformations, structural features, statistical features, and template matching and correlation (Badr and Mahmoud, 1995). The features can be manipulated in two ways: (i) interleaved control, in which an OCR system alternates between feature extraction and classification by extracting a set of features from a pattern, pass them on to the classifier then extract another feature and so on (Abdulazim and Hashish, 1989, Desouky *et al*., 1992). (ii) One step control, in which an OCR system extracts all the required features from a primitive and then make the classification (Yousefi *et al.*, 1990, Dabi *et al.*, 1990, Sheikh and Gundi, 1988).

AOTR systems, in particular, have employed different techniques in features extraction. Nazif (1995) uses template matching between the template of the radicals and the character image. Tolba *et al*., (1987) match the histograms of the input characters to those of the templates. A. Zeki and M. Zakaria (2000) introduced new primitives to reduce the effect of noise for handwritten features extraction.

(iv) Classification

It is the main decision stage of the OCR system in general. In this stage the features extracted from the primitive is compared to those of the model set. There are three methods to achieve the classification namely; structural, statistical, and using neural networks. El-Sheikh (1989), Haj-Hassan (1985), and Margner (1992) have used the first method where as Margner (1992) and Al-Emami and Usher (1990) have employed the second method. Finally, Ahmad *et al*., (1992) have introduced neural network as a classifier in OCR systems.

(v) Post-processing

As classification stage sometimes produces not an unique solution but a set of possible solutions, the role of the post-processing stage emerge for selecting the right solution. Some researchers (Amin *et al*., 1984) used word lexicon for selection conformation. Amin and Mari (1989) used the Hidden Markov Model in conjunction with the Viterbi Algorithm to choose the word of higher probability. In some other two researches Amin *et al*., (1991) have, in one, involved verifying the compatibility of the subject and the object in accordance with action implied by the verb for word recog-

nition, where as in the other paper, Amin used rules of phonetics along with a small lexicon to rule out incorrect words.

2.4 OCR Systems Evaluation

Characterizing the performance of Optical Character Recognition (OCR) systems is crucial for monitoring technical progress, predicting OCR performance, providing scientific explanations for the system behavior and identifying open problems (Kunungo and Resnik, 1999). OCR evaluation can be broadly categorized into two types: i) Blackbox evaluation and ii) whitebox evaluation. In blackbox evaluation an entire OCR is tested as an indivisible unit and its end-to-end performance system is characterized. Whitebox, on the other hand, characterizes the performance of individual sub-modules(Kunungo *et al*., 1998), whereas (Kunungo and Resnik, 1999) propose a new visualization method, which they call the accuracy scatter plot, for providing a visual summary of performance results. In addition they stated that the statistical and visualization methods presented in their article are very general and can be used for comparing accuracies of any two recognition systems, not just OCR systems. Kanungo *et al*., (1998) have reported an evaluation algorithm for the most popular Arabic OCR system (Sakhr OCR and OmniPage for Arabic). Mao and Kanungo (2000) made a comparative evaluation to quantitatively compare the performance of page segmentation algorithms. Badr *et al*., (1995) prepared a useful comparison among a vast number of Arabic OCR systems introduced by different researchers.

3.0 CONCLUSION

In this paper we tried to cover as much area of off-line Arabic handwritten character recognition researches as we can throughout the last two decades. We noticed that the most important problem facing handwritten recognition, in general, and Arabic handwritten in particular, is segmentation. Cursive script requires the segmentation of characters within the word. So many researchers have tried to tackle this problem each with his/her own technique. However, the problem of segmentation is still considered as the highest order dilemma facing the off-line Arabic handwritten character recognition systems. To the best of our knowledge Motawa *et al*., (1997) have recently, developed a good segmentation technique based on Mathematical Morphology.

Thinning algorithms are scanned and found that the best algorithm which can be used for binary Image is MB2 thinning algorithm(Amin *et al.*, 1996) since it features the quality of AFP3 (Guo and Hall, 1992) and speed of JC thinning algorithm (Jang and Chin, 1990). As far as skew detection is concerned, and to the best of the authors knowledge fast and accurate algorithm is the one designed by Amin *et al*., (1996).

Decision tree with different masks or neural network can give the best result for reliable classification. Although the decision tree is faster than neural networks for classification, building a neural network is much easier.

REFERENCES

Abbas, S., M. H. Al Muifraje, and M. I. Harba. 1986. Optimizing the digital learning network for recognition of the handwritten numerals used by Arabs. *Proc. European Conf*., Paris, France. 505 – 513.

Abdulazim, H., and M. A. Hashish. 1989. Automatic Recognition of Handwritten Hindi numerals. *Proc. Comp. EURO'89 VLSI and Computer Conf*., Kuwait: 426 – 441.

Abohaiba, I., and P. Ahmed. 1993. Restoration of temporal information in off-line Arabic handwriting. *Pattern Recognition*. 26(7): 1009 – 1017.

Abuhaiba, I., S. Mahmoud, and R. Green. 1991. Skeletonization of Arabic characters using clustering based skeletonization algorithm. *(CBSA), Pattern Recognition*. 24(5): 453 – 464.

Amin, A. 1991. Recognition of Arabic handprinted mathematical formulae. *Arabian J. Eng. Science*. 16(4): 531 – 535.

Amin, A., A. Kaced, J. Haton, and R. Mohr. 1980. Handwritten Arabic character recognition by the I.R.A.C. system. *Proc. 5th Inter. Conf. Pattern Recognition*, Miami, FL.: 729 – 731.

Amin, A. 1982. Machine recognition of handwritten Arabic words by the IRAC II system. *Proc. 6th Inter. Joint Conf. on Pattern Recognition*, Munich, FRG.: 34 – 36.

Amin, A., and G. Masini. 1982. Machine recognition of cursive Arabic words. in: *Application of Digital Image Processing IV*, San Diego, CA. SPIE-359.: 286 – 292.

Amin, A., A. Kaced, J. Haton, and R. Mohr. 1980. Hand written Arabic character recognition by the I.R.A.C. system. *Proc. 5th Internat. Conf. Pattern Recognition, Miami, FL.* 729 – 731.

Amin, A., and G. Masini. 1986. Machine Recognition of Multifont Printed Arabic Texts. *Proc. 8th Inter. Joint Conf. on Patt. Recog., Paris, France*.: 392 – 395.

Amin, A., G. Masini, and J. P. Haton. 1984. Recognition of Handwritten Arabic Words and Sentences. *Proc. 7th Inter. Joint conf. on Pattern Recognition.*: 1055 – 1057.

Amin, A. 1991. Recognition of Arabic Handprinted Mathematical Formulae. *Arabian Journal Eng. Sc*. 16(4): 531 – 535.

Amin, A. 1997. Off-line Arabic character recognition- A-survey. *4th International Conference Document Analysis and Recognition (ICDAR '97)*, GERMANY I-II: 596 – 599.

Amin, A., and J. F. Mari. 1989. Machine recognition and correction of printed Arabic text. *IEEE Trans. On system Man & Cybernetics, SMC.* 19(5): 1300 – 1306.

Amin, A., S. Fischer, T. Parkinson, and R. Shiu. 1996. Fast Algorithm for Skew Detection. *IS&T/SPIE, symp. on Elec. Image, San Jose,* USA.: 65 – 76.

Ahmad, M., and R. Ward. 1998. A rule-based system for thinning symbols to their central lines. *IEEE Journal of PAMI*.

Ahmad, M., A. Jaaly, G. Dreyfus, and S. Knerr. 1992. Recognizing Arabic Characters Using Neural Networks for Electronic Document Processing. *Proc. Conf. on the Use of Arabic Language in info. tech., Riyadh, Saudi Arabia*.

Al-Badr, B. 1993. On The Recognition of Arabic Documents. *Tech. Report, The Dept. of Computer Science and Eng*. Univ. of Washington Seattle.

Al-Badr, B., S. A. Mahmoud. 1995. Survey and Bibliography of Arabic Optical Text Recognition. *Signal Processing*. 41: 49 – 77.

Al-Badr, B., and R. Haralick. 1992. Recognition without segmentation: mathematical morphology to recognize printed Arabic. *Proc. 13th Nat. Comp. conf*., Riyadh, Saudi Arabia.: 813 – 829.

Al-Badr, B., and R. Haralick. 1994. Symbol recognition without prior segmentation. *Proc. IS&SPIE symp. On Electronic imaging Sc. and tech. Conf. Doc. Reco., San Jose,* CA. 2181: 303 – 314.

Al-Badr, B., and R. Haralick. *1995*. Segmentation-free word recognition with application to Arabic. *Proc. Of the 3rd Inter. Conf. on Document Analysis and Recognition*. 1: 355 – 359.

Al-Badr, B., and R. M. Haralick. 1992. Recognition without segmentation: Using mathematical morphology to recognize printed Arabic. *Proc. 13th National Computer Conf*., Riyadh, Saudi Arabia.: 813 – 829.

Al-Emami, S., and M. Usher. 1990. On-line Recognition of Handwritten Arabic Characters. *IEEE Trans. Patt. Anal. Machine Intell.* 12(7): 704 – 710.

Al-Fakhri, F. 1997. *On-line Computer Recognition of handwritten Arabic text*. Master thesis, USM, Malaysia.

Alimi, A., and O. Ghorbo. 1995. The analysis in an on-line Recognition System of Arabic Handwritten characters.

3rd Inter. Conf. On Document Analysis and Proc., Canada: 890 – 893.

Alimi, A., and O. Ghorbo. 1995. The analysis of error in an on-line recognition systems of Arabic handwritten characters. *Proceedings of the Third International Conference on Document Analysis and Recognition.* 2: 890 – 893.

Altuwaijri, M., and M. A. Bayoumi. 1998. A thinning algorithm for Arabic characters using ART2 Neural Network. *IEEE Trans. On circuits and systems*. 45(2): 260 – 264.

Al-Yousefi, H., and S. S. Udpa. 1990. Recognition of handwritten Arabic characters via segmentation. *Arab Gulf Journal, Scientist Research*. 8(2): 49 – 59.

Badi, K. and, M. Shimura. 1979. Machine recognition of Arabic cursive scripts. *Pattern Recognition Practice*. 315 – 323.

Baird, H. 1993. Calibration of document image defect models. *Proc. 2nd Annual symp. on Document Analysis and Information Retrieval*, Las Vegas. : 1 – 16.

Bin Omar, K. *et al.* 2000. The removal of secondaries of Jawi characters. *TENCON 2000 conf*., KL, Malaysia. II: 149 – 152.

Blumenstein, M., and B. Verma. 1999. A new segmentation algorithm for handwritten word recognition *Inter. Joint Conf. On Neural Networks, IJCNN'99*. 4: 2893 – 2898.

Bozinovic, R., and Srihari. 1989. Off-line Cursive Script Word Recognition. *IEEE Trans. Pattern Anal. Machine Intell*. 11(1): 68 – 83.

Bozinovice, R., and S. N. Srihari. 1989. Off-line cursive word recognition *IEEE Trans. On PAMI*. 11: 68 – 83.

Brown, M., and S. Ganapathy. 1983. Preprocessing techniques for cursive script word recognition. *Pattern Recognition*. 16(5): 447 – 458.

Bushofa, B., and M. Spann. 1997. Segmentation of Arabic Characters Using their Contour Information. *Proc. DSP97, Santorini, Greece*.: 283 – 287.

Chang, F., Ya-Ching Lu, and T. Pavlidis. 1999. Feature analysis using line sweep thinning algorithm. *IEEE Trans. On PAMI*. 21(2): 145 – 158.

Ding, Y., F. Kimura, Y. Miyake, and M. Shridhar. 1999. Evaluation and Improvement of slant estimation for handwritten words. *IEEE, 5th Inter. Conf. on Document Analysis and Recognition*, India.

El-Dabi, S., R. Ramsis, and R. M. Guindi. 1990. Arabic Character Recognition System: A Statistical Approach for Recognizing Cursive Typewritten Text. *Pattern Recognition*. 23(5): 485 – 495.

El-Desouky, A., M. Salem Abd El-Gwad, and H. Arafat. 1992. A handwritten Arabic character recognition technique for machine reader. *Inter. Journal. of mini microcomputer*. 14(2): 57 – 61.

El-sheikh, T. 1990. Recognition of handwritten Arabic mathematical formulas. *Proc. UK IT 1990 Conf*. Southampton, UK: 344 – 351.

El-Sheikh, T., and S. G. El-Taweel. 1989. Real-time Arabic handwritten character recognition. *Proc. 3rd Inter. Conf. On Image Proc. and its Appl., Warwick, IEE*. London, UK: 212 – 216.

El-Sheikh, T., and R. Guindi. 1988. Computer Recognition of Arabic Cursive Scripts. *Pattern Recognition*. 21(4): 293 – 302.

El-Wakil, M., and A. Shoukry. 1989. On-line recognition of handwritten isolated Arabic characters. *Pattern Recognition* 22(2): 97 – 105.

El-Wakil, M., and A. Shoukry. 1989. On-line recognition of handwritten isolated Arabic characters. *Pattern Recognition*. 22(2): 897 – 105.

Fakir, M., and C. Sodeyama. 1993. Machine recognition of Arabic printed scripts by dynamic programming matching method. *IEICE Trans. Inform. Systems*. 76(2): 235 – 242.

Fischer, S., A. Amin, and D. Drivas. 1995. Segmentation of the Yellow Pages. *3rd Inter. Conf. on Document Analysis and Recognition*, Montreal, Canada IEEE.: 605 – 609.

Goneid, A., M. Shalaby, S. K. Hindawi, T. Nazmi, and K. Monir. 1992. A fast Algorithm for the recognition of Arabic handwritten numerals using topological features. *Proc. 1st Inter. Conf. On AI applications*, Cairo, Egypt.

Govindan, V., and A. P. Shivaprasad. 1990. Character Recognition-A Review. *Pattern Recognition*. 7(25): 671 – 683.

Guo, Z., and R. W. Hall. 1992. Fast fully Parallel Thinning Algorithms. *Comp. Vision, Graphics and Image, Proc*. 55(3): 317 – 328.

Habib, M., and Abdesselam Bouzerdoum. 1994. A system for Arabic Character Recognition. *IEEE*: 4(1): 354 – 361.

Haj-Hassan, F. 1985. Arabic character recognition in: *P.A. Mackay, ed., Computer and the Arabic language*. Hemisphere, New York.: 113 – 118.

Hyder, S., and A. Koujah. 1988. Character recognition of cursive scripts. *Proc. 1st Inter. Conf. On Industrial and Eng. Applications of AI and Expert system IEA, AIE-88, Tullahoma*, TN: 1146 – 1150.

Impedovo, S., L. Ottaaviano, and S. Occhinegro. 1991. Optical character recogntion- a survey. *Inter. Journal of pattern recognition and artificial Intelligence*. 5(1&2): 1 – 24.

Jambi, K. 1992. A system for recognizing handwritten Arabic words. *Proc. 13th National Computer Conf*., Riyadh, Saudi Arabia.: 472 – 482.

Jang, B., and R. T. Chin. 1990. Analysis of thinning Algorithms using mathematical Morphology. *IEEE Trans. on Patt. Anal. and Mach. Intel*. 12(6): 541 – 551.

Jang, B., and R. T. Chin. 1992. One-pass parallel thinning analysis, properties, and quantitative evaluation. *IEEE Trans. On Pattern Analysis and Machine Intell*. 18(3): 267 – 278.

Kim, Y., W. S. Choi, and S. W. Kim. 1992. High-speed thinning processor for character recognition system. *IEEE Trans. On Consumer Electronics*. 38(4): 762 – 766.

Kunango, K., R. Haralick, and I. Philips. 1993. Global and local document degradation models. *Proceeding of International Conference on Document Analysis and Recognition*, Nangano, Japan.

Kanungo, T., G. A. Marton, and O. Bulbul. 1999. OmniPage vs. Sakhr: Paired Model eEvaluation of Two Arabic OCR Products. *In Proceedings of SPIE Conference on Document Recognition (San Jose, CA)*. 3651: 109 – 120.

Kanungo, T., and P. Resnik. 1999. The Bible, Truth, and Multilingual OCR Evaluation. *In Proc. of SPIE Conf. on Doc. Recognition and Retrieval VI, D. Lopresti and Y. Zhou, eds., (San Jose, CA)*.

Kanungo, T., G. E. Marton, and O. Bulbul. 1998. Performance Evaluation of Two Arabic OCR Products. *Proc. of AIPR Workshop on Adv. in Comp. Assist. Recognition., SPIE (W. DC)*. 3584.

Lam, L., S. W. Lee, and C. Y. Suen. 1992. Thinning methodologies – A Comprehensive survey. *IEEE Trans. On Pattern Analysis and Machine Intelligence*. 14: 869 – 885.

Mahmoud, S. 1994. Arabic character recognition using Fourier descriptors and character contour encoding *Pattern Recognition*. 27(6): 815 – 824.

Mao, S., and T. Kanungo. 2000. Empirical Performance Evaluation of Page Segmentation Algorithms. *In Proceedings of SPIE Conference on Document Recognition. and Retrieval, (San Jose, CA)*.: 303 – 314.

Margner, V. 1992. SARAT- A system for the recognition of Arabic printed text. *Proc. 11th IAPR Inter. Conf. on Pattern Recognition, The Hague*, The Netherlands.: 561 – 564.

Mori, S., K. Y. Yamamoto, and N. Yasuda. 1984. Research on Machine Recognition of Hand-printed Characters. *IEEE Trans. Pattern And Machine Intell*. 6(4): 386 – 405.

Motawa, D., A. Amin, and R. Sabourin. 1997. segmentation of Arabic cursive script *Proc. Of the 4th Inter. Conf. on Document Analysis and Recognition*. 2: 625 – 628.

Nazif, A. 1975. A System for the Recognition of the Printed Arabic Characters. *Master thesis, Faculty of Engineering*, Cairo University.

Nouh, A., A. Sultan, and R. Tolba. 1980. An approach for Arabic character recognition. Journal of Engineering and Science. King Saud University. 6(2): 185 – 191.

Pavlidis, T. 1993. Recognition of printed text under realistic conditions. *Pattern Recognition, Letter*. 14: 317 – 326.

Romeo, K., *et al.* 1995. A new approach for Latin/Arabic character segmentation. *Proc. Of the 3rd Inter. Conf. on Document Analysis and Recognition*. 2: 874 – 877.

Saadallah, S., and S. Yacu. 1985. Design of an Arabic character reading machine. *Proc. Computer Processing and Transmission of the Arabic Language workshop*, Kuwait.

Simon, J. 1992. Off-line Cursive Word Recognition. *Proc. IEEE*. 80(7): 1150 – 1161.

Shoukry, A. 1991. A sequential algorithm for the segmentation of typewritten Arabic digitized text. *Arabian Journal of engineering and Science*. 16(4): 543 – 549.

Stallings, W. 1976. Approaches to Chinese Character Recognition. *Pattern Recognition*. 8: 87 – 98.

Tolba, M., and E. Shaddad. 1990. On the automatic reading of printed Arabic characters. *Proc. IEEE Inter. Conf. on systems, Man Cypernet*, Los Angeles, CA.: 496 – 498.

Tolba, M., S. Wahab, and A. Salem. 1987. A Recognition Algorithm for Printed Arabic Character. *Proc. IASTED*

inter. Symp. In applied informatics, Switzerland.: 128 – 131.

Ullmann, J. 1982. *Applications of Character Recognition*. CRC press, Boca Raton. Chapter 9.

Wakahara, T., H. Murase, and K. Odaka. 1992. On-line Handwriting Recognition. *Proc. IEEE*. 80(7): 1181 – 1194.

Wesley, C., A. Harvey, and A. Jennings. 1997. Skew detection in handwritten scripts. *TENCON'97 IEEE Region 10 annual conf. on speech and image technologies for computing and telecommunication*. 1: 319 – 322.

Zainodin, I., D. Khairuddin, and S. Horani. 1994. Sequential thinning of binary images. *Sains Malaysia*. 32(4): 35 – 57.

Zeki, A., and Mohamad S. Zakaria. 2000. New Primitives to Reduce the Effect of Noise for Handwritten Features Extraction. *TENCON 2000 conf., KL, Malaysia*. II: 403 – 408.