IJDAR
DOI 10.1007/s10032-012-0188-6

ORIGINAL PAPER

A survey on Arabic character segmentation

Yasser M. Alginahi

Received: 17 February 2011 / Revised: 9 April 2012 / Accepted: 23 April 2012

Abstract Arabic character segmentation is a necessary step in Arabic Optical Character Recognition (OCR). The cursive nature of Arabic script poses challenging problems in Arabic character recognition; however, incorrectly segmented characters will cause misclassifications of characters which in turn may lead to wrong results. Therefore, off-line Arabic character segmentation is a difficult research problem and little research has been achieved in this area in the past few decades. This is due to both the cursive nature of Arabic writing in both printed and handwritten forms and the scarcity of Arabic databases and dictionaries. Most of the character recognition methods used in the recognition of Arabic characters are adopted from available methods used on handwritten Latin and Chinese characters; however, other methods are developed only for Arabic character segmentation. This survey presents the description of the Arabic script characteristics with an overview on OCR systems and a comprehensive review mainly on off-line printed Arabic character segmentation techniques.

Keywords Arabic OCR · Arabic character segmentation · Off-line recognition · Cursive script · Arabic OCR software

1 Introduction

Optical Character Recognition, usually abbreviated to OCR, is the machine replication of human reading and has been the subject of intensive research since the development of digital computers. OCR involves computer systems designed to translate images of typewritten text (typically captured by a scanner) into machine-editable text to translate pictures of characters into a standard encoding scheme representing them in ASCII format. The motivation behind text recognition research stems from the necessity to convert data from conventional media into electronic media to be used in many applications such as machine document processing, bank cheques processing, automatic mail sorting and routing, signature verification, product identification, publishing applications and machine vision. OCR is a field of research in artificial intelligence, pattern recognition and machine vision. In the last century, the research concentrated mainly on Latin, Chinese and Japanese characters, and it is not until the last two decades that other languages were seriously researched. These include: Arabic, Farsi, Korean, Hindi, Urdu, Bengal and so forth. More research is found in the area of Handwritten Latin as well as the increase in the interest in researching handwritten Arabic text. In [1], the survey by Lorigo and Govindaraju, published in 2006, presented Arabic handwritten recognition techniques and concentrated on providing the recognition rates, description of the methods, and techniques used to recognize handwritten Arabic text. This research work did not provide detailed explanation of the segmentation techniques used in the recognition engines surveyed; it only provided a short subsection on the segmentation techniques with about 7 references used in this segmentation section. Therefore, [1] is an excellent survey which provides detail of handwritten Arabic OCR engines; however, it does not provide much detail on the characters segmentation techniques used. Thus, this survey differs in its purpose from [1] since it concentrates on techniques related to characters segmentation for printed Arabic text. In addition it

Y. M. Alginahi (✉)
Department of Computer Science, College of Computer Science
and Engineering, Taibah University, P.O. Box. 344,
Al-Madinah Al-Munawarrah, Saudi Arabia
e-mail: alginahi@gmail.com; yginahi@taibahu.edu.sa
URL: http://www.taibahu.edu.sa/staff/yginahi/

Published online: 24 May 2012

provides an up-to-date comprehensive detail pertaining to Arabic characters segmentation techniques, detailed explanation of the characteristics of Arabic characters, sections on databases and commercially available OCR software and finally recommendations for future directions in this area of research.

The research of Arabic character recognition started in 1975 by Nazif [2] as compared to Latin's which can be traced back to the mid 1940s [3–5]. Commercial Latin OCRs have been used since early 1950s and 1960s, the first commercial system was installed at the Readers Digest in 1955 and the United States Postal Services has been using OCR in postal machines to pre-sort mail since 1965 [6]. Although, almost 1/4 of the world population use Arabic language in speaking, writing and/or reading [7], little progress in both on-line and off-line research has been achieved towards automatic Arabic character recognition; therefore, much more attention is needed to achieve better results. This is mainly caused by the lack of research funding, scarcity of databases and dictionaries, and the cursive nature of Arabic writing.

Off-line recognition is performed after the writing or printing is completed. This means that sometimes the temporal information of text is lost which, in turn, adds to the complexity problem of character segmentation [3]. It is not surprising that the recognition rate of Arabic characters is lower than those of non-cursive Latin characters. Therefore, in this paper, the characteristics of Arabic characters are explained in terms of a comprehensive up-to-date review of off-line Arabic character segmentation methods related mainly to printed text as well as a comprehensive overview on Arabic OCR systems.

2 Characteristics of Arabic characters

Arabic is used as the principal language in all Arab countries of the Middle East and Northern Africa. Arabic is the official language of 26 countries with approximately 300 million native speakers and 250 million non-native speakers [8,9]. It is considered the world's 5th most influential language [10]. Arabic is estimated as the 7th top language of the Internet in millions of users in 2010 [11]. It is also central to other languages in the Muslim world such as Farsi (Persian), Urdu, Sindhi and Pashto. Some minority languages in China such as Uighur, Kazakh and Kirghiz are all written using a modified Arabic script.

The Arabic Alphabet consists of 28 characters and has many characteristics, [3,5,12,13]. The main features of the Arabic writing are:

1. It possesses a cursive text even when printed and the letters are connected from the baseline of the word.
2. It is written from right to left with the exception of numbers, which are laid out left to right.
3. Its written form has no equivalent to capital letters.
4. Its letters change their shape depending on their position in the word, Table 1. A single character can contain from one to four shapes for each character or ligature, depending on the implementation. The four possible shapes: isolated; in which case the character is not linked to either the preceding or the following character. Final; in which case the character is linked to the preceding character, but not to the following one. Initial; in which case the character is linked to the following character but not to the preceding one. Finally, middle; in which case the character is linked to both the preceding and following characters.
   From the Arabic alphabets, Table 1, six letters can only be connected from the right (initial) these are: ا ، د ، و ، ز ، ر ، ذ. The appearance of any of these letters in the middle of the word form one or more sub-words, meaning there are more than one connected components in a single word, these sub-words may consist of one or more characters. Therefore, the shape of a character depends on the context. Examples of words containing more than one connected components include: (طارق - 3 sub-words), ياسر-2 sub-words) and (أروى - 4 sub-words). Examples of single connected words include: محمد ، عيسى and فلسطين. Also, as shown in Table 1, the three letters (ع ، غ ، ه) have four different shapes (glyphs) according to their location while the other letters have at most two different shapes depending on their position in the word.
5. Arabic characters are justified by a stretching baseline and space is used as word separator. The baseline is horizontal and runs through connected portions of the text. The baseline has the maximum number of text pixels; Fig. 1 shows the baseline of an Arabic text line. The direction of the baseline in an Arabic text is a helpful piece of information since it is the easiest way of deducing the orientation of the text. On its own, knowledge of the exact location of the baseline enhances the ability of the recognition system to correct the skew of the page as well as to separate the text into lines.
6. A character can be represented with a vowel or diacritic mark written over or under it, Fig. 2. The Fatha, Dummah, Mada'ah, and Sukkun are written above the letter, but Kesra is written below the letter. The Hamza can be considered as a diacritic or a special character, and can be written in different positions. Diacritics are signs that represent short vowels or other sounds, such as syllable endings and Tanween (the addition of an *n* sound at the end of a word) which is usually represented with double Fatha (ً), Dummah (ٌ), or double Kesra (ٍ) as shown in Fig. 3. Double Fatha and Dummah are written above the

Table 1 Different shapes of Arabic characters

Name	Isolated	Initial	Medial	Final
alif	ا			ـا
baa	ب	بـ	ـبـ	ـب
taa	ت	تـ	ـتـ	ـت
thaa	ث	تـ	ـتـ	ـث
jiim	ج	جـ	ـجـ	ـج
haa	ح	حـ	ـحـ	ـح
khaa	خ	خـ	ـخـ	ـخ
daal	د			ـد
dhaal	ذ			ـذ
raa	ر			ـر
zaay	ز			ـز
siin	س	سـ	ـسـ	ـس
shiin	ش	شـ	ـشـ	ـش
saad	ص	صـ	ـصـ	ـص
daad	ض	ضـ	ـضـ	ـض
taa	ط	طـ	ـطـ	ـط
dhaa	ظ	ظـ	ـظـ	ـظ
Ayn	ع	عـ	ـعـ	ـع
ghayn	غ	غـ	ـغـ	ـغ
faa	ف	فـ	ـفـ	ـف
qaaf	ق	قـ	ـقـ	ـق
kaaf	ك	كـ	ـكـ	ـك
laam	ل	لـ	ـلـ	ـل
miim	م	مـ	ـمـ	ـم
nuun	ن	نـ	ـنـ	ـن
haa	ه	هـ	ـهـ	ـه
waaw	و			ـو
yaa	ي	يـ	ـيـ	ـي

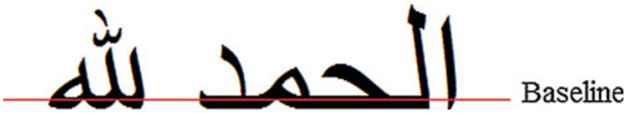


Fig. 1 Image of Arabic text showing the baseline

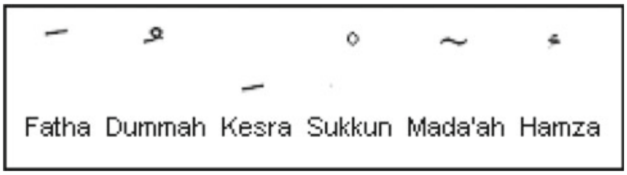


Fig. 2 Diacritic marks

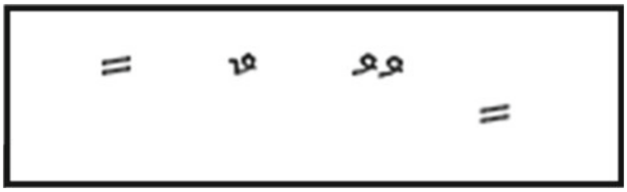


Fig. 3 Different Forms of Tanween

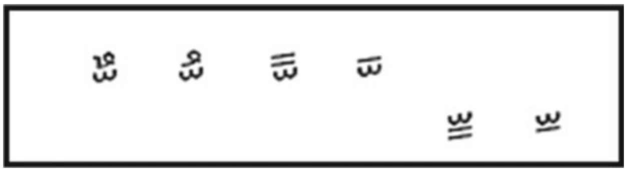


Fig. 4 Different Forms of Sha'ada

character and double Kesra is written below the character. Another diacritic mark is the Sha'ada which is written above the letter in the case of Fatha and Dummah and below the character in the case of Kesra, the Sha'ada looks like the number 3 rotated 90°clockwise, Fig. 4. A Sha'ada is used to show that the character is doubled and that the syllable is stressed; meaning a character with Sukkun came after another same character with a different diacritic mark; therefore, to avoid writing two characters that are the same, a Sha'ada is written with the diacritic of the short vowel implicitly representing another character which is not explicitly written.

The use of diacritics is very important in writing in order to differentiate between words that are written the same, but a single diacritic could completely change the meaning of the word. For example, the word كتب with a Dummah above the first letter kaaf ک is pronounced "kutub" which means books in Arabic and with the diacritic Fatha above the letter kaaf the word is pronounced "kataba" meaning wrote (the past tense of write) [12]. The use of diacritics, in regular writing, is not widely used and the advanced reader usually understands the meaning of the word from the context. However, for a non-Arabic reader it may be difficult to capture the meaning and could lead to misunderstanding of the text being read.

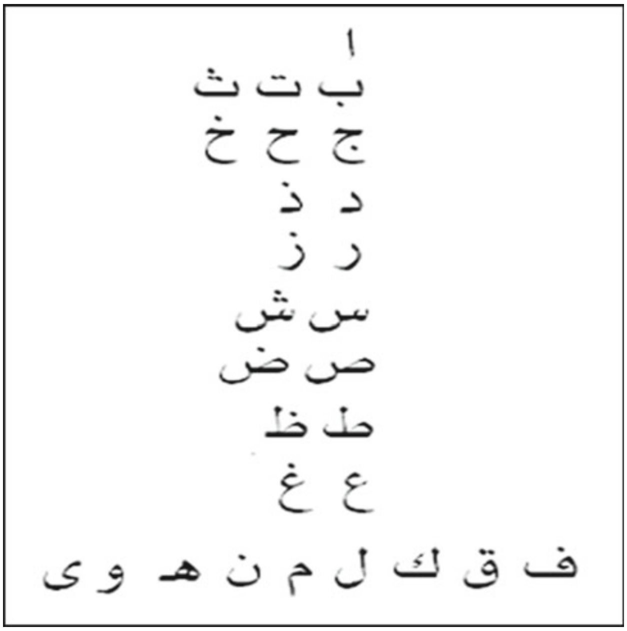


Fig. 5 Arabic alphabets

7. Several Arabic alphabet letters share the same shape, and are differentiated only in terms of the number and placement of dots on the letters. These dots may be referred to as desenders, if placed below the letters, or ascenders, if placed above the letter. Of the basic shapes shown in Fig. 5, we have two shapes of which each is used for three letters, six shapes of which each is used for two letters and the remaining ten are used for one letter each.
8. Arabic writing has also some ligatures, which are exceptions to the above joining letters' rules. The most commonly used is the laam-alif, لا, which is the combination of "laam" and "alif" others are lam-meem, لم, yaa-meem, يم and too many others as shown in Fig. 6. Ligatures denote pairs or, occasionally triples of characters such as the examples underlined in Fig. 6. These ligatures are not standard and are not presented by all fonts. Different fonts may contain different ligatures depending on the writing style presented by the font.
9. Additional characters: Ta-Marbuta, ـة ، ة, Hamza, ء, and Alif-Maqsurah, ى. The Ta-Marbuta can only be added to the end of a character, and is a special form of the letter ت. It can be written in two forms: isolated (ة) or final (ـة). Examples of words with Ta-Marbuta are: همزة - سهرة - قرية - مدينة. The Hamza, ء, can be considered an additional character, a desender or an ascender, and can be written in different forms; each form is given a different name. It can be written on alif, below alif, on waaw, by itself and in different forms as shown in Fig. 7. Hamza is also associated with some letters as can be seen below. In addition, it can be part of the letter kaaf, ك. The Alif-Maqsurah, ى, is pronounced Alif (ا) and it is always written at the end of a word, is not the last letter in the Arabic alphabet, ي, and is distinguished by the absence of the two dots.
10. Arabic characters have different lengths. As a result, wider characters are, sometimes, represented on some devices as two coded shapes.
11. The Arabic numerals, (0 **1** 2 3 4 5 6 7 8 9), are used in some Arab countries and the Western world, but the Hindi digits, (٩ ٨ ٧ ٦ ٥ ٤ ٣ ٢ ١ ٠), are widely used in most Arab countries.

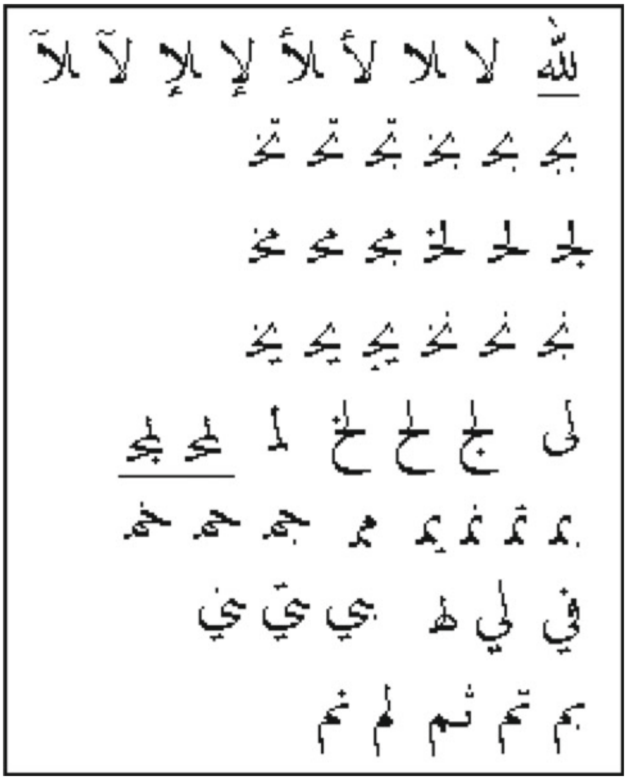


Fig. 6 Examples of Arabic ligatures

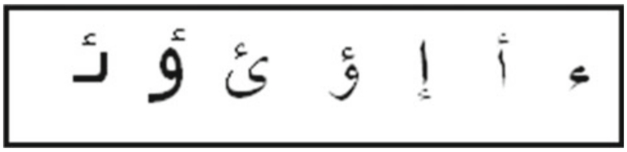


Fig. 7 Different forms of Hamza

The characteristics of Arabic text heavily influence the character segmentation rate which is one of the crucial and time-consuming steps in any Arabic segmentation based OCR system. Most of the recognition errors are attributed to this phase. The techniques for recognizing Arabic characters are not fundamentally different from those used for Latin OCR or any other language. However, the wider variability of the

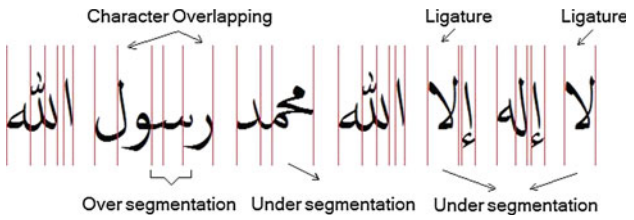


Fig. 8 Examples of over and under segmentation

Arabic character shapes and the linguistic rules associated with the Arabic writing need to be taken into consideration during the segmentation process. Also, more attention is needed to segment the characters in order to get a higher recognition rate. Typeset Arabic character recognition is harder than typewritten text of other languages due to the ligatures and character overlapping found in Arabic text resulting in over or under segmentation of characters as shown in Fig. 8. Recognition of Arabic handwritten text is even harder than typewritten text and the research in this area is progressing well as many techniques are being proposed and used. Many researchers find this area of research more challenging and are investigating techniques related to all different phases of Arabic OCR systems [14].

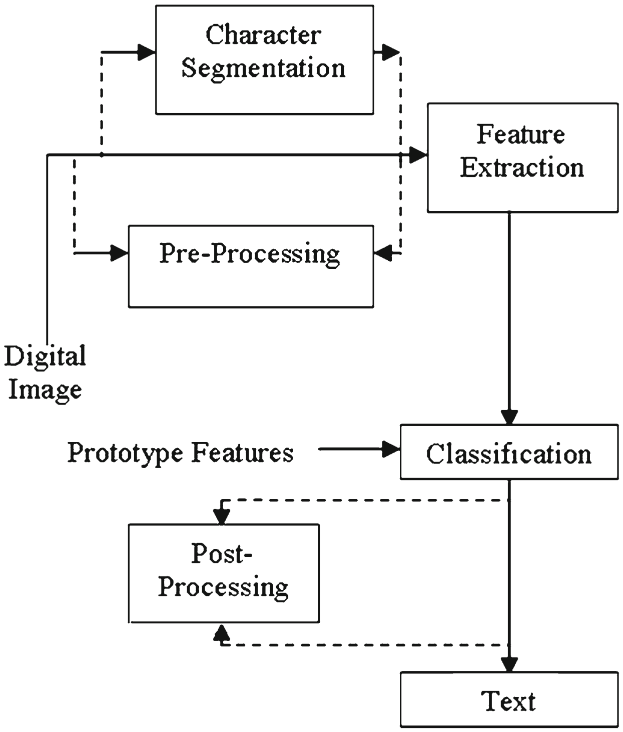


Fig. 9 A typical OCR system

3 A general OCR system

The main parts of a typical OCR system are shown in Fig. 9. The system goes through preprocessing, character segmentation, classification, recognition and post processing. A paper document is first transferred into a digital form i.e. a bitmap file by using a scanner. For most documents with complex or non uniform backgrounds, the image has to go through thresholding or binarization to remove the background and, for some document images page segmentation may be required to keep only the text. The characters in the binary image are then isolated or segmented to extract the character positions, which will be used in the next step of processing. Some preprocessing may be used depending on the quality of the character images. Such preprocessing could involve noise reduction, normalization of the character images, skew detection and correction or compression techniques which preserve the shape of the character such as thinning [15]. These steps work together to achieve one goal which is converting an image into an ASCII format that can be edited. Therefore, the output of each step propagates to the next stage in a pipeline fashion making the OCR system work as a whole and, if one stage fails the performance is significantly affected. In the classification stage, the extracted features are compared to the prototypes; if the features are matched or closely matched, the input character is classified into the appropriate class. In this stage, if a single classifier fails to yield high performance; several classifiers may be combined to give acceptable results [16]. The feature extraction, learning, classification and recognition steps of OCR are beyond the scope of this survey and for further details the reader may consult the following references [12,17] or other related references in scientific literature.

Some of the problems associated with character segmentation and lower recognition rates in Arabic OCR are:

- The poor knowledge and understanding of the Arabic language.
- The cursive nature of Arabic writing both printed and handwritten styles.
- Variation of character sizes in width and height comparing to Latin characters and other languages as well as different Arabic fonts as shown in Fig. 10.
- The presentation of different kinds of diacritics cause challenges in segmentation and character recognition.
- The use of ligatures where characters occupy a shared horizontal space creating vertically overlapping connected or disconnected blocks of characters.
- The character shapes change depending on their location in the word.
- The similarity in shape of some characters with only a change in the number of dots or diacritics.
- The omission of whitespace when writing in Arabic such as the case of the letter waaw و, which when written alone

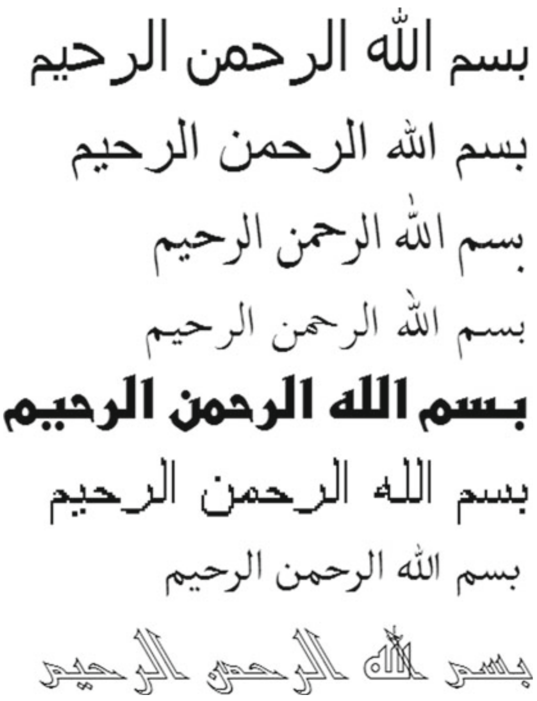


Fig. 10 Different Arabic fonts with uniform font size

means "and" therefore it should be treated as a word and written with white space before and after. However, many writers tend to omit the space either before or after and sometimes both. Similarly, writers tend to omit spaces between words when writing.

4 Benefits of OCR

OCR software has many benefits and can save individuals and businesses time and money. Some of the benefits of OCR are [18]:

- OCR software converts scanned text into a word processing file, providing the opportunity to edit documents and search for specific documents using a keyword or phrase.
- OCR classifies documents automatically saving time and effort.
- Data accuracy increases since data is extracted, validated and verified in seconds with minimal human intervention.
- The minimizing of manual data entry maximizes the benefits of businesses.
- Scanning paper documents saves on storage space by hauling the originals off to storage. A cabinet of files can be turned into a single Compact Disc.

Therefore, the more stable and better performance of an OCR system the higher the productivity of businesses in maximizing their return on investment.

5 Character segmentation techniques

Character segmentation is a necessary preprocessing step for character recognition in many OCR systems. It is an important step since incorrectly segmented characters produce misclassification or rejection of characters during the recognition process. The most difficult case in character segmentation is cursive script. The scripted nature of Arabic written language poses some difficult challenges for automatic character segmentation and recognition. Character segmentation of cursive text is the hardest, crucial, and time consuming step of any OCR system. It represents the main challenge in Arabic character segmentation systems, even more than the recognition phase and is considered the main source of recognition errors [3,19–21].

The vertical projection method or those based on the histogram of the image are considered one of the first attempted methods used in Arabic character segmentations [22–26] followed by other methods such as contour tracing [27,28], thinning [29–31], neural networks (NNs) [32,33], graph-based [34], morphological techniques [35–37] and Hidden Markov Models(HMMs) [38,39]. Most of these techniques are analytical where words are segmented into characters. Others use the holistic or segmentation-free approach with words being recognized without segmentation. The rest of the section provides an attempt to divide the character segmentation techniques into different categories.

5.1 Segmentation based on histogram and baseline

The use of the projection or histogram method simplifies the problem of character segmentation into a 1D system instead of a 2D system. In [40], Najoua and Noureddine presented a method based on modulated histogram, as well as the number of black segments in a line of pixels. The algorithm consists of the following steps: identifying text lines, segmenting text lines into Pieces of Arabic Words (PAW), separating each PAW into connected components, locating roughly limits of the different characters in the PAW, calculating the maximum number of black segments in a line of pixels, extracting primitives and finally using an error checker for detecting segmentation errors. This technique was applied on Arabic fonts such as Neskh, Bagdad, and Mehdi which have no overlapping characters or ligatures. The segmentation results were between 99 and 100 %, and when overlaying characters where tested as in Kaahra font the error checker detects errors which could not be corrected. This method is intended for specific fonts which do not deal with ligatures and could not be used with other fonts or handwritten Arabic character segmentation.

The technique presented by Parhami and Taraghi, [22] for the recognition of printed Farsi text which is also applicable to printed Arabic text, goes through two steps. The first is to isolate symbols within each sub-word and the second is recognition. The segmenting of parts is to determine the pen (script) thickness which is used to find candidate connection columns. As reported by the authors, the technique provided a 100 % recognition rate when used to segment Farsi newspaper headlines; however, the system is font dependent and smaller font size result in a lower recognition rate.

Amin and Masini in [25] proposed a system for segmentation and recognition that used horizontal and vertical projections and shape primitives. On 100 multi-font words, it achieved a character recognition rate of 85 % and word recognition rate of 95 %.

In [26] Ymin and Aoki presented a technique for segmenting Uygur characters. Uygur is a Turkish language used in the Xingjian Uygur autonomous region in China and Central Asia; it uses Arabic characters with few added letters. The technique presented consists of two segmentation steps; namely, topological segmentation and Quasi-topological segmentation. Topological segmentation traces the outer contour of the words from the edge of character strokes of the upper zone then it searches for any possibility breaks along the vertical projection. Next, the characters formed by loops are segmented into two or more parts which in itself is just a rough segmentation. Finally, the quasi-topological is used to section the characters based on combination of feature-extraction and character-width measurements. The authors claim a 93 % average success rate, with errors mainly induced by poor-quality printers and short horizontal links between consecutive characters.

In the recognition system proposed by Hamami and Berkani, the text was segmented into lines and connected components using horizontal and vertical projections, then the connected components, words and sub-words, were segmented into characters. The segmentation is based on the contextual properties of Arabic writing and gives a solution to the over-segmentation problem of some characters. The authors claim a recognition rate of 98 % using multi-font Arabic script [41].

Zheng et al. [42], proposed a new printed Arabic character segmentation algorithm, which is based on the vertical histogram and some other rules. Besides the structural characteristics between background regions and character components, characteristics of isolated Arabic characters, are also used to check whether the sub-word includes only one character. Then the vertical histogram and some other rules were used to find real segmentation points. Finally, sub-words were split at the segmentation points. The experimental results show that the algorithm achieved about 94 % correct segmentation.

In [21] and [43] the vertical projection of the middle zone instead of the entire word was used. Four text line zones were identified; namely, baseline, middle, upper and lower zones. The vertical projection of the middle zone was created and a fixed threshold was used for segmenting the word into characters. If the value of the vertical projection of the middle zone is lower than two thirds of the baseline thickness, the area is considered a connection area between two characters. Then any area follows the connection area with a larger value being considered as the start point of a new character as long as the profile is greater than one third of the baseline. This technique was used with the Naskh font and over-segmentation was seen with some characters such as, ش ، س. The problem was then solved in the recognition phase by combining more than one segment to produce the correct character.

In [44] El-Sheikh and Guindi developed a method that calculates the distance between the extreme point of intersection of the contour with a vertical line. The detected boundaries should be on the baseline and no secondaries should be detected. Then, for each sub-word the average vertical distance is calculated, if it is lower than a certain threshold then a silence region is detected. The same idea was also used with a modification in the width of the baseline in [45].

In [46,47] the character segmentation involves the building of vertical projection profile of the middle zone of the word. A fixed threshold is used for segmenting a word into characters. From the threshold level, the algorithm searches for the break along the vertical projection. Once the characters are segmented, a lower level of segmentation is applied to isolate the diacritic and dots that are associated with the characters; Fig. 11 shows the horizontal and vertical profiles of a word in Arabic.

The research in [48] proposes a new segmentation pre-recognition algorithm for Arabic handwritten text for which the algorithm goes through the following steps: the stroke width is measured, sub-words and dots are detected and noise removed. The dot groups and numbers are determined and assigned to a sub-word. Then a new image of each sub-word is produced. For each sub-word image loops are detected and the baseline optimized. Characters were Over-segmented

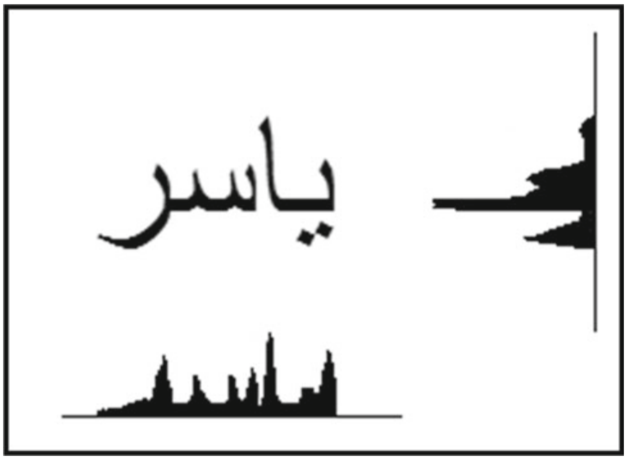


Fig. 11 Horizontal and vertical projection profiles

using gradients and “down-up” method. Finally, breakpoints in loops at edges are removed and using knowledge of letter shapes dot groups are assigned to letters. Segmentation is available for most, not all, letters, and the system does not currently handle markings like Sha’ada and Hamza. The letters not handled have a compound appearance, and the authors stated that future work would provide routines to recognize them even if segmentation breaks them.

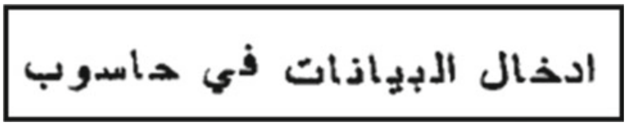
In [49] Zidouri et al. proposed a method for Arabic character segmentation, after applying preprocessing techniques, skeletonization of text, horizontal and vertical projections are employed to segment a page into separate lines and lines into sub-words respectively. Arabic characters are connected on the baseline. In order to dissect a sub-word into characters, the junction point between connected characters lies at baseline. After performing the above mentioned steps, an image with several guide bands is obtained. In order to select a correct guide band for sub-word dissection, several features are extracted from each guide band, the selection of these features are driven through several rules. If it satisfies the rules, then it is selected; otherwise, it is rejected.

In [50], El-Khaly and Sid-Ahmed showed that a thinned sub-word is segmented into characters by following the baseline of the sub-word. The sub-word is segmented when pixels start to move above or below the baseline.

The work of Abuhaiba, [51], laid, for the first time, the groundwork for the design of new fonts to produce discrete Arabic scripts instead of a cursive Arabic script. In his work, the history of Arabic calligraphy was presented since its start. Then, he called for breaking the cursive law of Arabic script which is possible by preserving all its other properties. The author selected a TrueType Arabic transparent font from which two new discrete Arabic script fonts were developed.

The character recognition step started with preprocessing to correct any skew followed by removal of solid lines and pepper noise then, the segmentation was performed using horizontal white cuts. The system produced excellent recognition rate of 99.5 % even though it was tested on 8-point fonts and poor-quality DeskJet printer. Therefore, it is possible to achieve a 100 % recognition rate with higher font-size and better quality printers/scanners. Figure 12 shows an example of a text line in discrete Arabic script fonts.

Amin and Mari in [52] presented a structural probabilistic approach to recognize Arabic printed text. Words were segmented into characters using vertical projections. Word recognition is based on Viterbi algorithm and can handle some identification errors. The system was tested on just a few words and no figures were reported about its performance. It seems that the technique has inherent ambiguity and deficiencies due to interconnectivity of Arabic text.



ادخال البيانات في حاسوب

Fig. 12 An example of a text line in discrete Arabic script font

It is very clear from the above techniques that the vertical projection method depends on the baseline. This method works very well for printed characters with no overlapping between them. The segmentation results are very poor with fonts forming ligatures and overlapping between characters [53,54]. The same problem is also observed with handwritten text due to the inter-line distance variability and the shift or skewness of the baseline [55,56]. Many other methods based on vertical projections are also available in the literature. In addition, many techniques use vertical projection as an initial or preprocessing stage to segment documents into lines then lines into words [25,47,57–64].

5.2 Segmentation based on contour tracing

Many techniques have been developed using contour tracing to solve the problem of character segmentation. In [27] the lower contour part is first examined to see where there are touching characters or an End-Ya (ي). Segmentation of touching characters is achieved by tracing the lower part of the contour from right to left. The lowest values in the vertical direction are recorded. The touching point is found to occur between two of these points and is the highest value in the vertical direction that satisfies the threshold condition mentioned above. Once a touching point is found, the characters are separated. This leads to dividing the contour into two or more parts depending on the number of touching characters. Consequently, the contour of the first part is extracted once more. The lower part of this contour is examined for occurrence of the End-Ya character. However, the segmentation and recognition procedures are interconnected so that each segmented character is first thinned and its features are then extracted. If it is not recognized, its segmentation is cancelled and the next segmentation point is taken and a recognition attempt is repeated. This self-correction procedure is necessary to increase the accuracy of both the segmentation and recognition procedures.

In an attempt to solve the problem of overlapping lower or upper strokes in handwritten Arabic text, Romeo-Pakker et al. [28], proposed a method using a contour-following algorithm which starts in the lowercase writing area and labels the detected contours. In the first stage, the junction segments connecting the characters to each other are detected by taking into account the writing line thickness. The second stage detects the upper contour of each word. The strokes are detected in order to find primary segmentation points. These points are analyzed with an automaton that considers the shape of the word for the determination of definitive segmentation points.

The segmentation in [65] is based on the contour of the main body of the words. In a first step, the start and end-point of the upper contour is determined. It is important to find the lower right and the lower left point of the contour because of the long vertical lines at the beginning of a word. Then, a segmentation of the upper contour into parts is made through a curvature of the same sign. Starting with a positive curvature for example, the change to a negative curvature will finish this segment and start with a new one. A low pass filter on the contour points is used to reduce the noise sensitivity of this procedure. In some cases where the horizontal line between the characters differs very much in length, but this length contains no information. Then, a horizontal line detector is used to mark these lines as unimportant for the recognition process. The classification in this system requires segmented characters and this segmentation is realized through the recognition step. A character can only be segmented, if the classification of the corresponding segments is successful.

The segmentation points in the Arabic character recognition system, [44], by El-Sheikh and Guindi, were based on minimal heights of word contours and the character classification was based on Fourier descriptors as features extracted from the segmented characters.

Peng et al. in [66] proposed a scheme for the segmentation of Arabic printed characters based on the analysis of the characteristics of character boundary. First, the baseline position of a given word is estimated, and the cure D(x) of the distance between the contour and the baseline is outlined. The candidate segmentation points are, then, found out by analyzing the curve D(x). Finally, structural rules are proposed to merge over-segmented characters. It is concluded that it is hard to distinguish similar characters found in different languages. The research did not provide specific recognition rate on Arabic text, but stated results in regard to other languages, Uighur/Chinese/English. The Uighur language characters are similar to Arabic and the recognition rate was found to be around 98 %.

Mehran et al. [67] investigates the Persian/Arabic scripts and found that the upper contour of the primary stroke of sub-words called PAWs (Piece of Arabic Word) has a high gradient at the junction points, and after most junction points, the vertical projection has a value larger than the mean. On the other hand, the pen tip is generally positioned near the baseline in the desired junction points. These features are used together to identify the junction points.

The method presented by Sari et al. [68] used the contour presentation to detect segmentation points by applying rules to local minima of the lower contour of each sub-word. Characters, vertically overlapped due to the writing style or slant, were processed in an advanced stage. The success rate was 86 % on a limited dataset of 100 words only. Therefore, the methods presented above show the need for combining contour features with other features in order to achieve better character segmentation which would ultimately provide higher recognition rate for OCR systems.

5.3 Segmentation based on thinning

The skeleton of an object provides its essential information. A number of approaches for thinning have been reported in the literature. The thinning algorithm proposed in [30] and [31] is based on clustering the character image. The algorithm employs the ART2 network which is a self-organized NN. The skeleton is generated by plotting the cluster centers and connecting adjacent clusters by straight lines. This method was used on isolated handwritten Arabic characters and recognition rate based on connected characters text was not provided meaning that this technique cannot be used to segment Arabic characters. Lam and Suen [69] referenced 138 publications in 1992. A simple search, today, for material on thinning identifies more than 200 papers which shows the importance of thinning in pattern recognition and the various approaches to yield skeletons of shapes. Tellache and Sid-Ahmed in [29] presented two parallel thinning algorithms used on isolated Arabic characters; the first is based on local operations to detect edge points, end-points and break-points. The second is a matching algorithm in which a set of eight templates and two images are used in the processing.

Cowell and Hussain in [70] used an iterative thinning algorithm with post processing to produce thinned forms of isolated Arabic characters. The authors also discussed the problems of thinning Arabic characters from poor quality image. In [71] thinning and stroke, segmentation were used in a preprocessing stage in a recognition system for handwritten Arabic text. Some researchers have used thinning in handwritten Arabic character recognition [72–74]. Thinning is an essential technique that could aid in solving the character segmentation problem; however, combining it with other techniques would be essential to guarantee better segmentation. Figure 13 shows an example of an Arabic word before and after using a thinning technique.

5.4 Segmentation based on NNs

Little research using NN in segmenting printed Arabic characters is reported in literature. The work in [32,33] was performed on handwritten Arabic writing and it is presented

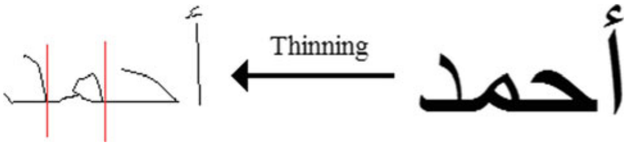


Fig. 13 An example of an Arabic word segmented after using a thinning technique

Table 2 Character segmentation methods using NNs [33]

Author	Method used	Dataset used	Accuracy (%)
Blumenstein and Verma [75]	Neuro-conventional method	Griffith University Latin handwriting database	81.21
Hamid and Haraty [33]	Feed-forward multilayer neural networks	Arabic handwritten—10,000 exemplars	69.72
Eastwood et al. [76]	ANN-based method	Cursive Latin handwriting form CEDAR database	75.90
Han and Sethi [78]	Heuristic algorithm	Latin handwritten works on 50 real mail envelopes	85.70
Lee et al. [77]	ANN-based method	Printed Latin alphanumeric characters	90
Sihari et al. [79]	ANN-based method	Handwritten zip codes	83

here to show that NN can be used in segmenting Arabic characters and that it should be suitable in printed Arabic characters. Hamid and Haraty [33] developed a technique that segments handwritten Arabic text. A recursive conventional algorithm was used for the initial segmentation of the text into connected blocks of characters. The algorithm, then, generates pre-segmentation points for these blocks. A NN is subsequently used to verify the accuracy of these segmentation points. Two major problems were encountered: though the segmentation phase proved to be successful in vertical segmentation of connected blocks of characters it could not segment characters that are horizontally overlapped. Second, segmentation of handwritten Arabic text depends largely on contextual information, and not only on topographic features extracted from these characters. The segmentation rate reported was 69.72 % and most of the errors were due to horizontal and vertical overlapping of characters. Ligatures could not be segmented and diacritics were miss-located especially as in handwriting, people tend to move things and not place them on top or below the associated characters. Therefore, NN is a powerful tool and using it in preprocessing stages of OCR to segment characters is a very promising and further research is encouraged in this area. Table 2 shows 6 methods of character segmentation techniques based on NNs [75–79], in addition to the method explained above [33] for Latin handwritten and printed text. The results confirm that low recognition rate is obtained with cursive handwritten text and better recognition rate is found with printed Latin text, however, the printed Arabic text shows very low recognition rate compared to the other NN techniques tested on Latin text this is due to the nature of the Arabic text and the difficulty in segmenting words into characters.

5.5 Segmentation methods based on graph theory

Several techniques have been developed using graph theory as an attempt to solve the character segmentation problem in Arabic scripts. Elgammal and Ismail character segmentation approach [34] is based on representing the text using line adjacency graph representation for the segmentation of cursive text. The segmentation is achieved by considering the relation between the text baseline and this graph. The approach can segment vertical overlapping of text. The graph representing the text is used to extract structural shape features such as strokes, loops and feature points that are used in the recognition. Using two different classifiers combined with some linguistic rules, the final recognition results are obtained using a regular grammar describing the formation of the characters from the basic scripts or segments. Figure 14 illustrates a text line and the strokes, feature points and loops for the Arabic word (الوطنية).

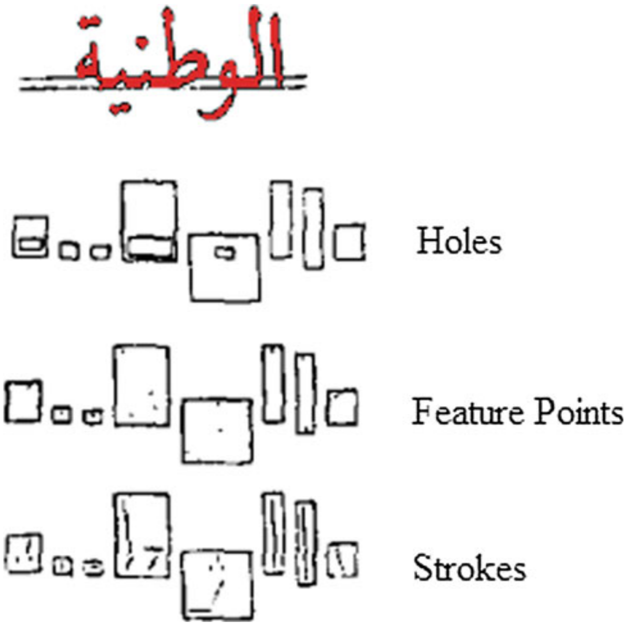


Fig. 14 Example of an Arabic word showing strokes, feature points and loops

The multi-queue model proposed by Xiu et al. uses a Positional Relationship Graph (PRG) to produce graphemes based on positional constraints with this PRG several parallel queues of graphemes are generated. The merging problem is inherently a process of walking in segmentation space, which is generated from the multi-queue structure with some defined metric. The optimization can be conducted to get the appropriate merging path. The experiments performed on handwritten Arabic characters using this model show that the principle of multi-queue merging scheme has positive impact on the system performance [80].

The ORAN off-line recognition of Arabic characters and numerals [81] is based on Modified Covering Run expression MCR. By using the correspondence between binary images and bipartite graphs, the MCR expression is obtained by constructing a minimum covering or maximum matching in the corresponding graph. The strokes of characters were described from the structural information obtained from the MCR expression according to some extracted features. Using zoning, the baseline was detected and the line of text was divided into four zones. Simple matching was, then, preformed against reference prototypes to recognize the text. Reference prototypes for the system are built according to a structural description of characters in some model document; therefore, this overcomes the problem of segmentation. Simple matching is performed for the candidate characters to reference prototypes. A recognition rate of more than 97 % is achieved.

The research in this area shows that graph theory needs to be extensively studied area to provide better segmentation techniques.

5.6 Segmentation based on morphological operators

Morphological operators have not been extensively investigated for the purpose of character segmentation and, very few methods have been developed. The technique in [35] is segmentation free technique based on describing symbols in terms of shape primitives. At recognition time, the primitives are detected on a word image using mathematical morphology operations. The system, then, matches the detected primitives with symbol models. This leads to a spatial arrangement of matched symbol models. The system conducts a search in the space of spatial arrangements of models and outputs the arrangement with the highest posterior probability as the recognition of the word. The advantage of using this whole word approach versus a segmentation approach is that the result of recognition is optimized with regard to the whole word. Results of preliminary experiments using a lexicon of 42,000 words show a recognition rate of 99.4 % for noise-free text and 73 % for scanned text.

Timsari and Fahimi [36] used morphological hit or-miss transformation to segment characters. Having the input words described in terms of some pre-defined patterns, the system knowledge base, holding descriptions for all characters, is searched for possible matches. Finding a match ends in the recognition of a character. This approach proves to be fast and reliable in practice.

5.7 Segmentation based on HMMs

HMMs have been used successfully in speech recognition applications [82–85]. In [38], HMM was used in off-line printed Arabic character segmentation, after the secondary strokes were removed a sliding window was used to scan each line from right to left. Features were extracted from the windows then applied to the HMM models. *M* different numbers of HMMs were constructed for each character or ligature. Using only two fonts, the simplified and traditional Arabic fonts of Microsoft word application a segmentation accuracy rate of 99 % was recorded. The system performance decreased as the number of models increased when using more font styles and including secondary strokes. In [86] HMM based speech recognition system performed well on OCR tasks with minimum changes and without pre-segmentation of data.

Touj et al. in [39] presented character segmentation relative to planar HMM-based model for off-line recognition of Arabic cursive handwritten Tunisian city names. El-Hajj et al. in [87] proposed a 1D HMM off-line handwriting recognition system employing an analytical approach. The system is supported by a set of robust language independent features extracted on binary images. Parameters such as lower and upper baselines are used to derive a subset of baseline dependent features. Thus, lexical variability due to lower and upper parts of words is better taken into account. In addition, the proposed system learns character models without character pre-segmentation.

The work of Rashwan [88] which started in 2000 and continued further by him and other researchers in [89] proposed an Arabic OCR system analogous to HMM-Based Automatic Speech Recognition system. In their work most recent work in 2008 [89], the authors, tried to overcome some of the following challenges found in Arabic OCR systems: connecting, dotting, multiple graphemes, ligatures and diacritics. The proposed system was tested on three different fonts (Simplified, Mudir and Traditional) with ligatures, the results show that the average word error rate was 3.3 % and the character error rate was 0.69 %.

From the above techniques using HMM, it is very clear that HMM has not been fully investigated in character segmentation of off-line Arabic printed text and, most of the work in Arabic handwritten text uses HMM as a classifier to recognize words [91–94].

5.8 Segmentation based on template matching

Template matching is a technique in digital image processing for finding small parts of an image which match a template image. Template matching has been used in character recognition applications. It is used as a technique to recognize characters or words with a reference to a stored database containing a set of images for the characters or words. This technique can be considered as a segmentation technique but, the characters or segments have to be chosen manually and stored to be used for comparison. In [95] a technique was proposed that searches for the occurrence of an angle formed by

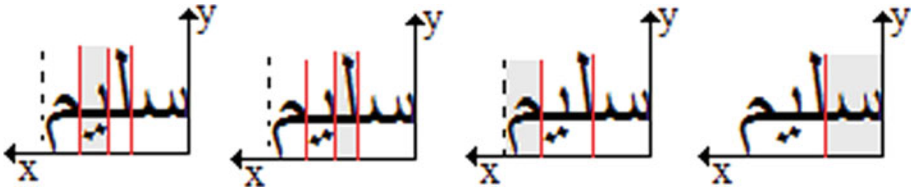


Fig. 15 Dynamic slide windows technique

the joining of two characters at the baseline. Using a 7 × 7 window the neighborhood of the characters is examined to decide on the segmentation. Although this method was able to achieve a good result, its success in finding the proper angle depends very heavily on the noise in the image.

Template matching is not a suitable technique for handwritten and or printed Arabic character recognition due to the different styles handwritings, in addition it also takes time to check all the templates especially with too many font styles and sizes, therefore, such a system is very slow and impractical.

5.9 Segmentation based on transforms

Techniques based on transforms such as Hough transform or Wavelet transforms have been developed to segment characters from Arabic scripts. Two techniques proposed by Touj et al. in [96] are based on feature extraction using the Hough Transform. The first uses the Standard Hough Transform (SHT) based approach by dynamic sliding window. This approach includes a pre-processing step feature extraction process based on SHT and classification step using HMM. The pre-processing step mainly eliminates the diacritics and centers the characters in fixed dimensioned image. Then, six directional quantified maps are obtained from the feature maps related to the orientation by using the Hough Transform. These characters are, then, passed through the HMM classifier. The dynamic sliding window technique is based on the recognition of the beginning and ending characters of the sub-word. Next, the remaining middle characters of the chain that composes the sub-word are identified. Therefore, for each sub-word target image the width of the dynamic window is computed. Computing the width of the dynamic window at the right side (beginning character), the left side (ending character) and sliding the window from the left to the right of the middle characters, for each window portion the Hough space is computed, then passed through the SHT/HMM character recognition engine, Fig. 15. This method is more sensitive to any mistakes during the serial-by-character recognition process and, it produced a recognition rate of 91 %. The second method in [96] used the Generalized Hough Transform (GHT) which is applied to the whole sub-word image instead of windowed portions of the image. The obtained GHT accumulators corresponding to the different characters models extract the characters composing of sub-word and estimate their relative positions. This method was more efficient than the former since it solves the cases of touching between sub-words in the text and the process is independent of the character positions in the sub-word. The recognition rate of this method was 97 %.

In [97,98] Broumandnia et al. introduced a technique based on wavelet transform, where the extracted wavelet coefficients are exploited in detecting underlying horizontal edges. The pre-processing is performed to remove artefacts on the bases of the projection of horizontal edges, the baseline is detected. The projection of horizontal edges and their location on baseline provide the segmentation points then, the classification method which is based on four NN classifiers, is used to distinguish the true segmentation points.

From the above methods, other transforms need to be investigated with their application to character segmentation.

5.10 Segmentation based on strokes, segments and tokens

Character segmentation methods are developed on the basis of projection profiles, baseline, contour tracing, NNs, graph theory, morphology, template matching, transforms and/or techniques combining more than one method may be applied to strokes, tokens and segments of characters. Then, these small parts are combined to form characters. In [99] Cheung et al. developed an algorithm for Arabic word segmentation which, also, separates horizontally overlapping Arabic words/sub-words based on combining character fragments. There is, also, a feedback loop to control the combination of character fragments for recognition. This recognition-based segmentation technique is developed by fragmenting Arabic words using their structural properties, connectivity points and Convex Dominant Points. Recognition is, then, preformed by combining fragments, using this technique the segmentation step is bypassed so there is no need to determine the actual character segmentation points. As there is no exact character segmentation points, some problems may occur such as that characters which look similar may be misrecognized. Another problem is that characters may deform or stack on other characters. The stack problem makes the character look different and leads to misclassification which can be solved by adding such sets of characters to the database. In addition, a horizontal fragmentation algorithm is required to segment the characters. The system was implemented and the results show 90 % recognition accuracy with a 20 characters per second recognition rate.

The work of Almuallim and Yamaguchi, [100], first, segment words into strokes, then geometrical and topological features are extracted. Following this, the elative positions of the classified strokes are examined and the strokes are combined in several steps into a string of characters which represents the recognized word. A maximum recognition rate of 91 % was achieved. The system failure, in most of the cases, was due to wrong segmentation of words into characters.

In [101], Ramsis et al. developed a technique for segmenting Arabic typewritten characters after recognition. As the characters are not separated yet, they assume that the rightmost columns of a word, the number of which equals the width of the smallest character, constitute a character. If a character is not found, another column is appended to the underlying portion of the word and moments are calculated and checked against the feature space of the font. This process is repeated until a character is recognized or the end of the word is reached. The technique allowed the system to handle overlapping and to isolate the connecting baseline between connected characters. However, it seems to be sensitive to font type and input pattern variations. The authors did not report figures on the system performance.

Zahour et al. [102] presented a method for automatic recognition of off-line Arabic cursive handwritten words based on a syntactic description of words. The features of a word are extracted and ordered to form a tree description of the script with two primitive classes: branches and loops. In this description, the loops are characterized by their classes and the branches by their marked curvature, their relationship, and whether they are in clockwise or counterclockwise direction. Some geometrical attributes are applied to the primitives that are combined to form larger basic forms. A character is, then, described by a sequence of the basic forms. The reported recognition rate of the system is 86 %.

In [103] Abuhaiba presented a text recognition system capable of recognizing off-line handwritten Arabic cursive text. A straight-line approximation of an off-line stroke is converted to a 1D representation from which tokens are extracted. The tokens of a stroke are re-combined to meaningful strings of tokens. The process of extracting the best set of basic shapes that represent the best set of token strings that constitute an unknown stroke was described. A method was developed to extract lines from pages of handwritten text, arrange main strokes of extracted lines in the same order as they were written, and present secondary strokes to main strokes. Presented secondary strokes are combined with basic shapes to obtain the final characters by formulating and solving assignment problems for this purpose. The system provided an overall sub-word and character recognition rates of 55.4 and 51.1 %, respectively.

The above methods show that more research is needed to investigate techniques applied on character tokens, segments and strokes. This can be done by applying the segmentation-recognition approach if the characters forming more than one segment are known to the system.

5.11 Holistic approach in Arabic character segmentation

The holistic approach, implicit segmentation (attempts to perform the task of segmentation and recognition simultaneously), or segmentation-free segmentation is motivated by the discoveries in psychological studies of the human reading process. In segmentation-free techniques, words are recognized as a whole without dissecting them into characters as is the case in speech processing applications [83,84,86].

In [104], the word shape is analyzed with a unique vector of features which, then, are matched against a database of prototyped feature vectors. In [105], Amin fed the attributes obtained from words into an inductive learning system. In [106,107], words are represented by a set of Fourier coefficients. The holistic word recognition techniques discussed for Arabic word recognition go through the steps of a typical OCR system from extraction of features, classification, where feature vectors are being compared against prototype feature vectors, and finally recognition of words. The segmentation-free approach in [108,109] is applied to four different Arabic typefaces, where ligatures and overlaps pose challenges to segmentation-based methods. The 2D Fourier transform was applied to the normalized polar image of the word representing each word with a template set Fourier coefficients. The recognition is based on a normalized Euclidean distance from the templates of features.

In [110] Tse and Bigun proposed a segmentation free OCR technique for Serto, the cursive alphabet of Syriac-Aramaic, with which Arabic writing shares some features. The system uses linear symmetry with a threshold of correlation for each character and ordered sequence of characters to be searched for has given results of 90 % correctly identified characters for the overall system ranging from 100 % for some characters to 59 % for others. The combination of using both ordered sequence of character and the appropriate threshold of correlation, together, yielded the best results. Yet the authors believe that further work is needed to make the system more flexible in terms of fonts and to speed up the execution of the system.

A major drawback of holistic approaches is that their use is usually restricted to a predefined lexicon. In [111] Dehghan et al. the authors developed a word recognition system for a specific application namely, the recognition of handwritten Iranian city names from postal addresses. In [39] a method based on HMM was developed for the recognition of handwritten Tunisian city names. This is because complete words are treated as symbols for recognition instead of letters. The training stage is mandatory to expand or modify the lexicon of words. This property makes these kinds of approaches limited to only static dictionary applications like bank check recognition or for on-line recognition on a personal computer with the recognition algorithm trained for a specific user and on a particular vocabulary [112,113].

On the other hand, the main advantage of the holistic techniques is that they bypass serious character separation problems. In principle, no specific segmentation algorithm for the specific script is needed and, recognition errors are mainly due to failures during the classification stage. For these reasons, more and more cursive script OCR systems use this technique for improving the recognition accuracy [35,59,114]. This approach is also known as segmentation-free recognition due to the absence of the character separation stage.

In conclusion, from the character segmentation methods surveyed in this paper, some are summarized in Table 3, it is very clear that the structural and statistical methods provide better results than others; thus, techniques based on the structural and statistical features of the characters seem to be promising for better performance. In addition, the survey in [1] also presents the features used in handwritten Arabic recognition and almost all of the features presented are structural and/or statistical features and this confirms the importance of further investigation of such techniques in the areas of printed and handwritten Arabic recognition. Finally, the unsatisfactory performance of OCR products, unavailability of comprehensive realistic databases and very low research support for this track of research are reasons discouraging researchers in the Middle East to extensively work in this area and this could be the reason that research groups outside of the Middle East are progressing well in this area.

6 Recent work on Arabic character segmentation

Arabic character segmentation continues to be a very important area of research. In the proposed work in [115] primary and secondary strokes of the sub-words are separated and, then, segmentation points are identified in the primary strokes. The vertical projection graph for each line is computed, which is, then, processed to generate a string indicating relative variations in pixels. The string is scanned further to produce characters from the sub-words. The Sindhi text, a super set of Arabic, was used for segmentation into characters. This method has been tested on images of text in many fonts and sizes, and the results show its success on their segmentation. It is observed that the characters like siin (س) and baa (ب) are segmented into more than one character, so it is required that such characters be given an extra attention at the later stages of the complete recognition system. No recognition rate was provided and, still, several characters are segmented into more than one part. Therefore, this method suffers from over segmentation of some characters.

Other work improved on previous techniques such as that of Zidouri in [116] he continued the project he started in [49] and proposed a general character segmentation technique which, he claims, is independent of font size and font type. The claim doesn't seem to be valid since only 4 fonts have been used in testing for this technique. The technique still produces over and/or under segmentation of characters. According to the author, these problems are solved by taking those characters that form a ligature (combination of two or more characters) as one class and the other problems are solved during the recognition step. In my opinion, the problem of over/under segmentation will escalate if more fonts are added to this system; this is the main problem contributing to low recognition rates in OCR.

Recently, the research on Arabic document analysis has been of interest to many researchers. In the last few years, there, also, has been recent work in different areas such as text segmentation from mixed documents [117], omnifont Arabic text recognition using HMM [118], online Arabic handwriting modeling [119], font recognition for Arabic scripts [120,121] and Farsi scripts [122], text-line segmentation of Arabic ancient handwritten document images [123], character segmentation for Arabic handwritten text recognition [124,125], online recognition of Arabic [126,127], Persian [98,126] and Urdu [128] text, text font estimation [129], … etc.

7 Arabic databases

According to [130], there are many reasons why there is a need to develop a reliable and robust Arabic OCR system: no current product is yet satisfactory enough to provide good performance, the Arabic language has a huge heritage to be digitized with the fear that wars could destroy many of these material and there is a large market of such a technology of over 300 million people speak Arabic, in addition other numerous interested parties such as intelligent agencies, businesses, tourism agencies, social research … etc.

In [131] AbdelRaouf et al. stated that "Excellent OCR now exists for Latin based languages, but there are few systems that read Arabic, which limits the penetration of Electronic Document Management Technology into Arabic speaking countries." In their work, the authors emphasized the necessity of creating a database of Arabic words and provided a comprehensive study and analysis of Arabic words by constructing a database for printed character recognition.

Table 3 Summary of some Arabic character segmentation techniques

Author(s)/reference	Character recognition method	Scope of application (printed/handwritten)	Database size	Accuracy
Zidouri [116]	A structural method based on the following: skeletonization, horizontal pixel count, vertical projection, decide on guide bands based on previous steps	Printed	200 images	90 %
Broumandnia [97]	Wavelet transform	Printed	1000 words different sizes and fonts	97.83
Timsari and Fahimi [36]	Morphology (Hit or miss operator)	Persian machine printed	2,000 words	98.30 %
Abuhaiba [51]	Horizontal white cuts (vertical projection)	Printed (discrete Arabic transparent 140 and discrete Arabic transparent 160 TrueType fonts)	contained 25,954 characters	99.5
Lorigo and Govindaraju [48]	Method depends on Baseline, strokes, loops	Handwritten	200 images (609 characters)	92.30 %
Fakir et al. [46]	Vertical projection	Handwritten	300 divided between training and testing	95 %
Elgammal and Ismail [34]	Graph-based segmentation	printed Arabic—Naskh fonts (more than 10 fonts—sizes 10–16	45 images	93.4
Najoua and Noureddine [40]	Method based on modulated histogram, and the number of black segments in a line of pixels	Printed (Arabic)—fonts (Neskh, Bagdad, and Mehdi) these fonts do not contain overlaying characters	500–1,000 words	Between 90 and 100 %
Parhami and Taraghi [22]	The segmenting of parts is to determine the pen (script) thickness which is used to find candidate connection columns	Printed (Farsi)	–	100 %—on Newspapers headings
Amin and Masini [25]	Horizontal and vertical projections and shape primitives	Multi-font Arabic text Printed	100 multi-font words	95 %
Ymin and Aoki [26]	Topological segmentation (traces the outer contour of the words) and quasi-topological (sections the characters based on combination of feature-extraction and character-width measurements)	Printed Uygur script	–	93.00 %
Hamami and Berkani [41]	Contextual properties of Arabic writing—structural method	Multi-font Arabic script	–	98.56 %
Zheng et al. [42]	Vertical histogram and some other rules.	Arabic (Printed)	–	94 %
Zidouri [81]	Based on modified covering Run expression MCR	Arabic (printed)	A dozen model of documents	97 %

Table 3 continued

Author(s)/reference	Character recognition method	Scope of application (printed / handwritten)	Database size	Accuracy
Nawaz et al. [21]	Vertical and horizontal projections	Arabic printed (Naskh font)	Many document images each containing about 200 characters	76 %
Bushofa and Spann [27]	Contour information	Printed (2 fonts—4 sizes each.)	1,065 characters from each font are tested	97.01 %
Hamid and Haraty [33]	Feed-forward Multilayer neural networks	Handwritten	10,000 exemplars	69.72 %
Touj et al. [96]	1- SHT based approach by dynamic sliding window 2- GHT based identification and Localization technique	Printed Arabic texts in Arabic transparent font	6,400 characters	Method 1–91 % Method 2–97 %

According to [131,132], there is no central organization responsible for generating an Arabic corpus; therefore, there is no standard reference list of Arabic words. From surveying the literature, currently there are very few large-scale image databases of Arabic printed text available for the scientific research committee. Furthermore, most of the developments in OCR research have been benchmarked on private databases. To the best knowledge of the author the only image based databases related to the research in this survey are: ERIM and APTI databases.

The Environment Research Institute of Michigan (ERIM) created a printed database of 750 pages (images) collected from books and magazines resources', but it seems difficult to have access to this database [133]. In the work carried by a joint collaboration between DIVA Group at the university of Fribourg (Switzerland), the REGIM group at University of Sfax (Tunisia) and the Software Engineering Unit at the Business Information System Institute (Switzerland) a synthetically generated database was developed using lexicons of 113,284 words, 10 fonts, 10 fonts sizes and 4 fonts styles. This database contains 45,313,600 single word images totaling more than 250-million characters. This database is called APTI for Arabic Printed Text Images [132,134]. To the best of the author's knowledge, no database with isolated characters was reported in literature and, if there are any they would be limited and used for specific applications.

The most recent database is the one developed by the Arabic Language Technology Center (ALTEC), [135], which provides a database with three Arabic fonts for character sizes from 12 to 22 and resolutions 200, 300 and 600 dpi all images were scanned or captured using different scanners and digital cameras; however, this database was limited to printed books and theses only which does not reflect the wide variety of documents available in hardcopies. The ALTEC website does not provide the size of the database and the database is not free since it requires a fee of $600 for a limited academic use and $6000 for a full database for commercial use.

On the other hand, Arabic text corpus or lexical databases are available from different associations or institutes, some of the research work in this area can be found in [131,136–138].

8 Arabic OCR software

In the last few decades, many OCR software solutions, commercial and open source have been developed to convert scanned images into text format, but very few supports Arabic documents. Very few Arabic OCR systems are available; however, their performance lags behind those developed for Latin or Chinese and this is mainly due to the cursive property of Arabic script [51]. According to www.aramedia.com [139], the following OCR software: Readiris Pro 11 Middle-East, VERUS, OmniPage for Arabic and Sakhr Automatic Reader are the most stable OCR software which support Arabic, in addition to other languages which use the Arabic script such as Farsi.

To the best knowledge of the author, very few references are available which provide some evaluation of Arabic OCR software. In [140,141], Kanungo et al. compared the performance evaluation of Sakhr OCR and OmniPage for Arabic text. In their evaluation, the authors established that the Sakhr OCR product has absolute page accuracy rates of 90.33 % compared to OmniPage with 86.89 %. This evaluation was performed on the SAIC Arabic image dataset and was only performed on pages which produced output. On the other hand, an evaluation of two of the earliest developed Arabic OCR software, during the early to mid 1990s: TextPert 3.7 Arabic and Al-Qari Al-Ali. The programs were tested and it was determined that Al-Qari Al-Ali is more powerful and performs better than TextPert but no

Table 4 Performance results for 3 best Arabic OCR

Company	Software version	Performance		Price
		Clean images (%)	Noisy images (%)	
ReadIris	Readiris 11 Pro Middle East Corporate Ed.	84	74	$999
NovoDynamics	Venus standard	94.5	71	$1,299
Sakhr	Automatic reader 9.0	57.8	48.6	$1,400

quantitative data of performance was presented in the evaluation performed [142]. The authors in [142] stated that the Al-Qari Al-Ali is very powerful and promising software. The evaluation results presented were according to the standards and status of technology at the time the evaluation was performed. Also, the evaluation provided the company with recommendations for improvements. It is fair to mention that the above two evaluations were performed in the 1990s and that they do not reflect the current progress in technology of the first and second decades of the 21st century.

Numerous OCR companies claim that their products provide high performance; however, in reality such a performance is rarely achieved. Most of the companies do not disclose information on the techniques/algorithms used in their system leading to more systems being developed instead of building on current technology. Most of the time systems break down when the input document is highly degraded/badly-illuminated/old/printed on low-quality papers/scanned from old photocopied documents, the layout structure of the document has not been used during the training process, and fonts used are not supported by the system and availability of ligatures and diacritics. In my opinion, most of the claims on high recognition rates found in research are valid only when such systems are used in applications with specific document structures trained using databases designed for those applications. Therefore, due to the large number of OCR software available mainly for languages using Latin characters and the infinite combinations of document types, e.g., Arabic/English, English/French, …, etc., it is possible that no engine may perform better with particular document type than another. Thus, OCR engines developed for specific languages are found performing overall better than those including many languages.

The OCR software performance analysis provided by www.itp.net/arabic for the three prominent Arabic OCR software, Readiris 11 Pro, Venus standard and Automatic Reader 90 is the most recent OCR software performance analysis available in literature, [90]. The performance results in Table 4 show Venus Standard OCR software provides the best results with an accuracy rate of 94.5 %; however, this software is considered the slowest among the others, i.e., it takes longer to process an image in order to produce an output text file. All these software must specify the language before the OCR engine can start the recognition process and in some software two languages can be specified such as in Automatic Reader 9.0. These software are considered expensive and they are mainly used commercially. Also, the unsatisfactory performance of OCR products, made people not interested in owning such software. This confirms that commercially available Arabic OCR software are far from perfection and far from being acceptable to be used in applications of archiving since more time would take to edit and approve such documents recognized by these systems. Therefore, it is recommended that more research is needed to provide better techniques to deal with the character segmentation and classification techniques in order to develop better OCR engines.

9 Observations and discussion

From the character segmentation techniques studied in this survey the following observations are noted: Character segmentation is considered a preprocessing stage in Arabic OCR Systems. Some techniques which are based on baseline detection, vertical projection, thinning, contour tracing, structural measures, and HMMs were used in developing techniques for character segmentation and feature extraction, then preparing the text for the recognition process. The problem of Arabic character segmentation is mainly due to touching and overlapping of characters, the presence of diacritics and ligatures, and the different widths of Arabic characters. Most of the techniques propose rules that apply to most characters and fail to segment other characters causing over or under-segmentation. Isolated characters may be over-segmented and overlapping characters may suffer from under-segmentation. Some of these problems could be solved by considering more than one segment for each character or applying implicit-segmentation where segmentation and recognition are applied simultaneously and still this solution will not guarantee the proper segmentation of all the characters.

The segmentation techniques developed for certain fonts may not be appropriate for other fonts. Most of the developments in printed Arabic character recognition have been benchmarked on private databases and; therefore, the comparison of different systems or research work is rather difficult unless the same database is used then the comparison would be a valid one. Table 3 summarizes some of the techniques explained in this survey and shows that none of the techniques presented use the same database.

Some techniques presented were only used on handwritten text. These techniques should be further investigated in printed Arabic text. Most of the work in literature is based on specific fonts where no ligatures or diacritics were present. Furthermore, in most cases these diacritics are called secondaries and removed as part of preprocessing. Specific fonts with limited data only are used. Techniques are intended mainly for specific applications on specific styles which tend to fail when used on other font types.

Databases available for printed Arabic image characters are very few. On the other hand, Arabic databases for specific applications are developed from different research work such as the work of Alohali et al. in [143] who developed Arabic cheque databases from real-life Arabic cheques for research in the recognition of handwritten cheques. The databases developed were, then, used in word recognition for legal check amounts using HMM as a classifier [144].

From the many techniques available, only limited exemplars or datasets were presented. The dataset could be limited to one font or few fonts, the number of dataset could be limited to few hundred words or images. An appropriate size Arabic database is not available and the results tend to be unacceptable on limited datasets. Therefore, most of the techniques presented do not provide the character segmentation rate as they only stated the overall recognition rate of the system. In doing so, the efficiency of the segmentation method is not accurate since other factors could be involved to improve the recognition rate. However, when it comes to 99 % recognition rate or higher accuracy, the best method is to design a font that makes it easy to segment the characters as is the case in OCR-B font used in the Machine Readable Zone of passports [145]. The OCR-B font prints all characters with the same height and width making it easy to dissect each character as a separate image. This could be difficult given the cursive nature of the Arabic writing. For specific applications using segmentation free techniques, the results are somewhat acceptable both in printed and handwritten Arabic text recognition. Abuhaiba in [51] stated that, " … to produce an Arabic OCR system with performance comparable to that for OCR systems of other languages, we believe that breaking the cursive law of Arabic script is a great step towards achieving that goal, i.e., new fonts are designed to write Arabic in a discrete manner in a way similar to other languages such as Latin alphabet languages and Chinese. If such a step is achieved, then almost all methods and techniques used in other OCR systems can be used in Arabic OCR systems as well."

The segmentation of Arabic characters is very challenging and a combination of techniques should be considered with segmentation-recognition approaches in order to guarantee acceptable recognition rates. The problem, mainly, lies in the touching and overlapping of characters. Segmentation of ligatures proved to be difficult and, in most cases where the researchers took such fonts producing ligatures, ligatures were considered as one character which, in turn, increased the number of classes to choose from during the recognition stage.

This study concludes that, in order to guarantee better results, hybrid or multi-segmentation methods should be used to combine the success of several methods. Hence, based on this survey techniques using structural and statistical features in the segmentation stage of the OCR engine are suggested to be further investigated in this area of research to be applied to both printed and handwritten Arabic OCR systems. Segmentation-recognition approaches are, also, effective in solving some issues where over-segmentation may be present, thus combining several techniques could result in better performance.

The priority in this research area is to develop Arabic databases of both printed and handwritten text to be used for research. The first step would be to develop real databases that can be used in research. The concentration should not be limited to specific applications such as cheques. Also limiting the recognition to the text without the diacritics causes problems in understanding the context when it comes to non-Arabic speakers. Parallel recognition systems should be used to recognize both diacritics and text separately at the same time.

10 Conclusion

In conclusion, from the techniques presented in this survey, it is very clear that none provided perfect segmentation results for a wide range of fonts making it difficult to achieve recognition rates above 99 %. Therefore, the problem of character segmentation in both printed and handwritten Arabic text is, still, a very challenging one and, still, needs to be further investigated. A suitable solution could be the development of discrete fonts as proposed by Abuhaiba by designing fonts which introduce white space between characters in a word or sub-word. It is impossible to develop a technique which is independent of font type or size due to vast number font type with different features. In addition, this survey recommends the following topics for further future investigation: develop a comprehensive real Arabic database, perform analysis of segmentation techniques used on both printed and handwritten Arabic text, investigate the use of combination of character segmentation techniques and improve current segmentation techniques used in commercial OCR software. Finally, researchers are trying to give more emphasis to handwritten text even though the research in the area of character segmentation of printed Arabic text is still not completely resolved as it is proven from the performance analysis of the available Arabic OCR software, Table 4.

References

1. Lorigo, L.M., Govindaraju, V.: Offline Arabic handwriting recognition: a survey. IEEE Trans. Patt. Anal. Mach. Intell. **28**(5), 712–724 (2006). doi:10.1109/TPAMI.2006.102Key: citeulike:9240539
2. Nazif, A.: A System for the Recognition of the Printed Arabic Characters. M.Sc. Thesis, Faculty of Engineering, Cairo University (1975)
3. Zeki, A.M.: The segmentation problem in Arabic character recognition: The state of the art. First International Conference on Information and Communication Technologies, ICICT, pp. 11–26 (2005)
4. Alshebeili, A., Nabawi, A., Mahmoud, S.: Arabic character recognition using 1-D slices of the character spectrum. J. Signal Process. **56**(1), 59–75 (1997)
5. Amin, A.: Segmentation of printed Arabic text. International Conference on Advances in Pattern Recognition, pp. 115–126 (2001)
6. Wikipedia: http://en.wikipedia.org/wiki/Optical_Character_Recognition (2010). Accessed 27 June 2010
7. Wikipedia: http://en.wikipedia.org/wiki/Arabic_language (2010). Accessed 23 Nov 2010
8. John, W.: Major Languages of the World, The New York Times Almanac. p. 492 (2002)
9. Wikipedia: Arabic Language.http://en.wikipedia.org/wiki/Arabic_language#cite_note-Proch-0 (2010). Accessed 3 Aug 2010
10. Andaman: http://www.andaman.org/BOOK/reprints/weber/rep-weber.htm (2011). Accessed 2 Jan 2011
11. Internet World Statistics:http://www.internetworldstats.com/stats7.htm (2010). Accessed 10 Dec 2010
12. Khorsheed, M.S.: Off-Line Arabic Character Recognition—A Review. Pattern Analysis and Applications, pp. 31–45 (2005)
13. Looklex Encyclopedia.: http://looklex.com/e.o/arabic_l.htm (2011). Accessed 13 Mar 2011
14. Al-Badr, B., Haralick, R.M.: A segmentation-free approach to text recognition with application to Arabic text. Int. J. Document Anal. Recognit. **1**(3), 147–166 (1998)
15. Alginahi, Y.M.: Chapter 1: Preprocessing Techniques in Character Recognition. Character Recognition, Edited by Minoru Mori, ISBN: 978-953-307-105-3, Sciyo, Available from: http://sciyo.com/articles/show/title/preprocessing-techniques-in-character-recognition (2010)
16. Alginahi, Y.M., Siddiqi, A.: Multi-stage hybrid Arabic/Indian numeral OCR system. Int. J. Comput.Sci. Inform. Secur. ISSN 1947–5500. **8**(1), 9–18. http://sites.google.com/site/ijcsis (2010)
17. Nixon, N., Aguado, A.: Feature Extraction and Image Processing, 2nd edn. ISBN: 978-0-12-372538-7, Elsevier Ltd., London(2008)
18. AnyDoc Software:www.ocrforanydoc.com. Automate your document processing and data capture.www.anydocsoftware.com/software/products/ocr/pdf/brochure.pdf (2010). Accessed on 10 Nov 2010
19. Jambi, K.: Design and Implementation of a System for Recognizing Arabic Handwritten Words with Learning Ability. Ph.D. Thesis. Illinois Institute of Technology, Chicago (1991)
20. Ali, M.O.: A New Pattern Matching Approach to the Recognition of Printed Arabic. Workshop on content visualization and intermedia representations (CVIR'98), University of Montreal, Montreal (1998)
21. Nawaz, S.N., Sarfraz, M., Zidouri, A., Al-Khatib, W.G.: An approach to off-line Arabic character recognition using neural networks. In: Proceedings of the 10th IEEE International Conference on Electronics, Circuits and Systems (ICECS 2003), vol. 3, pp. 1328–1331 (2003)
22. Parhami, B., Taraghi, M.: Automatic recognition of printed Farsi texts. Patt. Recognit. **14**(1–6), 395–403 (1981)
23. Adnan, A., Masini, G.: Machine recognition of Arabic cursive words. SPIE 26th International Symposium on Instrument Display. Application of Digital Image Processing IV, vol. 359, pp. 286–292. San Diego (1982)
24. Mori, S., Nishida, H., Yamada, H.: Optical Character Recognition. Wiley, New Jersey (1999)
25. Amin, A., Masini, G.: Machine recognition of multi-font printed Arabic Texts. In: Proceedings of International Conference on Pattern Recognition, Paris, France, pp. 392–395 (1986)
26. Ymin, A., Aoki, Y.: On the segmentation of multi-font printed Uygur scripts. 13th International Conference on Pattern Recognition, vol. 3, pp. 215–219 (1996)
27. Bushofa, B.M.F., Spann, M. Segmentation of Arabic characters using their contour information. 13th International Conference on Digital Signal Processing, vol. 2, pp. 683–686
28. Romeo-Pakker, K., Miled, H., Lecourtier, Y. A new approach for Latin/Arabic character segmentation. 3rd International Conference on Document Analysis and Recognition, vol. 2, pp. 874–877
29. Tellache, M., Sid-Ahmed, M.A., Abaza, B.: Thinning algorithms for Arabic OCR. IEEE Pac. Rim Conf. Commun. Comput. Signal Process. **1**, 248–251 (1993)
30. Altuwaijri, M., Bayoumi, M.: A new thinning algorithm for Arabic characters using self-organizing neural network. IEEE Int. Symp. Circ. Syst. **3**, 1824–1827 (1995)
31. Altuwaijri, M.M., Bayoumi, M.A.: A thinning algorithm for Arabic characters using ART2 neural network. IEEE Trans. Circ. Syst. II: Analog Digit. Signal Process. **45**(2), 260–264 (1998)
32. Hamid, A.: A neural network approach for the segmentation of handwritten Arabic text. International Symposium on Innovation in Information and Communication Technology. Amman, Jordan (2001)
33. Hamid, A., Haraty, R.: A neuro-heuristic approach for segmenting handwritten Arabic text. ACS/IEEE International Conference on Computer Systems and Applications (AICCSA 2001), pp. 110–113. Lebanon (2001)
34. Elgammal, A.M., Ismail, M.A.: A graph-based segmentation and feature extraction framework for Arabic text recognition. 6th International Conference on Document Analysis and Recognition, pp. 622–626 (2001)
35. Al-Badr, B., Haralick. R.M. Segmentation-free word recognition with application to Arabic In: Proceedings of the Third International Conference on Document Analysis and Recognition, vol. 1, 355–359 (1995)
36. Timsari, B., Fahimi, H.: Morphological approach to character recognition in machine-printed Persian words. In: Proceeding of SPIE. Document Recognition III. San Jose (1996)
37. Motawa, D., Amin, A., Sabourin, R.: Segmentation of Arabic cursive script 4th International Conference on Document Analysis and Recognition, vol. 2, 625–628
38. Gouda, A.M., Rashwan, M.A.: Segmentation of connected Arabic characters using hidden Markov models. IEEE International

Conference on Computational Intelligence for Measurement Systems and Applications, CIMSA, pp. 115–119 (2004)

39. Touj, S.M., Ben Amara, N., Amiri, H.: Segmentation stage of a PHMM-based model for off-line recognition of Arabic handwritten city names. IEEE Int. Conf. Syst. Man Cybernet. Vol. **4**, 6–9 (2002)
40. Najoua, B.A., Noureddine, E. A robust approach for Arabic printed character segmentation Third International Conference on Document Analysis and Recognition, vol. 2, 865–868 (1995)
41. Latifa, H., Daoud, B.: Recognition system for printed multi-font and multi-size Arabic characters. Arab. J. Sci. Eng. **27**(1B), 57–72 (2002)
42. Zheng, L., Hassin, A.H., Tang, X.: A new algorithm for machine printed Arabic character segmentation. Patt. Recognit. Lett. **25**(15), 1723–1729 (2004)
43. Sarfraz, M., Nawaz, S.N., Al-Khuraidly, A.: Off-line Arabic text recognition system. International Conference on Geometric Modeling and Graphics, London, England, pp. 30–36 (2003)
44. El-Sheikh, T.S., Guindi, R.M.: Computer recognition of Arabic cursive scripts. Patt. Recognit. **21**, 293–302 (1988)
45. Hashemi, M.R., Fatemi, O., Safavi, R.: Persian cursive script recognition. 3rd International Conference on Document Analysis and Recognition, vol. 2, pp. 869–873. Montreal, Canada (1995)
46. Fakir, M., Hassani, M.M: On the recognition of Arabic characters using Hough transform techniques. Malays. J. Comput. Sci. **13**(2), 39–47 (2000)
47. Fakir, M., Hassani, M.M., Sodeyama, C.: Recognition of Arabic characters using Karhunen- Loeve transform and dynamic programming. IEEE International Conference on Systems Man and Cybernetics, vol. 6, pp 12–15. 868–873, (1999)
48. Lorigo, L., Govindaraju, V. Segmentation and pre-recognition of Arabic handwriting In: Proceedings of the 8th International Conference on Document Analysis and Recognition, vol. 2, pp. 605–609
49. Zidouri, A., Sarfraz, M., Shahab, S.A., Jafri, S.M.: Adaptive dissection based sub-word segmentation of printed Arabic text. In: Proceedings of the 9th International Conference on Information Visualization, pp. 239–243 (2005)
50. El-Khaly, F., Sid-Ahmed, M.A.: Machine recognition of optically captured machine printed Arabic text. In: Proceedings of Pattern Recognition, vol. 23, pp.1207–1214 (1990)
51. Abuhaiba, I.S.I.: A discrete Arabic script for better automatic document understanding. Arab. J. Sci. Eng. **28**(1B), 77–94 (2003)
52. Amin, A., Mari, J.: Machine recognition and correction of printed Arabic text. IEEE Trans. Syst. Man Cybern. SMC **19**(5), 1300–1306 (1989)
53. Al-Badr, B., Mahmoud, S.: Survey and bibliography of Arabic optical text recognition. Signal Process. **41**(1), 49–77 (1995)
54. El-Dabi, S.S., Ramsis, R., Kamel, A.: Arabic character recognition system: a statistical approach for recognizing cursive typewritten text. Patt. Recognit. **23**(5), 485–495 (1990)
55. Amin, A.: "Arabic Character Recognition (1997) In: Handbook of Character Recognition and Document Image Analysis—(Chapter 15). Edited by H. Bunke and P. S. P. Wang. World Scientific. Singapore. pp. 397–420
56. Zahour, A., Taconet, B., Mercy, P., Ramdane, S.: Arabic hand-written text-line extraction. 6th International Conference on Document Analysis and Recognition, pp. 281–285. Seattle, Washington (2001)
57. Al-Yousefi, H., Udpa, S.S.: Recognition of Arabic characters. IEEE Trans. Patt. Anal. Mach. Intell. **14**(8), 853–857 (1992)
58. El Gowely, K., Dessouki, I., Nazif, A.: Multi-phase recognition of multi font photoscript Arabic text. 10th International Conference on Pattern Recognition ICPR, vol. 1, pp. 700–702. Atlantic City, New Jersy (1990)
59. Tolba, M.F., Shaddad, E.: On the automatic reading of printed Arabic characters. IEEE International Conference on Systems Man and Cybernetics, pp. 496–498. Los Angeles (1990)
60. Cheung, A., Bennamoun, M., Bergmann, N.W.: Implementation of a statistical based Arabic character recognition system. IEEE Region 10 Annual Conference on Speech and Image Technologies for Computing and Telecommunications, vol. 2. pp. 531–534. Brisbane, Australia (1997)
61. Abuhaiba, I., Mahmoud, S., Green, R.: Recognition of handwritten cursive Arabic characters. IEEE Trans. Patt. Anal. Mach. Intell. **16**(6), 664–672 (1994)
62. Ben Amara, N., Ellouze, N. A Robust approach for Arabic printed character segmentation 3rd International Conference on Document Analysis and Recognition, vol. 2, 865–868. Montreal (1995)
63. Abelazim, H., Hashish, M.: Arabic reading machine. 10th Saudi National Computer Conference. Riyadh, Saudi Arabia, pp. 733–743 (1988)
64. Abelazim, H., Hashish, M.: Automatic reading of bilingual typewritten text. Proc. CompEuro' **89**(VLSI and Computer Peripherals. Hamburg. 2), 140–144 (1989)
65. Margner, V.: SARAT-a system for the recognition of Arabic printed text. 11th International Conference on Pattern Recognition Methodology and Systems, vol. 2, Conference B, pp. 561–564 (1992)
66. Liangrui, P., Changsong, L., Xiaoqing, D., Hua, W.: Multilingual document recognition research and its application in China. Second International Conference on Document Image Analysis for Libraries, pp. 126–132 (2006)
67. Mehran, R., Pirsiavash, H., Razzazi, F.: A front-end OCR for Omni-font Persian/Arabic cursive printed documents. In: Proceedings of the Digital Imaging Computing: Techniques and Applications, pp. 56–60 (2005)
68. Sari, T., Souici, S.M.: Off-line handwritten Arabic character segmentation algorithm: ACSA. In: Proceedings of International Workshop Frontiers in Handwriting Recognition, pp. 452–457 (2002)
69. Lam, L., Suen, Y.: Thinning methodologies—a comprehensive survey. IEEE Trans. Patt. Anal. Mach. Intell. **14**(9), 869–885 (1992)
70. Cowell, J., Hussain, F.: Thinning Arabic characters for feature extraction. In: Proceedings Fifth International Conference on Information Visualization, pp. 181–185 (2001)
71. Goraine, H., Usher, M., Al-Emami, S.: Off-line Arabic character recognition. Computer **25**(7), 71–74 (1992)
72. Narima, Z., Messaoud, R., Mouldi, B. Neuro-Markovian hybrid system for handwritten Arabic word recognition In: Proceedings of the 10th IEEE International Conference on Electronics, Circuits and Systems, vol. 2, 878–881
73. Hosseini, H.M.M., Bouzerdoum, A.: A combined method for Persian and Arabic handwritten digit recognition. Australian and New Zealand Conference on Intelligent Information Systems, pp 80–83 (1996)
74. Mozaffari, S., Faez, K., Ziaratban, M. Structural decomposition and statistical description of Farsi/Arabic handwritten numeric characters Eighth International Conference on Document Analysis and Recognition, vol. 1, 237–241 (2005)
75. Blumenstein, M., Verma, B.: A segmentation algorithm used in conjunction with artificial neural networks for the recognition of real-world postal addresses. International Conference on Computational Intelligence and Multimedia Applications (ICCIMA '97), pp. 155–160. Gold Coast, Australia (1997)
76. Eastwood, B., Jennings, A., Harvey, A.: A feature based neural network segmenter for handwritten words. In: Proceedings of the International Conference on Computational Intelligence and Multimedia Applications (ICCIMA '97), pp. 286–290. Gold Coast, Australia (1997)

77. Lee, S.-W., Lee, D.-J., Park, H.-S.: A new methodology for gray-scale character segmentation and recognition. IEEE Trans. Patt. Anal. Mach. Intell. 1045–1051 (1996)
78. Han, K., Sethi, I. K.: “Off-line Cursive Handwriting Segmentation”, pp. 894–897. ICDAR ‘95, Montreal, Canada (1995)
79. Srihari, S. N.: Recognition of handwritten and machine-printed text for postal address interpretation. Patt. Recognit. Lett. 291–302
80. Xiu, P., Peng, L., Ding, X. (2006) Multi-queue merging scheme and its applications in Arabic script segmentation. In: Proceedings of the Second International Conference on Document Image Analysis of Libraries, pp. 24–29
81. Zidouri, A.: ORAN: a basis for an Arabic OCR system. International Symposium on Intelligent Multimedia, Video and Speech Processing, pp. 703–706 (2004)
82. Rabiner, L.R., Levinson, S.E.: A speaker-independent, syntax-directed, connected word recognition system based on hidden Markov model and level building. IEEE Trans. Audio, Speech Signal Process. **33**(3) (1985)
83. Afify, M.A.: Large Vocabulary Continuous Arabic Speech Recognition. Ph.D. Thesis, Faculty of Engineering. Cairo University (1995)
84. Brugnara, F., Faiavigna, D., Omologo, M.: Automatic Segmentation and Labeling of Speech Based on Hidden Markov Model. Speech Communication 12. North Holland (1993)
85. Rabiner, L.R., Wilpon, J.G., Soong, F.K.: High performance connected digit recognition using hidden Markov model. IEEE Trans. Audio, Speech Signal Process. **37**(8), 1214–1225 (1989)
86. LaPre, C., Ying, Z., Raphael, C., Schwartz, R., Makhoul, J. Multi-font recognition of printed Arabic using the BBN BYBLOS speech recognition system. IEEE International Conference on Acoustics, Speech, and Signal Processing, vol. 4, 2136–2139 (1996)
87. El-Hajj, R., Likforman-Sulem, L., Mokbel, C. Arabic handwriting recognition using baseline dependant features and hidden Markov modeling. Eighth International Conference on Document Analysis and Recognition, vol. 2, pp. 893–897 (2005)
88. Rashwan, M.A.: A new OCR system similar to ASR system. The 10th International Conference on Computing and Information, ICCI 2000, Kuwait (2000)
89. Rashwan, M., Fakhr, M., Attia, M., El-Mahallawy, M., Arabic OCR System analogous to HMM-based ASR systems; implementation and evaluation. J. Eng. Appl. Sci. Cairo University, www.Journal.eng.CU.edu.eg, Apr. 2008
90. www.itp.net/Arabic, “OCR software, group test, ”العربية – اختبار جماعي برامج التعرف على المحارف„
91. Alma’adeed, S., Higgens, C., Elliman, D.: Recognition of off-line handwritten Arabic words using hidden Markov model approach. 16th International Conference on Pattern Recognition, Vol. 3 , pp 481–484 (2002)
92. Bourouba, H., Bedda, M.: Hybrid approach DTW/HMMC for the recognition of the isolated Arabic words. International Conference on Information and Communication Technologies: From Theory to Applications, pp. 481–482 (2004)
93. Pechwitz, M., Maergner, V.: HMM based approach for handwritten Arabic word recognition using the IFN/ENIT—database. Seventh International Conference on Document Analysis and Recognition, pp. 890–894 (2003)
94. Dehghan, M., Faez, K., Ahmadi, M., Shridhar, M.: Holistic handwritten word recognition using discrete HMM and self-organizing feature map. IEEE Int. Conf. Syst. Man Cybern. **4**, 2735–2739 (2000)
95. Bushofa, B., Spann, M.: Segmentation and recognition of Arabic characters by structural classification. Image Vis. Comput. (IVC) **15**(3), 167–179 (1997)
96. Touj, S., Ben Amara, N., Amiri, H.: Two approaches for Arabic script recognition-based segmentation using the Hough transform. Ninth International Conference on Document Analysis and Recognition, vol. 2, pp. 654–658. ICDAR 2007 (2007)
97. Broumandnia, A., Shanbehzadeh, J., Nourani, M.: Segmentation of printed Farsi/Arabic words. International Conference on Computer Systems and Applications, pp. 761–766. Amman, Jordon (2007)
98. Broumandnia, A., Shanbehzadeh, J.: Fast Zernike wavelet moments for Farsi character recognition. Image Vis. Comput. **25**, 717–726 (2007)
99. Cheung, A., Bennamouon, M., Bergmann, N.W.: An Arabic optical character recognition system using recognition-based segmentation. Patt. Recognit. **34**, 215–233 (2001)
100. Almuallim, H., Yamaguchi, S.: A method for recognition of Arabic cursive handwriting. IEEE Trans. Patt. Anal. Mach. Intell. PAMI-9(5), 715–722 (1987)
101. Ramsis, R., El-Dabi, S.S., Kamel, A.: Arabic character recognition system, IBM Kuwait Scientific Centre, report No. KSC027 (1988)
102. Zahour, A., Taconet, B., Faure, A.: Machine recognition of Arabic cursive writing. In: Impedovo, S., Simon, J.C. (eds.) From Pixels to Features III: Frontiers in Handwriting Recognition, pp. 289–296. Elsevier Science Publishers B.V., Amsterdam (1992)
103. Abuhaiba, I.S.I.: Recognition of Off-Line Handwritten Cursive Text,” Ph.D. thesis, Department of Electronic and Electrical Engineering, Loughborough University, Loughborough, UK (1996)
104. Erlandson, E., Trenkle, J., Vogt, R.: Word-level recognition of multi-font Arabic text using a feature vector matching approach. In: Proceedings of the International Society for Optical Engineers, SPIE, vol. 2660, pp. 63–70 (1996)
105. Amin, A.: Recognition of printed Arabic text suing machine learning. In: Proceedings of the International Society for Optical Engineers, SPIE, vol. 3305, pp. 63–70 (1998)
106. Clocksin, W.F., Khorsheed, M.S.: Word recognition in Arabic handwriting. In: Proceedings of International Conference on Artificial Intelligence Applications. ICAIA. Egypt (2000)
107. Khorsheed, M.S., Clocksin, W.F.: Spectral features for Arabic word recognition. IEEE International Conference on Acoustics. Speech and Signal Processing, ICASP, Turkey (2000)
108. Khorsheed, M.S., Clocksin, W.F.: Structural features of cursive Arabic script. In: Proceedings Of British Conference on Machine Vision, pp. 422–431 (1999)
109. Khorsheed, M.S., Clocksin, W.F.: Multi-font Arabic word recognition using spectral features. 15th International Conference on Pattern Recognition **4**, 543–546 (2000)
110. Tse, E., Bigun, J.: A Base-line character recognition for Syriac-Aramaic. IEEE International Conference on Systems, Man and Cybernetics, pp. 1048–1055 (2007)
111. Dehghan, M., Faez, K., Ahmadi, M., Shridhar, M.: Off-line unconstrained Farsi handwritten word recognition using fuzzy vector quantization and hidden Markov word models. In: Proceedings of the 15th International Conference on Pattern Recognition, vol. 2, pp. 351–354 (2000)
112. Nadeem Ahmad Khan: A shape Analysis Model with Application to Character and Word Recognition. Technische Universiteit Eindhoven. Proefschrift. ISBN 90-386-1750-X (2000)
113. Casey, R.G., Lecolinet, E.: A survey of methods and strategies in character segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence **18**(7), 690–706 (1996)
114. Chen, C.H., DeCurtins, J.L.: Word recognition in a segmentation-free approach to OCR. Second International Conference on Document Analysis and Recognition, pp. 573–576 (1993)
115. Shaikh, N.A., Shaikh, Z.A., Ali, G.: Segmentation of Arabic text into characters for recognition. IMTIC 2008, CCIS 20, pp. 11–18 (2008)

116. Zidouri, A.: On multiple typeface Arabic script recognition. Res. J. Appl. Sci. Eng. Technol. **2**(5), 428–435 (2010)
117. Shirali-Shahreza, S., Manzuri-Shalmani, M.T., Shirali-Shahreza, M.: A skew resistant method for Persian text recognition. In: Proceedings for the IEEE Symposium on Computational Intelligence in Image and Signal Processing, pp. 115–120 (2007)
118. Khorsheed, M.S.: Off-line recognition of Omnifont Arabic text using the HMM ToolKit (HTK). Patt. Recognit. Lett. **28**(12), 1563–1571 (2007)
119. Boubaker, H., El Baati, A., Kherallah, M., Alimi, A.M., Elabed, H.: Online Arabic handwriting modeling system based on the graphemes segmentation. 20th International Conference on Pattern Recognition (ICPR), pp. 2061–2064 (2010)
120. Moussa, S.B., Zahour, A., Benabdelhafid, A., Adel, M.A.: New features using fractal multi-dimensions for generalised Arabic font recognition. Patt. Recognit. Lett. **31**(5), 361–371 (2010)
121. Slimane, F., Kanoun, S., Alimi, A.M., Ingold, R., Hennebert, J.: Gaussian mixture models for Arabic font recognition. 20th International Conference on Pattern Recognition (ICPR), pp. 2174–2177 (2010)
122. Khosravi, H., Kabir, E.: Farsi font recognition based on Sobel-Roberts features. Patt. Recognit. Lett. **31**, 75–82 (2010)
123. Boussellaa, W., Zahour, A., Elabed, H., Benabdelhafid, A., Alimi, A.: Unsupervised block covering analysis for text-line segmentation of Arabic ancient handwritten document images. 20th International Conference on Pattern Recognition (ICPR), pp. 1929–1932 (2010)
124. Al Hamad, H., Abu Zitar, R.: Development of an efficient neural-based segmentation technique for Arabic handwriting recognition. Patt. Recognit. **43**, 2773–2798 (2010)
125. Saeed, K., Albakoor, M.: Region growing based segmentation algorithm for typewritten and handwritten text recognition. Appl. Soft Comput. **9**, 608–617 (2009)
126. Harouni, M., Mohamad, D., Rasouli, A.: Deductive method for recognition of on-line handwritten Persian/Arabic characters. The 2nd International Conference on Computer and Automation Engineering (ICCAE) **5**, 791–795 (2010)
127. Omer, M.A.H., Shilong, M.: Recognition online Arabic pattern.. 3rd International Conference on Advanced Computer Theory and Engineering (ICACTE) **6**, 18–22 (2010)
128. Khan, K.U., Haider, I.: Online recognition of multi-stroke handwritten Urdu characters. International Conference on Image Analysis and Signal Processing (IASP), pp. 284–290 (2010)
129. Shirali-Shahreza, M., Shirali-Shahreza, S.: Persian/Arabic text font estimation using dots. In: Proceedings of the IEEE International Symposium on Signal Processing and Information Technology, pp. 420–425 (2006)
130. Altec, “Character Recognition Systems Overview”, viewed on 13/2/2012. http://www.alteccenter.org/page.php?pg=filesrepository/getRepository.php&main_cat=1&sub_cat=24
131. Ashraf, A., Colin A.H., Mahmoud, K. (2008) A database for Arabic printed character recognition. International Conference on Image Analysis and Recognition, ICIAR 2008, pp. 567–578
132. Slimane, F., Ingold, R., Kanoun, S., Alimi, M.A., Hennebert, J.: Database and evaluation protocols for Arabic printed text recognition. Internal Research Report (DIUF). University of Fribourg. Switzerland. Obtained from: http://diuf.unifr.ch/diva/APTI/publications.html (2009)
133. Schlosser, S.: ERIM Arabic Database. Document Processing Research Program, Information and Materials Applications Laboratory. Environmental Research Institute of Michigan (1995)
134. Slimane, F., Ingold, R., Kanoun, S., Alimi, M.A., Hennebert, J.: A new Arabic printed text image database and evaluation protocols. In: Proceedings of 10th IEEE International Conference on Document Analysis and Recognition (ICDAR 2009), pp. 946–950. Barcelona (Spain) (2009)
135. Altex, http://www.ALTEC-Center.org, viewed on 13/2/2012
136. Beesley, K.R.: Arabic finite-state morphological analysis and generation. In: COLING-96 Proceedings, Copenhagen, vol. 1. pp. 89–94 (1996)
137. Ramzi Abbes, J.D., Hassoun, M.: The architecture of a standard Arabic lexical database. Some figures, ratios and categories from the DIINAR.1 Source program. In: Workshop of Computational Approaches to Arabic Script-based Languages, Geneva, Switzerland (2004)
138. David Graff, K.C., Kong, J., Maeda, K.: Arabic Gigaword, 2nd edn. Linguistic Data Consortium. University of Pennsylvania, Philadelphia (2006)
139. Bell, J., Zemanek, P.: Test of two Arabic OCR programs. Manuscr.Orient. Int. J. Orient. Manuscr. Res. **1**(3), 55–57. http://www.islamicmanuscripts.info/.../Bell-Zemanek-1995-MO-1-3-Test-O.PDF (1995)
140. Kanungo, T., Marton, G.E., Bulbul, O.: Performance evaluation of two Arabic OCR products. In: Proceedings of AIPR Workshop on Advances in Computer Assisted Recognition. SPIE vol. 3584, Washington, DC (1998)
141. Kanungo, T., Marton, G.A., Bulbul, O.: OmniPage versus Sakhr: paired model evaluation of two Arabic OCR products. In: Proceedings of SPIE Conference on Document Recognition, San Jose, CA, vol. 3651, pp. 109–120 (1999)
142. Aramedia, The Best Arabic OCR Technology, viewed on 10th of June, 2010, http://aramedia.com/ocr.htm
143. Al-Ohali, Y., Cheriet, M., Suen, Ching: Databases for recognition of handwritten Arabic cheques. Patt. Recognit. **36**(1), 111–121 (2003)
144. Al-Ohali, Y.: Handwritten word recognition: application to Arabic cheque processing. Department of Computer Science, Concordia University, Montreal (2002)
145. Fujisawa, H.: Forty years of research in character and document recognition—an industrial perspective. Patt. Recognit. Vol. 41, pp. 2435–2446 (2008)