International Multilingual Academic Journal **IMAJ**
ISSN 2330-6440 Vol 3, No 1, Feb 2016
http://asair.org/publish/index.php/imaj/index

Enhancing the segmentation of Arabic characters using baseline information

Hadeel M. Al-Ateeq [a,b,*] **and AbdulMalik S. Al-Salman** [a]
[a] Computer Science Department, King Saud University, Riyadh, Saudi Arabia
hmalateeq@pnu.edu.sa, Salman@ksu.edu.sa
[b] Computer Science Department, Princess Nourah Bint Abdulrahman University, Riyadh, Saudi Arabia
hmalateeq@pnu.edu.sa
*Corresponding author

Abstract. Optical Character Recognition (***OCR***) system was present to provide an automatic recognition of large printed documents for archiving or processing which improves the interaction between human and machine in many applications. The proposed system was developed for Arabic language. Arabic language is written cursively and consists of 29 characters each with different shapes. The Arabic OCR (AOCR) system is divided into several steps: image acquisition, preprocessing, segmentation, feature extraction, recognition and post processing. Segmentation step segments the text into lines, then into glyphs and finally into characters. The most important and sensitive step is the character segmentation step which its result may affect the following steps and at the end the recognition rate. This study has concentrated on character segmentation by enhancing an already published algorithm in the literature. As a result, the new algorithm decreases the processing time and avoids the segmentation of descenders such as the letter Ra (ر) and End Ya shapes (ـي، ى، ئ).

Keywords: optical character recognition, Arabic language, character segmentation, baseline**.**

1 INTRODUCTION

Since the early days of digital computers, simulating the human reading has become one of the main topics of intensive research, due to the need for converting data present on paper documents into electronic readable format. This type of conversion has led to the development of efficient, practical and commercial applications such as automatic mail routing, check & signature verifications, office automation and a huge of different applications in banking, business & data entry and machine vision. These are used to automatically recognize both printed and handwritten information available in documents such as checks, envelopes, forms, text images, etc. (Sarfraz et al. 2003; Najoua and Noureddine 1995; Zeki 2005; Alginahi 2013; Al-Shatnawi and Omar 2014; Osman et al.2009).

So, OCR is an old data entry technique that provides an automatic recognition of large printed documents for archiving or processing which improves the interaction between human and machine in many applications. Its popularity increases each year with the rise of fast microprocessors. These microprocessors provide the machine with improved recognition techniques that increased the effectiveness in both read rates and accuracies (Sarfraz et al. 2003; Alginahi 2013; Osman et al. 2009; Abdulaziz and Alsaif 2008; Amor et al. 2006; Cheung et al. 1997; Amin 1997).

OCR systems differ from each other from various views: (1) Way of getting the text, (2) Mode of writing, (3) Number of fonts can be recognized, and (4) Text's connectivity (Zeki 2005). Online OCR describes the recognition of symbols while they are drawn on a digitizing tablet that operates by using a special pen for writing (Zeki 2005; Al-Shatnawi and Omar 2014; Hachour 2006; Amin 1997). In contrast, offline OCR describes the recognition of symbols (***input characters***) after they have been completely written which are read and digitized using an optical scanner (Zeki 2005; Al-Shatnawi and Omar 2014; Hachour 2006; Amin 1997).

There is a great demand on offline Arabic OCR for several purposes: Firstly, converting Arabic materials such as books and magazines into electronic forms and secondly, building bridges between the Arabs and other nations by providing universal access to historical and current Arabic literature (Beg et al. 2010). So, many techniques and intelligent methods such as fuzzy logic and neural networks are used to recognize Arabic characters with reasonable recognition rates (Albakoor et al. 2009; Amor et al. 2006).

The AOCR process in general is divided into several steps: image acquisition, preprocessing, segmentation, feature extraction, recognition and post processing (Zeki 2005; Alginahi 2013; Al-Shatnawi and Omar 2014; Abdulaziz and Alsaif 2008; Beg et al. 2010). Image acquisition step reads and digitizes the input image using an optical scanner. Preprocessing step cleans the document from noises, dots, etc. in order to enhance the quality of the document. Segmentation step segments the text into lines, words or sub words (***part of a word***) and finally characters. Feature extraction step extracts the features of each character. Recognition step recognizes the characters based on the features extracted from the previous step. Post processing step corrects the errors result from the previous step and resolves ambiguity by using additional tools such as spell checkers in order to improve the recognition accuracy (Zeki 2005; Alginahi 2013; Osman 2009; Beg et al. 2010).

The segmentation step, especially the character segmentation, is too important because the failure in segmenting characters correctly will lead to recognizing them incorrectly (Sari et al.2002). It can be used on both cursive such as Arabic and non-cursive scripts such as English but the most difficult one is the first one (Sari et al. 2002; Osaman 2013). Therefore, this study will concentrate on this part of the AOCR system.

The rest of the paper is organized as follows. In Section 2, the Arabic language characteristics are introduced to present the main challenges that face any AOCR system. In Section 3, the character segmentation algorithm is presented as well as its modification. The results of such modification are discussed in Section 4. Finally, Section 5 presents the conclusion.

2 ARABIC LANGUAGE CHARACTERISTICS

There are some Arabic characteristics that challenge the AOCR system. These are:

1. It is written from right to left.
2. It is composed from 40 characters (***see Table 1***), 10 numerals, punctuation marks, spaces and special symbols.
3. Each character can take up to four different forms/shapes in a word or sub word depending on its position (***see Fig. 1***); beginning form (***connected from the left ONLY***), middle form (***connected from both sides***), end form (***connected from the right ONLY***), or isolated form (***preceded by either a space or unconnected symbol and followed by a space or unconnected symbol***).
4. It is cursive which means that the characters are connected to each other, either from one side or both sides as mentioned earlier, along an imaginary horizontal line runs through the connected segments of the text known as baseline to make up a word or a sub-word (Shaikh and Shaikh 2005).
5. It has 29 different characters but due to the different shapes that some characters can take raise the number to 40. 31 of them can be classified into different groups so each group contains the characters that have exactly the same body but differ from each other in their complementary character ("A portion of a character that is needed to complement an Arabic character" (Amin and Mansoor 1997)). Deleting such complement leads to an unknown character.

6. It uses diacritics, which are short vowels and usually found in old manuscripts that are placed above or below the character. They are written as strokes either above or below the baseline. These vowels represent different meanings for the same word. For example "مدرسة" can be "school" or "teacher" depending on the used diacritics that solve such ambiguity.
7. The size of each Arabic character is inconstant (***in terms of height and width***) because each character has different shapes depending on its position in the sentence as mentioned earlier which increases the number of character shapes from 40 to 125 (***see Table 1***). In addition, the size also is inconstant between characters. Therefore, segmenting Arabic characters using fixed size of width (***Pitch segmentation***) is not applicable.
8. Some characters within the same word can overlap vertically without being touch which forms a "**ligature**" where the second character starts before the beginning or end of the first one. This makes determining the spaces between characters and words more difficult which leads to deal with some of them as a single character (***see Fig. 2***).
9. Existence of lines above and below the baseline known as: ascenders and descenders respectively. There are also holes in some of the Arabic (Zeki 2005; Sari et al. 2002).

Table 1. Arabic Characters in their different shapes

No	Character	Isolated	Beginning Form	Middle Form	End Form	Number of Shapes
1	Alef	ا	ا	ـا	ـا	2
2		أ	أ	ـأ	ـأ	2
3		إ	إ	ـإ	ـإ	2
4		آ	آ	ـآ	ـآ	2
5	Ba'	ب	بـ	ـبـ	ـب	4
6	Ta'	ت	تـ	ـتـ	ـت	4
7	Ta'amarbootah (a special form of Ta')	ة	-	-	ـة	2
8	Tha'	ث	ثـ	ـثـ	ـث	4
9	Jeem	ج	جـ	ـجـ	ـج	4
10	Ha'	ح	حـ	ـحـ	ـح	4
11	Kha'	خ	خـ	ـخـ	ـخ	4
12	Dal	د	د	ـد	ـد	2
13	Thal	ذ	ذ	ـذ	ـذ	2
14	Ra'	ر	ر	ـر	ـر	2
15	Zy	ز	ز	ـز	ـز	2
16	Seen	س	سـ	ـسـ	ـس	4
17	Sheen	ش	شـ	ـشـ	ـش	4
18	Sad	ص	صـ	ـصـ	ـص	4
19	Dhad	ض	ضـ	ـضـ	ـض	4
20	T'ah	ط	طـ	ـطـ	ـط	4
21	Th'ah	ظ	ظـ	ـظـ	ـظ	4
22	Ain	ع	عـ	ـعـ	ـع	4
23	Gain	غ	غـ	ـغـ	ـغ	4
24	Fa	ف	فـ	ـفـ	ـف	4
25	Qaf	ق	قـ	ـقـ	ـق	4
26	Kaf	ك	كـ	ـكـ	ـك	4
27	Lam	ل	لـ	ـلـ	ـل	4
28	Meem	م	مـ	ـمـ	ـم	4
29	Noon	ن	نـ	ـنـ	ـن	4
30	Haa'	ه	هـ	ـهـ	ـه	4
31	Waw	و	و	ـو	ـو	2
32	Ya'	ي	يـ	ـيـ	ـي	4
33	Hamza	ء	ء	ء	ء	1
34	Alef maqsoorah	ى	-	-	ـى	2

No	Character	Isolated	Beginning Form	Middle Form	End Form	Number of Shapes
35	Waw mahmoza	ؤ	-	ـؤ	ـؤ	2
36	Ya' mahmoza	ئ	ئـ	ـئـ	ـئ	4
37	Lam-alef (a combination of lam & alef)	لا	لا	ـلا	ـلا	2
38		لأ	لأ	ـلأ	ـلأ	2
39		لآ	لآ	ـلآ	ـلآ	2
40		لإ	لإ	ـلإ	ـلإ	2

الناس (End)	جسد (Middle)	سحاب (Beginning)	س (Isolated)

Fig. 1. Different forms of the Arabic character (سين)

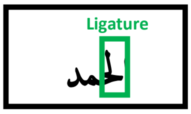

Fig. 2. Sample of Arabic ligature

3 THE ENHANCED CHARACTER SEGMENTATION ALGORITHM

The original algorithm that is used for segmenting characters was proposed in (Bushofa and Spann 1997). The main idea of this algorithm is divided into three parts: segmentation of touching characters, segmentation of the character End Ya shapes (ـي، ى، ئ) and segmentation of joined characters. The search area for all of these parts was based on the entire image by examining the upper and lower contours of the text. The upper contour is examined to search for the true segmentation points by using a threshold point t_1 while the lower contour is examined to search for the joined characters and the End Ya shapes characters.

The search area is enhanced by deducing the vertical scan. Instead of scanning the image vertically from top to bottom, it is scanned from the threshold value t_1 until the baseline ONLY because the characters are usually connected at the baseline (***see Fig. 3***). This led to three advantages: (1) decreasing the processing time by 86.42%, (2) avoiding the over-segmentation of descenders such as the letter Ra (ر), End Ya shapes (ـي، ى، ئ) characters because part of them appear below the baseline which is out of the search area's boundaries and finally (3) the unnecessarily for removing the complementary characters because they are also not included in the search area whether they are above or below the character's body.

However, in the case of detecting and segmenting the End Ya shapes (ـي، ى، ئ) characters, the pixel's color is examined. If it is changed between black and white and the number of changes is five (white → black → white → black → white) along the line (t_2) (***where t_2 is the symmetric line of t_1 around the baseline and between the left edge of the image and the first peak***), then this is an End Ya shapes characters. The first occurrence for the column of the last detected white pixel is the true segmentation point for such character.

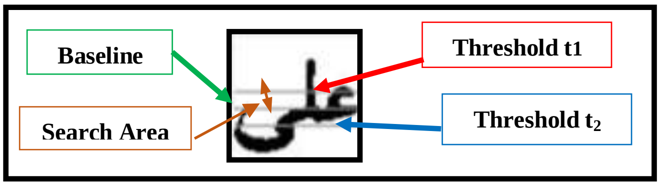

Fig. 3. Baseline, threshold t_1 and t_2

4 RESULTS

The sample used for testing the system is an Arabic book named as " القوانين الفقهية لابن جزي". Each sub word's image is scanned vertically from a certain threshold value until the baseline ONLY which led to decrease the processing time by average of 86.42% as mentioned earlier. The details of this percentage is illustrated in Table 2. Table 2 illustrates the average of three scanned pages and the average between them. In addition Fig. 4 illustrates the result of segmenting the End Alef maqsoorah character.

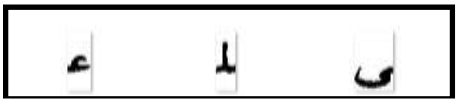

Fig. 4. Result of segmenting the word "على" which include the End Alef maqsoorah character

Table 2. The average percentage amount decreased from processing time of three scanned pages

Page #	Average % Decreased from Processing Time	Average % Decreased from Processing Time Between Pages
1	85.99%	86.42%
2	86.45%	
3	86.82 %	

5 CONCLUSION

The study tried to enhance the segmentation step in an AOCR system and especially the character segmentation sub step. Character segmentation extracts each character in each word/sub-word image separately by using the algorithm proposed in (Bushofa and Spann 1997). This algorithm has been enhanced by limiting the vertical search area for segmentation points to be from a certain threshold point to the baseline instead of the whole height of the image. As a result, the processing time has been decreased by 86.42 %, the segmentation of descenders such as the letter Ra (ر) and End Ya shapes (ـي، ى، ئ) has been avoided and extracting the complementary characters was not required.

REFERENCES

Albakoor, M., Saeed, K., & Sukkar, F. (2009). Intelligent System for Arabic Character Recognition, *World Congress on Nature & Biologically Inspired Computing*, pp. 982-987. http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5395597&isnumber=5393306

Abdulaziz, E. & Alsaif, K. (2008). Radon Transformation for Arabic Character Recognition, *International Conference on Computer and Communication Engineering, 2008. ICCCE 2008*, pp.433-438. http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4580642&isnumber=4580554

Alginahi, Y. (2013). A Survey on Arabic Character Segmentation, *International Journal on Document Analysis and Recognition (IJDAR)* , 16(2), pp. 105-126. http://link.springer.com/article/10.1007%2Fs10032-012-0188-6

Al-Shatnawi, A., & Omar, K. (2014). The Thinning Problem in Arabic Text Recognition - A Comprehensive Review, *Intentional journal of computer applications (0975 - 8887)*, 103(3). http://research.ijcaonline.org/volume103/number3/pxc3898969.pdf

Amin, A. (1997). Off line Arabic Character Recognition- A Survey , *Proceedings of the Fourth International Conference on Document Analysis and Recognition*, pp. 596–599.

Amor, N., Zarai, M., & Amara, N. (2006). Neuro-Fuzzy Approach in the Recognition of Arabic Characters, *Information and Communication Technologies, 2006. ICTTA '06. 2nd*, 1(), pp.1640-1644 http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=1684630&isnumber=35470

Amin, A., & Mansoor, W. (1997). Recognition of Printed Arabic Text using Neural Networks, *Proceedings of the Fourth International Conference on Document Analysis and Recognition*, pp. 612–615. http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=620576&isnumber=13496

Beg, A., Ahmed, F., & Campbell, P. (2010). Hybrid OCR Techniques for Cursive Script Languages – A Review and Applications, *2010 Second International Conference on Computational Intelligence, Communication Systems and Networks (CICSyN)*, pp.101-105. http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5614787&isnumber=5614044

Bushofa, B., & Spann, M. (1997). Segmentation of Arabic Characters Using their Contour Information, *1997 13th International Conference on Digital Signal Processing Proceedings, 1997. DSP 97*, 2(), pp. 683-686

Cheung, A., Bennamoun, M., & Bergmann, N. (1997). Implementation of a Statistical Based Arabic Character Recognition System, *Proceedings of IEEE, TENCON '97. IEEE Region 10 Annual Conference. Speech and Image Technologies for Computing and Telecommunications*, 2(), pp.531-534 http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=648261&isnumber=14128

Hachour, O. (2006). The Combination of Fuzzy Logic and Expert System for Arabic Character Recognition, *3rd International IEEE Conference on Intelligent Systems*, pp. 189-191. http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4155422&isnumber=4118323

Najoua, B., & Noureddine, E. (1995). A Robust Approach for Arabic Printed Character Segmentation, *Proceedings of the Third International Conference on Document Analysis and Recognition, 1995*, 2(), pp. 865-868. http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=602038&isnumber=13256

Osaman, Y. (2013). Segmentation Algorithm for Arabic Handwritten Text based on Contour Analysis, *2013 International Conference on Computing, Electrical and Electronics Engineering (ICCEEE)* , pp. 447 - 452. http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6633980&isnumber=6633894

Osman, Z., Hamandi, L., Zantout, R., & Sibai, F. N. (2009). Automatic Processing of Arabic Text, *International Conference on Innovations in Information Technology*, pp. 140-144. http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5413793&isnumber=5413354

Sarfraz, M., Nawaz, S., & Al-Khuraidly, A. (2003). Offline Arabic Text Recognition system, *Proceedings of the 2003 International Conference on Geometric Modeling and Graphics*, pp. 30 – 35

Sari, T., Souici, L., & Sellami, M. (2002). Off-line Handwritten Arabic Character Segmentation Algorithm: ACSA, *Eighth International Workshop on Frontiers in Handwriting Recognition, 2002. Proceedings*, pp. 452 - 457. http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=1030952&isnumber=22139

Shaikh, N., & Shaikh, Z. (2005). A Generalized Thinning Algorithms for Cursive and Non-Cursive Language Scripts, *9th International Multitopic Conference, IEEE INMIC 2005*, pp. 1-4. http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4133402&isnumber=4133397

Zeki, A. (2005). The Segmentation Problem in Arabic Character Recognition the State of The Art, *First International Conference on Information and Communication Technologies, 2005. ICICT 2005*, pp. 11-26. http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=1598538&isnumber=33619