Printed Arabic Optical Character Segmentation

Khader Mohammad[a], Muna Ayyesh[b], Aziz Qaroush[c], Iyad Tumar[d]

[a,b,c,d]Birzeit University, PO Box 14, Ramallah, Palestine,

ABSTRACT

A considerable progress in recognition techniques for many non-Arabic characters has been achieved. In contrary, few efforts have been put on the research of Arabic characters. In any Optical Character Recognition (OCR) system the segmentation step is usually the essential stage in which an extensive portion of processing is devoted and a considerable share of recognition errors is attributed. In this research, a novel segmentation approach for machine Arabic printed text with diacritics is proposed. The proposed method reduces computation, errors, gives a clear description for the sub-word and has advantages over using the skeleton approach in which the data and information of the character can be lost. Both of initial evaluation and testing of the proposed method have been developed using MATLAB and shows 98.7% promising results.

Keywords: Optical Character Recognition, Arabic characters, segmentation, recognition, machine Arabic printed text.

1. INTRODUCTION

Few efforts have been put in developing efficient Arabic OCR systems compared to the other languages. This is due to the complicated Arabic text characteristics, and the structure of letters which differs according to its position in the word; at beginning, middle, or end, many difficulties arise through the development process especially in the segmentation stage from which most of errors come and affects negatively the recognition rate. One of the major and important factors that enhances the information retrieval systems is the efficiency of the data entry operation from different sources of documents like invoices, receipts, bank statement, passport document and other systems as mentioned in [14,15,16].

Image acquisition is the first step in the character recognition process. The goal of this step is to transform the input text into a digitized image. The input might come from a camera or a scanner; the scanner is most commonly used as it is more convenient [11, 12].

The separation of writing into individual characters or segments is called segmentation. The segmentation stage is a necessary step in recognizing Arabic characters. An error in segmenting the basic shape of the characters will produce errors in the identification of each character. There are three types of segmentation, line segmentation, word segmentation and character segmentation. Horizontal projection can be used for line segmentation, vertical projection can be used for word segmentation and over-segmentation with pixel tracking can be used for character segmentation [12]. For sub words, contours can be used to separate them from the original word. Character

Image Processing: Algorithms and Systems XIII, edited by Karen O. Egiazarian, Sos S. Agaian, Atanas P. Gotchev, Proc. of SPIE-IS&T Vol. 9399, 939911 · © 2015 SPIE-IS&T · CCC code: 0277-786X/15/$18 · doi: 10.1117/12.2083587

recognition systems can be divided into two approaches depending on segmentation, segmentation-based systems and segmentation-free systems. Segmentation-based systems divide a single word into its primitives (characters or parts of a character) and recognize these characters separately. After applying vertical and horizontal projections to segment words into characters, a low level segmentation is applied for dots and other zigzag features. As for segmentation-free systems, recognition is done for words without segmenting them into characters or primitives [12, 13].

The segmented units can be characters, diacritics or other units depending on the recognition techniques used. Most of the diacritics documents such as dictionaries and holy books are transformed to digital formats that can be used for information retrieval techniques used in the search engines in which optical character recognition systems become one the main stage to transfer the documents. Having a segmentation algorithm that deals with Arabic text with diacritics is essential for success of Arabic OCR system. Intensive research has been carried out in this area with a large number of technical papers and reports in devoted to character recognition. This subject has attracted research interest not only because of the challenging nature of the problem, but also because it provides the means for automatic processing of large volumes of data in dictionary, books, postal code reading [1,2]. Table 1 summarizes previous art segmentation approaches.

The proposed approach benefits directly from the contour method approach by using only the up contour as shown in Figure 1 (A) and added new methods to do segmentation and deals with the diacritics.

Table 1: Splitting area ignore in case of SAD and DAD

Year/Reference	Approach	Recognition Rate	Disadvantage
2004 -[4]	Segmentation Based	96.5%	The algorithm is used only for 'Simplified Arabic' font type
2005 -[6]	Segmentation Based	97%	- A pre-processing technique is used to adjust the local base line for each sub-word. - A post processing stage is used to adjust the segmented characters
2008- [7]	Segmentation Based	Not reported	An iterative process was used to detect the connected components based on the connected black pixels in the sub-word - overhead-
2009 -[8]	Segmentation Based	Not reported	Some faults in the algorithm were registered, and that need extra work to solve the under and over segmentation problems
2010 -[9]	Holistic Based	92%	Each time the features are extracted from the image and feed them to some recognizer for identification purpose, and this leads to extra processing.
2013 -[10]	Segmentation Based	98.02%	- The word fragmentation is done without considering dots and other signs of the word's characters - Adjustment is done for merging small fragments that are parts of a character

2. PROPOSED APPROACH

The proposed segmentation algorithm uses the contour finding approach for sub-word segmentation. This approach has many advantages over finding the skeleton of the word, in which word's information can be lost, and that leads to less recognition rate.

Morphological operations have been applied to the sub-word to get its edge, then the up-contour is extracted by electing two points as beginning and end points on the original contour. The up-contour is formed by tracking the path from the beginning point to end point in counterclockwise direction. After extracting the up-contour, it goes for extra processing and new algorithm is applied to find the splitting areas over the up-contour. Some of these splitting areas are extra and ignored in case of sad (ص), dad (ض), seen (س), and sheen (ش). For this reason, these splitting areas are passed to a new algorithm to determine whether to be ignored or not. In this algorithm, some rules are applied to determine the sad, dad, seen and sheen characters.

The proposed segmentation approach provides a solution for overlapping characters using the connected components which many algorithms failed to apply. In addition, the diacritics are determined by using region calculations of binary image like centroid computation, The proposed steps as shown in Figure 1 (B) for text segmentation process are:

1. Convert the image to grayscale format and localize the text: Since the input image is type of text, it's more efficient to convert the image to grayscale format; in this step the image is examined, if it's in RGB which is the true color format for the image, then it is converted it to grayscale format by using rgb2gray function.

The brightness information is obtained by merging RED, GREEN, and BLUE in the true color image with ratio reach experimentally up to 30%, 60%, and 11% for each one respectively. This format is used in the segmentation algorithm side by side with the binary one. Figure 2 (A) shows the input image after it is converted to grayscale format.

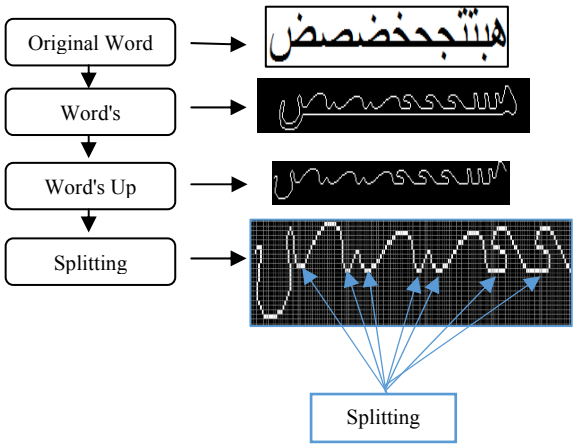
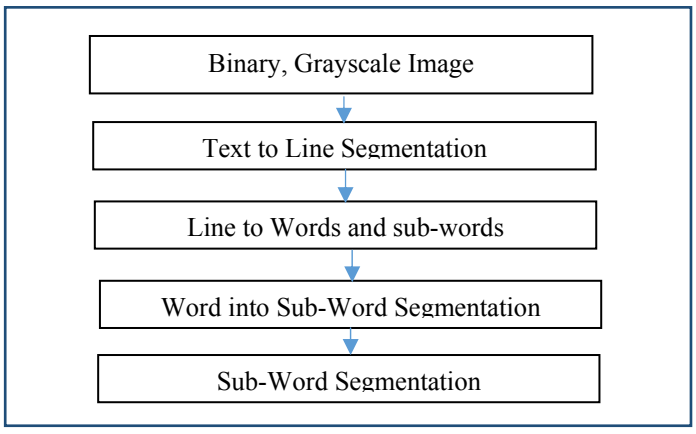
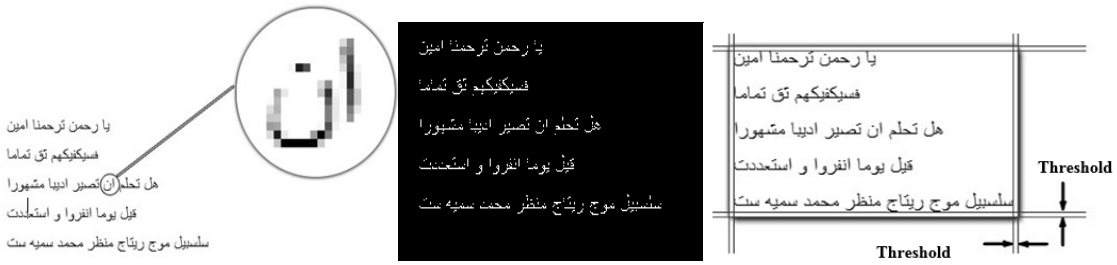


Figure 1: (A) Splitting areas extracted from the up-contour



(B): Proposed steps for Arabic text segmentation process

The text is cropped from the image by removing the surrounding white space (white in the original image, black in the binary image) is been conducted. The binary image as shown in Figure 2 (B) is cropped in this stage, the surrounding empty space is removed so the image is prepared to be passed to segmentation stage, the cropping is done by finding the regions in the image with white pixels, then by getting the maximum and minimum rows and columns positions in the image which have data, after that the boundaries are determined to be limited by maximum row, minimum row, maximum column, and minimum column. The grayscale also cropped by using the above values, just a threshold space value is added to keep all grayscale image data from loss because the size of cropped binary image is less than the size of the grayscale image, Figure 2 (C) shows the threshold results value added to grayscale image.



(A) (B) (C)

Figure 2 : (A) RGB input image (B) Binary image format from OCR input image grayscale image -generated from the cropped binary image (C) Threshold value added to the dimensions- of the gray image

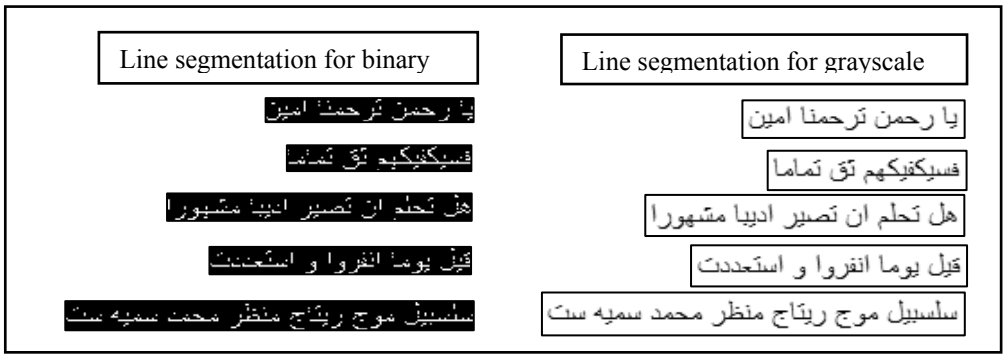


Figure 3: Text to line segmentation output

2. Text to Line Segmentation: In this step horizontal projection is used to extract the lines from the binary image, the binary image is scanned row by row to determine the margins with no data (fully black continuous space), Figure 3 shows the output of this stage. The binary image is represented by two dimensional matrix, and the position of the pixel is represented by the column and row values within the matrix. The margins with no data, has continuous rows with black pixels, the row with black pixels is represented in (1), where pi is a pixel at row position equals i.

$$\sum (Pi) == 0 \qquad (1)$$

3. Line to Words and sub-words Segmentation: The words and sub-words (which are separated by black spaces) are extracted in this stage using vertical projection technique. The given binary image (line segment) is scanned, and the indexes for columns with full black pixels are extracted. Then the indices values are grouped; each group contains continuous numbers for indices, for example the set of fully black columns in the image holds the following indices:
{50,51,52,53,91,92,93,94,95,96, 101,102,103,104,110,111,112,113}
The values in the previous are classified into 4 groups ; the set for the first group contains the following values {50,51,52,53}, the second set is {91,92,93,94,95,96}, the third set is {101,102,103,104}, the last set is {110,111,112,113}, the resulting sets are considered as splitting regions between words and sub-words (separated by the black space) in the given line image. The words and sub words are assigned by comparing the number of indices in each set with a threshold value. The threshold value is determined according to the width of the given line image. For example in the previous data set if the threshold value is considered to be 4, then every set contains more than 4 numbers it will be considered as a separation region between two words, else the separation region is located between two sub-words, each word or sub word locates between the maximum and minimum values in two different sequential sets; So the algorithm detects one word consist of two sub-words, one word consist of two sub-words, and one word consist of one sub-words respectively, refer to Figure 4 (A). The operation of splitting the line binary image goes side by side with the operation of splitting the line in grayscale image, but extra work is done to the resulting separated parts from the grayscale line; the values of boundary splitting region are modified by adding a threshold value to the boundaries. For example if the word locates between {40, 75} columns the grayscale word locates between {36, 79} in case of the threshold value equals 4, Figure 4 (B) shows the segmentation results of this stage.

4. Contour Extraction: the contour is extracted from the given grayscale image according to the different intensities found in the image, the resulting figure from the operation has four levels contour, the outer contour is extracted from the figure, then the figure is transformed to binary image.The resulting binary image which contains the outer contour has some problems because of the gaps appear in, so the contour image is passed to another stage to fill the gaps and to do some enhancements for the outer contour, like smoothing removing inside separate holes, Figure 5 shows the extracted contour for many sub-words.
5- Sub-Word Segmentation: The contour of the sub-word is passed to the stage as shown in figure 6, then passed to several processing steps, initially the main body of the sub word is determined and the diacritics like points also determined. The sub-word segmentation then passed through the following steps:

a) **Up-Contour Extraction:** In this step two points on the resulting contour are elected as start and end points for the path that will form the up contour. The start point locates in the first part of the first character in the sub-word contour, and the end point locates in the last part of the last character the sub-word contour. To determine the first part of the first character and

the last part of the last character, a threshold value first is determined according to the average length for the character for the current used pen size, so the first part of the first character locates between the maximum column index of the sub-word contour and second column which is determined by subtracting the previous threshold value from the max column index. In the same manner the last part of the last character is determined but in this time the region locates between the first column index in the contour and the second column is assigned by adding the previous threshold to the first column index value, 2- 2-2-b) b).

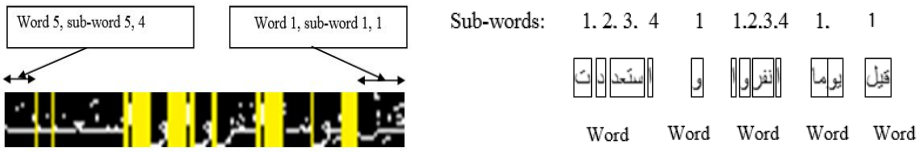


Figure 4: (a) Line to word and sub-word segmentation (b) Words and sub-words segmentation result

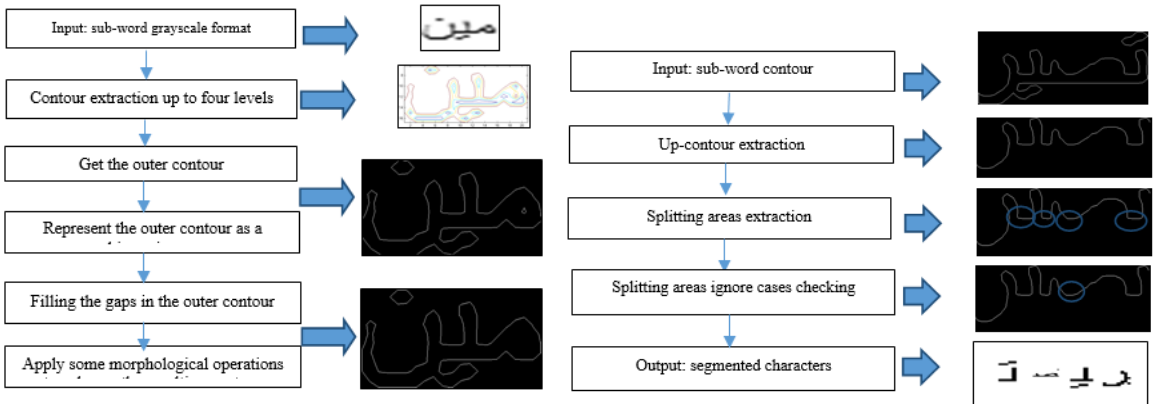


Figure 5 (a): Contour extraction steps *(b): Sub-word segmentation process*

b) Splitting Areas extraction : At this step the up-contour is scanned to extract the splitting areas, at which the sub-word can be segmented, by applying this approach no need to extract the baseline– the baseline detection and the errors appear from this step are eliminated by using this approach, the errors in assigning the baseline causing the major source of errors in the segmentation stage-, the extraction of the splitting areas is done by following:

1- Scan the up-contour row by row.

2- Get the continuous regions in every scanned row.

3- Check the resulting continuous regions in each scanned row.

4- Get the minimum column value for each region (the expected minimum column for the splitting area which should be examined)

To consider this point as one of the splitting region reference points some rules should be satisfied as following:

If one of the pixels in the previous row has the column index equals the current column index or equals the current column index minus one has value equals one then this point is considered as the first point in the splitting area.

5- Get the minimum column value for each region (the expected minimum column for the splitting area which should be examined) .
   To consider this point as one of the splitting region points some rules should be satisfied as following:
   If one of the pixels in the previous row has the column index equals the current column index or equals the current column index plus one has value equals one then this point is considered as the second point in the splitting area.
6- If one of points doesn't satisfy the conditions then this is not a splitting area (the splitting area always performs a local minima at where it locates, Figure 7, 8 shows the first and second reference points for the splitting areas.
7- The extracted splitting areas in the up-contour of the sub-word are sorted from right to left to be passed for extra checking, Figure 7 shows the extracted splitting areas.
8- The splitting areas are passed to additional checking since some of these areas are extra and should be ignored.

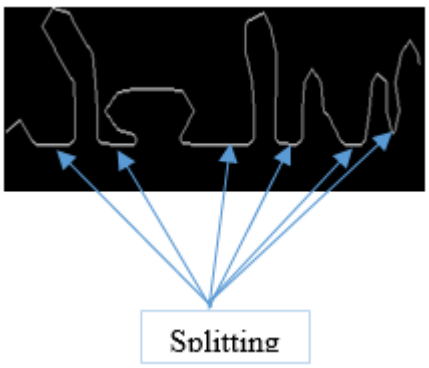


Figure 7: Up-contour splitting areas

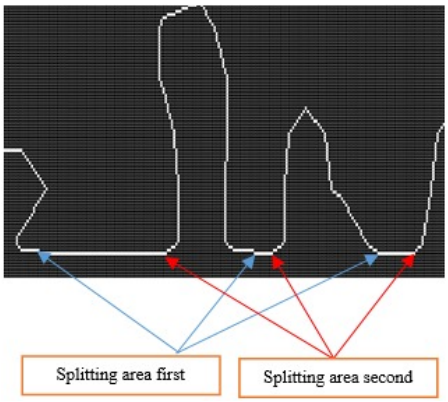
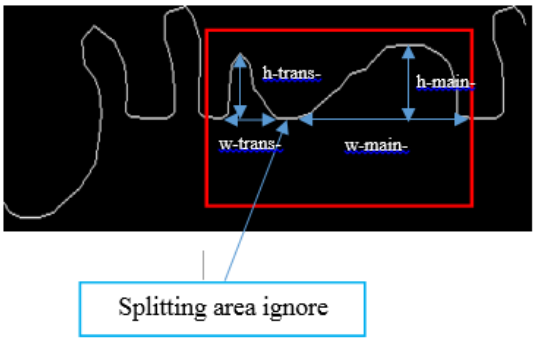


Figure 8: Splitting area reference points

The cases that the splitting areas should be ignored:

1- SAD (ص), DAD(ض), TAA(ط), and THAA (ض): SAD-DAD-IND algorithm is used, in this case the ratio between the height and the width of the transition area is determined, also the ratio between the height and the width of the of the main body of the character is determined, at the last the ration between the previous two ratios is calculated, if the ratio is between a certain threshold value **sad-dad-thres-values**, then the character satisfies the conditions and considered as SAD, DAD, TAA and THAA, refer to Figure.



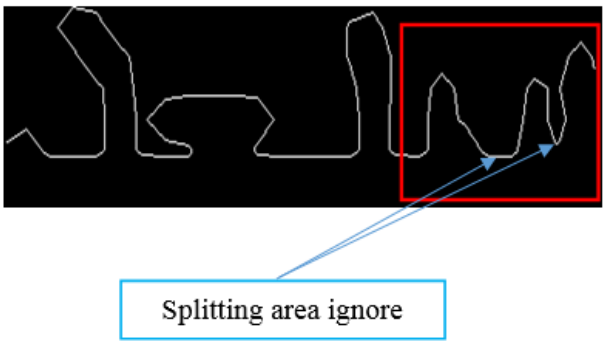


Figure 9: Splitting area ignore in case of SAD and DAD *Figure 10: Splitting areas ignore in case of SEEN and SHEEN*

2- SEEN(س) SHEEN(ش) case :
As

before the transition area should satisfy the ignore conditions. Then the area between the previous splitting area first reference point and the previous of the previous splitting area first reference point is checked if it has diacritics points. The checking for the existence of dots in this region is done by determining the overlapping diacritics first, then the overlapped area is determined, after that the ratio between the size of the overlapped area to the size of the diacritic (dot), if the calculated ratio is more than a threshold value, then the character in this case is sheen.

For seen case the number of the transition areas that stratifies the conditions of ignore are counted if they equals to three then the character is seen and the previous two splitting areas should be ignored in this case, Figure 10 shows the splitting areas that should be ignored in case of sheen letter.

3- The case of the last splitting area: the last splitting area should be ignored in some characters locate at the end of the sub-word,

As a future work will be working in improving accuracy for segmentation approach with diacritics.

3. CONCLUSION

Segmentation of Arabic text is error-prone. It is the stage where most of the errors occur and where the error in segmentation will result in classification errors. In this proposal, a new scheme is investigated and developed such that the segmentation is done in such a way to minimize errors and maximize the recognition rate. Initial results using the proposed scheme shows promising results (98.7%) of the Arabic character is achieved. Our proposed scheme overcome of the previous problems by

- Taking into consideration the overlapping between sub-words and characters,

- Depending on the study of the region properties for the given image, calculations for area, centroide, convex area and solidity have been done to determine the sub-words, in this step
- Using the connected components approach for the binary image that gives more accurate results
- Avoiding extra processing done by the iterative process.

REFERENCES

[1] Kchaou, G,Kanoun, S&&Ogier, J 2012, 'Segmentation and Word Spotting Methods for Printed and Handwritten Arabic Texts: A Comparative Study', in Proceedings International Conference on Frontiers in Handwriting Recognition, Pari, pp. 274-279.

[2] Ahmed, P && Al-Ohali, Y 1999, 'Arabic Character Recognition: Progress and Challenges', Journal of King Saud University - Computer and Information Sciences, vol.12, pp. 85-116.

[3] Hassin, A, Jian-hua, H && Xiang-long, T, 2003, 'Offline Arabic character Recognition system', Journal of Harbin Institute of Technology, vol. 10, no. 1, pp. 80-88.

[4] Dinges, L, Al-Hamadi,L, Elzobi,M, Al Aghbari,Z && Mustafa, H, 2011, 'Offline Automatic Segmentation based Recognition of Handwritten Arabic Words' , International Journal of Signal Processing, Image Processing and Pattern.

[5] Mostafa, G, 2004, 'An Adaptive Algorithm for the Automatic Segmentation of Printed Arabic Text' ,In proceeding of: 17th National Computer Conference, pp. 437-444

[6] Omidyeganeh , M, Nayebi , K , Azmi, R &&Javadtalab, A, 2005, 'A New Segmentation Technique For Multi font FARSI/ARABIC Texts' , in Proceedings IEEE International Conference Acoustics, Speech, and Signal Processing, pp. 757-760.

[7] AlKateeb, J, Jiang, J, Ren, J &&Ipson, S 2009, 'Component-based Segmentation of Words from Handwritten Arabic Text', International Journal of Computer Systems Science and Engineering, vol. 5,no. 1,pp. 54-58.

[8] Shaikh, N, Mallah, G &&Shaikh, Z, 2009, 'Character Segmentation of Sindhi, an Arabic Style Scripting Language, using Height Profile Vector' ,Australian Journal of Basic and Applied Sciences, vol. 3(4), pp. 4160-4169.

[9] Javed, S, Hussain, S, Maqbool, A, Asloob, S, Jamil, S &&Moin, H, 2010, 'Segmentation Free Nastalique Urdu OCR', Journal of World Academy of Science, Engineering and Technology.

[10] Alipour, M, 2013, 'A New Approach to Segmentation of Persian Cursive Script based on Adjustment the Fragments', International Journal of Computer Applications, vol.50, no.11, pp. 21-2 .

[11] [A. Hassin, H. Jian—hua, &&T. Xiang—lon."Offline Arabic Recognition System", Journal of Harbin Institute of Technology (New Series), Vol. 10, No. 1, 2003.

[12] Xiu P. Xiu, L. Peng, X. Ding, &&H. Wang . "Offline Handwritten Arabic Character Segmentation with Probabilistic Model", Dept. of Electronic Engineering, Tsinghua University.

[13] Optical Character Recognition Program ABBY FineReader, http://www.fairfield.edu/documents/library/lib_abbyy.pdf.2013

[14] Optical Character Recognition, http://en.wikipedia.org/wiki/Optical_character_recognition

[15] Khader Mohammad, Sos Agaian, 'Practical Recognition System for Text Printed on Clear Reflected Material'. ISRN Machine Vision journal, Volume 2012 (2012), Article ID 253863, 16 pages

[16] Khader Mohammad; Sos Agaian; Hani Saleh, 'Practical automatic Arabic license plate recognition system', Proc. SPIE 7881, Multimedia on Mobile Devices 2011; and Multimedia Content Access: Algorithms and Systems V, 78810V (18 February 2011);