An Efficient Character Segmentation Algorithm for Recognition of Arabic Handwritten Script

Amani Ali Ahmed Ali[1,2]
[1]Department of Computer Science
Kuvempu University
Shimoga, India
[2]Department of Computer Science
Taiz University
Taiz, Yemen
dramaniali2@gmail.com

Suresha M.[3]
[3]Department of Computer Science
Kuvempu University
Shimoga, India
srit_suresh@yahoo.com

***Abstract*—This manuscript proposed a new novel algorithm for vertical segmentation because the accuracy of recognition will be very high if there is a good character segmentation. For the purpose of locate the segmentation points used the word image thinning to fetch the width of stroke of one pixel and to detect the ligatures of Arabic characters geometry and shape are used in the segmentation procedure. The proposed segmentation approach is worked with touching characters, in case of ligatures touching segmentation existing between the characters of consecutive closed and the existence of ligatures within characters in the case of open characters. It evaluated on IFN/ENIT, AHDB and our dataset. It shows higher accuracy with an excellent performance by decreasing over-segmentation problem which showed through open characters segmentation and correct the segmentation in cases of the touching characters or the miss-segmentation errors in images of word.**

Keywords—Character Recognition; Character Segmentation; Thinning; Vertical Segmentation; Word Segmentation

I. Introduction

In the last decades the recognition of character in freestyle and unconstraint handwritten document images has an important attention which convert the scanning text into an editable text and digital format where can be stored in the computer for future use[1]. So segment the text line into characters is a very significant step in the character recognition of handwritten scripts also segmentation of either text line or word into characters becomes very complex because the unconstrained and connected nature of characters of the handwritten images. It has many-sided range of application as language based learning, for blind reading aid, cheque processing of bank, and postal automation. Character recognition consists of two basic phases which are the segmentation and recognition, which the segmentation is a basic phase in character recognition[2],[3].

In most of optical character recognition, failing in segmentation of handwritten text accurately will result definitely in poor recognition, regardless of how well the stages of past and following are designed. It is exceptionally evident that a significant offer of recognition mistakes is ascribed to the stage of segmentation. Consequently, numerous specialists have focused their researchers in improving the methodology of segmentation[2],[4]. Improvement of new algorithm of segmentation is trial to support in recognition field of Arabic handwritten written characters.

For languages like English, Latin and, Chinese have sufficient study research but in Arabic document images are still active and not sufficient, so Arabic character segmentation is more complex because different styles of writing which make character and text line segmentations very complex, because of huge number of characters with their different shape in different location and large character set with high similarity of characters these all make the task of segmentation of off-line scripts is not an easy where a good system should be deal with various styles of handwriting, various font sizes, connected characters. The main phases in segmentation of text handwriting document images are segment into characters, words and line and finally, identification of character. Word image noise is removing in the stage of pre-processing; classify every character uniquely by extracting and identifying discrimination features in the extraction of feature stage and the utilizing a classifier in the classification stage.

A novel in this manuscript for Arabic handwritten document images segmentation is done to segment the documents into characters entities. Fundamentally an improved strategy is incorporated into the suggested technique for vertically connected partition of words, and new technique for word segmentation depends on efficient recognition of gaps between characters using template matching method. The word image width and height is calculated for the purpose of ligatures analysis in an accurate manner, the word scanned from top to bottom vertically and counted the foreground pixels number in each column, every columns position are saved for which the pixels of foreground black is 0 or 1 and finally built character shapes dataset in different location by analysis of local feature which help in template matching method. The suggested method efficiency on different datasets with various collections of touching and overlapping

978-1-5386-9319-3/19/$31.00 ©2019 IEEE

characters is proved in the experimentation. The over segmentation problem is solved in excellent results. The Arabic Handwriting datasets used are AHDB, our dataset and IFN/ENIT.

The following appears the paper organized: the various segmentation challenges in section 2, related work introduced in section 3, proposed approach in detail in Section 4, experimental results presented in section 5 and section 6 introduced the conclusions with the future work.

II. Various Segmentation Challenges

The segmentation of handwritten has many difficulties as the following:

- Arabic handwritten has 28 characters written from right side to left with different shapes. One character refers to their position inside the word image, in which every character has two or four various shapes. Also there are some particular characters such as ى and ء, those all will raise the classes' number from 28 to be recognized into 112, and because of the following characters ز, ر ,ذ ,ـد, ا which are only six characters which are from right only can connected that decreases classes number from 112 to 100.
- The nature of connected characters of Arabic handwritten might be connected to each other, in a word more than two character can be connected.
- The six characters ز, ر ,ذ ,ـد, ا if showed inside a word that will separated the word into sub-word which are connected components block, so words divided by spaces
- Arabic handwritten has characters with differences in their styles of writing and shapes of various writers.
- Dots may be showed as a stroke or hat, touched dots, or as two separated dots.
- In Arabic characters there is a line called baseline which is an imaginary line in which these characters are connected on it. Baseline is so less than the beginning character width and as thick as a point of pen. So to define the connected character end there is a thin part where there is a needful condition which is detection of this thin part. Some characters as س in their middle have thin parts, so it isn't sufficient.
- Some Arabic handwriting characters might overlap together with their characters of neighboring as shown in the figure 1.
- In handwriting styles or some fonts, some characters' strokes as س or ش are omitted.

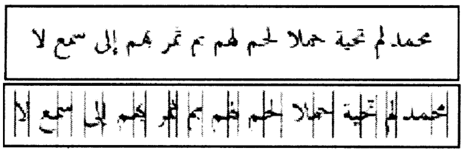


Fig. 1. Connected character segmentation as one character

III. Related Work

In any recognition system of character, the phase of segmentation is a significant task especially in the case of Arabic handwritten recognition. The methods' overview for handwritten characters' segmentation is presented in [5].

In [4] Methods are applied to achieve good segmentation using Hough transform and skeletonization for text segmentation and Gaussian mixture modelling framework method. That covers the possible false correction to create proficiency vertical connected characters' text segmentation. The segmentation of the Arabic words is pointed as a two class problem. The authors used fusion of Convex and Euclidean distance metrics to calculate the distance between neighboring overlapped components, which is classified as a distance of an intra-word or as an inter-word.

In [6] characters for word segmentation there were 3 basic approaches are defined which are holistic, internal segmentation and external segmentation. In holistic also called no segmentation, generic features of the entire word to purpose of recognition are extracted and no characters are extracted. In internal segmentation in same time both characters recognition and segmentation are accomplished. The most approach used to segmentation is the external segmentation which in prior to recognition of possible boundaries of the characters are found.

In [7] another segmentation method used which segments Arabic handwriting. Using characters' shapes information the algorithm over-segments evert word and after that extra breakpoints deleted. The accuracy rate that achieved in the algorithm was 92.2% and over segmentation the algorithm suffers from it.

In [8] another segmentation method used which depends on extracting characters connected block and search the features of topographic to identify possible points of the segmentation. In the block, the division of these points is depended on the character average width. The accuracy rate that achieved in the algorithm was 69.72%, with facing two main problems that failed in horizontal overlapping characters segmentation and Largely text segmentation of Arabic handwritten based on contextual information not only on extracted the features of the topographic from characters, which didn't deal with in the system.

In [1] another segmentation method used for text recognition which depends on analysis of subset of documents which are three main subset domains. To avoid detection failed authors used the skeletonization method, which did the correction in the situation of false alarms into build the segmentation of vertically connected characters. They achieved a high accuracy with two datasets which are 98.9 and 97.4 on AHDB and IFN/ENIT respectively.

In [9] another segmentation method used which depended on using tools of the mathematical morphology namely regularities and singularities. The segmentation candidates are the regularities. Applying an opening operation to the word image of the Arabic handwritten authors specified singularities along with regularities by subtracting the singularities from the original document images. The case of extracted from same segment two characters is existence. The accuracy rate that achieved in the algorithm was 81.88%.

IV. PROPOSED APPROACH

The following steps are the methodology which used to characters segmentation form Arabic document images as shown in the table 8.

Step 1: This called input step. Acquire the image which may be using a camera of digital or a scanner or where the image of input images was saved in the format of BMP or PNG or JPEG for next procedure and some were taking them from the dataset of IFN/ENIT.

Step 2: This called pre-processing step. It is important to remove noise from background to improve the word images quality to be utilized in the experiment of the segmentation, so the authors have to do the pre-processing to do input image enhancement to make them easy to do the next steps where the goal is to remove the conflict which inherent in connected character handwritten words. Because the samples of handwritten document might written on a colored background or noisy and also the word images' quality may be not good because the noise that presented in first step. Output of this stage used to increase the segmentation procedure performance. The following are the important steps:

- Noise Removal: In pre-processing, the noise which is small foreground components or dots may be presented into an image in the first step. Image noise here been removed from images
- The skew and slat correction incorporated by the method in [3].
- Thinning and Skeletonization: Because the writers may use various pen types with unequal width of stroke, a large variability amount might be introduced through the words of handwritten. The thinning procedure in this step of pre-processing helped in the proposed experiment to do a uniform stroke width of single pixel for all the words used.
- Thresholding: In the format of PNG with the images of RGB in this step of converted to the format of gray scale. To control the problems which may be appear like different colored backgrounds and noisy or various intensities of used pens and colors, in matrix of binary format these images of gray scale are converted, where the output images of this step binary images.

Step 3: The segmentation step. In an accurate way for the connected character analysis, the word image height and width of is calculated in [10]. From top side to bottom the image of word vertically scanned, in every column the pixels' number of foreground and column wise inside the inverted images of word are counted. Where the foreground sum with black pixels is 1 or 0 then every columns location of these are saved. The term PSC which denoted by Potential Segmentation Columns called for every identified column.

- Problem during segmentation: over-segmentation means in the whole word image in different groups, many sequential PSC are existent which the sum of pixels of the foreground are 1 or 0. In only 3 situations happened the over-segmentation issue.

  1. In case of Open Characters which didn't have any semi-loop or loop with presence of a ligature in every Open characters. Because this existence in character, the foreground pixels sum in these columns is 1 and the characters of such kind are over-segmented.
  2. In case of two sequential characters in the image of word are linked via a ligature and crossing this ligature the sum of pixels of the foreground are 1 in these columns.
  3. In case of two sequential characters in the image of word which didn't happen the touching characters and in this area the sum of pixels of the foreground of the columns are 0.

- The problem Solution: It has two possible solutions as the following:

  1. In case of in an image of word where between two sequential characters a clear vertical space is there, in that case taking the average in that area of all the PSC existence as the sum of pixels of the foreground for all these PSC columns is 0 the over-segmentation problem is removed.
  2. In case of Open characters where a ligature in character or between two sequential characters is there, via taking these PSC average where in a distance less than a threshold which is particular value means minimum distance among sequential PSCs along the word image width and the thinnest character width possible in an image of the word should be greater than threshold's value. For this case the authors assign the value 5 for the threshold. This means every PSCs those are divided through a distance of 5 pixels or less through another PSC will combined to a one SC, and via merging them into a one Segmentation Column (SC) then the over-segmentation is removed.

Step 4: To avoid the problems of touching and overlapping characters' segmentation, the authors built a dataset of characters' shapes in various location that store the main points by Local feature analysis which considers individual features. These features are the constructing blocks from which all character images can be constructed and distinguish the characters in different places. Local feature analysis selects features in each character that differ most from other characters such as, the loops and open characters and the areas where the ligatures of the changes. And after the previous segmentation in stage 3 for all components, the segmented component compare with every character inside database, form previous segmentation to detect other characters utilizing a template matching method. The following steps to do character's identity determination:

- The computer takes an image of that character as shown in figure 2.
- Specifies the points' pattern make that individual various most from other characters.
- System begins patterns' building, either based on average or randomly.
- For every picks, the computer builds a character image and compares it with target image to be identified.
- New patterns are created until a character image that matches with the target. When the match is found, algorithm segmented as one-character form the segmented components.

Fig. 2. Samples of characters pattern points

V. EXPERIMENTAL RESULTS

The input images are the images of word after thinning pre-processed which taken to the characters' segmentation. For further processing purpose to make the input images easier they are inverted to minimize the complexity of computation by the input images of binary complement where the black pixels become the background pixels and white pixels become the foreground pixels. So, in every word images column to count the white pixels of foreground which performed by 1 be easier. Now from the format of binary to a format of RGB the images are converted and in various color other than white and black to show the PSC becomes computationally easier. PSCs over-segmenting of the image of word in red color are printed. A PSC is when the sum of white pixels of the foreground is 1 or 0 for every column in the images of word, and the word image vertically cut. After that every PSCs are combined into one column which is called SC, in the case of it a distance less than a value of threshold which 5 pixels as mentioned before from each other. Finally, by updating the image black background with white background the final characters' segmentation is obtained.

From the samples of the handwritten of 1,050 different writers, 52,200 handwritten word samples are selected for evaluation of the suggested method of segmentation. The suggested method of segmentation performance was depended on the errors of segmentation of the three kinds, Bad Segmentations, Missed Segmentations and over segmentations which are mentioned in Table 1. Some words were bad segmented inside one location are as well as over-segmented in some other place in the case of incorrectly segmented words. As mentioned in table 1 such word kinds are counted in both of the classified of error as Bad-Segmented counted as well as Over-Segmented counted. In a few words similarly the point of correct segmentation was missed and shifted to some other area resulting in Bad-segmentation. In Table 1 showed the result of authors' proposed method after finished step 3 in the segmentation stage which tested on authors' dataset. In Table 2 showed the result of authors' proposed method after finished step 3 in the segmentation stage which evaluated on IFN/ENIT dataset. In Table 3 displayed result of authors' proposed method after finished step 3 in the segmentation stage which evaluated on AHDB dataset. Samples of result mentioned in Table 8 form three datasets. To increase the segmentation rate and correct some of the errors of segmentation phase, the authors check the characters' shape with different shapes in various locations as the phase of character recognition to separate or segment the characters those with touching and overlapping characters to make correct segmentation to them and those come under the false segmentation to delete them in which done in step 4 in the proposed method. In table 4 showed the result of proposed method after finished step 4 in the segmentation correction stage which tested on our dataset. In Table 5 showed the result of proposed method after finished step 4 in the segmentation correction stage which evaluated on IFN/ENIT dataset. In Table 6 showed the result of authors' proposed method after finished step 4 in the segmentation correction stage which evaluated on AHDB dataset. A sample of result displayed in Table 7.

Comparing the results of segmentation done by the suggested method with the results of segmentation of some other researchers is very difficult because various researchers used various databases of words of handwritten and reported the results of segmentation under different constraints like several authors supposed the noise absence, some researchers assembled the samples of handwriting from various writers number and so on. As the segmentation of character in word images achieved before the recognition phase of character, most of authors didn't mentioned the segmentation results and mentioned only the recognition results.

TABLE I. SEGMENTATION RESULT OF THE PROPOSED METHOD WITH OUR DATASET AFTER STEP 3.

	Correctly Segmented Words	**Incorrectly Segmented Words**		
		Bad-Segmented	***Miss-Segmented***	***Over-Segmented***
Words Number	45,697	4,049	197	2,257
Total	45,697	6,503		
Total Segmented rate %	87.5	12.5		
Total Number of Handwritten Words	52,200			

TABLE II. SEGMENTATION RESULT OF THE PROPOSED METHOD WITH IFN/ENIT AFTER STEP 3.

	Correctly Segmented Words	**Incorrectly Segmented Words**		
		Bad-Segmented	***Miss-Segmented***	***Over-Segmented***
Words Number	18,370	6,841	448	741
Total	18,370	8,030		
Total Segmented rate %	69.6	30.4		
Total Number of Handwritten Words	26,400			

TABLE III. SEGMENTATION RESULT OF THE PROPOSED METHOD WITH AHDB AFTER STEP 3.

	Correctly Segmented Words	Incorrectly Segmented Words		
		Bad-Segmented	*Miss-Segmented*	*Over-Segmented*
Number of Words	8,957	2,483	639	1,232
Total	8,957	4354		
Total Segmented rate %	67.3	32.7		
Total Number of Handwritten Words	13,311			

TABLE IV. SEGMENTATION RESULT OF THE PROPOSED METHOD WITH OUR DATASET AFTER STEP 4.

	Correctly Segmented Words	Incorrectly Segmented Words		
		Bad-Segmented	*Miss-Segmented*	*Over-Segmented*
Number of Words	50,169	1,201	318	512
Total	50,169	2,031		
Total Segmented rate %	96.1	3.9		
Total Number of Handwritten Words	52,200			

TABLE V. SEGMENTATION RESULT OF THE PROPOSED METHOD WITH IFN/ENIT AFTER STEP 4.

	Correctly Segmented Words	Incorrectly Segmented Words		
		Bad-Segmented	*Miss-Segmented*	*Over-Segmented*
Number of Words	22,626	2,639	283	852
Total	22,626	3,774		
Total Segmented rate %	85.7	14.3		
Total Number of Handwritten Words	26,400			

TABLE VI. SEGMENTATION RESULT OF THE PROPOSED METHOD WITH AHDB AFTER STEP 4.

	Correctly Segmented Words	Incorrectly Segmented Words		
		Bad-Segmented	*Miss-Segmented*	*Over-Segmented*
Number of Words	12,317	487	199	308
Total	12,317	994		
Total Segmented rate %	92.5	7.5		
Total Number of Handwritten Words	13,311			

TABLE VII. SAMPLE OF SEGMENTATION RESULT OF CHALLENGING WORD

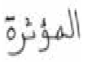
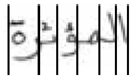
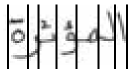
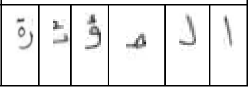
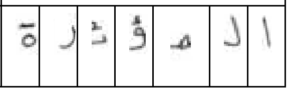
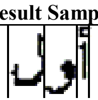
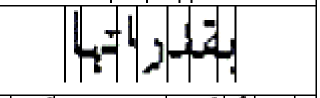
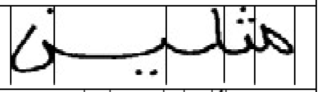
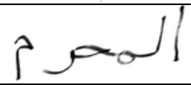
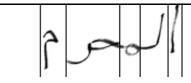
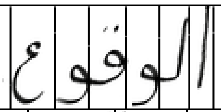
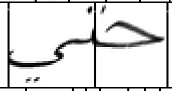
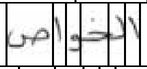
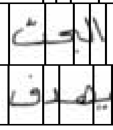
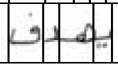
Word	Segmentation Results After Step 3	Segmentation Results After Step 4
		
		

TABLE VIII. SAMPLES OF SEGMENTATION RESULT S IN DIFFERENT DATASETS

Dataset	Input Samples	Result Samples
Authors		
		
		
IFN/ENIT		
		
		
AHDB		
		
		

VI. CONCLUSIONS AND FUTURE WORK

The proposed segmentation method decreased the problem of over-segmentation which showed through open characters segmentation also it can correct the segmentation in the cases of the touching characters or the miss-segmentation errors in images of word. This method has given excellent results in case of touching characters and ligatures segmentation existing between the characters of consecutive closed, with guaranteed correct segmentation in case of word document images without touching characters. in the case of open characters the existence of over-segmentation was because the existence of ligatures inside characters. The ligature inside characters in case of open characters were showed sometimes like they are link between two sequential characters and were over segmented through the suggested method.

In future work, the improving of the step of pre-processing and use some intelligent methods as a neural-based to validate the points of correct segmentation and do better with the improving of the accuracy of segmentation.

Acknowledgment

Taiz University of Applied Science, Department of Computer Science for supporting this research.

References

[1] M. Suresha, and A.A.A. Amani, “Segmentation of handwritten text lines with touching of line,” International Journal of Computer Engineering and Applications, vol. 12, No. 6, pp. 1-12, June 2018.

[2] D. Motawa, A. Amin, and R. Sabourin, “Segmentation of Arabic cursive script,” International Conference on Document Analysis and Recognition, vol. 2, pp.625-628, 1997.

[3] A.A.A. Amani, and M.Suresha, “A novel approach to correction of a skew at document level using an arabic script,” International Journal of Computer Science and Information Technologies, vol. 8, No. 5, pp. 569-573, 2017.

[4] A.A.A. Amani, and M. Suresha, “Efficient algorithms for text lines and words segmentation for recognition of arabic handwritten script,” In Emerging Research in Computing, Infromation, Communication and Application (ERCICA 2018), Springer, 2019.

[5] Y. Lu, and M. Shridar, “Character segmentation in handwritten words−an overview,” Pattern Recognition, vol. 29, No. 1, pp. 77-96. 1996.

[6] C. C. Tappert, C. Y. Suen, and T. Wakahara, “The state of the art in on-line handwriting recognition,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 12, No. 8, pp. 787-808, 1990.

[7] L. Lorgio, and V. Govindaraju, “Segmentation pre-recognition of arabic handwriting,” Proceedings of the Eighth International Conference on Document Analysis and Recognition, vol. 2, pp. 605-609, 2005.

[8] A. Hamid and R. Haraty, “A haraty neuro-heuristic approach for segmenting handwritten arabic text,” International Conference on Computer Systems and Applications (ACS), IEEE, pp. 110-113, June 2001.

[9] D. Motawa, A. Amin and R. Sabourin, “Segmentation of arabic cursive script,” Proceedings of the Fourth International Conference on Document Analysis and Recognition, vol. 2, pp. 625-628, August 1997.

[10] A. Rehman, D. Mohamad, G. Sulong, “Implicit vs explicit based script segmentation and recognition: a performance comparison on benchmark database,” Int. J. Open Problems Compt. Math., vol. 2, No. 3, pp. 352-364, 2009.