World of Computer Science and Information Technology Journal (WCSIT)
ISSN: 2221-0741
Vol. 2, No. 3, 98-104, 2012

A Three Stages Segmentation Model for a Higher Accurate off-line Arabic Handwriting Recognition

Said Elaiwat

School of Computer Science & Software Engineering (CSSE),
The University of Western Australia (UWA),
Australia 35Stirling Highway CRAWLEY WA 6009

Marwan AL-abed Abu-zanona

Department of Computer Science
Imam Muhammad Ibn Saud Islamic
University- Al-Ehsa Branch
Al-Ehsa – Saudi Arabia

Farah Hanna AL-Zawaideh
Department of Computer Information System
Irbid National University
Irbid, Jordan

Abstract - Arabic handwriting recognition considers a one of the hardest applications of OCR system. The reason of that relates to characteristics of Arabic characters and the way of writing cursively. Furthermore, no rules can control on handwriting way, different styles, sizes and curves make the process of recognition is very complex. On other side, the key for reaching to good recognition is by getting a correct segmentation. Actually, the way of segmentation is important, because if there is a small part is not clear in character that will reflect on recognition process. In this paper we aim to enhance the accuracy of off-line Arabic Hand Written text segmentation. Three stages are proposed to reach to highest ratio of segmentation. Line segmentation is the first stage, where it is proposed to separate each line. We depend on row density to predict spaces among lines. Second stage is Object segmentation and it is proposed to segment each word or sub word. Eight neighbors connectivity are used to detect connected pixels. Final stage is shape segmentation which is proposed to segment sub word to characters. The idea in this stage is finding segmentation points among branch points in the baseline. To apply that we propose four threshold values to investigate on each branch point. The result was satisfactory and the model proved a good ability to tackle different types of texts with bad samples.

Keywords- Arabic handwritten recognition; Segmentation; Image processing; Pattern recognition.

I. INTRODUCTION

Many years have been passed since the first Arabic segmentation and recognition of handwriting word was proposed. Since then, incredible progress has been made to enable computers to recognize, interpret and identify word printed and handwritten script. Extensive research has been carried out in terms of technical papers and reports by various researchers around the world. The motivation may be attributed to the challenging nature of the character segmentation and recognition problem and the countless number of commercial applications that it may be applied(Suen, Legault, Nadal, Cheriet and Lam 1993). many researchers have made big efforts in this area (Lorigo and Govindaraju 2006), Khatatneh 2006)), but unfortunately the result of Arabic handwriting recognition and segmentation still have not reached to the required level. The reasons for that relate to the nature of Arabic writing, where most words are written cursively and sometimes are not cursive depend on the character, some characters can be connected with others and some cannot. Most characters have three or four shapes according to their position in the word: “Isolated or Single”, “Beginning”, “Middle” and “End”. Also external object are used in Arabic writing like "dots", “Hamza” and movements that make the task of segmentation is more complicated. In additional, Characters that do not touch each other but occupy a shared horizontal space increase the difficulty of segmentation.

Furthermore, different writers and the same writer under different conditions write some Arabic characters in completely different ways (Al Hamad and Abu Zitar 2010). Figure1 explains shapes of Arabic characters

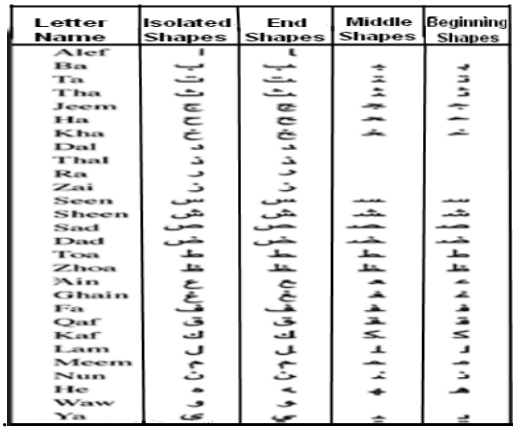

Figure1. Shapes of Arabic characters

There are two types of Arabic hand writing inputs, offline inputs and online inputs. People usually don't distinguish the difficulty between them. In online systems, recognition devices like (PDAs) where user writes on special screen by pen, the system here directly recognize on what was written. While in offline systems, usually scanned papers enter to system contain paragraph need to analyse and recognize. Offline systems contain steps more than online systems, we don't need to Image pre-processing in online system, also segmentation is easier each sub-word directly recognize. In offline system, sometimes we have to deal with big paragraph, multi lines and poor scan or old paper, which give a lot of noises.

Our contribution in this work propose a new algorithm that is able to segment a variety of Arabic handwritten words, different type of text included different text styles such as overlapping words, rotation in lines and more , included to these features , it can separate each character according to their position (beginning , middle , end, isolated) that's leads to facilitate the recognition process, included that the segmentation algorithm can deal with the image as a whole from the whole page to the isolated character.

Segmentation model is divided into three stages: line segmentation, object segmentation and shape segmentation. First two stages are simple but important to reduce the complexity of shape segmentation. Our model searches for segmentation points, these points appear when moving from character to another in the same object (word). So, in the third stage, we investigate on all branched points from baseline if they can be a segmentation points or not. We propose four threshold values to examine each branch point.

This paper paper is organized as the following: section two explains background and related work, segmentation model is reviewed in section three, section four presents experimental results for proposed model and finally conclusion and future work in section five.

II. BACKGROUND AND RELATED WORK

The main reason for this relates to the degree of complexity inherent in segmenting Arabic handwritten characters. Character segmentation and recognition systems in generic are described by (Impedovo, Ottaviano and Occhinegro 1991). Segmentation process is a first step before features extraction and classification. Actually if this step was not good enough, the result of next steps will be worse. it is important to realize the difficulty of segmentation, we cannot segment without knowing characteristics of each character, how they seem and how they connect to each others. This lead us to use recognition process to get full features, but recognition comes after segmentation and depends on it. So we cannot depend on recognition process to get features of characters which are needed in segmentation process, anyway knowing the structure of all characters in different positions in the word is the main key for good segmentation. (Ball, Srihari and Srinivasan 2006) presented comprehensive analysis of Arabic handwriting characters. Characteristics of Arabic writing were described in details, and the main problems and limitations were studied. Also they analyzed the main important features of written characters.

Nawaz, Sarfraz, Zidouri and Al-Khatib (Nawaz, Sarfraz, Zidouri and Al-Khatib 2003) proposed three steps to segmentation Arabic scripts. First step is Segmentation of text to lines by using the Horizontal Projection on the Document Image. Each Line of text is divided to three Zones; Upper Zone, Middle Zone, Base Line Zone and the Lower Zone. The Baseline zone is the zone with the highest density of Black Pixels. Second step is Segmentation of lines to words by using the vertical projection profile. Third step is segmentation of words to individual characters, the word is segmented to characters. Firstly, the vertical Projection of the middle zone is created. Next, the word is scanned from right to left. The connection area between two characters is determined according to the value of the vertical profile of the middle zone, if it was less than two thirds of the baseline thickness, the area will be a connection area. After that, determining the start of new character area, which is an area that has a larger value. This process is repeated until the full line is treated.

Recently, Jiang, Petkov, Alamri, He and Suen(Jiang, Petkov, Alamri, He and Suen 2009)proposed a new approach to segment and recognize on Arabic handwritten numeral touching, three stages were applied to reach to complete recognition. First stage is Segmentation of Touching Pairs where each image was divided into two regions to represent digits, 25 models were generated with different parameters to reach to candidate set of regions. Second stage is Isolated Digit Recognition, gradient features were used to extract feature from gray image (segmented region) while SVM was applied for classification. Final Stage is Post Processing Module, in this stage the best model among all models will be chosen according to the result of classification images in each model. The average rate of recognition in isolated digits was 98% , while it was 92.22% in Arabic digits, 90.43% in Urdu digits and 86.09% in Dari digits.

Al Hamad and Abu Zitar (Al Hamad and Abu Zitar 2010)applied a hybrid method to segment Arabic handwritten characters. The hybrid method consists of two components, first one is a heuristic component to segment text (pre-segmentation) and the second one is artificial neural network component, which was used to verify if pre-segmentation points are valid or not.

Arabic handwriting line segmentation based on Affinity Propagation was proposed by Almageed, Kang and Doermann(Kumar, Abd-Almageed, Kang and Doermann 2010), the proposed approach consists of two main steps, first step is coarse text line estimation where all diacritic components that don't represent main strokes were removed, and then local orientation detection and Shortest path algorithm were used to get similarity among primary components, while Breadth-first search and Affinity propagation were used to get different estimates. Second step is Diacritics Assignment; all diacritic components that had been removed in first step were assigned to the best coarse component. Accuracy rate was 96% with ability to deal textlines with uniform and non-uniform skew.

III. SEGMENTATION MODEL

The first step in proposed model is image pre-processing, as we know, digital capture of images produce noise from scanning devices, we need to remove or reduce this noise, most noise here are pepper and salt which give black dots in images, median filter and wiener filter were used for noise reduction, after that the image converts to black and white (binary image), figure2 illustrates proposed model.

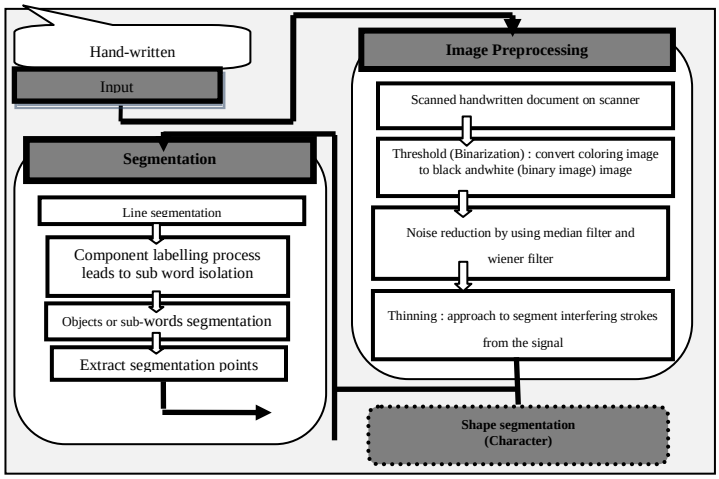

Figure 2.Three stages Segmentation model

Segmentation: In the proposed method, segmentation is performed in three levels: lines segmentation, objects or sub-words segmentation and characters segmentation.

A. Line segmentation

Algorithm of line segmentation depend on, firstly finding series of raw that the sum of each raw in this series is over than 0, now the first raw has sum over than 0 is upper point for line segmentation. Secondly the first raw that have sum equal zero is considered the lower point for line segmentation. Segmentation happens between upper and lower point to give the line, and so to gives all lines in the image. Figure 3 and 4 explain that.

Next algorithm represents line segmentation:

For n=1 to N

$$S(n)=\sum_{k=0}^{m} x(m,n) \qquad (1)$$

IfS(n)>0 and UP=0→UP= n

If S(n)=0 and S(n-1)>0→LP= n-1,L=Segment(UP,LP) , UP=0

Where x is an image with MxN dimension. $m \in M, n \in N$, S is summation of each row in the image, while UP, LP and L are upper point, lower point and segmented line respectively.

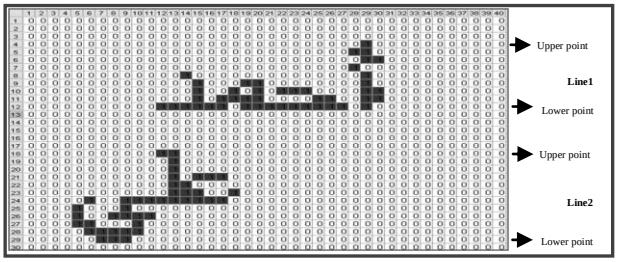

Figure.3. Line segmentation method

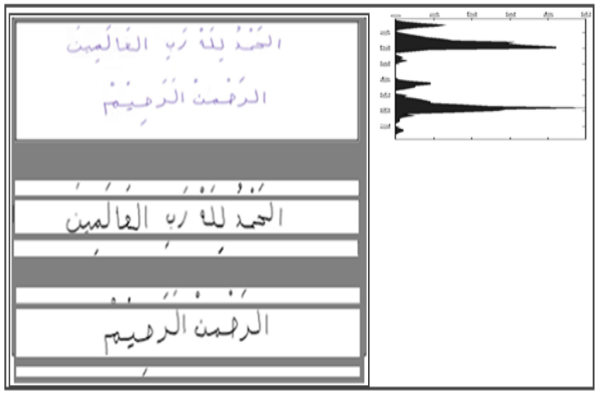

Figure 4.Line segmentation for small paragraph

B. Object segmentation

Component Labeling is simplest way to separate each object from other objects; different values are given for connected objects which is leads to sub-words isolation. Two types of connection among indexes (pixels): eight neighbors or four neighbors. In this work we use eight neighbors connectivity to cover all possible way of connection among pixels. Figure5 and 6 examples for object segmentation.

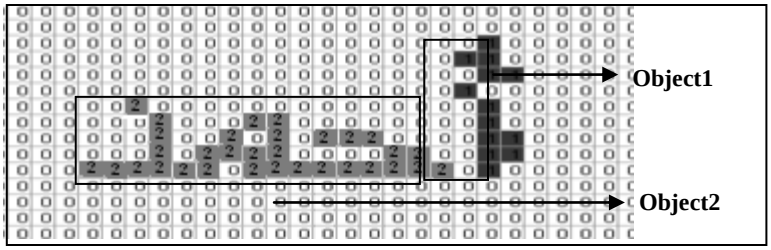

Figure 5.Component labelling for two object

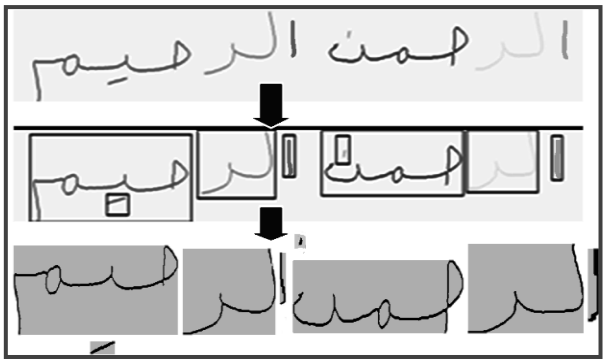

Figure 6. Object segmentation

C. Shape Segmentation

Shape segmentation is the most complex step in segmentation, in pervious steps we tried to reach to the simple way that can make shape segmentation easier, our idea in shape segmentation depends on finding segmentation points among characters, from these points we can separate each character, but also there are branch points in the same character, we cannot consider each branch point is segmentation point, so we need criteria to distinguish between branch points and segmentation points. Extract the segmentation points from branch points, if we can recognize to these point we can classify each character alone.

First step is finding the baseline; baseline here is a row which has maximum density in the object. Then searching for each branch point, branch points are points that branching from baseline, figure7 shows baseline and branch points.

Now, we will collect information for each branch point: position, high, width, max point in raw, min point in raw, max point in column and min point in column. Next step is checking if these branch points can be segmentation point or not. Four threshold values are proposed, one to classify between single shape and multiple shapes or object, and other three threshold values to detect the properties of segmentation point.

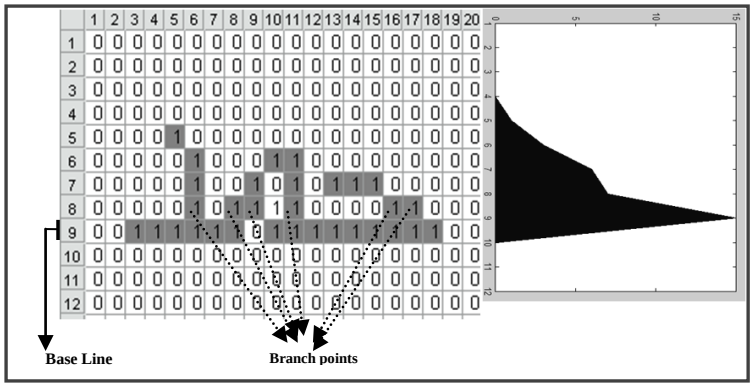

Figure7. Baseline and branch points detection

T1: maximum size of single shape

T2: minimum high of segmentation point

T3: minimum distance before and after circle (like meem character)

T4: minimum distance between two segmentation points

First threshold used to detect if this sub-word is single shape (character) or multiple shapes.

if >T1 → Single shape

Else → Multiple shapes

T2 used to separate segmentation points from branch points, if the high of branch point (H) equal or more than T2 then this point can be segmentation point.

If T2 → can be segmentation point

Figure 8.a illustrates T1 and figure8.b illustrates T2. After that, we need to detect circle in image, it is very important because finding any circle in image means new shape whatever in baseline or not, also most circle characters like "ha'a" have two segmentation points, T3 is used to remove any segmentation point around circle if the distance between them less than T3 and put new segmentation point, it useful in some situations when T2 failed to remove branch point with circle character like "sad" character, see figure 8.c.

If T3 → can be segmentation point

Else → Remove it, Add segmentation point after Circle

In some situations two segmentation points are very closely, T4 determine the minimum distance between characters or segmentation points, if the distance between them is less than T4 one of them well be deleted according to the position and high but if they are related to the same shape , the left segment point is deleted, figure8.d explains that.

If T4 → Leave them

Else → Remove one of them (left one),

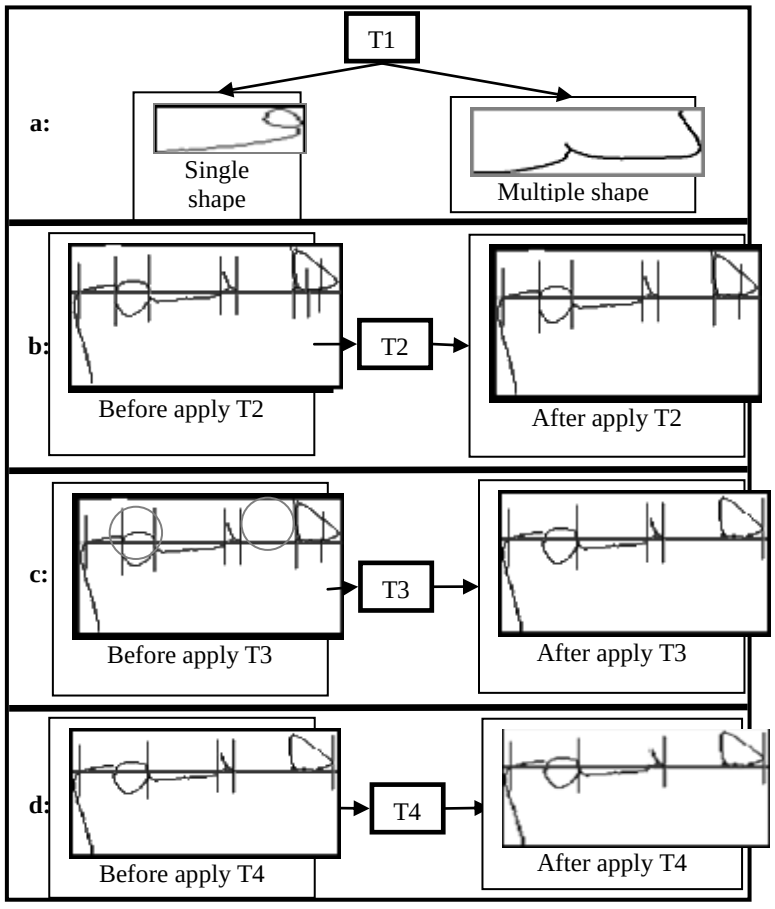

Figure 8. Applying threshold values to get segmentation points

times in some write maybe distance between two segment points for same shape is more than T4, for there any segment point connected with another segment point, the segment after shape is deleted, figure9.a explains that.

If → Leave them

Else → Remove P(i - 1),

We focus on make segmentation before the shape, but first shape not has before segment point so we must ignore first segment point, see figure9.b. Some shapes have segmentation point like 'Meem' character in the end of word, but not for new character, so any segmentation point closed to the end of sub word, it will be deleted, figure9.c illustrates that.After detect segmentation points, we start from left to right (from end to begin) and forward quarter the distance between current segment point and next segment point (from left to right) and make segmentation by cut or add zeroes in this point (before and after point in column).

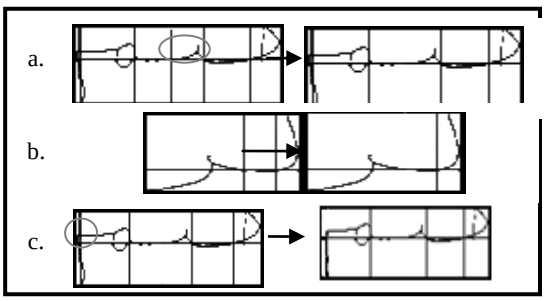

Figure 9. Remove some incorrect segmentation points

IV. Experimental Results

A. Database

In order to test our model, We applied it on three types of database; our local database, 'Hamad and Zitar' database (Al Hamad and Abu Zitar 2010) and INFINT database (INFINT database). Local database contains different texts from several writers with movements and multi-lines. Hamad and Zitar database consist of texts from ten writers with multi-lines but without movements. While IFN/ENITdatabase contains samples with single line and movements, the characteristic of IFN/ENITdatabase is containing many bad samples (sometimes cannot be read by human).

Number of words in local database is 400 .in Hamad and Zitar database we used 500 from their database. IFN/ENITdatabase Divided into 4 sets: 'a', 'b', 'c' and 'd' , we used samples from set 'a' about 300 words. The total number of words in our database reaches to 1200.Table1 illustrates that.

TABLE 1: CHARACTERISTIC OF DATABASES

Database	Number of words	Movements	Multi-lines
Local database	400	Yes	Yes
Hamad and Zitar database	500	No	Yes
IFN/ENITdatabase	300	Yes	No

B. Line Segmentation Result

In this stage each line must be separated, our algorithm for line segmentation is very simple and fast. The result of it was very good as it is shown in table2, where IFN/ENITdatabase contains single lines, so line segmentation is perfect. In local database correct line segmentation ratio reaches to 97%, while Hamad and Zitar database reaches to 95%, the average of line segmentation is . Anyway in some situations, it has a weak point when there is overlapping between lines, it will not be able to separate them correctly and maybe it gives multiple lines as a single line according to the overlapping, figure8.a shows incorrect line segmentation .

C. Object Segmentation Result

The objective of this stage is reaching to each object (sub-word) in the line, we used component labeling to recognize each object, that is good for connected object. In normal cases, each object is not connect with other objects or divided to more than one region, but because we try to cover all situations. In few situations, when the data is too bad some characters disconnected to two objects or more (must be in one object) in this case we can't get a correct object segmentation, but this case is too rare. Figure10.b illustrates it. From table2, correct object segmentation ratio in IFN/ENITdatabase was 94%, local database gave 98% and 'Hamad and Zitar'

database gave 97%. The average of correct object segmentation was 96.3%.

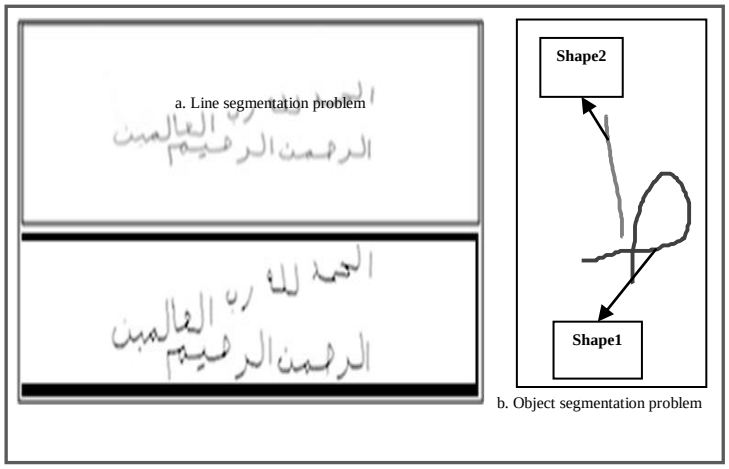

Figure 10. Incorrect segmentations

D. Shape Segmentation Result

This stage is the most important and hardest one, where we have to extract shapes or characters from each object. We proposed four threshold values to verify on classification points. The result was very efficient , it can treat some cases of overlaps between shapes, also it can determined some segmentation points in bad samples, in addition it can deal with circle shapes. Results of shape segmentation in IFN/ENITdatabase, Local database and 'Hamad and Zitar' database were 84%, 94% and 89% respectively. The average of shape segmentation was 89%.

Although, the result of shape segmentation was good, but it still has some weak points such as it depends on baseline algorithm, so if we have a problem in determine base line, then this will reflect on shape segmentations and give incorrect shapes or characters, also ithas a difficult to deal with some shapes that have high overlapping with other shapes.

Table 2:The result of proposed model

Database	**Line segmentation**	**Object segmentation**	**Shape segmentation**	**Final result of segmentation**
IFN/ENITdatabase	100%	94%	84%	78%
Local Database	97%	98%	94%	89%
Hamad and Zitar database	95%	97%	89%	81%
Average	97.3%	96.3%	89%	82.7%

Final result of segmentation was computed by summing error (missed segments) ratios of three stages. So in IFN/ENITdatabase, line segmentation error ratio was 0%.While object segmentation error ratio and shape segmentation error ratio were 6% and 16%. The total of error ratios is 22% and the result of correct line segmentation is 78%. Same computation in Local database and 'Hamad and Zitar' database, results of them are 11% and 19% for error ratios, and 89% and 81% for correct ratios respectively.

V. CONCLUSION

In this paper, we have studied the problem in Arabic hand written segmentation. We proposed a new model that has high ability to deal with characters in different situations (connected, isolated, in the beginning , in the middle, in the end), three stages were applied , first stages is line segmentation to separate each line, object segmentation to separate each word or sub-word and shape segmentation to separate each shape or character. The experimental results show the performance of it was very well and was able to perform this task with minimal error, the error for the missed segmentation criterion was favourable being within approximately 0.22%. Another important thing, our model has ability to classify character according to position of it (isolated, begin of word, middle of word, end of word), this classify is too important for text recognition.

For feature work, improving line segmentation and make it able to detect line with high curving is very important.Also, improving the ability of baseline detection even if it is not straight is important issue. Finally, working on more samples and more different writers.

ACKNOWLEDGMENT

This work was made possible by a grant from Imam Muhammad Ibn Saud Islamic University.

REFERENCES

[1] Al Hamad, H. A. and Abu Zitar, R. 2010. Development of an efficient neural-based segmentation technique for Arabic handwriting recognition.Pattern Recognition 43: 2773-2798.

[2] Ball, G. R., Srihari, S. N. and Srinivasan, H. 2006. Segmentation-Based And Segmentation-Free Methods for Spotting Handwritten Arabic Words. in Guy, L. (ed.), Tenth International Workshop on Frontiers in Handwriting Recognition: Suvisoft.

[3] Casey, R. G. and Lecolinet, E. 1996. A survey of methods and strategies in character segmentation.Pattern Analysis and Machine Intelligence, IEEE Transactions on 18: 690-706.

[4] Impedovo, S., Ottaviano, L. and Occhinegro, S. 1991. Optical character recognition-a survey.INT. J. PATTERN RECOG. ARTIF. INTELL,5: 1-24.

[5] Jiang, X., Petkov, N., Alamri, H., He, C. and Suen, C. 2009. A New Approach for Segmentation and Recognition of Arabic Handwritten Touching Numeral Pairs.Computer Analysis of Images and Patterns, 165-172. Springer Berlin / Heidelberg.

[6] Khatatneh, K. 2006. Probabilistic Artificial Neural Network for Recognizing the Arabic. Hand Written Characters. Citeseer.

[7] Kumar, J., Abd-Almageed, W., Kang, L. and Doermann, D. 2010. Handwritten Arabic text line segmentation using affinity propagation.Proceedings of the 9th IAPR International Workshop on Document Analysis Systems, Boston, Massachusetts: ACM.

[8] Lorigo, L. M. and Govindaraju, V. 2006. Offline Arabic handwriting recognition: a survey. Pattern Analysis and Machine Intelligence, IEEE Transactions on 28: 712-724.

[9] Nawaz, S. N., Sarfraz, M., Zidouri, A. and Al-Khatib, W. G. 2003. An approach to offline Arabic character recognition using neural networks. Electronics, Circuits and Systems, 2003. ICECS 2003. Proceedings of the 2003 10th IEEE International Conference on, 1328-1331 Vol.1323.

[10] Suen, C. Y., Legault, R., Nadal, C., Cheriet, M. and Lam, L. 1993. Building a new generation of handwriting recognition systems. Pattern Recognition Letters 14: 303-315.

[11] IFN/ENIT-database – DATABASE OF HANDWRITTEN ARABIC WORDS – http://www.ifnenit.com/

AUTHORS PROFILE

Said Elaiwat

School of Computer Science & Software Engineering (CSSE)

The University of Western Australia (UWA) , Australia

35Stirling Highway CRAWLEY WA 6009.

Marwan Abu-zanona is a lecturer in the Imam Muhammad Ibn Saud Islamic University. He received a Ph.D. in Artificial Intelligence from the Faculty of Computer Information Systems, University of Banking and Financial Sciences. His research interest in neural networks, artificial intelligence and software engineering areas. He has a wealth of expertise gained from his work experiences in Jordan, ranging from web development to network administration

Farah Al-Zawaideh is the Chairman of Computer Information System in Irbid National University from 2009 until now. He received a Ph.D. in Knowledge based systems from the Faculty of Computer Information Systems, University of Banking and Financial Sciences. His research interest in genetic algorithms, E-learning and software engineering areas. He has a wealth of expertise gained from his work experiences in Jordan, ranging from web development to network administration.