Using Heuristics Based Approach for Segmentation and Recognition of Printed Arabic Characters

Ihab Zaqout

Department of Information Technology
Faculty of Engineering and Information Technology
Al-Azhar University – Gaza, Palestine
ihabzakout@yahoo.com

Abstract

In this study, we propose a flexible template-matching algorithm for word segmentation, and structural analysis of features extraction is used for character recognition in the printed Arabic text. The input text image is preprocessed by the binarization and then by morphological operations. A vector quantization of the thinned image (VQTM) is created based on the idea of a freeman chain code tracking method. In the segmentation process, 113 character templates are compared for partially/completely existence in the VQTM. A non-linear filter is applied on the segmented regions to extract the termination and bifurcation features. The spatial distribution of the extracted features and other statistical characteristics are analyzed for the verification of recognition. Experimental results show that the overall recognition rate of the three fonts: Arabic transparent, simplified Arabic and traditional Arabic is 98.63%.

Keywords: *Template matching, Word segmentation, Features extraction, Character recognition, Vector quantization*

1. Introduction

Arabic language is ranked the fifth among common languages globally, where the number of speakers is about 7% of the world's population. A little research is done in the field of printed or handwritten Arabic Optical Character Recognition (AOCR) compared to research done in English Optical Character Recognition (EOCR). The Arabic character recognition is one of the most challenging tasks and exciting areas of research in Optical Character Recognition (OCR). Despite the growing interest in the work of researchers in the recognition of Arabic texts which started at the beginning of the eighties [2], until now there is no a comprehensive algorithm, due to the difficulty of syntax, semantics, and writing skills of Arabic characters.

Hidden Markov Model (HHM) technique is used to recognize the printed Arabic characters introduced by [4, 9]. Muhtaseb, *et al.,* [4] proposed a hierarchical sliding window technique is proposed to generate 16 features for each sliding window, while Hassin, *et al.,* [9] introduced an entirely transformation for each character/word into a feature vector and a vector quantization is used to transform the word skeleton into a sequence of symbols. Zidouri [5] introduced a sub-word segmentation and recognition. He used a three layered radial basis function network for training and 8-neighbor connected component algorithm is applied for segmentation.

A principal component analysis (PCA) is implemented on 200 binary images of size 32x32, features are extracted from the four edges and back propagation neural network (BPNN) is implemented for the Arabic character recognition [6]. A recognition-based segmentation for online handwritten Arabic text is proposed by Potrus and Ngah [10]. Their system utilizes the dominant point detection with harmony search (HS) to find the best combination of segmentation points. The best segmentation points are located by using a feedback system between HS and total word score. Bushofa and Spann [11] introduce Line segments and polygons features that are extracted from skeletonized strokes to classify the characters in two stages, and the decision tree is used to classify characters. If any character is rejected, it is compared with a set of predefined templates according to a minimum distance criterion.

An adaptive segmentation algorithm is proposed by [12] and the main feature used is "All Arabic characters start at a T-junction with the baseline". Other features deal with the beginning character in a word and the medial axis of the text is used to extract the structural representation. A comprehensive study of Arabic character recognition (ACR) techniques is presented [7 - 8].

Arabic text is distinguished from other languages because of the following characteristics [1]:

1. Isolated Arabic alphabet consist of 28 characters (ا، ب، ت، ... ي) as shown in Figure 1, which increases according to the position of the letter in the word, bringing the number to 113 as shown in Table 1. For example, the letter م (Meem) is written in four forms according to its position in the word (مـ is in the beginning of the word, ـمـ in the middle of a word, ـم at the end of the word, the letter م is isolated).
2. Printed or handwritten Arabic text is cursive, written from right to left and letters connected to each other on the baseline.
3. Some of Arabic characters are above the baseline, for example, ك (Kaf), ـتـ (Ta) and some of which is below the baseline, for example, ق (Qaf), ـر (Ra). The size depends on the position of a character in the word.

Arabic characters can be distinguished from each other by the number of components of the character, Some consist of one-part such as ر (Ra), مـ (Meem), و (Waw), etc., two-part such as ب (Ba), ك (Kaf), ن (Noon), etc., three-part such as ق (Qaf), ت (Ta), ي (Ya), and four-part such as ث (Tha), ش (Sheen). In addition, there are some ligatures such character لا (Lamalef).

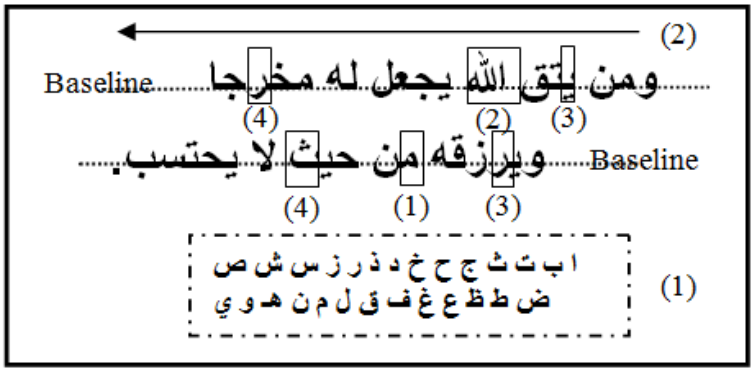

Figure 1. Characteristics of Arabic Text

Table 1. Arabic Letters

No.	Character	Start	Middle	End	Isolated	No. of shapes
1	Alef	-	ا، آ، أ	ى	ا، أ، إ، آ، ى	9
2	Ba	بـ	بـ	ب	ب	4
3	Ta	تـ	تـ	ة, ت	ة، ت	6
4	Tha	ثـ	ثـ	ث	ث	4
5	Geem	جـ	جـ	ج	ج	4
6	Ha'a	حـ	حـ	ح	ح	4
7	Kh'a	خـ	حـ	خ	خ	4
8	Dal	-	-	د	د	2
9	Dhal	-	-	ذ	ذ	2
10	Ra	-	-	ر	ر	2
11	Zeen	-	-	ز	ز	2
12	Seen	سـ	سـ	س	س	4
13	Sheen	شـ	شـ	ش	ش	4
14	Sad	صـ	صـ	ص	ص	4
15	Dad	ضـ	ضـ	ض	ض	4
16	Tah	طـ	طـ	ط	ط	4
17	Dhad	ظـ	ظـ	ظ	ظ	4
18	Aen	عـ	عـ	ع	ع	4
19	Ghen	غـ	غـ	غ	غ	4
20	Fa	فـ	فـ	ف	ف	4
21	Qaf	قـ	قـ	ق	ق	4
22	Kaf	كـ	كـ	ك	ك	4
23	Lam	لـ، لا، لأ، لإ، لآ	لـ	ل, لا، لأ، لإ، لآ	ل, لا، لأ، لإ، لآ	8
24	Meem	مـ	مـ	م	م	4
25	Noon	نـ	نـ	ن	ن	4
26	Ha	هـ	هـ	ه	ه	4
27	Waw	-	-	و	و	2
28	Ya	يـ	يـ	ي	ي	4
Total number of shapes (templates)						113

The remainder of this paper is organized as follows. The system framework is presented in Section 2. The word segmentation process is described in Section 3 and the recognition of Arabic characters is presented in Section 4. In Section 5, experimental results to test

performances of the proposed approach are given. Finally our conclusion and future work is given in Section 6.

2. System Framework

As shown in the Figure 2, the system framework diagram consists of three main tasks: preprocessing, segmentation, and recognition.

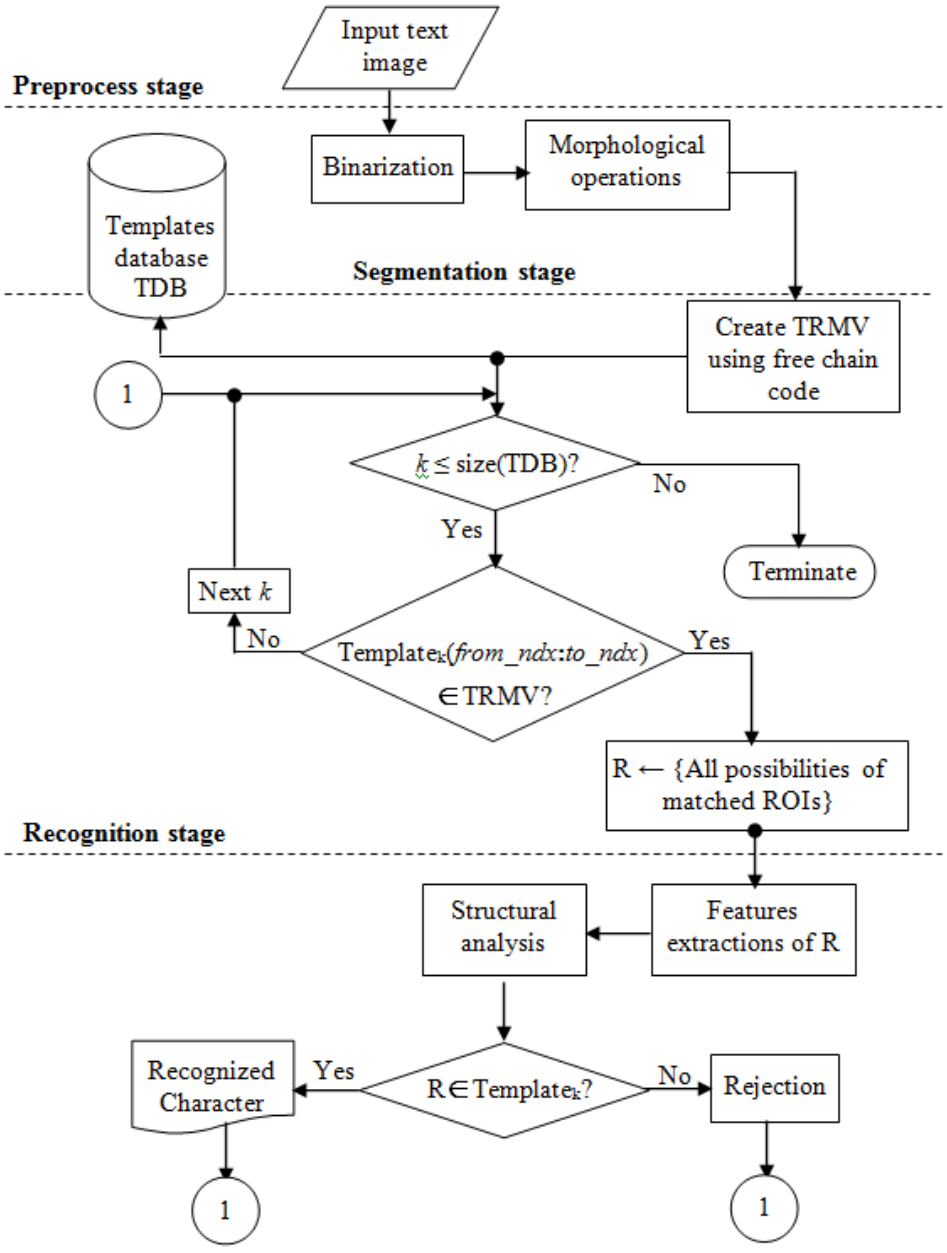

Figure 2. System Framework

3. Word Segmentation

A preprocess stage is always requested as a prerequisite step in digital image processing problems. In this research the preprocess stage starts with the conversion of the input image into a binary image.

Figure 3. Preprocessing Steps: (a) input, (b) binary, (c) spurred, and (d) thinned

To remove simple growths, the spurring morphological operation is applied on the binarized image, and then a thinning morphological operation is applied to produce a skeletonized text image. All preprocessing steps are resulted in Figure 3. The word or text segmentation is based on the creation of the vector quantization vector thinned image (VQTM) and then, a comparison between the created VQTM and the 113 character templates database.

3.1 Vector Quantization

The freeman chain code tracking method [3] and its work mechanism is depicted in Figure 4. It is implemented on the thinned image to produce the vector quantized thinned image (VQTM).

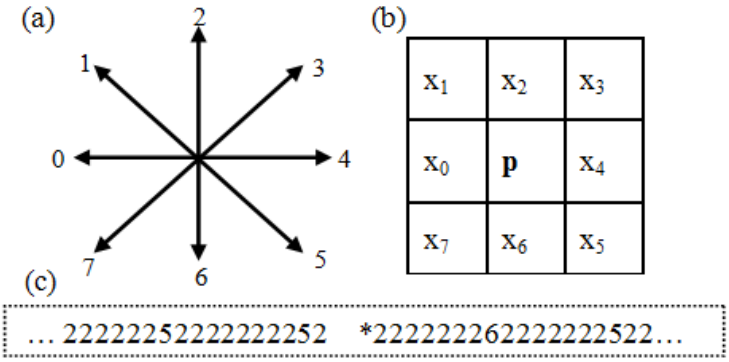

Figure 4. Freeman chain code, template, and a snapshot of VQTM

Only north-east (3 or X3), east (4 or X4), south-east (5 or X5), and south (6 or X6) directions are used in terms of neighborhood pixels to the center pixel p. For example, the VQTM of the text "صدق الله العظيم" consists of 592 white pixels (in terms of pixels directions 3, 4, 5 and 6) including the spaces between words and sub-words and its representation is partially depicted in Figure 4(c).

3.2. Character Matching

All character templates are compared against the VQTM to check their existence. Because of the font size of the input text affects the sequence of the pixels directions and their size in the VQTM, we are considered a flexible size of template in our approach starting from a small size and is gradually incremented until the full size of the template is reached. The algorithm of character matching process is illustrated in Figure 5.

```
// k: the total number of character templates
1.  for all templates k
2.      n = 0, i = 1, j = 4, c = 0, temp = Φ.
3.      while (j ≤ length(template(k)))
4.          while (template(k)[i:j] ∈ VQTM)
5.              j = j + 1;
6.          end while
7.          if (template(k)[i:j] ∉ temp) & (template(k)[i:j] ∈ VQTM)
8.              c = c + 1;
9.              temp(c) = template(k)[i:j];
10.         end if
11.         i = i + 1; j = i + 3;
12.     end while
13.     ndx = find(temp(c) ∈ VQTM) ± 5;
14.     n = n + 1;
15.     R_n = thinned_image(ndx);
16. end for
```

Figure 5. Character Matching Algorithm

4. Character Recognition

For each character template, there exist segmented regions arranged in set *R*, which are needed to be analyzed to check their structural characteristics. To accept or reject each segmented region in *R*, a study of the extracted features and their statistical arrangements are required.

To do so, a non-linear filter of window size 3x3 is applied to extract termination and bifurcation features as shown in Figure 6. Those features are the coordinates and orientations of termination and bifurcation points. Arabic characters are distinguished from each other by their statistical measurements, for example, number of regions, number of holes, number of termination and bifurcation points, spatial distribution of the extracted points, orientations, and minor-to-major axis lengths ratio, etc. Therefore, each segmented region in *R* is correctly classified if their structural properties match each other, otherwise, rejection operation for their non-existence (unclassified). In traditional Arabic font, some characters overlap each other and have completely

different shapes, for example, character ـيـ and ـجـ in word يجعل, ـمـ and ـخـ in word مخرجا, and ـيـ and ـحـ in word يحتسب. Their extracted features in terms of termination and bifurcation points are analyzed.

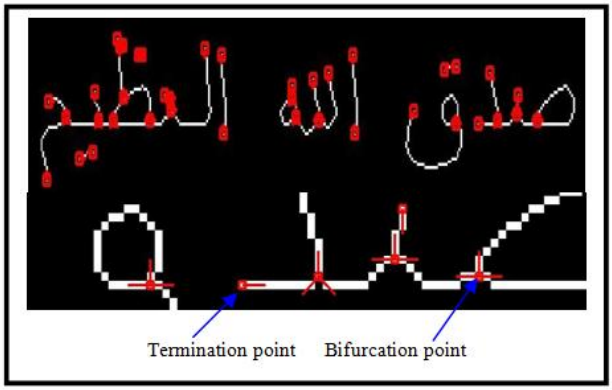

Figure 6. Features Extraction

5. Experimental Results

The proposed hybrid approach is implemented on MATLAB 6.5.1 release 13, a 2.40 GHz P4 processor and Windows XP operating system. It is tested on 31 printed Arabic text images which constitute 202 words and 806 characters for each single font type (31 input texts written in three fonts). Table 2 summarizes the performance of our approach using the three font types: Arabic Transparent (AT), Simplified Arabic (SA), and Traditional Arabic (TA). The CC indicates characters are correctly classified, IC incorrectly classified characters, and UC unclassified characters. For example, 38 characters constitute the traditional Arabic text "ومن يتق الله يجعل له مخرجا ويرزقه من حيث لا يحتسب", means that, 37 out of 38 characters are correctly classified (CC% = 97.37), one character is incorrectly classified (IC% = 2.63), and 2 characters are missed or unclassified (UC% = 5.26).

Table 2. The Performance of the Proposed Approach

Text #	Font Type	Text	CC %	IC %	UC %
1	AT	بسم الله الرحمن الرحيم	100	0	0
2	TA	ومن يتق الله يجعل له مخرجا ويرزقه من حيث لايحتسب.	97.37	2.63	5.26
…	…	…	…	…	…
…	…	…	…	…	…
…	…	…	…	…	…
29	TA	الكلمة الطيبة صدقة	100	0.06	0
30	AT	بل هم أحياء عند ربهم يرزقون	100	0.05	0
31	TA	صدق الله العظيم	100	0	0

A sample of our experimental results is shown in Figure 7. As shown in the figure, each recognized character is surrounded by a bounding box and its identification is written below it. Some characters may have several bounding boxes under one identity because of their possibility appearances in the word; for example, in the first word "بسم" consists of three characters: "باء، سين، ميم", the last character may be written as a single "م" or with extension "maddah ـ" as "ـم". Similarly, the third character "راء" in the third and fourth words "الرحمن" and "الرحيم", respectively, are recognized correctly into two shapes "ر" and "ـر", and so on.

Additionally, our approach is capable of recognizing the overlapped characters, as for example, character "م" is overlapping character "ح" in the third word "الرحمن" and character "ج" is overlapping character "ي" in word "يجعل", etc.

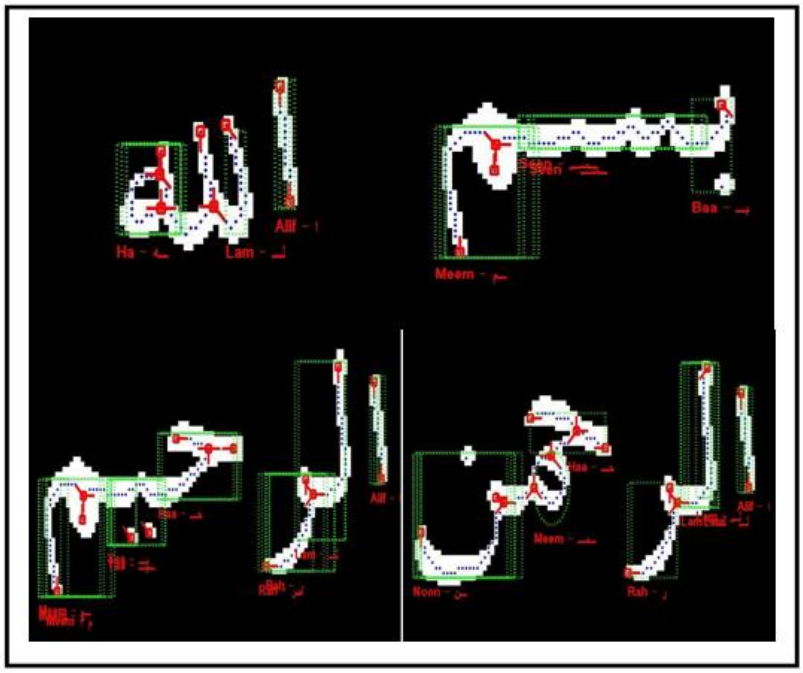

Figure 7. Character recognition in the traditional Arabic text "بسم الله الرحمن الرحيم"

The overall recognition rate of the three fonts: Arabic transparent, simplified Arabic and traditional Arabic is 98.63%.

6. Conclusion and Future Work

In this study, a flexible template matching algorithm is proposed for word segmentation, and features are extracted for character recognition as well, in the printed Arabic text. The input text image is preprocessed by the binarization and then by morphological operations. A vector quantization of the thinned image (VQTM) is created based on the idea of a freeman chain code tracking method. In the segmentation process, 113 character templates are compared for partially/completely existence in the VQTM, where a non-linear filter is applied

on the segmented regions to extract the termination and bifurcation features. The spatial distribution of the extracted features and other statistical characteristics are analyzed for the verification of recognition.

The proposed approach is tested on 31 printed Arabic text images which constitute 202 words and 806 characters. Despite the font size, our experimental results reports that the overall recognition rate of the three font types: Arabic transparent, simplified Arabic and traditional Arabic is 98.63%. Finally, applying the proposed approach to handwritten Arabic text will be a future work.

References

[1] I. Zaqout, "Printed Arabic Characters Classification Using a Statistical Approach", International Journal of Computers and Technology, vol. 3, no. 1, **(2012)**, pp. 1-5.
[2] B. Al-badr and S. Mahmoud, "Survey Bibliography of Arabic Text Recognition", Signal Processing, vol. 4, **(1995)**, pp. 49-77.
[3] H. Freeman, "On the Encoding of Arbitrary Geometric Configurations", IRE Trans. Pattern Analysis and Machine Intelligence Electronics and Computers, vol. 10, no. 2, **(1961)**, pp. 260–268.
[4] H. Muhtaseb, S. Mahmoud and R. Qahwaji, "Recognition of Off-line Printed Arabic Text Using Hidden Markov Models", Signal Processing, vol. 88, no. 12, **(2008)**, pp. 2902-2912.
[5] A. Zidouri, "On Multiple Typeface Arabic Script Recognition", Research Engineering and Technology, vol. 2, no. 5, **(2010)**, pp. 428 - 435.
[6] L. Zheng, "Recognition for Arabic Character Based on Edge and BPNN", Proc. of the World Congress on Engineering and Computer Science, vol. 2173, no. 1, **(2008)**, pp. 207-209.
[7] P. Ahmed and Y. Al-Ohali, "Arabic Character Recognition: Progress and Challenges", Journal of King Saud University, Computer & Information Sciences, vol. 12, **(2000)**, pp. 85-116.
[8] A. M. Al-Shantawi, F. H. Al-Zawaideh, S. Al-Salaimeh and K. Omar, "Off-line Arabic Text Recognition – An Overview", World of Computer Science & Information Technology Journal, vol. 1, no. 5, **(2011)**, pp. 184-192.
[9] A. Hassin, X. Tang, J. Liu and W. Zhao, "Printed Arabic character recognition using HMM", Journal of Computer Science & Technology, vol. 19, no. 4, **(2004)**, pp. 538-543.
[10] M. Y. Potrus and U. K. Ngah, "A Harmony Search Algorithm for Recognition Based Segmentation of Online Arabic Text", Proc. of International Conference on Engineering and Information Technology "ICEIT2012", **(2012)**, pp. 205-210.
[11] B. M. Bushofa and M. Spann, "Segmentation and Recognition of Printed Arabic Characters", BMVC '95 Proceedings of the 6th British conference on Machine vision, vol. 2, **(1995)**, pp. 543-552.
[12] G. Mustafa, "An Adaptive Algorithm for the Automatic Segmentation of Printed Arabic Text", 17th National Computer Conference, **(2004)**, pp. 437-444.

Authors

Ihab Zaqout received Ph.D. degree from Malaya University, Kuala Lumpur, Malaysia, in 2006 in computer science. He is currently an academic member of the faculty of Engineering and Information Technology, Al-Azhar University - Gaza, Palestine. His research interests include image processing, computer vision, and data mining.