PRINTED ARABIC CHARACTERS CLASSIFICATION USING A STATISTICAL APPROACH

Ihab Zaqout
Dept. of Information Technology
Faculty of Engineering & Information Technology
Al-Azhar University – Gaza
Palestine

ABSTRACT

In this paper, we propose simple classifiers for printed Arabic characters based on statistical analysis. 109 printed Arabic character images are created for each one of transparent, simplified and traditional Arabic fonts. Images are preprocessed by the binarization and followed by sequence of morphological operations. A non-linear filter is applied on the thinned ridge map to extract termination and bifurcation features. The thinned ridge map vectors (TRMVs) are created using a freeman chain code template. The spatial distribution and statistical properties of the extracted features are calculated.

Keywords

Freeman chain coding; character recognition; feature extraction; classification.

1. INTRODUCTION

This paper aims to introduce 109 classifiers including thinned ridge map vectors (TRMVs) for each one of transparent, simplified and traditional Arabic fonts. The work on TRMVs is left as a future work for Arabic text classification.The Arabic language occupies the fifth place among the languages most commonly used worldwide and the attainment of the proportion of Arabic speakers around 7% of the population of the world. The estimated number of Arabic speakers around the world is about 437 million people, including 85 million active users on the Internet. Published research on identifying the Arabic letters, whether printed or handwritten is very few compared to the published research on English character recognition. It 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 identification of Arabic texts which starts at the beginning of the eighties [1], until now there is no a comprehensive algorithm, due to the difficulty of writing rules of Arabic characters.

Zidouri [2] proposed a sub- word segmentation and recognition. A three layered radial basis function network for training and 8-neighbor connected component algorithm is applied for segmentation. In recognition, they use a PCA on 200 binary images of 32x32. A main line algorithm is proposed by Al-Jarrah *et al*. [3] for segmentation to tokenize the text and generates a set of 33 different tokens that represent the 28 Arabic characters and their different shapes and variation. A forward neural network is used to recognize the segmented characters. A recognition algorithm based on feature extraction and using a Fuzzy ART Neural Network is proposed by Almohri *et al*. [4]. Sarhan and Helalat [5] proposed a statistical analysis for feature extraction and ANN for recognition. The ANN is trained using the least Mean Squares (LMS) algorithm. Each typed Arabic letter is represented by a matrix of binary numbers that are used as input to a simple feature extraction system whose output, in addition to the input matrix, are fed to an ANN. Zheng [6] proposed feature extracted from the four edges and BPNN is implemented for recognition.Batawi and Abulnaja [7] proposed an optical character recognition voting (AOCRV) scheme based on the N-version programming (NVP) technique which is applied on 35 printed text samples. A generalized Hough transform is applied to recognize Arabic printed characters in different shapes is proposed by Sofien *et al*. [8]. It is tested on a set of 234,868 samples of Arabic characters in Arabic Transparent, Andalus and Traditional fonts. Hassin *et al*. [9] proposed a Hidden Markov model to recognize printed Arabic characters. Each character/word is entirely transformed into a feature vector and a vector quantization is used to transform the word skeleton into a sequence of symbols.

Arabic text is distinguished from other languages because of the following characteristics:
1,Arabic Alphabet consists of 28 characters (ا، ب، ت، ...، ي) as shown Fig. 1, which increases according to the position of the letter in the word, bringing the number to 109 as shown in Table 1. For example, the letter ش (sheen) is written in four forms according to its position in the word (if شـ is in the beginning of the word, ـشـ in the middle of a word, ـش at the end of the word, the letter ش is isolated).
2.Arabic text is cursive, whether printed or handwritten is written from right to left and letters connected to each other on the baseline.
3.Arabic characters differ in their standards, some of which is high for the baseline, some of which is lower than the baseline, for example, و (waw), ر (Ra), ز (zen). The size depends on the location of a character in the word.
4.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), ـم (meem), و (waw), etc., two-part such as ب (ba), ك (kaf), ن (noon), etc., three-part such as ق (qaf), ت (taa), ي(yaa), and four-part such as ث (thaa), ش (sheen). In addition, there are some ligatures such character لا (lamalef).

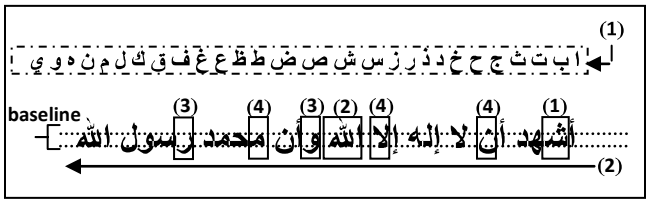

Fig 1: Characteristics of Arabic text

Fig.1 introduces a brief summary of the main characteristics of Arabic characters. This figure includes four main points; each point describes a fundamental individual feature.

Table 1. Arabic Letters

	Start	Middle	End	Isolated
Alef	-	ـآ ـا ـأ	ـى	إ أ ا آ ى
Ba	بـ	ـبـ	ـب	ب
Ta	تـ	ـتـ	ـة ـت	ة ت
Tha	ثـ	ـثـ	ـث	ث
Geem	جـ	ـجـ	ـج	ج
Ha'a	حـ	ـحـ	ـح	ح
Kha	خـ	ـخـ	ـخ	خ
Dal	-	-	ـد	د
Dhal	-	-	ـذ	ذ
Ra	-	-	ـر	ر
Zeen	-	-	ـز	ز
Seen	سـ	ـسـ	ـس	س
Sheen	شـ	ـشـ	ـش	ش
Sad	صـ	ـصـ	ـص	ص
Dad	ضـ	ـضـ	ـض	ض
Tah	طـ	ـطـ	ـط	ط
Dhad	ظـ	ـظـ	ـظ	ظ
Aen	عـ	ـعـ	ـع	ع
Ghen	غـ	ـغـ	ـغ	غ
Fa	فـ	ـفـ	ـف	ف
Ka'af	قـ	ـقـ	ـق	ق
Kaf	كـ	ـكـ	ـك	ك
Lam	لـ	ـلأ ـلا ـلآ ـلإ	ـل	ل
Meem	مـ	ـمـ	ـم	م
Noon	نـ	ـنـ	ـن	ن
Ha	هـ	ـهـ	ـه	ه
Waw	-	-	ـو	و
Ya	يـ	ـيـ	ـي	ي

The remainder of this paper is organized as follows. The system framework is presented in Sec. 2. The preprocessing is proposed in Sec. 3. In Sec. 4, the features extraction and statistical analysis are discussed. Experimental results are shown in Sec. 5 to demonstrate the reliability of our method. Finally our conclusion and future work is given in Sec 6.

2. SYSTEM FRAMEWORK

The following diagram as shown in Fig. 2 consists of three main tasks: preprocessing, features extraction and statistical analysis.

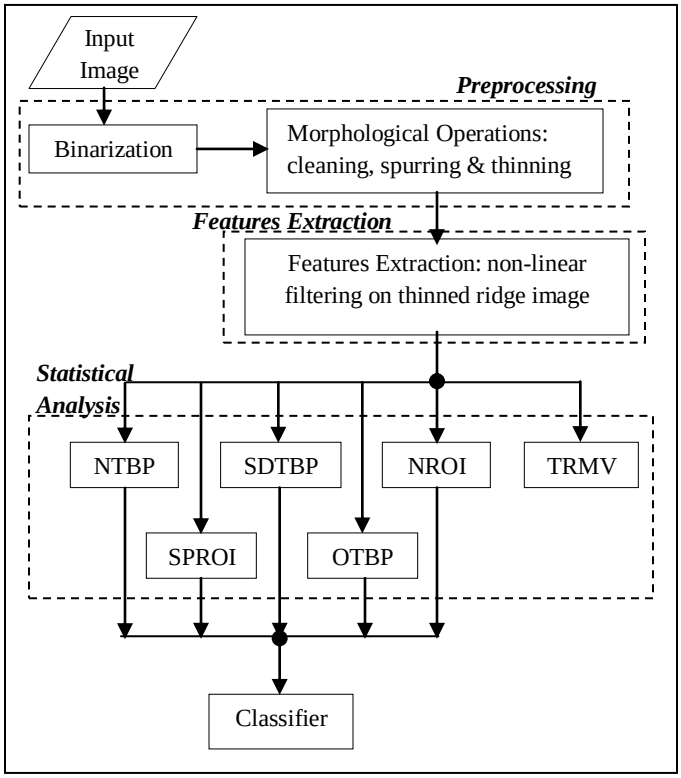

Fig 2: System Framework

The description of the above mentioned abbreviations in Fig. 2 are: NTBP: Number of termination & bifurcation points, SDTBP: Calculate the spatial distribution of termination & bifurcation points, NROI: Number of Region-of-interests, TRMV: Thinned ridge map vectors, SPROI: Statistical properties of Region-of-interests, and OTBP: Calculate orientation of termination and bifurcation points.

3. PREPROCESSING

In this stage, the input image is transformed into a binary image followed by sequence of morphological operations that are limited to cleaning, spurred and thinning. The overall preprocessing stage is implemented on transparent, simplified and traditional Arabic fonts and sample of results are shown in Fig. 3.

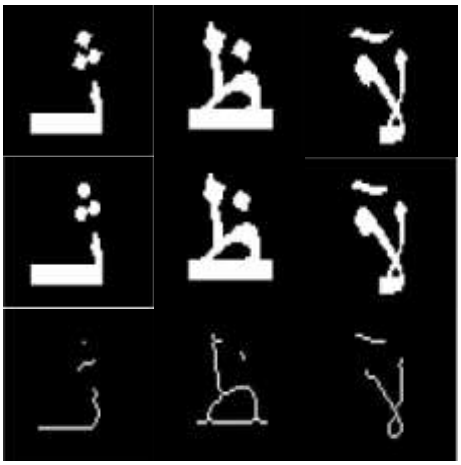

Fig. 3. Preprocessing steps. (a) Input image. (b) Binary Image. (c-e) Cleaning, Spurring, and Thinning operations

4. FEATURES EXTRACTION AND STATISTICAL ANALYSIS

A non-linear filter is applied on the thinned ridge map to compute the number of one-value of each 3-by-3 window. If the central is 1 and has only 1 one-value neighbor, then the central pixel is a termination. If the central is 1 and has 3 one-value neighbors, then the central pixel is a bifurcation. Otherwise the central pixel is a usual pixel. The orientation (***O***) for each single terminated or bifurcated pixel is calculated based on the following matrix:

$$O=\begin{bmatrix} 3\pi/4 & \pi/2 & \pi/4 \\ \pi & 0 & 0 \\ -3\pi/4 & -\pi/2 & -\pi/4 \end{bmatrix}$$

A non-linear filter is applied on the thinned ridge map to compute the number of one-value of each 3-by-3 window. If the central is 1 and has only 1 one-value neighbor, then the central pixel is a termination. If the central is 1 and has 3 one-value neighbors, then the central pixel is a bifurcation. Otherwise the central pixel is a usual pixel. The orientation (***O***) for each single terminated or bifurcated pixel is calculated based on the following matrix and its result is depicted as a sample in Fig. 4.

Each single alphabetic Arabic character has its own properties such as number of regions, number of holes, number of termination and bifurcation points, spatial distribution of termination and bifurcation points, orientation of termination and bifurcation points, and ROI minor-to-major axis lengths ratio.

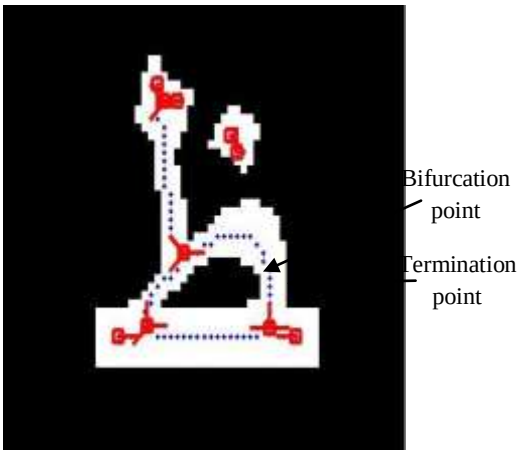

Fig 4: The implementation of non-linear filter on letter "Dhad" produces 6 termination and 4 bifurcation points

We introduce the freeman chain code tracking [10] to determine the TRMV as depicted in Fig. 5. North-east, east, south-east, and south directions are only used in calculation because of left-right and top-down image movements.

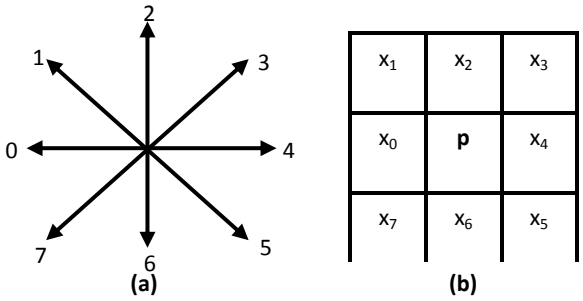

Fig 5. Freeman chain code and 3x3 template

5. EXPERIEMENTAL RESULTS

Our experiments are performed on 109 images for each single font by using MATLAB 6.5.1 release 13. The size of each letter image is 60×50 pixels. All results are obtained by using 2.40 GHz P4 processor under Windows XP. The processing time of each single letter is around 0.416 sec. The sample of our results of execution is depicted in Fig. 6. Each image shows its font type, letter name and its position in the word.

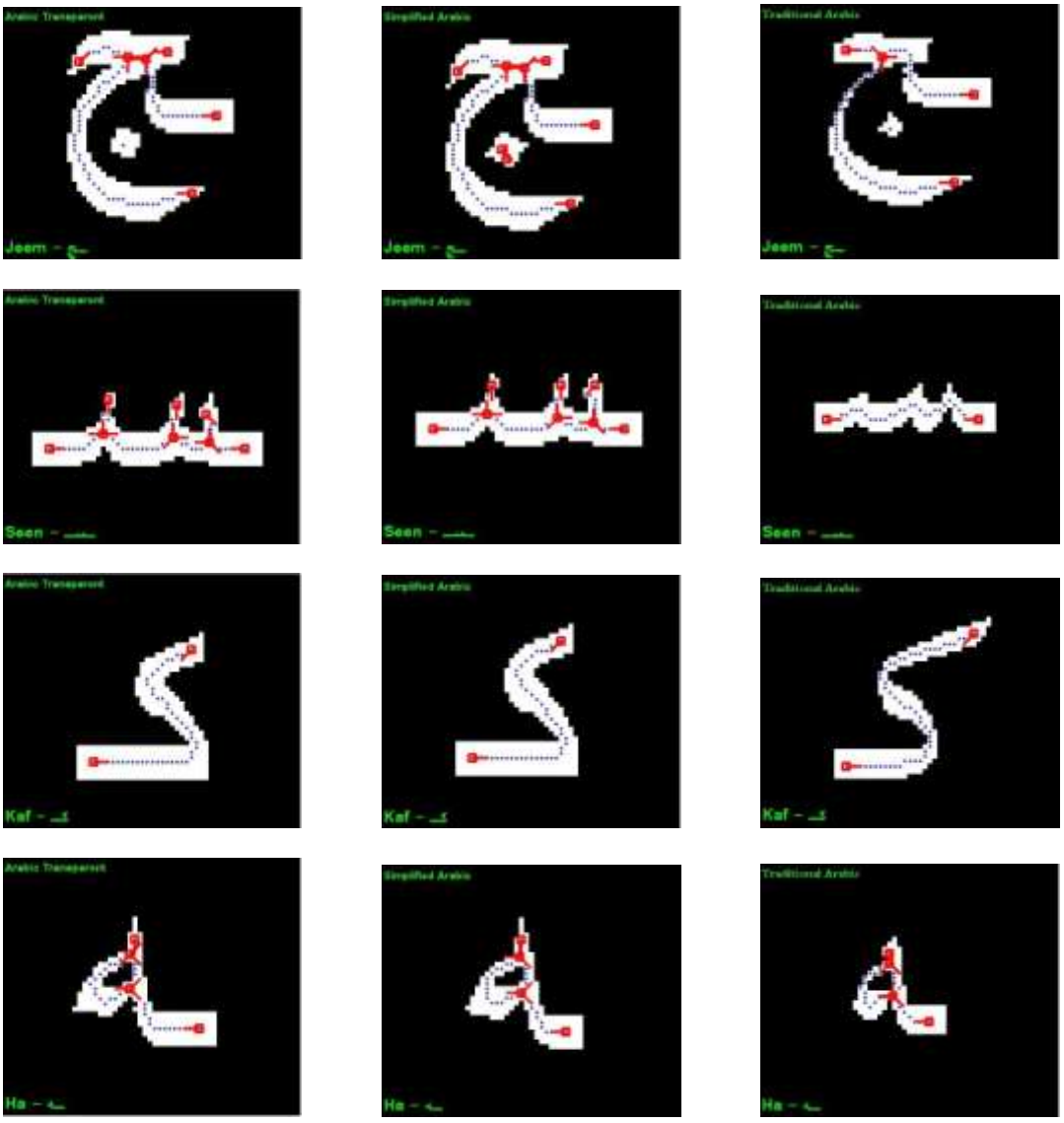

Fig 6: The implementation of our Statistical Approach

5. CONCLUSION & FUTURE WORK

In this paper, we have described a simple statistical approach for feature extraction and classification for printed Arabic character recognition. After preprocess of the input image, termination and bifurcation feature sets are extracted from the thinned letter image using the concept of the non-linear filter with window size 3x3. By using statistical concepts for analysis, the extracted termination and bifurcation features are used for classification. The overall performance is 100%.

In the future work, we will add more font types and use the created TRMVs in printed Arabic text segmentation.

REFERENCES

[1] Al-badr B., Mahmoud S. 1995. Survey Bibliography of Arabic Text Recognition. Signal Processing, 4: 49 – 77.

[2] Zidouri A. 2010. On Multiple Typeface Arabic Script Recognition. Research Engineering and Technology, 2(5):428 - 435.

[3] Al-Jarrah O., Al-Kiswany S., Al-Gharaibeh B., Fraiwan M. and Khasawaneh H. 2006. A New Algorithm for Arabic Optical Character Recognition. Proc. of the 5th WSEAS Int. Conf. on Artificial Intelligence, Knowledge Engineering and Data Bases, 211-224.

[4] Almohri H., Gray J. and Alnajjar H. 2008. A Real-time DSP_Based Optical Character Recognition System for Isolated Arabic Characters using the TI TMS320C6416T. Proc. of the IAJC-IJME International Conference.

[5] Sarhan A. and Al-Helalat O. 2007. Arabic Character Recognition using Artificial Neural Networks and

Statistical Analysis. Proc. of World Academy for Science, Engineering and Technology, 21: 32-36.

[6] Zheng L. 2008. Recognition for Arabic Character Based on edge and BPNN. Proc. of the World Congress on Engineering and Computer Science, 2173(1): 207-209.

[7] Batawi Y. and Abulnaja O. 2012. Accuracy Evaluation of Arabic Optical Character Recognition Voting Technique: Experimental Study. International Journal of Electrical & Computer Sciences (IJECS), 2(1):29 – 33.

[8] Touj S., Ben Amara N. and Amiri H. 2003. Generalized Hough Transform for Arabic Optical Character Recognition. Proc. of the 7th International Conference on Document Analysis and Recognition (ICDAR 2003), 2:1242-1246.

[9] Hassin A., Tang X., Liu J. and Zhao W. 2004. Printed Arabic Character Recognition using HMM. Journal of Computer Science & Technology, 19(4):538-543.

[10] Freeman H. 1961. On the Encoding of Arbitrary Geometric Configurations, IRE Trans. Pattern Analysis and Machine Intelligence Electronics and Computers. 10:260–268.