Isolated Multifont Arabic Character Recognition Using Fourier Descriptors

Shatha M. Noor
Department of Computer Science, College of Science, University of Al-Nahrain, Baghdad-Iraq.
E-mail: shatha_alhassany@yahoo.com.

Abstract

Optical Characters Recognition (OCR) has been an active subject of research since the early days of computers. Despite the age of the subject, it remains one of the most challenging and exciting areas of research in computer science. In recent years it has grown into a mature discipline, producing a huge body of work. Arabic character recognition has been one of the last major languages to receive attention. In this paper a simple and accurate method is proposed to recognize isolated Arabic characters using Fourier descriptors feature set and character's dots information represented by number of dots and their position. Eight commonly used font styles in different font sizes were used in the test, first each font style is tested separately and found the recognition ratio is excellent, then a combination of font styles were tested; and it was found that as more font styles used the recognition ratio decrease.

Keywords: Character Recognition, Binarization, Thinning, Segmentation, Chain Codes, Fourier Descriptor, Grand Class.

1.Introduction

Arabic language has a very rich vocabulary. More than200 million people speak Arabic as their native language, and over 1 billion people use Arabic language in several religion-related activities. The alphabet set used to write this language is the Arabic alphabet. There are also a number of languages that use the Arabic alphabet, such as Persian, Kurdi, and Jawi [1].

Character recognition is a pattern recognition application with the ultimate aim of simulating the human reading capabilities of both machine-printed, and handwritten cursive text. The systems currently available may read faster than humans, but cannot reliably read such a wide variety of texts nor consider context [2]. Arabic text is inherently cursive both in hand written and printed forms and is written horizontally from right to left. This propriety makes recognition more difficult especially when dealing with multi font characters. Many researchers have been working on cursive script recognition for more than four decades. Nevertheless, the field remains one of the most challenging problems in optical character recognition (OCR) [3]. Due to the cursive nature of the script, there are several characteristics that make recognition of Arabic distinct from the recognition of Latin scripts or Chinese [4].

Ahmed and Al-Ohali [5] discussed the progress and challenges in Arabic character recognition, Mori et al [6] gave an extensive review of the literature for off-line recognition in general. Amin [7] provided an extensive survey of off-line approaches to recognition of Arabic script. Trier et al [8] presented a survey on feature extraction methods for character recognition.

In this research work a simple and effective method is proposed for recognizing multi font isolated Arabic characters by arranging character classes in grand classes of characters of similar shapes then using Fourier descriptors to identify the grand class while using dots information represented by small number of dots and their position (above, within, or below) to the character to identify a specific class in the grand class. Fig. (1) shows the 28 classes of isolated Arabic characters

أ ب ت ث ج ح خ د ذ ر
ز س ش ص ض ط ظ ع غ ف
ق ك ل م ن ه و ي

Fig.(1) Isolated Arabic characters.

Arabic characters can be divided according to the number of dots for each character, they classified into the following 4 sets:

- 1st set includes all the characters without dots (ح د ر س ص ط ع ل م ه و)
- 2nd set includes all the characters that have only one dot (أ ب ج خ ذ ز ض غ ف ك ن)
- 3rd set includes all the characters with two dots (ت ق ي)
- 4th set includes all the characters with three dots (ث ش)

2. The Proposed System

Fig.(2) shows the proposed recognition system. As first step, each character image segment preprocessed; it is converted to gray image, and then to binary using thresholding method. The binary character object image is narrowed using thinning algorithm. Then, its points are collected using contour follower algorithm, and in the same time gather information about character objects dots. The character object start pixel is allocated to create the chain code. The registered coordinates of the collected character object chain code points are normalized. At the last step, the descriptive features for each character segment are calculated using Fourier Descriptors.

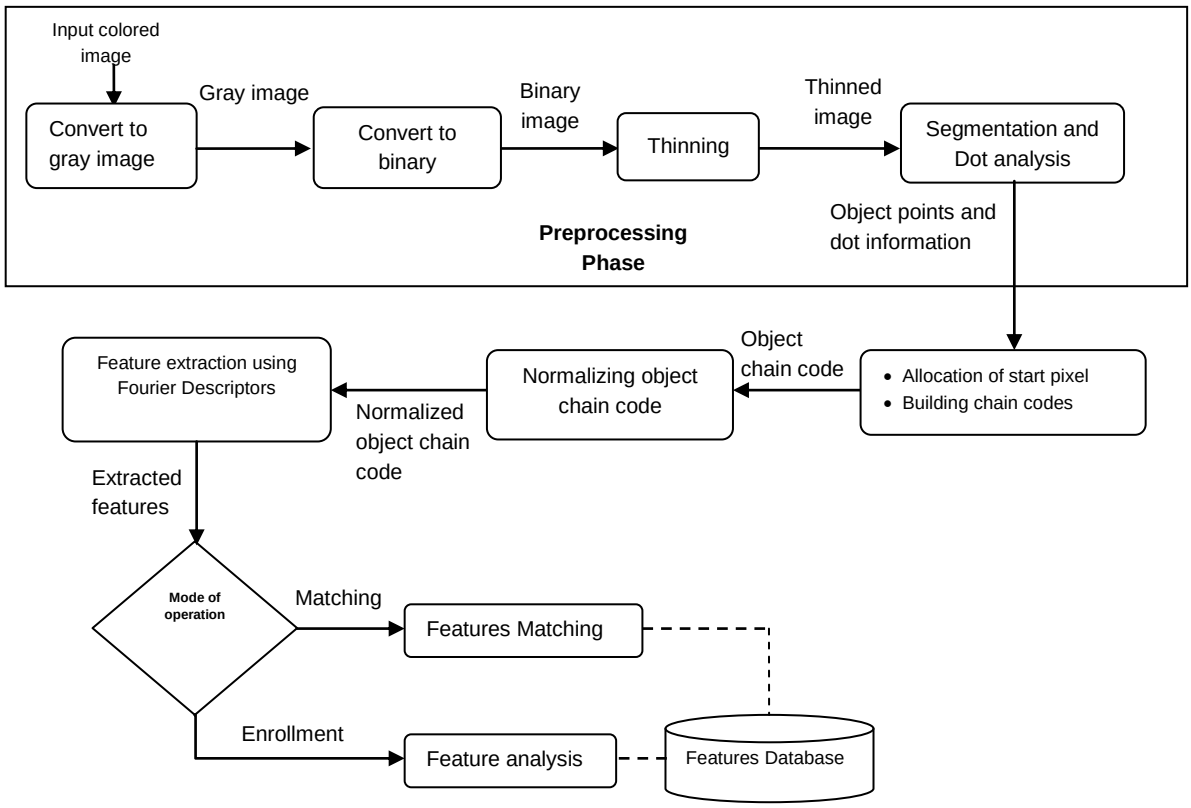

Fig.(2) The proposed system.

2.1 Conversion to Binary

The characterimageis converted to grayscale using the following equation

$$Gry_{xy} = \frac{R_{xy}+G_{xy}+B_{xy}}{3} \quad \text{.....(1)}$$

At the next step, the gray image is converted to binary image using thresholding method [9], the test trails indicated that the threshold value of 190 leads to best separation between the background and the charactersobject. The image was thinned using the Hilditch algorithm [10].

2.2Segmentation

The character object'spixels are extracted by scanning the image object's pixels from top to bottom and from left to right looking for object pixel (with gray value = 0). Once, an object point is met then the contour follower algorithm is activated to collect the object pixels.

After finding an object, a window around the object is opened to search for any near objects that may represent the dots of the current character object.

If no other object is found then the current object belongs to set number one. If other near objects are found then the objects are sorted according to their size (which is represented by the number of pixels for that object) where the largest object represent the character itself (because the character object itself is always larger than its dots) and the other objects are the dots for this character.

If the number of other objects (dots) is 3 then this object belongs to set number four, if the number of other object is 2 then this character belongs to set number three, if the number of other objects is 1 then this character belongs to set number two.

At the end of the segmentation phase, the characters object's pixels and the gathered information about their associated dots (if found; represented by the number of dots and their position relative to the character, above, below, or inside the character body) becomes available. The collected characters objects are send to the next phase for locating the start pixel and dot information is used later in the discrimination among character classes of the same grand class.

2.3Allocation of Start Pixel

In order to achieve high recognition rates, the Fourier Descriptors requires to trace pixels in the same direction for each character object, either clock-wise or counter clock-wise. The sane direction depends on which terminal pixel is selected as the start pixel. Arabic characters have zero to four terminal pixels; the character ا have no terminal pixels and some characters like س may have 4 terminal pixels. In order to solve this problem, the character segment space was divided into 2equally horizontal areas and the start pixel can be found by ordering the terminals pixels according to their occurrences in the 2areas starting with the upper area and if more than one terminal pixel is found in the same area then the terminal pixel with the least x coordinate is selected (and the first found terminal is adopted as the start pixel).

2.4Building Chain Codes

The algorithm proposed by Noor et al [11] is used to create the chain code for any thinned character segment.It is based on tracking the segment pixels sequentially and bypasses any problem may could occur (like, when the tracking visit a complex point like T-junction with more than 2 different paths).

2.5Normalizing Pixels Coordinates

After the establishment of the chain code for each character segment, the coordinates of all collected pixels are normalized to be invariant to translation. Equations (2-4) have been used for mapping x and y coordinates to their corresponding normalized values.

$$x_n = \frac{x - x_{min}}{D} \quad \text{.....(2)}$$

$$y_n = \frac{y - y_{min}}{D} \quad \text{.....(3)}$$

$$D = \max\{x_{max} - x_{min}, y_{max} - y_{min}\} \quad \text{.....(4)}$$

Where n is the number of pixels of the traced character segment, x_n and y_n are the normalized coordinates values, x and y are the image pixel coordinates, x_{min} is the smallest registered x coordinate value, x_{max} is the largest registered x coordinate value, y_{min} is the smallest registered y coordinate value, y_{max} is the largest registered y coordinate value.

2.6Extraction Features

Fourier descriptors are used to describe the objects shape in terms of its spatial frequency content.In this research they used as the main recognition criterion to produce a set (i.e., feature vector) consist of 17 features.Equations (5 - 8) are used to determine the 17 features. For each character segment its created chain-code consists of the pixels' normalized coordinates pairs (x_n, y_n):

$$F(n) = F_R(n) + j\, F_I(n) \quad \text{.....(5)}$$

$$F_R(n) = \sum_{i=0}^{M} x_n(i) \cos\left(\frac{2\pi i n}{M}\right) + y_n(i) \sin\left(\frac{2\pi i n}{M}\right) \quad \text{.....(6)}$$

$$F_I(n) = \sum_{i=0}^{M} -y_n(i) \cos\left(\frac{2\pi i n}{M}\right) + x_n(i) \sin\left(\frac{2\pi i n}{M}\right) \quad \text{.....(7)}$$

$$|F(n)| = \sqrt[2]{F_R(n)^2 + F_I(n)^2} \quad \text{.....(8)}$$

Where, n is the number of features (n=1, 2 … 16), F(n) is the Fourier descriptor or feature, $F_R(n)$ is the real part of Fourier descriptor,

$F_I(n)$ is the imaginary part of Fourier descriptor, M is the number of pixels in a character segment, $x_n(i)$ and $y_n(i)$ are the x and y normalized coordinates of pixel number i in a character segment.

The determined features values {F ()}were normalized, to be in the range between 0 and 1, using the following:

$$NFD = \begin{cases} 0 \; if \; FD \leq min_c \\ \frac{FD - min_c}{max_c - min_c} \; if \; FD > \min_c \; and \; FD < max_c \\ 1 \; if \; FD \geq max_c \end{cases} \quad \text{.....(9)}$$

Where min_c and max_c are computed using the following:

$$min_c = mean_c - 2.5 * std_c \quad \text{.....(10)}$$

$$max_c = mean_c + 2.5 * std_c \quad \text{.....(11)}$$

Where $mean_c$ is the mean of all collected feature values belong to the same character class, std_c is the corresponding standard deviation value. After determination of Fourier descriptor values, and before determining their normalized value, each feature value, F(i) for i=1 … 16, is divided by zero-order descriptor value, i.e. F(0).This division will let the new 16 features become scale invariant. Rotation invariant of the features is achieved by ignoring the phase information and taking only the magnitude values of Fourier features.

3. Results

In this research, the 8most commonly used font styles were adopted (Arabic Typesetting, Arial, Courier New, Microsoft Sans Serif, Segoe UI, Tahoma, Times new roman, and Traditional Arabic) and for each font style the 4 most common font sizes were considered (i.e., 12, 14, 16, and 18). Using Microsoft Office(2010) Word program, 8tables were created and each table contains character samples of a specific font style in 4 different font sizes. Fig. (3) shows some characters of different font styles and in 4 font sizes. Then all printed tables are scanned using a flatbed scanner with sampling resolution of 200 dbi, and the produced sample data were stored as color images, then each image was separated into 28 image stripes. The overall number of produced samples is 896 with 112 samples per font style.

ب	ج	ش	م	**Arabic Typesetting**
ب	ج	ش	م	**Arial**
ب	ج	ش	م	**Courier New**
ب	ج	ش	م	**Microsoft Sans Serif**
ب	ج	ش	م	**Segoe UI**
ب	ج	ش	م	**Tahoma**
ب	ج	ش	م	**Times New Roman**
ب	ج	ش	م	**Traditional Arabic**

Fig. (3) Some characters of different font styles with different font sizes.

To perform the tests, for each font style, the mean and standard deviation of each feature of each class were computed using the following:

$$\sum_{c=1}^{nc} \sum_{f=1}^{nf} mean(c,f) = \left(\frac{\sum_{s=1}^{ns} fdf(c,s,f)}{ns} \right) \quad \text{.....(12)}$$

$$\sum_{c=1}^{cn} \sum_{f=1}^{nf} std(c,f) = \frac{\sum_{s=1}^{ns} \sqrt{(fdf(c,s,f) - cmean(c,f))^2}}{ns} \quad \text{..... (13)}$$

Where c is the character class number, nc is the total number of character classes which is 28, f is the feature number, nf is the total number of features which is 16, s is the sample number, ns is the total number of samples per class which is 4, fdf (c,s,f) is the Fourier descriptor feature number f of sample number s of class number c, mean (c,f) is the mean of all features of class number c of feature number f, and std (c,f) is the standard deviation of all features of class number c of feature number f.

After computing the mean and standard deviation, the minimal distance of each feature of each character sample is computed using the following:

$$\sum_{c=1}^{nc} \sum_{s=1}^{ns} \sum_{f=1}^{nf} d(c,s,f) = \min_{tc=1 \; to \; nc} \left(\frac{fdf(c,s,f) - mean(tc,f)}{std(tc,f)} \right) \quad \text{.....(14)}$$

Now, if the minimal distance found was computed using the class number (tc) of the same class number (c)the character sample belong to, then this character sample was recognized correctly.

Algorithm (1) was used to find the best feature and best recognition rate. To find the best features combination for more than one feature, the distance of all of those features are summed to produce a final feature and the tests are performed on this final one.

```
bestrec = 0
for f = 1 to nf do
rec = 0
for c = 1 to nc do
for s = 1 to ns do
        d = |(fdf(c,s,f) – mean(1,f)) / std(1,f)|
mind = d
minc = 1
fortc = 2 to nc do
          d = |(fdf(c,s,f) – mean(tc,f)) / std(tc,f)|
if d < mind then
mind = d  :  minc = tc
end if
end for
ifminc = c then rec = rec + 1
end for
end for
if rec >bestrec then
bestrec = rec  :  bestf = f
end if
end for
recrate = bestrec x 100 / (nc x ns)
```

Algorithm (1): Finding best feature and best recognition rate

First, each font style samples which are composed of 28 separated character classes were tested and the conducted results are shown in Table (1), these results were produced using Fourier descriptors features only without the use of dots information.

It has been noticed that the 28 characters classes can be reduced to 18 grand classes after removing the dots from the characters, as shown in Fig.(4). Each grand class is a set of one or more character classes. After that, the Fourier descriptors was used to recognize the grand class while the dots information (number and position) were be used to identify the character member that belong to the grand class.

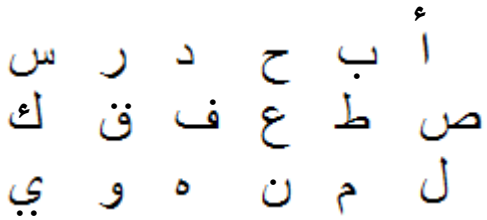

Fig. (4) Grand classes consisting of IsolatedArabic characters without dots.

Through experiments it has been found that Fourier descriptors are best used when Arabic characters are categorized into 15 grand classes:

[أ], [ب ت ث], [ج ح خ], [د ذ], [ر ز], [س ش], [ص ض], [ط ظ], [ع غ], [ف], [ق ل ن], [ك ي], [م], [ه], [و]

The results obtained by applying the method introduced in this work for every font style are shown in Table (2).By testing a combination of character samples of the most 2 commonly used font styles (i.e., Arial and Times new roman) the achieved recognition rate was 100% because these 2 fonts are very close in their shapes and thus have very similar Fourier descriptors, but when adding the Traditional Arabic font style to the test, the recognition ratio decreases to 93.2% as shown in Table(3), and as more font styles are used together in the test, the recognition ratio decreases more.

Table (1)
The test results when using Fourier descriptors combinations on different font styles, each consist of 28 character classes.

Font style	*Number of Fourier Descriptors used*	*Best Fourier descriptors combination*	*Number of recognized samples*	*Recognition ratio without Grand Classes*
Arabic Typesetting	6	0+1+8+9+11+12	89	79.5%
Arial	8	0+2+3+6+9+11+12+15	103	92%
Courier New	10	0+1+4+5+6+8+9+10+12+15	98	87.5%
Microsoft Sans Serif	8	0+2+3+5+8+11+13+14	97	86.6%
Segoe UI	10	0+2+4+6+7+8+9+10+12+14	105	93.8%
Tahoma	9	0+2+4+5+7+8+9+10+13	106	94.6%
Times New Roman	7	0+1+2+4+9+11+15	93	83%
Traditional Arabic	11	0+1+2+3+4+5+6+7+10+12+15	104	92.9%
Average:		88.7%		

Table (2)
The test results when using Fourier descriptors combinations on different font styles of the 15 grand classes.

Font style	*Number of Fourier Descriptors used*	*Best Fourier descriptors combination*	*Number of recognized samples*	*Recognition ratio with Grand Classes*
Arabic Typesetting	6	0+1+3+4+9+15	108	96.4%
Arial	5	0+1+2+10+15	112	100%
Courier New	9	0+1+3+4+5+7+9+12+15	110	98.2%
Microsoft Sans Serif	5	0+2+5+7+10	104	92.9%
Segoe UI	6	0+2+6+8+9+15	107	95.5%
Tahoma	5	0+2+3+5+10	109	97.3%
Times New Roman	4	0+1+9+14	112	100%
Traditional Arabic	5	0+1+3+7+8	112	100%
Average:		97.6%		

Table (3)
The test results when using Fourier descriptors combinations on 2 and 3 most commonly used font styles of the 15 grand classes.

Font style	*Number of Fourier Descriptors used*	*Best Fourier descriptors combination*	*Number of recognized samples*	*Recognition ratio with Grand Classes*
Arial+Times New Roman	6	0+1+4++9+10+14	224	100%
Arial + Times New Roman + Traditional Arabic	6	0+1+2+4+12+15	313	93.2%

3. Conclusions

The test results indicated that the use of grand classes with Fourier descriptors plus the characters dots information (represented by number and position) are excellent for recognizing isolated Arabic characters of a specific font style or of multi font styles similar in their shapes regardless of the font size. But, when using them in the recognition of multi font styles of different shapes the recognition rate decreases, and to increase the recognition, a multi template method or neural network method should be adopted.

References

[1] Khorsheed M. S., "Off-Line Arabic Character Recognition–A Review", *Pattern Analysis & Applications Journal*, Vol. 5, No. 1, pp. 31-45, 2001.

[2] Abdullah I. Al-Shoshan, "Arabic OCR Based on Image Invariants", *Proceedings of the Geometric Modeling and Imaging-New Trends (GMAI'06)*, pp. 150 – 154, 2006.

[3] Nadia Ben Amor , Najoua Essoukri Ben Amara, "An approach for Multifont Arabic characters features Extraction based on Contourlet Transform", *Ninth International Conference on Document Analysis and Recognition (ICDAR)*, Vol. 2, pp. 1048 – 1052, 2007.

[4] Nadia Ben Amor, "Multifont Arabic Character Recognition Using Hough Transform and Hidden Markov Models", *Proceedings of the 4th International Symposium on Image and Signal Processing and Analysis (ISPA)*, pp. 285 – 288, 2005.

[5] Pervez Ahmed, Yousef Al-Ohali, "Arabic Character Recognition: Progress and Challenges", Journal of King Saud University-Computer & Information Sciences, Vol. 12, No. 1, pp. 85-116, 2000

[6] MoriS., Suen C.Y. and Yamamoto K., "Historical review of OCR research and development", Proceedings IEEE, Vol. 80, Isuue. 7, pp. 1029-1058, 1992.

[7] Amin A, "Off-line Arabic character recognition - the state of the art", Pattern Recognition, Vol. 31, No. 5, pp. 517-530, 1998.

[8] Qivind Due Trier, Anil K. Jain and TorfinnTaxt, "Feature Extraction Methods for Character Recognition–A Survey",

Pattern Recognition, Vol. 29, No. 4, pp. 641-662, 1996.
[9] John C. Russ, Fifth Edition, "The Image Processing Handbook", CRC, 2007.
[10] Louisa Lam, Seong-WhanLee, Member, IEEE, and Ching Y. Suen, Fellow, IEEE, "Thinning Methodologies–A Comprehensive Survey", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 14, No. 9, September 1992.
[11] Shatha M. Noor, Ihab A. Mohammed, and Loay E. George, "Handwritten Arabic (Indian) Numerals Recognition Using Fourier Descriptor and Structure Base Classifier", Journal of Al-Nahrain University, Vol. 14, No. 2, pp. 215-224, 2011.

الخلاصة

يعد موضوع التعرف البصري للاحرف من المواضيع التي ما زالت قيد البحث منذ الايام الاولى للحاسبات. و بالرغم من قدم الموضوع، لكنه يدخل ضمن نطاق البحوث ذات التحديات و التشويق العالي في مجال التطبيقات في الحاسوب. في السنوات الاخيرة اصبح الموضوع اكثر نضجا و تحول الى نظام قائم بحد ذاته و انتجت كثير من الاعمال في هذا المجال. يعتبر موضوع التعرف على الحروف العربية من المواضيع التي لم تلق اهتماما الا مؤخرا و ذلك بسبب طبيعة الحروف كونها حروف متصلة حتى في حالة الحروف المطبوعة. في هذا البحث، تم اقتراح طريقة بسيطة و فعالة للتعرف على الحروف العربية المفصولة باستخدام واصفات فورير و معلومات النقاط المتمثلة بعددها و موقعها. تم استخدام ثمانية انواع من اكثر الخطوط الشائعة الاستخدام و بمختلف الاحجام، في البداية تمت الاختبارات على كل نوع من الخطوط بشكل منفصل و كانت النتائج ممتازة. تم اجراء الاختبارات على مجموعة انواع من الخطوط و اشرت النتائج انه كلما زادت عدد الخطوط المستخدمة في الاختبار قلت نسبة التمييز.