A New Approach to Segmentation of Persian Cursive Script based on Adjustment the Fragments

Mir Mohammad Alipour
Department of Computer Engineering, University of Bonab, Bonab 5551761167, Iran

ABSTRACT
Optical Character Recognition (OCR) is a very old and of great interest in pattern recognition field. The recognition of cursive scripts like Persian and Arabic languages is a difficult task as their segmentation suffers from serious problems in different languages. Segmentation is a process of dividing cursive words into smaller parts in order to decrease complexity and increase accuracy of recognition process.

In this paper, an improved segmentation method of the Persian script has been presented and to increase the quality of segmentation, some structural features of Persian language is used to adjust the fragments. This method is robust as well as flexible. It also increases the system's tolerances to font variations. The proposed method is able to segment existing Persian fonts up to 99.2% accuracy.

Keywords
Cursive Script, Persian, Segmentation, Optical Character Recognition, Adjustment the Fragments.

1. INTRODUCTION
Optical Character Recognition (OCR) is the task of categorizing an image as a character in a given script. The process is within the field of pattern recognition, which aims to develop new capabilities in computer systems that will allow higher levels of automation.

Persian or Farsi is the official language of more than 150 million peoples of the world. Like Arabic, Persian is a right to left script, but there are some differences like number of alphabets, font styles, vocabulary and signs, which make Persian OCR somehow different from Arabic. In last decades several researchers worked on Arabic OCR [1–6], and as a result today there exist some commercial OCR products for Arabic language. But in the field of Persian language, lots of OCR papers are about isolated character/digit recognition [7-10] and there are only a few papers in the field of printed text recognition [11-15]. It seems that the first paper about Persian printed text recognition is [16].

Generally, Persian/Arabic Recognition can be done offline or online. In offline recognition, papers, manuscripts or documents are scanned or captured, and finally are manipulated by OCR system. In online recognition application takes place during the writing process, many systems were developed for manipulating online Arabic OCR such as [17,18]. The online recognition system however is beyond the scope of this paper.

Segmentation is the most crucial and most difficult step in a Persian/Arabic OCR. It involves segmenting the text image into blocks, lines, words and ultimately into characters. A poor segmentation process produces misrecognition or rejection [19]. It is especially important for Persian OCR systems due to the cursive nature of Persian script and the fact that some Persian words overlap vertically [20]. Amzi and Kabir [11] proposed a new segmentation algorithm based on the conditional labeling of the upper contour. They also proposed a technique to adjust the local baseline for each sub-word. Menhaj and Adab [14] proposed a segmentation and recognition method for printed text recognition. They used Fourier descriptors as features and MLP as classifier.

In this paper, a new segmentation method of the Persian language based on adjustment the fragments is proposed. That is, first Persian words are fragmented using their structural properties and connectivity points and then produce characters by merging fragments. The proposed method is simple to implement and does not require lengthy numerical computations.

The organization of the paper is as follows: some background knowledge about the characteristics of Persian script is given in Section 2. The structure of the Persian/Arabic OCR system is given in Section 3. The processes involved in the image acquisition and preprocessing stages are also presented in this section. A detailed description of the proposed segmentation approach is found in Section 4. The proposed system's experimental results are presented in Section 5. Finally, the paper concludes in Section 6.

2. GENERAL CHARACTERISTICS OF PERSIAN SCRIPT
The Persian language (Farsi) was written in Cuneiform until the Arab conquest of Iran in 642CE,when it was replaced by Arabic script. However, Iran retained its own language and as a result, required augmentation of the Arabic alphabet to cater for new sounds. As such, any system recognizing Persian script will also recognize Arabic. In addition to the 28 Arabic characters, four additional characters are used for the sounds "Ch", "P", "G" and "Zhe"[2]. The form of the character, analogous to the upper and lower cases in English, depends upon it's position in the word and the characters which surround it. The four forms are Detached, Initial, Medial and Final. The alphabet, showing all four forms, is displayed in Fig. 1.

[1] This work is supported by University of Bonab under Research Projection 100-12.

[2] similar to the French pronunciation of "J"

Name	Detached	Initial	Medial	Final	Name	Detached	Initial	Medial	Final
Alef	ا	ا	ا	ا	Sad	ص	صـ	ـصـ	ـص
Be	ب	بـ	ـبـ	ـب	Zad	ض	ضـ	ـضـ	ـض
Pe	پ	پـ	ـپـ	ـپ	Ta	ط	طـ	ـطـ	ـط
Te	ت	تـ	ـتـ	ـت	Za	ظ	ظـ	ـظـ	ـظ
Se	ث	ثـ	ـثـ	ـث	Ein	ع	عـ	ـعـ	ـع
Jim	ج	جـ	ـجـ	ـج	Ghein	غ	غـ	ـغـ	ـغ
Che	چ	چـ	ـچـ	ـچ	Fe	ف	فـ	ـفـ	ـف
He	ح	حـ	ـحـ	ـح	Qaf	ق	قـ	ـقـ	ـق
Khe	خ	خـ	ـخـ	ـخ	Kaf	ک	کـ	ـکـ	ـک
Dal	د	د	ـد	ـد	Gaf	گ	گـ	ـگـ	ـگ
Zal	ذ	ذ	ـذ	ـذ	Lam	ل	لـ	ـلـ	ـل
Re	ر	ر	ـر	ـر	Mim	م	مـ	ـمـ	ـم
Ze	ز	ز	ـز	ـز	Nun	ن	نـ	ـنـ	ـن
Zhe	ژ	ژ	ـژ	ـژ	Vav	و	و	ـو	ـو
Sin	س	سـ	ـسـ	ـس	Hed	ه	هـ	ـهـ	ـه
Shin	ش	شـ	ـشـ	ـش	Ye	ی	یـ	ـیـ	ـی

Fig.1: The Persian Alphabet

Here we describe the most important features of Persian script from OCR point of view**:**

- Letters are written from right to left, but numerals are written from left to right. We should consider this attribute for regenerating recognized text.
- Words are aligned by a horizontal virtual line called "baseline" and are separated by long spaces. Each word is composed of one or more character(s) and/or sub-word(s). A sub-word is a combination of joint letters. Sub-words are separated by short spaces and if one sub-word has more than one character, each of them will be connected to its neighbors along the baseline. For example the word Persian "فارسی" is composed of one letter, "ر", and two sub-words, "فا" and "سی", each of them having two letters.
- Some letters only differ in number or position of their dots, e.g. letters Je "ج", He "ح", Khe "خ" and Che "چ" have the same shapes, but different dots.
- Some digits are very similar to some letters: digit 1 "1" and character Alef "ا", digit 0 "0" and dot ".", digit 5 "5" and character Ha "ه".
- Each letter can take some signs like Tashdid " ّ ", Hat as in " آ ", Sarkesh as in "گ".
- In some fonts, digits 4, 5 and 6 and letters Kaf and Ye, have two shapes, which both of them may be used, e.g. "۴" and "٤" for digit 4 and "ک" and "ك" for letter Kaf.
- Some characters may be horizontally overlapped with others in a document. Two characters "و" and "ج" of the word moj "موج" overlapped each other.

3. PERSIAN/ARABIC CHARACTER RECOGNITION

The typical Persian/Arabic OCR system consists of five components: image acquisition, preprocessing, segmentation, feature extraction and classification (recognition) [20]. Each of those contributes to the final recognition rate to improve of the OCR system.

3.1 Image Acquisition

Transforming the written text in papers or transcript to digital format is a necessary step in the OCR. In the image acquisition the paper is scanned or captured. The scanner speed, document types and scanning quality need to be considered in the document scanning.

3.2 Preprocessing

The aims of image preprocessing are to reduce the noise coefficients and to increase the readability of the input by the processing system. The preprocessing stage is also necessary to increase the uniformity in texts which is quite essential for recognition system.

The Preprocessing stage is the important stage of Persian/Arabic OCR. It directly affects the reliability and efficiency in the segmentation, feature extraction and classification process [21]. In order to improve the Persian/Arabic OCR system performances, generally, preprocessing stage should contain binarization, filtering and smoothing and thinning (Skeletonization).

3.2.1 Binarization

The Persian/Arabic OCR systems usually accept inputs in bi-level format or to be more specific binary format. Generally, the input text images in grayscale. Hence, we need a preprocessing stage called binarization. It converts from gray scale image to bi-level image taking into consideration a threshold pixel value for comparison. The threshold pixel value can be computed based on the histogram of the gray values of the images.

3.2.2 Filtering and Smoothing

Noise may appear in the images after scanning or binarization. It is necessary to remove the noise and smoothing the input text image to prepare the data for the further processing. Generally the Persian/Arabic OCR systems are very sensitive to noise. It influences negatively the system performances. The images processing Median or Gaussian filters usually use to remove the noise [22].

The nature of the Persian/Arabic writing or the binarization process may produce small holes or unwanted edges. These small holes and unwanted information can affect directly in the systems performance. The small holes should be closed and the unwanted information should be deleted by using the opening and closing morphology operation respectively [22].

3.2.3 Thinning the Text

Thinning is very important in Persian/Arabic OCR system. It simplifies the texts shapes for segmentation process, feature extraction, and classification. This is resulted in reducing the amount of data that need to be handled [20].

3.3 Segmentation

Segmentation problem is the most difficult and important issue in the Persian/Arabic OCR. It directly affects the feature extraction and classification process [23] [24]. Although some segmentation methods have been created for segmenting the Persian/Arabic text, the problem is still remain as an unsolved

issue. This may due to the complex characteristics of the Persian/Arabic written text.

The Persian/Arabic text segmentation methods can be classified broadly into two approaches: First is called holistic approach or segmentation-free approach. This technique aims to segment the text to words or sub-words [20] [24]. It splits the paragraph into separate lines and then split these lines into words or sub-words. The horizontal projection method is usually used to segment the paragraph into lines [25] while the vertical projection method is used to segment the lines into words or sub-words [6].

Second approach is called Analytical approach, in this method, the Persian/Arabic word or sub-words segment into small classifiable element or into fragments (tokens), these fragments could be parts of a character, characters, mixture of characters [24].

3.4 Feature Extraction

Feature selection is one of the most critical issues in character recognition as the recognition rate is dependent on the choice of features. It has a big influence on the classification stage [20]. Every character has some features that distinguish it from the other characters. Some of the commonly used features for character recognition are loops, holes, strokes, vertical lines, cusps, etc.

The Persian/Arabic text feature extraction methods can be classified broadly into three main groups, these groups are structural features, statistical features and Global Transformations: the structural features method, in this technique, features are usually extracted based on the text topologies; the structural features of Arabic text may include loops, the intersection points, dots, height, width, number of crossing points, and such [26] [27]. The second is called statistical features methods, these techniques are quick and effective, but may be affected by noise. The statistical features used for Arabic text recognition include: zoning, characteristic loci, crossings and moments [28] [29]. The third is called Global Transformation methods; the Global Transformation aim to shorten the text representation in order to get better results. The global transformations methods used for Persian/Arabic text recognition include: horizontal and vertical projections, coding, Hough transform, Gabor transform [30].

3.5 Classification

The classification process is carried out at the final stage to recognize the character. It assigns an unknown feature into a predefined class. AOCR systems can recognize the text by either the Holistic (Global) or Analytic strategies. The Holistic (Global) strategy recognizes the whole words or sub-words, as well as it does not require segmentation, and it works on limited number of vocabularies [31]. On the other hand the Analytic Strategy recognizes the segmented features, as well it requires segmentation, and can be applied on unlimited vocabularies [32].

4. PROPOSED SEGMENTATION APPROACH

In the proposed approach, segmentation of text into individual characters is performed in four steps:

1- Line Segmentation and Baseline Detection,
2- Word Segmentation,
3- Word Fragmentation,
4- Adjustment the Fragments.

4.1 Line Segmentation and Baseline Detection

This process aims to separate the pairs of consecutive lines in the text. The process is based on the analysis of the horizontal projection of the text. The segmentation between lines of text is determined by scanning through the profile from the first row. If the difference of the number of black pixels between two rows is larger than a predefined threshold, a new line of text is indicated (depicted in Fig. 2). The next large variation in the number of black pixels between another two rows indicates the bottom of the line.

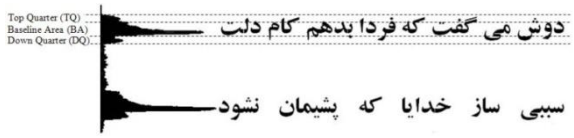

Fig. 2: Horizontal projection of a given Persian text and three sections of a text line (TQ, BA and DQ)

As can be noticed from the figure, the peak in the histogram corresponds to the baseline in the text. For improving the quality of segmentation, each line of the text is divided into three sections: top quarter of text line (TQ), baseline area (BA) and down quarter of text line (DQ) (as shown in Fig. 2).

4.2 Word Segmentation

After a text is segmented, it is scanned vertically. If in one vertical scan two or less black pixels are encountered then the scan is denoted by 0, else the scan is denoted by the number of black pixels. In this way a vertical projection is constructed. Now, if in the profile there exist a run of at least *k* consecutive 0's then the midpoint of that run is considered as the boundary of a word. The value of *k* is taken as a half of the text line height. An illustration of the method is shown in Fig. 3.

Fig. 3: Word Segmentation

4.3 Word Fragmentation

Word fragmentation involves the building of vertical projection of the connected parts of the word with baseline (without considering dots and other signs of the word's characters). A fixed threshold is used for segmenting a word into fragments. In other words, whenever the value of the vertical projection of the middle zone is less than two thirds of the baseline thickness, the area is considered a connection between two fragments. From the threshold level the algorithm searches for the break along the vertical projection.

Although these lines do not necessarily segment characters from a word correctly, the location and the number of them would directly affect the accuracy and speed of the OCR system [33]. It is important to note that this step only produces a sequence of fragments, while the segmentation of characters is confirmed at the next step. Fragmentation of the word ashkhas "اشخاص" without considering its dots is shown in Fig. 4.

Fig. 4: Word fragmentation lines

4.4 Adjustment the Fragments

In this step, fragments that is produced at word segmentation step, scanned from left to right and each fragment with its width less than a threshold value δ (an experimental value that is less than the average width of all fragments), checked for merging with neighbor fragments. Adjustment is done for merging small fragments that are parts of a character and are not whole characters. First let define Circle Form fragments (CF). The fragments that satisfy next four conditions are CF:

1- At the middle of the fragment and above the baseline the pixel's density is zero (hasn't any pixel in TQ section of the fragment),

2- At the middle of the fragment and under the baseline the pixel's density is high (continues stream of pixels in DQ section of the fragment),

3- The fragment hasn't any dot,

4- Pixels at middle of the fragment don't across the baseline.

CF is made at the end of these characters "س", "ش", "ص" and "ض". An example of CF fragment is shown in Fig. 5.

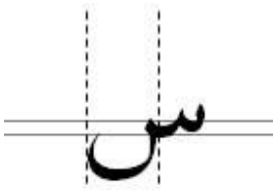

Fig. 5: An example of CF fragment

Proposed Algorithm for adjusting the fragments is shown in Fig. 6.

```
1   Sub Adjust_Fragments
2     For each line of the text Do
3       For each word on the current text line (from left to right) Do
4         CurFrag = Current Fragment
5         If  Width(CurFrag) ≥ δ Then
6           If  Previous(CurFrag) = CF then
7               'One of these characters, sad "ص" and zad "ض" are made up.
8             Merge CurFrag , Previous(CurFrag)
9           End If
10        Else If  TqPix(CurFlag) Is Empty And HasDot(CurFlag) = False Then
11          If  Width(Next( (CurFrag)) < δ And Previous(CurFrag) = CF Then
12              'One of these characters, sin "س" and shin "ش" are made up.
13            Merge Previous(CurFrag) , CurFrag , Next(CurFrag)
14          Else
15              'One of these characters, middle sin "سـ" and middle shin"شـ" are made up.
16            Merge CurFrag , Next(CurFrag) , Next(Next((CurFrag))
18          End If
19        End If
20      Next
21    Next
22  End Sub
```

Fig. 6: Algorithm of Adjusting the Fragments

Lines 7, 12 and 15 are comments. Function Width() returns width of given fragment (as its parameter) and function Previous() returns previous fragment of given fragment and function Next() returns next fragment of given fragment and function HasDot() returns True if given fragment has one or more dots and otherwise returns False (this test is done to exclude merge of fragments containing initial and middle forms of the characters be "بـ", pe "پـ", te "تـ", se "ثـ", noon "نـ" and ye "یـ") and function TqPix() returns pixel density of Top Quarter of the given fragment (this test is done to exclude merge of fragments containing characters alef "ا" and lam "ل"). For example Adjusting fragments of the word ashkhas "اشخاص" is shown in Fig. 7.

Fig. 7: Adjusted word fragments

5. EXPERIMENTAL RESULTS

To evaluate the efficiency and accuracy of proposed approach, we developed a program to segment Persian cursive script in offline mode. C# language and Visual Studio 2010 were used as developing language and environment respectively. Input data was 100 scanned pages of typed Persian text with 10 different Persian fonts and sizes, using a flatbed scanner. The accuracy of our proposed segmentation approach is shown in Table 1.

Our experiments showed 98.02% average accuracy in identifying right segments. As you can see from the Table 1, fonts like Nazanin and Titr which are similar to machine-printed style, have a high accuracy and overlapping fonts like Sina and Traffic which are more similar to handwritten style, have a low accuracy. It's notable that increasing the font size, it increases the accuracy of segmentation.

Table 1: Accuracy of Proposed Segmentation approach

Font	Size						Average
	8	10	14	18	24	36	
Zar	97.2	97.5	97.8	98	98.3	98.4	97.86
Nazanin	98.8	99.1	99.2	99.2	99.4	99.5	99.2
Titr	98.5	98.8	99.2	99.4	99.5	99.7	99.18
Badr	97.8	97.9	98.2	98.3	98.5	98.6	98.21
Nasim	96.7	97	97.2	97.5	97.9	98.1	97.4
Yagut	97.5	97.6	97.8	98.2	98.5	98.5	98.01
Traffic	96.6	96.9	97.2	97.5	97.6	97.8	97.26
Lotus	97.9	98.2	98.5	98.6	98.6	98.8	98.43
Roya	98.6	98.7	99	99.1	99.2	99.4	99
Sina	94.5	94.8	95.5	95.8	96.5	96.8	95.65
Average							98.02

6. CONCLUSION

Segmentation introduces the most serious problem in the development of cursive script OCR system including Persian/Arabic language scripts. In order to overcome this problem, we proposed a new segmentation approach based on adjustment the fragments. That is, we fragmented Persian words using their structural properties and connectivity points. Then, we produce characters by adjusting the fragments. The method is simple to implement and does not require lengthy numerical computations.

The system was tested on different groups of pages with different properties. The test contains 100 pages with 10 different fonts and sizes that are used regularly in Persian printing. Proposed segmentation approach has 98.02% average accuracy in identifying right segments. As seen from the result of our experiments, our technique is robust to different fonts of Persian scripts.

7. REFERENCES

[1] Abdelazim, H.Y., Hashish, M.A, 1988. Arabic reading machine. In: Proceedings of the 10th National Computer Conference, Jeddah, pp. 733–744.

[2] Al-Shoshan, A.I., 2006. Arabic OCR based on image invariants. In: Proceedings of the International Conference on Geometric Modeling and Imaging-New Trends, pp. 150–154.

[3] Amin, A., 1998. Off-line Arabic character recognition: the state of the art. Pattern Recognition. **31**(5), 517–530.

[4] Gouda, A.M., Rashwan, M.A. 2004. Segmentation of connected Arabic characters using hidden Markov models. IEEE International Conference on Computational Intelligence for Measurement Systems and Applications, USA pp. 115–119.

[5] Kurdy, B., AlSabbagh, M., 2004. Omnifont Arabic optical character recognition system. In: Proceedings of International Conference on Information and Communication Technologies: From Theory to Applications, pp. 469–470.

[6] Sarfraz, M., Nawaz, S.N., Al-Khuraidly, A., 2003. Offline Arabic text recognition system. In: Proceedings of International Conference on Geometric Modeling and Graphics, pp. 30–35.

[7] Khosravi, H., Kabir, E., 2007. Introducing a very large dataset of handwritten Farsi digits and a study on their varieties. Pattern Recognit. Lett. **28**(10), 1133–1141.

[8] Mansoory, S., Hassibi, H., Rajabi, F., 1998. A heuristic Persian handwritten digit recognition with neural network. In: The 6th Iranian Conference on Electrical Engineering, pp. 131–135.

[9] Nabavi, S.H., Ebrahimpour, R., Kabir, E., 2005. Recognition of handwritten Farsi digits using classifier combination. In: Third Conference on Machine Vision, Image Processing and Applications, Tehran, pp. 116–119.

[10] Soltanzadeh, H., Rahmati, M., 2004. Recognition of Persian handwritten digits using image profiles of multiple orientations. Pattern Recognit. Lett. **25**(14), 1569–1576.

[11] Azmi, R., Kabir, E., 2001. A new segmentation technique for omnifont Farsi text. Pattern Recognit. Lett. **22**, 97–104.

[12] Ebrahimi, A., Kabir, E., 2008. A pictorial dictionary for printed Farsi subwords. Pattern Recognit. Lett. **29**(5), 656–663.

[13] Mehran, R., Pirsiavash, H., Razzaziy, F., 2005. A front-end OCR for omni-font Persian/Arabic cursive printed documents. Digital Imaging Computing: Techniques and Applications, pp. 385–392.

[14] Menhaj, M.B., Adab, M., 2002. Simultaneous segmentation and recognition of Farsi/Latin printed texts with MLP. In: International Joint Conference on Neural Networks, pp. 1534–1539.

[15] Yazdi, S.A.B., A'rabi, B.N., 2007. Printed Farsi text recognition with simultaneous use of HMM. In: Dynamic Programming and SVM (in Farsi), Machine Vision and Image Processing, Mashhad.

[16] Parhami, B., Taraghi, M., 1981. Automatic recognition of printed Farsi texts. Pattern Recognit. Lett. **14**, 395–403.

[17] S. Al-Emami and M. Usher, 1990. On-line Recognition of Handwritten Arabic Characters. IEEE Trans. Patt Anal. Machine Intell. 12(7): 704 – 710.

[18] A. Alimi and O. Ghorbo, 1995. The analysis of error in an on-line recognition systems of Arabic handwritten characters. Proceedings of the Third International Conference on Document Analysis and Recognition. 2: 890 – 893.

[19] L. Hamami and D. Berkani, 2002. Recognition system for printed Multi-Font and Multi-Size Arabic Characters, Arabian Journal for Science and Engineering, Vol 27, Number 1B, 57-72.

[20] M .Liana, and G. Venu, 2006. Offline Arabic Handwriting Recognition: A Survey. IEEE, Transactions on Pattern Analysis and Machine Intelligence. 28: 712-724.

[21] F. Farooq, V .Govindaraju, and M. Perrone, 2005. Pre-processing Methods for Handwritten Arabic Documents. (ICDAR'05) Proceedings of the 2005 Eight International Conference on Document Analysis and Recognition, IEEE. 1. pp. 267-271.

[22] C. Gonzales. Rafael and E. Richard,Woods., 2002. Digital Image Processing. 2nd ed. Englewood Cliffs, NJ: Prentice-Hall.

[23] R. Safabakhsh, and P. Adibi, 2005. Nastaaligh Handwritten Word Recognition Using a Continuous-Density Variable-Duration HMM. The Arabian Journal for Science and Engineering. 30: 95-118. April.

[24] A.M. Zeki, 2005. The segmentation problem on Arabic character recognition – the state of the art. 1st International Conference on Information and Communication Technology (ICICT). pp. 11-26. Karachi, Pakistan.

[25] A. Amin, and J.F. Mari, 1989. Machine recognition and correction of printed Arabic text. IEEE Transactions on Systems, Man and Cybernetics (SMC), 19(5): 1300-1306.

[26] H. Goraine, M. Usher, and S. Al-Emami, 1992. Off-Line Arabic Character Recognition, Computer, vol. 25, pp. 71-74.

[27] A. Amin and H. Alsadoun, 1994. Hand printed Arabic character recognition system. Proceedings of the 12th International Conference A on Pattern Recognition, IAPR, pp 536–539.

[28] B. AL -Badr and S. Mahmoud . 1995. Survey and bibliography of Arabic optical text recognition. Signal Processing, 41(1): 49-77.

[29] H. Sanossian, 1996. An Arabic character recognition system using neural network. Proceedings of 1996 IEEE Signal Processing Society Workshop, Kyoto, Japan, IEEE, pp; 340–348.

[30] F. Zaki, S. Elkonyaly, A. Elfattah, and Y. Enab, 1986. A new technique for arabic handwriting recognition. Proceedings of the 11th International Conference for Statistics and Computer Science, Cairo, Egypt, pp; 171–180.

[31] A. Dehghani, F .Shabani and P. Nava, 2001. Off-Line Recognition of Isolated Persian Handwritten Characters Using Multiple HiddenMarkov Models, Proc. Int'l Conf. Information Technology: Coding and Computing, pp. 506-510.

[32] S. Mozaffari, K. Faez, and M. Ziaratban, 2005. Structural Decomposition and Statistical Description of Farsi/Arabic Handwritten Numeric Characters, Proc. Int'l Conf. Document Analysis and Recognition, pp. 237-241.

[33] A. Cheung, M. Bennamoun, and N. W. Bergmann, 1997. A New World Segmentation Algorithm for Arabic Script, DICl'A'97, pp. 431-435, Auckland, New Zealand.