J. Basic. Appl. Sci. Res., 2(9)9563-9568, 2012

ISSN 2090-4304
Journal of Basic and Applied Scientific Research
www.textroad.com

Farsi Font and Font Size Recognition Based on Analyzing Binarization Effect on Small Components of Document Images

Yaghoub Pourasad[1], Azam Ghorbani[2], Saman ghouparanloo[3]

[1]Department of Electrical Engineering, Urmia University of Technology, Urmia, Iran
[2]Department of Engineering, Saveh Brach, Islamic Azad University, Saveh, Iran
[3]Department of Electrical Engineering, Sharif University of Technology, Tehran, Iran

ABSTRACT

This work presents a new method for Farsi font and font size recognition. There are few papers about Farsi font recognition and in these researches only font of an image document is recognized. On the other hand, finding the font size of a text, in addition to its font, can be very useful in document image analysis. Therefore, we present an approach that recognizes the font, and finds the font size of a document image. The method is based on binarization of document and then analyzing the effect of binarization on the document, including the size and shape of dots and broken strokes, which are formed in binarization step to recognize text font and font size. To evaluate our system, a database including 10*49 text images of 7 different fonts and 7 different sizes are formed using paint software, and recognition rate of 95.7% is achieved. This method is applicable on some other languages such as Arabic and Urdu.
KEYWORDS: Font size, Font recognition, Binarization, Farsi document image, Histogram.

1. INTRODUCTION

Today document images such as scanned text documents are widely used. For example there are many documents that are scanned and reserved electronically in some libraries but Computers aren't able to search or understand context of such documents. One way to overcome this problem is OCR (Optical Character Recognition). An OCR system consists of several modules that one of them is character recognition [1]. It is obviously clear that understanding the font and font size of text of document image, can help us to have better results in character recognition. Font recognition and font size estimation can be very helpful in retrieval systems, too. The field of text font recognition in document images especially in Farsi language is new and needs more attention. There are two common approaches in font recognition field: first is based on typographical features and second is based on textual features. In the first approach, features like character weights, space width and various projections are used. Whereas in second approach textual features are extracted using wavelet transform, Gabor filter or other techniques. In [2], an approach for the recognition of Farsi fonts is proposed. In this paper font recognition is performed in line level using a feature based on Sobel and Roberts gradients in 16 directions, called SRF. SRF is extracted as texture features for the recognition. This feature requires much less computation rather than other textual features and therefore can be extracted very faster than common textual features like Gabor filter, wavelet transform or momentum features. The reported recognition rate is about 94.2% using 5000 samples of 10 popular Farsi fonts. In [3], an approach for Arabic font recognition is presented. Their proposal is to use a fixed length sliding window for the feature extraction and to model feature distributions with Gaussian Mixture Models (GMMs). The main advantage of this approach is that a priori segmentation into characters is not necessary and the authors reports performances above 99% on a set of 9 different fonts and 10 different sizes. In [4], the use of global texture analysis for Farsi font recognition in machine-printed document images is examined. They consider document images as textures and use Gabor filter responses for identifying the fonts. Two different classifiers including Weighted Euclidean Distance (WED) and Support Vector Machine (SVM) is used for classification. Authors reported average accuracy of 85% with WED and 82% with SVM classifier on 7 different face types and 4 font styles. All above references that are about font recognition [2, 3, 4], are font size independent and don't give information about font size of document. Although methods based on typographical features and approaches based on textual features are common methods, but there are a few other works that are different of these approaches. In [5], first, dots of document are extracted and size of dots is estimated using weighted sum variance. Then pen width is supposed to be nearly square of dot size. But for font size estimation as writers have noticed, there isn't fixed relation between pen width and font size; therefore they assumed an approximate relation between font size and pen width. This approach is fast but

***Corresponding Author:** Yaghoub Pourasad ,Department of Electrical Engineering, Urmia University of Technology, Urmia, Iran, Y Pourasad@ee.Kntu.ac ir,Tel:+982144831769

only estimates an approximate value for font size and doesn't recognize the font of text of document. There are some papers that calculate pen width and use it in recognition part of OCR systems but don't discuss about font and font size. One can use this approaches to calculate pen width and then has an approximate value for font size. The most common method for finding pen width, is using horizontal or (and) vertical projection profile [6, 7], and obtaining base line or height of each line [8]. Anyway, these methods only calculate pen width and can give an approximate value for font size but don't recognize font of document. In [9], first, second and third order moments of the input image are used as features and correlation coefficients are used to recognize Farsi fonts. In [10, 11, 12, 13,14], some other papers about Farsi font recognition have been presented.

In this work we don't calculate pen width to estimate font size. Our method directly calculates the font size and recognizes the font of Farsi documents by using the size of bounding boxes of single dot or double dot components and also small strokes that are formed after binarization. In Farsi, there are more than 500 different fonts. Developing a system that considers all these fonts is difficult and useless. Therefore, we concentrate on 7 widely used fonts and 7 different font sizes.' Lotus',' Nazanin', 'Mitra',' Yaghut',' Zar',' Koodak','Homa' , are some of the most popular fonts in Farsi that we focused on them. The font sizes that we considered in this paper are, 8,10,12,14,16,18,20.

There are main differences between Farsi and English scripts, so most of the methods which applied for English documents aren't applicable on Farsi documents. There are 32 basic characters in Farsi scripts and shape of these characters may change according to their position (beginning, middle, end or isolated) in the word. Each character can take up to four different shapes, as result there are 128 different shapes for all of Farsi alphabet. In addition, Farsi script is written from right to left and moreover, the characters of the words in Farsi texts are connected to each other both in handwritten and printed texts. In English texts, each word is composed of some letters with similar letter size. This feature is used to recognize the font size of a text. But in Farsi, each word is composed of sub-words. The sizes of sub-words - a part of word that all of its letters are connected- are greatly different. Most of the Farsi characters (18 out of 32) have one, two or three dots which can be situated at the top, inside or bottom of the characters.

In this paper, using special characteristics of Farsi scripts, a new method for font recognition has been presented. In proposed method it is said that when a word is written in a specific font and font size, its dots form, varies in comparison with the case that same word is written in a different font and font size. We use this characteristic of Farsi scripts for font recognition. For this purpose, we constructed a dataset including some feature vectors for each font and font size of Farsi scripts. When a document image is entered for font recognition, its feature vectors are extracted and compared with stored feature vectors in dataset, and its font and font size is recognized. The most advantage if proposed method is that, it can recognize font size of document image in addition to font face.

This paper is organized as follows. In section 2 our new method and used dataset description are described. In this section, we first, describe our proposed method and then, explain how we construct two datasets in order to extract and store feature vectors for each font. Section 3 is related to results and discussions. In this section, testing situation of proposed method, obtained results of proposed method, results of other methods, and finally, comparison of our results with other similar results is presented. In section 4 which is conclusion, a brief summary of proposed method, its implementation, and its results are presented.

2. MATERIALS AND METHODS

Our experiments show that the most frequent components of almost every Farsi documents are dots. Dot components may consist of one, two or three dots. In cases that they are double or triple, it is very probable that they connect to each other. It is very interesting and useful that dot components in different fonts and font sizes have different sizes and shapes after binarization. Figure 1 shows some single dots in different fonts before binarization and after binarization. In the first row of figure 1, single dots before binarization and in the second row after binarization are showed. In (a) a dot in 'homa font with font size of 18' is showed. The dot in (b) is in 'homa font with font size of 12', (c) is a dot with 'Zar font with size of 18', (d) is a dot with 'Zar font with font size of 12', (e) is a dot with 'Lotus font with size of 18', and (f) is a dot with 'Lotus font with font size of 12'.As is shown in the second row of figure 1, these single dots, have different shapes after binarization. In the third row of figure 1, size of bounding box of these components is presented. It is clearly seen that single dots in different font and font sizes represents different shapes and sizes.

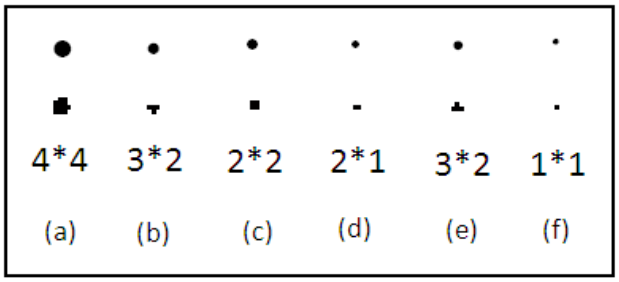


Fig 1: Some single dots in different fonts before and after binarization

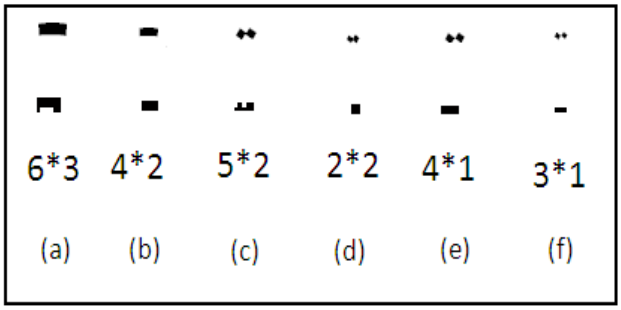


Fig 2: Some double dots in different fonts before binarization and after binarization

Figure 2 shows some double dot components in different fonts before binarization and after binarization. In the first row of figure 2 double dots before binarization and in the second row after binarization are showed. As is shown in the second row of figure 2, these double dots, have different shapes after binarization. In the third row of figure 2, size of bounding box of these components is shown. It is clearly seen that as seen in single dot component, double dot components in different font and font sizes represents different shapes and sizes, too. We can use this behavior of dots to describe every Farsi font and font size. Another point that is used in this paper is that with binarization of Farsi documents in some fonts and especially in small font sizes (8, 10 and 12) very small broken strokes is formed. On the other expression, binarization is caused that some parts of a word that were connected before binarization be disconnected after binarization. We call these new components broken strokes and size of the bounding box of the most of these broken components are similar to size of bounding box of single dot or double dot components. In fact binarization is caused to decrease the quality of text of document. But we benefit of this failure and use this bad effect as a feature. Our experiments show that different fonts represent different but constant and permanent behaviors while binarization with a constant threshold value. It means that binarization of a document that is written in specific font and font size, with a special threshold, leads to generation of broken strokes with predictable sizes. It is interesting that issue of context of document don't have important effect on this behavior. For example if a Farsi document be written in 'Lotus font with font size of 10', binarization will cause to decrease the quality of image and to form broken strokes that their size is similar with double dot components (3*1). If we apply same value as a threshold for binarization of a same context but in font 'Yaghut' with font size of 10, broken strokes with size of 4*1 will generate.

As mentioned earlier, size of broken strokes is similar to single dot and double dot components. In this paper we considered 7 widely used fonts and 7 widely used font sizes. Thus totally we have 49 states and for each state we experimentally extracted size of bounding box of single dot and double dot components and broken strokes. After that, we described each state with its 3,4or 5 most frequent sizes. These sizes are related to bounding box of single dot and double dot and broken strokes components. With analyzing histogram of each state we can find its most frequent sizes that are related to dots and broken strokes. After analyzing each state we can describe that state with its 3, 4 or 5 more frequent sizes.

When we want to find the font and font size of an unknown document, its features are obtained in the same manner that described. Then its features are compared with the features of all 49 states that have been obtained and reserved before. If features of query document is compatible with the features of any of 49 states, font and font size of that document is recognized and presented.

3. RESULTS AND DISCUSSION

In this paper we constructed two sets of text document images. In first set we constructed 5 images for every state (one font of 7 fonts and one font size of 7 font sizes). We used this set to extract robust features for every state. In second set we constructed 10 images for every state. Second set is used for testing the system. In construction of both sets we tried to have images with different issues and different sizes. For example we made images that their issues were about electronics, chemistry, sports, etc. In these images there are documents that have only a few lines and documents with more than 10 lines. For second set that is used for test, we constructed documents that had figures and graphics or English words beside Farsi text. In order to construct both sets first, we prepared a text in Microsoft word software. Then using print screen key of keyboard, a picture of that text was provided. After that, using paint software, we did necessary corrections and then saved it in bmp format. For all images these steps have been done.

To obtain features of a document that was in the special font with special font size (one state of 49 states), we first binarized that document with threshold value of 1.3T. Where 1.3 is a typical coefficient and T is obtained from Otsu´s global thresholding method. Then connected component algorithm applied to binerized document. After that, histogram of components with sizes smaller than 7*6 obtained.

Experimental results show that in 49 described states, dots and broken strokes have bounding boxes with sizes smaller than 7*6. For this reason, we focused on histogram of components with sizes between 1*1 up to 7*6. In figure 5 some histograms for different states are showed. In these figures the horizontal coordinate is related to size of bounding box of connected components. As it is seen, values of horizontal coordinates are from 1 up to 42. The numbers in the range of [0-6] in the horizontal coordinates are symbols for the connected components with sizes 1*1, 1*2, 1*3, 1*4, 1*5, 1*6. And the numbers in the range of [7,12] are symbols for the connected components with sizes repectively,2*1, 2*2, 2*3, 2*4, 2*5, 2*6,etc.

While extracting the features of states, after obtaining their histograms, we observed that histograms of some states were completely different with others histograms; but there were some states that their histograms were slightly similar with some other states. When histogram of one state is completely different with others, its features are different with others features. Features of these states are describable with only 2 or 3 main components. But when 2 or 3 main components of two or more states are similar to each other, we forced to use 4 or 5 main components of that state as its feature. In figure 3 some different histograms that are related to different fonts and font sizes, are showed.

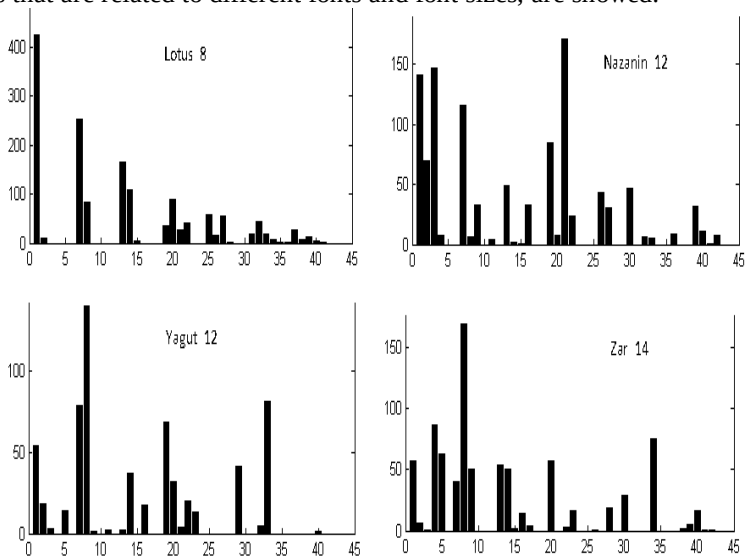


Fig 3: Histogram of size of bounding box of some fonts and font sizes

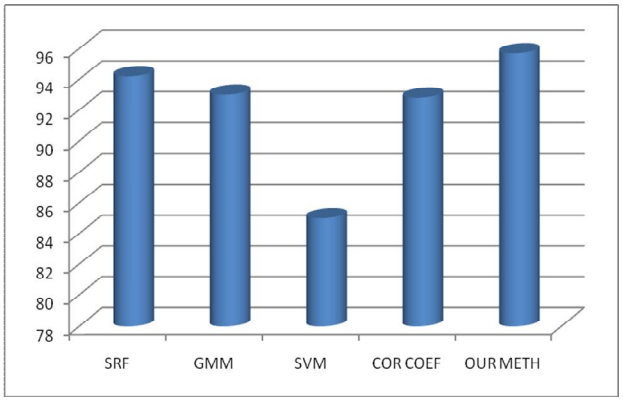
To test our approach we used second set of document images that there were 10 images for every state in it. While testing system we observed that the recognition of bigger fonts (16, 18 and 20) is better than smaller types and probability of mistake in their recognition is very low which is because of their features that are completely different with other´s. whereas small font sizes such as 8, 10 are mistakable with each other because their main components are slightly similar.

In this approach all feature extraction and test stages have been done using MATLAB software in a 2.4 GHz Pentium PC. Recognition rate is more than 95.7%. This 4.3% mistake is often related to some small fonts. In some cases that we considered them as mistakes, were situations that for a query document, 2 different states were recognized through our

system. For example for a query document that was written with 'Nazanin font size of 10', our system recognized 'Nazanin font with font size of 8' besides of 'Nazanin font with font size of 10'. Experimental results show that our approach is very fast. For example for recognizing an A4 document which is full of text, less than 0.1 second is required. Reason of this advantage is related to very few features that are considered for each state. As mentioned before in our method, each state is described with 2, 3, 4, or 5 features; while in other papers feature vectors are very bigger. For example in [2], which is one of the best papers about Farsi font recognition, length of feature vector is 512, or in [14], length of feature vector is 644. Another important advantage of proposed method is its high recognition rate. In table 1 recognition rate of some papers has been showed. In [2], a method for Farsi font recognition is proposed which is based on SRF (Sobel and Robert Features). In this paper using SRF features a method, and neural networks, font of Farsi document images with recognition rate of 94.2% is recognized. In [3], gaussian mixture models (GMM) are used for Arabic font recognition. Recognition rate of this method is 93%. In [4], using support vector machine (SVM), font of Farsi document images with recognition rate of 85% is presented. In [9], using correlation coefficients (COR COEF), font of Farsi document images with recognition rate of 92.8% is recognized. These results in addition to recognition rate of our method are presented in table 1, and graph1.

Table 1: Recognition rate of some works and our work

Method	Recognition Rate
SRF [2]	94.2%
GMM [3]	93%
SVM [4]	85%
CORE COEF[9]	92.8%
Our method	95.7%



Graph1: Recognition rate of some works and our work

As seen in table 1 and graph1, recognition rate of our method is better in comparison with other papers. The most important reason for high recognition rate of our method is that we used dots of letters in document images. The fact is that in every Farsi document image certainly there exist a lot of dots therefore we have very repeatable and robust feature vectors.

Another advantage of our proposed method is that our method can recognize font size of document images in addition to font face, but other papers don't mention to this matter.

4. Conclusions

In this paper we present a new approach that recognizes the font and font size of Farsi document images. In this paper the size of bounding boxes of single dot or double dot components and broken strokes is used as features of different fonts and font sizes. This approach is very fast and represents recognition rate more than 95.7% while testing on 7 fonts in 7 font sizes. The system easily can be expanded for testing more fonts and font sizes. One important advantage of this system is that existence of pictures or drawings or words of other languages such as English doesn't disturb performance of system. This approach is applicable for some other similar languages such as Arabic language. In future works we will try to increase recognition rate of system and have more reliable results.

REFERENCES

1. Pourasad, Y., H. Hassibi, M. Banaeyan, 2011. Persian character recognition based on spatial matching. International Review on Computers and Software (I.RE.CO.S). Vol. 6. No. 1, PP: 55-59.

2. Khosravi, H. and E. Kabir, 2010. Farsi font recognition based on sobel-roberts features. Journal of Pattern Recognition Letters 31(1), PP: 75-82.

3. Slimane, F., S. Kanoun, a. m. Alimi, R. Ingold, and J. Hennebert, 2010. Gaussian Mixture Models for Arabic Font Recognition".International Conference on Pattern Recognition. PP: 2174-2177.

4. Borji, A., M. Hamidi, 2007. Support Vector Machine for Farsi font recognition. Word Academi of science, Engineering and Technology, 28. (int. j. intell. technol. 2(3)). PP: 10-13.

5. Shirali, M. H., Shirali, S., 2006. Farsi/Arabic text font estimation using dots. IEEE International Symposium Signal Processing and Information Technology. PP: 420-425.

6. Mehran, R., H. Pirsiavash, and F. Razzazi, 2005. A Font-End OCR for Omni-Font Persian/Arabic Cursive Printed Documents. Proceedings of Digital Image Computing: Techniques and Applications (DICTA 05), PP: 385-392.

7. Omidyeganeh, M., K. Nayebi, R. Azmi, and A. Javadtalab, 2005. A New Segmentation Technique for Multi Font Farsi/Arabic Texts. Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing. (ICASSP05), vol. 2, PP: 757-760.

8. Bushofa, B.M.F. and M. Span, 1997. Segmentation of Arabic characters using their contour information. Proceedings of 13th International Conference on Digital Signal Processing Proceedings (DSP 97), Greece, vol. 2, PP: 683-686.

9. Rashedi, E., H. Nezamabadi-pour, S. Saryazdi, 2007. Farsi font recognition using correlation coefficients (in Farsi). In: 4th conf. on Machine Vision and Image Processing, Ferdosi Mashhad.

10. Chaudhuri B.B., U. Garain, 1998. Automatic detection of italic, bold and all-capital words in document images. In: Proceedings of 14th International Conference on Pattern Recognition, (ICPR), Brisbane ,Australia, PP: 610-612.

11. Pourasad, Y., H. Hassibi, A. Ghorbani, 2011. Farsi font recognition using holes of letters and horizontal projection profile. First conference of Innovative computing Technology (INCT), vol. 241, part 5, PP: 235-243.

12. Pourasad, Y., H. Hassibi, A. Ghorbani, 2012. Farsi word spotting and font size recognition. Journal of Procedia Technology, vol. 1, PP: 372-377.

13. Pourasad, Y., H. Hassibi, A. Ghorbani, 2012. Farsi font face recognition in letter level. Journal of Procedia Technology, vol. 1, PP: 378-384.

14. Pourasad, Y., H. Hassibi, M. Banaeyan, 2011. Farsi font recognition based on spatial matching. 18th international conference on systems, signals, and image processing, Sarajevo, Bosnia & Herzegovina, 16-18 June, PP: 71-74.