IJNES

Internatıonal Journal of Natural and Engineering Sciences 5 (3): 17-20, 2011
ISSN: 1307-1149, E-ISSN: 2146-0086, www.nobel.gen.tr

Farsi Font Recognition in Document Images Using PPH Features

Yaghoub POURASAD[1] Houshang HASSIBI[1] Azam GHORBANI[2]

[1] Department of Electrical and Computer Engineering, K. N. Toosi University of Technology, Tehran, IRAN

[2] Department of engineering, Islamic Azad University, Saveh branch, Saveh, IRAN

***Corresponding Author**
e-mail: y_pourasad@ee.kntu.ac.ir

Received: June 06, 2011
Accepted: June 15, 2011

Abstract

In this paper a new approach for font recognition of Farsi document images is presented. In this approach using two features PP and H, font and font size of a Farsi document image is recognized. Feature H is related to holes of imaged text document. Feature PP is related to horizontal projection profile of text lines of document image. This approach is applied on 7 widely used Farsi fonts and 7 font sizes. A dataset containing 10*49 images and another dataset including 110 images were used for testing and recognition rate more than 93.7% obtained. Images were made using paint software and were noiseless and without skew. This approach is fast and is applicable for other languages that are similar to Farsi, such as Arabic language.

Key Words: Farsi Font Recognition, Document Image, Projection Profile, Base line.

INTRODUCTION

Nowadays very documents 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. It is clear that understanding the font and font size of text of a document image, can help us to have better results while character recognition. It 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 textural features. In the first approach, features like character weights, space width and various projections are used. Whereas in second approach textural features which are extracted using wavelet transform, Gabor filter or other techniques, are used. In [1] an approach for recognition of Farsi fonts is proposed; in which, font recognition is performed in line level using a feature called SRF. This feature is based on Sobel and Roberts gradients in 16 directions. SRF is extracted as a texture feature for the recognition. This feature requires much less computation in comparison with other textural features and therefore can be extracted very faster than common textural 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 [2] 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. Its authors reported performance about 99% on a set of 9 different fonts and 10 different sizes. In [3] the use of global texture analysis for Farsi font recognition in machine-printed document images is examined. They use Gabor filter responses for identifying the fonts. Two different classifiers including Weighted Euclidean Distance (WED) and Support Vector Machine (SVM) are 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 [1, 2, 3], are font size independent and don’t give information about font size of document.

Although methods based on typographical features and approaches based on textural features are common methods, but there are a few other works that are different from these approaches. In [4] 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 as writers have noticed, there isn’t a 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 only estimates an approximate value for font size and doesn’t recognize the font of text of document. In [5] first, second and third order moments of the input image are used as features and correlation coefficients are used to recognize Farsi fonts. As mentioned, one important part of any OCR system is recognition part, and one of tasks that is done in recognition module, is pen width calculation. The field of pen width calculation is near to the field of font size recognition and some approaches that are used in pen width calculation can be used in font size recognition. But it should be noted that with knowing the pen width, we only have an approximate value for font size. So, many approaches find pen width and use it in recognition part, and then if it is required, can give 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 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 document by using the PPH features. In fact we use horizontal projection profile and bounding box size of words holes to recognize font and font size of a document image.

In Farsi, there are more than 500 different fonts. Developing a system that can consider 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, and 20.

This paper is organized as follows. In section 2 dataset description, in section 3 feature extraction and in section 4 experimental results are presented and finally section 5 is conclusion.

MATERIALS AND METHODS

Dataset Description

In order extract features and evaluate the approach, three sets of imaged text documents were constructed. In first set we constructed 5 document images for each state of 49 states (7 fonts and 7 font sizes). We used this set to extract robust features for each state. In second set we constructed 10 document images for each 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 (8, 10, 12, 14, 16, 18, and 20). For example we made images that their issues were about electronics, chemistry, sports, etc. In these images there were documents that had only a few lines and documents with more than 10 lines. For second set that was used for test, we constructed documents that were written in one state of 49 predefined states. In third set we constructed 110 images of fonts that were different with 49 predefined states. In order to construct all three 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.

Table 1. Number and percentage of each type of errors

Error type	Error numbers	Error percentage
Error 1	6/490	1.2%
Error 2	4/490	0.8%
Error 3	8/490	1.6%
Error 4	3/110	2.7%
Total errors	21	6.3%

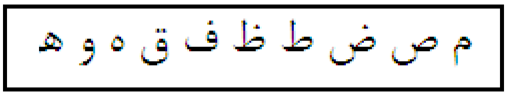


Figure 1. Some letters that have hole

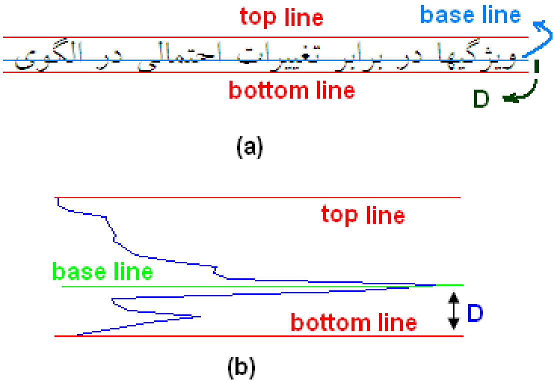


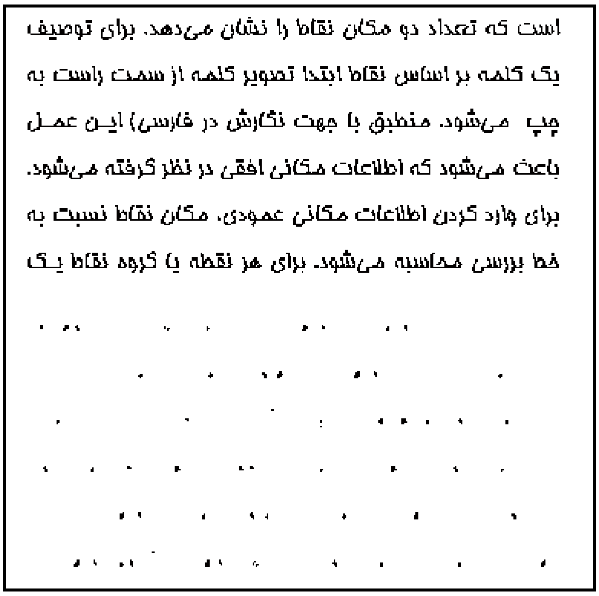
Figure 2. A Farsi text line, its horizontal projection profile, base line, top line and bottom line.

Feature Extraction

A large number of Farsi alphabet letters have one or two holes. These holes represent different shapes and sizes in different fonts and font sizes. Therefore with analyzing the holes of a document image we can approximately recognize the font and font size of that document image. In figure 1 some Farsi letters that have hole, are showed.

Another feature that can be helpful for font recognition of a text line is its horizontal projection profile. Several features can be extracted from horizontal projection profile of a text line. One of them is the height of text line. Another feature can be the distance between top of text line and base line. Base line is part of a text line that the most letters of text line are written on it. Third feature is distance between bottom of text line and base line. We show this feature with D. In figure 2 a Farsi text line, its horizontal projection profile, base line, top line and bottom line are showed. Experimental results show that in Farsi documents, for every text line, third feature is more permanent and reliable than first and second features. It can be said that in absence of some especial letters such as (ق ، ع ، ح) for a text line, D is approximately half of font size of that text line. For example, for font size 8, D is 4. Or for font size 10, often D is 5. While extracting D features of text lines, existence of special letters (ق ، ع ، ح) must be considered. For example for text lines written in 'Nazanin 8' font, if there is one or more especial letters D=6 and else, D=4.

Another useful feature that can be extracted from horizontal projection profile of a text line is the position of second order maximum or third order maximum of projection profile in comparison with position of base line. In some fonts second order maximum is above the base line whereas in some other fonts second maximum is under the base line. In some fonts especially in small fonts such as 8, 10, there is only one maximum in horizontal projection profile of each text line and that maximum is related to base line.



است که تعداد دو مکان نقاط را نشان می‌دهد. برای توصیف
یک کلمه بر اساس نقاط ابتدا تصویر کلمه از سمت راست به
چپ می‌شود. منطبق با جهت نگارش در فارسی) این عمـل
باعث می‌شود که اطلاعات مکانی افقی در نظر گرفته می‌شود.
برای وارد کردن اطلاعات مکانی عمودی، مکان نقاط نسبت به
خط بررسی محاسبه می‌شود. برای هر نقطه یا گروه نقاط یـک

Figure 3. A Farsi document and its extracted holes

H Feature Extraction

As mentioned before, we constructed 5 document images for each state in set1. In order to extract H features of one state, all 5 images of that state, are binarized using threshold value of 1.4*T; Where T is threshold value that is obtained from Otsu global thresholding method and 1.4 is a selective value that has been obtained experimentally. After changing the gray scale document image to binary, holes of text of document are extracted and then connected component algorithm is applied. Then histogram of bounding box size of holes is obtained. With analyzing all 5 histograms of one state, we can register all important sizes of holes as that state's H feature. In figure 3, a Farsi text document and its extracted holes are illustrated. In figure 4 histogram of bounding box size of 4 states are illustrated. As seen in the figure 4, bounding box size of holes and consequently, histograms of them in different fonts and font sizes are different; thus, we use this feature of fonts to describe and recognize them

PP Feature Extraction

To extract PP features of text lines of a document image, after binarization step, horizontal projection profile of that document image is obtained. From resultant projection profile, location of text lines is determined. For all text lines of document, D is calculated. For each state of 49 states two value of D is registered; one with existence of special letters (ح ، ع ، ق) and another without existence of special letters. Another feature that is considered as a feature for each state after horizontal projection profile is the number of projection profile maximums and position of them in comparison with position of base line. Experimental results show text lines that are written in small fonts such as 8 and 10 have only one maximum in their horizontal projection profiles while larger fonts have two, three or more maximums. First order maximum (max1) projection profile corresponds to base line. Position of second order (max2) or third order maximum (max3) in comparison with position of base line is different in different fonts and font sizes. It means that in some fonts, max2 is above the base line but in other fonts max2 is under the base line. Also max3 may be above or under the base line. In figure 5 horizontal projection profile (PP) of four different states are showed. In (a), PP of text lines of a document image written with font of Nazanin 8 is represented. This PP has one maximum. PP of (b) has one maximum, too. In (c) max2 is under the base line whereas in (c) max2 is above the base line.

For all 49 states of set1, PP and H features extracted and registered. When a query document image is given to our font recognition system, PP and H features of it, is extracted and is compared with the features of 49 states that have been extracted and reserved before. If features of query be compatible with the features of one of the 49 states, font and font size of that query document image will be recognized.

RESULTS AND DISCUSSION

To test our approach, we first used set2. Testing results show that PP features for small fonts are more precision and reliable than larger fonts. Because in small fonts such as 8,10,existence of special letters (ح ، غ ، ق) increases only one or two pixels to D; whereas in bigger fonts such as 18 or 20, existence of those letters increases even up to 4 pixels to D. But in bigger font sizes, H feature is more helpful. Because in bigger font sizes holes won't be decomposed after binarization and almost all holes are extractable while in small font sizes such as 8 , holes are filled or broken after binarization and aren't extractable

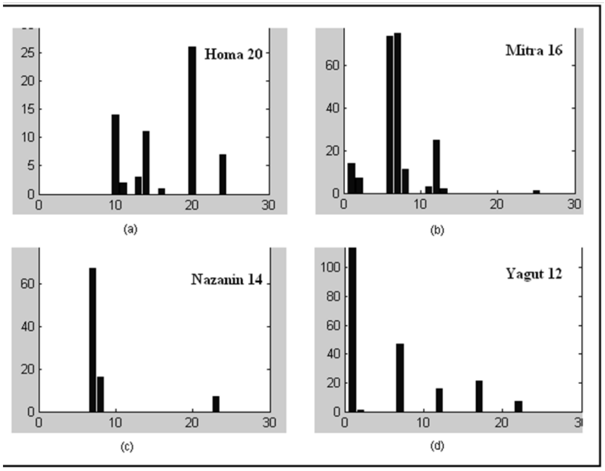


Figure 4. Histogram of bounding box size of 4 states

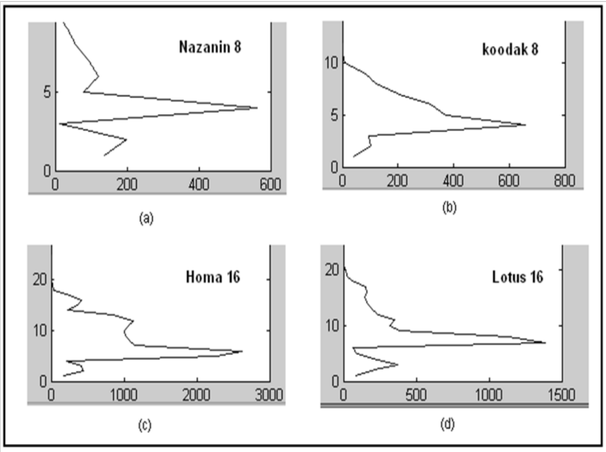


Figure 5. Horizontal projection profile of four different states

easily. In font size 8, even after holes extraction, almost all fonts present similar H features, hence only we can recognize font size but can't recognize font type. But for font sizes 12 and greater, font and font size is exactly recognizable.

While testing system was observed that fonts which substantially are thicker, such as 'Homa' and 'Koodak', represent better results in comparison with substantially thin fonts such as 'Zar'; because in thick fonts, holes of even small font sizes are easily extractable.

After testing system with set2, we used set3 for system testing. While testing, whether set2 or set3, four types of errors were observed:

Error1: Query document image was from set2 but our system didn't recognize any state.

Error2: Query document image was from set2 but system recognized an incorrect state of set2.

Error3: Query document image was from set2, system recognized an incorrect state in addition to correct state.

Error3: Query document image was from set3, system instead of announcement of 'no state', recognized an incorrect state.

In table 1 number and percentage of each type of errors is showed. As seen in this table, our system error rate is 6.3% and recognition rate is 93.7%. Experimental results show that our approach is fast. Reason of this advantage is related to very few features that are considered for each state. For example for recognizing an A4 document which is full of text, less than 0.3 second is required.

In this approach whatever the number of text lines and the words that have holes, be more, PP and H features will be more reliable and recognition rate will be better.

CONCLUSION

In this paper a new method for font recognition is presented. Most of the papers that are about font recognition are font size independent and only recognize the font; but don't recognize the font size of document image. Our approach recognizes both font and font size of a document image. For this purpose we used PPH features that PP features are extracted with forming horizontal projection profile and H features are related to size of bounding box of holes of document. We applied this approach on 7 widely used fonts and 7 font sizes and recognition rate of 93.7% obtained. Whatever the number of text lines and words be more, PP and H features will be more reliable and recognition rate will be better. This approach is fast and is applicable for other languages that are similar to Farsi, such as Arabic language.

REFERENCES

[1] Khosravi H, Kabir E. 2010. Farsi font recognition based on sobel-roberts features. Pattern Recognition Letters. 31: 75-82.

[2] Slimane F, Kanoun S M, Alimi A, Ingold R, hennebert J. 2010. Gaussian Mixture Models for Arabic Font Recognition. In: International Conference on Pattern Recognition.

[3] Borji A, Hamidi M. 2007. Support Vector Machine for Farsi font recognition. J. Word Academi of Science, Engineering and Technology. 28.

[4] Shirali-shahreza MH, Shirali-shahreza S. 2006. Farsi/Arabic text font estimation using dots. In: IEEE International Symposium Signal Processing and Information Technology.

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

[6] Omidyeganeh M, Nayebi k, Azmi R, Javadtalab A. 2005. A New Segmentation Technique for Multi Font Farsi/Arabic Texts. In: Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing. (ICASSP05). 757-760.

[7] Bushofa BMF, Span M. 1997. Segmentation of Arabic characters using their contour information. In: Proceedings of 13th International Conference on Digital Signal Processing Proceedings (DSP 97). 683-686.

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