Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2019) 27: 1477 – 1488

doi:10.3906/elk-1804-92

Farsi document image recognition system using word layout signature

Cem ERGÜN[1*], **Sajedeh NOROZPOUR**[2]

[1]Department of Computer Engineering, Faculty of Engineering, Eastern Mediterranean University, Famagusta, Northern Cyprus

[2]Department of Mathematics, Faculty of Art and Sciences, Near East University, Nicosia, Northern Cyprus

Received: 12.04.2018 • **Accepted/Published Online:** 29.01.2019 • **Final Version:** 22.03.2019

Abstract: In this paper, a new representation of Farsi words is proposed to present the keyword spotting problems in Farsi document image retrieval. In this regard, we define a signature for each Farsi word based on the word connected component layout. The mentioned signature is shown as boxes, and then, by sketching vertical and horizontal lines, we construct a grid of each word to provide a new descriptor. One of the advantages of this method is that it can be used for both handwritten and machine-printed texts. Finally, to evaluate the performance of our system in comparison to other methods, a database that contains 19,582 printed Farsi words is examined, and after applying this approach, a recall rate of 98.1% and a precision rate of 94.3% are obtained.

Key words: Farsi document image retrieval, word spotting, word layout signature, optical character recognition

1. Introduction

Due to the increase in digital libraries and paper documents in offices, their organization and management now take significant amounts of time and energy. This problem appears more often when a specific document among a huge volume of documents is needed. In order to solve such difficulties, paper documents have to be scanned and archived; then, to find a specific document that is needed, some methods are established. This process is called document image retrieval, which has been a hot topic in recent years.

To search for a keyword in document images, first of all, by optical character recognition (OCR), we have to convert the format of document images from pictorial format to text format, which is translatable by the machine [1], and then by the use of the traditional methods of document retrieval, the target word is sought in the text. Although OCR is frequently used by researchers in this area, it has some disadvantages that cause OCR to be inappropriate in all retrieval cases. The most important of these disadvantages is that it costs a lot in converting huge amounts of documents and also it is not sufficiently successful in applying it on low quality texts and documents with complicated layout. Additionally, there is no robust OCR method available yet for Farsi language scripts [2, 3]. In order to overcome these problems, researchers suggested another method for document image retrieval that is called keyword spotting or, more simply, word spotting [4].

Historically, word spotting was first defined in the context of speech processing [5–7]; later on, it was also developed in the context of document image processing in machine-printed texts [8–10]. In document image processing, keyword spotting system gives a "yes" or "no" answer to the user's query by spotting the keyword without doing any letter recognition [11, 12].

*Correspondence: cem.ergun@emu.edu.tr

In recent years, much research has been done in the field of keyword spotting in document images, mostly for the English language with Latin letters, and some work has been done on Korean [13], Chinese [14], and Arabic [15–17] languages, as well. So far, there are few papers related to keyword spotting in the Farsi language. For example, in [18, 19], a system was presented for machine-printed image retrieval of a Farsi word. The main idea used is based on font recognition of document images and the correction is done on the font face and the font size of the query word according to the document's image of the keyword before searching. The similarity between the user's query and images involved in document images is done based on the XNOR similarity measure. Then the topological features of the image, such as the number of holes, number of ascenders/descenders, and number of dots, are used to improve the results. The method is based on pixel resolution and is limited to training fonts. This means that it does not have the capability of extending to more font faces and also has an extra step to recognize the font size, which has a heavy computational load for the system. In [20], by using Farsi topology features such as number of dots, number of subwords, and number of holes, a new way of coding and retrieval of Farsi document images was shown. The work in [20] also contains a way to detect fonts in Farsi texts, which is based on tiny connected components. In another paper published by Ebrahimi and Kabir [21], a method based on the whole shape of words and subwords was presented. Here, principal component analysis (PCA) is used for compressing feature vectors. Then k-means is used for clustering of subwords and the average of each cluster is placed in one pictorial dictionary. Furthermore, an interesting method for retrieval of Farsi document images was introduced in [22], which is independent of recognition. Here, the upper contours of words are extracted and then a picture dictionary of these features is made, and each subword is shown as a combination of contour strokes that includes upper, lower, and middle positions of the baseline. As another example, the work proposed in [23] depends on the feature of the shape of printed words in the recognition of Arabic texts written in three different fonts, two of which are synthetic. Several features such as dots, directional segments, directional cavities, junctions and endpoints, connectors, inner word spaces, and descenders of the Arabic printed words are extracted and saved in a dictionary. The proposed method published recently in [24] determined the ratio of the subword width to the subword height and confined the search range to this ratio. This ratio is calculated according to the symbol positions on a pixel by pixel basis. The large number of subwords is the disadvantage of this method.

As was mentioned earlier, most of the studies on this topic were done in the English language, and we will use some of them in this paper. For instance, a method of retrieval of English document images that is based on word shape coding was done in [25]. In that method, the authors used topological features such as character holes, ascenders/descenders, and character reservoirs. The impressive point of this method is that documents can be retrieved by word shape coding based on both the query of the keyword and the query of the document image. The advantage of shape coding over character encoding is that it has high accuracy in documents with low quality and does not have letter segmentation errors. The attempt made in [26] depends on word recognition without considering OCR. First, a document image retrieval system is divided into two phases: online and offline phases. Then indexing operations are done in the offline system and retrieval operations are used in the online system. Some features are used in the features field such as height to width ratio, word area density, center of gravity, vertical projection, top–bottom shape projections, and upper grid features. Keyvanpour and Tavoli used a feature weighting method to improve their system, which is based on the correlation of features [27].

The main challenges to create a system with good precision in Farsi language scripts are letters' cohesion, existence of dots and symbols, overlapping letters, merging of adjacent letters, and the complication of letters' layouts. Farsi words have a special complication in their letter layout. This property can be considered as a

positive and useful property of Farsi words. The locative layout structure of a word image and classification of components of its layout have useful information for recognizing Farsi words. According to a literature review above and considering the method discussed in [2, 3], in this paper, we propose a new model for machine-printed Farsi text retrieval based on the similarities of layout of components in Farsi words. The new method is actually the implementation of the method proposed in [28, 29]. In [28, 29] the retrieval procedure only needs to compute the signature of the query image and compare it to the other images in the dataset, and the method only considers components as paragraphs or pictures and does not count word by word. In our method, we will consider the layout signature of any words in a text as single word. This method is also tested on some document images.

The remainder of this paper is organized as follows: Section 2 describes our proposed method, Section 3 summarizes the experimental results, and, lastly, Section 4 presents conclusions of this paper.

2. Proposed method

2.1. Preprocessing

In this section, in order to characterize word images using their layouts, we assume that a word image has been deskewed [30, 31] and segmented [31, 32] into components. In our proposed method, any Farsi word is demonstrated by bounding boxes. During implementation, the "bwlabel" function is used in MATLAB software to extract label connected components for binarized word images. Next, using the "regionprops" command, a bounding box representation of words is constructed. Each bounding box contains four segments (two vertical segments and two horizontal segments) with start point (x_s, y_s) and endpoint (x_e, y_e). The collection of all bounding boxes in word layout is shown by $B = \{b_1, b_2, ..., b_{n_B}\}$, where n_B equals the number of bounding boxes. The order of boxes is considered to be started by the upper box on the left-hand side. This means that the counting of boxes is started by b_1, which is located on the upper left side, and this process goes on in the same way. To remove possible ambiguities, the horizontal and vertical lines are denoted by h_i and v_i, respectively.

Clearly, $h_i : (x_i^s, y_i) \rightarrow (x_i^e, y_i)$, and similarly, $v_i : (x_i, y_i^s) \rightarrow (x_i, y_i^e)$. After extracting all of the horizontal and vertical lines, we arrange them in ascending order based on their y-component for horizontal lines and x-component for vertical lines. It should be noticed that the numeration of h_is is started from top to bottom of the box and v_is are ordered from left to right. The sequences containing all ordered lines (horizontal and vertical) are denoted by H and V, respectively, and can be expressed as follows:

$$\begin{cases} H = \{h_1, h_2, ...h_{n_H}\} & y_i \leq y_j \quad \text{if} \quad i \leq j \\ V = \{v_1, v_2, ...v_{n_V}\} & x_i \leq x_j \quad \text{if} \quad i \leq j, \end{cases} \tag{1}$$

where n_H is the number of all horizontal lines and n_V is the number of vertical lines. The collection of all boxes and each box's vertical and horizontal lines is called a word layout signature (WLS). Now, to make all explanations more clear, consider the word Semnan in Figure 1, which is the name of a city in Iran.

The bounding box representation of Semnan is shown in Figure 2, and after removing the letters, the word layout boxes and horizontal and vertical layout lines of it are demonstrated in Figures 3a, 3b, and 3c, respectively.

In comparison to the other methods for keyword spotting, our method does not depend on the font size, font face, or handwritten types of the word and the signatures that are extracted are almost equal for different

Figure 1. Farsi machine-printed word Semnan.

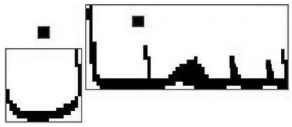

Figure 2. The bounding box representation of handwritten Farsi word Semnan.

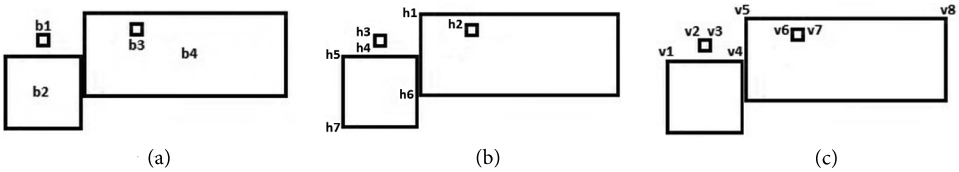

Figure 3. Word layout b_i (a), horizontal layout lines h_i (b), vertical layout lines v_i (c).

types and sizes of handwriting. To illustrate this, we demonstrate four different types of handwriting of the word Persian in Figure 4.

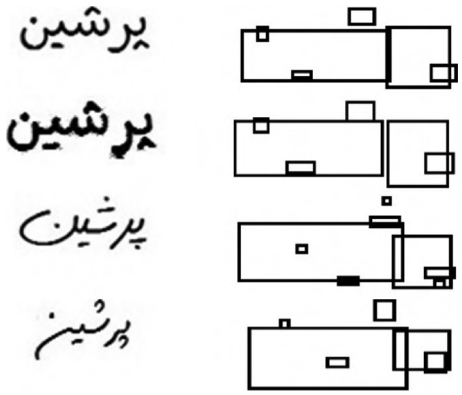

Figure 4. Word layout signatures of Persian.

2.2. Feature extraction

Before starting the following section, first of all, the concept of a grid must be defined. To do so, by considering the groups of horizontal and vertical lines in Eq. (1), Grid(H,V) is obtained as a simple operation, the combination of lines done on the WLS. In Figures 5a and 5b, the grid and cells of the word in Figure 2 are shown, respectively.

Now it is the time to construct descriptors for Farsi document image retrieval. The mentioned descriptor contains 21 components that will be clarified later on. To evaluate its entries, first and foremost, we consider

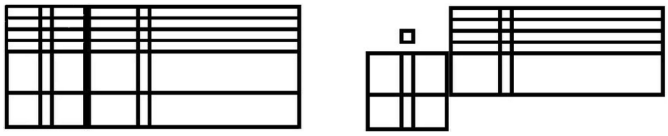

Figure 5. The grid (a) and cells (b) of the word layout of Farsi word Semnan.

the following functions as in [28]:

$$B_k = Intersec\big(b_k, Grid(H,V)\big)(i,j) = \begin{cases} 1 & \text{if } b_k \cap Grid(H,V) \neq \emptyset \\ 0 & \text{if } b_k \cap Grid(H,V) = \emptyset \end{cases} \tag{2}$$

and

$$segment\big(b_k, Grid(H,V)\big) = \sum_{i=1}^{n_H} \sum_{j=1}^{n_V} Intersec\big(b_k, Grid(H,V)\big)(i,j), \tag{3}$$

where $i = \{1, ..., n_H\}$, $j = \{1, ..., n_V\}$, $k = \{1, ..., n_B\}$, and $b_k \in B$.

Eq. (2) produces an $n_H \times n_V$ matrix B_k for any box or cell in word layout signature and the segment of each box is defined as the summation of entries of its matrix, which will be appended to the end of descriptors.

According to relations above, the structure of feature vectors can be constructed as shown in Table 1.

Table 1. The structure of feature vectors.

Position	Feature
1st position	n_H
2nd position	n_V
3rd position	n_B
4th–21st positions	$segment\big(b_k, Grid(H,V)\big)$

Let us find these remaining descriptors of the word shown in Figure 3. As can be seen, there are 7 horizontal lines and 8 vertical lines. In this case, we have four cells, which means that there must be four 7×8 matrices to represent the intersection of any cell with $Grid(H,V)$. Now, using Eq. (2), we obtain the matrices below:

$$B_1 = \begin{bmatrix} 0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0 \\ 0&1&1&0&0&0&0&0 \\ 0&1&1&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0 \end{bmatrix}, \quad B_2 = \begin{bmatrix} 0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0 \\ 1&1&1&1&0&0&0&0 \\ 1&1&1&1&0&0&0&0 \\ 1&1&1&1&0&0&0&0 \end{bmatrix}, \quad B_3 = \begin{bmatrix} 0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&1&1&0 \\ 0&0&0&0&0&1&1&0 \\ 0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0 \end{bmatrix},$$

$$B_4 = \begin{bmatrix} 0&0&0&0&1&1&1&1 \\ 0&0&0&0&1&1&1&1 \\ 0&0&0&0&1&1&1&1 \\ 0&0&0&0&1&1&1&1 \\ 0&0&0&0&1&1&1&1 \\ 0&0&0&0&1&1&1&1 \\ 0&0&0&0&0&0&0&0 \end{bmatrix}.$$

Eq. (3) implies that the segments of each box can be calculated by counting the number of the 1s in each matrix (i.e. $b_1 = 4$, $b_2 = 12$, $b_3 = 4$, and $b_4 = 24$). Finally, the feature vector is formed after concatenating these segments according to the order in Table 1:

$$i.e. < 7, 8, 4, 4, 12, 4, 24, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0 > .$$

The length of the feature vector is set to 18 in our database, which means that the largest word has 18 bounding boxes. If the number of cells is less than 18, the remaining components of the segment will be zero. The sensitivity of this type of representation to font size and font face is very low. To show this, we compare two different font faces, Nazanin and Roya, with sizes 14, 16, and 18 for the Farsi words Semnan in Figure 1 and Ahraz in Figure 6.

Figure 6. Farsi machine-printed word Ahraz.

The results are shown in Figure 7, which shows that the feature vectors are very close to each other.

Font/ Size	Feature Vector	Bounding Box
Nazanin/ 14	<6,10,5,4,12,10,6,10,0, 0,0,0,0,0,0,0,0,0,0,0,0>	
Roya/ 14	<7,10,5,4,12,10,10,10,0, 0,0,0,0,0,0,0,0,0,0,0,0>	
Nazanin/ 16	<6,10,5,4,12,10,6,10,0, 0,0,0,0,0,0,0,0,0,0,0,0>	
Roya/ 16	<7,10,5,4,12,10,10,10,0, 0,0,0,0,0,0,0,0,0,0,0,0>	
Nazanin/ 18	<6,10,5,4,12,10,6,10,0, 0,0,0,0,0,0,0,0,0,0,0,0>	
Roya/ 18	<7,10,5,4,12,10,10,10,0, 0,0,0,0,0,0,0,0,0,0,0,0>	

(a)

Font/ Size	Feature Vector	Bounding Box
Nazanin/ 14	<8,8,4,6,12,6,28,0,0,0, 0,0,0,0,0,0,0,0,0,0>	
Roya/ 14	<8,8,4,6,12,6,28,0,0,0, 0,0,0,0,0,0,0,0,0,0>	
Nazanin/ 16	<8,8,4,6,12,6,28,0,0,0, 0,0,0,0,0,0,0,0,0,0>	
Roya/ 16	<8,8,4,6,16,6,28,0,0,0, 0,0,0,0,0,0,0,0,0,0>	
Nazanin/ 18	<8,8,4,6,16,6,28,0,0,0, 0,0,0,0,0,0,0,0,0,0>	
Roya/ 18	<8,8,4,6,16,6,28,0,0,0, 0,0,0,0,0,0,0,0,0,0>	

(b)

Figure 7. Feature vectors of the words Ahraz (a) and Semnan (b) for some Farsi fonts with font sizes 14, 16, and 18.

In order to improve the accuracy of the results, not only is the letter layout method used, but also topological features such as dots, holes, ascenders, and descenders are applied. In the Farsi alphabet, the number of letters with at least one dot, hole, descender, or ascender are 17, 10, 18, and 6 (out of 32), respectively. This implies that these features are very useful to increase the precision rate and these play an important role in distinguishing the correct instances among the selected phrases from the document images. By applying these features, the number of components in descriptors will increase by four, which are located at the end of the descriptors, and also these yield an increase in precision without affecting recall significantly. This causes $F_{measure}$ to increase. The mentioned features are shown in Figure 8.

2.3. Measure for word spotting

The process presented in this section helps us to find a word image that we are looking for in our dataset. Once the descriptors for each word are extracted, we can continue to task spotting, the search for the existence of keyword occurrences in a document image. The procedure of detecting a query word image among a huge number of documents is depicted in Figure 9.

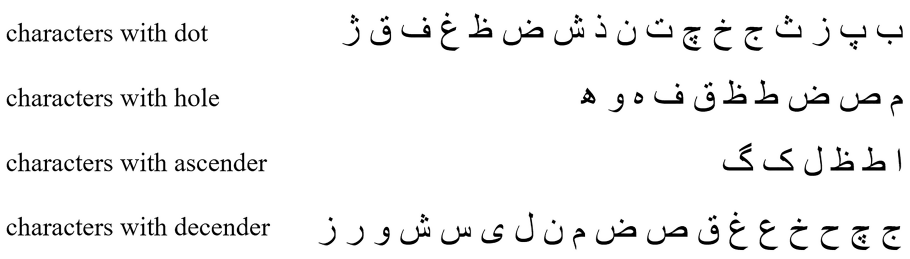

Figure 8. Farsi letters with topological features.

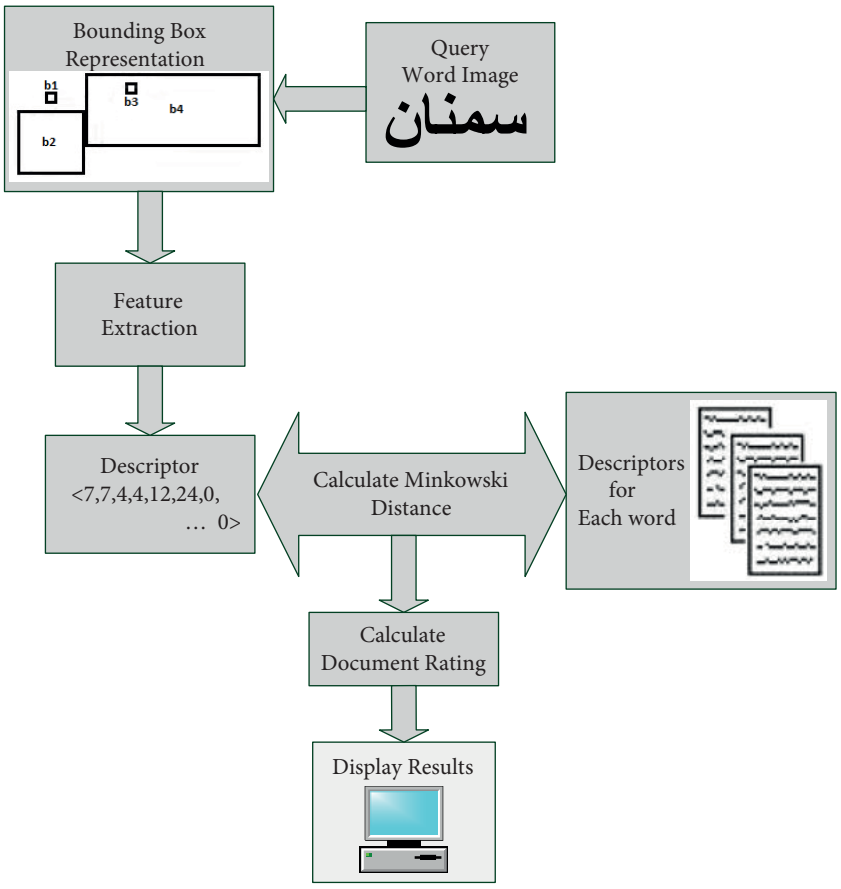

Figure 9. The block diagram of the matching process using WLS.

As can be seen, the comparison process recognizes the word images involved in documents that have a lot of similarities to the extracted feature. At first, the descriptors of query word images are created. Then they are compared to the feature vectors of all documents in the dataset. This comparison is done by calculating the Minskowski distance (MD) between query descriptors (Q_k) and descriptors of each word in the documents as

shown in Eq. (4):

$$MD(i) = \sum_{k=1}^{21} |Q_k - W_k(i)|, \tag{4}$$

where $MD(i)$ denotes the Minskowski distance between Q_k and the ith word image in the dataset, which is shown by $W_k(i)$. Then, the similarity rate, which is in the range of 0 to 100, is calculated for each word image as in Eq. (5):

$$R_i = 100(1 - \frac{MD(i)}{max(MD)}). \tag{5}$$

Here, $max(MD)$ equals the maximum distance $MD(i)$ found in our dataset. Clearly, if $\frac{MD(i)}{max(MD)}$ approaches 1, the similarity rate will decrease, which displays that the result is farther away from our target.

3. Experimental results

The experiments are carried out on a 2.6 GHz dual core processor with 4 GByte RAM PC. To implement the system explained in this paper, two datasets are considered. The first dataset is used to evaluate the execution time for the proposed spotting system as opposed to the Farsi commercial OCR[1] system. This dataset contains 10 full pages of handwritten Farsi text. Figure 10 is a part of a page of dataset 1.

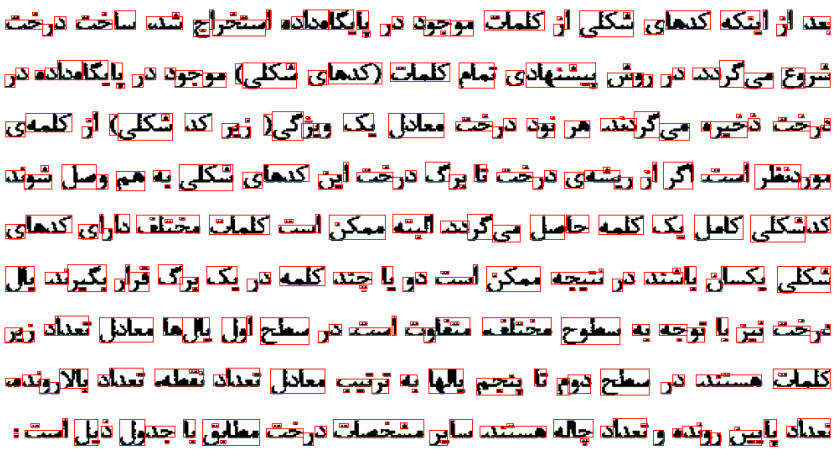

Figure 10. A part of one page used for dataset generation.

We know that OCR means the recognition of printed or written texts by computer through scanning of the text character by character and analyzing it. We compare the performance of the OCR system for dataset1 with our proposed system, and the results are shown in Table 2.

As can be seen from the table, our method is three times faster to spot a page.

In another experiment, a second dataset is used to determine the efficiency of the system by using WLS extraction. This dataset includes 19,582 unique word images, which are selected from the dataset used in [34], which contains 2.6 million words (this corpus is also known by its author's name, Bijankhan). Most of these words are numbers, prepositions, and conjunctives; therefore, we eliminated most of the the unnecessary words to get 19,582 word images. There are 502 Farsi printed words, which are combinations of different types of

[1] http://farsiocr.ir

Table 2. Average elapsed time for total recognition for one and ten pages of Farsi text.

	Average elapsed time for	
	one page	10 text pages
Commercial Farsi OCR	6.9 s	68.152 s
Proposed method (page segmentation + feature extraction using WLS)	(1.56 s+0.64 s)= 2.2 s	(14.215 s+7.190 s)= 21.405 s

handwritten and machine-printed words with variant sizes and font faces. Each word is repeated at least 40 times. The lengths of words vary between 4 and 12 characters. The proposed system is tested in two modes: considering topological features and not considering them.

To investigate the efficiency of the method, we have to check the ability of the system to find the relevant query image within a dataset. To do so, three criteria [19], precision, recall, and $F_{measure}$, are calculated as shown in Eqs. (6), (7), and (8), respectively.

$$P = \frac{\text{Number of correctly identified images}}{\text{Number of correctly identified images + Number of identified images that were not correct}}, \tag{6}$$

$$R = \frac{\text{Number of correctly identified images}}{\text{Number of correctly identified images + Number of rejected images that were correct}}, \tag{7}$$

$$F_{measure} = \frac{2PR}{R+P}. \tag{8}$$

In our implementation, we evaluate threshold δ as the similarity measure from 0.8 to 1 in increments of 0.05 and the documents with similarity rate more that δ will be shown to the user as the result. Tables 3 and 4 below summarize the average precision, recall, and $F_{measure}$ rate of the words in dataset2 using different threshold δ values by not considering topological features and by considering them.

Table 3. Average precision, recall, and $F_{measure}$ for different δ values, without applying topological features.

δ	**0.8**	**0.85**	**0.9**	**0.95**	**1**
Precision	0.763	0.824	0.892	0.923	0.941
Recall	1	0.979	0.953	0.863	0.753
$F_{measure}$	0.866	0.895	0.921	0.892	0.837

Table 4. Average precision, recall, and $F_{measure}$ for different δ values, with topological features applied.

δ	**0.8**	**0.85**	**0.9**	**0.95**	**1**
Precision	0.832	0.874	0.943	0.931	0.972
Recall	1	0.987	0.981	0.864	0.753
$F_{measure}$	0.908	0.927	0.962	0.896	0.849

Clearly, as precision rate increases, recall rate decreases, and vice versa. Therefore, we have to maximize either recall or precision to construct a balanced recognition system. For this reason, we should use $F_{measure}$,

which is the harmonic mean of recall and precision. Putting the parts together, as δ decreases, the recall increases since the system will identify more descriptors and the images close to the query. However, while recall increases, the precision decreases since not only are true positive results (the number of correctly identified images) increasing, but also false positive results (the number of identified images that were not correct) will increase. It is noticeable that the average $F_{measure}$, which shows the efficiency of our method, takes its highest value, which is 0.962, at $\delta = 0.9$. One can deduce that to achieve the highest performance, δ must be considered as 0.9.

Lastly, let us have a quick look at the other methods used in [19, 20] and their efficiency rate, and then compare them to our proposed method. These results are shown in Table 5.

Table 5. Best precision, recall, and $F_{measure}$ rates for some similar works and our proposed method.

	Precision	Recall	$F_{measure}$
Method in [19]	0.975	0.921	0.947
Method in [20]	0.923	0.840	0.879
Proposed system	0.943	0.981	0.962

The highest performance of the system defined in [19] is at $\delta = 0.85$ with precision of 0.975 and recall of 0.921, and consequently, $F_{measure}$ is found as 0.947. The results in [20] are 0.923, 0.84, and 0.88, respectively. In comparison to the other methods, among all of the Farsi spotting systems, our approach based on WLS shows the highest performance in terms of recall and $F_{measure}$ and it has less complexity during the spotting process.

4. Conclusion

Although printed documents and handwritten ones are different from each other, this difference is not viewable in some cases. Our word spotting system is not vulnerable since this method is accomplished for the general Farsi printed word images, whether they are handwritten or machine-printed. Using the proposed method, a new descriptor is defined for each word, which does not depend on the font face and size. Therefore, the time required to recognize the font size is eliminated, which causes the system to be faster. In this method, when a keyword is entered to be spotted in the document without considering its font face and its size, the descriptor of that word will be created. By using the Minskowski distance, the spotted word will be compared with the feature vector of all the words in the dataset. Then the word with less distance and the highest similarity rate will be shown to the user. We have found that using topological features assists in increasing the efficiency rate.

On a final note, the results of implementations confirm that this document image retrieval system is faster and higher in performance than those already in use.

References

[1] Kameshiro T, Hirano T, Okada Y, Yoda F. A document image retrieval method tolerating recognition and segmentation errors of optical character recognition using shape feature and multiple candidates. In: IEEE 1999 International Conference on Document Analysis and Recognition; Bangalore, India; 1999. pp. 681–684. doi: 10.1109/ICDAR.1999.791879

[2] Mehran R, Pirsiavash H, Razzazi F. A front-end optical character recognition for omni-font Persian/Arabic cursive printed documents. In: IEEE 2005 Digital Image Computing, Techniques and Applications; Queensland, Australia; 2005. pp. 385–392. doi: 10.1109/DICTA.2005.3

[3] Mozaffari S, Faez K, Margner V, El-Abed H. Lexicon reduction using dots for off-line Farsi/Arabic handwritten word recognition. Pattern Recognition Letters 2008; 29 (6): 724-734. doi: 10.1016/j.patrec.2007.11.009

[4] Doermann D. Indexing and retrieval of document images: a survey. Computer Vision and Image Understanding 1998; 70 (3): 287–298. doi: 10.1006/cviu.1998.0692

[5] Myers CS, Rabiner LR, Rosenberg AE. On the use of dynamic time warping for word spotting and connected word recognition. Bell Lab System Technical Journal 1981; 60 (3): 303–325. doi: 10.1002/j.1538-7305.1981.tb00243.x

[6] Rose RC, Paul DB. A hidden Markov model based keyword recognition system. In: IEEE 1990 International Conference on Acoustics, Speech and Signal Processing; Albuquerque, NM, USA; 1990. pp. 129–132. doi: 10.1109/ICASSP.1990.115555

[7] Knill KM, Young SJ. Speaker dependent keyword spotting for accessing stored speech. Cambridge, UK: Cambridge University Engineering Department, 1994.

[8] Kuo SS, Agazzi OE. Keyword spotting in poorly printed documents using pseudo 2-D hidden markov models. IEEE Transactions on Pattern Analysis Machine Intelligence 1994; 16 (8): 842–848. doi: 10.1109/34.308482

[9] Cho BJ, Kim JH. Print keyword spotting with dynamically synthesized pseudo 2D HMMs. Pattern Recognition Letter 2004; 25 (9): 999–1011. doi: 10.1016/j.patrec.2004.02.014

[10] Chen FR, Wilcox LD, Bloomberg DS. Word spotting in scanned images using hidden markov models. In: IEEE 1993 International Conference on Acoustics, Speech and Signal Processing; Minneapolis, MN, USA; 1993. pp. 1–4. doi: 10.1109/ICASSP.1993.319732

[11] Murugappan A, Ramachandran B, Dhavachelvan P. A survey of keyword spotting techniques for printed document images. Artificial Intelligence Review 2011; 35 (2): 119-136. doi: 10.1007/s10462-010-9187-5

[12] Giotis AP, Sfikas G, Gatos B, Nikou C. A survey of document image word spotting techniques. Pattern Recognition 2017; 68: 310-332. doi: 10.1016/j.patcog.2017.02.023

[13] Kim SH, Park SC, Jeong CB, Kim JS, Park HR et al. Keyword spotting on Korean document images by matching the keyword image. In: ICADL 2005 International Conference on Asian Digital Libraries, Implementing Strategies and Sharing Experiences; Bangkok, Thailand; 2005. pp. 158–166.

[14] Lu Y, Tan CL. Chinese word searching in imaged documents. International Journal of Pattern Recognition and Artificial Intelligence 2004; 18 (2): 229–246. doi: 10.1142/S0218001404003137

[15] Srihari SN, Srinivasan H, Huang C, Shetty S. Spotting words in Latin, Devanagari and Arabic scripts. Indian Journal of Artificial Intelligence 2006; 16: 2–9.

[16] Srihari SN, Srinivasan H, Babu P, Bhole C. Spotting words in handwritten Arabic documents. In: SPIE 6067 Document Recognition and Retrieval XIII; San Jose, CA, USA; 2006. pp. 606702-1–606702-12. doi:10.1117/12.643107

[17] Srihari SN, Srinivasan H, Babu P, Bhole C. Handwritten Arabic word spotting using the CEDARABIC document analysis system. In: SDIUT 2005 Proceedings of Symposium on Document Image Understanding Technology; Collage Park, MD, USA; 2005. pp. 123–132.

[18] Pourasad Y, Hassibi H, Ghorbani A. Farsi word spotting and font size recognition. Procedia Technology 2012; 1: 372–377. doi: 10.1016/j.protcy.2012.02.077

[19] Pourasad Y, Hassibi H, Ghorbani A. A word spotting method for Farsi machine-printed document images. Turkish Journal of Electrical Engineering and Computer Sciences 2013; 21 (3): 734-746. doi: 10.3906/elk-1107-26

[20] Pourasad Y, Hassibi H, Ghorbani A. A Farsi/Arabic word spotting approach for printed document images. International Journal of Natural and Engineering Sciences 2012; 6 (1): 15-18.

[21] Ebrahimi A, Kabir E. A pictorial dictionary for printed Farsi sub words. Pattern Recognition Letters 2008; 29 (5): 656-663. doi: 10.1016/j.patrec.2007.11.008

[22] Akbari M, Azmi R. Document image database indexing with pictorial dictionary. In: SPIE 7546 International Conference on Digital Image Processing, International Society for Optics and Photonics; Singapore, Singapore 2010. p. 75462R. doi: 10.1117/12.856302

[23] Erlandson EJ, Trenkle JM, Vogt RC. Word-level recognition of multifont Arabic text using a feature-vector matching approach. In: SPIE 2660 Proceedings of the International Society for Optical Engineers; San Jose, SFO, USA; 1996. p. 2660-08. doi: 10.1117/12.234725

[24] Miri E, Razavi SM, Mehrshad N. Search space reduction in printed Persian sub word recognition by a heretical method. Indian Journal of Science and Technology 2017; 10 (9): 17485. doi: 10.17485/ijst/2017/v10i9/110158

[25] Lu S, Li L, Tan CL. Document image retrieval through word shape coding. IEEE Transactions on Pattern Analysis and Machine Intelligence 2008; 30 (11): 1913-1918. doi: 10.1109/TPAMI.2008.89

[26] Zagoris K, Ergina K, Papamarkos N. A document image retrieval system. Engineering Applications of Artificial Intelligence 2010; 23 (6): 872-879. doi: 10.1016/j.engappai.2010.03.002

[27] Keyvanpour M, Tavoli R. Feature weighting for improving document image retrieval system performance. International Journal of Computer Science Issues 2012; 9 (3): 125-130.

[28] Naveen, Guru DS. Retrieval of document images based on page layout similarity. Lect Notes Comp Sci 2007; 4398: 136-148. doi: 10.1007/978-3-540-71545-0_11

[29] Liu J, Jain AK. Image-based form document retrieval. In: IEEE 1998 International Conference on Pattern Recognition; Brisbane, Australia; 1998. pp. 503-513. doi: 10.1109/ICPR.1998.711221

[30] Das AK, Chanda B. A fast algorithm for skew detection of document images using morphology. International Journal on Document Analysis and Recognition 2001; 4 (2): 109-114. doi: 10.1007/PL00010902

[31] Guru DS, Punitha P, Mahesh S. Skew estimation in digitized documents: a novel approach. In: ICVGIP 2004 Indian Conference on Computer Vision, Graphics and Image Processing; Kolkata, India; 2004. pp. 314-319.

[32] Das AK, Saha SK, Chanda B. An empirical measure of the performance of a document image segmentation algorithm. International Journal on Document Analysis and Recognition 2002; 4 (3): 183-190. doi: 10.1007/s100320100

[33] Jain AK, Yu B. Document representation and its application to page decomposition. IEEE Transactions on Pattern Analysis and Machine Intelligence 1998; 20 (3): 294-308. doi: 10.1109/34.667886

[34] Isapour S, Homayounpour M, Bijabkhan M. The prediction of Ezafe construction in Persian by using probabilistic context free grammar. In: CSICC 2008 International Conference of Computer Society of Iran; Kish Island, Iran; 2008. p. ACCSI13-100.