IJICTR International Journal of Information & Communication Technology Research ITRC

Volume 7- Number 2- Spring 2015 (59-70)

HWS: A Hierarchical Word Spotting Method for Farsi Printed Words Through Word Shape Coding

Mohammadreza Keyvanpour
Department of Computer engineering, Alzahra University
Tehran, Iran
keyvanpour@alzahra.ac.ir

Reza Tavoli
Departments of Computer Engineering, Islamic Azad University, Qazvin Branch
Qazvin, Iran
tavoli@qiau.ac.ir

Saeed Mozaffari
Electrical and Computer Department, Semnan University
Semnan, Iran
mozaffari@semnan.ac.ir

Received: May 25, 2014-Accepted: February 16, 2015

Abstract—**Word shape coding (WSC) is a method of document image retrieval (DIR) based on keyword spotting. By using this method, a word can be recognized in the document image, only by identifying some of the features of the word. In this paper, a hierarchical word spotting method, namely HWS, is presented for Farsi document image retrieval through WSC. In HWS method, document images are retrieved by using a new indexing method. In HWS, at first the words in the document images are shape coded based on topological properties. These features include number of sub-words, ascenders, descenders, and holes.A new feature that has been used for this paper is dot's position in word. Six features are obtained which are one top dot, two top dots, three top dots and one bottom dot, two bottom dots, and three bottom dots. Precision of retrieval increases by using these features. Then, all of the shape codes are indexed by building a tree. Retrieval is done based on keyword query in the tree. The results show that the proposed technique is very fast for large volumes of documents. Time complexity for successful and non-successful searching is $O(\log_k^n)$ .This value is better than values in ordinal method. Also, time complexity for indexing is $O(\log_k^n)$. The HWS method is tested on Bijankhan database. 87867 common words from this database are used for building the dictionary. Test results show that average of precision is 0.83 and average recall is 0.94.**

Keywords-Tree indexing;Information Retrieval;Document Image;word shape coding; Farsi document.

I. Introduction

Nowadays, with growing digital libraries and paper documents in offices, management and organization of paper documents takes a long time. These problems will be increased when a special document is needed to be searched in a huge volume of documents. So in order to make paperless offices, paper documents must be scanned and archived. Therefore, after scanning and archiving documents, methods of searching a specific document among other documents are needed. This is called Document Image Retrieval. There are many methods of document image retrieval. DOERMANN, in 1998, has done a case study on the methods of document image retrieval [1]. Keyvanpour Mohammad Reza and Tavoli Reza have presented a framework for classification and evaluation of document image retrieval and indexing methods comprehensively [2]. In this framework the methods of document image retrieval are divided into two main categories: methods based on Text components and methods based on Non-text components. Non-text component-based approach includes Signature-based [3, 4], logo-based [5] and screen layout-based methods [6]. Text component-based approach consists of two different groups: keyword spotting method and optical character Recognition (OCR). In OCR, at first, images of scanned documents are converted to ASCII text. Then retrieval is done according to traditional methods [2, 23, 24]. In traditional retrieval methods tokenization, stemming and removal of stop words have been used. Then, weighting of terms has been done [1]. OCR method is not a suitable method in the case of huge volumes of documents. This method has abundant weaknesses especially for Farsi/Arabic language. So keyword spotting methods are presented bypassing OCR without any documentary full conversion. Shape coding method in keyword spotting methods is divided into two main categories:

In the first group, a word is broken into letters and WSC is performed for any letters. This is called CSC method or Analytical Strategies [7]. In the second group, indexing and coding are done according to the general form of the word or sub-word; this is called WSC Method or Holistic Strategies [7]. In this paper, a new indexing method for Farsi/Arabic document image retrieval based on WSC is presented. In the proposed method, at first the words existing in the images of documents are coded according to the features of shape topologies. These features contain number of sub-words, dots, ascenders, descenders and holes. In previous methods, just the number of dots was used as a feature, but in the proposed method the position of dot in the word is also considered. In Farsi language, most of the words have one, two or three dots above or under the letters. By using this method, accuracy will be increased significantly. Another problem in aforementioned methods is that searching to find a shaped code is sequentially which takes a long time in a huge volume of documents and searching will be slowed. In this method, extracted shaped codes are indexed in a tree. The speed of retrieval increases considerably by using this method, because in this retrieval instead of searching the whole database for finding a word, only a tree is searched. This method is used in a huge volume of document images and its efficiency is much more precious. The paper is organized as follows. In the next section, the related works are investigated. In section 3, the proposed method is presented. In section 4, the time complexity of proposed method is described. In section 5, the results of the test are presented. Finally, section 6 concludes the paper.

II. RELATED WORKS

In recent years, lots of works have been done on the retrieval of document images in Farsi, Arabic, English and Chinese languages. Some of the important works are described in the following. Ebrahimi and Kabir presented a method for retrieval of Farsi document images that is based on the whole shape of words and sub-words [8, 9]. In this method, PCA is used for compacting feature vectors. Then K-MEANS is used for clustering of sub- words and the average of each cluster is placed in one pictorial dictionary. Akbari and Azmi have presented a method for retrieval of Farsi document images that is independent of recognition [10]. In this method, the upper contours of words are extracted and then a picture dictionary is made of these features. Pourasad Yaghoub at el. have presented a method for coding and retrieval of Farsi document images. They also presented a method for font detection in Farsi texts that is based on tiny connected components [9]. Erlandson et al have used the shape of printed words as features in the recognition of Arabic texts written by three common fonts [11]. Several features such as dots, tittles, the line segment, endpoints and connectors, holes, inner word space and descenders of the Arabic printed words are extracted and saved in a dictionary. Dehghan and Faez have used a hidden Markov model for recognition of handwritten names of cites of Iran [12]. A several works are also done in English language and some of them are examined in this paper. Shijian Lu at el have presented a method for retrieval of English document images that is based on WSC [13]. In this method, they have used topological features such as character holes, ascenders/ descenders and character reservoirs. In this method, documents can be retrieved by WSC both based on the query of keyword and based on the query of the document image. The advantage of WSC over character encoding is that it has higher accuracy in low quality documents and doesn't have the letters' segmentation error. Similarly, several methods in WSC are mentioned in [13] that use some predefined codes. Zagoris et al have used a document image retrieval system which is based on word recognition without considering OCR [14]. This system is divided into two phases: online and offline. Indexing operations are done in offline system and retrieval operations are used in 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 Mohammad Reza and Tavoli Reza have used a feature weighting method to improve this system. This method is based on the correlation of features. By using this method, precision and recall rate is increased in the document image retrieval system [15].

III. PROPOSED METHOD

Block diagram of the proposed method is shown in Fig. 1. The structure consists of two phases: online and offline. Indexing operations are done in offline phase. Retrieval operations are done in online phase.

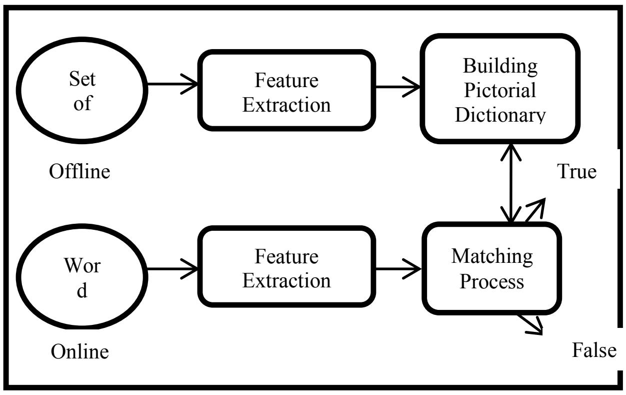
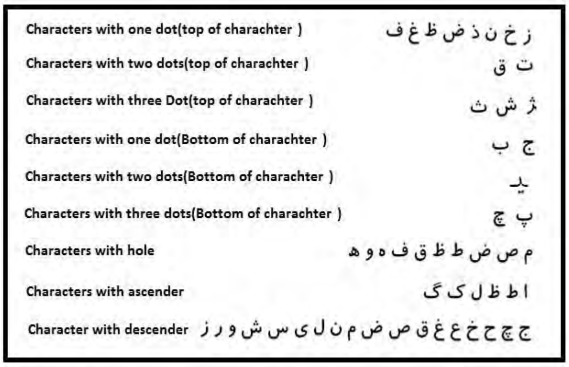


Fig. 1. Block Diagram of Proposed Method

A. Feature extraction-Word shape coding(online & offline)

This activity is same in the offline and online phases. Most of retrieval methods are divided into two groups. In the first group, word is segmented into its characters and then features are extracted from the image of each character. These features create a word descriptor. This method is called CSC. The second group uses the global shape of words or sub-words. This method is called the WSC. Advantages and disadvantages of these groups are illustrated in [16]. In this paper, the WSC is used. The proposed system is based on four feature sets that are extracted from every word. The feature sets are:



Characters with one dot(top of charachter)	ز خ ن ذ ض ظ غ ف
Characters with two dots(top of charachter)	ت ق
Characters with three Dot(top of charachter)	ژ ش ث
Characters with one dot(Bottom of charachter)	ج ب
Characters with two dots(Bottom of charachter)	يـ
Characters with three dots(Bottom of charachter)	پ چ
Characters with hole	م ص ض ط ظ ق ف ه و ھ
Characters with ascender	ا ط ظ ل ک گ
Character with descender	ج چ ح خ ع غ ق ص ض م ن ل ی س ش و ر ز

Fig. 2. Farsi letters; features such as one top dot, two top dots, three top dots, one low dot, two low dots and three low dots, holes, ascenders and descenders.

Generally, approaches based on shape coding are sufficient for document images with average and high quality [25,26] and the proposed method is the same and document without quality like historical documents don't have enough precision. It was said previous, the approaches based on shape coding are divided two categories: CSC and WSC. The methods based on CSC have character segmentation errors [27] and they don't have good accuracy for low quality documents. And the methods based on WSC don't have any character segmentation error and they worked better on the low quality documents. This purposed method is under the second method and features are extracted just from the whole word and have high accuracy than CSC methods.

- **Number of sub-words:** The English words are made of some characters, but in Farsi, each word is made of some sub-words shown in Fig. 3(b). A sub-word is a continued part of a word. In this paper the number of sub-words is a feature. In Farsi/Arabic handwritten script, there is no difference in the within word space (i.e. the white space between the sub-words) [22]. But boundary of words and sub-words in printed document is specified and constant. There are many approaches to detect the location of sub-words and one of them is the vertical projection profile. In a binary text image, often foreground pixels are 0 and background pixels are 1. We first changed the document image to its complement. In order to complement such an image, it is sufficient to change 0's to 1's and 1's to 0's. The vertical projection profile VPP [j] is defined as the number of black pixels that are residing in each column j. if an m × n image is shown with F, its complement is $\widetilde{F}$ [18].

$$VPP[j] = \sum_{i=1}^{m} \widetilde{F}(i,j) \qquad (1)$$

Where m is the number of rows and n is the number of columns in the image. The indexes of the *VPP*, for which *VPP* =0, indicates the location of the separated sub-words (Fig. 3).

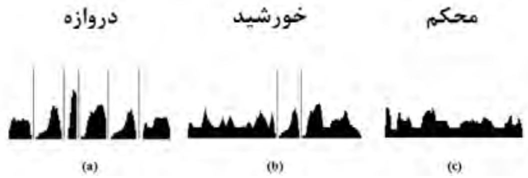


Fig. 3. The detection of sub-words through the analysis of VPP.

- **Number of dots:** Another feature of Farsi language is having dots shown in Fig. 6(c). Most of the Farsi characters (17 out of 32) have one, two or three dots which can be located at the bottom, inside or top of the characters that is shown in Fig. 2. The combination of the number of dots and their position can be used as a feature. Therefore, six features are obtained which are one top dot, two top dots, three top dots and one bottom dot, two bottom dots, and three bottom dots. The method proposed in [17] is used for finding dots. A block diagram of dot extraction algorithm is shown in Fig. 4.

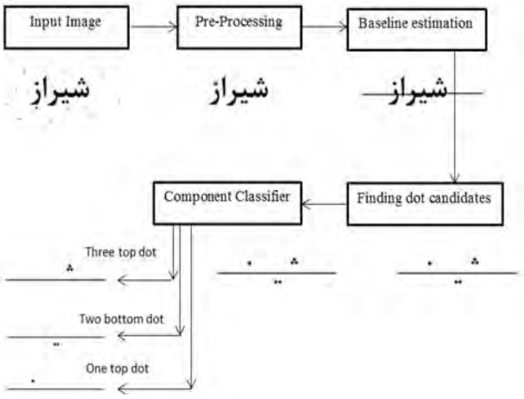


Fig. 4. A Block Diagram of dot extraction algorithm. First row: gray level input image. Second row: binary image after preprocessing. Third row: input image and its baseline (thin line) after baseline estimation. Fourth row: candidate dots

after removing small strokes located on or near the baseline. Fifth row: merging close dots [17].

- **Number of ascenders/ descenders:** ascender is the top part of a letter and descenders is the bottom part of a letter shown in Fig. 6(d) and 6(e). This feature is used in English, Arabic and Farsi languages. Six letters have ascenders in Farsi language and eighteen letters have descenders in Farsi language that is shown in Fig. 2. Ascenders/descenders can be easily obtained by observing the above of the X-line and below the base-line of the text, respectively [13, 9]. Base-line and x-line can be detected by horizontal projection profile that is shown in Fig. 5. The horizontal projection profile HPP [i] is defined as the number of black pixels that are residing in each row i. if an m×n image is shown with *F,* its complement is $\widetilde{F}$ **.**

$$HPP[i] = \sum_{j=1}^{n} \widetilde{F}(i, j) \qquad (2)$$

Where n is the number of columns in the image.

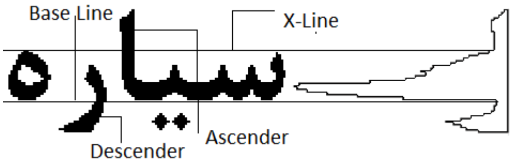


Fig. 5. The detection of ascenders and descenders through the analysis of the horizontal projection profiles and the illustration of the x line and base-line of text.

- **Number of holes:** A closed component of a part of a letter is called a hole. The hole is a significant feature in English, Farsi and Arabic languages. In Farsi about 10 letters have got at least one hole that is shown in Fig. 2. The way of finding the hole is fully-described in [13, 9]. Character holes can be detected based on the closeness of the detected white run components shown in Fig. 6(f). A white run component is closed if all neighboring pixels on the left of the first and on the right of the last constituent white run are text pixels.

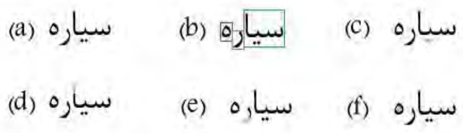


Fig. 6: (a) main picture (b) sub-word (c) dot (d) ascender (e) descenders (f) hole.

After extracting features of each word, a shape code is attributed to each feature. After extracting shape features of a word, they are turned to make shape code of the word. The Fig. 7 indicates feature extracting and shape code creating for all of the database words. Example of WSC for some words is shown in table 1.

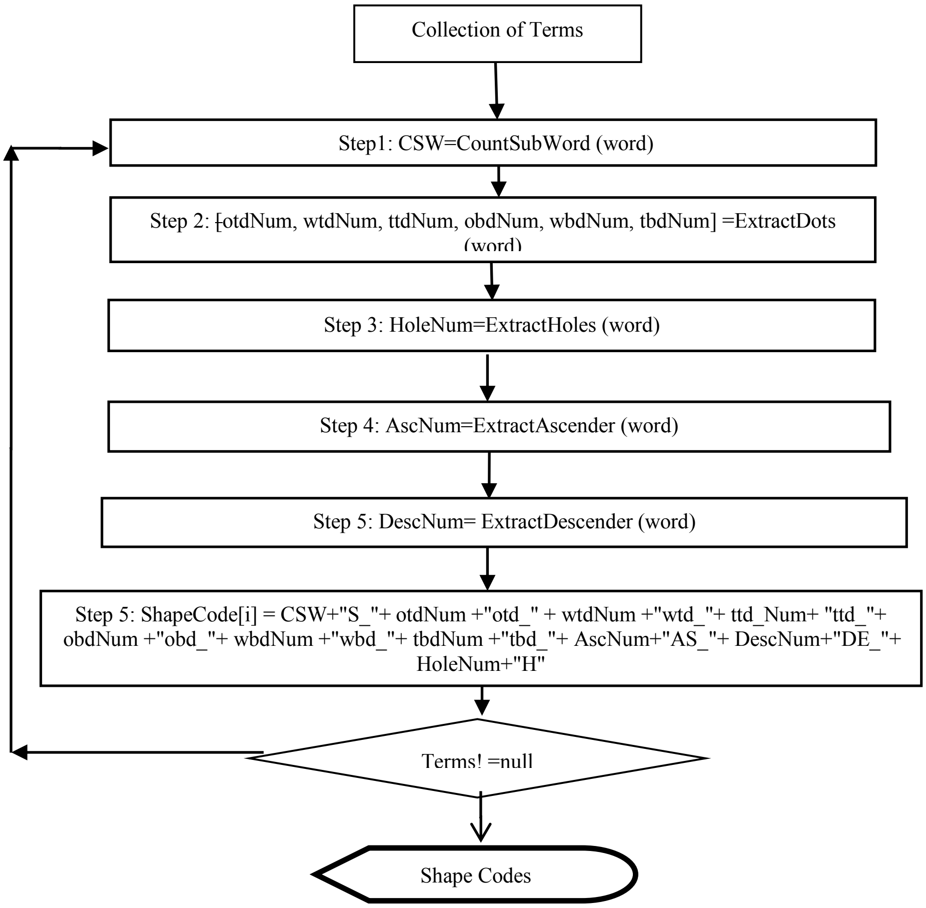

Collection of Terms
Step1: CSW=CountSubWord (word)
Step 2: [otdNum, wtdNum, ttdNum, obdNum, wbdNum, tbdNum] =ExtractDots (word)
Step 3: HoleNum=ExtractHoles (word)
Step 4: AscNum=ExtractAscender (word)
Step 5: DescNum= ExtractDescender (word)
Step 5: ShapeCode[i] = CSW+"S_"+ otdNum +"otd_" + wtdNum +"wtd_"+ ttd_Num+ "ttd_"+ obdNum +"obd_"+ wbdNum +"wbd_"+ tbdNum +"tbd_"+ AscNum+"AS_"+ DescNum+"DE_"+ HoleNum+"H"
Terms! =null
Shape Codes

Fig. 7. Feature extracting and shape code creating flowchart

Table I. Examples of WSC

Words	Shape Codes	Description
اولین	3S_1otd_0wtd_0ttd_0obd_1wbd_0tbd_2AS_2DS_1H	3 Sub-words, 1 two bottom Dot, 1 one top dot ,2 Ascenders, 2 Descenders, 1Hole
سیاره	3S_0otd_0wtd_0ttd_0obd_1wbd_0tbd_1AS_2DS_1H	3 Sub-words, 1 two bottom Dot, 1 Ascenders, 2 Descenders, 1Hole
خارج	3S_1otd_0wtd_0ttd_1obd_0wbd_0tbd_1AS_3DS_0H	3 Sub-words, 1 one top dot, 1 one bottom dot, 1 Ascenders, 3 Descenders, 0Hole
منظومه	2S_2otd_0wtd_0ttd_0obd_0wbd_0tbd_1AS_4DS_5H	2 Sub-words, 2 one top dot, 1 Ascenders, 4 Descenders, 5Holes
شمسی	1S_0otd_0wtd_1ttd_0obd_0wbd_0tbd_0AS_3DS_1H	1 Sub-words,1 three top do t, 0 Ascenders, 1 Descenders, 1Hole
دیده	3S_0otd_0wtd_0ttd_0obd_1wbd_0tbd_0AS_0DS_1H	3 Sub-words, 1 two bottom Dot, 0 Ascenders, 0 Descenders, 1Hole
شد	1S_0otd_0wtd_1ttd_0obd_0wbd_0tbd_0AS_1DS_0H	1 Sub-words, 1 three top dot, 0 Ascenders, 1 Descenders, 0Hole
دوجین	3S_1otd_0wtd_0ttd_1obd_1wbd_0tbd_0AS_3DS_1H	3 Sub-words, 1 two bottom Dot, 1 one top dot, 1 one bottom do t,0 Ascenders,2 Descenders, 1Hole
مدار	3S_0otd_0wtd_0ttd_0obd_0wbd_0tbd_1AS_2DS_1H	3 Sub- words, 0 Dots, 1 Ascenders, 2 Descenders, 1Hole
اطراف	4S_1otd_0wtd_0ttd_0obd_0wbd_0tbd_3AS_1DS_2H	4 Sub-words, 1 one top dot, 3 Ascenders, 1 Descenders, 2Hole

Table II. Tree structure

IS Root	IS Leaf	Feasible values	abbreviation	Feature	Level Number
Yes	No	1S, 2S, 3S, 4S, …	S	Number of Sub- words	First Level
No	No	0otd, 1 otd, 2 otd, …	otd	Number of one top dot	Second Level
No	No	0wtd, 1wtd, 2wtd, …	wtd	Number of two top dots	3th Level
No	No	0ttd, 1ttd, 2ttd, …	ttd	Number of three top dots	4th Level
No	No	0obd, 1 obd, 2 obd, …	obd	Number of one bottom dot	5th Level
No	No	0wbd, 1wbd, 2wbd, …	wbd	Number of two bottom dots	6th Level
No	No	0tbd, 1tbd, 2tbd, …	tbd	Number of three bottom dots	7th Level
No	No	0AS, 1AS, 2AS, …	AS	Number of Ascenders	8th Level
No	No	0DS, 1DS, 2DS, …	DS	Number of Descenders	9th Level
No	Yes	0H, 1H, 2H, …	H	Number of Holes	10th Level

B. *Building Pictorial Dictionary-Tree Indexing of shape codes(Offline)*

In information retrieval, one of the important measures is speed. In the proposed method, pictorial dictionary is built hierarchal. In the following, tree structure and inserting in tree will be described in subsection 3.2.1 and 3.2.2.

1) Tree structure

A tree is a graph without any circle. Each graph is usually defined as the following:

$$G = (V, E) \tag{3}$$

Where, V is a set of the nodes (vertices) and E is a set of edges (links). After extracting shape codes of database words, building the tree is commenced. In the proposed method, all words (shape codes) in database are saved in the tree. Each node of the tree is known as a feature of the target word. The completed shape code of a word is created by linking between the root and leaves of the tree. Sometimes different words may have the same shape code, so two or three words may place in one leaf. The number of the tree's edges is different, considering the different levels. In the first level the number of edges is equal to the number of sub- words. In next levels the numbers of edges is equal to the number of one top dot, two top dots, three top dots, one bottom dot, two bottom dots, three bottom dots, ascenders, descenders and the number of holes sequentially. Other features of the tree are shown in the table 2:

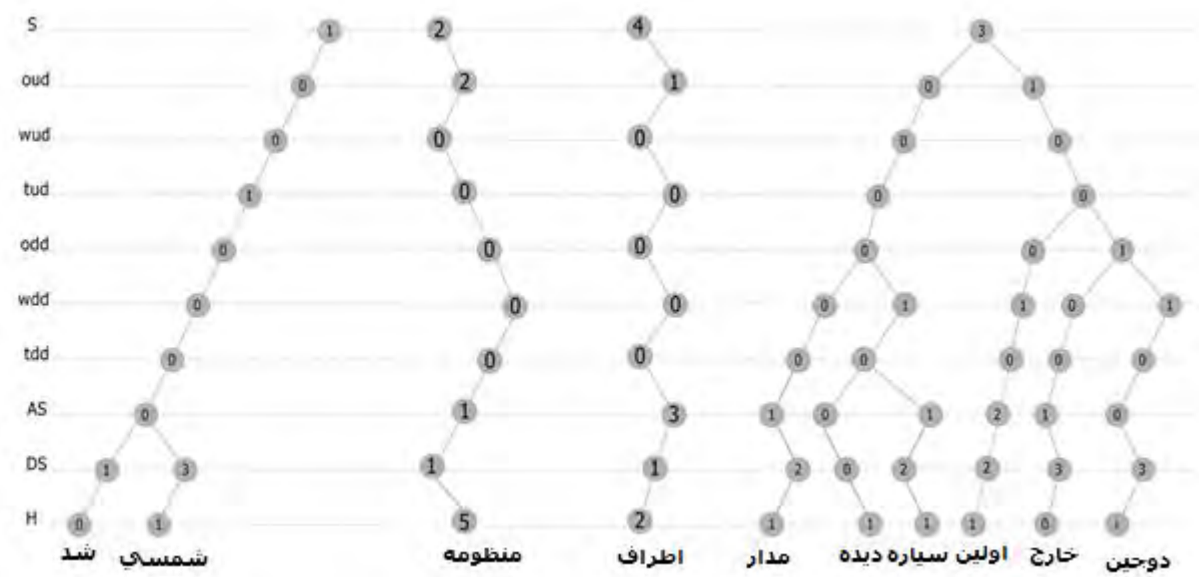


Fig. 8. Indexing of the words in Table 2

2) Inserting in the tree

In the proposed method, building a word is commenced in the root node. At first, the shape code of a word is broken and then building the tree is begun. The first part of a shape code places in tree root (first level). The first level feature is the number of sub- words that is shown in some shape codes such as 1S, 2S, and 3S and etc. The 1S code is equivalent to one sub- word. The 2S, 3S codes are equivalent to two, three sub- words. A tree is created for each of them. The first part of the shape code sits in one of the trees considering its amount. Other parts of the shape code of the target word sit in the same tree. After the first part of the shape codes lied in tree root, it is the second part of shape code's turn. The number of one top dot is the second part of the shape code that lies in second level of the tree. The 1td, 2td, 3td shape codes are equivalent to one, two and three top dots respectively. Similar works will be done in other levels of the shape code from third level to the last one. The only difference between these levels and previous levels is the desired feature. The indexing of words in table 2 is shown in the Fig. 8:

According to Fig. 8, number of sub- words is inserted in the first level that is shown by "S" and number "1" on the node represents a (means one) sub-word. Other features are inserted in the second level and next levels to the last one.

C. *Matching Process – retrieval (Online)*

At first user enters the intended query. Then the shape code of the query word is created and the shape code will be broken into its components. In the proposed new method, searching for a word is commenced from the root. The first part of query's shape code is compared to the tree roots. If the target shape code is not equal to any of tree roots, it won't be available in any of desired trees and the search will be unsuccessful and should be stopped. Otherwise the search will be continued and the second part of query's shape code is searched in the second level of the tree and so on. In each level the intended part of the shape code of the query entered by user is compared to all of the nods of that level. If it is not equal to any of nodes, the search will be unsuccessful and should be stopped. But if it is equal to one of nodes, the search will be stopped in that level and search enters the next level and the rest of search is done for children of successful nods; in fact that part of the tree is pruned. The search is continued in this way until the last level. If the search is continued to tree leaves, the search will be successful and desired word will be found. Otherwise, if the search is finished in each level, it will not be successful. According to above description, searching of word   is illustrated in Fig. 9.

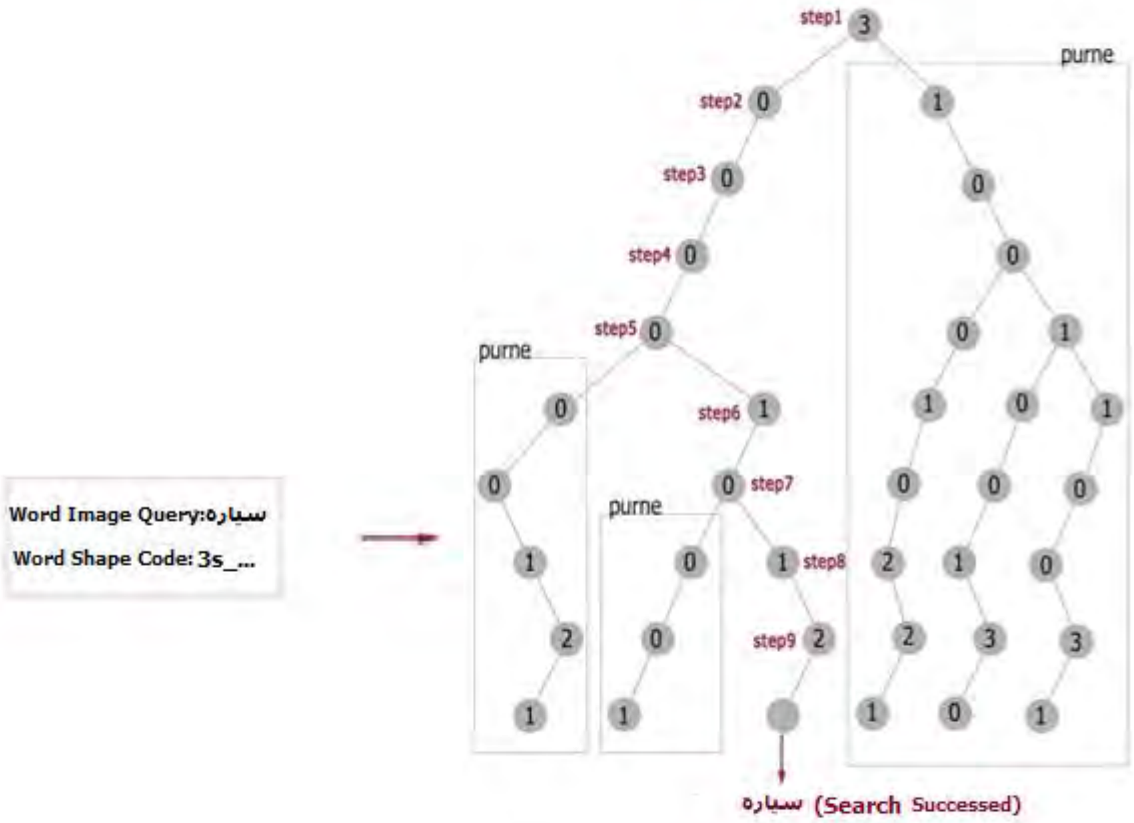


Fig. 9. Searching "سیاره" in a tree

IV. Time Complexity

A. Time complexity of searching

Table 3 displays time complexity of a successful search:

Table III. Time complexity of a successful search in the proposed method

	Worst	Average	Best
Proposed Method	$O(\log_k^n)$	$O(\log_k^n)$	$O(\log_k^n)$
Other methods[16, 8,18]	$O(n)$	$O(n)$	$O(1)$

According to the table 3, the purposed method is able to search a word in the database in a shorter time, because a big part of the tree is pruned and eliminated in each step and the search is stopped in the pruned sections.

In other methods [16, 8, 18], each shape code is checked with all of stored shape codes in dictionary and it takes long time for huge volume of words. If number of records of dictionary is n, sequential time complexity is $O(n)$ and for the proposed method is $O(\log_k^n)$. Searching in Binary Search Tree is as the logarithm base 2, but here k is logarithm base (k>2). K is the number of feature values in each tree level. K has a variable value in each level; therefore for simplicity we assign constant value for k that is the minimum value among of all possible values. Reason of this selection is that the worst case for time complexity is considered. This value is also better than $O(n)$. In the best form, the unsuccessful search may be stopped at the first level, so time complexity is $O(1)$ in the best form. In the worst form, search may be continued until the last level and become unsuccessful in the last level. The time complexity of an unsuccessful search is given in the table 4:

Table IV. time complexity of an unsuccessful search in the proposed method

	Worst	Average	Best
Proposed Method	$O(\log_k^n)$	$O(\log_k^n)$	$O(1)$
Other methods[16, 8,18]	$O(n)$	$O(n)$	$O(n)$

In the table 4 it is shown that in an unsuccessful search the proposed method performs better and quicker than other discussed methods.

Also, the proposed method in comparison with other methods used in Latin words have better complexity in time, for example in [25] and [26] time complexity is $O(n)$ at successful or unsuccessful search.

B. Time complexity of Insert operation

Time complexity of insert operation has two forms: First form: target word is available in the tree.

This form is similar to the successful search which means that the search is done until the last level and just the number of target document lies in the tree leaf, so time complexity in three worst, best and average situations is $O(\log_k^n)$.

Second form: target word is not available in the tree.

This form is similar to the unsuccessful search which means that from the level that search is stopped there, the rest of the main shape code of the target word is inserted in the tree and the number of document is inserted on the leaf in the last level. Time complexity in three worst, best and average situations are shown in table 5:

Table V. Time complexity of INSERT operation in the proposed method when the desired word is available in the tree.

	Worst	Average	Best
Proposed Method	$O(\log_k^n)$	$O(\log_k^n)$	$O(\log_k^n)$
Other methods[16,8,18]	$O(1)$	$O(1)$	$O(1)$

Table VI. Time complexity of INSERT operation in the proposed method the desired word is not available in the tree.

	Worst	Average	Best
Proposed Method	$O(\log_k^n)$	$O(\log_k^n)$	$O(1)$
Other methods[16,8,18]	$O(1)$	$O(1)$	$O(1)$

According to tables 5 and 6, time complexity of INSERT operation in the proposed method is higher than the usual method. But this is not an important problem, because the INSERT operation is done in the offline phase and in an indexing step, also this operation is performed only once.

V. EXPERIMENTAL RESULTS

A. Dataset Generation

In order to implement and evaluate proposed system, 2 datasets were constructed. The first dataset (training dataset) contained 87867 Farsi printed words (9 images for each state) and was used only for feature extraction in the offline phase. The second dataset (testing dataset) contained 128 Farsi printed words and was used for retrieval operation in the online phase. In this dataset, the Farsi document images were printed and scanned at a resolution of 150 dpi carefully, without noise and skew. The reason for such carefulness while printing and scanning is that we guessed that the skew and noise, especially the salt and pepper noise, may have a bad effect on our approach. On the other way, in the similar processes precision is the same at print and scan [16, 18].

In the suggested system, the database of document images is built of Bijankhan database manually [19]. This collection is made of daily news and common text collections which has 2.6 million words. This dataset is used in many applications of information retrieval that has a lot of words [20, 21]. Most of these words are contained numbers (such as '256'), preposition (such as 'از' , 'به'), conjunctive (such as 'و') words with less frequency and uncorrected words. Therefore, most of words are not necessary and should be eliminated. In this paper, a special approach that is in [4] is used for building dictionary. By using this approach, existing words in dictionary are meaningful words in search and words with high frequency. Special approach is defined in below:

At first, some words of the dataset are selected randomly for building dictionary. Then, commonly words with frequently above 30 are selected. Also, the words which have less than three letters are omitted for the reasons that no one searches them (such as 'از' , 'به'). Number of selected words by using this process is 9763.

These words are written by 'Lotus', 'Nazanin' and 'Mitra', some of the most popular fonts in Farsi scripts. Font sizes in this contest were 12, 14 and 16. Defining 1 special font in 1 special font size as a 'state' in this contest, we have 9 different states. Each document image in both datasets was written in 1 font face out of 3 defined font faces and in 1 font size out of 3 defined font sizes. The selected words have between 3 to 10 letters. Number of words with 3 to 10 letters is shown in the Fig. 10.

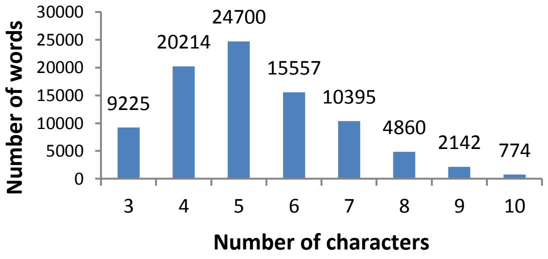


Fig. 10. Distribution of existing words in database with 3 to 10 letters

B. Implementation

The recommended system is implemented by Visual Studio 2008 and is established based on .NET Framework 4. In this system XML is used for saving the tree and its reloading. During indexing, the desired tree is built once and is put in XML file. During retrieval, XML file is loaded primarily and retrieval operation will be done next.

C. Evaluation

The performance of the retrieval by query keywords is evaluated. Precision, Recall and F-Measure criteria are used for evaluating of proposed system's efficiency [13]. Precision is equal to the fraction of number of correctly searched words to total number of searched words. Recall is equal to the fraction of number of correctly searched words to total number of corrected words. 32 random query keywords with Nazanin font are used in the database for calculating the precision and recall. 32 random query keywords with "Nazanin" font and "Mitra" font with size 12, 14 are used in testing phase. The size of test dataset is 128. 32 random words are listed in Fig. 11. These three criteria are expressed by the following equations [13].

گوناگون	اولین	هوایی	براندازی	اخلاقی	آنبرگ	پیروز	لرزش ها
خبرگزاری	خارج	منظومه	اردیبهشت	تغییراتی	تابعیت	برابر	ایران
همچنین	گفته	شمسی	انتشار	شکست	داشتند	مجازات	الیاف
بیماری	شنیدم	سیاره	تحقیق	بارگاه	بجنگد	رویاروی	غرضی

Fig. 11. 32 random query keywords used in the evaluation

$$p = \frac{TP}{TP + FP} \tag{4}$$

$$R = \frac{TP}{TP + FN} \tag{5}$$

$$F - Measure = \frac{2PR}{R + P} \tag{6}$$

Where TP (True Positive) is total number of correctly spotted words, FP (False Positive) is total number of spotted words which are misrecognized; FN (False Negative) is total number of words which are not spotted. Gained values of the recall and precision of proposed method and methods in [16] and [18] are illustrated in Fig. 12 and Fig. 13:

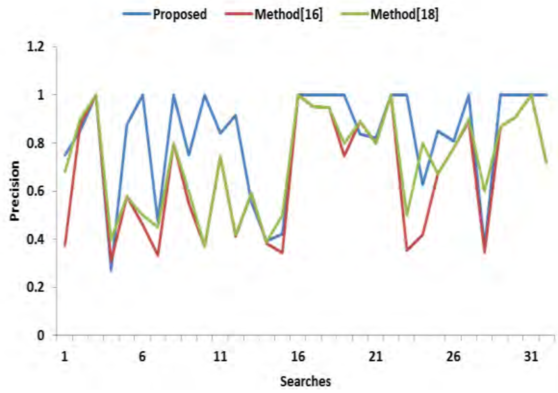


Fig. 12. Variation of precision coefficient of 32 variable searches in the proposed method and methods in [16] and [18] (Font Face: Nazanin, Font Size: 14).

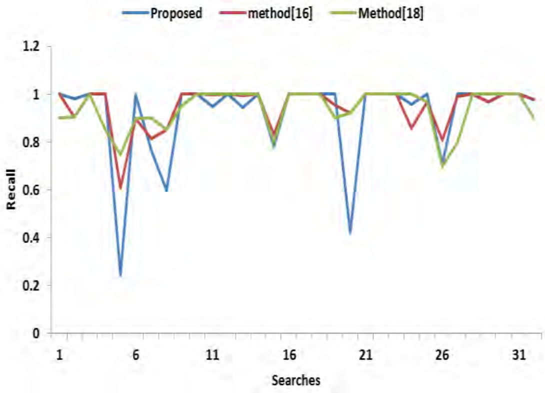


Fig. 13. Variation of recall coefficient of 32 variable searches in the proposed method and methods in [16] and [18] (Font Face: Nazanin, Font Size: 14).

In this paper, 32 keywords have been used for testing according to Fig. 11, the number of retrieved words for every query keyword is different and these numbers for calculating of precision are between 45 to 386 words for query keywords. Also these numbers for calculating of recall are just between 38 to 362.

In this paper proposed method is compared with two methods [16] and [18]. First reason of selecting [16] and [18] is similarity between proposed method and them. For example in method [16], shape coding of Farsi printed words are based on topologically features such as Sub-words, hole, ascender/descenders, dot. Also, the method in [18] is extended from the method in [16]. The second reason is evaluation criteria. Third reason is that these compared methods were presented recently. As it is obvious in Fig. 12, in the suggested method, the average of precision rate is increased significantly by considering the position of the dot in the word feature.

There are some reasons for our method's high precision/recall. The most of these is that our system use position of dots as new features which reduces False Negative and the precision measure increased based on True positive which is constant. Given that a large percentage of the Farsi words have dots [17] and these dots are located in different positions, the proposed method is higher asset performance where the variation of the query is high and performance is low where dots diversities are lower or without dot. As some of the sections shown in Fig. 13, sometimes, the precision of the proposed method are less than two aforementioned methods. This is due to misdiagnosis the dot number and even the dot positions that are the weaknesses of the proposed method. Moreover, our method lacks of character segmentation errors. It has better effectiveness compared with the methods which suffer from the character segmentation errors. Next, an example is given for explaining of this reason.

In addition these two methods, several researches have been done on spotting word in handwriting and machine printed documents in Farsi, Latin and other languages. For example, in [13, 25, 26], They have used ascenders, descenders, character holes, deep eastward concavity, deep westward concavity, i-dot connector and middleline intersection as features for shape coding in Latin words. Weakness of these methods is that dot feature was not used for Farsi printed words shape coding and it caused decreasing efficiency these methods than proposed method. In [27], a method is presented for CSC for spotting words that have character segmentation error and in comparison with our proposed method has a lower efficiency.

In [28] a new successful method in Arabic and Farsi documents is presented but this method has character segmentation error too and has a lower efficiency than our proposed method. In recent years, new methods presented in Arabic handwriting documents for spotting words [22]. Most of these methods used learning based approaches for solving multiwriter challenges in handwritten documents. The weakness of these methods is that lexicon dependent [22], however the proposed method is lexicon independent. Applying these methods to machine printed Farsi/Arabic document isn't suitable.

For example precision for "لرزش ها" word in the proposed method is 0.75 and precision for this word is 0.37 in method [16]. "لرزش ها" word consists of 4 sub- words, 2 ascenders, 3 descenders, 2 holes and 4 dots. If shape code for "لرزش ها" word is made according to the method in [16], total number of retrieved words will be increased which is 8 in this special example. If the shape code of this word is made according to the proposed method, total number of retrieved words will be less than the method in [16] which is 4 in this special sample. Then according to equation (4), the amount of precision is increased because the fractions in both methods have the same numerator and the denominator in the proposed method is decreased. Also in method used in [18] at first XNOR matching between word query and indexed words in database are done and then topological features are checked, therefore precision is increased in this method than method in [18]. Comparison between suggested method and the methods in [16] and [18] with Nazanin font and size 14 is shown in Table 8:

Table VII. Comparison between suggested method and previous methods *(Font Face: Nazanin, Font Size: 14)*

	Precision	Recall	F-Measure
Retrieval by method In[18]	0.72	0.93	0.81
Retrieval by method In [16]	0.67	0.95	0.78
Retrieval by the proposed method	0.83	0.94	0.88

As shown in Table 8, the average precision in [16] and [18] is 0.67 and 0.72, respectively. Also, average recall in [16] and [18] is 0.95 and 0.93, respectively. But In proposed method the average precision is 0.83 and the average recall becomes 0.94. By adding new features, total number of retrieved words decreased and according to the equation (4) precision increased. Also, As regards decreasing the number of retrieved words, the number of retrieved relevant words less decreased. Finally according to the equation (5) the amount of recall less decreased. Comparing proposed methods with methods in[16] and [18]; the proposed method has more precision but less recall. In Spite of degradation of recall, f-measure enhanced that shows an improvement by the proposed method.

Also Comparison between suggested method and the methods in [16] and [18] with other fonts is shown in the Table 9:

Table VIII. Comparison between suggested method and previous methods *(Other Fonts)*

Font + Size	Methods	Precision	Recall	F-Measure
Mitra 14	Retrieval By Method In[18]	0.74	0.89	0.80
	Retrieval By Method In [16]	0.68	0.93	0.78
	Proposed Method	0.77	0.91	0.83
Mitra 16	Retrieval By Method In[18]	0.71	0.93	0.80
	Retrieval By Method In [16]	0.66	0.96	0.78
	Proposed Method	0.78	0.94	0.85
Nazanin 14	Retrieval By Method In[18]	0.72	0.93	0.81
	Retrieval By Method In [16]	0.67	0.95	0.78
	Proposed Method	0.83	0.94	0.88
Nazanin 16	Retrieval By Method In[18]	0.70	0.90	0.79
	Retrieval By Method In [16]	0.68	0.92	0.782
	Proposed Method	0.79	0.95	0.87

Mitra font is very similar with Nazanin font and because of it, their performance are not different, but Tahoma font is very different and it has problem in extracting of ascending and descending and it causes decreasing TP and at result recall and precision decreases a little.

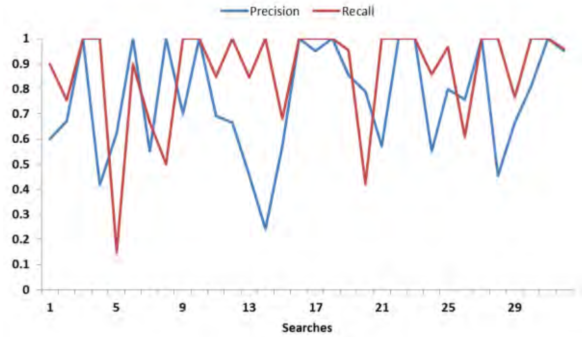


Fig. 14. The variation of the precision and recall coefficients for 32 searches with the query font name ‘‘Titr". The mean precision is 0.76 and the mean recall is 0.86

Furthermore, in order to test the robustness of the features relative to the type of fonts, the query font was changed to ‘‘Titr" and the same 32 searches were made (Table 8). These two fonts have many differences without compromising the shape of the word. The precision and recall values obtained are depicted in Fig. 14. The mean precision and the mean recall values are 0.76 and 0.86 respectively.

VI. Conclusion and Future Works

In this paper, a tree indexing method is presented for Farsi Document Image Retrieval. The proposed method decreases the space of the search significantly by pruning each part of the tree step by step. This causes that the searching speed of a key word which is done with $O(\log_k^n)$ be increased. The search is done. For increasing the amount of precision a new feature is used that is the position of dot in the word. In addition to ascender, descenders, hole and sub- word features, the position of dot in the word is used in this method. Because in Farsi language one, two and three dots can be placed above and below of the words, there will be six different forms. By using this method the precision increases significantly. According to the performed tests, the amount of the precision is enhanced in comparison to the previous methods. For future work, a learning based method can be used for building a tree. This causes the tree's performance can be evaluated with different classifiers and structures. Also statistical features can be used to building tree instead of structural features.

References

[1] D. Doermann. “The Indexing and Retrieval of Document Images: A Survey”, Computer Vision and Image Understanding, vol. 70, no. 3, 1998, 287-298.

[2] M. Keyvanpour, R. Tavoli,” Document Image Retrieval: Algorithms, Analysis and Promising Directions”, International Journal of Software Engineering and Its Applications vol. 7, no. 1, 2013, pp.93-106.

[3] G. Zhu, Y. Zheng and D. Doermann. “Signature-Based Document Image Retrieval”, ECCV, Springer-Verlag, Berlin, Heidelberg ,Part 3, LNCS, Vol. 5304, 2008,752-765.

[4] H. Srinivasan and S. Srihari. “Signature-Based Retrieval of Scanned Documents Using Conditional Random Fields”, Computational Methods for Counterterrorism, Springer-Verlag, Berlin, Heidelberg, ISBN 978-3-642-01140-5, 2009, pp.17-32.

[5] Z. Li, M. Schulte-Austum and M. Neschen,” Fast Logo Detection and Recognition in Document Images”, International Conference on Pattern Recognition, 2010,pp.2716-2719.

[6] C. Shin and D. S. Doermann,” Document Image Retrieval Based on Layout Structural Similarity”, IPCV, 2006, pp.606-612.

[7] Z. Bahmani, R. Azmi. “Farsi/Arabic Document Image Retrieval through Sub –Letter Shape Coding for mixed Farsi/Arabic and English text”, IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 5, No 1, 2011,pp.166-172.

[8] A. Ebrahimi, “A pictorial dictionary for printed Farsi sub words”, Pattern Recognition Letters, vol.29, 2008, pp.656-663.

[9] Y. Pourasad, A. Ghorbani, S. ghouparanloo. “Farsi Font and Font Size Recognition Based on Analyzing Binarization Effect on Small Components of Document Images”, Journal of Basic and Applied Scientific Research Vol.9, No.2, 2012, pp.9563-9568.

[10] M. Akbari, R. Azmi, “Document Image Database Indexing with Pictorial Dictionary”, In Second International Conference on Digital Image Processing. International Society for Optics and Photonics, 2010, pp. 75462R-75462R.

[11] E. J. Erlandson, J.M. Trenkle, R.C .Vogt, “Word-level recognition of multifont Arabic text using a feature-vector matching approach”, Proceedings of the SPIE, Document Recognition III, San Jose, 1996,pp.63-71.

[12] M .Dehghan, K .Faez, M.Ahmadi and M. Sridhar, “Handwritten Farsi (Arabic) word Recognition: a holistic approach using discrete HMM”, Pattern Recognition, Vol. 34, No. 5, 2001, pp.1057-1065.

[13] Shijian Lu, Li Linlin, Chew Lim Tan. “Document Image Retrieval through Word Shape Coding”, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 30, NO. 11, 2008, pp.1913-1918.

[14] K .Zagoris, E. Kavallieratou, and N. Papadakos. “A document image retrieval system”. Engineering Applications of Artificial Intelligence, Vol .23, No. 6, 2010, pp. 872- 879.

[15] M. Keyvanpour and R. Tavoli, “Feature Weighting for Improving Document Image Retrieval System Performance”, International Journal of Computer science Issues, Vol. 9, No. 3, 2012, pp.125-130.

[16] Y. Pourasad, H. Hassibi, Azam Ghorbani, “A Farsi/Arabic Word Spotting Approach for Printed Document Images”, International Journal of Natural and Engineering Sciences, Vol .6, No.1, 2012,pp.15-18.

[17] Saeed Mozaffari, Karim Faez, Volker Margner, Haikal El-Abed. “Lexicon reduction using dots for offline Farsi/Arabic handwritten word recognition”, Pattern Recognition Letters, vol.29, 2008, pp. 724-734.

[18] Y. Pourasad, H. Hassibi, A. Ghorbani. ”A word spotting method For Farsi machine-printed document images”, Turkish Journal of Electrical Engineering and Computer Sciences, vol.21, 2013, pp.734-746.

[19] Bijankhan, Mahmoud. “The role of the corpus in writing a grammar: An introduction to software”, Iranian Journal of Linguistics Vol.19, No. 2, 2004.

[20] P.Saeedi, H. Faili. “Feature engineering using shallow parsing in argument classification of Farsi verbs”. In Artificial Intelligence and Signal Processing (AISP), 16th CSI International Symposium on, IEEE, 2012, pp.333-338.

[21] A, Azimizadeh, , M. M. Arab, and S. R. Quchani, “Farsi part of Speech tagger based on Hidden Markov Model”, In 9th International Conference on the Statistical Analysis of Textual Data,2008.

[22] M. Khayyat, L. Louisa, and C.Y.Suen. "Learning-based word spotting system for Arabic handwritten documents." Pattern Recognition, vol.47, no.3, 2014, pp.1021-1030.

[23] Tan, Chew Lim, Xi Zhang, and Linlin Li. Image Based Retrieval and Keyword Spotting in Documents. Handbook of Document Image Processing and Recognition, 2014.

[24] Kesidis, Anastasios L., Eleni Galiotou, Basilios Gatos, and Ioannis Pratikakis. "A word spotting framework for historical machine-printed documents." International Journal on Document Analysis and Recognition (IJDAR) ,Vol 14, no. 2 ,2011,pp. 131-144.

[25] Bai, Shuyong, Linlin Li, and Chew Lim Tan. "Keyword spotting in document images through word shape coding." In Document Analysis and Recognition, 2009. ICDAR'09. 10th International Conference on, 2009, pp. 331-335.

[26] Lu, Shijian, and Chew Lim Tan. "Retrieval of machine-printed latin documents through word shape coding." Pattern Recognition 41, no. 5 ,2008,pp. 1799-1809.

[27] A. F. Smeaton and A. L. Spitz. “Using character shape coding for information retrieval”. 4th International Conference on Document Analysis and Recognition, August 1997, pp. 974–978.

[28] R. Saabni, J. El-Sana, “Keyword searching for Arabic handwritten documents”, Proceedings of the 11th International Conference on Frontiers on Handwritten Recognition, 2008, pp. 271–277.

Mohammadreza Keyvanpour is an Associate Professor at Alzahra University, Tehran, Iran. He received his B.Sc. in software engineering from Iran University of Science &Technology, Tehran, Iran. He also received his M.Sc. and Ph.D. degree in software engineering from Tarbiat Modarres University, Tehran, Iran. His research interests include image retrieval and data mining.

Reza Tavoli received his B.Sc. in software engineering from Iran University of Science & Technology, Behshahr, Iran. He received his M.Sc. in software engineering from Islamic Azad University, science & Research Branch, Tehran, Iran. Currently, He is pursuing Ph.D. in software engineering at Islamic Azad University, Qazvin Branch, and Qazvin, Iran. His research interests include document image retrieval and data mining.

Saeed Mozaffari received his B.Sc., M.Sc. and Ph.D. degrees in Electronic Engineering from Amirkabir University of Technology, Tehran, Iran in 1999 , 2001 and 2006 respectively. Mozaffari is currently with Electrical and Computer Engineering Department, Semnan University. His research interests include Digital Image Processing, Computer Vision, Neural Networks and Pattern Recognition.