

A Trainless Recognition of Handwritten Persian/Arabic Letters using Primitive Elements

Mehrdad Moradi
Faculty of Electrical and
Computer Engineering
Shahid Beheshti University,
Tehran, Iran

Mohammad Eshghi
Faculty of Electrical and
Computer Engineering
Shahid Beheshti University,
Tehran, Iran

Karim Shahbazi
Faculty of Electrical and
Computer Engineering
Arak Branch Islamic Azad
University, Arak, Iran

ABSTRACT

This paper aim at applying primitive elements composing Persian/Arabic letters to recognize offline handwritten letters. To do so, eight primitive elements are used that with them all Persian/Arabic letters are shaped. Having undergone these modifications, the letters get thinned, and then enter the primitive extraction phase. At this stage through using strokes, the letters' primitive elements would be extracted and by making Stroke Identification Vector (SIV) and then comparing with Character Identification Vectors (CIV) the recognition is gained. Then the number and location of dots on the letters and also the location of the letters towards the baseline was extracted. According to this method, unlike common recognition methods of handwritten methods, no training is required and two processes of separation and recognition are accomplished simultaneously. This algorithm is rule base and according several rule recognition is done. The accuracy of this algorithm in digit recognition is 98.8%. Also recognition on isolated Persian/Arabic words that approximately constitute 50% of the sub words in Persian/Arabic texts is 88.7%, for two letters sub word that make up about 32% of text is 81.4%, for three letters sub word that approximately constitute 12% of the texts is 73.6% and four letters that make up 5% of text the accuracy is 69.7%.

Keywords

Persian/Arabic OCR, Handwritten Recognition, Primitive, joint manuscripts.

1. INTRODUCTION

Over the recent years, the recognition of Persian/Arabic handwritten letters has been of great significance [1]. This importance stems from the various applications of this knowledge in science and industry. Reading application forms, sorting postal packages, reading textbooks and research ability and classification of these text books are also among the applications of this recognition technique.

Generally, letter recognition techniques are divided in two types, offline and online letter recognitions. In the online [2] method, letters can be detected while it being written and in the offline [3] recognition, written letters detected after they have been scanned.

To recognize the letters, some features are extracted; these extraction techniques are called structural [3] and statistical [2]. The structural approach entails extracting some characteristics such as the ratio of the length of a letter to its width. In the statistical method, different features of the letters such as the number of intersections are extracted.

Therefore there are three common types of letter recognition techniques: syntactic method, structural method and a method

which is based on a decision-making technique such [4], [5], [6] and [7]. Due to the irregular and various structures of letters, the decision-making approach has been used more commonly. These methods entail training and suffer from the difficulty of proper training, being time consuming and dependency on databases.

The algorithm presented in this study is an offline words recognition method. The structural approach is used to extract letter's features, but no training of any kind has been used for recognition. In proposed algorithm, normalization and thinning process have employed for avoiding size dependency. Furthermore it used modified Hough transform to eliminate noisy lines and consequently to properly recognize the main lines of letters.

In Section two of the current study turns to explaining problems associated with the recognition of the handwritten letter. The third section focuses on describing the algorithm used in this study. Finally, the last sections of this article present the assimilation results accompanied by an example and conclusion and recognition percentage respectively.

2. BACKGROUND

In Persian/Arabic letters, there are some characteristics that make recognition of these languages much harder than Latin languages (e.g. English) [8]. These characters have many difficulties for recognizing them. The most important problems are listed below:

- In Persian/Arabic manuscripts, all letters are written in the fixed place above or under a straight line, which is called the baseline.
- Dots in Persian/Arabic letters are important. Roughly 56% of Persian/Arabic letters have dots. Some Persian/Arabic letters differ each other just in the position and number of their dots and some of them do not have any dot. These dots can be mistaken by the noise of a scanned text [3].
- Sometimes in handwritten and typed Persian/Arabic text, letters may have overlapped, and may therefore in their separation and recognition some mistakes happen [3].
- Persian/Arabic words consist of one or more sub words. For example word "توانا" have three sub words "ا", "نو", "تا" and "تا"[3].

Table 1. The 8 primitives of Persian/Arabic letters

Kind of text: A Journal paper		
Number of letters in sub words	Repetition in text	percent Repetition
1	5446	49.01
2	3405	30.64
3	1341	12.06
4	772	6.94
5	114	1.07
6	28	0.25
7	6	0.05
Total	11112	100

Kind of text: An educational book

Number of letters in sub words	Repetition in text	percent Repetition
1	19378	50.52
2	12244	31.92
3	4757	12.40
4	1491	3.88
5	445	1.16
6	37	0.09
7	3	0.00
Total	38355	100

Kind of text: A final theses









Number of letters in sub words	Repetition in text	percent Repetition
1	11833	48.59
2	8024	32.95
3	2854	11.71
4	1214	4.98
5	359	1.47
6	62	0.25
7	5	0.02
Total	24352	100

- Some letter forms in Persian/Arabic text have closed curves in their shapes, while others don't. This characteristic can be used as a feature to recognize some letters [8].
- There are different script styles for Persian/Arabic language, e.g. Naskh, Nastaligh, Koofi, etc. The shapes of some letters are very different in these styles [8].
- Persian/Arabic handwriting varies from person to person and it is different from printed versions.
- There are no regular and fixed patterns for handwriting and almost each person has a different handwriting.

Table 1 shows the number of one to seven sub word letters in these texts. One, two and three letter sub words are most important because they contain almost 94% of Persian/Arabic sub words and Just 6% of Persian/Arabic sub words have more than three letters.

It approximated every Persian/Arabic letter using a set of primitive elements (or simply primitives) and dots [8]. It was practical to choose the set of primitives such that no two letters would be approximated with the same sequence of primitives. Thus, any Persian/Arabic letter would uniquely recognize according to sequence of its primitives and the number and the positions of its dots. The proposed set of primitives consists of eight elements, which are shown in Table 2 [8].

Table 2. The 8 primitives of Persian/Arabic letters

Row	Shape of primitives	Name of primitives	Code of primitive
1		Backslash	B
2		Slash	S
3		Vertical	V
4		Horizontal	H
5		D shape	D
6		C shape	C
7		U shape	U
8		Circle	O

3. PROPOSED ALGORITHM

The proposed algorithm is made up of three phases including preprocessing, main process and recognition. First the image needs to be scanned and then put into the system. In the preprocessing phase the image is converted to a binary image and then filter them so that are ready to enter the main processing phase. In continue all of the dots along with their locations and also the baseline is measured. This process is being followed by thinning the image and extracting its skeleton. During the main process, the image characteristics are extracted and this composed of eight primitive elements.

In order to recognize the elements each letter has been decomposed into simple parts that is called strokes. Each stroke is a line or curve whose all pixels have maximally two adjacent. So with this technique, all the lines that compose these strokes are recognized.

To achieve chain codes, modified Hough Transform has been used. For detecting strikes, the primitive elements which constitute the strokes need to be specified, and then by coupling these elements and considering the number of the dots and their location of the letters towards the baseline, a vector to present this stroke's features, have made. These features have been shown in Fig. 1.

Primitive 1	Primitive 2	Primitive 3	Point location	Number of point	location of the letters
-------------	-------------	-------------	----------------	-----------------	-------------------------

Figure 1. SIV and CIV

Having gone through all the aforementioned procedures, SIVturn to the recognition phase. In recognition phase, the stroke chart is compared with some other charts called recognitions vectors to find the expected letter. These vectors have a structure similar to that of the stroke vectors and each demonstrates a specific letter. With a decision tree, true characters are found.

In the following, three phases of the proposed algorithm to recognize characters in a handwritten Persian letters (using mentioned primitives) are described in details.

3.1 Pre processing

In this stage the image is pre-processed so it is ready for the main process. First, thresholding is performed on the image, then considering the level of thresholding, it is converting into a binary image and finally the image is filtered to eliminate the noise.

Furthermore, the image is normalized to a certain size. To do so, the letter must be thinner. Pixel normalization is based on the number of pixel in each column. Finally, after going through the above stages the image is ready to be processed.

3.2 Main Process

In this step the image would be scanned to extract its desirable characteristics. First it is attempted to detect the baseline using horizontal histograms. The baseline of each row of the text is the maximum of the horizontal histogram of that line. In continue it turns to check letter dots. At first, it is clarify that each letter is made up of one major part or consists of different parts. If the image is made up of separate parts, the number of the dots is detected and their length and width has been calculated. Besides, it can be recognize the position of dots by comparing each segment with the baseline. Finally the dots are eliminated so that body of letter is left.

Furthermore, thinning of the letter started to extract letter's skeleton. It has to be thinned enough to reduce the letter to just one pixel and keep the letter structure at the same time. For example, the jags should be reminded intact. This procedure makes the extraction of the primitives easier and it also contributes to the main processing stage.

To do so, the proposed method offered in Article [9] is used. In this study thirty masks have been used to thin the image. Twenty masks to eliminate the border pixel and ten to reconstruct the important pixels which have already been eliminated. In continue characteristics extraction is explained.

3.2.1 Feature extraction

The characteristics that need to be extracted to the eight primitive elements (shown in Table 2).

At the beginning of this process the closed curve is detected which exist in the letters [3]. After that in order to achieving the simplest stroke, these closed curves are eliminated.

At this point, it needs to decompose the letters into their strokes. To do this, the thinned letters have to first be scanned and then the fork points is detected. To complete the process, the Modified Hough Transform is used; otherwise there are a lot of extraneous noises which can make the recognition task more difficult.

Besides, in order to detect the main lines, the Chain Code Method [3] is used. Following this method, the right-most dot scanning is start. It should be mentioned that it is a dot with only octagonal pixels around it. At this point of the process, considering the position of the octagonal pixel around each

pixel, an array can be achieved that shows the position of the surrounding pixel which constitutes the Chain Code. Fig. 2 shows the body of the letter and its Chain Code.

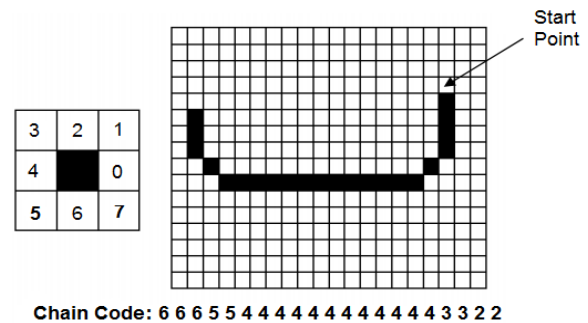


Figure 2. The octagonal neighborhoods and the skeleton of the letter "ب" and its Chain Code

To minimize the noise created by the lines, 22.5° margins are used. As presented in Fig. 3, there are two separate parts to the letter, one with zero degree angles, and the other is a curve with 0, 135, 90, and 45. Following rules are used to determine the type of the elements.

- Rule 1: the primitive elements "V", "S", "H", and "B" are determined by means of line angles.
- Rule 2: if the combination of the lines in S collection is one of the following, it is called "C-SHAPE" (1).

$$S = \left[\begin{array}{l} \{0 \pm 22.5, 45 \pm 22.5, 90 \pm 22.5, 135 \pm 22.5, 0 \pm 22.5\} \\ \{0 \pm 22.5, 90 \pm 22.5, 0 \pm 22.5\} \\ \{0 \pm 22.5, 45 \pm 22.5, 135 \pm 22.5, 0 \pm 22.5\} \\ \{45 \pm 22.5, 90 \pm 22.5, 135 \pm 22.5, 0 \pm 22.5\} \end{array} \right] \quad (1)$$

- Rule 3: the "D-SHAPE" primitive is the rotated of S collection by 180 degrees.
- Rule 4: the "U-SHAPE" primitive is the rotated of S collection by 90 degrees.

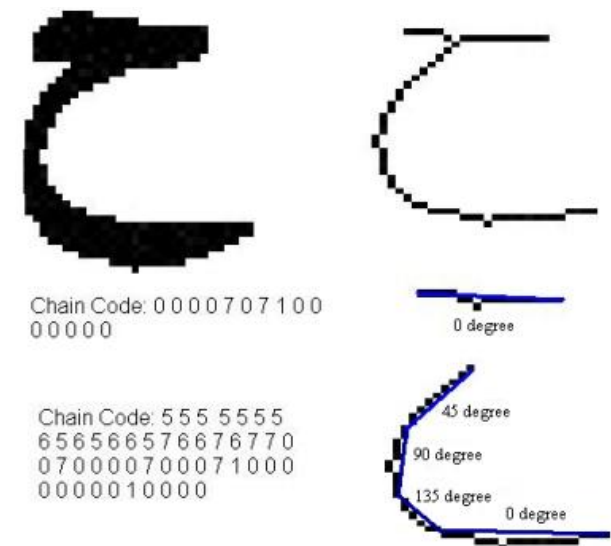


Figure 3. Persian/Arabic Letter "ح" with its thinned image. The lines are extracted by using the chain codes

As illustrated in Fig. 3, for letter "ح" there are two parts. One part is a line that has 0 degree angle ('H' primitive) and the other is a curve that has lines of 45, 90, 135 and 0 degrees.

The second part, according to rule 2, is a “C-SHAPE” primitive. Primitive of this letter is therefore “CH” [3].

3.2.2 Stroke Detection Vector

This vector can be drawn using this method and primitive element extraction method. As shown in Fig. 1 this vector includes five positions for the element types, two for the number and position of the dots, one for determining the location of the letter against the baseline. For instance, in case of the letter “ق”, the stroke detection chart is presented as “U0U1U”. “U0” are the primitive elements of the letter and one stands for the number of the dots in the letter. Additionally, “U” at the bottom of the chart suggests that the letter is mainly lying above the baseline.

After extracting primitives, the basic elements of letters should be detected. First, this vector is compared with discrete characters vector. If it is matched, then a discrete character is exist. If the operation of the group is not matched to any vectors, sweep is begin from end of SIV and compared with vector identifying vector. The rest of the match vector compared to the initial letters of recognition and recognition is done. Table 3 shows the vector identifying letters. For the first and the last letters are the same tables.

In spite the charts used in [3] and [9], most letters in this chart consist of a various combination of primitive elements which are different in length. This phenomenon could be traced back to the variety existing in the Persian/Arabic scrip styles in addition to the partial format of the letters, for example, in case of the letter “ق” the circle might be left incomplete.

4. SIMULATION TEST RESULT

A computer simulation is used to test the proposed Persian/Arabic OCR algorithm. The rules to determine the alphabet Persian/Arabic letter is shown in Table 4. A data base of ICH Persian/Arabic words, written by 15 persons is used by this simulation to determine its accuracy. As an example, Fig. 4 shows the result of applying the algorithm on the image of letter “ق”. This Figure also shows the thinned image and also the stroke components of the letter and the SIV of this letter is presented in Table 3.



Figure 4. The main image, thinned and strokes of the letter “ق”

Table 3. The SIV of letter “ق”

P3	P2	P1	Point Number	Point Position	Position relative to the baseline
U	O	-	U	2	I

The proposed algorithm in this paper is also tested against Database IFHCDB [10]. The result of proposed Algorithm to detect and accurately identify sub word is given in Fig. 5.

In Table 5 summarized the some OCR systems with different algorithm along with accuracies and datasets.

Table 4. The primitive of Persian/Arabic separate letters

letters	Primitives set of			Points		Position relative to the baseline
	P1	P2	P3	Position	Number	
ا	V			0	0	I
آ	V			U	2	U
ب	U			D	1	U
پ	U			D	3	U
ت	U			U	2	U
ث	U			U	3	U
ج	C			D	1	D
	C	O				
چ	C			D	3	D
	C	O				
ح	C			O	0	D
	C	O				
خ	C			U	1	I
	C	O				
د	D			0	0	U
ذ	D			U	1	U
ر	D			0	0	D
ز	D			U	1	D
ژ	D			U	3	I
س	U			0	0	I
	D	U				
	U	U				
ش	U			U	3	U
	D	U				
	U	U				
ص	U	O		U	0	I
	D	U				
	C	D				
ض	U	O		U	1	I
	D	U				
	C	D				
ط	D	H	O	0	0	U
	V	H	O			
	V	U	O			
ظ	D	H	O	U	1	U
	V	H	O			
	V	U	O			

ع	C			0	0	D
	C	C				
	C	H				
	C	U				
غ	C			U	1	D
	C	C				
	C	H				
	C	U				
ف	U			U	1	U
	U	O				
	H	O				
ق	U			U	2	I
	U	O				
	H	O				
ک	U			0	0	U
	D			U	3	
گ	U			U	3	U
	D					
ج	C			0	0	I
	D					
چ	C			0	0	I
	D					
	D	O				
ن	U			U	1	D
و	D	O		0	0	I
ه	O			0	0	U
ی	U			0	0	D

5. CONCLUSION

Preprocessing is the first step in proposed algorithm. In this step, the image was converted into a binary image and then filtered. The purpose of filtering is to eliminate image noises. Furthermore, based on the letter thickness, algorithm will normalize the size of the letter which is a contributing factor in enhancing the accuracy of the algorithm. Then baseline was detected and number and position of dots was calculated and separated from the letter. In next stage, the letter is thinned and stroke was detected and decomposed. Finally, Hough Transform extracts the strokes of the main line.

By using of these lines, primitive elements are now able to recognize. These elements are Slash, Backslash, Vertical, Horizontal circle, C shape, D shape and U shape.

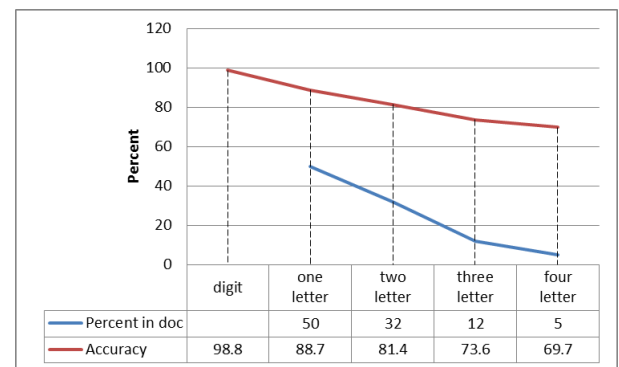


Figure 5. The precision of proposed algorithm and the percentage of the number of sub word in Persian/Arabic

These elements along with information on the position of the letters toward the baseline in addition to the position of their dots can be found in the stroke detection chart. Letter recognition can be achieved through comparing the aforementioned chart with the letter recognition chart.

This algorithm requires no training and by taking some quick and simple steps is capable of recognizing isolated letters and digit with the accuracy of 88.7% and 98.8% respectively. For two three and four letter sub words with the accuracy of 81.4%, 73.6% and 69.7%. In average precision of this method is 82.71%.

Table 5. Comparison of handwriting OCR system

Authors	Language	Features	Classification	Dataset	Accuracies
[11]	Urdu	DCT	HMM htk toolkit	1259 Unique	92%
[12]	Farsi	Fusion of statistical and structural	HMM	1000 Online ligature samples	Testing: 87.5% Training: 92.9%
[13]	Arabic	Sliding window and structural	HMM and re-ranking	26459 and 32492	83.55%
Ours algorithm	Farsi	Primitives	-	60000	82.71%

Comparing with similar methods, the proposed method has notable accuracy despite of eliminating training stage and large data set testing.

6. REFERENCES

- [1] F. Bortolozzi, A.S. Brito, L.S. Oliveira, M. Morita, Recent “advances in handwritten recognition,” in: U. Pal, S.K. Parui, B.B. Chaudhuri (Eds.) , Document Analysis, pp. 1–30, 2008 .
- [2] A. Malaviya, C. Leja, L. Peters, “004Dulti-script handwriting recognition with FOHDEL,” Proceedings of NAFIPS'96, IEEE Press, Berkeley, pp. 147-151, 1996.
- [3] Sh. Ensafi, M. Eshghi and M. Naseri, “Recognition of separate and adjoint Persian letters using primitives”, Proceedings of IEEE Symposium on Industrial Electronics & Applications, Vol. 2, Kuala Lumpur, pp. 611-616, 2009.
- [4] H. A. Al-Muhtaseb, S. A. Mahmoud, and R. S. Qahwaji, “Recognition of off-line printed Arabic text using Hidden Markov Models,” Signal Processing, vol. 88, pp. 2902-2912, 2008 .
- [5] S. Singh and A. Amin. “Neural network recognition and analysis of hand-printed characters”, Proc. IEEE International Joint Conference on Neural Networks IJCNNP8, IEEE World Congress on Computational Intelligence, Anchorage, Alaska, May 4-9, 1998.
- [6] C.L. Liu, K. Nakashima, H. Sako, H. Fujisawa, “Handwritten digit recognition: benchmarking of state-of-the-art techniques,” Pattern Recognition, Vol. 36, No. 10, pp. 2271–2285, 2003.
- [7] M. Hanmandlu, K.R. Murali Mohan, H. Kumar, “Neuralbased handwritten character recognition,” in: Proceedings of Fifth IEEE International Conference on Document Analysis and Recognition (ICDAR'99), Bangalore, India, 20–22, pp. 241–244, 1999.
- [8] Sh. Ensafi, M. Miremadi, M. Eshghi, M. Naseri and A. Keipour, “Recognition of Separate and Adjoint Persian Letters in Less than Three Letter Subwords Using Primitives,” Proceedings of Iran 17th Electrical Engineering Conference, Tehran, pp. 11-13, 2009.
- [9] B. K. Jang and R.T. Chin, "One-pass parallel thinning: analysis, properties, and quantitative evaluation", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 14, No. 11, pp.1140-1129, 1992.
- [10] S. Mozaffari, K. Faez, F. Faradji, M. Ziaratban and S.M. Golzan “A comprehensive isolated Farsi/Arabic character database for handwritten OCR research,” Proc. 10th International Workshop on Frontiers in Handwriting Recognition (IWFHR), La Baule, France , pp. 385-389, 2006.
- [11] Javed, S.T., S. Hussain, A. Maqbool, S. Asloob, S. Jamil and H. Moin, 2010. Segmentation free Nastalique Urdu OCR. Proceedings of World Academy of Science, Engineering and Technology, 46: 456-461.
- [12] Ghods, V., E. Kabir and F. Razzazi, 2013a. Decision fusion of horizontal and vertical trajectories for recognition of online Farsi sub words. Eng. Appl. Artificial Intell., 26: 544-550.
- [13] AlKhateeb, J.H., J. Ren, J. Jiang and H. Al-Muhtaseb, 2011. Offline hand-written Arabic cursive text recognition using Hidden Markov Models and re-ranking. Pattern Recogn. Lett., 32(8):1081-1088.