World Applied Sciences Journal 13 (2): 239-243, 2011
ISSN 1818-4952
© IDOSI Publications, 2011

Persian Characters Recognition Based on Spatial Matching

Yaghoub Pourasad, Houshang Hassibi and Majid Banaeyan

Department of Electrical and Computer Engineering,
K.N. Toosi University of Technology, Tehran, Iran

Abstract: In recent years, several studies have been performed on the language recognition and many types of software have been developed specially in international languages such as English. Although, there are many types of software to recognize characters in English language there is great need to develop algorithms to perform on Persian (Farsi) languages. In this paper we propose a novel method to recognize the characters of Persian language using a method based on spatial matching of contour points. Using the Euclidean distance between spatial descriptors and the gradient value in each point, increased the confidence during matching process. Various Persian language fonts are used in our testing database and obtained matching accuracy is more than 98%.

Key words: Character · Recognize · Spatial matching · Euclidean distance

INTRODUCTION

To date, many efforts have been made to build digital libraries which digitize high-volume archives of paper documents (patent, legal tomes, historical documents) to provide the public with free and easy on-line access. These digital libraries store scanning images, which keep visual information such as layout and decorations. However, this leads to difficulties in document retrieval, because traditional text information retrieval techniques totally fail when documents are simply presented as raw bit-maps. A feasible solution for these documents retrieval is Optical Character Recognition (OCR). In the last decades a big research effort has been spent aiming at the development of automatic text reading systems. Although OCR systems proved to be powerful enough to meet the requirements of many users, room for improvement still exists and further research efforts are required.

Farsi/Arabic text recognition, which was not researched as thoroughly as Latin, Japanese, or Chinese, is receiving a renewed interest not only from Farsi/Arabic-speaking researchers but also from non-Arabic-speaking researchers [1, 3]. This has resulted in the improvement of the state of the art in Farsi/Arabic text recognition in recent years. Higher recognition rates were reported and more practical data is being used for testing new techniques. In addition to the traditional applications like check verification in banks, office automation and postal address reading, there is a large interest in searching scanned documents that are available on internet and also for searching handwritten manuscripts. There are many researches on Arabic Optical Text Recognition [4, 5] and also the different stages of an Arabic text recognition system such as, a database for Arabic handwritten text recognition [6], a database for Arabic handwritten checks [7], pre-processing methods [8], segmenting of Arabic text [9], different types of features that are used [10-11] and multiple classifiers [12, 13]. Due to the advantages of Hidden Markov Models (HMM) many researchers have used them for Arabic text recognition [14-15].

Some researches are presented in Farsi optical text recognition [16-19]. Persian documents have some special characteristics compared with the English documents. Some of these characteristics in Persian are:

- The Persian scripts are cursive.
- There are 32 basic characters in Persian scripts and shape of these characters may change according to their position (beginning, middle, end or isolated) in the word. Each character can take up to four different shapes, as result there are 128 different shapes for all of Persian alphabets.

Corresponding Author: Yaghoub Pourasad, Department of Electrical and Computer Engineering,
K.N. Toosi University of Technology, Tehran, Iran. P.O. Box: 15875-4416.
Tel: 98 21 8888 2991-3, Fax: 98 21 8879 7469, E-mail: dpoorasad@yahoo.com.

- Most of the Persian characters have one, two or three dots which can be situated at the top, inside or bottom of the characters.
- Non uniformity of words size which is important feature of Farsi texts.

In this paper we propose a new method for recognition of Farsi characters by using a matching technique to recognize query characters from database.

This paper is organized as follows: in section 2, matching a lgorithm and its different steps is described. In section 3, experimental results are analyzed and finally in section 4, conclusion is given.

Matching Algoritm: One Image is indicated as a set of pixels in 2D area. Each pixel in this area is equal to a point in Cartesian coordinates. The aim of matching problem is to correspond two images as two series of pixels. The images which used in this paper have gray level scales with bright background while the characters are dark. Finding two characters as similar means that, the pixel which illustrates them have similar brightness value in spatial domain. Hence, if the size of two images is same, by increasing the number of similar points, the result of matching process is higher. In proposed method, the Euclidean distance between pixel's location and the value of gradient in different points is utilized for matching process, which is described in the following section.

Extracting Contour Points: Since edges have more changes in comparison of other part of the character, they have more information and are used to indicate the contour points. The number of these points is usually high and reliable in different images but in order to avoid having so many contour points which cause the matching process consume a lot of time, the number of these points is decreased in this method. For this purpose a sampling process is applied on contour points to decrease them to a value which indicated with n. The sampling process should be random and also uniform. The input character and its correspondence sample points have shown in Figure 1(a) and 1(b) respectively.

Gradient of Sampling Points: The gradient of a point indicates the direction of maximum intensity change in the area of its location. The gradient vector of a sample point is defined in equation.

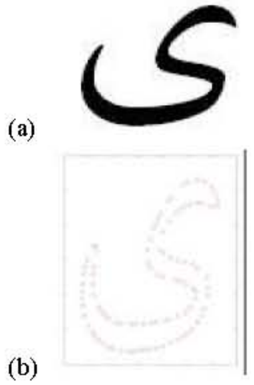


Fig. 1: The input character image (a) and the n sample of contour points, where n = 100(b)

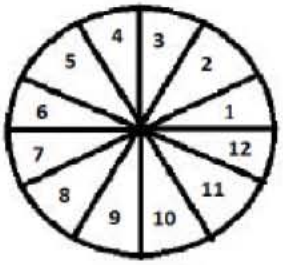


Fig. 2: The twelve gradient intervals

$$\begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \partial I / \partial x \\ \partial I / \partial y \end{bmatrix} \quad (1)$$

Where I, is the input image. Magnitude and direction of this vector are:

$$|G| = \sqrt{G_X^2 + G_y^2} \quad (2)$$

$$\theta = \tan^{-1}(G_y / G_x) \quad (3)$$

For each of the sampling points which are extracted from previous step, the gradient vector is computed. In this case, to reduce the computational time we use twelve intervals of gradient vectors including {0-30, 30-60, 60-90... 330-360}, which are locateduniformly around a circle. Figure (2) shows twelve intervals of gradient.

Spatial Matching Algorithm: In this section we compare the image of input character with all saved images in the database and recognize which of the images in the database is corresponds to the input image. First we chose n points of input image and then we compute the Euclidean distance between them and store it in a n ×n matrix (Q). The (i,j) index of this matrix indicates the Euclidean distance between points i and j.

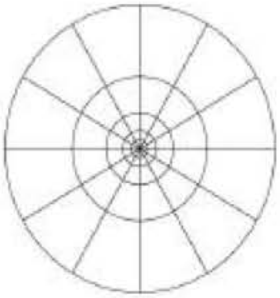


Fig. 3: The graphically representation of SGDD

The Euclidean distance between these points $p_1(x_1,y_1)$ and $p_2(x_2,y_2)$ is described as equation (4).

$$d = \sqrt{(P(X_1) - P(X_2))^2 + (PY_1) - P(Y_2))^2} \tag{4}$$

To decrease the computation time during matching process, the Euclidean distance between different points is divided logarithmically to five different intervals, where the minimum and maximum distance are 1.3335 and M respectively. For example if we suppose M = 100, the logarithmic intervals are y= [1.3335 3.9242 11.54 33 100]. In this case, each Euclidean distance locates in one of the defined intervals. Therefore, for each sample point a Spatial Gradient Difference Descriptor (SGDD) is defined. In addition to Euclidean distance, respective slope is defined between two point's p (x_1,y_1) and p (x_2,y_2). Respective slope is described in equation (5).

$$\Delta G_{12} = \frac{P_{1y} - P_{2y}}{P_{1x} - P_{2x}} \tag{5}$$

As we divided gradient vector to twelve intervals, the respective slope is divided to twelve intervals, too. Each sample point is located in five Euclidean distance intervals and twelve slope intervals respect to other points. It means for each point we will have a descriptor vector SGDD with length of 60 (5 ×12 = 60), which is illustrated graphically in Figure (3).

Experimental: To evaluate the proposed method, we constructed a database including all letters of Farsi (Persian) language. For each letter, 10 images (various fonts and sizes) were stored and for each image in the database, n sample points were extracted and for each point, descriptor vector (SGDD) was saved. During matching process by using MATLAB software, the SGDD vector of each point of the input image is compared with the SGDD vector of each point of the database images. Then, the Euclidean distance and respective slope in each similar interval are compared in order to optimal assignment of correspondence points. Finally the gradient value of each point is used as a property which helps us to better correct matching. We suppose K (the number of correspondence points) is the most important criteria for our judgment about the result of matching. Hence, the confidence term is defined as:

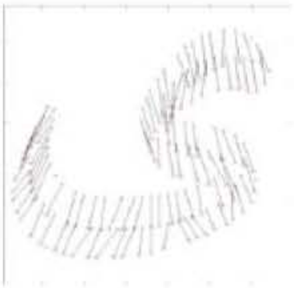
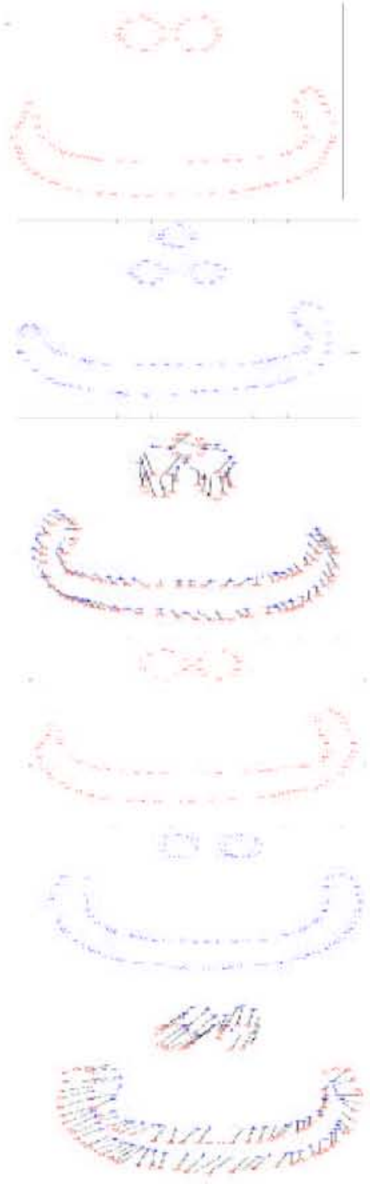


Fig. 4: Input image and result together (K = number of correspondence point)

$$c = k/n \tag{6}$$

The maximum value of C which is resulted from comparison between the input image and the database images is related to similar characters. In this way, we can recognize the input character among all characters in the database. An example of input character and the result of the proposed method are illustrated together in Figure (4).

Since there are Italic and bold font styles in addition to various fonts, to avoid wrong distinguishing and to increase the accuracy of the method the averaging sum of square differences (SSD) between two image is used. In some cases the number of correspondence points (thus c) between 2 different, but almost similar characters is equal with the (c) of 2 exactly similar characters in different fonts. In these cases the average of SSD between two images is used and the lower SSD is related to better matching. For example correspondence points (and therefore c) of characters "ت", "ث" which are in same font is equal to correspondence points of characters "ت", "ت" that are in two different fonts. The result for two cases is illustrated in Figure (5) and Table (1). As illustrated in Rable (1), "c" parameter in both matching ("ت" with "ث" that are different characters in same font and "ت" with "ت", that are same characters in different fonts), is 94%, but SSDs are different.SSD between same characters even with different fonts is lower than different



(5.a)

(5.b)

(5.c)

(5.d)

(5.e)

(5.f)

Fig. 5: Matching images in same font (a), (b), (c). Matching image in deferent fonts (d), (e), (f)

characters with same font. This shows that in this method with considering "c" and SSD, we can exactly recognize query character among database. Table (2) shows 3 collections of characters. In the first part of Table (2) there are some characters that are completely (100%) recognizable from each others. For example two characters such as "ا" and "ع" in every fonts or sizes, will recognized 100% correctly. In second part of Table (2) there are characters that with 99% of probability are correctly recognizable from each others; and only 1% of probability exists for mistake.

Table 1: The result of SSD and C for the same font and different fonts

Character in same font	SSD	Characters in different fonts	SSD	C
"ت", "ت"	0.021	"ت", "ت"	0.018	94%
"ک", "گ"	0.019	"ک", "گ"	0.014	91%

Table 2: Three collections of characters and their matching accuracy

Characters	Accuracy of matching
ا س ش ص ض ط ظ ع غ ل م ن و ه ی	100%
ب پ ت ف ق ج چ ح خ ر ز ژ د ذ	99%
ک گ	98%

CONCLUSION

In this paper a novel method is proposed to recognize Persian characters which utilize the Euclidean distance and the gradient direction in each point as a descriptor vector. In Table (2), is illustrated that this algorithm exactly finds the match of each character.

REFERENCES

1. Arivazhagan, M., H. Srinivasan and S. Srihari, 2007. In the Proceedings of SPIE.
2. Femiani, J., M. Phielipp and A. Razdan, 2005. Document Image Understanding Technology. In the Proceedings of the 2005 Symposium in CollegePark, Maryland.
3. Jin, J., H. Wang, X. Ding and L. Peng, 2005. In the Proceedings of SPIE-IS&T Electronic Imaging, 5676: 48-55.
4. Khorsheed, M., 2002. Off-line Arabic Character Recognition - A Review. Pattern Analysiss & Applications, 5: 31-45.
5. Amara, N. and F. Bouslama, 2005. Classification of Arabic script using multiple Sources of information. International Journal on Document Analysis and Recognition, 5(4): 195-212.
6. Almaadeed, S., D. Elliman and C. Higgins, 2002. In the Proceedings of Eighth Int"l Workshop Frontiers in Handwriting Recognition, pp: 485-489.
7. -Ohali, Y., M. Cheriet and C. Suen, 2003. Databases for recognition of handwritten Arabic cheques. Pattern Recognition, 36: 111-121.
8. Farooq, F., V. Govindaraju and M. Perrone, 2005. In the Proceedings of the Ninth International Conference on Document Analysis and Recognition, ICDAR 05, IEEE Computer Society, Seoul, Korea, 1: 267-271.

9. Haraty, R. and A. Hamid, 2002. In the Proceedings of Intl Conference Computer Science, Software Eng, Information Technology, e-Business and Applications.
10. Amin, A., 2003. Recognition of hand- printed characters based on structural description and inductive logic programming. Pattern Recognition Letters, 24: 3187-3196.
11. Mozaari, S.K. Faez and M. Ziaratban, 2005. In the Proceedings of the Ninth International Conference on Document Analysis and Recognition, ICDAR05, Seoul, Korea, pp: 1.
12. Farah, N., L. Souici, L. Farah and M. Sellami, 2004. In the Proceedings of Artificial Intelligence: Methodology, Systems and Applications.
13. Farah, N., A. Ennaji, T. Khadir and M. Sellami, 2005. Document Analysis and Recognition. In the Proceedings of Int"l Conf., pp: 222-226.
14. Almaadeed, S., C. Higgens and D. Elliman, 2004. Knowledge-Based Systems, 17: 75-79.
15. Al-Qahtani, S. and M. Khorsheed, 2004. Artificial Intelligence and Soft Computing. In the Proceedings of Eighth IASTED Int"l Conf.
16. Azmi, R. and E. Kabir, 2002. A new segmentation technique for omnifont Farsi text. Elsevier Pattern Recognition Letters, pp: 97-104.
17. Khosravi, H. and E. Kabir, 2007. A very large database of handwritten Farsi digits and a study on their varieties. Pattern Recognition Letters, 28(10): 1133-1141.
18. Parhami, B. and M. Taraghi, 1981. Automatic Recognition of Printed Farsi Text. Pattern Recognition, 14: 395-403.
19. Shahrezea, M., K. Faez and A. Khotanzad, 1995. In the Proceedings of International Conference on Image Processing, 3: 436-439.