

## A Comparative Study of Persian/Arabic Handwritten Character Recognition

Alireza Alaei  
 Department of Studies in Computer  
 Science, University of Mysore  
 Mysore, 570006, India  
 alireza20alaei@yahoo.com

Umapada Pal  
 Computer Vision and Pattern  
 Recognition Unit, Indian Statistical  
 Institute, Kolkata-108, India  
 umapada@isical.ac.in

P. Nagabhushan  
 Department of Studies in Computer  
 Science, University of Mysore  
 Mysore, 570006, India  
 pnagabhushan@compsci.uni-mysore.ac.in

**Abstract** — In recent years, many techniques for the recognition of Persian/Arabic handwritten documents have been proposed by researchers. To test the promises of different features extraction and classification methods and to provide a new benchmark for future research, in this paper a comparative study of Persian/Arabic handwritten character recognition using different feature sets and classifiers is presented. Feature sets used in this study are computed based on gradient, directional chain code, shadow, under-sampled bitmap, intersection/junction/endpoint, and line-fitting information. Support Vector Machines (SVMs), Nearest Neighbour (NN), k-Nearest Neighbour (k-NN) are used as different classifiers. We evaluated the proposed systems on a standard dataset of Persian handwritten characters. Using 36682 samples for training, we tested the proposed recognition systems on other 15338 samples and their detailed results are reported. The best correct recognition of 96.91% is obtained in this comparative study.

**Keywords-** Persian/Arabic Character Recognition; Handwriting Recognition; Under-sampled bitmaps; Chain Code; Gradient Features; Shadow Features.

### I. INTRODUCTION

Many potential applications such as postal automation, Bank cheque processing, and automatic data entry depend on handwritten character recognition. These applications promise the recognition of handwritten characters to be an active research area for many years and at the outset high accuracies of handwritten character recognition have been reported for some languages such as English, Chinese, and Japanese languages [1, 2].

In recent years, the recognition of Persian/Arabic scripts is receiving increasing attention and many approaches for the recognition of Persian/Arabic handwritten characters have been proposed by researchers [1, 2]. However, high accuracy of recognition has not been achieved from existing recognition systems of Persian/Arabic handwritten characters [8-10]. The main reasons for getting low accuracy of recognition can be referred to existence of many characters with very much similarity in shape and outline, multiple forms of dots and its position in characters, and more number of classes compared to English and other western languages.

Persian/Arabic language mostly dominates in Iran, Middle East, north of Africa and all Arab countries. Arabic alphabet set contains 28 basic characters. Persian alphabet set includes 32 basic characters derived from Arabic alphabet

set. In the Persian alphabet set, there are four characters (پ, چ, ک, گ) in addition to the Arabic alphabet set where they have similar main shapes with other alphabets and only differ by the number of dots. The basic shapes of Persian and Arabic characters are shown in Figure 1 and the characters indicated by a ‘\*’ in Figure 1 are absent in the Arabic alphabet set. In Persian, Arabic, Urdu and Pashto alphabet sets there are several letters that share the same basic form and differ only by a complementary part. The complementary part can be only one dot, a group of dots (2 or 3 dots) or a slanted bar (diacritic). Different shapes of group of dots are shown in Figure 2. In Figure 3, we show some characters with the same basic form but differ only in the number and position of dots.

1	2	3	4	5	6	7	8
ا, آ	ب	*پ	ت	ث	ج	*چ	ح
9	10	11	12	13	14	15	16
خ	د	ذ	ر	ز	*ژ	س	ش
17	18	19	20	21	22	23	24
ص	ض	ط	ظ	ع	غ	ف	ق
25	26	27	28	29	30	31	32
ک	*گ	ل	م	ن	و	ه	ی

Figure 1. Printed samples of isolated Persian characters.

Three dots				
Two dots				

Figure 2. Different shapes of group of two and three dots in Persian handwritten documents.

ح	ج	خ	چ	ب	پ	ت	ث
ح	ج	خ	چ	ب	پ	ت	ث

Figure 3. Exmples of some printed and handwritten characters with the same basic shape but differ only by the number and position of dots.

Many features extraction techniques based on fractal code [3], moment [4], structural features [5, 9], Concavity pixel distribution [6], projection [7], longest-run, gradient operator [9] and wavelet [8] have been detailed in the literature. For classification, different types of Neural Networks and Vector Quantization [3-6, 9], Hidden Markov Model [7], SVM's [8] and K-Nearest Neighbour [9] have been employed. To support the research in Persian/Arabic handwritten recognition, some handwriting databases have been introduced and a competition of Persian/Arabic handwritten character recognition has also been accomplished [9, 18].

To get idea of recognition results of different feature sets and classifiers in Persian/Arabic scripts, in this paper a comparative study of Persian/Arabic handwritten character recognition results on a standard dataset of Persian/Arabic handwritten characters is reported. Similar to the benchmarks of digit recognition reported in literature [19, 20], this comparative study can also provide a new benchmark for fair comparison of results on a common dataset for future research.

Many feature extraction and classification techniques have shown good results in handwritten Latin, Persian and Bangla numeral recognition [19, 20]. To compare the performances of different recognition systems, 4 different classifiers and 8 different feature sets based on under-sampled bitmaps [11, 12], modified contour chain-code [10, 14, 15], gradient image [16, 19], shadow code [13], intersection/junction/end points and line-fitting information [20, 20] of the binary images are used here. Classifiers like Nearest Neighbour (NN), k-Nearest Neighbour (3-NN and 5-NN) and Support Vector Machines (SVMs) are considered. In our study, the recognition systems are extended on complete set of Persian characters (32-class problem) instead of only 8-class problem. The 8 classes are obtained by visually grouping the similar shape characters from 32 classes of Persian/Arabic isolated characters [3, 5, 7, 8, 10]. Moreover, recognition accuracy is also improved in our study. The 32-class problem considered in our study is shown in Figure 4.

The organization of the rest of the paper is as follows: In Section II we illustrate feature extraction techniques and Section III describes classifiers. Experimental result and discussion are described in Section IV. Finally, we present conclusion and future work in Section V.

## II. FEATURE EXTRACTION

In the following subsections a brief description of different feature sets used in our comparative study is given.

### A. Under-sampled bitmap features

Feature extraction based on under-sampled bitmap is a simple technique used in literature by many researchers [10, 11]. The under-sampled bitmaps consists of dividing each input image into a number of non-overlapping blocks of similar size. Then, the number of black pixels is counted in each block. This generates an input matrix with each element being an integer in the range 0 to the size of non-overlapping block. Dividing these values by the size of the block, the

values are normalized between 0 and 1. Under-sampling process reduces dimensionality of the features compared to whole image size and provides invariance to small distortions and slant [12].

In our study, after binarizing the input image, minimum-bounding box of the input image is obtained (Figure 6). Then for better result and independency of features to size and position (invariant to scale and translation), minimum-bounding box of the image is converted into a normal size of 50×50 pixels based on aspects ratio (Figure 7). This size is decided from the experiment. To compute the under-sampled bitmaps features, the normalized image (50×50) is divided into 25 non-overlapping blocks of 10×10 (Figure 7). Then, the number of black pixels is counted in each block. This generates an input matrix of 5×5 with each element being an integer in the range 0 to 100. The result of this process before normalization is shown in Figure 8. Dividing these values by size of the block (100), the values are normalized between 0 and 1. Since, the normalized image is divided into 25 blocks, 25 features are obtained for each input character.

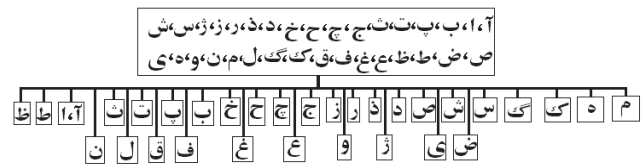
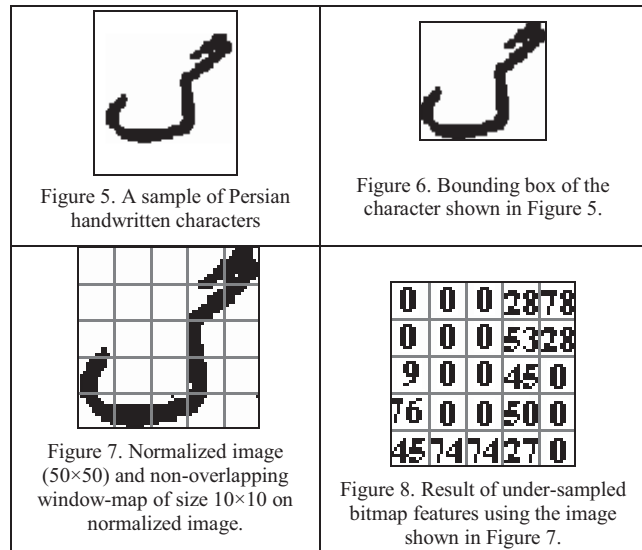


Figure 4. The proposed two-stage Persian character classification scheme

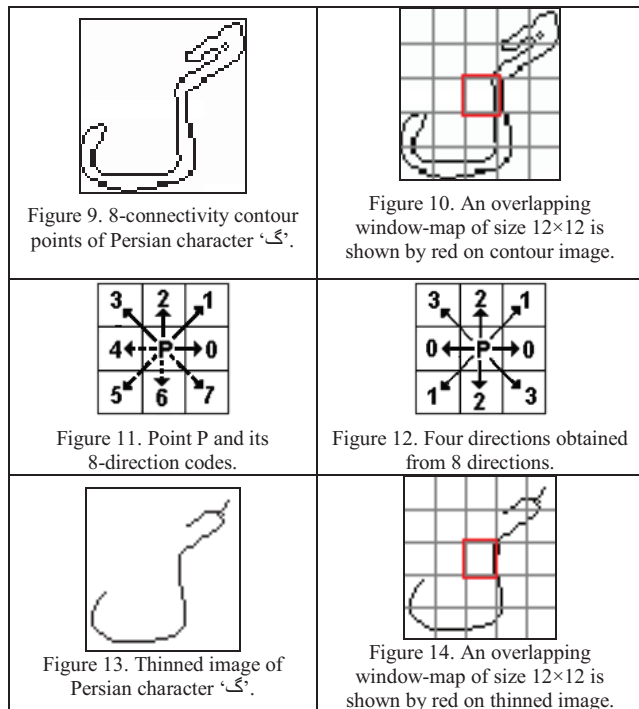


### B. Directional chain code features

The chain code directions information of the contour points of the input image have been used as features for different purposes including numeral and character recognition [10, 14, 15]. Contour information can give the idea about character shape and its thickness. To compute directional chain code features, similar to the under-sampled feature extraction technique, at first, minimum bounding box of the input image is obtained and the minimum bounding box is normalized into 50×50 pixels using aspects ratio. Using the normalized binary image, the contour points of the

character image are found based on 8-connectivity formula (Figure 9). The image contour is scanned horizontally by keeping an overlapping window-map of size 12×12 (Figure 10) on the image from the top left most point to down right most point (25 overlapping blocks). For each overlapping block, the chain code frequencies for all 8 directions are computed. Instead of expressing the features in terms of 8 directions, we simplify the features into 4 directions (Figure 12): (i) Horizontal direction code (direction 0 and 4), (ii) Vertical direction code (direction 2 and 6), (iii) Diagonal direction code (direction 1 and 5) and (iv) Off diagonal direction code (direction 3 and 7). Thus, in each block, four features representing the frequencies of these four directions are obtained. As a result, for each image we obtain 100 (25×4) features from 25 blocks. The reason for choosing an overlapping window-map (one pixel from each side) instead of non-overlapping window-map is to preserve the information between a window-map and its neighbouring blocks. Moreover, based on an experimental study we have extracted feature sets from non-overlapping as well as overlapping blocks. We have tested the system with such features and achieved better accuracy with overlapping blocks when compared with non-overlapping blocks [10].

Furthermore, instead of using contour information of the normalized image, the normalized image is thinned using a thinning algorithm [21] to obtain 100 directional chain code features from the thinned image. The thinned image and an overlapping window-map of size 12×12 are shown in Figure 13 and 14 respectively.



### C. Gradient features

To get gradient features [16], at first, the input image is normalized into 54×54 pixels using aspects ratio of bounding box's height and width and then a 2×2 mean

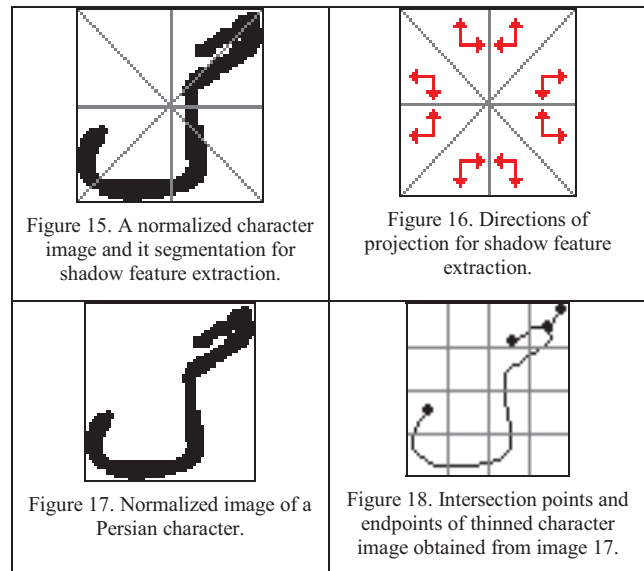
filtering is applied 5 times on the input image. The normalized gray image is segmented into 81 (9×9) blocks of size 6×6. Next, directions of gradient are quantized into 16 directions with 22.5 degree intervals and the strength of the gradient is accumulated with each of the quantized direction. Strength of gradient ( $SG$ ) and direction of gradient ( $\theta$ ) are defined as follows:

$$SG = \sqrt{(\Delta u)^2 + (\Delta v)^2}, \quad \theta(x, y) = \tan^{-1} \frac{\Delta v}{\Delta u}$$

where  $u=f(x+1,y+1)-f(x,y)$ ,  $v=f(x+1,y)-f(x,y+1)$  and  $f(x,y)$  is a gray scale at  $(x, y)$  point. Finally, using Gaussian filter, 9×9 blocks are down sampled into 5×5 blocks and as a result 400 (5×5×16) gradient features are obtained. Details of gradient feature extraction technique can be found in [16].

### D. Shadow features

Shadow is basically the length of projection on a particular direction. To extract shadow features [13], at first, the normalization procedure (as mentioned earlier) is performed to get a normalized character image. Then the normalized image is divided into eight octants as shown in Figure 15. For each octant, shadow of character segment is computed on two perpendicular sides (horizontal and vertical). Thus, 16 set of shadow features are obtained. The 16 directions of projection are shown in Figure 16. Since, in each side 25 values are computed as shadow features, a total of 400 (16×25) shadow features are obtained for each character image.



### E. Intersection/Junction/End points

Intersection features are extracted from thinned character image, which is first normalized into 52×52 pixels. For thinning process, algorithm proposed in [21] is utilized. Thinned character image is then divided into 16 segments each of size 13×13 pixels. For each segment, the number of endpoints and intersection/junctions are found

and counted separately. An intersection point is defined as a pixel point which has more than two neighbouring pixels in 8-connectivity while an endpoint has exactly one neighbour pixel. Figure 18 shows the endpoints and intersection point obtained from a Persian thinned character. Thus, 32 (16×2) features from 16 constituent segments of the character image is computed, out of which the first 16 features represents the number of end points and rest 16 features represents the number of intersection/ junction points.

#### F. Straight Line-Fitting features

A straight-line  $y = a + bx$  is defined by two parameters  $a$  (intercept) and  $b$  (slope), which can be considered as unique properties of a straight-line. To compute these features, at first, normalization procedure is performed to normalize input character image into 52×52 pixels. Employing a thinning algorithm [21] on the normalized character image, a thinned image of character sample is obtained. The thinned image is then segmented into 16 segments of size 13×13. For each segment, a straight line is fitted using least minimum square (LMS) method of line fitting. The values of  $a$  and  $b$  are calculated as follows:

$$a = (\sum(x_i \times y_i) \times \sum x_i - \sum x_i^2 \times \sum y_i) / (\sum x_i \times \sum x_i - N \times \sum x_i^2)$$

$$b = (\sum(x_i \times y_i) \times N - \sum x_i \times \sum y_i) / (N \times \sum x_i^2 - \sum x_i \times \sum x_i)$$

where  $x_i$  and  $y_i$  are the co-ordinates of a pixel located in the thinned image.

Intercept ( $a$ ) feature can be used directly as a feature; however, slope ( $b$ ) cannot be an efficient feature. Straight lines with slopes approaching +0 and -0 has approximately a similar orientation, however, it is necessary to be represented with different values [22]. For the purpose, two features  $f_1$  and  $f_2$  are defined based on value of  $b$  (slope) as follow:

$$f_1 = 2 \times b / (1 + b^2)$$

$$f_2 = (1 - b^2) / (1 + b^2)$$

Since, 16 segments are considered for feature extraction and in each segment an intercept ( $a$ ) and based on slope ( $b$ ) two values of  $f_1$  and  $f_2$  are calculated, a total of 48 (16×3) features are considered as line-fitting features for each character sample.

### III. CLASSIFICATION

In this research work, Nearest Neighbour (NN), 3-NN, 5-NN, and Support Vector Machine (SVM) are utilized for classification of Persian/Arabic characters using different feature sets discussed in Section II.

The NN achieves consistently high performance amongst different techniques of supervised statistical pattern recognition without a priori assumptions about the distributions of training examples. A new sample is classified by determining the nearest distance between the new sample and the training samples. Label of the nearest training sample indicates the label of the new sample.

The  $k$ -NN classifier extends this idea by considering the  $k$  nearest training samples and assigning the label of the majority to the new sample. The value of  $k$  is generally small and odd (3, 5 or 7) to break ties. The distances

between the new sample and training samples can be calculated using Euclidian, Mahalanobis or City-block distance measure. In this paper, the Euclidian distance measure is used for experimentation. Values of 3 and 5 are also used for  $k$  in our experimentation.

The SVM is initially defined for two-class problem and it looks for the optimal hyper-plane, which maximized the distance margin between the nearest examples of both classes, named Support Vector. Different kernels such as linear, polynomial and Gaussian can be used for classification purpose. Detail of SVM can be found in [17].

Here, we tested the proposed systems with the SVMs of different kernels and we received the best result using Gaussian kernel. The proposed architectures of our SVM based classifiers include 32 one-against-other Gaussian kernel SVMs, which were trained using training samples of IFHCDB [18].

### IV. RESULT AND DISCUSSION

For experimentation, a Persian/Arabic handwritten isolated character dataset (called as IFHCDB) [18] is considered. The IFHCDB includes a total of 52020 handwritten character samples in which 36682 samples are considered for training and the rest (15338 characters) is kept for test. The database is not a uniform dataset and hence number of samples in each class is different in both training as well as testing parts [18].

#### A. Performances obtained using different feature sets and classifiers

Using eight different feature sets and four classifiers, we computed 32 results obtained from binary images of Persian/Arabic handwritten characters and the results are tabulated in Table I. A graphical representation of all recognition results is also shown in Figure 19.

From the experiment, we noted that SVM classifier provided the best results among all the classifiers considered in this research work. The best recognition accuracy of 96.91% is obtained using 400 dimensional gradient features and SVM classifier.

The intersection features provided the worse results among all feature sets in all the classifiers used in this study. The lowest recognition accuracy of 55.20% is obtained employing the NN classifier on intersection features. The highest performance (93.13%) among the results obtained employing the NN classifier on different feature sets is achieved using 100 directional features. Shadow features provided the recognition accuracies of 93.30% and 93.27% using 3-NN and 5-NN classifiers, respectively.

From Table I it can be noted that, though gradient features provided the best recognition result (96.91%), a large number of features (400) is used to achieve this higher result. However, using only 25 under-sampled bitmap features a comparatively good recognition accuracy of 94.41% is obtained. Moreover, directional and shadow features are also very efficient.

From Table I, the significance of the directional features obtained from contour points is evident with respect to the directional features computed from skeleton points as well as fusion of directional features obtained from contour and skeleton points.

Table I. DIFFERENT RESULTS OBTAINED WITH 32-CLASS PERSIAN HANDWRITTEN CHARACTERS.

Features	Classifiers	Recognition rates on 32 Classes (%)			
		NN	3-NN	5-NN	SVM
400 Gradient features		92.34	92.93	93.17	<b>96.91</b>
400 Shadow features		92.87	<b>93.30</b>	<b>93.27</b>	95.96
100 Directional features (Contour image)		<b>93.13</b>	93.25	92.61	96.03
100 Directional features (Thinned image)		91.09	91.45	90.86	94.23
100 features (fusion of Directional features of Contour and thinned images)		93.12	93.14	92.56	95.83
48 Line-fitting features		81.19	79.97	78.53	84.12
32 Intersection features		55.20	60.21	60.86	80.66
25 Under-sampled features		92.20	92.80	92.82	94.41

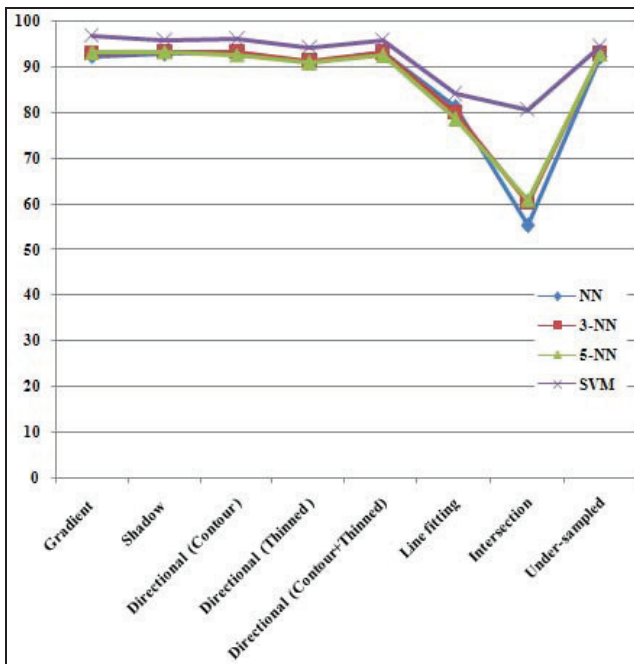


Figure 19. Recognition results obtained from different features and classifiers.

### B. Confusion pairs

A statistical study of all confusion matrices obtained from all recognition systems proposed in this paper proved that most of confusions and misclassifications occurred

between the characters having similar shapes. The major confusions in our experimentations were (ج، ح)، (د، ر)، (ص، س) and (ل، ن). The confusions between (ج، ح)، (د، ر)، (ص، س) and (ل، ن) were 19%, 1.3%, 2.3% and 2.88%, respectively. This happened because these characters look very similar in shape and they differ by only a very small complimentary part such as dot and stroke, which make very little differences between them.

### C. Erroneous samples

Shape and size of dot(s) of a character differ from writing of person to person, therefore discriminating between isolated character classes with the similar basic shape are very complicated task and it needs content-based knowledge to recognize them accurately. This knowledge can be the position of a character in a word and its neighbouring characters located before/after the character in the word.

Some misclassified samples obtained during our experiment are shown in Table II. This table clearly shows that all the errors have resulted in because of very similar shape characters as well as writing styles of different individuals. Some errors also occurred because of bad segmentation of samples from the handwritten forms during data collection.

Table II. SOME ERRORS OF THE PROPOSED SYSTEM (NUMERAL DENOTES THE CLASS LABEL).

Classes	6 ج	10 د	17 ص	27 ل
Test Samples				
Classified Character & its Label	8	12	15	29
Correct Handwritten character & its Class label	6 ج	10 د	17 ص	27 ل

### D. Comparison of results

To have a comparison between the best result obtained in our experimentation and the results of state-of-the-art methods for the recognition of Persian/Arabic handwritten characters we have noted the performances of some existing works in literature and they are shown in Table III. From Table III it may be noted that most of the existing works have been evaluated on individual datasets of smaller in size.

The highest accuracy of 96.91% on 32 classes is obtained from the proposed system utilizing gradient information and SVM based classifiers. Moreover, an improvement of 0.23% in recognition accuracy is obtained from the proposed system

in this paper compared with the system proposed in [10]. Experimental results also demonstrate efficiency of feature set as well as classification technique.

## V. CONCLUSION

In this paper, comparative recognition results of different classifiers and feature sets towards the recognition of Persian/Arabic handwritten isolated characters are reported. This comparative study can provide a new benchmark for future research.

From the experimental results, it is evident that higher recognition accuracy (96.91%) is obtained in this research work. Moreover, most of the misclassified samples are due to similar shape classes. The recognition of such similar shaped characters, which differ only with a small complementary part like dot(s) or stroke, is a complicated task. In future, we plan to use combination of the features and classifiers and also to add some structural features such as number of dot(s) and their positional information for the improvement of the recognition accuracy.

Table III. RESULTS OF DIFFERENT ALGORITHMS

Algorithms	Dataset size		Accuracy (%)	
	Train	Test	No. of Classes	Test
Mozaffari et al. [3]	3200	2880	8	87.26
Dehghan and Faez [4]	1600	1600	20	96.92
Ziaratban et al. [5]	11471	7647	8	93.15
Shanbehzadeh et al. [6]	1800	1200	32	87.00
Dehghani et al. [7]	NA	NA	8	71.82
Mowlaei and Faez [8]	3200	2880	8	93.75
Alaei et al. [10]	36682	15338	32	96.68
<b>Proposed system with 400 gradient features &amp; SVM based classifier</b>	36682	15338	32	96.91

## ACKNOWLEDGMENTS

The authors would like to acknowledge the help received from Mr. Srikanta Pal of Griffith University, Australia.

## REFERENCES

- [1] S. N. Srihari and G. Ball, "An Assessment of Arabic Handwriting Recognition Technology", CEDAR Technical Report TR-03-07, 2007.
- [2] L. M. Lorigo and V. Govindaraju, "Offline Arabic handwriting recognition: a survey", IEEE PAMI, vol. 28, pp. 712-724, 2006.
- [3] S. Mozaffari, K. Faez and H. Rashidy Kanan, "Recognition of Isolated Handwritten Farsi/Arabic Alphanumeric Using Fractal Codes", Image Analysis and Interpretation 16th Southwest Symposium, pp. 104-108, 2004.
- [4] M. Dehghan and K. Faez, "Farsi Handwritten Character Recognition With Moment Invariants", Proceedings of 13th International Conference on Digital Signal Processing, Vol. 2, pp. 507-510, 1997.
- [5] M. Ziaratban, K. Faez and F. Allahveiradi, "Novel Statistical Description for the Structure of Isolated Farsi/Arabic Handwritten Characters", Proceedings of the 11<sup>th</sup> ICFHR, pp. 332-337, 2008.
- [6] J. Shanbehzadeh, H. Pezashki, A. Sarrafzadeh, 'Features Extraction from Farsi HandWritten Letters', Proceedings of Image and Vision Computing, pp. 35-40, 2007.
- [7] A. Dehghani F. Shabani P. Nava, "Off-line Recognition of Isolated Persian Handwritten Characters Using Multiple Hidden Markov Models", Proceedings of Int'l Conf. on Information Technology: Coding and Computing, pp. 506-510, 2001.
- [8] A. Mowlaei and K. Faez, "Recognition of Isolated Handwritten Persian/Arabic Characters and Numerals Using Support Vector Machines", Proceedings of XIII Workshop on Neural Networks for Signal Processing, pp. 547-554, 2003.
- [9] Saeed Mozaffari and Hadi Soltanizadeh "ICDAR Handwritten Farsi/Arabic Character Recognition Competition", Proceedings of 10th ICDAR, pp.1413-1417, 2009.
- [10] Alireza Alaei, P. Nagabhushan and Umпада Pal, "A New Two-stage Scheme for the Recognition of Persian Handwritten Characters," In Proc. Twelfth International Conference on Frontiers in Handwriting Recognition (ICFHR10), pp. 130-135, 2010.
- [11] Garris, M. D. et al., "NIST Form-Based Handprint Recognition System", NISTIR 5469, 1994.
- [12] Alpaydin, E., Kaynak, C., "Cascading Classifiers", Kybernetika, 34(4), pp. 369-374, 1998.
- [13] D. J. Burr, "Experiments on neural net recognition of spoken and written text", IEEE Trans. Acoust., Speech, Signal Processing, 36(7), pp. 1162-1168, 1988.
- [14] F. Kimura, T. Wakabayashi, S.Tsuruoka and Y. Miyake, "Improvement of handwritten Japanese character recognition using weighted direction code histogram", Pattern recognition, 30(8), pp. 1329-1337, 1997.
- [15] Alireza Alaei, P. Nagabhushan and Umпада Pal, "Fine Classification of Unconstrained Handwritten Persian/Arabic Numerals by Removing Confusion amongst Similar Classes," Proceedings of 10th ICDAR, pp. 602-606, 2009.
- [16] M. Shi, Y. Fujisawa, T. Wakabayashi, and F. Kimura, "Handwritten numeral recognition using gradient and curvature of gray scale images", Pattern Recognition, Vol.35, pp.2051-2059, 2000.
- [17] V. Vapnik, "The Nature of Statistical Learning Theory", Springer Verlag, 1995.
- [18] S. Mozaffari, K. Faez, F. Faradji, M. Ziaratban and S.M. Golzan "A comprehensive isolated Farsi/Arabic character database for handwritten OCR research", Proceedings of 10th IWFHR, pp. 385-389, 2006.
- [19] C. L. Liu and C. Y. Suen, "A new benchmark on the recognition of handwritten Bangla and Farsi numeral characters", In Proc. 11th ICFHR, pp. 1- 6, 2008.
- [20] C.-L. Liu, K. Nakashima, H. Sako, H. Fujisawa, "Handwritten digit recognition: Benchmarking of state-of-the-art techniques," Pattern Recognition, 36 (10), pp. 2271-2285, 2003.
- [21] Rafael C. Gonzalez, Richard E. Woods, Digital Image Processing, Second Edition, Prentice Hall India, 2002.
- [22] S. Arora, D. Bhattacharjee, M. Nasipuri, D. K. Basu, M. Kundu, "Combining Multiple Feature Extraction Techniques for Handwritten Devnagari Character Recognition," Third international Conference on Industrial and Information Systems, pp. 1- 6, 2008.