A New Benchmark on the Recognition of Handwritten Bangla and Farsi Numeral Characters

Cheng-Lin Liu

National Laboratory of Pattern Recognition (NLPR), Institute of Automation, Chinese Academy of Sciences, PO Box 2728, Beijing 100190, P.R. China
E-Mail: liucl@nlpr.ia.ac.cn

Ching Y. Suen

Center for Pattern Recognition and Machine Intelligence (CENPARMI), Computer Science Department, Concordia University, 1455 de Maisonnneuve Blvd. West, Montreal, Quebec H3G 1M8, Canada
E-Mail: suen@cs.concordia.ca

Abstract

The recognition of Indian and Arabic handwriting is drawing increasing attention in recent years. To test the promise of existing handwritten numeral recognition methods and provide new benchmarks for future research, this paper presents some results of handwritten Bangla and Farsi numeral recognition on binary and gray-scale images. On proper pre-processing, feature extraction and classification, we achieved very high accuracies on three databases: ISI Bangla numerals, CENPARMI Farsi numerals, and IFHCDB Farsi numerals. The benefit of recognition on gray-scale images is justified.

Keywords: Bangla numeral recognition; Farsi numeral recognition; Pre-processing; Feature extraction; Classification.

1. Introduction

In last decades, most attention of character recognition was paid to English and Chinese/Japanese characters. Particularly, the recognition of Latin numeral characters has attracted much attention because it is a handy case for testing various techniques (pre-processing, feature extraction, and classification) and has many applications. Many effective recognition methods have been proposed and very high accuracies have been reported [1, 2, 3, 4].

In recent years, the recognition of Indian and Arabic scripts is receiving increasing attention [5, 6]. In India, the most frequently used scripts include Devenagari, Bangla, Tamil, Oriya, and so on. Arabic and Farsi (Persian) scripts are mainly used in the Middle East and they are very similar to one another. For such scripts, some researchers have designed specific methods while some others use existing generic character recognition methods. To promote the research in this field, some handwriting databases have been released, and competitions of Arabic handwriting recognition and Tamil character recognition have been held.

This paper focuses on the recognition of Indian and Arabic numerals, particularly, Bangla and Farsi numeral recognition because some public databases are available [7, 8, 9]. Though some research works have contributed to Bangla numeral recognition [10, 11, 12] and Farsi numeral recognition [13], they rarely used common sample databases, and some of the reported accuracies are not very high. Thus, this work aims to provide new benchmarks for fair comparison of recognition methods on standard databases.

We evaluate our recognition methods on three databases: ISI Bangla numerals [7], CENPARMI Farsi numerals [8], and IFHCDB Farsi numerals [9]. The methods of image pre-processing, feature extraction and classification have shown superior performance in handwritten Latin numeral recognition [2, 3]. The character images in the three databases are gray-scaled with 256 levels. We use proper pre-processing techniques, including thresholding and gray level normalization, to overcome gray level variations. Our methods yield very high recognition accuracies on the databases, and the benefit of recognition on gray-scale images is justified.

2. Previous Works

Bhattacharya and Chaudhuri proposed a multiresolution wavelet analysis and majority voting approach and applied to handwritten Bangla numeral recognition [10]. They combine three MLP neural networks on wavelet features of three resolutions by majority vote and reported a correct rate of 97.16% with 0.76% rejection on 5,000 test samples. Wen et al. combine three recognizers by majority vote, one of them is based on Kirsh gradient (four orientations), dimensionality reduction by PCA and classification by SVM (support vector machine). They reported a correct rate of 95.05% with 0.93% error on 10,000 test samples. Pal et al. reported results of handwritten numeral recognition of six Indian scripts, including Bangla [12]. They use quadratic classifier on 16-direction gradient histogram feature extracted by Roberts masks. By 5-fold cross validation on 14,650 samples, they obtained a high accuracy of 98.99%.

For handwritten Farsi numeral recognition, Soltanzadeh et al. extract outer profile features [13] and use one-versus-all SVM classifiers, with polynomial and radial basis function (RBF) kernels. On 3,939 test samples, two SVM classifiers gave accuracies of 99.44% and 99.57%. Solimanpour et al. tested the same method of [13] on the Farsi handwriting database of CENPARMI [8]. Using the verifying data for selecting kernel parameters for SVM classifier, they obtained a test accuracy of 97.32%. This lower accuracy indicates that the samples in CENPARMI database are more challenging than those in [13].

3. Databases

We evaluate our recognition methods on three databases: ISI Bangla numerals [7][1], CENPARMI Farsi numerals [8], and IFHCDB Farsi numerals [9][2].

The ISI Bangla numeral database has 19,392 training samples and 4,000 test samples in total. The images are gray-scaled, some with noisy background, and the gray level of foreground (stroke regions) varies considerably. In the CENPARMI Farsi numeral database, each of 10 classes has 1,100 samples for training, 200 samples for verifying, and 500 samples for testing. The images are gray-scaled, mostly with clean background, but the gray level of foreground is considerably variable. The IFHCDB Farsi database has 17,740 numeral images in total. For classes ‘4’ and ‘6’ each has two different shapes, but we exclude the samples of minor shapes in our experiments. On doing so, we have 12,292 samples for training and 5,268 samples for testing. Some samples of the three databases are shown in Fig. 1.

4. Recognition Methods

Each sample undergoes pre-processing of image, feature extraction, and classification. The feature vectors of training samples are used to learn the parameters of a classifier.

4.1 Pre-Processing

We use proper thresholding, gray level normalization and size normalization techniques to remove recognition-

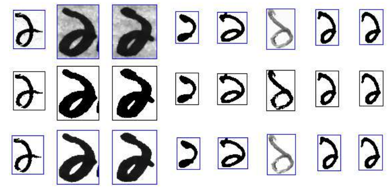

Figure 2. Examples of thresholding. Upper: original image; Middle: binarized; Lower: background eliminated.

irrelevant information. The major role of thresholding is to remove the background noise. By thresholding a gray-scale image, we can obtain both the binary image (with level 1 for foreground and level 0 for background) and the background-eliminated gray-scale image. For selecting the threshold, we use the classical algorithm of Otsu [14], which performs satisfactorily (Fig. 2).

The thresholded gray-scale images have variable foreground gray levels over different samples. To eliminate the dependence of feature values on gray levels, we re-scale the gray levels of foreground pixels of each sample into a standard mean of 210 and deviation of 20.

For size normalization, the most popular method is the linear normalization (LN): bounding the character strokes with a rectangle and linearly mapping the rectangle into a standard size (usually a square). Better performance can be achieved using moment normalization (MN) [3], which aligns the centroid (center of gravity) of character image with the geometric center of normalized plane, and re-bounds the character according to second-order moments. We also evaluate a nonlinear moment-based normalization method, called bi-moment normalization (BMN) [15], which uses two quadratic functions to map pixel coordinates. The MN and BMN methods have been shown to perform superiorly [3].

The above normalization methods apply to both binary and gray-scale images. A binary image can be normalized to either a binary image or a gray-scaled (we call pseudo-gray) image [3]. To alleviate the deformation of normalization for elongated characters, the aspect ratio of character is controlled by an aspect ratio mapping function (square root of sine) [3]. Some examples of gray level normalization and size normalization (by three methods) are shown in Fig. 3.

4.2 Feature Extraction

Various up-to-date character features have been evaluated in [2, 3]. The feature of local stroke orientation/direction distribution is widely used and is among the best for character recognition of various scripts. This type of feature is described by a concatenation of localized histograms of orientation/direction elements, from

[1] Indian Statistical Institute (ISI), Kolkata, India, http://www.isical.ac.in/ ujjwal/download/database.html

[2] AmirKabir University of Technology, Tehran, Iran, http://ele.aut.ac.ir/imageproc/downloads/IFHCDB.htm Contact Prof. Karim Faez (kfaez@aut.ac.ir)

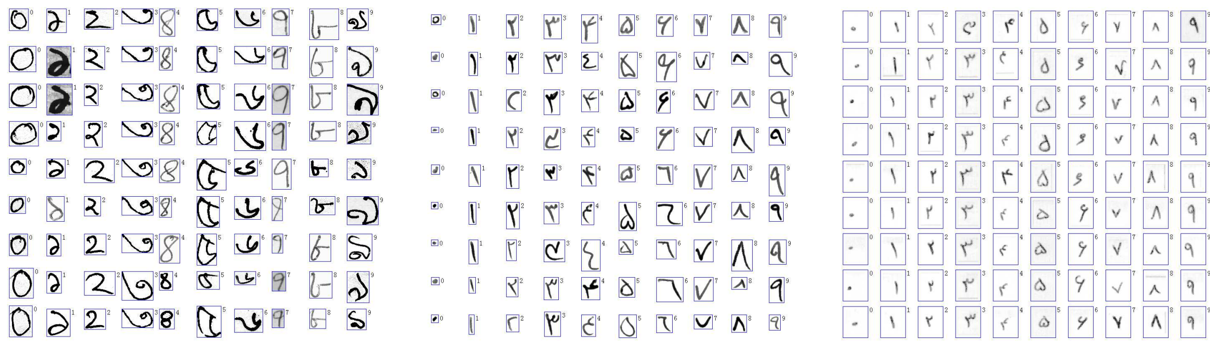

Figure 1. Samples of three numeral databases (from left to right): ISI Bangla, CENPARMI Farsi, IFHCDB Farsi.

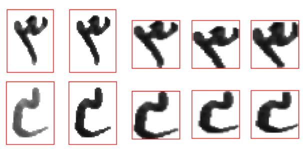

Figure 3. Examples of gray level normalization and size normalization. From left: background eliminated, gray level normalized, linear size normalized, moment normalized, bi-moment normalized.

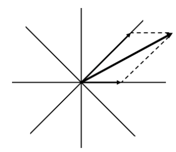

Figure 4. Eight discrete directions and decomposition of gradient vector.

either stroke contour or gradient (applies to both binary and gray-scale images). The gradient histogram feature generally outperforms the contour (chaincode) histogram feature and 8-direction histogram features generally outperform 4-orientation histogram features.

Based on the above situation, we take the gradient direction histogram feature. The gradient can be calculated using different operators. Kirsh and Roberts masks have been applied to Bangla numeral recognition by Wen et al. [11] and Pal et al. [12], respectively. Kirsh masks directly calculate the gradient magnitude of four orientations, whereas Roberts masks (2×2 neighborhood) and Sobel masks (3×3 neighborhood) calculate the x- and y- components of local gradient vector. We use the Sobel operator, which outperforms the Kirsh masks [2], and will compare the Sobel operator with the Roberts operator.

For calculating local orientation/direction histograms, the gradient vector at each pixel is assigned to discrete directions either by tangent angle quantization or parallelogram decomposition. The latter, decomposing a vector into two directions using the parallelogram rule, can minimize the quantization error. Fig. 4 shows the eight discrete directions and the decomposition of a gradient vector. The number of discrete directions can be easily extended to 12, 16, or any other number.

On decomposing the gradient vectors of all pixels into discrete directions, feature values are extracted from each direction map by zoning or low-pass filtering with down-sampling. Low-pass filtering (also called blurring) can better deal with the translation of strokes. We use the Gaussian filter and relate its variance parameter to the sampling interval according to the Nyquist Sampling Theorem [2]:

$$\sigma_x = \frac{\sqrt{2}t_x}{\pi},$$

where σ_x is the standard deviation of Gaussian function, and t_x is the sampling interval.

4.3 Classification

We use six types of classifiers for classification on the character features: MLP neural network, modified quadratic discriminant function (MQDF) [16], discriminative learning quadratic discriminant function (DLQDF) [17], polynomial network classifier (PNC) [18], class-specific feature polynomial classifier (CFPC) [19], and one-versus-all SVM classifier [20]. The MLP, MQDF, DLQDF, PNC and SVM classifiers have been evaluated for handwritten Latin numeral recognition in [2].

We use the MLP with one hidden layer, with the connecting weights estimated by the error back-propagation (BP) algorithm minimizing the squared error criterion. The PNC is a single-layer network with the linear and polynomial (up to second order in our case) terms of features as the inputs. To reduce the complexity, the number of features is reduced by PCA. The CFPC uses class-specific PCA subspace features for each class, and incorporates the subspace reconstruction error as well.

The MQDF is the modification of ordinary QDF by replacing the minor eigenvalues of each class with a constant, and so, results in reduced complexity and improved generalization performance. We optimize this constant via 5-fold cross validation on the training data set. The DLQDF has the same structure as the MQDF, with the parameters optimized by discriminative learning under the minimum classification error (MCE) criterion [21].

For the SVM classifier, the selection of kernel function is important. The polynomial kernel and the RBF kernel are popular. Our experience tells us that in character recognition, the SVM with RBF kernel mostly outperforms that with polynomial kernel. In our experiments, the variance parameter of the RBF kernel is set to be proportional to the average within-class variance (square Euclidean distance to class mean). This way of parameter selection works satisfactorily.

5. Results and Discussions

We combine three types of normalized images, three size normalization methods, and six classifiers in handwritten Bangla and Farsi numeral recognition. The three image types are: binary image normalized from binary image, gray-scale image normalized from binary image (pseudo-gray image), and gray-scale image normalized from gray-scale image. The three size normalization methods are: linear normalization (LN), moment normalization (MN), and bi-moment normalization (BMN).

For extracting 8-direction gradient feature, the size of normalized image is set to 35×35 pixels. On each of eight direction maps, 5×5 feature values are extracted by Gaussian filtering and down-sampling. The resulting feature vector is 200D. We also tried 12-direction and 16-direction features to verify the optimum resolution of direction decomposition, and found that they did not outperform the 8-direction feature in this work. For saving space, we do not list the recognition results of 12-direction and 16-direction features here.

We also compared three schemes for gradient direction computation: Sobel operator with vector decomposition (Fig. 4), Roberts operator with vector decomposition, and Roberts operator with tangent angle partitioning. The average accuracies of three schemes on nine image-normalization combinations are shown in Table 1. We can see that the Sobel operator with vector decomposition is indeed the best choice.

Table 1. Average accuracies (%) of three gradient direction computation schemes.

	MQDF	DLQDF	PNC	CFPC
Sobel-vector	98.47	99.13	99.07	99.16
Roberts-vector	98.31	99.08	99.07	99.05
Roberts-tagent	98.15	99.94	98.94	98.98

The classifier structures were empirically set as follows. The MLP has one hidden layer of 100 nodes. The MQDF and DLQDF classifiers use 40 principal eigenvectors per class. The MQDF has a parameter estimated by 5-fold cross validation on the training set, and the DLQDF inherits initial parameters from the MQDF. The PNC uses 70D PCA subspace, and the CFPC uses 50D class-specific PCA subspaces. The SVM classifier uses an RBF kernel with the variance parameter equal to 0.5 times the average within-class sample variance.

The performance of MLP is sensitive to the initialization of weights. For the CENPARMI database, we trained with three different sets of initial weights, and selected the one performing best on the verifying data. On the other two databases, the network was trained three times on 4/5 of training data and evaluated on the remaining 1/5 data. The network with the best initial weights is then re-trained on the whole training set.

The recognition accuracies on the test sets of three databases are shown in Tables 2, 3 and 4, respectively. In these Tables, the highest accuracy of each row over different classifiers is highlighted, and the average accuracy of each column is given. Since the accuracies of the latter four classifiers (DLQDF, PNC, CFPC, SVM) are comparable and are remarkably higher than MLP and MQDF, we also give the average accuracy of the four high-accuracy classifiers (denoted by "Aver{3-6}") for comparing the image types and normalization methods.

We have some observations from the results. For comparing the image types, we focus on the average accuracies of four classifiers ("Aver{3-6}"). We can see that except two of 18 cases, for all the three databases and three normalization methods, the average accuracies of recognition on pseudo-gray normalized images are higher than those on binary normalized images, and the average accuracies of recognition on gray-scale images are higher than those on pseudo-gray images. Comparing the normalization methods, we can see that except five of 18 cases, the average accuracies of moment normalization are higher than those of linear normalization and the average accuracies of bi-moment normalization are higher than those of moment normalization. Comparing the classifiers, the SVM (with RBF kernel) gives the highest accuracy in 12 of 27 cases, the CFPC gives the highest accuracy in seven cases, the DLQDF in five cases, the PNC in three cases, and the MQDF in one case.

For comparing with previous results, we found only one published result evaluated on the same database with us. Solimanpour et al. tested the same method of [13] on the Farsi handwriting database of CENPARMI [8] and obtained a test accuracy of 97.32%. Our accuracies on the same test set using four high accuracy classifiers are mostly higher than 98.80%, and the highest one is 99.16%, achieved on gray-scale images.

Table 2. Test accuracies (%) of 8-direction features on ISI Bangla numerals.

Image	Norm	MLP	MQDF	DLQDF	PNC	CFPC	SVM	Aver{3-6}
Binary	LN	98.35	98.05	99.00	98.85	98.83	**99.08**	98.99
	MN	98.28	98.25	98.85	98.72	98.92	**98.97**	98.87
	BMN	98.75	98.50	99.00	99.15	99.17	**99.20**	99.13
P-gray	LN	98.58	98.17	98.97	98.92	**99.08**	99.00	98.99
	MN	99.00	98.75	99.28	99.13	**99.35**	99.22	99.25
	BMN	98.97	98.78	99.28	99.30	**99.38**	99.30	99.32
Gray	LN	98.60	98.22	**99.13**	99.03	98.95	99.10	99.05
	MN	98.80	98.67	99.30	99.25	**99.40**	99.25	99.30
	BMN	98.90	98.83	**99.40**	99.30	99.35	99.30	99.34
Average		98.69	98.47	99.13	99.07	99.16	99.16	

Table 3. Test accuracies (%) of 8-direction features on CENPARMI Farsi numerals.

Image	Norm	MLP	MQDF	DLQDF	PNC	CFPC	SVM	Aver{3-6}
Binary	LN	98.32	97.60	98.74	98.76	98.76	**98.78**	98.76
	MN	98.30	97.94	98.80	98.76	98.74	**98.96**	98.82
	BMN	98.16	97.98	98.76	**98.86**	98.78	**98.86**	98.82
P-gray	LN	98.60	97.78	98.82	98.86	98.86	**98.90**	98.86
	MN	98.48	98.26	98.96	98.78	98.98	**99.04**	98.94
	BMN	98.42	98.36	98.92	**98.96**	98.82	98.94	98.91
Gray	LN	98.54	97.80	98.86	**99.00**	98.90	98.86	98.91
	MN	98.56	98.42	98.92	98.98	**99.16**	99.04	99.03
	BMN	98.82	98.42	99.00	98.98	99.06	**99.10**	99.04
Average		98.47	98.06	98.86	98.88	98.90	98.94	

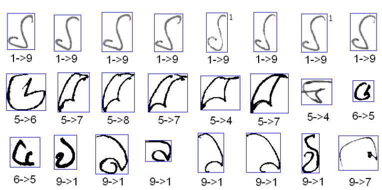

Figure 5. Misclassified samples of ISI Bangla numeral database.

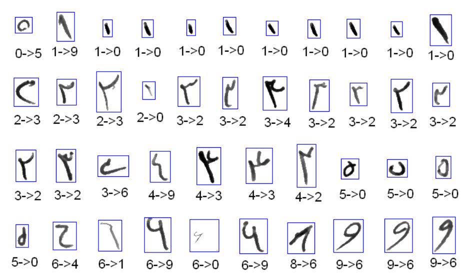

Figure 6. Misclassified samples of CENPARMI Farsi numeral database.

The misclassified test samples of ISI Bangla and CENPARMI Farsi, by CFPC on gray-scale image with moment normalization, are shown in Fig. 5 and Fig. 6, respectively. We can see that in Bangla numerals, there are many confusions between '1' and '9', and some peculiarly written samples of '5' are misclassified. In Farsi numerals, quite a few samples of '1' and '5' are misclassified as '0', some samples of '3' are wrongly classified as '2', and '6' and '9' are frequently confused. The misclassification of '1' as '0' occurs because '0' has many training samples with filled loops, while some samples of '1' are rather wide. The confusion between '5' and '0' is attributed to the loop in '5'. Combining multiple recognizers can reduce the number of misclassifications and produce some rejections, but this is not the aim of this work.

6. Conclusion

In this work, we applied some advanced character recognition methods to handwritten Bangla and Farsi numeral recognition and evaluated the performance on three public databases: ISI Bangla numerals, CENPARMI Farsi numerals, and IFHCDB Farsi numerals. We justified the benefit of recognition on gray-scale images and the superiority of moment normalization and bi-moment normalization. The feature used is the gradient direction histogram feature, and four classifiers (DLQDF, PNC, CFPC and SVM) yield very high accuracies. The highest test accuracies on the three databases are 99.40%, 99.16%, and 99.73%, respectively. These databases were recently published and have not been widely evaluated by the community, except that a previous work reported an accuracy of 97.32% on the CENPARMI Farsi numerals. Thus, this work provides a new benchmark.

To further improve the recognition performance, possible future works are as follows. (1) Although the gradient direction feature performs superbly overall, comple-

Table 4. Test accuracies (%) of 8-direction features on IFHCDB Farsi numerals.

Image	Norm	MLP	MQDF	DLQDF	PNC	CFPC	SVM	Aver{3-6}
Binary	LN	99.28	99.32	99.58	99.56	**99.62**	99.56	99.58
	MN	99.43	99.53	99.54	99.51	99.49	**99.62**	99.54
	BMN	99.41	99.56	99.64	99.62	99.56	**99.66**	99.62
P-gray	LN	99.41	99.35	99.62	99.58	**99.64**	99.56	99.60
	MN	99.49	99.53	99.66	99.62	99.58	**99.70**	99.64
	BMN	99.45	99.60	**99.70**	99.60	99.64	99.68	99.66
Gray	LN	99.45	99.49	**99.66**	99.53	99.62	99.62	99.61
	MN	99.49	**99.62**	99.60	99.60	99.58	99.60	99.60
	BMN	99.39	99.70	**99.73**	99.68	99.70	99.70	99.70
Average		99.42	99.52	99.64	99.59	99.60	99.63	

mentary features may help discriminating confusing characters. For example, Farsi numerals '0' and '1' can be better separated by considering the original size before normalization, slant and inertia ratio. (2) For classifier design, it is better to select model parameters (classifier structures) by cross validation rather than empirically as in our experiments. (3) Combining multiple classifiers can improve the recognition accuracy or rejection capability. The classifiers to be combined can use various techniques of normalization and feature extraction, have different structures or even classify different sets of classes.

References

[1] C.Y. Suen, K. Liu, N.W. Strathy, "Sorting and recognizing cheques and financial documents," *Document Analysis Systems: Theory and Practice*, S.-W. Lee and Y. Nakano (eds.), LNCS 1655, Springer, 1999, pp. 173-187.

[2] C.-L. Liu, K. Nakashima, H. Sako, H. Fujisawa, "Handwritten digit recognition: Benchmarking of state-of-the-art techniques," *Pattern Recognition*, 36 (10): 2271-2285, 2003.

[3] C.-L. Liu, K. Nakashima, H. Sako, H. Fujisawa, "Handwritten digit recognition: Investigation of normalization and feature extraction techniques," *Pattern Recognition*, 37 (2): 265-279, 2004.

[4] P. Zhang, T.D. Bui, C.Y. Suen, "A novel cascade ensemble classifier system with a high recognition performance on handwritten digits," *Pattern Recognition*, 40(12): 3415-3429, 2007.

[5] A. Amin, "Off-line Arabic character recognition: The state of the art," *Pattern Recognition*, 31: 517-530, 1998.

[6] U. Pal, B.B. Chaudhuri, "Indian script character recognition: A survey," *Pattern Recognition*, 37 (9): 1887-1899, 2004.

[7] B.B. Chaudhuri, "A complete handwritten numeral database of Bangla - A major Indic script," *Proc. 10th IWFHR*, La Baule, France, 2006, pp.379-384.

[8] F. Solimanpour, J. Sadri, C.Y. Suen, "Standard databases for recognition of handwritten digits, numerical strings, legal amounts, letters and dates in Farsi language," *Proc. 10th IWFHR*, La Baule, France, 2006, pp.3-7.

[9] S. Mozaffari, K. Faez, F. Faradji, M. Ziaratban, S.M. Golzan, "A comprehensive isolated Farsi/Arabic character database for handwritten OCR research," *Proc. 10th IWFHR*, La Baule, France, 2006, pp.385-389.

[10] U. Bhattacharya, B.B. Chaudhuri, "A majority voting scheme for multiresolution recognition of handprinted numerals," *Proc. 7th ICDAR*, Edinburgh, Scotland, 2003, pp.16-20.

[11] Y. Wen, Y. Lu, P. Shi, "Handwritten Bangla numeral recognition system and its application to postal automation," *Pattern Recognition*, 40 (1): 99-107, 2007.

[12] U. Pal, T. Wakabayashi, N. Sharma, F. Kimura, "Handwritten numeral recognition of six popular Indian scripts," *Proc. 9th ICDAR*, Curitiba, Brazil, 2007, Vol.2, pp.749-753.

[13] H. Soltanzadeh, M. Rahmati, "Recognition of Persian handwritten digits using image profiles of multiple orientations," *Pattern Recognition Letters*, 25 (14): 1569-1576, 2004.

[14] N. Otsu, "A threshold selection method from gray-level histogram," *IEEE Trans. System Man Cybernet.*, 9: 62-66, 1979.

[15] C.-L. Liu, H. Sako, H. Fujisawa, "Handwritten Chinese character recognition: Alternatives to nonlinear normalization," *Proc. 7th ICDAR*, Edinburgh, Scotland, 2003, pp.524-528.

[16] F. Kimura, K. Takashina, S. Tsuruoka, Y. Miyake, "Modified quadratic discriminant functions and the application to Chinese character recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, 9 (1): 149-153, 1987.

[17] C.-L. Liu, H. Sako, H. Fujisawa, "Discriminative learning quadratic discriminant function for handwriting recognition," *IEEE Trans. Neural Networks*, 15 (2): 430-444, 2004.

[18] U. Kreßel, J. Schürmann, "Pattern classification techniques based on function approximation," *Handbook of Character Recognition and Document Image Analysis*, H. Bunke and P.S.P. Wang (eds.), World Scientific, 1997, pp.49-78.

[19] C.-L. Liu, H. Sako, "Class-specific feature polynomial classifier for pattern classification and its application to handwritten numeral recognition," *Pattern Recognition*, 39 (4): 669-681, 2006.

[20] C.J.C. Burges, "A tutorial on support vector machines for pattern recognition," *Knowledge Discovery and Data Mining*, 2 (2): 1-43, 1998.

[21] B.-H. Juang, S. Katagiri, "Discriminative learning for minimum error classification," *IEEE Trans. Signal Processing*, 40 (12): 3043-3054, 1992.