Direction Code Based Features for Recognition of Online Handwritten Characters of Bangla

Ujjwal Bhattacharya, Bikash K. Gupta and Swapan K. Parui
Computer Vision and Pattern Recognition Unit, Indian Statistical Institute, Kolkata-108, India
ujjwal@isical.ac.in, bikash_gupta11@yahoo.com, swapan@isical.ac.in

Abstract

In the present article, we describe a novel direction code based feature extraction approach for recognition of online Bangla handwritten basic characters. We have implemented the proposed approach on a database of 7043 online handwritten Bangla (a major script of the Indian subcontinent) character samples, which has been developed by us. This is a 50-class recognition problem and we achieved 93.90% and 83.61% recognition accuracies respectively on its training and test sets.

1. Introduction

With the spread of computers in all parts of the world including developing countries like India, useful user interfaces have gained enormous importance as well as popularity. Online handwriting recognition system is such an interface. Requirement of such recognition technology has gained further importance with easy availability of personal and portable computing devices such as Tablet PCs, PDAs etc. at affordable prices. A few systems for online recognition of handwriting are currently available. However, these systems take care of only a few scripts of the developed nations. No such system for online recognition of handwriting in an Indian script is available.

In the case of online handwriting, the data consists of a sequence of (x, y) coordinates along the trajectory of the pen together with the information related to pen-up and pen-down conditions. Thus, a sample online data consists of a temporal sequence of strokes used to write the character and, in turn, each stroke consists of a temporal sequence of (x, y) coordinates in pen-down condition. During the last few decades, significant progress has been made towards the development of both online and offline handwriting recognition technologies for different scripts. Related review works can be found in [1- 4].

India is a multilingual country of more than 1 billion population with 18 constitutional languages and 10 different scripts. However, there does not exist sufficient work on online handwriting recognition of any Indian script. To the best of our knowledge, there exists only one work [5] on recognition of on-line handwritten numerals and another work [6] for online handwritten alphabetic characters of Bangla, the second most popular language and script of India. It is also the official language of Bangladesh, a neighbour of India.

Existing approaches for online handwriting recognition include template matching based methods [6], fuzzy rule-based methods [7], statistical methods among a few others. In the present article, we propose a recognition scheme for online handwritten characters based on local chaincode histogram feature and multilayer perceptron (MLP) classifier. Local chaincode histogram feature [8] had been used for offline recognition of handwriting in different scripts. Recently, a comparative study of this off-line feature *vis-a-vis* several other popular offline features used for handwriting recognition had been reported in [9]. Also, this chaincode feature has recently been studied for offline recognition of handwritten characters of a few Indian scripts [10].

However, the above local chaincode histogram feature cannot be directly computed for online handwritten data which is sequential in nature. In the present paper, we propose a new approach for obtaining similar features of online handwritten characters.

Since there does not exist a standard relevant database, we developed a representative database of online handwritten characters of Bangla. This database may be obtained free of cost by fellow researchers for academic purposes. The proposed recognition method has been trained and tested using the present database. Recognition accuracies obtained by the proposed method are encouraging.

2. Handwritten Bangla character Database

2.1. Bangla script

All major Indian scripts including Bangla are mixtures of syllabic and alphabetic scripts. They are varied in character and form. Like most of the Indian languages the script of Bangla came from the ancient Indian script, Brahmi. This script runs from left to right and it has no equivalent to capital letters of Latin scripts. The set of basic characters of Bangla consist of 11 vowels and 40 consonants. However, since the shapes of two consonant characters are the same, there are 50 different shapes in the Bangla basic character set. Ideal (printed) forms of these 50 shapes of Bangla basic characters are shown in Fig.1.

অ (A)	141	আ (AA)	139	ই (I)	140	ঈ (II)	141	উ (U)	141
ঊ (UU)	138	এ (E)	141	ঐ (AI)	140	ও (O)	141	ঔ (AU)	141
ঋ (R)	141	ক (KA)	141	খ (KHA)	141	গ (GA)	141	ঘ (GHA)	141
ঙ (NGA)	141	চ (CA)	141	ছ (CHA)	141	জ (JA)	141	ঝ (JHA)	141
ঞ (NYA)	141	ট (TTA)	141	ঠ (TTHA)	141	ড (DDA)	141	ঢ (DDHA)	141
ণ (NNA)	141	ত (TA)	141	থ (THA)	141	দ (DA)	141	ধ (DHA)	141
ন (NA)	141	প (PA)	141	ফ (PHA)	141	ব (BA)	141	ভ (BHA)	141
ম (MA)	141	য (YA)	141	র (RA)	141	ল (LA)	141	শ (SHA)	141
ষ (SSA)	141	স (SA)	141	হ (HA)	141	ড় (RRA)	141	ঢ় (DHRA)	141
য় (YYA)	141	ং (ANUS)	141	ঃ (VISARG)	141	ঁ (BINDU)	141	ৎ (KHAND)	141

Figure 1. Ideal shapes of Bangla basic characters, their pronunciations and the number of samples in the present database.

The difficulty in automatic recognition of these handwritten characters of Bangla arises from the facts that this is a moderately large symbol set, shapes are usually extremely cursive even when written separately and there exist quite a few groups of almost similar shape characters in their handwritten forms. Examples of such groups of characters consisting of two or more characters with significant shape similarity are shown in Fig. 2.

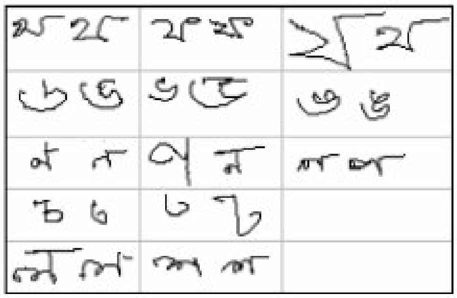


Figure 2. Sets of confusing characters (two samples in each character class are shown).

2.2. Database of online handwritten Bangla characters

In the present work, we have developed a database of online handwritten isolated Bangla characters. These samples have been collected using WACOM Intuous 2 tablet. The maximum sampling rate is 200 points per second. No restriction was imposed on the writers except for specifying rectangular regions for writing each character sample. Each person was asked to write samples at two to five different points of time. This is to capture the variation in one's handwriting style that may occur from one point of time to another. Each sample is stored as a text file consisting of the (x, y) coordinates and pressure among a few other information of the pen tip along its trajectory. Here, we considered only the coordinates of the positions of pen tip having non-zero pressure. The pen up and pen down situations are identified in terms of zero and non-zero pressures.

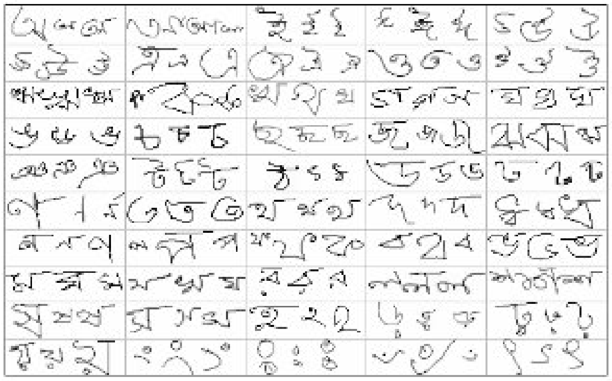


Figure 3. Samples from the present database

The present database consists of 7043 samples of online handwritten Bangla basic characters. The numbers of samples in each class are shown in Fig.1. 114 persons of different groups with respect to age, sex and level of education have written these samples. Each person provided one to three samples in each category. We have taken 100 samples randomly from each class to form the training set. The rest of the samples form the test set. A few samples from this database are shown in Fig.3.

Although online (temporal) data carry more information than offline samples, such temporal information at the same time introduces additional variability (difference in number of strokes and the order of writing) to the handwritten samples. These situations are very frequent in the character classes ঊ, আ, ঋ, শ, ল and জ.

3. Proposed Methodology

A part of the pen trajectory between a pen-down to the next pen-up situations is called a stroke. Most of the character classes of the present database have samples which are written using one as well as more than one strokes. However, there are a few classes, for example, ঊ (UU), র (RA), ং (ANUS) etc., in which all the samples are written using multiple strokes.

3.1. Preprocessing

In the preprocessing stage, initially, the points forming an input character shape are re-sampled by removing the redundant points (multiple occurrences of the same coordinates due to holding down the pen at one or more positions on its trajectory).

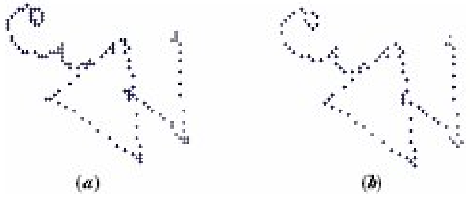


Figure 4. (*a*) Sampling points of an input character with 3 strokes, (*b*) The same character after first resampling.

Next, the points (save for the critical points where the pressure changes from zero to non-zero or *vice-versa*) forming the input character are re-sampled to obtain a new sequence of similar points, which are approximately equidistant. To accomplish this, we compute the modal value of the distribution of the distances between consecutive points along all the stretches (strokes) corresponding to the pen-down situations. If the distance between any two consecutive points of the sequence is less than the above modal distance, then the latter point is removed. This helps to reduce the variations in writing speeds along different parts of an individual character. In Fig. 4, all the points of an input character ঋ(R), consisting of three strokes, before and after re-sampling are shown.

Finally, all of these re-sampled points are translated by shifting the origin of the coordinate system to (x_{min}, y_{min}), where x_{min}, y_{min} are respectively the minimum of x- and y-coordinates of all the re-sampled points of the input character.

3.2. Feature extraction

In the present work, we use a new direction code histogram feature for recognition of online handwritten characters. This feature has certain similarity to the same for offline handwritten image data [8]. In the present problem, the sequence (temporal) data of online handwritten sample is divided into several sub-divisions for computation of this direction code feature.

3.2.1. Extraction of subdivisions. The whole trajectory of the pen (corresponding to non-zero pressure) forming a character sample is divided into N subdivisions. Each character sample is composed of one or more strokes and to determine the number of subdivisions of the i-th stroke, we obtain its length (L_i) by summing the distances between consecutive points forming the i-th stroke. The total length of the character sample is obtained as $L = \sum_i L_i$. Now, we obtain the number of subdivisions of each stroke as $N_i = round\left(\frac{L_i}{L} N\right)$, where i runs over the number of strokes present in the input character sample. If $\sum_i N_i \neq N$ due to round-off error, then minor adjustment is made to make $\sum_i N_i = N$.

If the number of points (re-sampled) in an individual stroke i is not a multiple of N_i, then its constituent points (save for the two terminal or critical points) are re-sampled for the second time to obtain a new set of n_i (nearest multiple of N_i) points which are approximately equidistant.

As an example, we consider the character sample in Fig. 4(*a*). It has three strokes consisting of 46, 27 and 28 sample points at the end of the preprocessing stage. The ratio of the lengths of the three strokes is 4:3:3. So, if we consider 10 subdivisions of the character sample, then the numbers of subdivisions of these three strokes are respectively 4, 3 and 3. Now, since the number of points in the first stroke is not a multiple of 4 (number of its subdivisions), its points are resampled for the second time generating a new set of 48 (nearest multiple of 4) points. Similarly, the third stroke is also resampled generating a new set of 27 points. However, the number of points in the second stroke being a multiple of 3, it is not considered for the second time resampling of its points. The final set of points after second time resampling of the character in this example and also the subdivisions of its strokes are shown in Figs. 5(*a*) and 5(*c*) respectively.

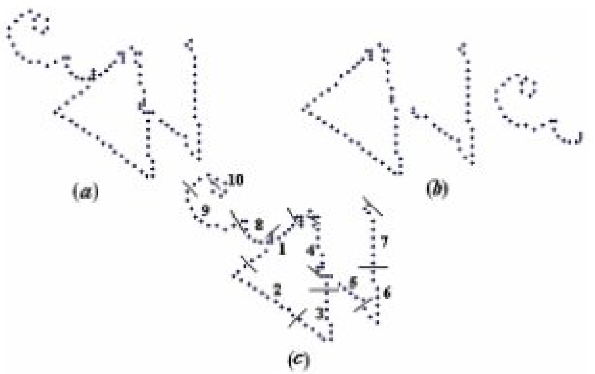


Figure 5. (*a*) Character in Fig. 4(*a*) after second time resampling, (*b*) Three strokes from left to right are according to the temporal order, (c) subdivisions and their temporal order are shown.

3.2.2. Direction code representation of strokes. Let the sequence of points in the i-th stroke be P_1, P_2, …, P_{n_i}, where n_i is the final (after re-sampling) number of points in the stroke. Now, let the angle made with the x-axis while moving from P_r to P_{r+1} be α_r, $r = 1, 2, \ldots, n_i - 1$ ($0 \le \alpha_r < 360°$). Here, the change in direction while moving from one point to the next one is important. Thus, the directions from one point to the next along a stroke can be effectively quantized into one of 8 possible values, *viz.* 1,2,…,8 according to the Freeman's direction code [11] as shown in Fig.6(a). In particular, if $337.5° \le \alpha_r < 360°$ or $0° \le \alpha_r < 22.5°$, then the corresponding direction code is 1. If $22.5 + (k-1) \times 45° \le \alpha_r < 22.5 + k \times 45°$, then the direction code is k+1, for k = 1,…,7. The initial direction code in a stroke is assumed to be 0. Each stroke of an input online handwritten pattern is thus represented in terms of the direction codes. The direction code representation of the online character sample shown in Fig.5(a) is shown in Fig.6(b).

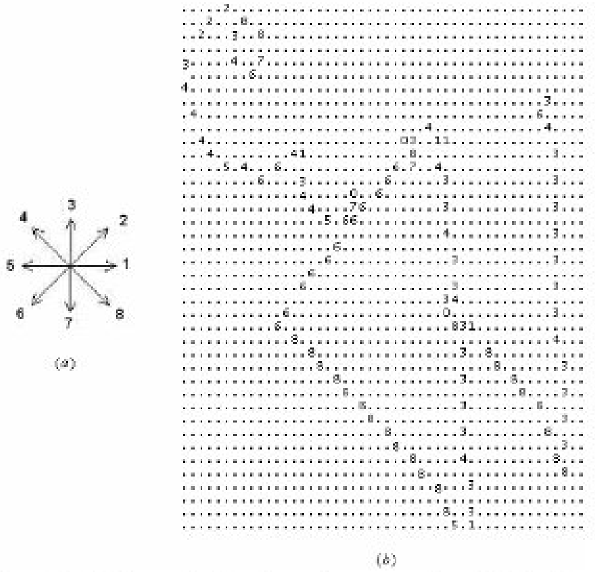


Figure 6. (a) Scheme for direction code representation; (b) Direction code representation of character sample shown in Fig. 5(a).

3.2.3. Computation of features. In each of the subdivisions, a local histogram of the direction codes is calculated. Since the directions are quantized into one of 8 possible values, viz. 1, 2, …, 8, in addition to the initial code '0', the histogram for each subdivision has 9 components. Also, for the position information, we use the coordinates of its *CG* (center of gravity) as additional features. Thus, the feature vector for each subdivision has 9+2 = 11 components. If there are N subdivisions (N = 10 in the present implementation) of the whole sequence, then the proposed feature vector consists of 11×N (110 in our implementation) components.

Feature vector components corresponding to the direction codes are normalized with respect to the total number of points in each subdivision. On the other hand, the feature components corresponding to the x- and y- coordinates of the *CG* are normalized with respect to the width and height of the character sample.

3.3. Classification

In the present recognition work, we use multilayer perceptron (MLP) for the classification task. The well-known backpropagation (BP) algorithm [12] is normally used for the training of an MLP classifier. A common problem of choosing MLP as the classifier is finding a suitable choice of its architecture (number of hidden nodes) and values of different parameters (such as learning rate and momentum factor) of the BP algorithm. There exist a number of modified BP algorithms which take care of one or more of these problems of the original BP algorithm. In the present implementation, we considered a modified BP algorithm [13] using self-adaptive learning rate values.

On the other hand, for a proper choice of hidden layer size of MLP classifier, we made several simulations using different number of hidden nodes. We obtained classification results on the test set of our database corresponding to MLP classifiers with 30, 40, 50, 60, 70 and 80 hidden nodes. Among these choices of the hidden layer, the best recognition performance on the test set has been obtained by the MLP classifier with 70 hidden nodes. This MLP classifier consists of 110 nodes (size of the input feature vector is 110) in the input layer and 50 (total number of classes) nodes in its output layer. Simulation results obtained by this MLP with 110×70×50 architecture are reported in the following section.

Another important aspect of MLP training is the proper selection of the point of termination of the iterative learning of BP algorithm. Usually, during the initial iterations of BP training, the system error [12] gradually decreases on both the training and test sets. However, after a certain number of learning iterations, this error further decreases on the training set while it starts increasing on the test set. The point of time, when the error on the test set increases during at least three consecutive iterations for the first instance, is noted and the training before the error started to increase has been considered as final.

4. Experimental Results

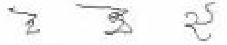
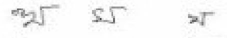
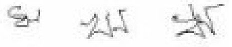
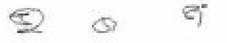
To the best of our knowledge, there does not exist any standard database of handwritten online Bangla basic characters. We have trained and tested the proposed recognition scheme using the samples of the database described in Section 2.

The proposed recognition scheme using 110 dimensional feature vector and an MLP classifier with 70 hidden nodes has provided recognition accuracies of 93.90% on the training set and 83.61% on the test set respectively (with 5000 training and 2043 test samples). Recognition percentages against each individual class are shown in Table 1. From this table it can be seen that in a few classes such as ঊ(UU), খ(KHA), জ(JA), ঠ(TTHA), ষ(SSA), the recognition accuracies on the test samples are much lower than the same on the training samples. This situation could perhaps be avoided if more samples are available for the training purpose. A few of these misclassified samples are shown in Table 2.

TABLE 1. RECOGNITION PERCENTAGES IN DIFFERENT CLASSES

Character Class	Train	Test	Character Class	Train	Test	Character Class	Train	Test
অ (A)	92.00	82.93	ছ (CHA)	100.00	85.71	ভ (BHA)	100.00	85.71
আ (AA)	99.00	89.74	জ (JA)	99.00	71.43	ম (MA)	91.00	80.95
ই (I)	81.00	75.00	ঝ (JHA)	97.00	80.95	য (YA)	96.00	80.95
ঈ (II)	97.00	95.12	ঞ (NYA)	93.00	80.95	র (RA)	88.00	95.24
উ (U)	91.00	78.04	ট (TTA)	79.00	73.68	ল (LA)	100.00	90.48
ঊ (UU)	100.00	73.68	ঠ (TTHA)	92.00	71.43	শ (SHA)	100.00	95.24
এ (E)	100.00	85.37	ড (DDA)	91.00	71.43	ষ (SSA)	96.00	71.43
ঐ (AI)	84.00	85.00	ঢ (DDHA)	99.00	85.71	স (SA)	99.00	85.71
ও (O)	99.00	78.04	ণ (NNA)	92.00	85.71	হ (HA)	99.00	95.24
ঔ (AU)	84.00	85.71	ত (TA)	96.00	90.48	ড় (RRA)	97.00	85.71
ঋ (R)	72.00	71.43	থ (THA)	97.00	80.95	ঢ় (DHRA)	96.00	90.48
ক (KA)	91.00	95.24	দ (DA)	99.00	95.24	য় (YYA)	97.00	80.95
খ (KHA)	96.00	73.68	ধ (DHA)	84.00	71.43	ং (ANUS)	99.00	95.24
গ (GA)	85.00	71.43	ন (NA)	91.00	71.34	ঃ (VISARG)	100.00	100.00
ঘ (GHA)	99.00	85.71	প (PA)	93.00	85.71	ঁ (BINDU)	97.00	85.71
ঙ (NGA)	81.00	71.43	ফ (PHA)	97.00	90.48	ৎ (KHAND)	97.00	90.48
চ (CA)	95.00	90.48	ব (BA)	99.00	90.48			

TABLE 2. A FEW EXAMPLES OF MISCLASSIFIED SAMPLES

Character Class	Misclassified Samples
ই	
খ	
ঋ	
জ	
ঠ	
ষ	

5. Conclusions

In the present article, we have described a new feature vector based on direction code histogram for recognition of online handwritten Bangla basic characters. We have simulated the proposed scheme on a relevant database developed by us. This is a 50-class problem and there are only 100 training samples per class. Considering the small size of the available training data set the achieved recognition accuracy is encouraging.

We also simulated a similar but larger feature vector by dividing the temporal data corresponding to an input character into 15 subdivisions. The size of this larger feature vector is 15×11=165. However, this did not improve the recognition accuracy.

Acknowledgment

The authors would like to acknowledge the partial financial support of the Department of Information Technology, Govt. of India towards this research work.

6. References

[1] C. C. Tappert, C. Y. Suen, and T. Wakahara, "The state of the art in on-line handwriting recognition," *IEEE PAMI.*, vol. 12, 1990, pp. 787-808.

[2] O. D. Trier, A. K. Jain and T. Taxt, "Feature Extraction Methods for Character Recognition - A Survey," *Pattern Recognition*, vol. 29, 1996, pp. 641 – 662.

[3] R. Plamondon and S. N. Srihari, "On-Line and Off-Line Handwriting Recognition: A Comprehensive Survey," *IEEE Trans. on PAMI*, 22(1), 2000, pp. 63-84.

[4] N. Arica, F. Yarman-Vural, "An Overview of Character Recognition Focused on Off-line Handwriting," *IEEE Trans. SMC, Appl. & Rev.*, vol. 31, 2001, pp. 216–233.

[5] S. K. Parui, U. Bhattacharya, B. Shaw, K. Guin, "A Hidden Markov Models for Recognition of Online Handwritten Bangla Numerals", *Proc. of the 41st National Ann. Convention of CSI*, India, 2006, pp 27-31.

[6] U. Garain, B. B. Chaudhuri, T. Pal, "Online Handwritten Indian Script Recognition: A Human Motor Function Based Framework," *Proc. of the 16th Int. Conf. on Pattern Recognition*, 2002, pp. 164-167.

[7] J. A. Fitzgerald, F. Geiselbrechtinger and T. Kechadi, "Application of Fuzzy Logic to Online Recognition of Handwritten Symbols," *9th IWFHR*, 2004, pp. 395-400.

[8] J. Cao, M. Sridhar, F. Kimura, and M. Ahmadi, "Statistical and neural classification of handwritten numerals: a comparative study", Proc. of 11th ICPR., Hague, 1992, pp. 643-646.

[9] C.-L. Liu, K. Nakashima, H. Sako, H. Fujisawa, "Handwritten digit recognition: investigation of normalization and feature extraction techniques", Pattern Recognition, vol. 37, pp. 265-279.

[10] U. Bhattacharya, M. Shridhar and S. K. Parui, On Recognition of Handwritten Bangla Characters, *Proc. of the 5th Indian Conf. on Computer Vision, Graphics and Image Processing*, Springer-Verlag, 2006, pp. 817-828.

[11] H. Freeman, Computer processing of Line-drawing Images. *ACM Computing Surveys*, vol. 6, 1974, pp. 57-97.

[12] D. E. Rumelhart, G. E. Hinton, R. J. Williams, Learning internal representations by error propagation. Institute for Cognitive Science Report 8506, San Diego: University of California, 1985.

[13] U. Bhattacharya, S. K. Parui, "Self-adaptive learning rates in backpropagation algorithm improve its function approximation performance", Proc. of the IEEE Int. Conf. on Neural Networks, 1995, pp. 2784-2788.