HMM Based Online Handwritten Bangla Character Recognition using Dirichlet Distributions

Chandan Biswas, Ujjwal Bhattacharya, Swapan Kumar Parui
CVPR Unit, Indian Statistical Institute, 203, B. T. Road, Kolkata - 108, India
chandanbiswas08@yahoo.com, {*ujjwal,swapan*}*@isical.ac.in*

Abstract

A reasonably large database of online handwritten Bangla characters has been developed. Such a handwritten character sample is composed of one or more strokes. Seventy five such stroke classes have been identified on the basis of the varying handwriting styles present in the character database. Each character sample is a sequence of strokes emanating from these stroke classes. Another database of handwritten Bangla strokes has been developed from the character database. This is the first such database for Bangla script. Certain stroke level features are defined on the basis of certain extremum points which represent the stroke shape reasonably well. The proposed character classification method is a two-stage approach. First, a probability distribution is estimated for each stroke class using the stroke features and then an HMM based character classifier is designed using each stroke class as a state. The parameters of both the stroke class distributions and the character class HMMs are estimated on the basis of the training set having 29,951 character samples. The character level recognition accuracy obtained by the proposed method on the test set having 8,616 samples, is 91.85 %.

1. Introduction

Recognition studies on online handwritten Indian scripts did not receive much attention until recently. However, situation has started changing now and the Indian Government is funding relevant research projects in several Indian scripts. Both for standard and portable miniature computing devices, non-keyboard based methods for data entry have additional importance in the Indian context since its scripts have large symbol sets. A few standard databases of online handwritten characters have already come into existence.

Bangla is the second most popular language of the Indian subcontinent used by more than 220 million people across the two neighbouring countries India and Bangladesh. This script is used to write texts in Assamese and Manipuri in addition to Bangla. Bangla script is syllabic and not alphabetic. Also, each of them consists of a smaller number of letters in their independent forms, called basic characters and an additional larger number of letters formed by combination of two or more independent basic characters.

Bangla online handwriting recognition had been studied in [1, 2, 3, 4, 5]. In [1], an HMM-based recognition system was used for Bangla online handwritten numerals. Direction code histogram feature was considered in [2] for recognition of Bangla basic characters. In [3], a Hidden Markov model based recognition scheme was presented for recognition of online handwritten Bangla basic characters. Recently, a database of online handwritten Bangla basic characters and its recognition based on sub-stroke features was described in [4]. Bangla handwritten cursive word recognition was studied in [5].

Also, Devanagari and Bangla online handwritten character recognition has been studied in [6].

The main contributions in the present paper are (i) development of a database of stroke level samples that was semi-automatically generated from a database of online handwritten Bangla basic character samples and (ii) development of a novel hidden Markov model (HMM) based handwritten Bangla character classifier where the stroke classes form the state space. Feature extraction from the stroke data is done on the basis of certain extremum points which largely retain the essential shape of the stroke. We consider Dirichlet distributions for the stroke classes. The parameters of such a distribution are estimated on the basis of the training samples of a stroke class that may come from one or more character classes.

978-0-7695-4774-9/12 $26.00 © 2012 IEEE
DOI 10.1109/ICFHR.2012.213

Figure 1. Basic characters (in printed form) are shown in bold. On the right of a basic character, the corresponding stroke classes are shown using a sample from each such stroke class.

2. Online handwritten Bangla character database

The set of basic characters of Bangla consists of 11 vowels and 39 consonants. In Fig.1, the printed symbols in bold show the ideal shapes of Bangla basic characters. The present database of Bangla online handwritten basic character samples was developed jointly by Indian Statistical Institute, Kolkata and Hewlett-Packard Labs, Bangalore, India. The database contains 38,567 samples and it is divided into a training set of 29,951 samples and a test set of 8,616 samples.

2.1 Data collection

Online handwritten Bangla character samples were collected using different devices including tablet PC, Wacom Intuous 2 tablet, and a pen paper based device Genius G-Note 7000. The pen paper based device was used frequently for data collection purpose since they make data collection more natural to the writers. The native writers were provided with forms containing boxes printed on common paper and were asked to write Bangla characters, one in each box. The ink from each of the boxes is extracted and stored in the UNIPEN format using a data collection tool developed by us.

Each character is composed one or more strokes in its handwritten form. For example, each of the two characters shown in Fig.2, is normally written as a sequence (not necessarily in the same order) of three strokes where the rightmost stroke (the horizontal line segment) may or may not be present. Note that the leftmost (as well as the rightmost) strokes for the two characters have the same shape and belong to the same stroke class. Also, in some handwriting style, the first two strokes in the first character become one single stroke (see the second character in the second row in Fig.1 and the corresponding strokes). Thus, in our database, these two characters gave rise to five different handwritten stroke classes. For the entire database, we have identified 75 such stroke classes (shown in Fig.1). In other words, any character sample in our database is composed of one or more strokes from these 75 stroke classes. It is to be noted here that a handwritten sample from a character class does not always have stroke samples from the same stroke classes. Also, the order in which the stroke samples occur may vary from one character sample to another. For example, Fig.3 shows the different ways in which the first character in Fig.2 is

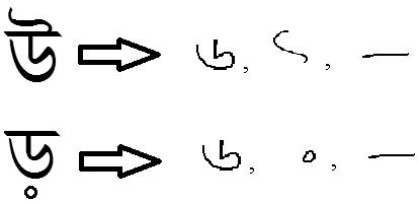

Figure 2. Two basic characters are shown on the left and respective sets of strokes found in our database are shown on the right. The first strokes of both characters are the same and belong to one stroke class. Similarly, the third strokes of both of them belong to another stroke class.

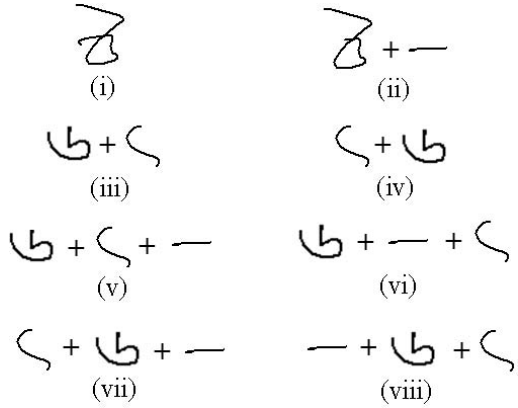

Figure 3. Samples (in our database) of the first character in Fig.2 are written in 8 different ways.

written in our database.

We have developed a database of samples of these 75 stroke classes (denoted by $\mathbf{R} = \{R_1, \cdots, R_{75}\}$) from the database of handwritten samples of 50 character classes in the following way. For each handwritten character sample, the strokes (pen down to pen up) are first identified and each of these strokes is then assigned to the corresponding stroke bin. This stroke annotation is done semi-automatically using a GUI based software developed in our laboratory.

3. Preprocessing

The input character sample is a sequence of 2-dimensional points. Preprocessing is done to normalize the position and size of the sample and to remove local noise so that the extracted features from the sample become robust. First, the repeated points are removed from the data. The whole sequence of points is then translated towards the top and the left so that the topmost point touches the top margin and the leftmost point touches the left margin. Next the points are scaled so that the vertical distance between the topmost and the bottommost points becomes 100 units. Scaling is the same in both x and y directions so that the original aspect ratio is preserved. At this stage, the sequence of points is divided into several subsequences called strokes where a stroke is the data collected from one pen down position to the next pen up position. The next preprocessing operations will be applied on individual strokes. The smoothing operation is employed on the stroke sample in the following way. The first and the last points of the stroke are kept intact. Each other point A_i is replaced by the average of the three consecutive points A_{i-1}, A_i and A_{i+1}. This smoothing is repeated twice to get rid of the local distortions that may be present in the stroke. The sequence of points in the stroke is now $\{A_i = (x_i, y_i), i = 1, 2, \cdots, M\}$. We next generate a sequence $\{P_i = (x_i, y_i), i = 1, 2, \cdots, N\}$ of a smaller number of equidistant points lying on the polyline defined by $\{A_i = (x_i, y_i), i = 1, 2, \cdots, M\}$. Here P_1=A_1, P_N=A_M ($N < M$) and the distance between P_i and P_{i+1} along the polyline is equal to 10 units for all i. Recall that the height of the whole word sample is normalized at 100 units.

4. Stroke features and their distribution

The center of gravity of a stroke $\{P_i = (x_i, y_i), i = 1, 2, \cdots, N\}$ is denoted by (X, Y) where X and Y are the arithmetic means of x_i and y_i respectively. The length L of the stroke is defined as the sum of the Euclidean distances between P_i and P_{i+1}, $i = 1, 2, \cdots, N-1$.

In order to construct the other features of the stroke, we identify three sets of extremum points of a stroke in the following way. Consider three consecutive points P_{i-1}, P_i and P_{i+1}. P_i is said to be an extremum point if one of the following eight conditions holds.

- x_i is less than or equal to both x_{i-1} and x_{i+1}.
- y_i is less than or equal to both y_{i-1} and y_{i+1}.
- x_i is greater than or equal to both x_{i-1} and x_{i+1}.
- y_i is greater than or equal to both y_{i-1} and y_{i+1}.
- $x_i + y_i$ is less than or equal to both $x_{i-1} + y_{i-1}$ and $x_{i+1} + y_{i+1}$.
- $x_i + y_i$ is greater than or equal to both $x_{i-1} + y_{i-1}$ and $x_{i+1} + y_{i+1}$.

- $x_i - y_i$ is less than or equal to both $x_{i-1} - y_{i-1}$ and $x_{i+1} - y_{i+1}$.
- $x_i - y_i$ is greater than or equal to both $x_{i-1} - y_{i-1}$ and $x_{i+1} - y_{i+1}$.

In all the above eight cases, at least one inequality should hold.

Let $\{Q_j, j = 1, 2, \cdots, n\}$ be the sequence of the extremum points as detected above in the same order as they appear in the stroke sample. In other words, $\{Q_j\}$ is a sub-sequence of $\{P_i\}$. Note that the points $\{Q_j\}$ are not necessarily equidistant. Now, the original stroke sample is represented as a polyline by joining the points Q_j and Q_{j+1} $(j = 1, 2, \cdots, n-1)$ by a line segment (call it the j-th segment). By replacing the points $\{P_i\}$ by the points $\{Q_j\}$, we reduce the size of the stroke data without losing much information about the shape and structural information of the stroke. For example, in the first row of Fig. 4, the pattern on the left is given by 78 points $\{P_i\}$ while that on the right is given by 15 points $\{Q_j\}$. It can be seen that the essential shape is preserved by $\{Q_j\}$. Also, the same is true for the pattern shown in the second row of this Figure. The pattern on the left of this second row is represented by 117 points where as the pattern on the right is represented by only 28 points.

Let, for the j-th segment, θ_j be the angle made with the x-axis by the pen movement from Q_j to Q_{j+1}. Note that θ belongs to $[0, 360)$.

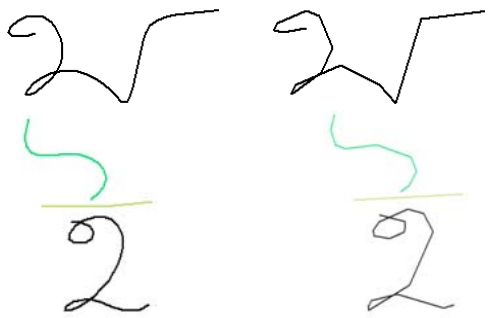

Figure 4. Two samples from two character classes are shown on the left. Each of them is represented by respective set of points P_i. These two samples when represented by the respective sets of extremum points Q_j, are shown on the right.

The whole range of $[0, 360)$ is divided into 8 disjoint intervals of width 45 each. The k-th interval is defined by $(k-1) \times 45 - 22.5 \leq \theta < (k-1) \times 45 + 22.5$, $(k = 1, 2, \ldots, 8)$. Now, for all j, the j-th segment is placed in the k-th bin B_k if θ_j belongs to the k-th interval. We now define the stroke features $\{U_k, k = 1, 2, \ldots, 8\}$ such that U_k is the sum of the lengths of the segments belonging to the k-th interval. Each U_k is normalized dividing it by $(U_1 + \cdots + U_8)$. Now, the 11-dimensional feature vector of a stroke is $(U_1, \cdots, U_8, X, Y, L)$ on the basis of which classification is made in the proposed method.

We assume here that for a stroke class, the feature vector follows a probability distribution. A natural choice for $(U_1, \cdots, U_8)$ is the Dirichlet distribution [7] and that for (X, Y, L) is a trivariate normal distribution. Thus, the probability distribution function of $(U_1, \cdots, U_8, X, Y, L)$ is expressed as

$$f(u_1, \cdots, u_8, x, y, l) = g_1(u_1, \cdots, u_8) g_2(x, y, l) \quad (1)$$

It is assumed here that the features $U_1, \ldots, U_8$ are independent of the features X, Y, L. The Dirichlet distribution $g_1(u_1, \cdots, u_8)$ and the trivariate normal distribution $g_2(x, y, l)$ are given as

$$g_1(u_1, \cdots, u_8) = \frac{\Gamma(\sum_{i=1}^{8} \alpha_i)}{\prod_{i=1}^{8} \Gamma(\alpha_i}) \prod_{i=1}^{8} u_i^{\alpha_i - 1} \quad (2)$$

where $\alpha_1, \cdots, \alpha_8$ are the parameters of Dirichlet distribution and Γ indicates the gamma function.

$$g_2(v) = \frac{exp\{-0.5(v-\mu)^T \Sigma^{-1} (v-\mu)\}}{\{(2\pi)^{3/2} |\Sigma|^{1/2}\}} \quad (3)$$

where $v = (x, y, l)$ and μ, Σ are the mean vector and covariance matrix of v.

For each stroke class, we need to obtain the distribution $f(u_1, \cdots, u_8, x, y, l)$. For this, the parameters $\alpha_1, \cdots, \alpha_8$, μ, Σ are estimated from the samples of a stroke class obtained from the training set of character samples. Estimation of μ and Σ is straightforward while $\alpha_1, \cdots, \alpha_8$ are estimated using the algorithm proposed in [9]. Let $f_i(u_1, \cdots, u_8, x, y, l)$ be the estimated probability distribution function for the stroke class R_i.

Now, for each character class, we consider its training samples and the stroke samples present in these samples. We then identify the stroke classes that these stroke samples of the character class come from. Let these stroke classes be $\mathbf{R'} = \{R'_1, \cdots, R'_m\}$. The value of m varies from one character class to another and ranges from 1 to 10 in the training set of character samples. In Fig.1, for each character, the corresponding stroke classes are shown. It can be seen only one character has 1 stroke class and only one character has 10 stroke classes (it is shown in Fig. 5 also). Now, for each character class, we find the frequency of stroke samples

coming from a stroke class. Thus m such frequencies are obtained. These frequencies are then normalized by dividing them by the total number of stroke samples and these normalized frequencies are $p_1, \cdots, p_m$ which are used as the prior probabilities of stroke classes for a character class. These character specific stroke classes $\{R'_i\}$ and the prior probabilities $\{p_i\}$ will be useful at the character level recognition.

5. HMM classifier for handwritten character recognition

5.1 Hidden Markov models

The hidden Markov model (HMM) is a doubly embedded stochastic process with an underlying stochastic process that is not observable (it is hidden), but can only be observed through another set of stochastic processes that produce the sequence of observations [8].

Depending upon the transitions there are two types of HMM : an HMM allowing for transitions from any state to any state is called a fully connected HMM. On the other hand, an HMM where the transition goes from one state to itself or to a unique follower, is called a left-right HMM or Bakis model.

The HMM may be discrete or continuous depending on whether the observation vectors emanating from a state follow a continuous probability density function or a discrete probability distribution. Quite often continuous observations are quantized as discrete signals so that a discrete HMM can be used. However, in some applications, there might be serious degradation after such quantization of continuous observations. It is advantageous to use continuous HMMs in such cases.

5.2 Proposed HMM Classifier

An HMM with the state space $S = \{s_1, \cdots, s_r\}$ and state sequence $Q = q_1, \cdots, q_T$ is defined as γ =(π, A, B) where the initial state distribution is given by π =$\{\pi_i\}$, π_i = Prob ($q_1 = s_i$), the time-homogeneous state transition probability distribution by A =$\{a_{ij}\}$ where a_{ij} = Prob ($q_{t+1} = s_j / q_t = s_i$) and the observation symbol probability distributions by *B* = $\{b_i\}$ where $b_i(O_t)$ is the distribution for state *i* and O_t is the observation at instant *t*. The HMM here is continuous and fully connected.

The problem now is how to efficiently compute $P(O/\lambda)$, the probability of an observation sequence $O = O_1, \ldots, O_T$ given a model λ =(π,A,B). For a classifier of K classes of patterns, we have K separate HMMs denoted by λ_j, $j = 1, \cdots, K$. Let an input pattern X of an unknown class have an observation sequence O. The probability $P(O/\lambda_j)$ is computed for each model λ_j and X is assigned to class c whose model shows the highest probability. That is,

$$c = \arg \max_{1 \le j \le K} P(O/\lambda_j) \tag{4}$$

For a given λ, $P(O/\lambda)$ is computed using the well known forward algorithm [8]. Note that the observation sequence $O = O_1, \cdots, O_T$ in our problem is the sequence of feature vectors extracted from the strokes (arranged in the order in which these strokes are written) that are present in a handwritten character sample. T is the number of strokes in the sample. The states here are the stroke classes, that is, $S = \{R'_1, \cdots, R'_m\}$ and $b_i(O_t)$, the distribution for state *i*, is in fact $f_i(u_1, \cdots, u_8, x, y, l)$ defined at the end of Section 4. Here the value of K is 50 and $r = m$.

5.2.1 Estimation of HMM Parameters

The state space and the state conditional distributions are now determined. The parameters to be estimated next are the initial state distribution $\{\pi_i\}$ and the state transition probability distribution $\{a_{ij}\}$.

Let, for a training character sample, the stroke feature vectors extracted from the stroke sequence present in the sample are O_1, $O_2, \cdots, O_T$. For each O_t, compute (p_i are defined at the end of the last section)

$$h_i(O_t) = p_i b_i(O_t) / \{\sum_{j=1}^{m} p_j b_j(O_t)\} \tag{5}$$

and O_t is assigned to state k where

$$k = \arg \max_{1 \le i \le m} h_i(O_t). \tag{6}$$

This assignment to respective states is done for all L observation sequences (L is the number of training samples). Let the state sequence thus generated from an observation sequence O_1, $O_2, \ldots, O_T$ be $q_1, \ldots, q_T$. From these L state sequences, the estimates of the initial state distribution probabilities are computed as $(1 \le i \le m)$

$$\pi_i = \frac{\textit{number of occurrences of } \{q_1 = s_i\}}{L} \tag{7}$$

The state transition probability estimates $a_{i,j}$ are computed as $(1 \le i, j \le m,\ 1 \le t \le T-1)$

$$a_{i,j} = \frac{\textit{number of occurrences of } \{q_t = s_i \ \& \ q_{t+1} = s_j\}}{\textit{total number of occurrences of } \{q_t = s_i\}} \tag{8}$$

The above HMM parameter estimates are fine-tuned using re-estimation by Baum-Welch forward-backward algorithm.

6. Results and discussions

In the present study, there are 50 Bangla character classes. The total number of training samples is 29,951 where the classes have varying number of samples. The maximum number of training samples in a class is 1506 and the minimum number of training samples in a class is 401. The total number of test samples is 8,616 where the maximum and minimum numbers of test samples in a class are 347 and 135 respectively.

The number of stroke classes associated with a character class varies from one character to another. Fig. 5 gives the number of character classes having a particular number of stroke classes. For example, there are 3 character classes having 7 stroke classes. Details are shown in Fig. 1.

In our recognition approach, we do not explicitly classify a stroke sample into a stroke class, but use a stroke class as a state in the HMM based character classifier. On the basis of the training stroke samples of each stroke class, we estimate the probability density functions based on Eq.1.

Then for each of the character classes, we first identify the stroke classes. For each training sample in the character class, we generate a state sequence based on Eq.6. On the basis of these state sequences, the initial state distribution and the transition probabilities are estimated using Eq.7 and Eq.8. Thus, all the HMMs are now ready. For the final character classification, we use Eq.4 for an unknown character sample. The recognition accuracy based on the test set is found to be 91.85%. This improves the recognition accuracy of 87.7% reported earlier in [3].

No of Stroke Classes	1	2	3	4	5	6	7	8	9	10
No of Character Classes	1	13	7	13	4	6	3	1	1	1

Figure 5. Number of character classes having the same number of stroke classes.

7. Conclusions

In the present paper, we have reported a database of Bangla online handwritten strokes. This is the first database of its kind and has been developed from a database of Bangla online handwritten character samples through a semi-automatic annotation tool. The proposed online handwritten character recognition is based on an HMM where the states are the stroke classes.

Acknowledgement: This work has been partially supported by the Dept. of Information Technology, Govt. of India.

References

[1] S. K. Parui, U. Bhattacharya, B. Shaw, K. Guin, A hidden Markov model for recognition of online handwritten Bangla numerals, Proc. of the 41st National Ann. Conv. of CSI, pp. 27-31, 2006.

[2] U. Bhattacharya, B. K. Gupta and S. K. Parui, Direction code based features for recognition of online handwritten characters of Bangla, Proc. of the 9th ICDAR, vol. 1, pp. 58-62, 2007.

[3] S. K. Parui, K. Guin, U. Bhattacharya, and B. B. Chaudhuri, Online handwritten Bangla character recognition using HMM, Proc. of 19th Int. Conf. on Pattern Recognition, 2008.

[4] T. Mondal, U. Bhattacharya, S. K. Parui, K. Das and V. Roy, Database generation and recognition of online handwritten Bangla characters, Proc. of the Int. Workshop on Multilingual OCR (MOCR), Article No. 9, ACM Int. Conf. Proceeding Series, Barcelona, Spain, 2009.

[5] U. Bhattacharya, A. Nigam, Y. S. Rawat and S. K. Parui, An analytic scheme for online handwritten Bangla cursive word recognition. Proc. of the 11th ICFHR, pp. 320-325, 2008.

[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, pp. 164-167, 2002.

[7] S. Kotz, N. Balakrishnan and N.L. Johnson, Continuous multivariate distributions, volume 1: Models and applications, second edition, John Wiley and Sons, New York, 2000.

[8] L. Rabiner, A tutorial on hidden Markov models and selected applications in speech recognition. Proceedings of the IEEE, vol. 77(2), pp.257-286, 1989.

[9] N. Wicker, J. Muller, R. K. R. Kalathur and O. Poch, A maximum likelihood approximation method for Dirichlet's parameter estimation, Computational Statistics and Data Analysis, vol. 52(3), pp.1315-1322, 2008.