

A Novel Approach of Bangla Handwritten Text Recognition using HMM

^aPartha Pratim Roy*, ^bPrasenjit Dey, ^cSangheeta Roy, ^aUmapada Pal, ^dFumitaka Kimura

^aCVPR Unit, Indian Statistical Institute, Kolkata, India

^bDept. of CSE, Institute of Engineering & Management, Kolkata

^cTata Consultancy Services, Kolkata, India

^dGraduate School of Engg., Mie University, Mie, Japan

*email: 2partharoy@gmail.com

Abstract—This paper presents a novel approach for offline Bangla (Bengali) handwritten word recognition by Hidden Markov Model (HMM). Due to the presence of complex features such as headline, vowels, modifiers, etc., character segmentation in Bangla script is not easy. Also, the position of vowels and compound characters make the segmentation task of words into characters very complex. To take care of these problems we propose a novel method considering a zone-wise break up of words and next perform HMM based recognition. In particular, the word image is segmented into 3 zones, upper, middle and lower, respectively. The components in middle zone are modeled using HMM. By this zone segmentation approach we reduce the number of distinct component classes compared to total number of classes in Bangla character set. Once the middle zone portion is recognized, HMM based forced alignment is applied in this zone to mark the boundaries of individual components. The segmentation paths are extended later to other zones. Next, the residue components, if any, in upper and lower zones in their respective boundary are combined to achieve the final word level recognition. We have performed a preliminary experiment on a dataset of 10,120 Bangla handwritten words and found that the proposed approach outperforms the custom way of HMM based recognition.

Keywords—*Handwriting Recognition, Bangla Character Recognition, Indian Script Recognition, HMM.*

I. INTRODUCTION

Recognition of handwritten words has long been an active research area because of its vast potential applications. Some of its potential application areas are postal automation, bank cheque processing, automatic data entry, etc. There are many research works towards handwritten text recognition in Roman [1], Japanese/Chinese [2] and Arabic scripts [3]. Although many investigations have been made towards the recognition of isolated handwritten characters and digits of Indian scripts [15], only a few pieces of work [14, 16] exist towards offline handwritten text recognition in Indian scripts. Bangla is the second most popular language in India and fifth most popular language in the world. About 200 million people of Eastern India and Bangladesh use Bangla script for communication. So, there is a need of development for OCR systems capable

of handling handwritten Bangla script. The OCR involving printed Bangla script has already been addressed in [6, 10, 11]. But problems related to OCR of handwritten Bangla script still constitute an unexplored area of research. An example of Bangla handwritten document image is shown in Fig. 1. Many works have been done for character level segmentation [14, 16] in Bangla script. It is reported that due to the presence of noise, touching characters, etc., the segmentation of characters from a string may fail often. Sometimes characters may generate disjoint character components through preliminary segmentation process. Proper classification and reunification of these components using recognition based segmentation are other challenging tasks for segmentation of handwritten Bangla script. Overlapping and touching characters, which frequently occur in Bangla writing style, create more hindrance in recognizing characters of the words.

In the past decades stochastic approaches such as Hidden Markov Models (HMMs) have been widely applied to perform text recognition task [5]. HMMs are effective for modelling unconstrained text-string. This is mostly due to their ability to cope with non-linear distortions and incomplete information. Mainly two approaches named segmentation based [7] and holistic based [8] approaches are used for the word recognition purpose. A combination of these two approaches has also been used in [9]. In practice, a HMM can be employed to represent a whole word or, alternatively, sub-word units such as characters which can be concatenated to form general strings. Only a few pieces of work [17, 18] are done in Bangla handwritten word recognition using HMM. Almost all these methods consider the recognition as word wise HMM model creation. In these approaches feature extraction was performed from the entire word and recognition was performed with the help of lexicon. The main drawback of these word-based HMM models is that the recognition process is limited to a set of words only. Also, for each word a large number of training data is needed. An unknown word which was not trained by the models will not be recognized. To avoid this task, HMMs are trained on sub-word units such as characters which can be concatenated to form general strings. Character based HMM models [5] have been successfully used for recognition of arbitrary set of words in English/Latin scripts. An advantage of latter systems is that they allow recognizing unknown

Though, character based HMM models are popular in the literature of text recognition, the process may not be directly useful in Indian scripts, especially in Bangla. It is due to the fact that in Bangla, combination of vowels, modifiers and characters lead to a huge number of character classes. Thus, sufficient data for each combination will be necessary for training the respective class models. To overcome this problem, here we propose an efficient approach to reduce the number of classes in HMM for Bangla word recognition. Effectively, we gain the recognition performance using these lesser number of characters.

Bangla vowels	আ	ই	ঈ	উ	ঊ	ঋ	এ	ঐ	ও	ঔ
Modified shape	।	ি	ী	ু	ূ	ৄ	ে	ৈ	ো	ৌ
When attached to consonant ক	কা	কি	কী	কু	কূ	কৄ	কে	কৈ	কো	কৌ

UPPER ZONE

MIDDLE ZONE

LOWER ZONE

HEAD LINE (MATRA)

BASE LINE

Fig.4: Different zones of a Bangla text line.

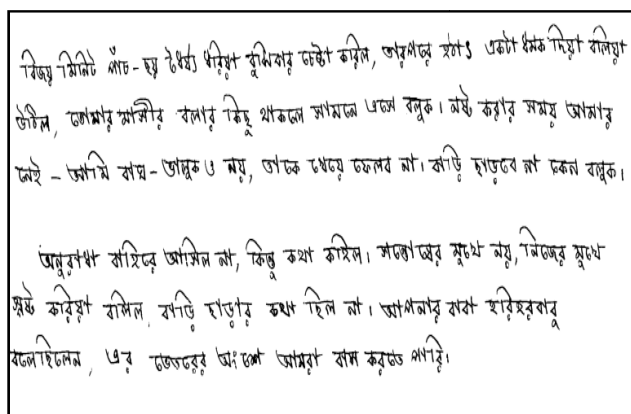


Fig.1: A part of document containing Bangla handwritten text.

The alphabet of the modern Bangla script consists of 11 vowels and 39 consonants. The alphabets are shown in Fig. 2. Writing style in this script follows from left to right. In Bangla most of the characters have a horizontal line or ‘*Matra*’ at the upper part (See Fig. 2). When two or more characters sit side by side to form a word, *Matras* of them generally touch. In this script a vowel following a consonant takes a modified shape. Depending on the vowel, its modified shape is placed at the left, right (or both), or bottom of the consonant. These modified shapes are called modified characters. Examples of modified character are shown in Fig. 3. A consonant or vowel following a consonant sometimes takes a compound orthographic shape called as compound character. There are about 280 compound characters in Bangla script [6]. A Bangla text line can be partitioned into three zones. The upper-zone (Z_U) denotes the portion above the *Matra*, the middle zone (Z_M) covers the portion between *Matra* and base-line, and the lower-zone (Z_L) is the portion below base-line. Different zones in a Bangla text line are shown in Fig. 4.

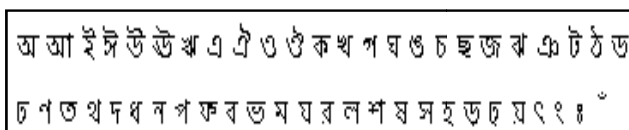


Fig.2. Basic characters of Bangla alphabet are shown. First eleven are vowels and rests are consonants.

Our proposed recognition approach involves two steps: middle zone recognition and combination with modified characters in other two zones. Fig. 5 presents a flow chart of our approach. The structure of the paper is organized as follows. Section II gives a brief review of preprocessing and zone segmentation approach. Section III describes the proposed approach of word recognition using zone segmentation. An explanation of database and experimental result are given in Section IV. Finally, conclusion is inferred in Section V.

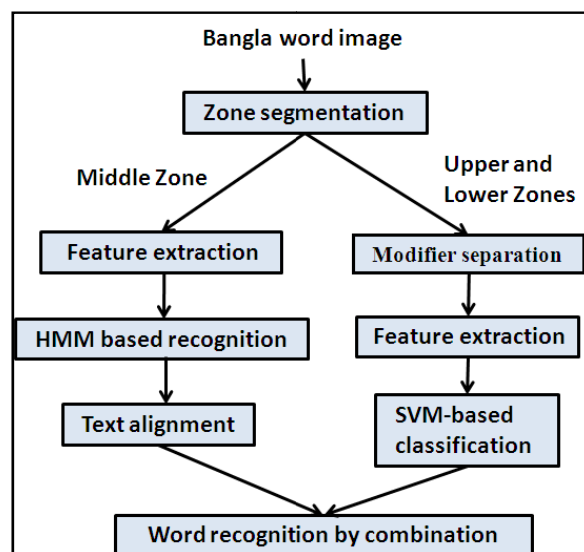


Fig.5: Block diagram of the proposed framework

II. PREPROCESSING AND ZONE SEPARATION

Preprocessing and Word Separation: The document image is converted into binary image using global histogram-based Otsu binarization method. The binary document is segmented into individual text lines using a line segmentation algorithm [4]. Here, some seed components of a line are obtained from smoothed text regions of document. The upper and lower boundary information of a text line is obtained from background regions. Next, foreground seed components and boundary information are used to segment the text-lines. Once

lines are separated, Run Length Smoothing Algorithm (RLSA) [4] is next applied on each text line to get individual words as a component. A connected component labeling is applied to find the bounding box of the word patches in the line. Next, using the patch mask, the original word is considered from the binary image. After that, the skew correction is performed using Hough transform. The slant estimation and correction are done based on the chain code information of the contour of the word image.

Zone Segmentation: Once we collect all the Bangla words from a document image, we segment the words into 3 zones. To segment the individual characters from the segmented word, we first detect the *Matra* in that word. In literature, *Matra* is determined by projection analysis in horizontal direction and considering the row with highest peak. But, due to the free flow nature of handwriting the *Matra* is rarely a perfect straight line. It is often curvy and broken.

To solve this problem, we propose an efficient method to detect the *Matra*. Let H be the height of Z_M of the word image; H is calculated by taking mode of height list taken from top and bottom most pixels of each column of the word image. Next, we consider the row with highest peak as a first estimated location (L_{M1}) of *Matra*. To confirm this location, we find the busy zone of the upper half (U_M) that includes L_{M1} . Since in Bangla script, the upper zone contains fewer components than that in the middle zone, the portions below the *Matra* will be more cluttered than that of above the *Matra*. Hence, there will be a sharp decline in projection peak in U_M while moving from lower the *Matra* to above the *Matra*. We mark the row where a sharp decline in projection is observed as the second estimated location (L_{M2}) of *Matra*. If both L_{M1} and L_{M2} are very close, less than $H/10$, we confirm the location of *Matra* row as L_{M1} otherwise L_{M2} is considered. Estimated *Matra* is denoted in red line in Fig.6.

Now after estimating the *Matra* row we create a window of *Matra* region (W_M) of height $H/5$ pixels keeping the *Matra* in middle. It is checked that the curvilinear *Matra* resides in W_M in 98% of the words from dataset. Next, we extract the skeleton of the word image and find the high curvature points, junction points, and end points of the skeleton image. These points are marked as ‘P’. Now, we find the lines between consecutive ‘P’s in horizontal direction within W_M . If any line emerging from ‘P’ crosses the W_M , we consider it as a portion of a letter and discard the line. Only those lines between ‘P’s which are passed within W_M are considered. If we find more than one pixels in a single column we consider the upper most pixel. In some words *Matra* may be broken and discontinuous. There, we join the two nearest *Matra* pixels using standard Bresenham algorithm. Next we mark the modifiers in the upper zone by checking the upper portions of *Matra*.

To separate the modifiers in the lower zone, we consider the projection profiles of the word for analysis. We observe that there will be sharp decline in projection peak when it moves from middle zone to lower zone. It is because pixels in middle zone are more cluttered and closed, whereas, pixels in lower zone are more sparse. We mark the row where a sharp decline in histogram is observed. This baseline separates middle zone from lower zone. Some examples of zone segmentation are shown in Table I. It is to be noted that often components in middle zone are touching whereas the components in upper and lower zones are segmented.

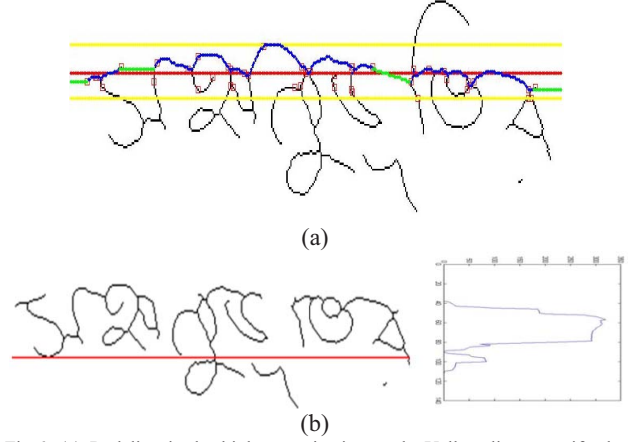


Fig.6: (a) Red line is the highest projection peak. Yellow lines specify the frame. Red squares represent high curvature points, corners and junction points within this frame. Blue line denotes the detected *Matra*. Green line joins the nearest *Matra* pixels in case of broken *Matra*. (b) The figure shows the projection peaks in horizontal direction from middle and lower zones only. The valleys in peaks denote the transition of these zones.

Table I: Few examples of zone segmentation in Bangla word images

Original Image	Upper Zone	Middle Zone	Lower Zone
স্বপ্ন	স	ব	প্ন
আশ্রিত	আ	শ্রি	ত
অশ্রুজল	অ	শ্রু	জল
ভুক্তি	ভ	ক্ট	ি
অবোধ	অ	বো	ধ

III. TEXT RECOGNITION

A) Middle Zone Component Recognition

We apply HMM based stochastic sequential classifier for recognizing the components in middle zone. The feature extraction and HMM models for middle zone are described as follows.

Feature Extraction: Local gradient histogram (LGH) [19] has been used for feature extraction in our approach. Here, a sliding window traverses the image from left to right in order to produce a sequence of overlapping sub-images. Each window is sub-divided into 4×4 (4 rows and 4 columns) regular cells and from all pixels in each cell a histogram of gradient orientations is calculated.

The gradient vector is divided into an L bin histogram. Each bin specifies a particular octant in the angular radian space. Here we consider 8 bins $360^\circ/45^\circ$ of angular information. The histogram is formed by adding up $m(x, y)$ to the bin indicated by quantized $\Omega(x, y)$. The concatenation of the 16 histograms of 8 bins provides a 128-dimensional feature vector for each sliding window position.

Hidden Markov Model: The feature vector sequence is processed using left-to-right continuous density HMMs [19]. One of the important features of HMM is the capability to

model sequential dependencies. An HMM can be defined by initial state probabilities π , state transition matrix $A = [a_{ij}]$, $i, j=1,2,\dots,N$, where a_{ij} denotes the transition probability from state i to state j and output probability $b_j(O_k)$ modeled with continuous output probability density function. The density function is written as $b_j(x)$, where x represents k dimensional feature vector. A separate Gaussian mixture model (GMM) is defined for each state of model. Formally, the output probability density of state j is defined as

$$b_j(x) = \sum_{k=1}^{M_j} c_{jk} \mathcal{N}(x, \mu_{jk}, \Sigma_{jk}) \quad (1)$$

where, M_j is the number of Gaussians assigned to j , and $\mathcal{N}(x, \mu, \Sigma)$ denotes a Gaussian with mean μ and covariance matrix Σ and c_{jk} is the weight coefficient of the Gaussian component k of state j . For a model λ , if O is an observation sequence $O = (O_1, O_2, \dots, O_T)$ which is assumed to have been generated by a state sequence $Q = (Q_1, Q_2, \dots, Q_T)$, of length T , we calculate the observation's probability or likelihood as follows:

$$P(O, Q | \lambda) = \sum_q \pi_{q1} b_{q1}(O_1) \prod_{t=2}^T a_{q_{t-1} q_t} b_{q_t}(O_t) \quad (2)$$

where π_{q1} is initial probability of state 1.

In the training phase, the transcriptions of the middle zone of the word images together with the feature vector sequences are used in order to train the character models. The recognition is performed using the Viterbi algorithm. For the HMM implementation, we used the HTK toolkit [20].

HMM based Text Alignment: For estimating the boundaries of the characters in a Bangla word, Viterbi forced alignment (FA) has been used. It is the process of finding the optimal alignment of a set of Hidden Markov Models. The segmentation path is refined through iterative alignment and retraining, called embedded training. Using the alignment algorithm we obtain the character segments (S_1, S_2, \dots, S_n) of a given word hypothesis (see Fig.7). We generate N -best Viterbi list composed of N hypotheses. N -best lists are generated to obtain a set of likely word hypotheses. Characters are segmented from the word hypotheses. These associate different labelling and segmentation part. Every pair of segment and label in the list is given a confidence measure called HMM log-likelihood. Among all, the best word hypothesis can be chosen based on addition of recognition result of the modifier levels in upper and lower zones. The process is discussed below.



Fig.7: Character alignment in middle-zone of two Bangla words using the Viterbi algorithm.

B) Recognition of Modifiers in Upper and Lower Zone

The isolated components which were included in upper and lower zones are segmented using connected component (CC) analysis and next they are recognized and labeled as text character using 400 dimensional Gradient feature and Support Vector Machine (SVM). Upper zone modifiers e.g. $\smile, \frown, \cup, \cap, \cup, \cap$ and Lower zone modifiers e.g. $\alpha, \beta, \gamma, \delta, \epsilon, \zeta, \eta, \theta$

are separately considered for classification in this proposed approach. The feature extraction and classification process for upper and lower zones are mentioned below.

Feature Extraction: 400 dimensional gradient based feature [22] is used in our system for classification. The component image is normalized into 126x126 size and converted to gray-scale image by applying a set of mean-filtering. Next the resultant gray image is segmented into 9X9 blocks. Roberts filter is applied next to obtain gradient image. The direction of gradient is quantized into 16 directions and the gradient strengths are accumulated in each quantized direction. Histograms of 16 quantized directions are computed in each of 9x9 blocks. Finally, 9x9 blocks are down sampled into 5x5 by a Gaussian filter. Thus, we get $5 \times 5 \times 16 = 400$ dimensional feature.

Classification: The gradient features of the modifiers in upper and lower zones are fed into a Support Vector Machine (SVM) for classification. SVM is a popular classification technique and has been successfully applied in a wide range of applications. Given a training database of M data: $\{x_m | m=1, \dots, M\}$, the linear SVM classifier is defined as: $f(x) = \sum_j \alpha_j x_j + b$ where, x_j is the set of support vectors and the

parameters α_j and b have been determined by solving a quadratic problem. The linear SVM can be extended to a non-linear classifier by replacing a kernel function. Details of SVM can be found in [21]. We feed 400 dimensional gradient feature into a SVM classifier (Gaussian kernel with Radial Basis Function) for classification of modifier components.

For each component we compute the recognition confidence using our SVM classification process and rank the class labels using confidence scores in descending order.

C) Word Recognition by Combining Zone-wise Results

In this step we combine the HMM-based recognition result of middle zone with SVM-based recognition result of modifiers of upper and lower zones. As mentioned earlier, we apply a character alignment strategy in HMM based middle zone recognition result to have the boundary estimation of each characters. These boundary information are next mapped into upper and lower zones for combination of modifiers. We show an example in Fig. 8 how the boundary estimation is extended in other zones.

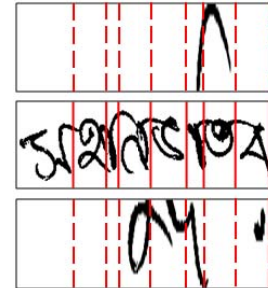


Fig.8: Mapping of alignment of middle zone in upper and lower zones for component combination.

The modifiers in upper and lower zones are next combined with the specific character in middle zone according to positional information. It may happen that some modifiers are wrongly positioned or they are written bigger so that the middle zone character can not be decided. To avoid this problem we check neighbour characters in middle zone for combination and take the best hypothesis from the lexicon list.

IV. EXPERIMENTAL RESULTS

To the best of our knowledge, there exists no standard database to evaluate the performance of Bangla handwritten word recognition. For experiment of our handwritten word recognition scheme, we collected a total of 10,120 handwritten word samples. These words were considered from 60 handwritten document images from individual of different professions. Also, we combined a subset of city-name dataset [14] in this work. These city name samples are collected from handwritten address block of Indian postal documents as well as from individuals using some specially designed forms. The database of word images is divided into two subsets: 7,725 images for training, and 2,395 for testing. We have used the dictionary of size 2,520 words.

Middle-Zone Recognition Results: We considered continuous density HMMs with diagonal covariance matrices of GMMs in each state. During HMM training, 128 Gaussian mixtures and 7-state left-to-right HMM are used. A set of validation data containing 2200 words is used to learn these parameters. Some results of HMM-based recognition in middle zone components are shown in Table II. We show top 3 choices of the hypotheses here. It is to be noted that the correct result appear among these top choices. The hypothesise results are next combined with modifiers to get the final result. In Table III, we show the quantitative results with top 5 choices. The ground truth of these words having only middle zone components were obtained for this purpose.

Modifier Recognition Results: Table IV shows the results of modifier recognition experiment. Row 1 shows the performance of upper zone modifiers and Row 2 shows the performance of lower zone modifiers.

Table II: Top 3 choices of recognition results of middle zone components

Image	ফুটপাত	ঘুরিয়া	আইনি
Choice 1	ফটপাত	ঘাবয়া	আয়ন
Choice 2	ফটপাথ	বাকা	আতথ
Choice 3	ফটবল	যাবনা	আহান

Table III: Middle zone recognition accuracy with HMM.

Choices	Top1	Top2	Top3	Top4	Top5
Accuracy	80.71%	89.63%	92.47%	93.86%	94.32%

Table IV: Recognition performance of modifiers in upper and lower zones

	Training	Testing	Accuracy
Upper Zone	2380	1736	83.72%
Lower Zone	1429	1273	89.08%

Word Recognition Results: Few examples of Bangla word recognition results are shown in Table V. We have obtained the best accuracy as 84.22% by combining the zone wise recognition results. Table VI shows the performance of combination results. Here, Top N denotes that the true word is present among the N-best word hypotheses. With Top 5 choices the accuracy reaches upto 94.83%.

Table V: Qualitative word recognition result combining middle zone with other zone modifiers

Image	Final Result
ফুটপাত	ফুটপাত
ঘুরিয়া	ঘুরিয়া
আইনি	আইনি
রাজনীতির	রাজনীতির
সহানুভূতির	সহানুভূতির

Table VI: Combined word recognition results of 3 zones with proposed approach.

Choices	Top1	Top2	Top3	Top4	Top5
Accuracy	84.22%	87.93%	90.37%	92.90%	94.83%

For comparison, we have developed HMM-based word recognition without segmenting the zones. We have kept the same parameter set-up for learning the HMM models. From the test set we have achieved 67.18%. The lower accuracy is mainly due to insufficient data in learning the character models.

Error Analysis: While analyzing the errors in the experiment, we have found that detection of the baseline is not correct always. The baseline which separates the middle zone from the lower zone is difficult to detect due to variant writing styles. If the letters of the word are irregular in size, or if too many modifiers are used in lower zones, then we may not find any sharp change in projection peak between the middle zone and lower zone of the image. We show an example in Fig. 9 where the zone segmentation was wrong. It was due to improper detection of projection valley analysis of the word image. Thus the modifiers in lowers zone were not segmented properly.



Fig 9: Example of erroneous baseline detection to segment the lower zone from middle zone.

V. CONCLUSIONS

In this paper, we have proposed a novel approach of Bangla handwritten text recognition. A complete architecture is provided to use HMM-based segmentation free technique for efficient recognition. As the proposed approach segments the zones of words, the training data entailed for character modelling is less. Combining LGH feature with HMM followed by segmentation helps to recognize the words in zone-wise. Since the modifiers in upper and lower zones are distinct; SVM classifier is used for the identification of these modifiers. Finally, zone-wise results are combined together to obtain the final word. Experiments are conducted to evaluate the efficiency and effectiveness of the proposed method. Our experiments have shown that zone based segmentation does have strong recognition capability. Zone-wise recognition, very important in Bangla, could offer new insights into several similar type languages such as Devanagari, Gurumukhi, etc.

REFERENCES

- [1] A. L. Koerich, R. Sabourin, and C. Y. Suen, "Recognition and verification of unconstrained handwritten words". IEEE Trans. Pattern Analysis and Machine Intelligence, 27, pp. 1509–1522, 2005.
- [2] C. L. Liu and M. Koga and H. Fujisawa, "Lexicon driven segmentation and recognition of handwritten character strings for Japanese address reading", IEEE Trans. Pattern Analysis and Machine Intelligence, vol.24, pp.1425-1437, 2002.
- [3] V. Märgner and Haikal El Abed, "Arabic handwriting recognition competition", Proc. 9th International Conf. on Document Analysis and Recognition, pp. 1274-1278, 2007.
- [4] P. P. Roy, U. Pal and J. Lladós, "Morphology Based Handwritten Line Segmentation Using Foreground and Background Information", Proc. International Conf. on Frontier of Handwriting Recognition, pp. 241-246, 2008.
- [5] A. B. Bernard, F. Menasri, R. H. Mohamad, C. Mokbel, C. Kermorvant, and L. Sulem. Dynamic and contextual information in HMM modeling for handwritten word recognition. IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 33, pp. 2066-2080, 2011.
- [6] U. Pal, B. B. Chaudhuri, "OCR in Bangla: an Indo-Bangladeshi Language", Proceedings of the 12th IAPR International Conference on Pattern Recognition B:ComputerVision & Image Processing, 1994.
- [7] R. Plamondon and S. N. Srihari, "On-line and off-line handwritten recognition: A comprehensive survey", IEEE Trans. Pattern Analysis and Machine Intelligence, vol.22, pp.62-84, 2000.
- [8] S. Madhvanath and V. Govindaraju, "The role of holistic paradigms in handwritten word recognition", IEEE Trans. Pattern Analysis and Machine Intelligence, vol.23, pp.149-164, 2001.
- [9] M. Y. Chen, A. Kundu and J. Zhou, "Off-line handwritten word recognition using a hidden markov model type stochastic network", IEEE Trans. Pattern Analysis and Machine Intelligence, vol.16, pp.481-496, 1994.
- [10] U. Pal and B. B. Chaudhuri, "Indian script character recognition: A survey", Pattern Recognition, vol.37, pp.1887-1899, 2004.
- [11] A. Hasnat, S. M. Habib and M. Khan."A high performance domain specific OCR for Bangla script", Int. Joint Conf. on Computer, Information, and Systems Sciences, and Engineering (CISSE), 2007.
- [12] Open_Source_Bangla_OCR:http://sourceforge.net/project/showfiles.php?group_id=158301&package_id=215908.
- [13] F. Kimura, S. Tsuruoka, Y. Miyke and M. Shridhar, "A Lexicon Directed Algorithm for Recognition of Unconstrained Handwritten Words", IEICE Trans. Inf. and System, Vol.E7-D, pp.785-793, 1994.
- [14] U. Pal, R. Roy, F. Kimura: Multi-lingual City Name Recognition for Indian Postal Automation. Proc. International Conf. on Frontier of Handwriting Recognition, pp. 169-173, 2012.
- [15] U. Bhattacharya, B. B. Chaudhuri: Handwritten Numeral Databases of Indian Scripts and Multistage Recognition of Mixed Numerals. IEEE Trans. Pattern Anal. Mach. Intell. 31(3): pp. 444-457, 2009.
- [16] U. Pal, R. Roy, F. Kimura: Handwritten Street Name Recognition for Indian Postal Automation. International Conf. on Document Analysis and Recognition, pp. 483-487, 2011.
- [17] T. Bhowmik, U. Roy, S. K. Parui: Lexicon Reduction Technique for Bangla Handwritten Word Recognition. In Proc. Document Analysis Systems, pp. 195-199, 2012
- [18] T. Bhowmik, S. K. Parui, U. Roy: Discriminative HMM training with GA for handwritten word recognition. In Proc. ICPR, pp. 1-4, 2008.
- [19] J. R. Serrano and F. Perronnin, "Handwritten word-spotting using hidden Markov models and universal vocabularies", Pattern Recognition, vol. 42(9), pp. 2106-2116, 2009.
- [20] S. Young. The HTK Book, Version 3.4. Cambridge Univ. Eng. Dept., 2006.
- [21] V. Vapnik, "The Nature of Statistical Learning Theory", Springer Verlag, 1995.
- [22] U. Pal, N. Sharma, T. Wakabayashi and F. Kimura, "Handwritten Numeral Recognition of Six Popular Indian Scripts", In Proc. 9th International Conf. on Document Analysis and Recognition, pp. 749-753, 2007.