A Comparative Study of Features for Handwritten Bangla Text Recognition

Ayan Kumar Bhunia
Dept. of ECE,
Inst. of Engg. & Management
Kolkata, India
ayanbhunia007@gmail.com

Ayan Das
Dept. of ECE,
Inst. of Engg. & Management
Kolkata, India
das.ayan.iem@gmail.com

Partha Pratim Roy
Dept. of CSE
Indian Institute of Technology, Roorkee, India
proy.fcs@iitr.ac.in

Umapada Pal
CVPR Unit
Indian Statistical Institute
Kolkata, India
umapada@isical.ac.in

***Abstract*— Recognition of Bangla handwritten text is difficult due to its complex nature of having modifiers and headlines features. This paper presents a comparative study of different features namely LGH (Local Gradient of Histogram), PHOG (Pyramid Histogram of Oriented Gradient), GABOR, G-PHOG (Combined GABOR and PHOG) and profile feature by Marti-Bunke when applied in middle zone recognition of Bangla words using Hidden Markov Model (HMM) based framework. For this purpose, a zone segmentation method is applied to extract the busy (middle) zones of handwritten words and features are extracted from the middle zone. The system has been tested on a sufficiently large and variation-rich dataset consisting of 11,253 training and 3,856 testing data. From the experiment, it has been noted that PHOG feature outperforms other features in middle zone recognition. Since PHOG feature outperform others, we use this feature for full word recognition, For this purpose initially upper and lower zone components are recognized by PHOG features and SVM classifier. Finally, the zone-wise results are combined by the context information of the corresponding components in each zone to obtain the word level recognition.**

Keywords—Handwritten Text Recognition, Hidden Markov Model, Bangla Script Recognition

I. Introduction

Though, the automatic recognition of printed text has achieved a great success rate, the recognition rate of handwritten text is not high. The main hindrance behind the difficulties of making a handwritten OCR is its huge variation in writing style and complex shapes of text characters. There are many research works towards handwritten text recognition in Roman [1], Japanese/Chinese [2] and Arabic scripts [3]. Recognition of Indian scripts like Bangla, Hindi, etc. [6] are more difficult due to their complex syntax and spatial variation of the characters when combined with other characters. These scripts contain a headline (Matra) which often increases the complexity because of its touching nature with the characters. Hence, developing an efficient Bangla OCR system will help the recognition systems of similar kind of Indian scripts (Hindi, Punjabi, etc.).

A. Bangla handwritten text recognition

Bangla, the sixth most widely used writing system in the world [19] and it, is spoken by 200 million people in Eastern India and Bangladesh. Bangla alphabets contain a total of 50 characters including 11 vowels and 39 consonant [6]. The vowels are modified when connected to any consonant and they form "Modifiers". These modified shapes are placed at the left, right (or both), top or bottom of the consonant. Most of the characters have a horizontal line or '*Matra*' at the upper part (See Fig. 1). When two or more characters are placed side by side to form a word, *Matras* of them generally touch each other. There is also an invisible separation line between middle and lower zone, generally known as a *base line.* Different zones in a Bangla word are shown in Fig. 1.

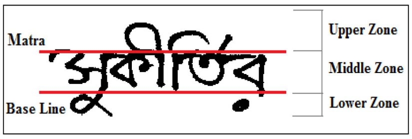

Fig.1. Three zones of Bangla script – upper, middle and lower zones are separated by *Matra* and *Base line.*

Bangla text characters, like other languages, suffer from distortions depending on the writing style of the person. It is observed that due to the touching nature of the components, the segmentation of characters often fails. The "Slant and Skew" nature of handwritten text (see Fig.2) makes the task more difficult. Due to presence of modifiers in non-uniform skew and slanted word images, feature extraction from such word images is not easy. Our recognition framework has been designed to take care of these issues. Here an exhaustive comparative study is performed to evaluate different features in middle zone recognition, and consequently the full word recognition of Bangla words using PHOG feature (as it outperforms other features) is reported. To our knowledge this is the first work where PHOG feature has been used in HMM for handwritten Bangla word recognition.

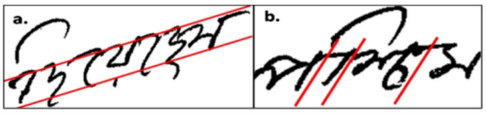

Fig.2. (a) Redlines mark the *Matra* and *Base line* in a Skewed text. (b) A word image shows the slanted segmentation lines between characters.

B. Related Work

Although many research works have been investigated for isolated handwritten characters and digits recognition in Indian script [6], but only a few pieces of work exist towards offline handwritten recognition in Indian script [5]. Recognition of printed Bangla script has already been addressed in [6]. Mainly segmentation [7] and holistic [8] based approaches are used for the word recognition purpose. A number of papers [5] have been developed using character level segmentation in Bangla script. One major problem in such approaches is that segmentation often fails due to overlapping and touching characters. A combination of segmentation and holistic based approaches has been tried in [9]. Only a few pieces of work [10] have been done in Bangla handwritten word recognition using HMM. All these methods adapted lexicon based holistic word recognition where the main drawback is that it fails to give result for unknown words. Character based HMM models [11] have been successfully used for recognition of arbitrary set of words in English/Latin scripts which allows to recognize unknown word. Recently Roy et al.[12] used character based HMM for Bangla text recognition. Present paper extends the work of [12] with better feature and recognition performance.

The rest of the paper is organized as follows. The word recognition framework is explained in Section II. We have demonstrated the performance of different features in Section III. Finally, conclusions and future work are presented.

II. RECOGNITION FRAMEWORK

We have used the HMM-based framework [12] for Bangla handwritten text recognition. The system consists of five main steps namely, preprocessing, zone segmentation, recognition of middle zone text, recognition of upper and lower zone component , and combination of zone wise results to get final results. The architecture of the system has been shown in Fig.3 with a Bangla word example. The details of each of these steps are discussed briefly in following subsections.

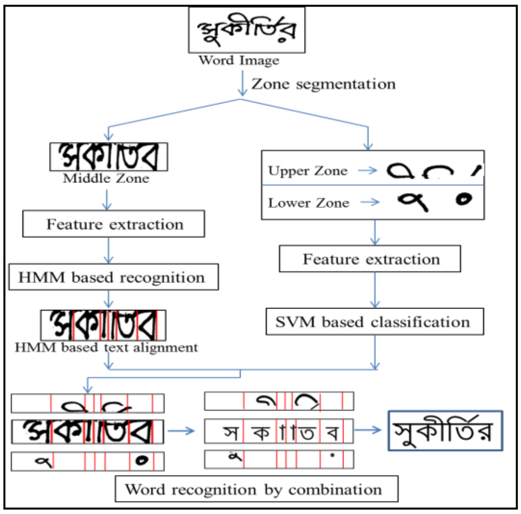

Fig.3. Outline of our word recognition framework

A. Preprocessing

The document image is first binarized by a global thresholding method using Otsu's algorithm. Next, the binary document is segmented into individual text lines using a line segmentation algorithm [13]. After lines are separated, run-length smearing algorithm is applied and connected components are considered as words. For skew-correction, we consider all the points on the extreme bottom of the text stroke and use *Linear Regression* analysis on these points to find out the best fitted line. The slope of the straight line δ represents skew of the text. Thereafter, a rotation by δ is done to correct the skew. The slant angle is determined and corrected using the vertical projection histogram [4]. The algorithm finds the height of the peak in vertical projection analysis at different angle after doing shear transform and considers the angle having the largest peak.

B. Zone Segmentation

To segment the zones we determine the hypothesis of *Matra* using the algorithm defined in our previous work [12]. Next, we form a surrounding window (*W*) of height *H/5* and width of word image around the *Matra* region. *H* is the height of the word image. The skeleton of the image in window *W* is searched to extract the high curvature points, junction points and end points. The skeleton lines which emerge from these points and go outside the *W* are discarded as these are considered as character portion. Remaining portions in *W* are considered as portions of *Matra* [12]. Next, we separate the modifier from the upper portion of *Matra* and call them upper zone modifiers. To separate the lower zone we follow the projection profiles of the word for analysis. The sharp decline in projection peak is noted while moving from middle zone to lower zone. We analyze the histogram and mark the row from where the sharp decline starts. This row, called as base line, isolates the lower zone from middle zone.

C. Middle zone text recognition

The middle zone is the primary portion in Bangla text region where characters are often touching with each other. We apply Hidden Markov Model (HMM) based sequential classifier [14] for recognizing the touching components in this zone. The HMM is used for its capability to model sequential dependencies. An HMM can be defined by its initial state probabilities, state transition matrix and output probability matrix. A separate Gaussian Mixture Model (GMM) is defined for each state of model. The recognition is performed using the Viterbi algorithm. In the training phase, the transcriptions of the middle zone of the word images are used to train the character models.

Feature extraction: We have developed five different features [LGH, PHOG, profile feature [17] by Matri-Bunke, GABOR and G-PHOG to compare the performance in middle zone component recognition., These features are briefly described below.

1) *LGH* [15] feature was used for word spotting in Latin text recognition. A sliding window is being shifted from left to right of the word image with an overlapping between two consecutive frames. Each sliding window patch is next divided into 4x4 cells and from each cell, histogram of gradient (with 8 bins) is computed. The final feature vector is the concatenation of 16 histograms which gives a 128 dimensional feature vector for each sliding window position.

2) *PHOG* [16] is the spatial shape descriptor empowered by spatial layout and local shape, comprising of gradient orientation at each pyramid resolution level. To extract the feature from each sliding window, we divide it into cells at

several pyramid levels. The grid has 4^N individual cells at N resolution level (*i.e.* $N=0,1,2,..$). Histogram of gradient orientation of each pixel is calculated from these individual cells and is quantized into L bins. The concatenation of all feature vectors at each pyramid resolution level provides the final PHOG descriptor. At any individual level, it has $L \times 4^N$dimensional feature vector where N is the pyramid resolution level (i.e. N=0, 1, 2….). So, the final PHOG descriptor consists of $L \times \sum_{N=0}^{N=K} 4^N$dimensional feature vector, where K is the pyramid level. In our implementation, we have limited the level to $K=2$ and considered 8 bins 360º/45º of angular information. Thus, we have (1×8) + (4×8) + (16×8) = (8+32+128) = 168 dimensional feature vector at each sliding window position (see Fig.4).

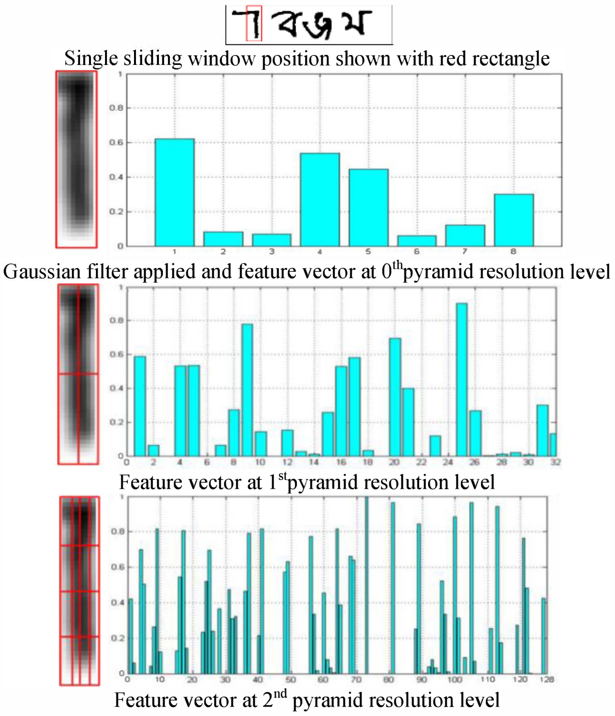

Fig.4. Computation of PHOG feature vector

3) *Marti-Bunke feature* [17] consists of nine features computed from foreground pixels in each image column. Three global features are used to capture the fraction of foreground pixels, the centre of gravity and the second order moment. Remaining six local features comprise of the position of the upper and lower profile, the number of foreground to background transitions, the fraction of foreground pixels between the upper and lower profiles and the gradient of the upper and lower profile with respect to the previous column, which provides dynamic information.

4) *GABOR* features has been applied successfully in Arabic text recognition [20]. We have developed this feature for our text recognition. For this purpose, Gabor filtering in four orientations (0°, 45°, 90° and 135°) are applied and then we used the magnitude as the response for feature extraction. After filtering, the image frame is divided equally into 12 rows. Next, we concatenate the features in each grid to have 48-D Gabor features.

5) We have also made an experiment with a combination of Gabor and PHOG features called *G-PHOG* feature. The idea of G-PHOG is motivated from [21] where Gabor feature has been combined to improve the result. From the experiment using G-PHOG feature, it is noted that the efficiency of Gabor feature can be improved by combining it with PHOG descriptor.

D. Upper and lower zone component recognition

The isolated components which are present in upper and lower zones are segmented using connected component (CC) analysis are recognized and labelled as text characters. Upper zone modifiers (ি , ী , ৈ , র্ , ঁ) and lower zone modifiers (ু, ূ, ৃ, ্র, ্, ়) are separately considered for classification to minimize the error. For classification, the images are resized into 150x150 and then PHOG features of 168 dimensions are extracted from upper and lower zone modifiers. PHOG feature is considered as it provides better result in the experiment. Support Vector Machine (SVM) classifier [18] has been used to classify these components. The *Radial Basis Function (RBF)* kernel function was used in our case because of its better performance. The upper and lower modifiers have been separated by the alignment information as discussed in the following subsection. Combining with the middle zone result, the labels of modifiers are used afterwards to form the entire word.

E. Combination of zone-wise results

For estimating the boundaries of the characters in a Bangla word, Viterbi forced alignment *(FA)* has been used. With embedded training, the optimal alignments of a set of HMMs are found. After alignment the character segments of a given word, hypotheses are obtained (see middle zone alignment in Fig.5). We generate *N*-best Viterbi list composed of *N* hypotheses. *N*-best lists are generated to obtain a set of likely word hypotheses. These associate different labeling and segmentation part. Every pair of segment and label in the list is given a confidence measure by 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. It is performed as follows.

After computing the zone-wise recognition results, they are combined to form the whole word (see Fig.5). The results of alignment are mapped to characters. Let, the result of middle zone recognition is $C_{M\_1}, C_{M\_2} ... C_{M\_N}$ where N is the number of alignments. After mapping in upper and lower zones, the component label of SVM classification are $C_{U\_1}, C_{U\_2} ... C_{U\_N}$ and $C_{L\_1}, C_{L\_2} ... C_{L\_N}$ respectively. Finally, the whole word will be $C_1, C_2 ... C_N$ where $C_i = \boldsymbol{F}(C_{M\_i}, C_{U\_i}, C_{L\_i})$, $i = 1, 2, 3...N$ and $\boldsymbol{F}$ is a mapping function defined according to the Bangla character set. Due to complex handwriting styles, some upper/lower zone modifiers may shift from their original alignment position. To find a particular alignment of C_i at position i we may need to combine the upper and lower zone information from previous $(i-1)^{th}$ or next $(i+1)^{th}$ alignment position. Hence the combination rule of C_i will become $C_i=\boldsymbol{F}(C_{M\_i},C_{U\_i},C_{U\_i+1},C_{U\_i-1},C_{L\_i},C_{L\_i+1},C_{L\_i-1})$. The mapping function $\boldsymbol{F}$ can be simplified by some heuristic properties of Bangla script, e.g., if a particular middle zone character has only one possible upper (lower) zone modifier, then there is no need of recognizing modifier in lower (upper) zone.

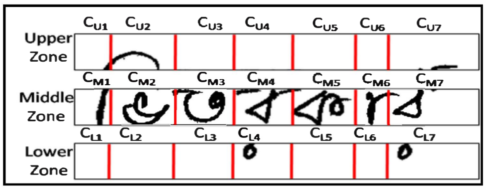

Fig.5. The middle zone alignment result applied to upper and lower zones for modifier separation purpose. All extracted regions are fed in SVM for classification

III. EXPERIMENT & RESULT ANALYSIS

For experiment of the handwritten word recognition scheme, we collected a total of 15,109 handwritten word samples. These words were considered from 60 handwritten document images from individual of different professions. The words are extracted by a line segmentation method followed by word segmentation. Among these word images 11,253 images are used for training and rest 3,856 samples are for testing. A list of 1,547 words is considered in the lexicon.

A. Middle zone recognition

We considered continuous density HMMs with diagonal covariance matrices of GMMs in each state. Five different features are extracted from the training samples. A number of Gaussian mixtures (16, 32, 64, 128 and 256) and state numbers (6, 7, 8, and 9) are tested to evaluate the performance. The qualitative recognition results of middle zone using five features are shown in Fig.6.

	PHOG	LGH	G-PHOG	GABOR	M-B
	✓সামনে	✓সামনে	✓সামনে	✓সামনে	✗মাহনে
	✓আসল	✓আসল	✓আসল	✗আমল	✓আসল
	✓বাঝবাব	✗কাবব	✗বাঝযা	✓বাঝবাব	✗কাববাব
	✓আপনাব	✗আপন	✓আপনাব	✗পাবচয	✓আপনাব
	✗বলপবক	✓বলোছলেন	✗কাহল	✗পাড়যা	✗বালল

Fig.6. Few examples of middle zone recognition results using different features indicating correct (by tick) and incorrect (by cross) labels.

Next, the middle zone recognition results are combined with upper and lower zone modifiers to get the final word level. Because of this flexibility we have analyzed upto top 5 choice results and considered all of them with combination of upper and lowers zone modifiers. The stacked column chart in Fig.7 shows the performance using different Gaussian Mixture and top N choices. It is noted that the 32 Gaussian Mixture PHOG feature provides the best results achieving upto 90.87% accuracy with top 5 choices. PHOG outperformed the other features in middle zone recognition. The nearest LGH feature achieved 89.05% accuracy. The performance with varying state number is shown in Fig.8 and from the figure it can be seen that with state number 6 the best result is obtained.

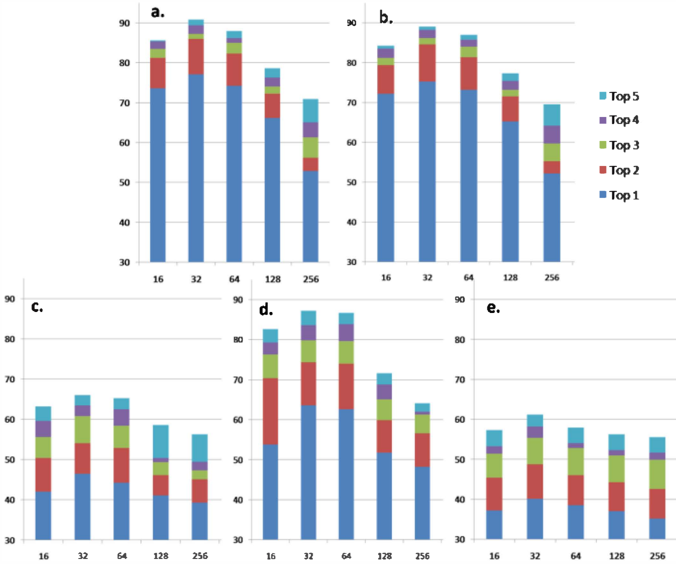

Fig.7. Performance accuracy(along Y axis) plotted against Gaussian Mixtures(along X axis) with Top N choices as parameter for (a) PHOG, (b) LGH, (c) GABOR, (d) G-PHOG, (e) Marti-Bunke feature.

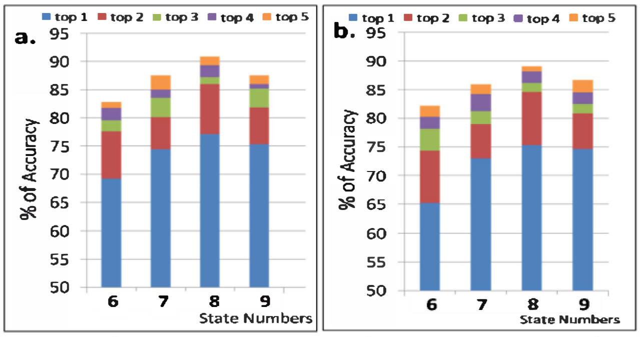

Fig.8. Comparison of accuracy between different state numbers with (a) PHOG & (b) LGH features

Some examples of middle zone recognition results with PHOG feature taking top N choices are shown in Fig.9. It may be noted that first choice from middle zone does not provide correct results always. Correct results are found by combining upper and lower modifiers.

Image					
Choice 1	✓আবাব	✗বনলতা	✗কবব	✓জাম	✗সমান
Choice 2	✗তাবাব	✗কমলতা	✓বলাব	✗জামা	✗সময
Choice 3	✗অবাক	✓কলকাতা	✗কলম	✗জন	✗মহান
Choice 4	✗আকাব	✗পলক	✗কমাব	✗জল	✗সামথ
Choice 5	✗তাব	✗কানন	✗বলা	✗হকাব	✓সমথন

Fig.9. Recognition results of middle zone components considering 5 choices using PHOG feature.

B. Upper and lower zone results

The PHOG feature based recognition results with modifiers in upper and lower zones are shown in Table 1. We have collected a total of 1,723 upper zone modifiers and 1,437 lower zone modifiers from the training dataset. To check the performance we have considered 500 modifiers for testing in each of these zones. The qualitative result using SVM is shown in Fig.10.

Table I: SVM classification result of the upper & lower zone modifiers

	Training data	Testing data	Accuracy (%)	
			Top 1	Top 2
Upper zone	**1,223**	**500**	**87.66**	**97.23**
Lower zone	**937**	**500**	**84.07**	**96.15**

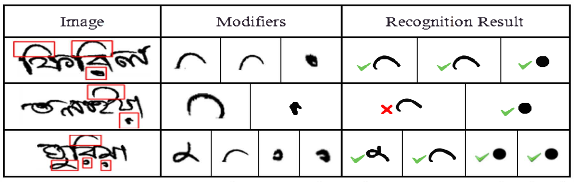

Fig.10. Some examples of modifier classification by SVM with indication of correct (by tick) & incorrect (by cross) one.

C. *Full word recognition result*

After getting zone-wise results from three zones, they are combined to form the full word. The combination is performed according to the mapping function F as discussed in Section II. Some examples of the full word recognition results are shown in Fig.11. The recognition performances at full word level are shown in Table II. We have achieved accuracy of 80.21% and 90.87% with top 1 and top 5 choices, respectively. We obtained 85.74% accuracy when tested on the same dataset reported in [12].

Table II: Accuracy of full word recognition using PHOG feature

Top Choice #	**Top 1**	**Top 2**	**Top 3**	**Top 4**	**Top 5**
Accuracy	**80.21**	**87.01**	**88.16**	**89.88**	**90.87**

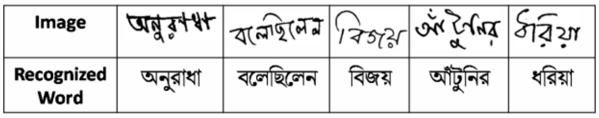

Fig.11. Results of full word recognition

D. *Error analysis*

Because of complex handwriting nature, some Bangla word images are not recognized properly by our system. As our system detects *Matra* and segments the upper and middle zones, unavailability of the *Matra* often made the system fail. Fig.12(a) shows an example where the horizontal projection detects highest peak in wrong place. Hence analyzing the isolated characters in a word would improve such errors and we will work on it in future. Our system may not work for touching of middle zone characters and lower zone modifiers occur at two different position (Fig.12(b)). The present system considers the touching at a single place otherwise we do not segment the component into middle and lower zones.

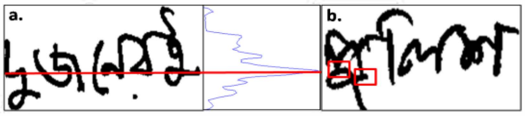

Fig.12. Few examples of erroneous results happened due to improper segmentation of zone modifiers. (a) The *Matra* was not detected properly. (b) Touching of lower zone modifiers at two different positions shown by two red rectangles.

IV. CONCLUSION

In this paper we have performed a comparative study of different features in Bangla handwritten text recognition using HMM. Due to presence of different modifiers proper zone segmentation plays an important role to reduce the number of possible combinations in a character set. According to the experiment result, we have obtained an encouraging result. It was also noted that PHOG feature outperforms other features for middle zone recognition.

REFERENCES

[1] L. Koerich, R. Sabourin, and C. Y. Suen, “Recognition and verification of unconstrained handwritten words”. T-PAMI,vol.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 T-PAMI, vol.24, pp.1425-1437, 2002

[3] V. Märgner and H.E. Abed, “Arabic handwriting recognition competition”, ICDAR, vol.2, pp. 1274-1278, 2007.

[4] E. Kavallieratou, N. Fakotakis, and G. Kokkinakis, “Slant estimation algorithm for OCR system” Pattern Recognition,vol.34, pp. 2515–2522, 2001.

[5] U. Pal, R. Roy, F. Kimura, “Multi-lingual City Name Recognition for Indian Postal Automation”. ICFHR, pp.169-173, 2012.

[6] U. Pal and B. B. Chaudhuri, “Indian script character recognition: A survey”, Pattern Recognition, vol.37, pp.1887-1899, 2004.

[7] R. Plamondon and S. N. Srihari, “On-line and off-line handwritten recognition: A comprehensive survey”, IEEE T-PAMI, vol.22, pp.62-84, 2000.

[8] S. Madhvanath and V. Govindaraju, “The role of holistic paradigms in handwritten word recognition”, IEEE T-PAMI, 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 T-PAMI, vol.16, pp.481-496, 1994.

[10]T.Bhowmik, U.Roy, S.K.Parui, “Lexicon Reduction Technique for Bangla Handwritten Word Recognition”, DAS, pp. 195-199, 2012

[11]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 T-PAMI, vol.33, pp. 2066-2080, 2011.

[12]P. P. Roy, P. Dey, S. Roy, U. Pal, F. Kimura, “A Novel Approach of Bangla Handwritten Text Recognition using HMM”, ICFHR, pp.661-666, 2014.

[13]P. P. Roy, U. Pal and J. Lladós, “Morphology Based Handwritten Line Segmentation Using Foreground and Background Information”, ICFHR, pp. 241-246, 2008.

[14]S. Young. The HTK Book, Version 3.4, 2006.

[15]J. R. Serrano and F. Perronnin, “Handwritten word-spotting using Hidden Markov Models and universal vocabularies”, ICPR,vol.42, pp.2106-2116, 2009.

[16]Y. Bai, L. Guo, L. Jin, Q. Huang, “A novel feature extraction method using Pyramid Histogram of Orientation Gradients for smile recognition". ICIP, pp.3305-3308, 2009.

[17]U. V. Marti and H. Bunke, “Using a statistical language model to improve the performance of an HMM-based cursive handwriting recognition system,” IJPRAI, vol. 15, pp. 65–90, 2001.

[18]V. Vapnik, “The Nature of Statistical Learning Theory”, Springer Verlang, 1995.

[19]Wikipedia reference, http://en.wikipedia.org/wiki/Bengali_alphabet.

[20]J. Chen, H. Cao, R. Prasad, A. Bhadwaj, and P. Natarajan, “Gabor features for offline Arabic handwriting recognition”, Document Analysis Systems, pp.53–58,2010.

[21]H. Cao, R. Prasad, P. Natarajan, “Handwritten and Typewritten Text Identification and Recognition Using Hidden Markov Models”, ICDAR, pp. 744-748, 2011.