Off-line Sinhala Handwriting Recognition using Hidden Markov Models

S. Hewavitharana[†]
sanjika@cmb.ac.lk

H. C. Fernando[‡]
chandrika@sliit.lk

N.D. Kodikara[†]
nihal.uoc@mail.cmb.ac.lk

[†]Department of Computer Science, University of Colombo, Colombo, Sri Lanka
[‡]Sri Lanka Institute of Information Technology, Colombo, Sri Lanka

Abstract

This paper describes a method to recognize off-line handwritten Sinhala characters, the language used by the majority of Sri Lanka. The classification approach is based on discrete hidden Markov models. A subset of the Sinhala alphabet was chosen for the study. The unknown characters are first pre-classified into one of three character groups, based on the structural properties of the text line. This resulted in 99.9% accuracy. The HMM classifier is then used for the final recognition. The system was trained using 750 handwritten character images. A separate set of 500 character images was used to test the system. All the characters were written by 5 different writers on a pre-formatted paper. Results have shown 64.3% recognition rate for the first choice and 92.1% recognition rate up to the third choice.

1. Introduction

The process of handwriting recognition involves extraction of some defined characteristics called features to classify an unknown handwritten character into one of the known classes. A typical handwriting recognition system consists of several steps, namely: preprocessing, segmentation, feature extraction, and classification. Several types of decision methods, including statistical methods, neural networks, structural matching (on trees, chains, etc.) and stochastic processing (Markov chains, etc.) have been used along with different types of features. Many recent approaches mix several of these techniques together in order to obtain improved reliability, despite wide variation in handwriting.

Sinhala is one of the official languages of Sri Lanka used by the majority of people. It belongs to the Indo-Aryan group of languages along with Hindi, Gujarati, Bengali and other north Indian languages. Sinhala script is alphabetic in nature and is written from left to right. The alphabet consists of 61 symbols: 18 vowels, 2 semi-vowels and 41 consonants. Letters in a word are written separated from each other, in a non-cursive manner.

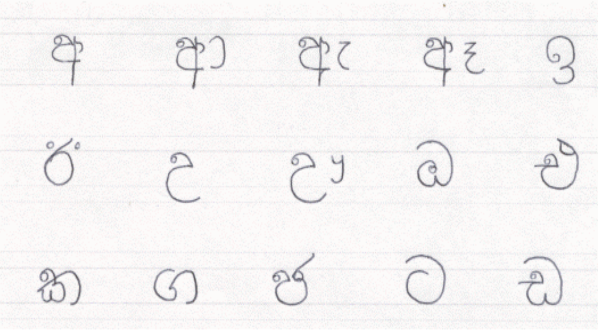

Figure 1: A sample of handwritten Sinhala characters

There have been only a few attempts in the past to address the recognition of printed or handwritten Sinhala language script. Mannapperuma [1] has used a structural approach to recognize printed characters while Rajapakse [2] has used a neural network approach to handwritten character recognition.

In this study, we exploit the use of Hidden Markov Models (HMMs) for off-line Sinhala handwriting recognition. HMMs have been widely used in the field of speech recognition [3] and more recently in handwriting recognition [4] [5]. Although most of these handwriting recognition applications concentrate on cursive handwriting and on-line handwriting, there have been attempts where HMMs were used on isolated handwritten characters [6].

A subset of the Sinhala alphabet, which consists of the most commonly used 25 letters, is used for the study. The recognition process is split into two major sections: preliminary classification and recognition using HMMs. Firstly, structural properties of the handwritten character line are used to pre-classify an unknown character into a subset of its candidate characters. Then, the candidate characters are further analyzed using HMMs.

This paper is organized as follows. Section 2 explains the process of reference line identification and segmentation. In Section 3, we propose the method for preliminary classification based on the reference lines identified in the previous section. It also includes the feature extraction process. Recognition by hidden Markov modelling is then described in Section 4. Experimental results for a sample of handwritten images are provided in Section 5 followed by concluding remarks in Section 6.

2. Preprocessing and Segmentation

We use A4 sized paper to collect sample handwriting. A single document typically includes 10 lines with 3-4 words in each line. The documents are scanned at a resolution of 100 dpi and binarized using an adaptive thresholding technique. Then, the image is segmented into constituent text lines using the horizontal projection profile. For each line of text, four reference lines are extracted namely: upper line, upper baseline, lower baseline and lower line. These four lines determine three zones namely: the upper, core and lower zones.

The process of reference line extraction is similar to the method described in [7]. Zero values in the projection profile correspond to horizontal gaps between lines. The maximum and minimum zero value positions adjoining a text line are taken as the line boundaries corresponding to the *lower line* and *upper line* respectively. The *upper baseline* and *lower baseline* are identified using the first derivative of the horizontal projection profile. The local extrema of the first derivative in the two halves of the text line image are assumed to correspond to the two baselines.

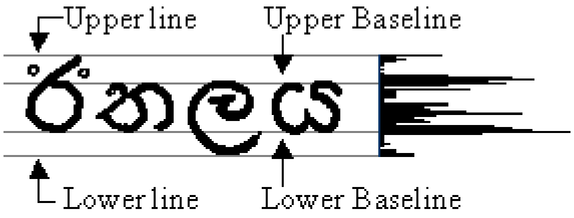

Figure 2: Reference line extraction using horizontal projection profile and its first derivative

In some text lines, the upper or lower zones do not contain any character, in which case, the direct application of the above method may fail. Therefore, some heuristic rules are included in the segmentation procedure to handle such situations.

To simplify the process of reference line extraction, a pre-formatted paper is used to collect handwriting, which contained all four reference lines on it. However, these lines are completely eliminated during the binarization of the image and has no effect on the segmentation.

After the reference lines have been found, the individual words and characters are extracted using the vertical projection profile of each text line.

Once the characters are segmented, the minimum bounding box of each character is identified eliminating the white space around it. Upper and lower boundary values of the minimum bounding box relative to the character line, are send to the next stage for preliminary classification.

3. Preliminary Classification and Feature Extraction

Sinhala characters can be classified into three non-overlapping groups based on their relative heights in the 3-zone frame (Figure 3).

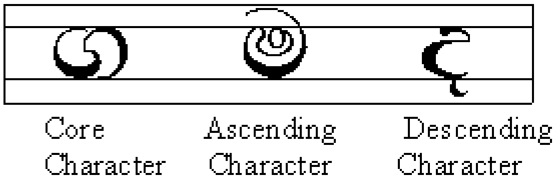

Figure 3: Three pre-classification groups

The aim of the preliminary classification process is to classify an unknown character into one of the above three groups so that the number of candidate characters is restricted to the members of that group.

For each character, its upper and lower boundaries are compared with the four reference lines, and the pre-classification is performed according to the following algorithm (Figure 4).

```
Input:
UB /* Upper Boundary of the character */
LB /* Lower Boundary of the character */
uBase /* Upper Baseline of the line */
lBase /* Lower Baseline of the line */
```

```
Output:
char_class /* preliminary class */

for each character in the image
if (UB>=uBase)^(LB<=lBase)
  /* core character */
   char_class = core
elseif (UB<UBase)^(LB<=lBase)^(LB>UBase)
   /* ascending charcter */
   char_class = ascender
elseif (UB>=Ubase)^(LB>lBase)^(UB<lBase)
   /* descending charcter */
     char_class = descender
```

Figure 4: Pre-classification algorithm.

Following table lists all the characters in the selected domain, categorized into the above three groups.

Group	Characters of the group	No.
Core Characters	ක ග ත න ප ය ස හ	8
Ascending Characters	ඊ ඵ ඹ ජ ට ඩ ණ බ ම ර ව	11
Descending Characters	අ ඉ උ ද ල ළ	6

Table 1: Preliminary segmentation of the character set

The images of segmented characters are then rescaled into 32x32 pixel size, using a bilinear interpolation technique. Each image is divided into vertical and horizontal strips (Figure 5). Each strip is then subdivided into sections of size 4x4 pixels. A vector is created in each of the two directions using the pixel density of each section.

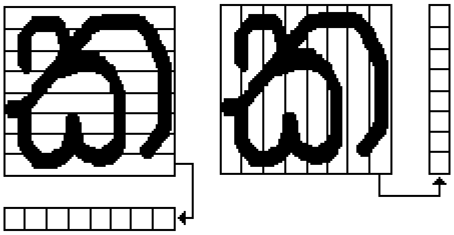

Figure 5: Creation of feature vectors in horizontal and vertical directions

4. Recognition using HMMs

A discrete HMM is characterized by the following [3]:

1. **N**, the number of states in the model. The set of states in the model is $S = \{1, 2, \ldots, N\}$, the state at time t is denoted as q_t.

2. **M**, the discrete alphabet size. We denote the individual symbols as $V = \{v_1, v_2, \ldots, v_M\}$.

3. **A** = $\{a_{ij}\}$, the state transition probability distribution where
$$a_{ij} = \mathrm{P}[q_{t+1} = S_j \mid q_t = S_i],\ 1 \leq i, j \leq N.$$

4. **B** = $\{b_j(k)\}$, the observation symbol probability distribution in state j, where
$$b_j(k) = \mathrm{P}[v_k \text{ at } t \mid q_t = S_j],\ 1 \leq j \leq N$$
$$1 \leq k \leq M$$

5. **π** = $\{\pi_i\}$, the initial state distribution, where
$$\pi_i = \mathrm{P}[q_1 = S_i],\ \ 1 \leq i \leq N$$

A compact notation for the above HMM would be,
$$\lambda = \{A, B, \pi\}.$$

Maximum likelihood parameter estimation for HMM is obtained by the iterative procedure, the *Baum-Welch algorithm* [8], with multiple observation sequences. Then, for a given observation sequence $O = \{O_1, O_2, \ldots, O_T\}$, the HMM is used to compute $\mathrm{P}[O \mid \lambda]$, where T is the length of the sequence.

Two HMMs are created for every character, one for modelling the horizontal information and the other for modelling the vertical information. The discrete hidden Markov character models are trained using standard procedures [8][9]. The number of states for all the character HMMs is fixed and no skip states are allowed. Only the pre-classified candidate characters are passed on for HMM recognition.

Two log probabilities for each candidate character are calculated using the horizontal direction HMM and vertical direction HMM. Then, the log probabilities are added together to obtain a final 3-best list for character recognition.

5. Experimental Results

The training set contained 750 characters with 30 samples from each character class. The test set contained about 500 characters.

All the data were obtained from five writers on the pre-formatted papers. Each document contained 10 text lines of some meaningful words with approximately 100 characters. Writers were allowed to write freely with a varying frequency of characters of each class.

A total of 50 text lines were subjected to segmentation and reference line identification. The segmentation procedure was able to correctly identify all the reference lines. The preliminary classification resulted in 99.9% accuracy. The pre-classified images were then fed into HMM recognition system.

The results of the HMM recognition are as follows.

Top 1	Top 2	Top 3
64.3%	84.5%	92.1%

Out of 500 characters investigated, 321 characters were successfully classified into the correct group, resulting 64.3% recognition accuracy for the first choice. Over 90% recognition accuracy was reported for the top 3 choices.

6. Concluding Remarks

In this paper we have presented a system for recognizing handwritten Sinhala characters. A discrete hidden Markov model based classifier was used for the recognition, yielding classification accuracy of 64.3% for the first choice and 92.1% for the top three choices. Our feature extraction method reduced the two dimensional spatial information of character images into a single dimension array of values, thereby throwing away some information. This might have under-represented the character classes and hence contributed to the low recognition accuracy of the first choice. Since handwritten characters can be considered as a construction of line segments at different orientations and lengths, an orientation selective method such as Gabor filtering should produce effective features [10]. We hope to modify the feature vector with measurements obtained from filtered oriented parts using the above method. Future versions of the system are planned to cater for the full Sinhala alphabet including consonant modifiers. The use of a language model, which introduces linguistic knowledge into the system and thereby improves recognition accuracy, is also under consideration. We strongly feel that the preliminary classification method used in this study is applicable to other Indian languages such as Tamil.

References

[1] H. Mannapperuma, *A Method to Recognize Sinhala Characters*, Dissertation submitted for B. Sc. (Computer Science), University of Colombo, October 1994.

[2] R. K. Rajapakse, *A Neural Network based Character Recognition System for Sinhala Script*, M. Sc. Thesis, University of Colombo, October 1995.

[3] L. R. Rabiner, A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition, *Proc. IEEE* , vol. 77, no. 2, 1989, pp. 257-285.

[4] H. Bunke, M. Roth and E. Talamazzini, Off-line Cursive Handwritten Recognition using Hidden Markov Models, *Pattern Recognition*, vol. 28, no. 9, 1995, pp. 1399-1413.

[5] H.J. Kim, K.H. Kim, S.K. Kim and J.K. Lee, On-line Recognition of Handwritten Chinese Characters based on Hidden Markov Models, *Pattern Recognition*, vol. 30, no. 9, 1997, pp. 1489-1500.

[6] G. Loudon, C. Hong, Y. Wu and R. Zitserman, The Recognition of Handwritten Chinese Characters from Paper Records, *IEEE TENCON, Digital Signal Processing Applications*, 1996, pp. 923-926.

[7] R.M. Bozinovic and S.N. Srihari, Off-line Cursive Script Word Recognition, *IEEE Trans. on PAMI*, vol. 11, no. 1, 1989, pp. 68-83.

[8] F. Jelinek, *Statistical Methods for Speech Recognition*, MIT-Press, 1998.

[9] L. R. Rabiner and B. H. Juang, *Fundamentals of Speech Processing*, Prentice-Hall, 1993.

[10] Y. Zhu, T. Tan and Y. Wang, Font Recognition based on Global Texture Analysis, *IEEE Trans. on PAMI*, vol. 23, no. 10, 2001, pp. 1192-1200.