A New HMM for On-Line Character Recognition Using Pen-Direction and Pen-Coordinate Features

Yoshinori Katayama, Seiichi Uchida, and Hiroaki Sakoe
Faculty of Information Science and Electrical Engineering, Kyushu University, 819-0395, Japan
{yosinori,uchida}@is.kyushu-u.ac.jp

Abstract

A new hidden Markov model (HMM) is proposed for on-line character recognition using two typical features, pen-direction feature and pen-coordinate feature. These two features are quite different in their stationarity; pen-direction feature is stationary within every line segment of a stroke whereas pen-coordinate feature is not. In the proposed HMM, these contrasting features are used in a separative and selective way. Specifically speaking, pen-direction feature is outputted repeatedly at intra-state transition whereas pen-coordinate feature is outputted once at inter-state transition. The superiority of the proposed HMM over the conventional HMMs was shown through single-stroke and multi-stroke character recognition experiments.

1. Introduction

HMM has been employed in on-line character recognition because of its promising ability to model geometric and temporal deformations of a single stroke. HMM is a network model with several states. At a transition between states, it outputs features which represent some local property of a stroke, according to a probability distribution assigned to the state transition. The parameters of the probability distribution determine the range of the deformations.

There are several issues on designing HMM. The first issue is its topology and the number of states. For on-line character recognition, left-to-right topology with self-transition has been employed in general. The number of states has often been determined according to the complexity of the stroke to be modeled. The second issue is its output features for representing strokes. This paper mainly concerns the second issue.

Pen-direction feature and pen-coordinate feature have been widely employed in on-line character recognition. Pen-direction feature has almost always been employed as an output feature of conventional HMMs [1, 2, 3]. In contrast, pen-coordinate feature has not always been employed in HMM whereas it has been employed in dynamic programming-based methods. This may be because pen-coordinate feature is non-stationary within a stroke and this property does not agree with the assumption that features are stationary within a state of HMM. As shown in Section 2, however, the lack of pen-coordinate feature induces the serious problem on, especially, recognizing multi-stroke characters; roughly speaking, relative position and length of strokes cannot be regulated properly without pen-coordinate feature.

The proposed HMM utilizes both pen-direction and pen-coordinate features. The main idea is a separative and selective use of these two features. Specifically speaking, pen-direction feature is outputted repeatedly at intra-state transition (i.e., within a state) and pen-coordinate feature is outputted once at inter-state transition (i.e., between states). By this separative and selective use, pen-direction feature represents stationary parts of a stroke and pen-coordinate feature represents non-stationary parts (i.e., start, end, and bending parts) of a stroke.

2. Conventional HMMs

2.1. HMM only with pen-direction feature

Let $xy_1, \cdots, xy_t, \cdots, xy_T$ represent a single stroke, where $xy_t = (x_t, y_t)^t$ is pen-coordinate feature at time t. Pen-direction feature at time t is often defined from pen-coordinate feature by $\theta_t = \arg(xy_t - xy_{t-1})$.

Figure 1 shows a conventional HMM, hereafter called θ-HMM [1, 2, 3]. In θ-HMM, a stroke is supposed to be a sequence of line segments and a state is assigned to each line segment. For example, a θ-HMM for an "L"-shaped stroke has two states. Since pen-directional feature is stationary within a line segment, pen-direction features outputted from a state properly represent the direction of the line segment.

Training of θ-HMM can be done by the well-known Baum-Welch algorithm, where, for example, the following forward variable $\alpha_t(i)$ is calculated:

$$\alpha_t(i) = \alpha_{t-1}(i-1)\, a_{i-1,i}\, b_{i-1,i}(\theta_t) + \alpha_{t-1}(i)\, a_{i,i}\, b_{i,i}(\theta_t), \quad (1)$$

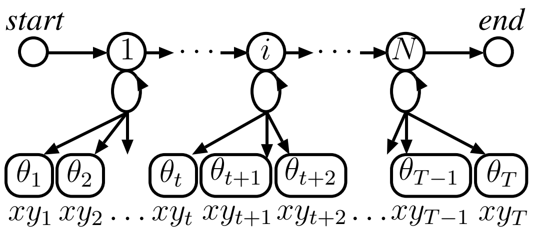

Figure 1. θ**-HMM.**

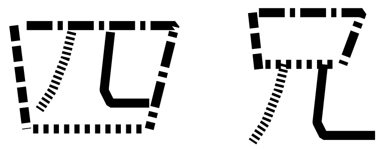

Figure 2. Multi-stroke characters with same pen-direction features.

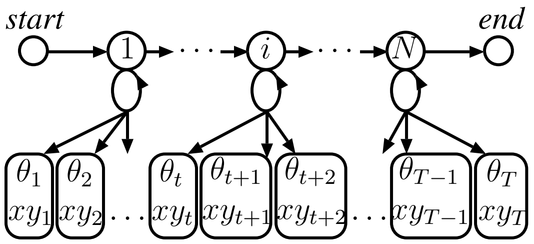

Figure 3. $xy\theta$**-HMM.**

where $a_{i,j}$ is the transition probability from state i to j, and $b_{i,j}(\theta_t)$ is the observation probability of θ_t at the transition from state i to j. Note that, the formulation of (1) is somewhat different from the popular one for the later discussion and $b_{i-1,i}(\theta_t) = b_{i,i}(\theta_t)$ in θ-HMM.

θ-HMM has been successfully applied to single-stroke character recognition (where a single HMM represents a whole character); however, it is hard to expect its good performance on multi-stroke character recognition (where a single θ-HMM represents a single stroke of a multi-stroke character). This is because relative position and length of strokes cannot be regulated without pen-coordinate feature. Performance will be far degraded on stroke-order free condition. For example the characters in **Figure 2** can not be distinguished without pen-coordinate feature in stroke-order free condition. Previous attempts [1, 2] have dealt with this problem by concatenating strokes with virtual strokes at pen-up parts. That is, they treat multi-stroke characters as single-stroke characters. Consequently, they must prepare HMMs for all possible stroke order variations.

2.2. HMM with pen-coordinate feature

As shown in **Figure 3**, it is possible to design an HMM[4], called $xy\theta$-HMM, where pen-coordinate feature is used simultaneously with pen-direction feature. It is also possible to design another HMM[2], called xy-HMM, where only pen-coordinate feature is used. As noted before, pen-coordinate feature is non-stationary, i.e., its distribution changes largely with time. A naive remedy to deal with this non-stationarity in HMM is a drastic increase of states ($N \sim T$); this remedy, however, may spoil the merit of HMM and requires a larger number of training samples for estimating its parameters. In fact, deterministic framework such as stochastic DP [5, 6] will be more reasonable than the remedy.

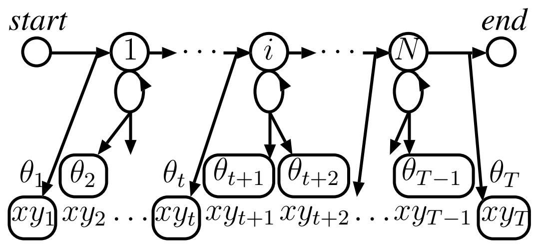

Figure 4. The proposed (xy/θ)**-HMM.**

3. The proposed HMM

Figure 4 shows the proposed HMM, hereafter called (xy/θ)-HMM, where pen-direction and pen-coordinate features are selectively utilized in a single HMM with careful consideration for their different stationarity. Similarly to θ-HMM, (xy/θ)-HMM is comprised of states, each of which is assigned to a line segment of a stroke. The key and novel idea of (xy/θ)-HMM is to introduce pen-coordinate feature for representing the start and the end position of the line segment. Pen-coordinate feature is outputted at each inter-state transition without repetition. In contrast, pen-direction feature is used for representing the direction of the line segment and outputted at every intra-state transition repeatedly like θ-HMM.

The forward variable $\alpha_t(i)$ is calculated by

$$\begin{aligned} \alpha_t(i) &= \alpha_{t-1}(i-1)\, a_{i-1,i}\, b_{i-1,i}(xy_t) \\ &\quad + \alpha_{t-1}(i)\, a_{i,i}\, b_{i,i}(\theta_t), \end{aligned} \tag{2}$$

where $b_{i-1,i}(xy_t)$ is a pen-coordinate feature distribution at the inter-state transition from state $i-1$ to i. Eq. (2) shows that pen-direction and pen-coordinate features are selectively observed at intra-state and inter-state transitions, respectively. Similarly to θ-HMM, training of (xy/θ)-HMM can be done by the Baum-Welch algorithm, whose details are omitted here.

Table 1. Statistics of dataset. The parenthesized number shows the number of samples with regular stroke order.

#strokes	5	10	15	20
#categories	67	76	21	2
#samples	28,525 (22,473)	17,605 (10,710)	3,336 (1,506)	624 (155)

Table 2. Error rates (%) at single-stroke character recognition.

#original strokes	5	10	15	20
θ-HMM	8.62	0.14	0.0	0.0
(xy/θ)-HMM (proposed)	1.38	0.04	0.0	0.0

Table 3. Error rates (%) at multi-stroke character recognition.

#strokes	5	10	15	20
θ-HMM	69.42	59.73	62.13	19.1
xy-HMM	10.04	4.58	2.25	1.0
$xy\theta$-HMM	7.81	1.91	1.08	0.0
(xy/θ)-HMM (proposed)	3.80	0.47	0.39	0.0

4. Experimental results

Two experiments were conducted for showing the superiority of the proposed (xy/θ)-HMM over θ-HMM, xy-HMM, and $xy\theta$-HMM. The first experiment was recognition of single-stroke characters, each of which was artificially created by concatenating strokes of a multi-stroke character. The second experiment was stroke-order free recognition of multi-stroke characters.

4.1. Data sets

The public on-line Chinese character database called “HANDS-kuchibue_d-97-06-10”[7] was used as experimental data set. From the database, samples with 5, 10, 15, and 20 strokes were used in the dataset of the experiment. **Table 1** shows statistics of the dataset. All the samples were linearly rescaled to be 128 × 128, smoothed, and resampled.

4.2. Training of stroke HMMs

Each model of a Chinese character, i.e., a multi-stroke character was represented as a “set” of HMMs, each of which represents a stroke. For example, five stroke HMMs were prepared for a five-stroke Chinese character “右” (“*right*”=“/”+“–”+“|”+“¬”+“_”). The parameters of the HMM (i.e., the mean and (co)variance of the Gaussian probability distribution) were trained by using 2/3 samples of the dataset. The number of states varied from one to four (except for the start and the end states) according to the number of line segments of the stroke. For example, an HMM for an “L”-shaped stroke has two states.

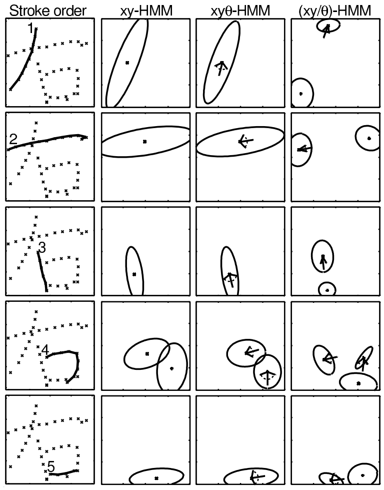

Figure 5. Learned feature distributions.

Figure 5 shows the learned Gaussian distributions of the output features of xy-HMM, $xy\theta$-HMM, and the proposed (xy/θ)-HMM trained for a Chinese character (“右”). Each ellipsoid shows the mean value and the $\pm 2\sigma$-range of pen-coordinate feature. A short line set emerged from the center of the ellipsoid shows the mean value and the $\pm 2\sigma$-range of pen-direction feature.

As shown by this figure, the pen-coordinate feature distributions of xy-HMM and $xy\theta$-HMM have very large variance. This is because non-stationary pen-coordinate features was forcefully assigned to one state. These unfocused distributions may over-estimate the likelihood of the strokes with unexpected shapes and then cause misrecognitions.

On the other hand, the proposed (xy/θ)-HMM succeeded to obtain the pen-coordinate feature distribution of the start, end, and bending positions of the stroke while keeping proper locality. For example, the bending position at the fourth stroke was learned properly.

4.3. Recognition of single-stroke characters

A comparative study between the proposed HMM and the conventional θ-HMM made through a recognition experiment of single-stroke characters. In this experiment, the character model of a class was prepared as a long HMM by concatenating the stroke HMMs of the character. Similarly, each test sample was prepared as a long stroke by artificially concatenating strokes of a sample written in the regular stroke order. Thus, for example, 22,473 samples were used for evaluating the

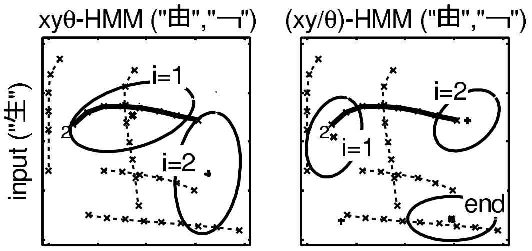

Figure 6. Improved sample. Ellipsoids represent the feature distributions of the "¬"-shaped stroke of each HMM.

recognition rate of 5-stroke characters as shown in **Table 1**.

Table 2 shows the recognition result. Note that those rates were obtained through 3-hold cross validation. The proposed (xy/θ)-HMM always achieved lower error rate than θ-HMM. This result shows that pen-coordinate feature is necessary even for the recognition of single-stroke characters[1]. Further investigation for revealing the characteristics of the proposed HMM will be made in the next experiment.

4.4. Recognition of multi-stroke characters

Another comparative study made through a recognition experiment of multi-stroke characters. Since those characters have writing order fluctuation, a stroke-order free recognition method must be used for optimizing one-to-one correspondence between input strokes and stroke HMMs under the maximum likelihood criterion. For this correspondence optimization, the cube search algorithm [8, 9] was employed.

Table 3 shows the error rates. The proposed (xy/θ)-HMM achieved the lowest error rate and its superiority over the other HMMs was indicated experimentally. It is also shown that the performance of θ-HMM was very poor in multi-stroke character recognition; this is because of the lack of pen-coordinate feature as noted in Section 2.1.

Figure 6 shows a sample "生"("*live*") which was correctly recognized by the proposed HMM and misrecognized as "由"("*reason*") by $xy\theta$-HMM, which achieved the best rate among the conventional HMMs. In this example, the top "$-$"-shaped stroke of the input "生" wrongly fitted to the $xy\theta$-HMM of the "¬"-shaped stroke of "由" with a high likelihood value. This over-estimation was caused by the unfocused pen-coordination feature distribution of the stroke HMM. This problem was solved at (xy/θ)-HMM; the top "$-$"-shaped stroke of the input "生" could not be fitted to the $(xy/\theta) - HMMof$the "¬"-shaped stroke of "由" because the stroke HMM properly requests that "¬"-shaped stroke should be ended at around right-bottom corner.

[1]The result also suggests that the proposed method will be effective for recognizing other characters such as English cursive script.

5. Conclusion

A novel HMM, called (xy/θ)-HMM, has been proposed for on-line character recognition. In the proposed HMM, not only pen-direction feature but also pen-coordinate feature were selectively utilized in a single HMM with careful consideration for different stationarity of pen-direction feature and pen-coordinate feature. The training results of experiments showed that the proposed HMM could learn the positions of the start, the end, and the bending points of a stroke and local directions between those points. The experimental results of both single-stroke character recognition and multi-stroke character recognition showed superiority of the proposed HMM over the conventional HMMs. Especially, the results established a clear superiority over the popular HMM (θ-HMM) where only the pen-direction feature was used.

References

[1] M. Nakai, N. Akira, H. Shimodaira, and S. Sagayama. Sub-stroke approach to HMM-based on-line Kanji handwriting recognition. *Proc. ICDAR*, pages 491–495, 2001.

[2] H. Itoh and M. Nakagawa. An on-line handwritten character recognition method based on hidden markov models. *Technical report of IEICE*, PRMU97-85, 1997 (In Japanese.).

[3] J. Tokuno, Y. Yang, G. P. da Silva, A. Kitadai, and M. Nakagawa. Pen-coordinate information modeling by SCPR-based HMM for on-line Japanese handwriting recognition. *Proc. ICPR*, 3:348–351, 2006.

[4] K. C. Jung, S. M. Yoon, and H. J. Kim. Continuous HMM applied to quantization of on-line Korean character spaces. *Pattern Recognit. Letters*, 21(4):303–310, 2000.

[5] S. Nakagawa. Speaker-independent consonant recognition in continuous speech by a stochastic dynamic time warping method. *Proc. ICPR*, pages 925–928, 1986.

[6] C. Bahlmann and H. Burkhardt. The writer independent online handwriting recognition system *frog on hand* and cluster generative statistical dynamic time warping. *IEEE Trans. PAMI*, 26(2):299–310, 2004.

[7] M. Nakagawa and K. Matsumoto. Collection of on-line handwritten Japanese character pattern databases and their analyses. *IJDAR*, 7(1):69–81, 2004.

[8] H. Sakoe and J. P. Shin. A stroke order search algorithm for online character recognition. *Research Report on ISEE of Kyushu Univ.*, 2(1):99–104, 1997 (In Japanese.).

[9] C. L. Liu, S. Jaeger, and M. Nakagawa. Online recognition of Chinese characters: the state-of-the-art. *IEEE Trans. PAMI*, 26(2):198–213, 2004.