Generation of Hierarchical Dictionary for Stroke-order Free Kanji Handwriting Recognition Based on Substroke HMM

Mitsuru NAKAI, Hiroshi SHIMODAIRA and Shigeki SAGAYAMA[†]

Graduate School of Information Science,
Japan Advanced Institute of Science and Technology
{mit,sim}@jaist.ac.jp

[†]Graduate School of Information Science and Technology,
The University of Tokyo
sagayama@hil.t.u-tokyo.ac.jp

Abstract

This paper describes a method of generating a Kanji hierarchical structured dictionary for stroke-number and stroke-order free handwriting recognition based on sub-stroke HMM. In stroke-based methods, a large number of stroke-order variations can be easily expressed by just adding different stroke sequences to the dictionary and it is not necessary to train new reference patterns. The hierarchical structured dictionary has an advantage that thousands of stroke-order variations of Kanji characters can be produced using a small number of stroke-order rules defining Kanji parts. Moreover, the recognition speed is fast since common sequences are shared in a substroke network, even if the total number of stroke-order combinations becomes enormous practically. In experiments, 300 different stroke-order rules of Kanji parts were statistically chosen by using 60 writers' handwritings of 1,016 educational Kanji characters. By adding these new stroke-order rules to the dictionary, about 9,000 variations of different stroke-orders were generated for 2,965 JIS 1st level Kanji characters. As a result, we successfully improved the recognition accuracy from 82.6% to 90.2% for stroke-order free handwritings.

1. Introduction

The hidden Markov model (HMM) has been successfully applied to alphanumeric on-line handwriting recognition [3, 5] and Japanese handwriting recognition [1, 9]. In their study, the "character-based HMM" has been employed, hence the number of HMMs is identical to the number of distinct characters to be recognized. To recognize more than 6,000 Kanji characters, we have proposed "substroke-based HMM" approach [6] in which every Kanji character can be represented as a concatenation of only 25 kinds of substroke models. The advantages of substroke HMM are summarized as follows. 1) The memory required for models and dictionary is small. 2) Recognition speed is fast by using efficient substroke network search. 3) Un-trained characters can be recognized by just adding their definitions of substroke sequence to the dictionary. 4) Characters written with different stroke-order can be recognized by using multiple definitions in the dictionary. 5) Writer adaptation is performed with a few training characters. In this paper, we focus on the 4th advantage since the other advantages were already discussed in a previous paper [6].

Some stroke-number and stroke-order free recognition methods have been proposed [8, 10]. Also, we have proved that substroke HMM based method does not depend on the stroke-number [7] since the pen-up-down information is not utilized for handwriting features. On the stroke-order problem, there are mainly two approaches; first one is to register different reference patterns in the dictionary [1, 4] and second one is to search the stroke-order simultaneously in a decoding process [8, 10]. In the stroke-based model, the second approach is useful and all possible stroke-order variations can be expressed as a permutation of stroke models. However, some pruning techniques are required since stroke-numbers of Kanji characters are large and it is not practical to search all of stroke-order variations. To solve this problem, we search only probable stroke-order variations by using prior knowledge of stroke-order rules defined in hierarchical structured dictionary. In the following sections, we describe a statistical method of extracting effective stroke-order rules from samples and propose a fast recognition method for various stroke-order handwritings.

2. Handwriting Recognition Based on Substroke HMM

2.1. Feature Vectors

In this paper, we use only pen positions (x, y) though a pen tablet also provides other information such as pen pressure [7] and pen tilts. Let $(\Delta x, \Delta y)$ be the difference between two consecutive pen positions sampled every certain period and (r, θ) be the feature vector, where $r = \sqrt{(\Delta x)^2 + (\Delta y)^2}$ means a velocity of the pen movement and θ means the direction of the velocity vector.
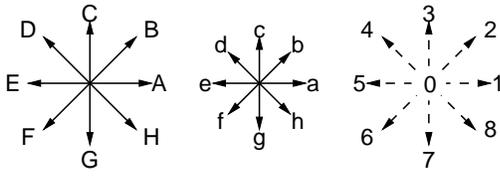
Figure 1. Substroke categories: A–H (a–h) are long (short) substrokes with pen-down and 0–8 are directions of pen-up movement.
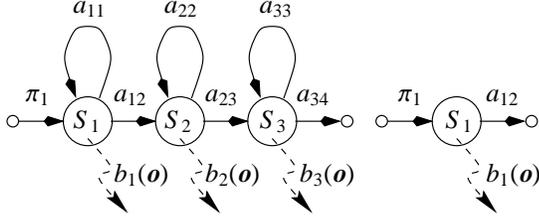
Figure 2. Substroke HMMs : (Left) pen-down model, (Right) pen-up model.

2.2. Substroke HMMs

We model 25 substrokes of eight directions as shown in Figure 1; eight long substrokes ('A'–'H'), eight short substrokes ('a'–'h'), eight pen-up movement ('1'–'8') and one pen-up-down movement ('0'). The HMMs of these substrokes have a topology of left-to-right model as shown in Figure 2. The pen-down models have three states representing the changes of substroke velocity, while pen-up models have only one state without self-loop probability. Here, let $\lambda^{(k)} = (A^{(k)}, B^{(k)}, \pi^{(k)})$ be the set of HMM parameters of substroke k, in which

$A^{(k)} = \{a_{ij}^{(k)}\}$: the state-transition probability distributions from state S_i to S_j,

$B^{(k)} = \{b_i^{(k)}(o)\}$: the probability distributions of observations o at state S_i,

$\pi^{(k)} = \{\pi_i^{(k)}\}$: the initial state probability distributions.

The observation probability distribution is represented by an M-mixture of Gaussian distributions given by

$$b_i(o) = \sum_{m=1}^{M} c_{im} \frac{\exp\left(-\frac{1}{2}(o - \mu_{im})^t \Sigma_{im}^{-1}(o - \mu_{im})\right)}{\sqrt{(2\pi)^n |\Sigma_{im}|}}$$

with the mean vector μ_{im}, the covariance matrix Σ_{im} and the weighting coefficient c_{im}.

2.3. Recognition

Based on Bayes decision rule, our system decodes a time sequential feature of handwriting $O = o_1 o_2 \cdots o_T$, ($o_t =$

Figure 3. Hierarchical structured dictionary of six Kanji characters.

(r_t, θ_t)) to a character W^* which gives maximum likelihood among all characters W:

$$W^* = \underset{W}{\operatorname{argmax}} P(W|O) = \underset{W}{\operatorname{argmax}} \frac{P(O|W)P(W)}{P(O)}.$$

Since $P(O)$ is independent of W and $P(W)$ is assumed to be equal in all W, the recognition result is

$$W^* = \underset{W}{\operatorname{argmax}} P(O|W).$$

Here, $P(O|W)$ means the probability that the feature vector sequence O is produced from a concatenated HMM of substroke sequence $W = w_1 w_2 \cdots w_N$,

$$P(O|W) = \sum_{\text{all } q} P(O, q|W)$$

$$P(O, q|W) = \prod_{n=1}^{N} P(o_{T_{n-1}+1} \cdots o_{T_n}, q_{T_{n-1}+1} \cdots q_{T_n}|w_n)$$

$$= \prod_{n=1}^{N} \left(\pi_{q_{T_{n-1}+1}}^{(w_n)} \prod_{t=T_{n-1}+1}^{T_n} a_{q_t, q_{t+1}}^{(w_n)} b_{q_t}^{(w_n)}(o_t) \right)$$

where $q = q_1 q_2 \cdots q_T$ is a state sequence that outputs O, and a state-transition from substroke w_n to substroke w_{n+1} is occurred at time T_n.

2.4. Hierarchical structured Kanji Dictionary

Hierarchical structured dictionary is useful to define stroke-order rules common to many characters systematically [2]. In Figure 3, six Kanji characters ("語", "言", "吾", "五", "口", "一") can be defined by nine substroke models ('A', 'F', 'a', 'g', 'h', '2', '3', '5', '6'). The rule "語 = 言 2 吾"means that Kanji "語" is defined by combining "言" and "吾" with pen-up model '2', where Kanji "言" and Kanji "吾"are treated as Kanji parts of "語".

3. Generation of Kanji Dictionary for Various Stroke-orders

To decode Kanji handwritings with various stroke-orders, we take an approach of defining multiple stroke-

| r1 | r2 | r3 | r4 | r5 | r6 |

(a) correct stroke order

(b) input pattern
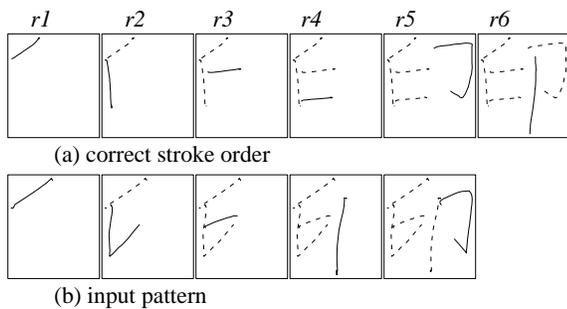
Figure 4. Correct stroke-order and sample pattern of different stroke-order of "印".

印 = 匚 2 卩
匚 = ´ 0 | (s) 3 =
卩 = ⌐ 4 |
= = ‾ 6 ‾

| (s) = f
| (s) = g
‾ = a
| = G
⌐ = a g d

(a) original rules in hierarchical structured dictionary

卩 = ⌐ 3 ⌐   (r_6 - r_5)
= = ‾ 4 ‾   (r_4 - r_3)

(b) additional rules for different stroke-order

印 = f 0 g 3 a 6 a 2 a g d 4 G
| f 0 g 3 a 6 a 2 G 3 a g d
| f 0 g 3 a 4 a 2 a g d 4 G
| f 0 g 3 a 4 a 2 G 3 a g d

(c) expanded substroke sequences

Figure 5. Hierarchical structured dictionary and expanded sequences of "印".

orders in the dictionary, which we call "multiple stroke-order Kanji dictionary (MS dictionary)" in contrast with the conventional "single stroke-order Kanji dictionary (SS dictionary)". In following procedures, the MS dictionary can be generated automatically by using hand-made hierarchical SS dictionary and a large number of samples written by free stroke-order.

In order to show details of generation process statistically, JAIST IIPL (Japan Advanced Institute of Science and Technology, Intelligence Information Processing Laboratory) handwriting database is used here. Among the database, the γ_1 set is written by free stroke-order and the γ_2 set is written with correct stroke-order and correct stroke-number. Both of the datasets contains only 1,016 educational Japanese Kanji characters and those were collected from more than 60 writers. Using these 1,016 Kanji characters, we estimate stroke-order variations of 2,965 JIS 1st level Kanji characters.

Procedure 1: Searching Stroke-orders

Using handwritings of γ_1 set, we can collect various actual stroke-orders. At first, character-dependent stroke HMMs (not substroke HMMs) for each character are trained by using γ_2 set. Only in this stroke-order search, a normalized absolute coordinate (N_x, N_y) is used for a feature vector in addition to the velocity vector (r, θ) described in Sec. 2.1. Then, the stroke-order of handwritings in γ_1 set is determined as an optimal sequence of character-dependent stroke HMM, which gives maximum likelihood among all permutations of strokes. However, the character with stroke-number N produces $N!$ stroke-order variations and it is not practical to search all of them. For example, Kanji "議" consists of 20 strokes, which is the maximum stroke-number in educational Kanji characters. Therefore, we expand stroke-order hypotheses time-synchronously and prune them to the limited number of different stroke-orders by employing beam search method.

Figure 4 shows an example of Kanji "印" whose stroke-number is $N = 6$. Figure 4 (a) shows a correct stroke-order and those strokes $r_n(n = 1, \cdots, 6)$ have been trained

as character-dependent stroke HMMs by using amount of handwritten character "印". Figure 4 (b) shows example of different stroke-order and sequence r_1 - r_2 - r_4 - r_3 - r_6 - r_5 will give maximum likelihood. As shown in this example, our stroke HMM based method does not depend on the stroke-number.

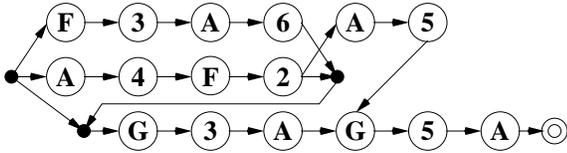**Procedure 2: Extracting Stroke-order Rules**

By referring to the hierarchical structured SS dictionary, we can define stroke sequence production rules for different stroke-orders. In case of Kanji "印", the SS dictionary defines it as shown in Figure 5 (a) and additional stroke-order rules are extracted as shown in (b) on a basis of minimum stroke-number criterion. This criterion makes the total number of different stroke-orders of all Kanji characters the maximum. For example, the rule "卩" affect to 18 characters ("印", "卯", "卸", "却", "脚", "禦", "卿", "仰", "櫛", "迎", "御", "昂", "節", "即", "叩", "命", "柳", "抑") in JIS 1st level Kanji. From JAIST IIPL γ_1 set, new 2,061 rules were extracted. Strictly, a direction of pen-up movement should be changed in combining different stroke-order rules. However, in this paper, we use same pen-up directions with original rules in order to simplify the MS dictionary.

Procedure 3: Adding to the Hierarchical Structured Dictionary

Based on an observation frequency of each rule in each Kanji part, we give a priority to new rules for adding to the dictionary. Table 1 shows top five rules. For example, Kanji "卩" is used as part of three Kanji characters ("命", "印", "節") in γ_1 set and 198 samples are obtained in total. Among them, 78.8 % (156 samples) are written with correct

Table 1. Top five Kanji parts of incorrect stroke-order in JAIST IIPL γ_1 set.

part	correct order	incorrect order	rate	example
凵	｜ 4 ㇄	㇄ 3 ｜	0.941	収
厂	ノ 2 一	一 5 ノ	0.891	成,感,蔵
坐	从 4 ｜ 4 二	从 6 · 4 ｜ 5 一	0.868	座,挫
ナ	ノ 3 一	一 4 ノ	0.859	右,布,有
爿	ノ 2 ＝ 3 ｜	ノ 2 ｜ 4 ＝	0.750	斉,済,剤

Figure 6. Example of a substroke network for three Kanji characters: "右 (incorrect stroke-order) (A4F2G3AG5A)", "右 (F3A6G3AG5A)", "左 (A4F2A5G5A)", "口 (G3AG5A)".

stroke-order and the other 21.2 % (42 samples) are reversed stroke-order, which is the 67th highest frequent stroke-order in additional 2,061 rules.

Procedure 4: Expanding the Dictionary to Network

The substroke network used in the handwriting recognizer is expanded from hierarchical structured dictionary. Then, different stroke-order rules are assigned to different state transition paths and various stroke-orders are produced by combinations of stroke-order rules. For example of "印", there are four combinations as shown in Figure 5 (c).

Our system decodes all characters and those stroke-order variations by using single DAWG (Directed Acyclic Word-Graph) as shown in Figure 6, where common prefixes and suffixes of substroke sequences are merged to same path. In this example, two stroke-order variations of "右", which are occurred in 1st stroke ('F') and 2nd stroke ('A'), are assigned to different paths. Also, "右 (incorrect stroke-order)" and "左" share four nodes of prefix and three nodes of suffix. Table 2 shows relationship between the number of additional stroke-order rules, the total number of substroke definitions for 2,965 characters and the number of nodes required for each network. A linear-structured network does not merge prefixes and suffixes, and the number of its nodes is increased in proportion to the number of definitions. In a tree-structured network, which merges only prefixes, the number of nodes is cut down by the half. On the other hand, in the DAGW structured network, a node reduction rate becomes higher as the total number of definitions becomes large.

Table 2. Relationship between the number of additional stroke-order rules and the number of network nodes.

# addition- al rules	# defini- tions	# network nodes		
		Linear	Tree	DAWG
0	2,965	71,014	38,227	13,264
50	3,532	86,841	45,745	14,172
100	4,175	104,433	53,747	15,182
200	6,417	168,208	80,958	17,109
300	9,035	246,778	109,917	19,141
400	13,153	375,504	157,324	20,922
500	16,093	472,659	185,359	22,555
1,000	17,930	530,944	214,451	30,477
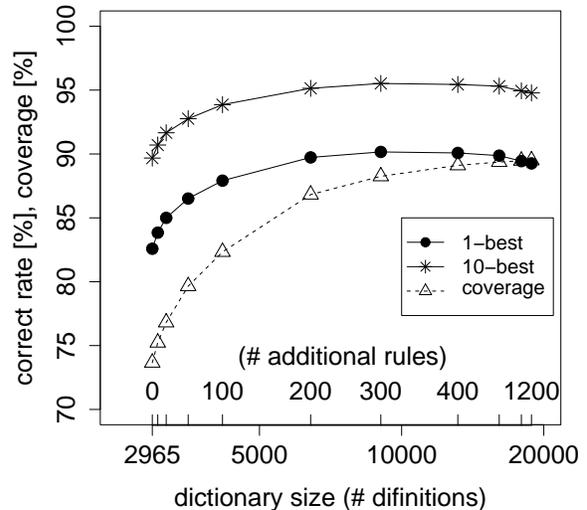
Figure 7. Correct rate and stroke-order coverage by varying dictionary size.

4. Experiments

4.1 Evaluation of MS Dictionary

Handwriting database used in this evaluation is the JAIST IIPL database. Among them, we used three kinds of datasets; γ_1, γ_2, and η sets. As mentioned in Sec. 3, the γ_1 set is used for generating the MS dictionary. To train substroke HMMs, total 25,400 samples of 25 writers from the γ_2 set were used. The η set was used for recognition, which consists of 2,965 Kanji characters of JIS 1st level; 174,935 characters were collected from 60 writers with free stroke-order. In order to realize a real time processing, a beam search width is fixed to 3,000, which corresponds to the total number of selected hypotheses at each time.

Figure 7 shows correct recognition rate, 10-best accumulative recognition rate and stroke-order coverage in the

Table 3. Correct rate for open / closed handwriting datasets.

	improvement of % correct	
	open writers	closed writers
closed characters	82.8 → 90.0	84.7 → 93.4
open characters	81.8 → 89.5	85.0 → 92.5

Table 4. Comparison of correct rate with stroke-order free recognition.

Writer ID	SS dic.	MS dic.	Free (10-best)
0226	94.9	96.7	48.9 (80.5)
0241	65.5	78.0	23.3 (56.9)

dictionary by varying the number of additional rules from 0 to 1,200. The number of definitions in the conventional SS dictionary is 2,965 and correct recognition rate is 82.6%. In case of adding 300 rules, where the number of definitions becomes 9,035, stroke-order coverage increases to 88.2% and correct recognition rate becomes the maximum to 90.2%. Also, the 10-best accumulative recognition rate is improved from 89.7% to 95.5%.

The evaluation dataset (η set) contains closed 1,016 characters within 2,965 JIS 1st level Kanji characters and closed 9 writers within 60 writers, i.e., same characters and writers in training dataset (γ_1) used for generating the MS dictionary. Table 3 shows correct recognition rates according to character sets and writer sets in the case of adding top 300 rules. From the table, we can find that every evaluation set was improved more than 7% and conclude that common stroke-order rules for many characters and writers were effectively added.

4.2 Comparison with Completely Stroke-Order Free Recognition

As a comparative experiment, we performed completely stroke-order free handwriting recognition that does not use prior knowledge of stroke-order statistic. To realize a real time processing, we produced all possible stroke-orders time-synchronously and pruned low-scored hypotheses with 88,950 ($30 \times 2,965$ characters) beam-width. The other experimental conditions were same with previous experiments, except only two writers were used; ID-0226 has written 93.4% of his handwritings with correct stroke-order and ID-0241 has written only 46.1%. As shown in Table 4, correct recognition rate of stroke-order free search is extremely lower than that of using our generated MS dictionary since a large number of ambiguous substroke sequences, which are similar to the other characters, has been expanded.

5. Conclusion

We have proposed an automatic method of generating multiple stroke-order Kanji dictionary and its substroke network for stroke-order free handwriting recognition. In the future, to achieve higher recognition rate with smaller dictionary size, we will investigate a superior criterion than the minimum stroke-number criterion for selection of stroke-order rules.

References

[1] H. Itoh and M. Nakagawa. "An On-line Handwritten Character Recognition Method based on Hidden Markov Model" (in Japanese). *Technical report of IEICE*, PRMU97-85:95–100, July 1997.

[2] C. Komota, M. Nakagawa, and N. Takahashi. "Grammatical Representation of Hierarchical Structure of Kanji Patterns and Its Advantage for On-Line Recognition of Simplified, Distorted and Wrong Stroke Order Patterns" (in Japanese). *IEICE Trans. (D)*, J70-D(4):777–784, Apr. 1987.

[3] A. Kundu and P. Bahl. "Recognition of Handwritten Script: A Hidden Markov Model Based Approach". *Proc. ICASSP '88*, 2:928–931, Apr. 1988.

[4] S. Masaki, M. Kobayashi, O. Miyamoto, Y. Nakagawa, and T. Matsumoto. "Automatic Registration of Templates with Different Stroke Orders for On-Line Character Recognition RAV" (in Japanese). *Technical report of IEICE*, PRMU96-210:135–142, Mar. 1997.

[5] R. Nag, K. H. Wong, and F. Fallside. "Script Recognition Using Hidden Markov Models". *Proc. ICASSP '86*, 3:2071–2074, Apr. 1986.

[6] M. Nakai, N. Akira, H. Shimodaira, and S. Sagayama. "Substroke Approach to HMM-based On-line Kanji Handwriting Recognition". *Proc. ICDAR'01*, pages 491–495, Sept. 2001.

[7] M. Nakai, T. Sudo, H. Shimodaira, and S. Sagayama. "Pen Pressure Features for Writer-Independent On-line Handwriting Recognition Based on Substroke HMM". *Proc. ICPR 2002*, 3:220–223, Aug. 2002.

[8] J.-P. Shin and H. Sakoe. "Stroke Correspondence Search Method for Stroke-Order and Stroke-Number Free On-Line Character Recognition — Multilayer Cube Search —" (in Japanese). *IEICE Trans. (D-II)*, J82-D-II(2):230–239, Feb. 1999.

[9] K. Takahashi, H. Yasuda, and T. Matsumoto. "On-line Handwritten Character Recognition Using Hidden Markov Model" (in Japanese). *Technical report of IEICE*, PRMU96-211:143–150, Mar. 1997.

[10] T. Wakahara, A. Suzuki, N. Nakajima, S. Miyahara, and K. Odaka. "Stroke-Number and Stroke-Order Free on-Line Kanji Character Recognition as One-to-One Stroke Correspondence Problem". *IEICE Trans.*, E79-D(5):529–534, May 1996.