Online Handwritten Japanese Character String Recognition Incorporating Geometric Context

Xiang-Dong Zhou, Jin-Lun Yu, Cheng-Lin Liu
National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences
PO Box 2728, Beijing 100080, P.R. China
{xdzhou, jlyu, liucl}@nlpr.ia.ac.cn

Takeshi Nagasaki, Katsumi Marukawa
Central Research Laboratory, Hitachi Ltd.
1-280 Higashi-koigakubo, Kokubunji-shi, Tokyo 185-8601, Japan
{takeshi.nagasaki.sk, katsumi.marukawa.xb}@hitachi.com

Abstract

This paper describes an online handwritten Japanese character string recognition system integrating scores of geometric context, character recognition, and linguistic context. We give a string evaluation criterion for better integrating the multiple scores while overcoming the effect of string length variability. For measuring geometric context, we propose a statistical method for modeling both single-character and between-character plausibility. Our experimental results on TUAT HANDS databases show that the geometric context improves the character segmentation accuracy remarkably.

1. Introduction

Handwritten documents consist of character strings rather than isolated characters as elementary units because the characters are not apparently separated. Due to the variability of character size and spacing, segmenting characters reliably prior to classification is infeasible. So, character string recognition is generally accomplished by an integrated segmentation and recognition approach [1][2]. To improve the recognition accuracy, the scores of character recognition, geometric context and linguistic context should be integrated for segmentation path evaluation. This is often evaluated by an approximate joint probability function of the candidate segmentation and its string class [3][4]. Since the joint probability is biased to short strings, it tends to yield over-merging errors.

The geometric context (the compatibility of character size, position and between-character relationship, etc., with respect to the string layout) can help disambiguate the uncertainty in character segmentation, but this information has not been explored sufficiently. For English word recognition, Xue et al. [5] compute for a character pattern the distance between its geometric features and the class expectation. Gader et al. [6] train a neural network to measure the probabilities of a number of super-classes on a pair of neighboring characters. Koga, et al. measure the score of some geometric features (called peripheral feature therein) heuristically [7], but the lack of a principled framework limits the robustness of system. Fukushima et al. [3] and Nakagawa et al. [4] incorporate the likelihood of geometric features into the path scores, but only simple features (character size, inter-character and between-character gap) are used.

In this paper, we present a path evaluation criterion incorporating the scores of character recognition, geometric context and linguistic context into a united framework, which can overcome the effect of string length variability. For measuring geometric context, we propose a statistical method to evaluate both the attributes of single-character patterns (unary geometric features) and the relationships between neighboring characters (binary geometric features).

The performance of our character string recognition system was evaluated in experiments on the TUAT HANDS databases [8]. We evaluated the effectiveness of the proposed path evaluation criterion and the effect of geometric context in terms of segmentation accuracy with and without linguistic context. The results show that both unary and binary geometric features are crucial for improving the segmentation accuracy.

2. Recognition System Overview

The block diagram of our online handwritten Japanese character string recognition system is shown in Fig. 1. In pre-processing stage, the input string pattern trajectory is smoothed. By merging the strokes that heavily overlap horizontally, the pre-segmentation module over-segments the string pattern into a sequence of primitive segments. Consecutive segments are combined to generate candidate character patterns, which are represented in a segmentation candidate lattice (Fig. 2). The segmentation paths are evaluated by the classification scores of their constituent patterns, combined with the scores of geometric context and linguistic context. The optimal path, in the sense of maximum score (or minimum cost), gives the segmentation-recognition result.

With the proposed path evaluation criterion, as will be detailed in Section 3, the often used dynamic programming technique does not guarantee finding the optimal path. We instead use beam search [2] to find the path of approximately maximum score in moderate computation time. In beam search, the partial search paths ending at an intermediate search node are sorted and only a limited number (three at maximum in our experiments) of partial paths with maximum partial scores are retained for extension.

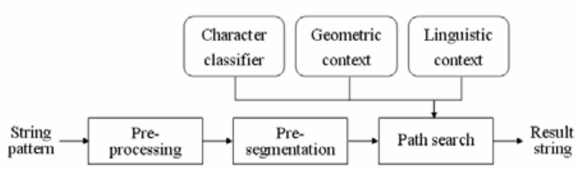

Figure 1. Diagram of online handwritten Japanese character string recognition system.

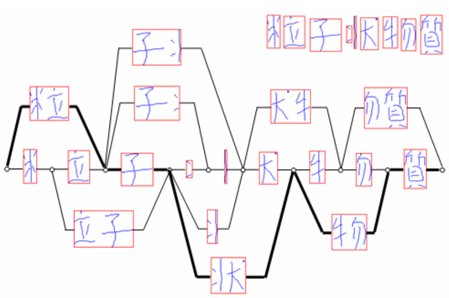

Figure 2. Segmentation candidate lattice of a character string. Each node represents a separation point and each edge represents a candidate pattern.

3. Path Evaluation Criterion

A path in the segmentation candidate lattice consists of a sequence of candidate character patterns. By character classification, each pattern is associated with several candidate classes with the corresponding scores. The procedure of string recognition can be separated into two steps: first, to find the optimal string label for each segmentation path under certain optimization criterion, and second, to find the optimal segmentation path (associated with its optimal string label) under another criterion. The implementation of the above two steps separately is computationally expensive, however. When the two criteria meet with some conditions, the two steps can be unified in a single search procedure.

The string recognition process is detailed below.

- **Step 1**: given an segmentation path composed of a candidate character sequence $O = o_1, \ldots, o_{T_O}$, choose an optimal string label $Q_O^* = q_1^*, \ldots, q_{T_O}^*$ associated with O, i.e.

$$Q_O^* = \arg\max_{Q_O} f_O(Q_O), \tag{1}$$

where $f_O(Q_O)$ is the criterion function for the string label Q_O of O.

- **Step 2**: among all the paths associated with their respective "optimal" string labels obtained in step 1, find the optimal path with its optimal string label:

$$(Q_O^*, O^*) = \arg\max_{(Q_O^*, O)} g(Q_O^*, O), \tag{2}$$

where $g(Q_O^*, O)$ is the criterion for the segmentation path O associated with its optimal string label Q_O^*.

For Step 1 is a labeling problem for a given path, the most widely used criterion is the maximum a posteriori (MAP) probability, which is equivalent to maximizing the joint probability $P(Q_O, O)$ or log-likelihood $\log P(Q_O, O)$ for a given O:

$$Q_O^* = \arg\max_{Q_O} \log P(Q_O, O), \tag{3}$$

in which

$$\log P(Q_O, O) = \sum_{t=1}^{T_O} \Big(\log P(q_t \mid q_{t-1}) + \log P(o_t^c \mid q_t) + \log P(o_t^{g1} \mid q_t) + \log P(o_t^{g2} \mid q_{t-1}, q_t) \Big), \tag{4}$$

where $P(q_t \mid q_{t-1})$ is the transition probability from character label q_{t-1} to q_t, $P(o_t^c \mid q_t)$, $P(o_t^{g1} \mid q_t)$ and $P(o_t^{g2} \mid q_{t-1}, q_t)$ are the conditional probabilities for character features, unary and binary geometric features, respectively, T_O is the string length of the path O. Reasonably, we assume that o_t^c, o_t^{g1} and o_t^{g2} are conditionally independent with each other.

For Step 2, the joint probability is often used as the criterion (e.g. [3][4]). It was pointed out that the joint probability is improper for comparing different paths (different sequences of patterns), because long sequences tend to have smaller joint probabilities than short ones. Using the joint probability as the criterion in step 2, the resulting "optimal" segmentation path tends to have fewer characters. This will raise the segmentation error of merging multiple characters into one pattern. Normalizing the joint probability or likelihood with respect to the string length turns out to overcome this effect [2][9].

3.1. Proposed Path Evaluation Criterion

To evaluate the plausibility of the string label-pattern sequence pair (Q_O^*, O) (string-sequence pair, in brief) free from the effect of path length variability, we define four plausibility terms for measuring linguistic context, character recognition score, unary and binary geometry, respectively:

$$I_l = \frac{1}{T_O}\left(\sum_{t=1}^{T_O} \log P(q_t \mid q_{t-1})\right), \tag{5}$$

$$I_c = \frac{1}{T_O}\left(\sum_{t=1}^{T_O} \log P(o_t^c \mid q_t)\right), \tag{6}$$

$$I_{g1} = \frac{1}{T_O}\left(\sum_{t=1}^{T_O} \log P(o_t^{g1} \mid q_t)\right), \tag{7}$$

$$I_{g2} = \frac{1}{T_O}\left(\sum_{t=1}^{T_O} \log P(o_t^{g2} \mid q_{t-1}, q_t)\right), \tag{8}$$

where $P(q_1 \mid q_0) = P(q_1)$ in (5) and $P(o_1^{g2} \mid q_0, q_1) = P(o_1^{g2} \mid q_1)$ in (8). The plausibility of the pair (Q_O^*, O) is defined as a weighted sum of the above four terms, i.e.

$$I = \lambda_1 I_l + \lambda_2 I_c + \lambda_3 I_{g1} + \lambda_4 I_{g2}, \tag{9}$$

where $\lambda_1 + \lambda_2 + \lambda_3 + \lambda_4 = 1$. I is used as the criterion in Step 2.

We aim to unify the two optimization steps in a single search process for computational efficiency. In Step 2, if the criterion $g(Q_O^*, O)$ has the following property for a given path O:

$$g(Q_O^*, O) = \max_{Q_O} g(Q_O, O), \tag{10}$$

we have

$$(Q_O^*, O^*) = \arg\max_{(Q_O, O)} g(Q_O, O), \tag{11}$$

and the optimal string-sequence pair can be found in a single search process. If we take $g(Q_o, O) = I$ and

$$f_O(Q_O) = \sum_{t=1}^{T_O}\Big(\lambda_1 \log P(q_t \mid q_{t-1}) + \lambda_2 \log P(o_t^c \mid q_t) + \lambda_3 \log P(o_t^{g1} \mid q_t) + \lambda_4 \log P(o_t^{g2} \mid q_{t-1}, q_t)\Big), \tag{12}$$

Eq. (10) will be satisfied. So, we can use the criterion of Eq. (9) to evaluate all the string-sequence pairs in path search.

We use a weighted average of log-likelihood measures to replace the probabilistic joint likelihood of (4). This is because the probabilities cannot be estimated accurately in practice, and instead, approximate likelihood (or log-likelihood) measures are often used for classification.

The above model is a generalization and formalization of the normalized path score of [9], which was originally used to compare the word scores of different lengths, and of that in [2], which was applied to numeral string recognition.

To specialize the terms in (5)-(9), $P(q_t \mid q_{t-1})$ is the transition probability from class q_{t-1} to q_t, which represents the linguistic context in bi-gram. $P(o_t^c \mid q_t)$ is the likelihood of pattern o_t^c with respect to class q_t. If Gaussian densities are assumed for the defined classes, the output of the modified quadratic discriminant function (MQDF) classifier [10], which is employed as the character classifier in our system, is proportional to $-\log P(o_t^c \mid q_t)$ [2].

$P(o_t^{g1} \mid q_t)$ and $P(o_t^{g2} \mid q_{t-1}, q_t)$ measure the unary and binary geometric context, respectively.

3.2. Geometric Context Modeling

$P(o_t^{g1} \mid q_t)$ is the class-conditional probability of the unary geometric feature vector o_t^{g1} for a given class q_t. To estimate $P(o_t^{g1} \mid q_t)$, another MQDF classifier is trained on the unary geometric features.

In our system, we extract nine unary geometric features: the normalized height, width, the sum of inner gaps, the square root of bounding box area and the diagonal length of bounding box of the candidate pattern with respect to the average height of the string, the normalized vertical center, upper and lower bound relative to the vertical center of the string, and the logarithm of the aspect ratio of the candidate pattern.

$P(o_t^{g2} \mid q_{t-1}, q_t)$ is the class-conditional probability of the binary geometric feature vector o_t^{g2} between two successive candidate character patterns. For large character set string recognition, it is almost impossible to get sufficient training samples covering every class pair. In our work, the character classes are clustered into six super-classes by grouping the mean vectors of the unary geometric features (including normalized width, height and vertical position only) of all character classes on a training set of string characters using the k-means algorithm. A pair of successive characters thus belong to one of 36 binary super-classes. The training string character samples, re-labeled to six unary super-classes, are used to estimate the Gaussian probability density functions of 36 binary super-classes. After the above processing, the binary geometric probability $P(o_t^{g2} \mid q_{t-1}, q_t)$ is substituted by $P(o_t^{g2} \mid \tilde{q}_{t-1}, \tilde{q}_t)$, where $\tilde{q}_{t-1}$ and $\tilde{q}_t$ are the unary super-classes of q_{t-1} and q_t.

The binary geometric feature vector consists of four features: the normalized gap width, the distance between the upper bounds, lower bounds and center lines of two successive candidate patterns with respect to the average height of the string.

4. Experimental Results

We evaluated the performance of our method in experiments on character string patterns generated from the online Japanese characters in the TUAT HANDS databases [8].

We selected the JIS level-1 Kanji characters and the 380 symbols (3,345 categories in total) for classifier training. The character classifier (MQDF) is trained on the Nakayosi database and tested on the Kuchibue database with a correct recognition rate 90.42%. Each character pattern, with the modified centroid-boundary alignment (MCBA) method for nonlinear normalization and a normalization-cooperated method for 8-direction feature extraction, is represented by a 512-dimensional feature vector [11]. The feature vector is reduced 160D by Fisher linear discriminant analysis for accelerating classification.

Two quadratic discriminant function (QDF) classifiers are trained on the unary and binary geometric features, respectively, from character patterns (3,345 categories) and pairs of characters generated from the Nakayosi database.

The bi-gram probability table used in our system is estimated from the text corpus of the Japanese Mainichi Newspaper, and the probabilities for the words that do not appear in the table are set to a small constant 10^{-5}.

The test string patterns are true strings extracted from the Kuchibue database, with the between-character gaps narrowed down. The details of the test string patterns are listed in Table 1 and some examples are shown in Fig. 3.

Table 1. Specification of test string patterns.

String number	Total character number	Total class number	Average character number
72,110	1,217,217	1,537	16.88

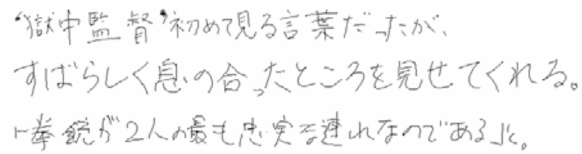

Figure 3. Examples of test string patterns.

The performance is evaluated in term of he measures mentioned in [4]: the character recognition rate (*crr*), recall (*R*), precession (*P*) and *F* measure, which are defined as

$$crr = \text{number of correctly recognized characters} \; / \; \text{number of total characters} \quad (13)$$

$$R = \text{number of correctly detected segmentation positions} \; / \; \text{number of true segmentation positions} \quad (14)$$

$$P = \text{number of correctly detected segmentation positions} \; / \; \text{number of detected segmentation positions} \quad (15)$$

$$F = \frac{2}{1/R + 1/P}. \quad (16)$$

Table 2. Performance without linguistic context.

Weights				Measures (%)			
λ_1	λ_2	λ_3	λ_4	*crr*	*R*	*P*	*F*
0	1	0	0	59.68	97.65	68.30	80.38
0	0.2	0.8	0	66.87	97.55	75.72	85.26
0	0.1	0.15	0.75	80.33	97.82	87.65	92.46

Table 3. Performance with linguistic context.

Weights				Measures (%)			
λ_1	λ_2	λ_3	λ_4	*crr*	*R*	*P*	*F*
0.95	0.05	0	0	77.78	97.46	79.27	87.43
0.67	0.03	0.3	0	89.09	97.56	91.93	94.66
0.58	0.02	0.1	0.3	92.69	97.64	96.64	97.14

Table 2 and Table 3 show the effect of geometric context on string segmentation-recognition performance without and with linguistic context, respectively. The four weights corresponding to linguistic context, character recognition score, unary geometry and binary geometry, respectively, were empirically selected for maximizing the recognition performance. In either Table 2 or Table 3, the first row of rates gives the results without geometry, the second row gives the results with unary geometry, and the last row gives the results with both unary and binary geometry.

We can see that in the case either with or without linguistic context, the incorporation of geometric context improves the character recognition and segmentation accuracies remarkably. The binary geometry further significantly improves the accuracy based on unary geometry. Specifically, in recognition with linguistic context, the segmentation accuracy is improved from 87.43% to 94.66% by unary geometry, and further to 97.14% by binary geometry.

5. Conclusion

To improve the performance of online handwritten Japanese character string recognition, we proposed a path evaluation criterion which combines multiple scores and can overcome the path length variability. For modeling the geometric context, we presented a statistical method to evaluate both unary and binary geometric features. We implemented experiments on string patterns extracted from the TUAT HANDS databases. The results show that the integration of geometric context can improve the performance of string segmentation-recognition remarkably, and the system gives an overall high performance.

Acknowledgements

This work was supported by the Central Research Laboratory of Hitachi Ltd., Tokyo, Japan. The authors thank the Nakagawa Laboratory of Tokyo University of Agriculture and Technology (TUAT) for providing the online handwriting databases.

References

[1] H. Fujisawa, Y. Nakano, and K. Kurino, “Segmentation Methods for Character Recognition: From Segmentation to Document Structure Analysis,” *Proc. IEEE*, vol. 80, no. 7, pp. 1079-1092, 1992.

[2] C.-L. Liu, H. Sako, and H. Fujisawa, “Effects of Classifier Structures and Training Regimes on Integrated Segmentation and Recognition of Handwritten Numeral Strings,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 26, no. 11, pp. 1395-1407, Nov. 2004.

[3] T. Fukushima and M. Nakagawa, “On-line Writing-box-free Recognition of Handwritten Japanese Text Considering Character Size Variations,” *Proc. 15th Int’l Conf. Pattern Recognition*, Barcelona, Spain, Vol.2, pp.359-363, 2000.

[4] M. Nakagawa, B. Zhu, and M. Onuma, “A Model of On-line Handwritten Japanese Text Recognition Free from Line Direction and Writing Format Constraints,” *IEICE Trans. Inf. & Syst.*, vol. E88–D, no.8, pp.1815–1822, Aug. 2005.

[5] H. Xue and V. Govindaraju, “Incorporating Contextual Character Geometry in Word Recognition,” *Proc. Eighth Int’l Workshop on Frontiers in Handwriting Recognition*, Ontario, Canada, pp.123-127, 2002.

[6] P. D. Gader, M. Mohamed, and J.-H. Chiang, "Handwritten Word Recognition with Character and Inter-Character Neural Networks," *IEEE Trans. System Man Cybernetics, Part B*, vol. 27, no. 1, 158-164, Feb. 1997.

[7] M. Koga, T. Kagehiro, H. Sako and H. Fujisawa, “Segmentation of Japanese Handwritten Characters Using Peripheral Feature Analysis,” *Proc. 14th Int’l Conf. Pattern Recognition*, Brisbane, Australia, Vol.2, pp.1137-1141, 1998.

[8] S. Jaeger and M. Nakagawa, “Two On-line Japanese Character Databases in UNIPEN Format,” *Proc. Sixth Int’l Conf. Document Analysis and Recognition*, Seattle, WA, pp.566-570, 2001.

[9] S. Tulyakov and V. Govindaraju, “Probabilistic Model for Segmentation Based Word Recognition with Lexicon,” *Proc. Sixth Int’l Conf. Document Analysis and Recognition*, Seattle, WA, pp. 164-167, 2001.

[10] F. Kimura, K. Takashina, S. Tsuruoka, and Y. Miyake, “Modified Quadratic Discriminant Functions and the Application to Chinese Character Recognition,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 9, no. 1, pp. 149-153, Jan. 1987.

[11] C.-L. Liu and X.-D. Zhou, “Online Japanese Character Recognition Using Trajectory-Based Normalization and Direction Feature Extraction,” *Proc. 10th Int’l Workshop on Frontiers in Handwriting Recognition*, La Baule, France, pp. 217-222, 2006.