On-line Handwritten Japanese Text Recognition free from Constrains on Line Direction and Character Orientation

Masaki Nakagawa and Motoki Onuma

Graduate School of Technology, Tokyo University of Agriculture and Technology,
2-24-16 Naka-cho, Koganei-shi, Tokyo, 184-8588 Japan
motoki@hands.ei.tuat.ac.jp

Abstract

This paper describes an on-line handwritten Japanese text recognition method that is liberated from constraints on writing direction (line direction) and character orientation. This method estimates the line direction and character orientation using the time sequence information of pen-tip coordinates and employs writing-box-free recognition with context processing combined. The method can cope with a mixture of vertical, horizontal and skewed lines with arbitrary character orientations. It is expected useful for tablet PC's, interactive electronic whiteboards and so on.

1. Introduction

Demand to remove remaining writing constraints from on-line handwriting recognition is getting larger since people write more freely on enlarged surfaces of PDA, electronic whiteboards, tablet PC by Microsoft and on new paper-based on-line handwriting environments such as the Anoto pen and paper [1], e-pen [2] and so on [3].

Not only a single character recognition with a writing box or frame imposed for each character, handwritten text recognition without writing boxes or frames (writing-box-free recognition) has been also needed, and our previous research [4] was employed into products. Further demand is conceived to recognize handwriting made on large surfaces by people. We mix horizontal lines and vertical lines and even write text slantingly.

Most of the previous publications and systems have been assuming only horizontal lines of text [4, 5, 6] while we have been trying to relinquish any writing constraint from on-line text input. We proposed a method to recognize mixtures of horizontal, vertical and slanted lines of text with assuming normal character orientation [7], but left handwriting recognition with characters are also rotated like handwritings often made on whiteboards.

In this paper, while improving the above-mentioned method and solving remaining problems, we present an enhanced method to recognize on-line handwriting of arbitrary line directions and character orientations as well as their mixtures. The method incorporates a single-character-recognizer without any modification.

Section 2 of this paper defines line direction and character orientation. Section 3 presents the detailed flow of processing. Section 4 presents examples of recognition and some results of preliminary evaluation. Section 5 concludes this paper.

2. Line direction and character orientation

Here, we define some terminologies. A stroke means a series of pen-tip coordinates sampled from pen down to pen up. Character orientation is used to specify the direction of a character from its top to bottom while line direction is used to designate the writing direction of a sequence of characters until it changes (Fig. 1). Although the line direction is the same as common sense, the character orientation might be the opposite from it. We define them in this way since they are consistent with pen-tip movement directions to write Japanese characters.

A text line is a piece of text separated by new-line and large space and it is further divided into text line elements at the changing points of writing direction. Each text line element has its line direction (Fig. 2). The Line direction and the character orientation are independent.
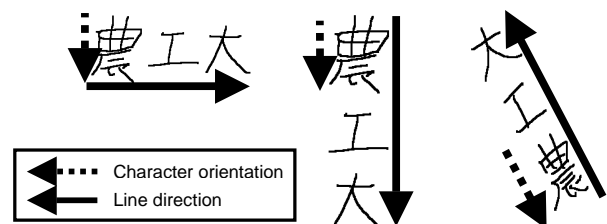

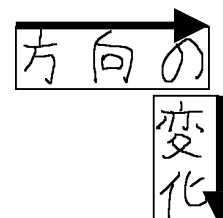**Fig. 1. Line direction and Character orientation.**

Fig. 2. Text line element and Line direction

3. Structure of recognition process

Recognition of handwritten Japanese text liberated from constraints on line direction and character orientation is composed of the steps shown in Fig. 3. The following sections describe them in more detail.
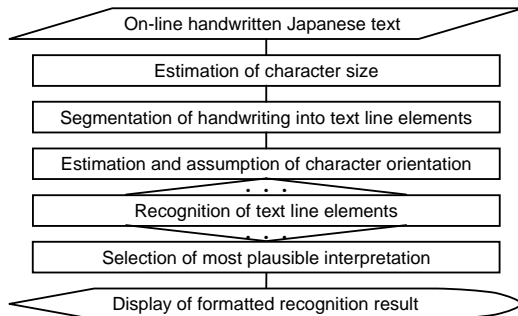

Fig. 3. Flow of processing.

3.1. Estimation of character size

This step estimates the average character size from all the strokes written on a tablet. We assume that most of Japanese characters have the square shape so that the length of one side represents the character size. This size is used to segment handwriting into text line elements, to segment a text line element into characters, to recognize characters and so on.

For each stroke, we take its bounding box and measure its longer side. We sort them, abandon the smaller half and take the average of the remaining larger half. We remove the former since they are short strokes appearing among longer strokes and make the character size estimation too small. This is simple, but produces pretty reliable estimation on the character size. Fig. 4 depicts the method.
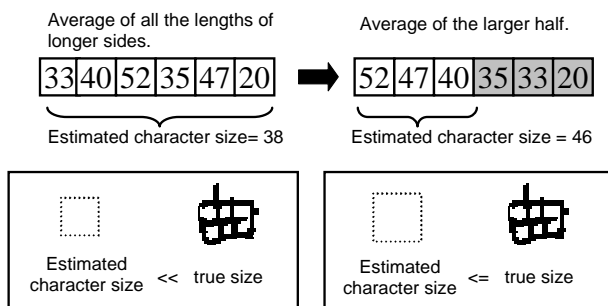

Fig. 4. Estimation of character size.

3.2. Segmentation into text line elements

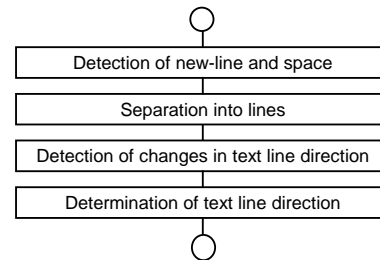This step is composed of sub-steps as shown in Fig. 5. Description on each follows.

Fig. 5. Flow of segmentation into text line elements.

(1) Detection of new-line and space

Pen movement between consecutive strokes is represented by a vector from the ending point of the preceding stroke to the starting point of the succeeding stroke and often called an off-stroke or a dark stroke. Off-strokes within a text line are short while those between text lines are considerably long.

Compared with the estimated character size, if there exist some off-strokes that are much longer than the size, we apply clustering to all the off-stokes and divide them into the two groups as shown in Fig. 6.
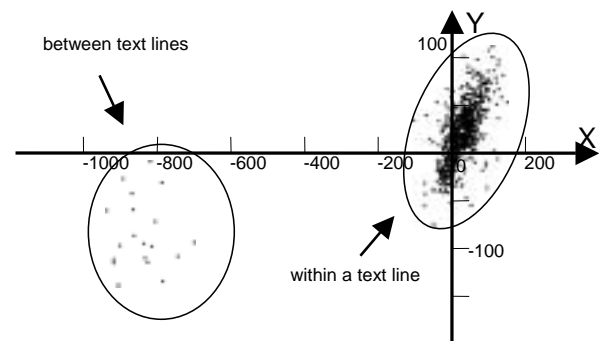

Fig. 6. Clustering of off-strokes.

(2) Detection of the change of writing direction

When we write a list of items, we sometimes write a header horizontally and write items vertically as shown in Fig. 7. When we write some text around a figure or a picture, we sometimes write text horizontally and then the remaining vertically. For these cases, we must find changing points of writing direction, segment the text there into text line segments and assign the correct line direction to each text line element. Otherwise, text is recognized assuming wrong line direction.

In order to detect changing points of writing direction, we employ a recursive procedure similar to that to detect corner points [9]. Among a series of pen-tip coordinates forming a handwritten text line, it finds the most distant point (MDF) from the line connection the starting point and ending point of the series of coordinates, and if the distance is larger than the threshold then apply the same procedure to the line from the starting point to the MDF and that from the MDF to the ending point with detecting

multiple points of directional change as shown in Fig. 8. Thus, a text line segmented by new-line or large space is further segmented into text line elements having different writing directions.

Here, it is worth noting that points thus detected may not be the best segmentation points, i.e., ending or starting points of character patterns but might be within a character pattern. The subsequent steps treat them indecisively.
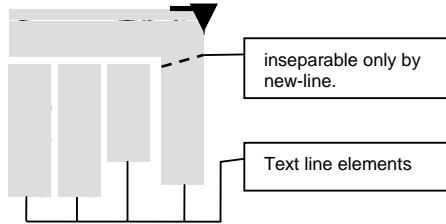

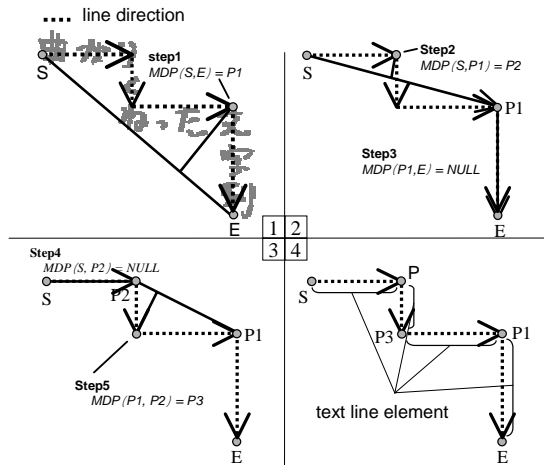**Fig. 7. Mixture of horizontal and vertical writing.**

Fig. 8. Detection of directional changing points.

(3) Estimating the character size per text line element

Since the character size may vary among text line elements, we estimate the character size again for every element. Although the method is the same as described in 3.1, it is applied for each element, so that the character size is more accurately estimated for each.

(4) Calculation of line direction

The direction from the starting point to the ending point of a text line element is assigned as the line direction of the text line element.

3.3. Estimation and assumption of character orientation

This is made by the two steps as shown in Fig.9. It produces multiple hypotheses and the succeeding recognition stages determine the best estimation.
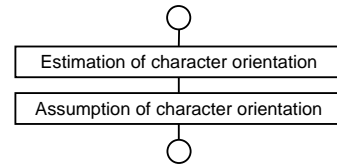

Fig. 9. Flow of processing (3.3)

(1) Estimation of character orientation

When Japanese characters are written, principal pen movement within real strokes is the same as the character orientation or $\pi/2$ left to it. This is because Japanese characters, especially Kanji characters, are composed of downward and rightward strokes. Because of this, if we take the histogram of displacement direction of pen-tip coordinates, we will see two peaks as shown in Fig. 10. These peaks are not so stable if characters are few in a text line element, but they become more stable as the number of characters increase.

Therefore, we can estimate the character orientation from the histogram of displacement direction for a text line element. Once, the character orientation is estimated, the text line element can be recognized by rotating characters until their orientation become downward.

Let us assume the intensity of the histogram at the angle θ as $f(\theta)$. Then, compute $f(\theta) *f(\theta+\pi/2)$ as shown in Fig. 11. This is to find the overlap between $f(\theta)$ and $f(\theta+\pi/2)$. If we can find a single and strong peak, this means the peak at θ and that of $\theta+\pi/2$ are notable and θ is the character orientation. In order to make the peak detection more robust, we convolute the Gauss function $g(\delta) = exp(-\delta 2/\sigma 2)$ so that it works for slanted characters.

If the character orientation is restricted to down or up directions, we can simplify the above and only consider Y-coordinate displacement.
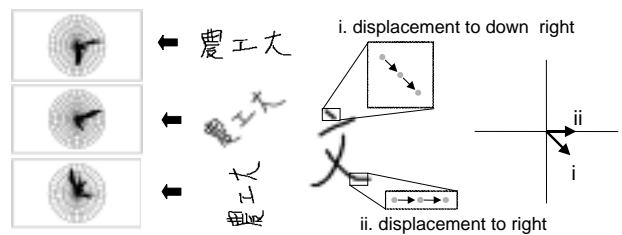

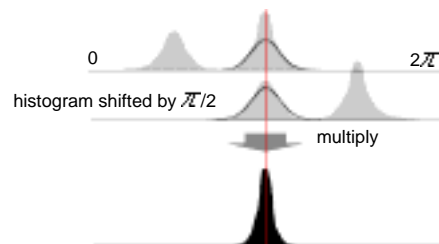**Fig. 10. Two peaks in pen movement direction.**

Fig. 11. Estimation method of character orientation.

(2) Assumption of character orientation

This step assumes character orientation commonly appearing in relation to the line direction of a text line element. Assumed orientations are the same and opposite of the line direction, two perpendicular orientations to it and the four orientations (up, down, right and left) to the orientation of the input tablet as shown in Fig. 12. We can expand the assumption more than these orientations or restrict them when the character orientation has the strong sign or it is confined from the applications.
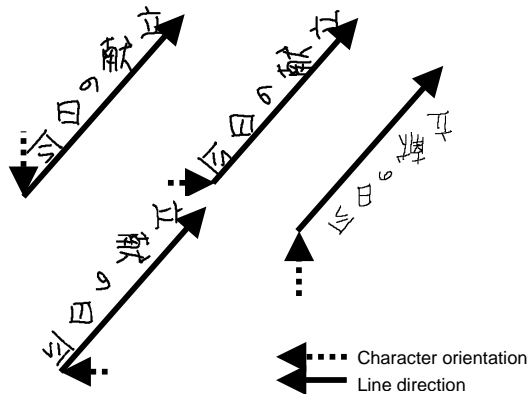

Fig. 12. Direction assumption of a character

3.4. Recognition of text line elements

Assuming each of the character orientations derived above, recognition of a text line element is made for each orientation according to the flow shown in Fig. 13.
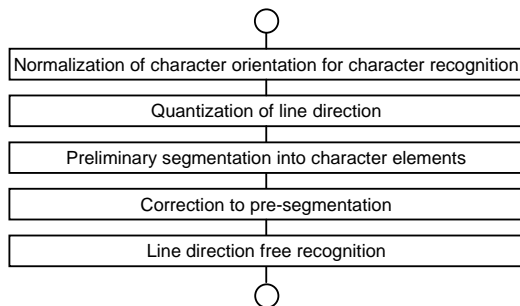

Fig. 13. Flow of recognition for text line elements.

(1) Normalization of character orientation

A text line element is rotated so that the estimated character orientation is downward as shown in Fig. 14. Since it is a sequence of pen-tip coordinates, its rotation to an arbitrary angle is easy and quick. We call the rotated pattern as a character orientation normalized (CO-normalized in short) text line element. For the display of formatted recognition result, the center and amount of rotation is kept for each CO-normalized text line element.
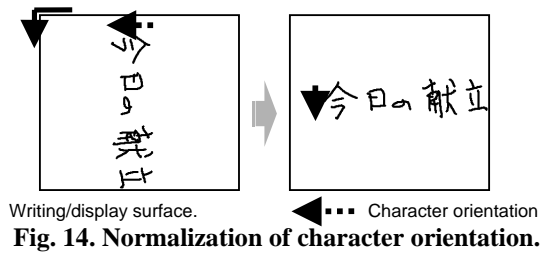

Fig. 14. Normalization of character orientation.

(2) Quantization of line direction

We apply the same amount of rotation to the line direction calculated in 3.2 and call the revised line direction as CO-normalized line direction. This direction is quantized into 4 directions. The Quantization can be finer but 4 quantization is adequate and effective to prevent the CO-normalized text line element from being segmented excessively. For example, if you are writing horizontally to the right direction, then a leftward stroke or off-stroke is used to merge its preceding strokes crossed by the (off-)stroke. In general, if we know the line direction, pen movement of the opposite direction is used to merge its preceding strokes with the result that hypotheses on segmentation can be decreased, which is then effective to speed up the text recognition and to increase the recognition rate.

(3) Preliminary segmentation into character elements

Each text line element is hypothetically segmented into character elements using its estimated character size, the projected distance between the gravity centers of strokes to x-axis when the revised line direction is horizontal or y-axis when vertical, and the overlap between the bounding box of a stroke with that of another stroke. Clear segmentation points are marked "divide", Points that should not be segmented are "combine" and points undetermined are marked "vague" as shown in Fig. 15.
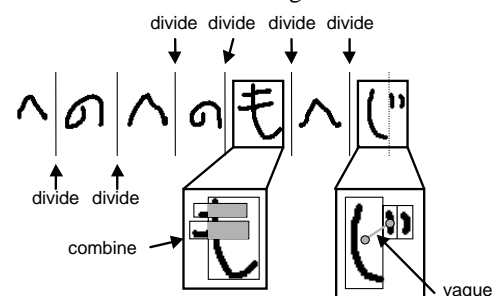

Fig. 15. Segmentation into characters.

(4) Correction to preliminary segmentation

By classifying points judged "vague" into either of "divide" or "combine" as much as possible, the search space of the succeeding text recognition can be reduced. Here, we consider the overlap of the bounding box of a character element and that of another element rather than a stroke. The process is depicted in Fig. 16.
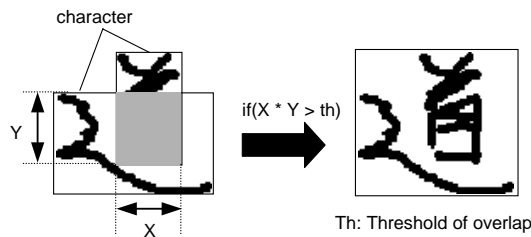
Fig. 16. Correction to preliminary segmentation.

(5) Line direction free recognition

Every CO-normalized text line element has the downward character orientation, so that it is recognized by the line direction free recognizer already developed [7]. This recognizer employs segmentation likelihood, character recognition likelihood, context (bi-gram) likelihood and character size likelihood [4, 5].

3.5. Selection of most plausible interpretation

The recognition of text lines described above is applied for every CO-revised text lines for each text line element. This step selects the highest recognition score and the CO-revised text line that has produced it.

3.6. Display of formatted recognition result

This step displays the recognition result close to the original handwriting. Namely, the recognition result is displayed with fonts are placed according to the original line direction, character orientation, average character size, and average character interval for every text line element.

4. Result

An example of recognition is shown in Fig. 17. The method works for a mixture of text lines that have various line directions and character orientations. We preliminarily evaluated the method using the database HANDS-Kondate_t_bf-2001-11 collected from 100 people. Pages 1 to 11 are just horizontal handwritings and Pages 12 to 22 are just vertical lines so that we added different line directions and character orientations by rotating the original 22 pages x 100 people's handwritings by the amount of 30, 90, 130 and 240 degrees as shown in Table 1.
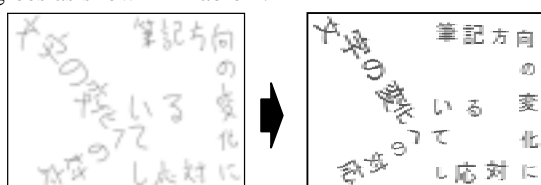

Fig. 17. An example of recognition.

Estimation of character orientation is quantized by 30 degrees that is a tolerance of our character recognizer. We can see that the method is estimating the character orientation and line direction almost correctly.

Table 1. Estimation rates of character orientation and line direction.

performance \ rotation	estimation on character orientation	estimation on line detection
original (0)	99.30	96.79
30 degree	99.33	96.55
90 degree	99.30	97.36
130 degree	98.38	96.61
240 degree	99.64	96.57

5. Conclusion

This paper has presented an on-line handwritten Japanese text recognition method that is liberated not only from writing boxes but also from constraints on line direction and character orientation. This method estimates the line direction and character orientation using the time sequence information of pen-tip coordinates and employs a writing-box-free recognizer with context processing combined. As far as a preliminary test is made, the method is working for a mixture of vertical, horizontal and skewed lines with arbitrary character orientations. We expect that the method would be needed for the large surfaces of tablet PC's, interactive electronic whiteboards and so on.

There remains future research for evaluation of the method using a large database of the target handwritings, improvements of the method based on the evaluation and the user interfaces.

References

[1] http://www.anoto.com.

[2] http://www.e-pen.com.

[3] http://www.otmtech.com.

[4] T. Fukushima and M. Nakagawa: On-line Writing Box Free Recognition of Handwritten Text based on Stochastic Models, (in Japanese) *IEICE Technical Report, PRMU*, Vol.98-139, No.489, pp.25-30 (1998.12).

[5] T. Fukushima and M. Nakagawa: On-line Writing-box-free Recognition of Handwritten Japanese Text Considering Character Size Variations, *Proc. of 15th ICPR*, Barcelona, Vol.2, pp.359-363 (2000.9).

[6] S. Senda and K. Yamada: A Maximum-Likelihood Approach to Segmentation-Based Recognition of Unconstrained Handwriting Text, *Proc. of 6th ICDAR*, Seattle, pp.184-188 (2001.9).

[7] Y. Inamura, T. Fukushima and M. Nakagawa: An On-line Writing-box-free and Writing-direction Free Recognition System for Handwritten Japanese Text, (in Japanese) *IEICE Technical Report, PRMU*, Vol.100-37, No.135, pp.17-24 (2000.6).