Detection and Segmentation of Touching Characters in Mathematical Expressions

Akihiro Nomura*, Kazuyuki Michishita*, Seiichi Uchida**, and Masakazu Suzuki***
* Graduate School of Mathematics,
** Faculty of Information Science and Electrical Engineering,
* * * Faculty of Mathematics,
Kyushu University, 6-10-1 Hakozaki, Higashi-ku, Fukuoka-shi, 812-8581 Japan

Abstract

A technique for the detection and the segmentation of touching characters in mathematical expressions is presented. In the detection stage, a connected component initially recognized into some category is judged as a candidate of touched characters if its feature values deviate from the standard feature values of the category. In the segmentation stage, two component characters of the candidate are decided by the comparison with touching character images synthesized from two single character images. Experimental results showed the effectiveness on the accuracy improvement of the recognition of mathematical expressions.

1 Introduction

Recognition of mathematical expressions is an indispensable technique for applying OCR to scientific documents [3]. For its realization, several procedures particular to the mathematical expressions should be investigated; namely, extraction of the mathematical expressions from the documents, analysis of their two-dimensional structures, and recognition of mathematical symbols.

In addition to these procedures, the segmentation of touching characters is also important to recognize the mathematical expressions. In fact, about a half or more of the misrecognitions in the mathematical expressions are due to touching characters in the OCR developed by the authors. Such misrecognitions will increase when retrodigitizing historical mathematical documents [4, 6]. This is because those documents are often poorly printed on low quality papers and therefore many touching characters are contained in their scanned images.

Conventional segmentation techniques for ordinary text parts, such as projection-based segmentation techniques [1, 5], can not be applied directly to the mathematical expressions. This is because the characters in the mathematical expressions are often arranged two-dimensionally and they might be touched in non-horizontal directions. An alternative way is to use holistic recognition techniques (e.g., [9]) where all possible character combinations are stored and used as references for avoiding explicit segmentation. These techniques, however, also can not be applied to the mathematical expressions, because there are numerous mathematical symbols and their combinations.

In this paper, a technique for the detection and the segmentation of touching characters in the mathematical expressions is proposed. Figure 1 shows the diagram of the proposed technique. In the detection stage, the candidates of touching character images are selected from all connected components by evaluating their initial recognition results. For example, if a connected component of touching characters "f^2" is recognized as "Γ", its several feature values, such as aspect ratio and peripheral feature, are compared to their standard values of the class "Γ". In the case, there will be some difference between their aspect ratios, and thus this connected component "f^2" will be selected as a candidate of touching characters.

In the segmentation stage, each candidate of touching characters is then separated into two individual characters. Here, a recognition-based segmentation scheme is employed for the separation. Specifically, each connected component in the candidates is compared to touching character images synthesized from two single character images. If the connected component is near-perfectly matched to an image synthesized from "f" and "2", the connected component is finally recognized as "f" and "2". For font-style and font-size adaptive segmentation, the single character images (i.e., "f" and "2") are collected from the same document. While this simple segmentation technique is a kind of the sliding window technique [2], the proposed technique is extended so that characters touching in diagonal or other directions can be separated.

The proposed technique can be used for the detection and the segmentation of the normal texts, while this paper focuses on its performance on the mathematical expressions. In addition, while it is assumed in this paper that each touching character image consists of only two component characters, the proposed technique can be easily extended to deal

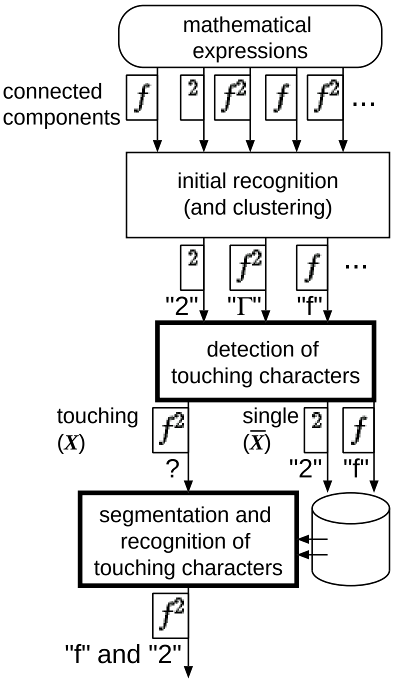


Figure 1. Diagram of the proposed technique.

with the touching character image which consists of three or more characters.

There has been only a few attempts which focus on the detection and the segmentation of the touching characters in the mathematical expressions, in spite of its importance. Okamoto et al. [7] have proposed a technique where the detection is done by the simple thresholding on the recognition score. If a connected component is detected as touching characters, the connected component is firstly blurred and then separated into its component characters along the "valley" (i.e., the continuous local minima) on the intensity surface. This technique has the following two drawbacks. (i) This naive detection scheme will easily fall in the dilemma between over-detection and under-detection. (ii) The performance of this segmentation strategy will be degraded when the component characters are heavily touching or when many valleys are detected around the thin parts of the characters. Lee and Lee [8] have proposed a technique where the segmentation is performed on a one-dimensional sequence of curve segments representing a connected component. Compared to their technique, the proposed technique will be more straightforward and robust because it does not require the delicate process to represent a character image as a sequence of curve segments.

2 Detection of Touching Characters

As shown in Fig. 1, the aim of the proposed technique for the detection of touching characters is to decompose all connected components in a document into two sets $\boldsymbol{X}$ and

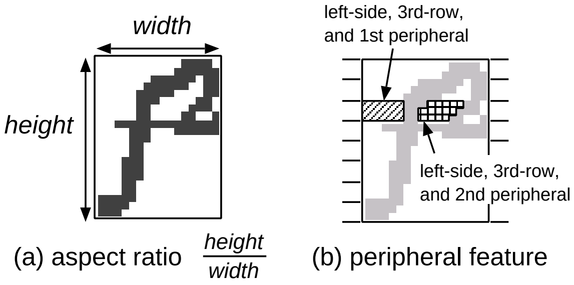


Figure 2. Two features used in the detection of touching characters.

$\overline{\boldsymbol{X}}$. The members of $\boldsymbol{X}$ are the "candidates" of touching characters and those of $\overline{\boldsymbol{X}}$ are the rests. This decomposition is done by a posterior evaluation of the result of an initial recognizer. The posterior evaluation of a connected component x (e.g., initially recognized as "Γ") is composed of following three steps.

1. *Feature extraction*: Several geometrical features are extracted from x.
2. *Comparison*: Those feature values are compared to their standard values (of the category "Γ").
3. *Judgment*: If their difference is larger than a threshold (i.e., permissible difference) α, the connected component x is judged as a touching character image and classified into $\boldsymbol{X}$. Otherwise, x is judged as a single character image and classified into $\overline{\boldsymbol{X}}$.

The features used in the proposed detection technique should be sensitive to the difference between the single character image (e.g., actual "Γ") and the touched character image (e.g., "f^2" recognized as "Γ"). In our experiment, two features, aspect ratio and peripheral feature, were employed (Fig. 2). The peripheral feature is 32-dimensional (= 2 (left/right) sides $\times$ 8 rows $\times$ 2 depths).

3 Segmentation of Touching Characters

The proposed segmentation technique is a kind of recognition-based segmentation techniques. Thus, the recognition results of the touching characters can be obtained simultaneously as well as their segmentation results. The segmentation of a candidate $x \in \boldsymbol{X}$ is composed of the following four steps (Fig. 3).

1. *Search for the first component character*: A single character image $y \in \overline{\boldsymbol{X}}$ is chosen and its thickened image y' is created. Then, y' is matched to one of four corner areas of x. If the matching is successful, that is, all black pixels in the corner area are covered by those of y', y is assumed as the first component character of

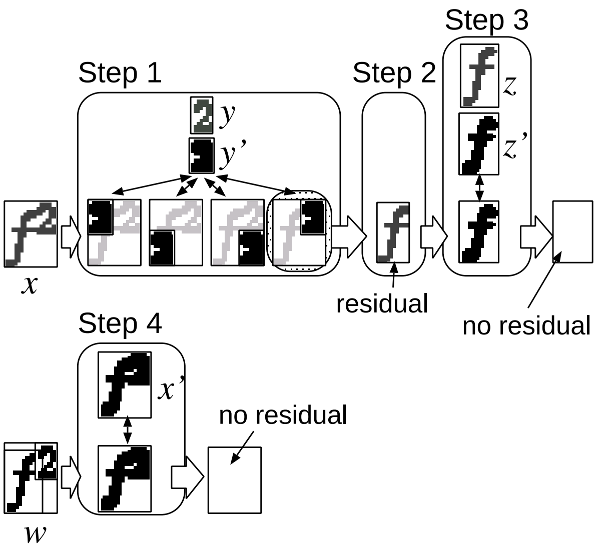


Figure 3. Four steps for the segmentation of touching characters.

x. This process is repeated for other corner areas and other $y \in \overline{\boldsymbol{X}}$ until the matching becomes successful.

2. *Creation of residual image*: A residual image is created by removing the matched corner part from x.

3. *Search for the second component character*: A single character image $z \in \overline{\boldsymbol{X}}$ is chosen and its thickened image z' is created. Then z' is matched to the residual image of x. If the matching is successful, z is assumed as the second component character of x. This process is repeated for other $z \in \overline{\boldsymbol{X}}$ until the matching becomes successful.

4. *Verification*: A touching character image w is synthesized from two single character images y and z detected by Step 1 and 3, respectively. Then the thickened image of x, denoted as x', is matched to w. If the matching is successful, i.e., all the black pixels of w are covered by those of x', the connected component x is finally judged as a touching character image composed of y and z.

In Step 1, four corners are examined in order to separate mathematical characters touching in a horizontal, vertical, or diagonal direction. Since at least one component character of any touching character images lies in one of the four corner, any touching character can be separated into two component characters by Step 1.

Single character images improperly contained in the candidates of touching characters provided by the detection technique of Section 2 can be rejected as single character images in the above segmentation technique. This is because for such images the residual image provided by Step

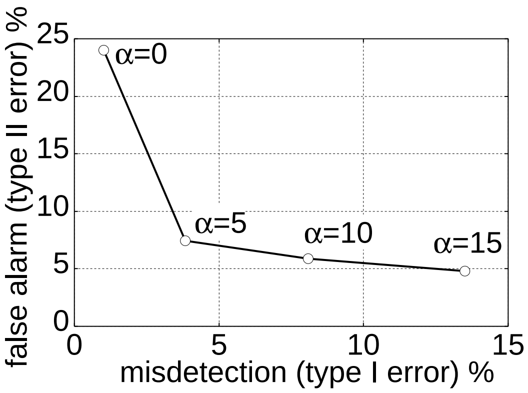


Figure 4. Accuracy of the proposed detection technique.

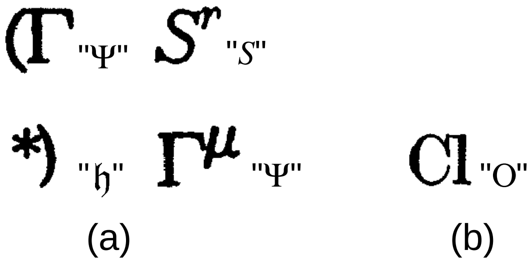


Figure 5. (a) Touching character images successfully selected as the candidates of touching characters and (b) a touching character image misdetected. (Their initial recognition results are also shown.)

2 will become an empty image or the matching at Step 1 and 3 will not be successful with any $y, z \in \overline{\boldsymbol{X}}$.

The thickening operations in Step 1 and 3 are necessary to eliminate trivial differences at the matching. The verification of Step 4 is necessary to detect the false segmentations due to *inclusion*. For example, the touching characters "$O($" might be separated into "Θ" and "(" without Step 4 because the all black pixels of "Θ" will be covered by those of "O".

In this technique, the single character images (i.e., $y, z \in \overline{\boldsymbol{X}}$) are not fixed; they are collected from the subjected document automatically by the preceding detection stage. This approach possesses the merit that the segmentation can be done adaptive to the font-style and the font-size of the document. This approach, however, also possesses the drawback that the proposed technique fails if $\overline{\boldsymbol{X}}$ does not contains the two component characters of x.

4 Complexity Reduction by Clustering

For computational efficiency, both of the detection and the segmentation techniques are applied to several *representatives* (centroids) instead of all connected components.

For example, even though there are many component characters initially recognized as "Γ", only a few representatives of them are examined in the proposed technique.

The selection of these representatives is done by a clustering technique based on a sequential appending and splitting procedure. Each connected component is appended to the cluster of its nearest representative one after the other. If the variance of the cluster exceeds some threshold, the cluster is split into two independent clusters. This simple clustering technique is far faster than conventional iterative clustering techniques, such as k-means method.

The connected components belonging to the same cluster are to have the same destiny in the detection and the segmentation stages; if their representative is decided as a pair of touching characters, they are also recognized as the same touching characters. Thus, the threshold in the above clustering procedure should be set to a small value (i.e., enough clusters should be prepared) so that true "Γ" and false "Γ"(="f^2") do not belong to the same cluster.

5 Experimental Results

5.1 Database

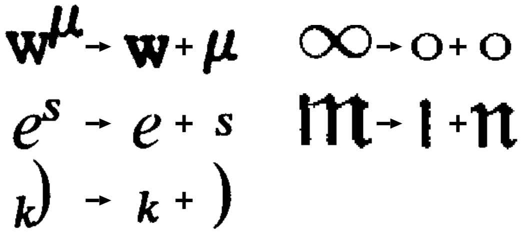
The detection and the segmentation experiments were performed on 21 scanned mathematical documents from 11 journals. The number of total pages was 391. Characters (connected components) in their mathematical expressions were subjected to the experiment and the other characters, i.e., characters contained in normal text parts were not. The number of the subjected connected components was about 140,000. Among them, the number of touching characters was 2978 (about 2% of all). For each character, its ground truth (category and the flag of touching or single) was attached manually.

5.2 Initial Recognition and Clustering

Each connected component of the mathematical documents was initially recognized as a single character by a recognizer. As this result, the connected component which consists of two (or more) touching characters was forcedly recognized as a single character [1]. The recognition rate by the initial recognizer was 92.9% (9710 misrecognitions). Since almost all 2978 touching character images were to be misrecognized, about 60% (=(2978×2)/9710) of misrecognitions by the initial recognizer were due to touching characters.

The clustering technique of Section 4 was then applied to the connected components. In our experiment, 140,000 connected components were clustered into 13,291 clusters.

[1] A single character originally comprised several connected components (such as "i" and "j") is specially treated as a single connected component as much as possible in the recognizer.



(a) (b)

Figure 6. (a) Touching character images successfully separated and (b) single character images unnecessarily separated.

Among these clusters, 909 clusters (about 7% of all clusters) were of touching characters.

The computations required by the detection and the segmentation techniques were reduced to about 1/10 by the clustering. In fact, the average computation times for the detection and the segmentation stages for one page were 1.34 s and 1.00 s, respectively, on a PC (Pentium III, 1GHz).

The clustering was performed within each document. Accordingly, touching character images of a document were to be separated by single character images from the same document.

5.3 Detection Performance

Figure 4 shows the accuracy of the proposed detection technique plotted by changing the threshold α. The X-axis is so-called type I error (misdetection) and the Y-axis is so-called type II error (false alarm). About 96% (=4% type I error) of the connected components of touching characters were successfully selected as the candidates at $\alpha = 5$. As shown in Table 1, the 12866 candidates consist of 2864 actually touching character images and 10002 single character images at $\alpha = 5$. Namely, many type II errors were occurred. These type II errors, however, were not serious because most of them can be successfully rejected as single character images as we shall see in Section 5.4. Figure 5 shows the examples of touching characters successfully detected and misdetected by the proposed technique.

5.4 Segmentation Performance

Table 1 shows the result of the segmentation of the 12866 (=2864+10002) candidates. About 51% (=1468/2864) of actual touching characters were success-fully separated into their component characters. Although this rate itself is not very high, the rate means that about a half of misrecognitions due to touching characters can be reduced by the proposed technique. As this result, the recognition rate of the OCR can be significantly improved by the proposed technique (as we shall see Section 5.5).

The three major reasons of the 1396 failures on the separation of actually touching characters were as follows: (i)

Table 1. The result of the segmentation of touching characters. Non-parenthesized numbers are the numbers for connected components (i.e., characters) and parenthesized numbers are the numbers for clusters. The term "success" means the successful separation for touching character images and the successful rejection for single character images.

	actual number (ground truth)		candidates of touching char.		segmentation success		segmentation failure	
touching	2978	(909)	2864	(859)	1468	(409)	1396	(450)
single	134375	(12382)	10002	(1334)	9984	(1326)	18	(8)

The single character images (y and z) which compose a touching character image (x) were not contained in $\overline{\boldsymbol{X}}$. (ii) There was a trivial but non-negligible shape difference between single character and touching character. (iii) A touching character image was separated into two incorrect characters (e.g, "mn"→"nm"). The first reason was the most serious one (40% of all 1396 failures). Its possible remedy is the employment of at least one fixed single character image for each category in addition to the single character images adaptively collected into $\overline{\boldsymbol{X}}$. Another remedy is the use of single character images created by rescaling the single character images in $\overline{\boldsymbol{X}}$. The latter remedy will be more promising than the former. This is because even if the segmentation of touching character image "f^2" is failed due to the lack of a subscript-sized image of "2", its normal-sized version can be often found in $\overline{\boldsymbol{X}}$.

As shown in Table 1, 99.8%(=9984/10002) of the single character images improperly contained in the candidates were successfully rejected as single character images. This result indicates the robustness of the proposed segmentation technique.

Figure 6 (a) shows the successful results of the proposed segmentation technique. From these results, it is shown that characters touching in non-horizontal directions can be separated by the proposed segmentation technique. Figure 6 (b) shows two single character images unnecessarily separated.

5.5 Recognition Performance

In order to ensure the effectiveness of the proposed technique, the improvement on recognition rate by the successful segmentation was measured. As noted in Section 5.2, the recognition rate of the initial recognizer was 92.9%. The rate was improved to 95.1% by the proposed technique. This Improvement is very significant because the number of the misrecognitions were reduced from 9710 to 6792, that is, reduced to 70%.

6 Conclusion

A technique for the detection and the segmentation of touching characters in mathematical expressions was presented. In its detection stage, a connected component is selected as a candidate of touched characters if its several feature values deviate from the standard feature values of its initially recognized category. In the segmentation stage, two component characters of the candidate are decided by the comparison with touching character images synthesized from two single character images. Experimental results showed the effectiveness of the proposed technique on the accuracy improvement of a recognition system for the mathematical expressions.

Acknowledgement: This work was supported in part by The Ministry of Education, Culture, Sports, Science and Technology in Japan under a Grant-in-Aid for Scientific Research No. 14380182.

References

[1] R. G. Casey and E. Lecolinet, "A survey of methods and strategies in character segmentation," *IEEE Trans. Pat. Anal. Mach. Intell.*, 18(7):690–706, 1996.

[2] R. G. Casey and G. Nagy, "Recursive segmentation and classification of composite character patterns," *Proc. 6th Int. Conf. Pat. Recog.*, 2:1023–1026, 1982.

[3] K.-F. Chan and D.-Y. Yeung, "Mathematical expression recognition : a survey," *Int. J. Doc. Anal. Recog.*, 3(1):3–15, 2000.

[4] K. Dennis, G. O. Michler, G. Schneider, and M. Suzuki, "Automatic reference linking in distributed digital libraries," *Proc. Workshop on Document Image Analysis and Retrieval (DIAR-03)*, To appear, 2003.

[5] Y. Lu, "Machine printed character segmentation – an overview," *Pattern Recognition*, 28(1):67–80, 1995.

[6] G. O. Michler, "Report on the retrodigitization project "Archiv der Mathematik"," *Archiv der Mathematik*, 77:116–128, 2001.

[7] M. Okamoto, S. Sakaguchi, and T. Suzuki, "Segmentation of touching characters in formulas," in *Doc.t Anal. Systems: Theory and Practice. Third IAPR Workshop, DAS'98. Selected Papers* (Lecture Note in Computer Science vol.1655), Springer-Verlag, 1999.

[8] H.-J. Lee and M.-C. Lee, "Understanding mathematical expressions in a printed document," *Proc. 2nd Int. Conf. Doc. Anal. Recog.*, 502-505, 1993.

[9] X. Wang, V. Govindaraju, and S. Srihari, "Holistic recognition of handwritten character pairs," *Pattern Recognition*, 33(12):1967–1973, 2000.