Document Layout and Reading Sequence Analysis by Extended Split Detection Method

Noboru Nakajima, Keiji Yamada and Jun Tsukumo

C&C media Research Laboratories, NEC.
4-1-1 Miyazaki, Miyamae-ku, Kawasaki, 216-8555, Japan
{noboru,yamada}@ccm.cl.nec.co.jp

Abstract. This paper describes an Extended Split Detection Method that can hierarchically segment a machine-printed page image with a complex layout into smaller layout elements. The method performs piecewise-linear segmentation using many kinds of separator elements such as field separators, lines, edges of figures, and edges of white background areas. Furthermore, this method represents an analyzed layout of a hierarchical structure in a tree data structure, in which all nodes are traversed according to the simple rules for generating the reading sequence. We demonstrated that the new method increases the correct character line segmentation rate by 15.5%, to 95.5%, and we achieved a correct reading sequence generation of 88.1%.

1 Introduction

We have been producing more and more electronic documents. Some of them are transported as electronic mail or generated by digital publishing tools. Documents in electronic forms can be easily retrieved by text search and can be reused for other documents. They sometimes can be machine-translated into other languages or spoken by computers. There is a great demand to obtain the same benefits from paper documents that can be obtained from electronic ones. To solve this, it is important not only to precisely recognize machine-printed characters but also to analyze the layout structure of a printed document. Accurate understanding of hierarchical document layout can enable the correct extraction of the reading sequence from a paper document and can provide paper documents with the versatile benefits of electronic documents. Many methods have been proposed for analyzing document layout. Kise [2] and Okamoto [8] proposed methods that regard large white background regions as separators and segment images by using them. Jain [3] proposed a method which uses texture features and discriminates character regions from the others. Though most of the proposed methods aim at exact extraction of complicated layout components such as character lines and figures, the reading sequence of the document cannot be reproduced from only a set of layout components, but must be manually edited with great effort by an operator.

S.-W. Lee and Y. Nakano (Eds.): DAS'98, LNCS 1655, pp. 336-347, 1999.

In order to address this problem, it is essential not only to segment a document image into layout components, but also to extract a hierarchical structure of the layout objects [1]. Tsuji [4] proposed a method named the split detection method, which hierarchically segments an image based on pixel projection patterns and reproduces a hierarchical layout structure according to the recursive segmentation of the layout objects. However, this method cannot segment intricate layout objects because it divides layout objects by a straight line in every segmentation.

We propose a method which expands on the split detection method. It uses piecewise-linear segmentation as well as straight-line segmentation and can analyze a complicated layout structure in order to reproduce a hierarchical layout structure which is represented as a layout tree. Reading sequence analysis from the extracted hierarchical layout structure is also described.

2 Split Detection Method

The split detection method segments a document image (from a page to layout objects such as columns, blocks, and character lines) into subregions in the recursive segmentation manner [4,9]. The recursive segmentation process generates a tree that represents the hierarchical layout structure of segmented layout objects.

The Split Detection Method calculates the possibility of a split which divides a region into subregions and determines the directions and locations of splits by using rules based on both periodicity $\overline{\tau}$ and separability measurement η (cf. Fig. 2) [4].

Periodicity $\overline{\tau}$ is defined as the average of distances between neighboring peaks in a projection pattern. Separability measurement η is obtained from the Fisher ratio when a partial projection pattern including two neighboring peaks is regarded as a probability distribution.

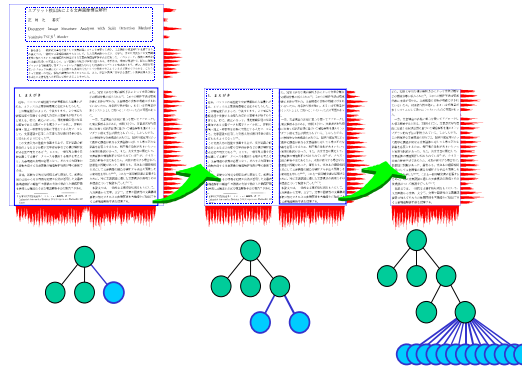


Fig. 1. Layout analysis by hierarchical segmentation.

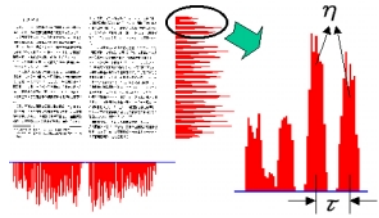
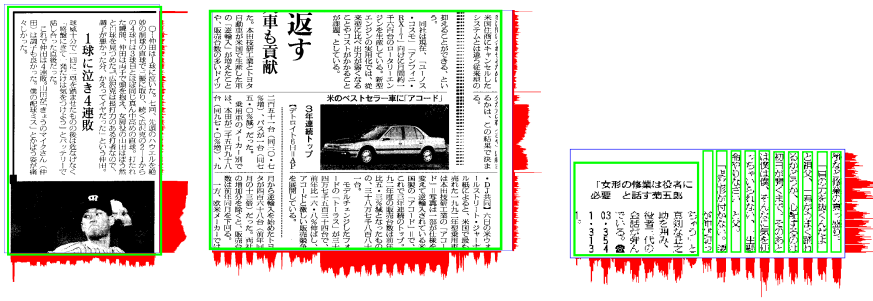


Fig. 2. Segmentation on projection pattern.

The separability measure of a gap between two neighboring peaks in the projection pattern represents the likelihood of the gap being a split in the layout object. As shown in Fig. 2, when periodicity $\overline{\tau}$ is observed in the projection pattern along the vertical axis, the periodicity $\overline{\tau}$ is used to detect split candidates with vertical orientation. This is based on the rule that a white gap between blocks must be wider than the gaps between character lines. In the example in Fig. 2, the gap in the projection pattern along the horizontal axis was detected as a split, because the gap was wider than the periodicity of the projection pattern along the vertical axis.

In order to detect splits, this method estimates character sizes and the widths of gaps between characters, character lines, and blocks. Therefore this method is robust against these variations. Because this method recursively detects splits from a whole document page to subregions, it can easily extract the hierarchical layout structure of the document page. This is accomplished under the condition that the document is arranged according to the ordinary typesetting rules such as, the width of the inter-block gap > the width of the inter-character-line gap > the width of the inter-character gap.



(a) (b) (c)

Fig. 3. Errors in the split detection method. (a) Enclosed text. (b) Complex block arrangement mixture of vertical and horizontal typesetting `j`. (c) Mixture of heterogeneous text (non-rectangular text areas).

But because gaps in projection patterns are used to detect split candidates, when the region is enclosed by field separators as shown in Fig. 3 (a), the gap which corresponds to the correct split is not detected and the correct split is therefore missed. Furthermore, where heterogeneous layout objects are as intricate as those shown in

Fig. 3 (b) and Fig. 3 (c), incorrect segmentation results are obtained because periodicity or separability measurement can not be correctly estimated from the projection patterns.

3 Extended Split Detection Method

3.1 Concept and Outline of the Proposed Method

The proposed method uses several kinds of separator elements as well as splits, which are used in the split detection method. They are field separators, edges of layout objects, large white background regions, and so on. The conventional split detection method has only to select the widest split from the segmenting region recursively in order to extract the hierarchical layout structure from a document image. However, when many separator elements are used for the segmentation of a document, they generate many possible segmented subregions in terms of combinations of separator elements. In order to extract the correct hierarchical layout structure from the document image, we have to select the most important separator elements from a segmenting region and segment this region into subregions. However, the kind of separator elements that should be selected depends on the kinds of segmenting regions and segmented subregions. For example, we should use the periodicity of character lines in order to segment a text block into character lines. However, we don't use the periodicity of the blocks but use field separators and large white regions as keys to segregate a title and text blocks from a document page.

To solve these requirements, we propose the new document layout analysis method described below (cf. Fig. 4.). The details of the processes used in the method are described after section 3.2.

Step 1 Separator element detection
Objects are detected in this step, such as (1) field separators, (2) large white background regions, (3) figures, tables, and photograph regions, and (4) the borders of the segmenting regions. All of them as well as splits which are extracted by using the split detection method are called separator elements.

Step 2 Subregion candidate generation
Suppose that a region in a document is segmented into subregions. Initially, a region is the whole document image. The process generates subregion candidates enclosed by a set of separator elements.

Step 3 Segmentation rule selection
The subregion candidates are classified into columns, blocks, character lines, etc., according to the feature values which are extracted from them. The class of the region in the segmentation is already known because it was obtained during the previous segmentation. Then the segmentation rules and their parameter values are selected depending on the class of the region and the classes of subregion candidates. Furthermore, the conventional split detection method is applied to subregion candidates and generated splits are stored as a kind of separator element.

Step 4 Separator element verification

This process compares feature values extracted from subregion candidates and separator elements that have parameter values with the selected segmentation rules in order to verify the subregions and separator elements. If separator elements are accepted, the subregions divided by the separator elements are registered under that node in the tree of the hierarchical layout structure, which indicates the region in the segmentation. If the registered subregions are not character lines, each of the subregions is regarded as a region for segmentation and processes in ***Step* 2, 3, and 4** are recursively executed.

Step 5 Reading sequence analysis

All over the hierarchical layout structure, nodes which have the same parent node are sorted based on their positions and character line directions included in the regions corresponding to the individual nodes.

We describe the process in each step in detail below.

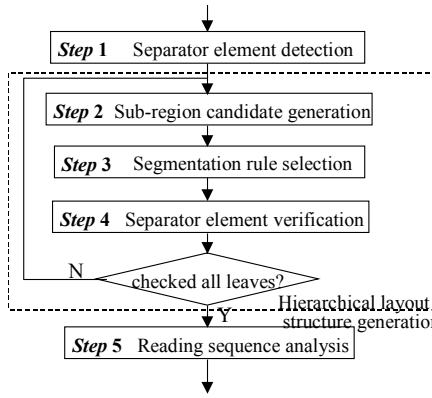


Fig. 4. Processing flow of Extended Split Detection Method.

3.2 Separator Element Detection *(Step 1)*

Separator elements are extracted in the split and merge manner [5-7] from an input image. First, connected components are extracted from the reduced image with four times lower resolution than the original. As found in the example shown in Fig. 5 (a), some characters are combined together. Almost all pixels in the photograph area belong to a component.

Then, a feature vector is extracted from the part of the original image inside the bounding rectangle area which corresponds to each connected component in the reduced image. It is made up of a black pixel density, an area, a line density, etc. The feature vectors of the connected components are classified into a photograph class, a field separator class, a character class, and so on. Every class has its reference vector and the decision function was designed based on the Mahalanobis distance in the feature domain. Roughly speaking, a photograph class has a large black-pixel-density.

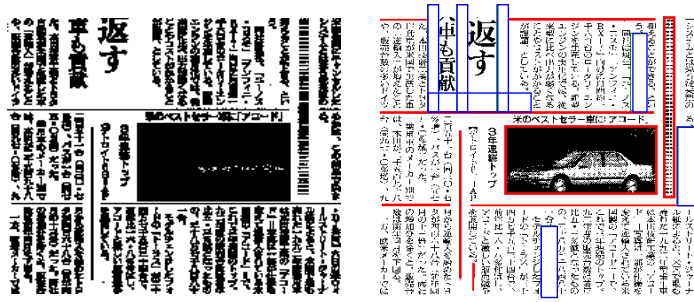
A character class has a small black-pixel-density and a relatively large line density as well as a small width or a small height because its features are extracted from the original image with fine resolution. Here, an initial tree data structure is constructed; the root node which indicates the whole document image and all the connected components are registered with their classes as child nodes of the root node.

Next, we detect separator elements using connected components with feature vectors.

Large white regions are extracted using the same method Okamoto [3] proposed. The white rectangles in Fig. 5 (b) indicate these regions. They become a kind of separator element.

Field separators of straight lines are easily detected as combinations of very thin connected components. Dotted lines and decorated field separators have periodic patterns which include small connected components of equal size. The edges of block areas such as photographs, tables, and figures are also determined as kinds of separator elements.

Examples of detected separator elements are indicated by the bold boxes in Fig. 5 (b).



(a) (b)

Fig. 5. Separator element detection. (a) Low resolution image (reduced image). (b) Separator element detection result.

3.3 Hierarchical Layout Structure Generation *(Steps 2-4)*

After extracting separator elements the process can extract a hierarchical structure from a document image and represent it in a tree data structure. In order to hierarchically divide a document image, we have to adaptively select the best region borders at each level of the hierarchy. The conventional split detection method selects the widest white straight gap as the best split. However, we increased the number of kinds of separator elements and extended straight splits to the piecewise linear borders of regions. Therefore, in order to select the most important separator elements, the process has to change segmentation rules according to both the classes of the segmenting regions and the classes of the segmented subregions. By estimating the classes of regions before and after segmentation, the proposed method can select segmentation rules and their parameter values that adapt to the classes of the regions.

(1) Subregion Candidate Generation *(Step 2)*

Every separator element is extended in its lengthwise direction until it meets other objects. Extended separator elements are traced in the region before segmentation, and partial regions which are enclosed by separator elements are detected as subregion candidates.

(2) Subregion Segmentation Rule Selection *(Step 3)*

Detected subregion candidates are classified into region classes; for example, a character line class, a line segment class, a block class, an undefined region class, a region of blocks class, and a region of an undefined region and blocks class.

First, separator elements are removed from the subregion candidate. Pixel values in the subregion candidate are projected onto the horizontal axis and the vertical axis. Accumulated pixel values are represented in a one-dimensional pattern which is called a projection pattern.

From the obtained projection patterns, those features are extracted for subregion classification. These are the periodicity and intervals of peaks in the projection pattern, white space width, and so on. Classification of the subregions is carried out according to the rule base with extracted features. For example, a subregion should be classified into a block class if it has a periodicity whose pitch is as large as the character size in its projection pattern. Otherwise, it is classified into an undefined region class.

The classes of the subregion candidates are obtained in the above method. And the class of the region in the current segmentation is known because it is a document page class if this segmentation is the first one; otherwise, the region class is obtained during the upper-level segmentation.

(3) Separator Element Verification (*Step* 4)

Region features and separator features are used to verify the segmentation concerning a region and its subregions. Region features are described above as periodicity of each projection pattern and so on. Separator features consist of a length, a width, a direction, and so on.

The region features and the separator features are examined according to the rule selected based on the region classes. For example, if width□ of separator elements are wider than the period of the projection pattern of each subregion, the subregions are designated as blocks and the separator elements are designated as spaces between the blocks.

If the segmentation is determined, the child nodes which indicate the subregions are generated under the parent node which indicates the region before segmentation in the hierarchical layout structure. And the connected components belonging to each subregion are moved to the descendant nodes of the child nodes.

An example of a hierarchical layout structure extraction is shown in Fig. 6.

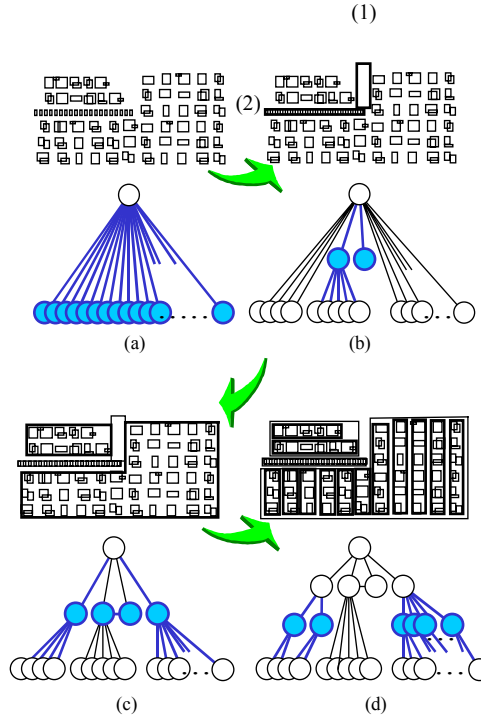


Fig. 6. Process example for the extended split detection method.

Rectangles in the upper part of Fig. 6 (a) show the minimum bounding rectangles of the connected components. The lower part of Fig. 6 (a) is a tree which represents an initial tree of a hierarchical layout structure. Leaf nodes that corresponds to connected components are directly connected to the root node that correspond to a page image.

Then the separator element is detected. The bold line (1) in Fig. 4 (b) indicates a large white region, and (2) indicates a dotted line. Both are extracted as separator elements. At this time, the two separator elements are registered in the tree as nodes. The node of the dotted line has many child nodes which indicate dots. The node of the white region has no child node.

After tracing adjacent separator elements, regions enclosed by the separator elements are extracted as subregion candidates (cf. Fig. 6 (c)). Leaf nodes under the node of the region in the current segmentation are connected to one of the new nodes according to their positions.

The segmentation process enables intricate regions to be correctly segmented. It is applied to each subregion recursively and the hierarchical layout structure is generated from the whole page image (cf. Fig. 6 (d)).

3.4 Reading Sequence Analysis *(Step 5)*

Once a hierarchical layout structure is extracted, the reading sequence analysis is carried out by sorting nodes in the generated layout tree.

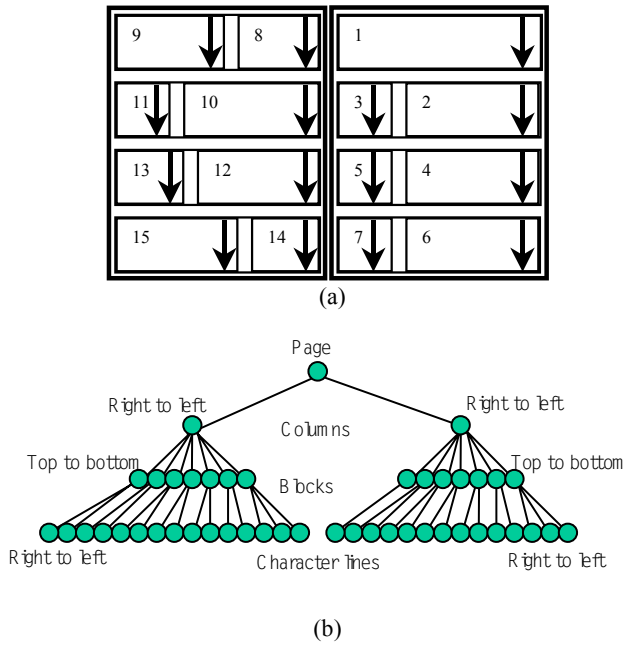


Fig. 7. Reading sequence analysis. (a) Reading sequence (rectangle: block, number: reading sequence, arrow: character line direction). (b) Tree data for hierarchical layout structure.

For the case where child nodes belong to the character line class whose direction is vertical and their parent node is in the block class, character lines are sorted from right to left depending on their position.

For the case where child nodes belong to the character line class whose direction is horizontal and their parent node is in the block class, character lines are sorted in descending order from top to bottom.

When either the parent node is not in the block class or child nodes are not in the character line class, if leaf nodes under the child nodes are in the character line class with vertical orientation then the child nodes are sorted from right to left and then from top to bottom. If the leaf nodes are in the character line class with horizontal orientation, the child nodes are sorted from left to right and from top to bottom. This operation is applied to every set which consists of a parent node and its child nodes and obtains a geometrically adequate reading sequence.

Field separators of straight lines are easily detected as combinations of very narrow connected components. Dotted lines and decorated field separators have periodic patterns which include small connected components of equal size. The edges of block areas such as photographs, tables, and figures are also determined as kinds of separator elements.

Examples of detected separator elements are indicated by the bold boxes in Fig. 5 (b).

4 Experiment

We examined the effectiveness of the proposed method using twenty-four newspaper images. The images were printed in multi-column and complex layouts and they include figures, photographs, and tables with texts.

Tested images included 2770 character lines, of which 2646 were extracted correctly. Therefore, the recall ratio was 95.5%. Furthermore, 15 character lines were extracted from figure regions and noises by mistake. This means that the precision ratio was 99.4%. 68 out of 71 photograph regions (95.8%) were extracted correctly. 2442 character lines (88.1%) were correctly sorted in the reading sequence. Here, the regions such as figures, captions of figures, and boxed articles were excluded from the examination of reading sequence generation because the reading order of these regions was ambiguous.

Examples of the layout analysis results are shown in Fig. 8. In Fig. 8 (a), text regions located at the center and left-bottom position were extracted correctly. The reading sequence was also obtained correctly. In the left-bottom part of Fig. 8 (b), a region of horizontal character lines broke into a region of vertical character lines, but the blocks were segmented correctly.

Over-segmentations were observed in the block where all the horizontal character lines accidentally had white spaces in the same horizontal position. The block was incorrectly segmented into the two parts. There were also under-segmentation errors because of the separator element detection failure.

Figure 9 is an example of a document in which the reading sequence is difficult to analyze. Because a figure and a title broke into the text block, the two character lines have the possibility to follow on the end of the second character line in the right-top block, as shown by the two arrows in Fig. 9. Linguistic information is required to solve this problem.

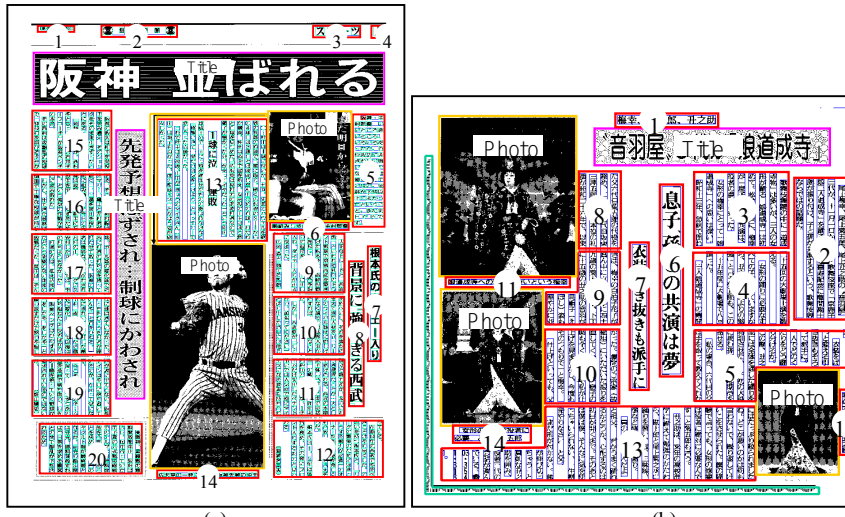


Fig. 8. Layout analysis results of the proposed method (The numbers in the figures indicate the reading sequences.)

Fig. 9. An example of reading sequence analysis error.

5 Conclusion

We proposed a document layout analysis method which can hierarchically segment a document image into regions not only by using straight lines but also by using piecewise-linear borders. This enables intricate regions to be segmented. Furthermore, this method can generate a reading sequence of segmented blocks from the extracted hierarchical layout structure. Experiments using newspaper images demonstrated the accuracy of the proposed method. Without using character recognition, 95.5% of the character lines were correctly extracted which was a 15.5% increase compared to the conventional method. The precision ratio was 99.4%. Without character recognition, 88.1% of character lines were correctly sorted in the reading sequence.

We will improve the recall and precision of the document layout analysis method by using character recognition and linguistic processing.

Acknowledgments

The authors would like to thank their colleagues in the Pattern Analysis technology group of C&C Media Research Labs, NEC Corporation for their helpful discussions. We would also like to thank Mr. Toshiyuki Tanaka of NEC Information Systems Corporation for constructing the experimental programs.

References

1. Y. Y. Tang, S. W. Lee, and C. Y. Suen, "Automatic Document Processing: a Survey", *Pattern Recognition*, Vol. 29, No. 12, pp. 1931-1952, 1996.
2. A. K. Jain and Y. Zhong, "Page Segmentation Using Texture Analysis", *Pattern Recognition*, Vol. 29, No. 5, pp. 743-770, 1997.
3. M. Okamoto and M. Takahashi, "A Hybrid Page Segmentation Method", *Proc. ICDAR*, pp. 743-748, 1993.
4. Y. Tsuji, "Document Image Analysis for Generating Syntactic Structure Description", *Proc. ICPR*, pp. 744-747, 1988.
5. A. K. Jain and Bin Yu, "Page Segmentation Using Document Model", *Proc. ICDAR*, pp. 34-38. 1997.

6. K. Etemad, D. Doeman, and R. Challappa, "Multi-scale Segmentation of Unstructured Document Pages Using Soft Decision Integration", *Pattern Recognition*, Vol. 30, No. 9, pp. 1505-1519, 1997.
7. Y. Ishitani, Document Layout Analysis Based on Emergent Computation, *Proc. ICDAR*, pp. 45-50, 1997.
8. K. Kise, O. Yanagida, and S. Takamatsu, "Page Segmentation Based on Thinning of Background", *Proc. ICPR*, pp. 788-792, 1996.
9. J. Liu, Y. Y. Tang, Q. He, and C. Y. Suen, "Adaptive document segmentation and geometric relation labeling: algorithm and experimental results", *Proc. ICPR*, pp. 763-767, 1996.