PERGAMON

Pattern Recognition 35 (2002) 485–503

PATTERN RECOGNITION
THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY

www.elsevier.com/locate/patcog

Automatic generation of structured hyperdocuments from document images☆

Ji-Yeon Lee[a], Jeong-Seon Park[a], Hyeran Byun[b], Jongsub Moon[c], Seong-Whan Lee[a,*]

[a]*Center for Artificial Vision Research, Department of Computer Science and Engineering, Korea University, Anam-Dong, Seongbuk-ku, Seoul 136-701, South Korea*

[b]*Department of Computer Science, Yonsei University, 134 Shinchon-dong, Seodaemoon-ku, Seoul 120-749, South Korea*

[c]*Department of Electronics and Information Engineering, Korea University, Chochiwon, Yeongi-kun, Chungnam 339-800, South Korea*

Received 13 January 2000; accepted 28 December 2000

Abstract

As sharing documents through the World Wide Web has been recently and constantly increasing, the need for creating hyperdocuments to make them accessible and retrievable via the internet, in formats such as HTML and SGML/XML, has also been rapidly rising. Nevertheless, only a few works have been done on the conversion of paper documents into hyperdocuments. Moreover, most of these studies have concentrated on the direct conversion of single-column document images that include only text and image objects. In this paper, we propose two methods for converting complex multi-column document images into HTML documents, and a method for generating a structured table of contents page based on the logical structure analysis of the document image. Experiments with various kinds of multi-column document images show that, by using the proposed methods, their corresponding HTML documents can be generated in the same visual layout as that of the document images, and their structured table of contents page can be also produced with the hierarchically ordered section titles hyperlinked to the contents.

Keywords: Structured hyperdocument; Multi-column document; Document conversion; Document image understanding; Logical structure analysis

1. Introduction

The growing popularity of the internet has been recently and continually increasing the demand to have documents accessible and retrievable through the World Wide Web, for the purpose of sharing them via the internet. Inevitably, this has given rise to the need for the automatic conversion of paper document images, as well as digital documents, into hyperdocuments.

As for the conversion of electronic documents into hyperdocuments, many methods and commercial tools have been developed and are now being used in real applications. As for the conversion of paper documents, however, only a few research works have been conducted. Furthermore, these research works were primarily concerned with the conversion of single-column document images, and the images have been limited to containing only text and image objects [1–3]. Unfortunately, an automatic conversion of complex and various multi-column document images has not been dealt with, but the necessity of representing such documents in the form of hyperdocuments is continuously increasing.

☆This research was supported by Creative Research Initiatives of the Korean Ministry of Science and Technology. A preliminary version of this paper has been presented at the 15th International Conference on Pattern Recognition, Barcelona, September 2000.

* Corresponding author. Tel.: + 82-2-3290-3197; fax: + 82-2-926-2168.

E-mail address: swlee@image.korea.ac.kr (S.-W. Lee).

0031-3203/01/$20.00
PII: S0031-3203(01)00026-7

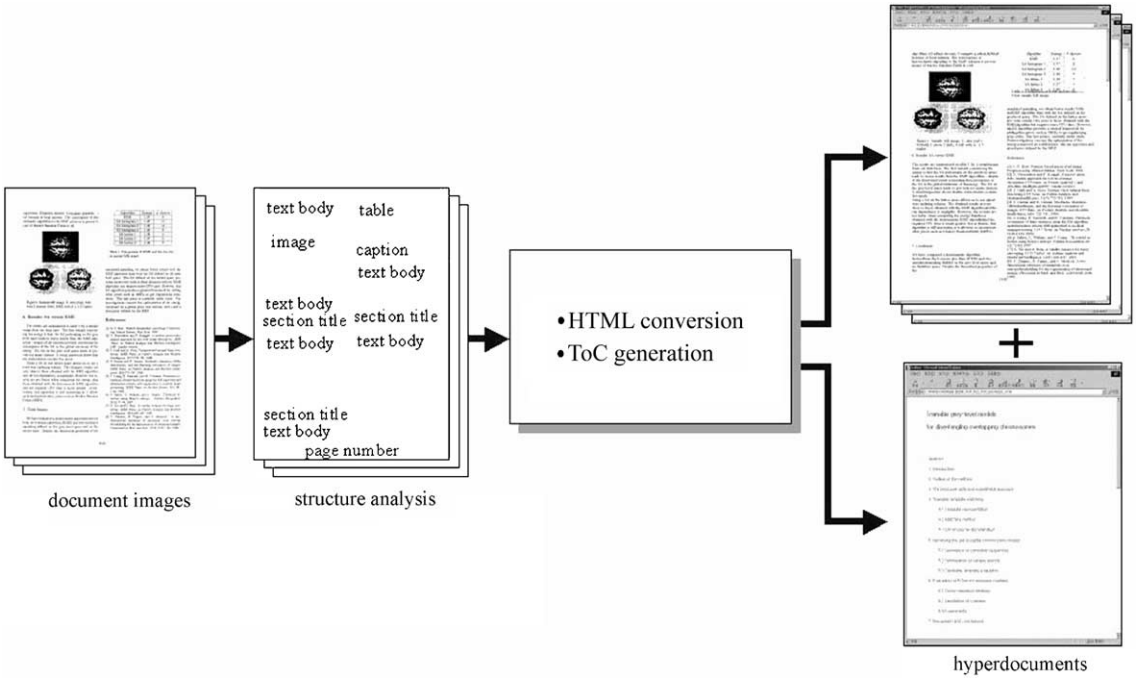


Fig. 1. Overall process of the proposed method.

In this paper, we propose two methods that convert multi-column document images into HTML documents; one is implemented using the table structure and the other using their layer structure. We also suggest a method for generating a table of contents (ToC) page through a logical structure analysis (logical labeling) [4–7]. Fig. 1 illustrates the overall process of the automatic generation of the structured hyperdocuments from multi-column document images.

In the process, geometrical and logical structure analysis are performed on multi-column document images. In the geometrical structure analysis, it classifies all objects in the input document images into image, table, and text object. After character recognition, logical structure analysis provides labels such as section title, caption, page number, header, footer, etc., to the text objects. The proposed methods are then applied to the result of the structure analysis. In converting multi-column document images into hyperdocuments, it is desirable for the screen display of the hyperdocuments to be consistent with the layout of the paper document images, so as to preserve their logical flow and appearance. To do so, we use the table and layer structure for the conversion stage. Finally, for generating a structured table of contents page, only the section titles from text objects are extracted and ordered hierarchically. The generated table of contents page provides the logical flow of the input documents and hyperlinks to the corresponding contents.

Although tags in HTML are much more limited in availability and representation than those of XML or SGML, HTML provides a convenient way for the sharing and retrieving of various and complex document images through the internet, after they are automatically converted into their corresponding structured hyperdocuments in HTML.

This paper is organized as follows: In Section 2, we review the previous works related to the conversion of document images into HTML documents and to the logical structure analysis of document images. In Section 3, we describe two proposed conversion methods of multi-column document images. In Section 4, we describe how to generate a table of contents page by extracting section titles from the text objects. In Section 5, experimental results on various kinds of complex multi-column document images are analyzed to evaluate the performance of the proposed conversion method. Finally, conclusions and further studies are given in Section 6.

2. Related works

In order to convert paper document images into hyperdocuments, document structure analysis (geometrical structure analysis and logical structure analysis) must be performed. Geometrical structure analysis generally classifies homogeneous regions in a document image into

Table 1
A brief comparison of previous works and the proposed method

	Year	Features			
		Table object	Multi-column	ToC generation	Hyperlink
Tanaka et al. [1]	1998	O	X	X	X
Kieninger et al. [2]	1998	O	X	X	X
Worring et al. [3]	1999	X	X	O	O
Faure [4]	1999	X	X	O	X
The proposed method	2000	O	O	O	O

text, image, and table objects. Logical structure analysis labels the objects with respect to their logical senses and establishes the relationship among the objects (classified as texts, images and tables).

In the generation of a structured table of contents page, many approaches to logical structure analysis have utilized the geometrical information (location, size, type) of objects, in order to label them as header, footer, caption, page number, title, and so on [5,7,8].

Lin et al. [6] analyzed a table of contents page for representing the correct logical structure of an entire documents of book. However, it is difficult to analyze the logical structure of documents correctly without a table of contents page. It is also hard to find exact chapters or section titles in case that there exist many misrecognized numbers.

Ishitani [7] defined each line in text objects and then logically labeled them as title, list, formula, paragraph, etc., by segmenting or grouping them.

Faure [4] extracted a list of first-order headings from input document images by detecting and sorting capitalized headings, numbered headings, and key headings (headings containing keywords). This method has demerits in that it depends on the accuracy of character recognition and it considers only first-order headings.

To date, there has been only limited research on the conversion of paper documents into hyperdocuments. Tanaka and Tsuruoka [1] proposed a new method for converting table objects to hyperdocuments. In this method, nodes and corners are defined, and a node property matrix of a real table object is constructed. A node type is decided and the appropriate tags are inserted in the HTML document according to the node property matrix. This method takes much time, since this method tries to match the node with all possible nodes defined in the matrix.

Kieninger and Dengel [2] proposed a table conversion method based on the cell shapes of a table object. This method is very much restricted by the varieties of cell shapes in a table.

Kochi and Saitoh [9] described a method that correctly names objects in a document on the intervention of a user for some fixed labels, and generated a HTML document using this information.

Worring and Smeulders [3] implemented a system that converts a manual consisting only of image and text objects into its corresponding hyperdocuments. This system provides a user friendly interface with figures, texts, a list of figures and a table of contents. It also establishes hyperlinks to proper text in the contents, to a figure or a text in the figure. However, the visual frame of the converted hyperdocument is quite different from that of the original manual.

Table 1 illustrates a brief comparison of previous works and the proposed method on the conversion of document images into hyperdocuments. As seen in Table 1, our system supports all 4 features, while in the conventional methods the transformation is limited to only a few objects.

3. HTML conversion of multi-column document images

We propose two methods to convert multi-column document images into hyperdocuments; one is based on the table structure and the other is based on the layer structure.

3.1. An approach based on the table structure

As a result of the geometrical structure analysis, a document image is divided into several kinds of objects, each of which is classified as text, image, or a table object. It is very easy to represent text and image objects by inserting simple tags in an HTML document without any specialized operations. On the other hand, table objects, having various formats, need some manipulations for conversion.

We propose a new algorithm for converting table objects in a paper document image into table objects in HTML format and applying this to the conversion of

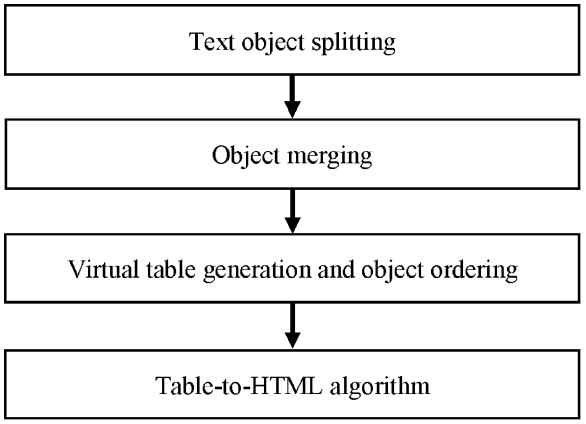


Fig. 2. Conversion procedure based on the table structure.

a multi-column document image into a hyperdocument as it is. In doing so, we split and/or merge, if necessary, the objects of the document image, so as to change them to a format convertible into an HTML document. Fig. 2 shows the overall conversion procedure based on the table structure.

3.1.1. Text object splitting

Among the classified objects, image and table objects are no longer split. However, text objects need to be split. Later, in order to create a table of contents or change a document image into a table structure, when a set of text objects with different meanings has been classified as one text object during the geometrical structure analysis, it is necessary for the text object to be split into individual meaningful objects.

In general, a text object is labeled as one of the following: a section title, text body, caption, header, footer, page number and so on. For our purpose, we consider only section title, text body and page number among them, because they are sufficient for building a structured table of contents.

Initially, we classify the lines in a text object into four types, as follows:

Let *TL* be the current text line and *TO* be a text object that includes *TL*.

Let also *th* be the font size of *TL*, which is used as a threshold here.

(1) If $(|TL_{sx} - TO_{sx}| > th)$ and $(|TL_{ex} - TO_{ex}| < th)$, then *TL* is defined as *Indented Line* (*IL*).
(2) If $(|TL_{sx} - TO_{sx}| < th)$ and $(|TL_{ex} - TO_{ex}| > th)$, then *TL* is defined as *Entered Line* (*EL*).
(3) If $(|TL_{sx} - TO_{sx}| > th)$ and $(|TL_{ex} - TO_{ex}| > th)$, then *TL* is defined as *New Line* (*NeL*).
(4) If $(|TL_{sx} - TO_{sx}| < th)$ and $(|TL_{ex} - TO_{ex}| < th)$, then *TL* is defined as *Normal Line* (*NoL*).

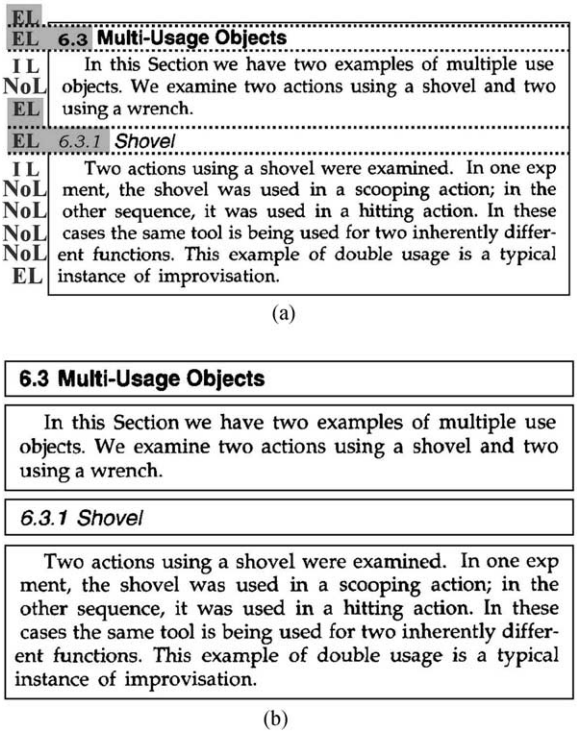


Fig. 3. An example of splitting a text object. (a) A text with line lables. (b) Split text objects.

A text object can be split according to the arrangement of the text lines defined above. The font size used as a threshold is the value obtained after OCR recognition. Initially, the line right above the starting line of the text objects is set to *EL*. As shown in Fig. 3(a), when a text line in a text object is of any type starting with a special pattern and its previous line is of *EL* type, it is split from the text object as another text object regarded as a section title candidate. The objects above and below the section title candidate, then, are moved further apart from it, as shown in Fig. 3(b). The *special pattern* means that the pattern is in the form of [(number or text) + symbol(including space)]. Using this pattern, we can extract the section title candidates without regard to the accuracy of character recognition. This is achieved by considering the starting character of a text line as a text as well as a number by way of provisions against misrecognition of a number as a text.

3.1.2. Object merging

To have the (split) objects fit into a virtual table format, we merge them and modify their coordinates.

Fig. 4 illustrates the merging stage of text objects. First, we divide a document image horizontally into regions where the value of a horizontal projection profile is zero.

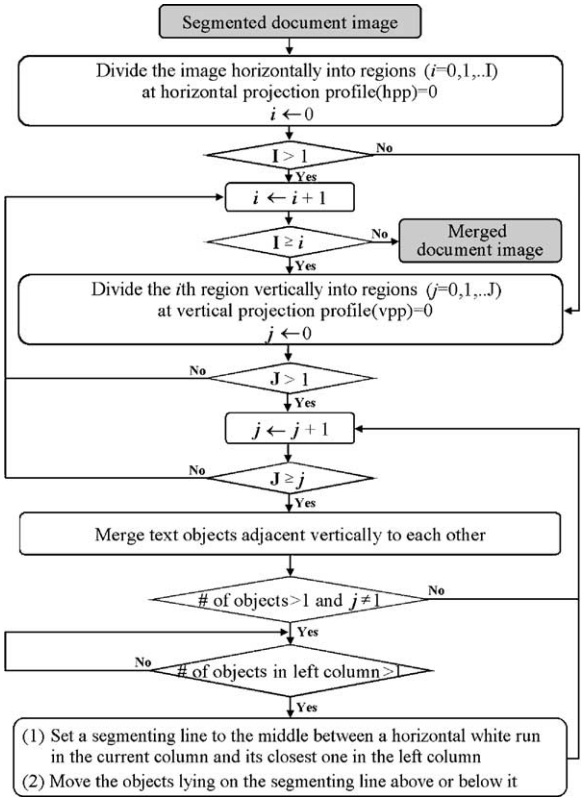


Fig. 4. Flowchart for merging text objects.

Second, for each of the horizontally divided regions, we divide it vertically where the value of a vertical projection profile is zero. Finally, from left to right and from top to bottom, object merging takes place in each region.

When the objects in a column are all text, they are merged into one text object. Otherwise, a virtual segmenting line is drawn in the middle between a horizontal white run in the current column and its closest one in the left column, and the objects lying on the segmenting line are moved above or below it.

3.1.3. Virtual table generation and object ordering

A virtual table is created in the merging stage, with each of the split and/or merged objects regarded as a cell of the table. The objects are arranged in the order that the cells of a table object are created in an HTML document.

The criteria for the arrangement of objects are as follows:

Let O_i and O_j be the ith and jth objects, respectively. Let x_i^{TL} and y_i^{TL} be the top and leftmost x and y coordinates of the O_i.

(1) If y_i^{TL}, y_j^{TL}, O_i and O_j satisfy $|y_i^{TL} - y_j^{TL}| <= th$ and $x_i^{TL} < x_j^{TL}$, $i,j = 1, \ldots, n$, then O_i and O_j exist in the same row and O_i has priority over O_j.

(2) If y_i^{TL} and y_j^{TL} satisfy $|y_i^{TL} - y_j^{TL}| > th$ and $y_i^{TL} < y_j^{TL}$, $i,j = 1, \ldots, n$, then O_i and O_j exist in different rows and O_i has priority over O_j.

In comparison of the top and leftmost y coordinates of two objects, if the absolute difference between the two values is equal to or smaller than the threshold, the two objects exist in the same row. At this time, the object whose leftmost x coordinate is smaller has priority over the other. On the contrary, if the absolute difference between the two values is larger than the threshold, the two objects exist in different rows. At this time, the object whose top y coordinate is smaller has priority over the other. In this way, we can rearrange objects in a table format representable on web browsers.

Fig. 5 shows the result images of each stage of the table structure-based approach.

3.1.4. Table-to-HTML conversion algorithm

In this section, we describe the proposed table-to-HTML conversion algorithm. The algorithm converts not only a table object in a document image into that of its HTML document, but also a multi-column document image into its corresponding HTML document, by considering it as a table object as a whole. Therefore, we can convert table objects and multi-column document images into HTML documents using the same algorithm.

```
For (i=0; i < Row_N; i++) {
    r=i;
    For (j=0; j < Column_N; j++) {
        l=j;
        If (Cell[r][l]_T and Cell[r][l]_L==1) {
          Colspan=Rowspan=1; // initialization
          While (Cell[r][l]_R !=1) { Colspan++; l++; }
          While (Cell[r][l]_B !=1) { Rowspan++; r++; }
        }
    }
}
```

$Cell[i][j]_T$, $Cell[i][j]_B$, $Cell[i][j]_L$ and $Cell[i][j]_R$ are the top, bottom, left and right lines of the cell, located in the ith row and the jth column of a table, respectively. Row_N and $Column_N$ are the number of rows and the number of columns of a table, respectively. While each cell is checked from left to right and from top to bottom, tagging is performed.

If the top and left line values of the current cell are all one (1: real line, 0: virtual line; see Fig. 6), the right and bottom line values are checked. *Colspan* is the number of the cells inspected rightward until the cell's right value is one. *Rowspan* is the number of the cells inspected downward until the cell's bottom value is one. When *Colspan*

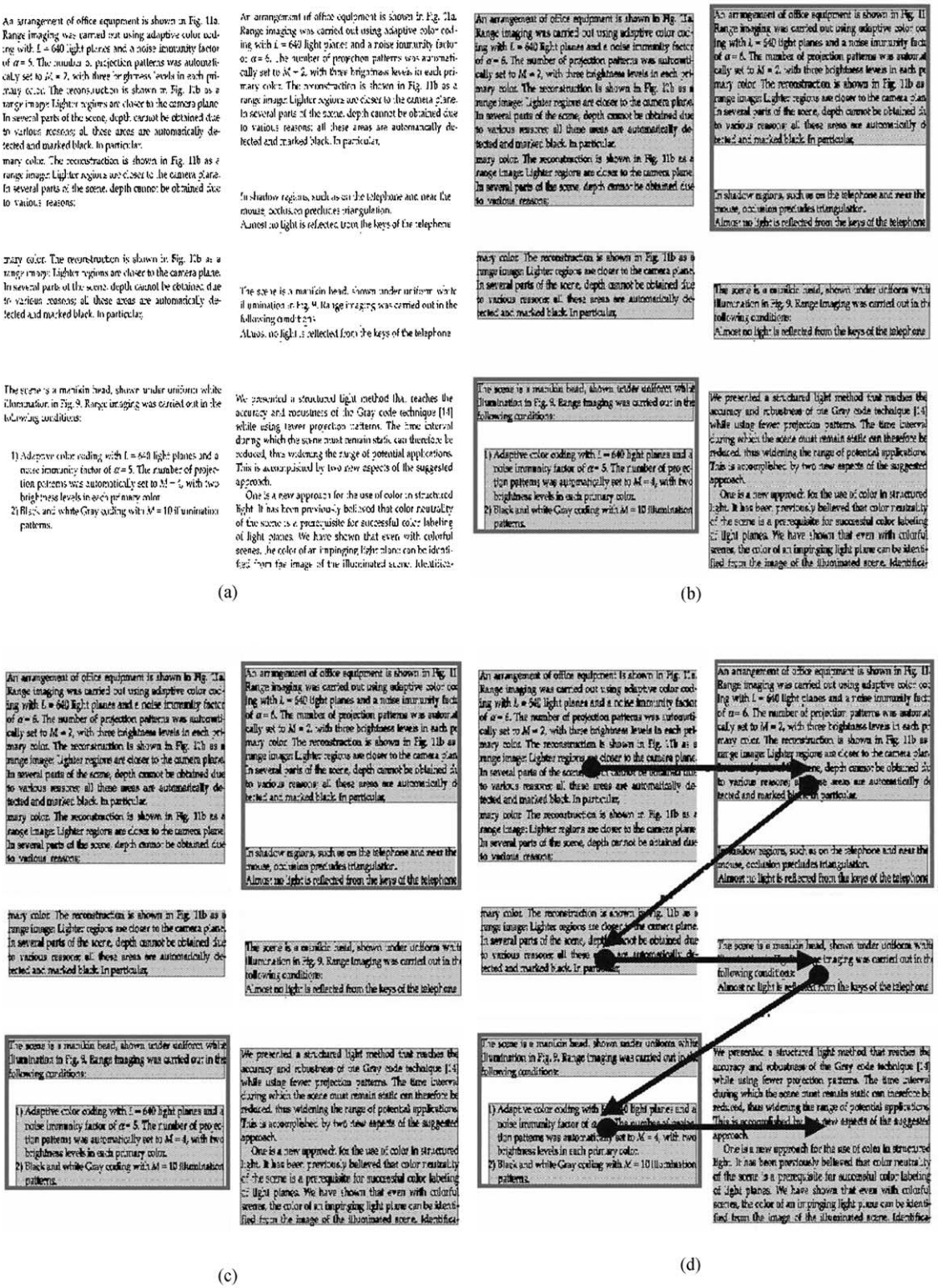


Fig. 5. Result images of each conversion stage based on table structure. (a) Original document. (b) Splitting and merging. (c) Virtual table generation. (d) Object ordering.

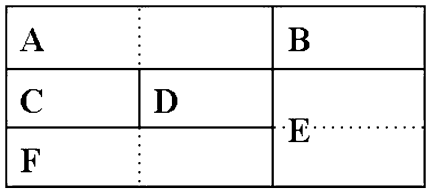


Fig. 6. The real line (solid line) and virtual line (dotted line) of the table.

and *Rowspan* are initialized, we insert ⟨TR⟩⟨TD⟩ tags if a new row starts; otherwise, we insert only a ⟨TD⟩ tag. In this way, when the processing on the last cell of the table is completed, the input multi-column document image is converted into its corresponding HTML document by table-to-HTML algorithm.

3.2. An approach based on the layer structure

In this section, we use the layer structure for the conversion of multi-column document images. A layer tag is one of the dynamic HTML(D-HTML) functions, which is supported by Explorer and Netscape browsers of version 4.0 or higher. Since a layer can be described as a container for any HTML elements, it serves well for texts, images, tables, plug-ins, other layers, and so on. Moreover, their arrangement is flexible. Owing to these properties of the layer structure, we are able to represent a hyperdocument on a browser, having a strong resemblance to the original complex multi-column document image.

Fig. 7 shows the overall conversion procedure based on the layer structure.

3.2.1. Object resizing

In order to have the converted HTML document fit into the screen's size regardless of the size of the input document, the objects in the image need to be resized in a constant rate before conversion.

When the size of the browser on which the hyperdocument is represented is maximized, it can be regarded as the width resolution of the user's screen. Then, the conversion rate (CR) for resizing can be defined as the ratio of the width of the input paper image (P_w) to the width resolution of the user's screen (S_w) as given below:

$$CR = P_w/S_w.$$

This conversion rate (CR) is also applied to font size and line space as follows:

- $\bar{O}_{fs} = O_{fs} \times CR$, where O_{fs} and $\bar{O}_{fs}$ are the font sizes of a text object and its resized object, respectively.
- $\bar{O}_{ls} = (\bar{O}_h - \bar{O}_{fs} \times TL_n)/(TL_n - 1)$, where $\bar{O}_{ls}$, $\bar{O}_h$, and TL_n are the resized line space, the height of a text object, and the number of the lines in the text object, respectively.

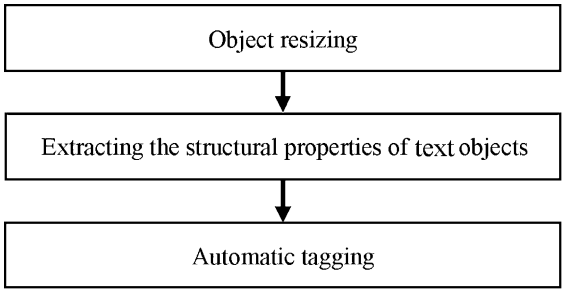


Fig. 7. Conversion procedure based on the layer structure.

Table 2
General properties of a layer tag

Property and attribute	Description
ID	Designates identification handle or name of layer
Position	Decides absolute or relative coordinates of layers
Left, top, width, height	Represents left, top, width, and height values of layer
Clip	Decides visible area of layer
Visibility	Hides or shows layer
Z-index	Designates an order of piling layers

It is very important to convert font size and line space of a paper document image into those of its HTML correctly, otherwise, objects can be overlapped or the sizes of spaces between objects can be much different from one another.

3.2.2. Extracting the structural properties of text objects

For more exact conversion, we must derive concrete properties of the objects. Table 2 shows some of the properties of the layer tag. Among them, we need the position, left, top, width, and height information, because they are sufficient for extracting various properties of a text object—alignment, indentation, line space, entered line, etc.

As a result of the geometrical structure analysis, we already know the starting and ending x and y coordinates of the objects and the font sizes of the text objects. Using such physical information, we classified lines of a text object as Indented Line, Entered Line, New Line and Normal Line, in Section 3.1.1. We can represent various kinds of text lines by inserting appropriate tags into an HTML document according to their types as shown below.

Let *TL* be the current text line.

(1) If *TL* is *Indented Line* (*IL*),
then "$text - indent : N$;" is inserted to the HTML document.

(2) If *TL* is *Entered Line* (*EL*),
then "⟨BR⟩" tag is inserted to the HTML document.
(3) If *TL* is *New Line* (*NeL*),
then "*text* − *indent* : *N*;" and "⟨BR⟩" tags are inserted to the HTML document.

3.2.3. Automatic tagging

Tagging is done automatically in the process of acquiring the properties of the objects. The original document image is shown in Fig. 8(a) and the HTML document code generated through conversion based on the table structure in Fig. 8(b) and the HTML document code

Figure 1 A young elite oil palm with developing fruit bunches.

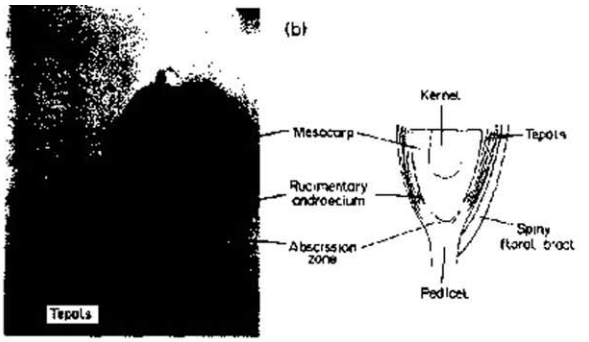


Figure 2 (a) Mature fruit showing the stamen primordia (rudimentary androecium), the tepals and the mesocarp. (b) Diagrammatic representation of the longitudinal section of a mature fruit.

results in the fall of the fruit to the ground, occurs only after the first stage is complete and involves a special role for the cells of the rudimentary androecium. Usually, the cells of the circlet immediately adjacent to the fruit base (position 2), though frequently some of those adjacent to the tepal bases (position 3), will undergo separation (figure 3c) so that the fruit falls free from the enclosing tepals. The upper parts of the tepals are by then brown and dry. The shed fruits are therefore either naked of any basal attachments (figure 4a) or may have fragments of the rudimentary androecium still adhering.

A naked fruit leaves a complete circlet of its rudimentary androecium still attached to the tepal bases and partly to the pad of stalk tissue at position 1 (figure 4b), but where some of the androecial ring clings to the fruit, the remaining tissue of the circlet stays with the tepals on the spike.

In exceptional circumstances, when spikes of fruit have been removed from a bunch before they are fully ripe, fruit shedding will eventually occur across the bases of the tepals (positions 4 and 5). Even in these conditions, separation always occurs first at position 1, to be followed by the second stage at either position 4 or 5. Such fruit are therefore shed still enclosed in the ring of tepals (figure 4c) and normal separation at positions 2 and 3 at the margins of the rudimentary androecium is bypassed.

The signal for harvest

On the plantations, palms are checked every few days for bunches at the appropriate stage of ripeness for cutting. Conventionally, the falling of a few ripe fruit to the ground is taken as the harvest signal, and for the taller palms this affords point of ripeness of a bunch that is otherwise difficult to see among the bases of the fronds. But before the bunch reaches the factory many more fruit are shed and so a yield loss is established.

A sexual aberration

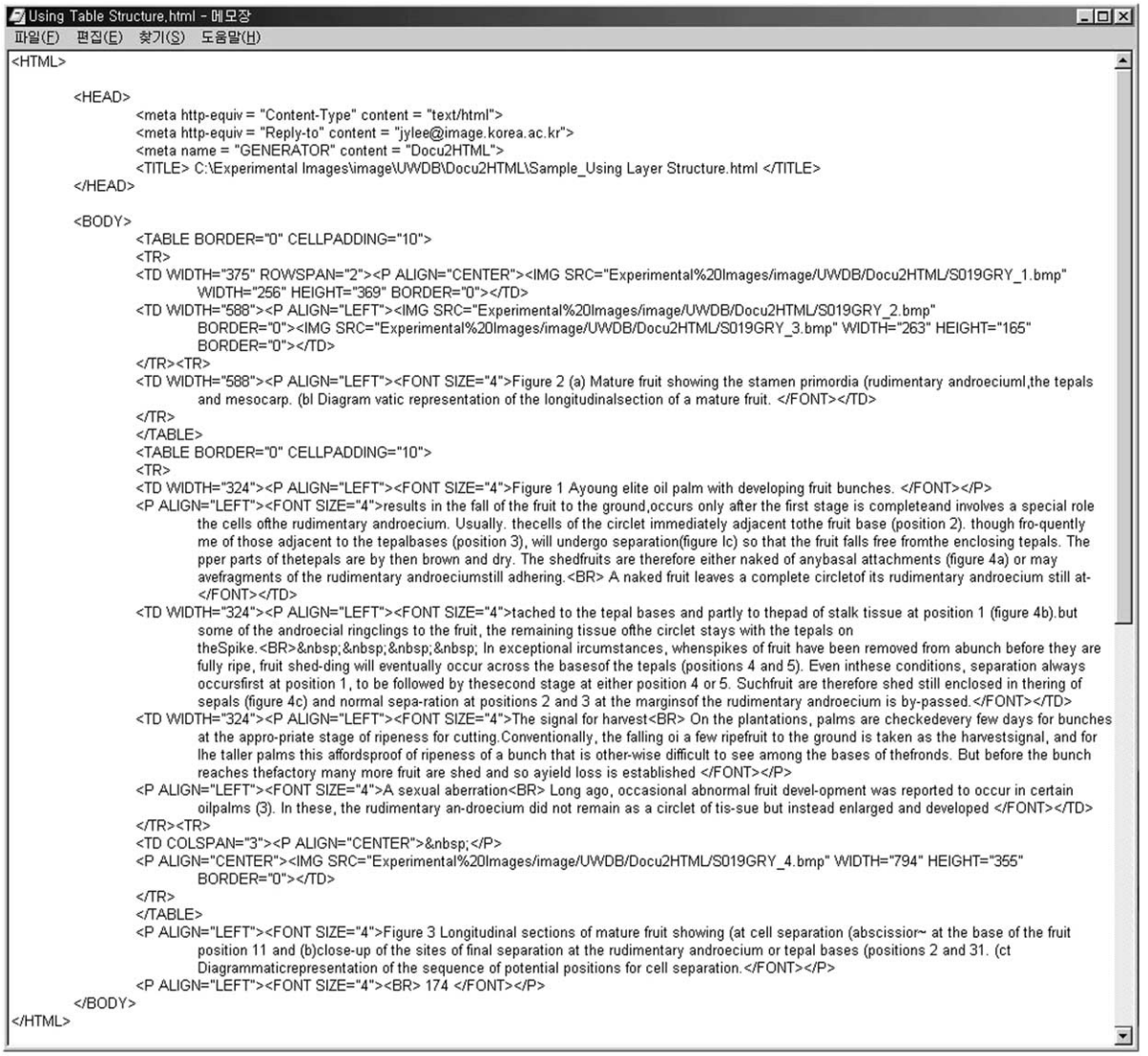
Long ago, occasional abnormal fruit development was reported to occur in certain oil palms [3]. In these, the rudimentary androecium did not remain as a circlet of tissue but instead enlarged and developed

Figure 3 Longitudinal sections of mature fruit showing (a) cell separation (abscission) at the base of the fruit (position 1) and (b) close up of the sites of final separation at the rudimentary androecium or tepal bases (positions 2 and 3). (c) Diagrammatic representation of the sequence of potential positions for cell separation.

174

(a)

Fig. 8. Comparison of converted HTML document codes. (a) Original document image. (b) The converted HTML document using the table structure. (c) The converted HTML document using the layer structure.



Using Table Structure.html - 메모장

파일(F) 편집(E) 찾기(S) 도움말(H)

```
<HTML>

        <HEAD>
                <meta http-equiv = "Content-Type" content = "text/html">
                <meta http-equiv = "Reply-to" content = "jylee@image.korea.ac.kr">
                <meta name = "GENERATOR" content = "Docu2HTML">
                <TITLE> C:\Experimental Images\image\UWDB\Docu2HTML\Sample_Using Layer Structure.html </TITLE>
        </HEAD>

        <BODY>
                <TABLE BORDER="0" CELLPADDING="10">
                <TR>
                <TD WIDTH="375" ROWSPAN="2"><P ALIGN="CENTER"><IMG SRC="Experimental%20Images/image/UWDB/Docu2HTML/S019GRY_1.bmp"
                        WIDTH="256" HEIGHT="369" BORDER="0"></TD>
                <TD WIDTH="588"><P ALIGN="LEFT"><IMG SRC="Experimental%20Images/image/UWDB/Docu2HTML/S019GRY_2.bmp"
                        BORDER="0"><IMG SRC="Experimental%20Images/image/UWDB/Docu2HTML/S019GRY_3.bmp" WIDTH="263" HEIGHT="165"
                        BORDER="0"></TD>
                </TR><TR>
                <TD WIDTH="588"><P ALIGN="LEFT"><FONT SIZE="4">Figure 2 (a) Mature fruit showing the stamen primordia (rudimentary androeciuml,the tepals
                        and mesocarp. (bl Diagram vatic representation of the longitudinalsection of a mature fruit. </FONT></TD>
                </TR>
                </TABLE>
                <TABLE BORDER="0" CELLPADDING="10">
                <TR>
                <TD WIDTH="324"><P ALIGN="LEFT"><FONT SIZE="4">Figure 1 Ayoung elite oil palm with developing fruit bunches. </FONT></P>
                <P ALIGN="LEFT"><FONT SIZE="4">results in the fall of the fruit to the ground,occurs only after the first stage is completeand involves a special role
                        the cells ofthe rudimentary androecium. Usually. thecells of the circlet immediately adjacent tothe fruit base (position 2). though fro-quently
                        me of those adjacent to the tepalbases (position 3), will undergo separation(figure lc) so that the fruit falls free fromthe enclosing tepals. The
                        pper parts of thetepals are by then brown and dry. The shedfruits are therefore either naked of anybasal attachments (figure 4a) or may
                        avefragments of the rudimentary androeciumstill adhering.<BR> A naked fruit leaves a complete circletof its rudimentary androecium still at-
                        </FONT></TD>
                <TD WIDTH="324"><P ALIGN="LEFT"><FONT SIZE="4">tached to the tepal bases and partly to thepad of stalk tissue at position 1 (figure 4b).but
                        some of the androecial ringclings to the fruit, the remaining tissue ofthe circlet stays with the tepals on
                        theSpike.<BR>&amp;nbsp;&amp;nbsp;&amp;nbsp;&amp;nbsp; In exceptional ircumstances, whenspikes of fruit have been removed from abunch before they are
                        fully ripe, fruit shed-ding will eventually occur across the basesof the tepals (positions 4 and 5). Even inthese conditions, separation always
                        occursfirst at position 1, to be followed by thesecond stage at either position 4 or 5. Suchfruit are therefore shed still enclosed in thering of
                        sepals (figure 4c) and normal sepa-ration at positions 2 and 3 at the marginsof the rudimentary androecium is by-passed.</FONT></TD>
                <TD WIDTH="324"><P ALIGN="LEFT"><FONT SIZE="4">The signal for harvest<BR> On the plantations, palms are checkedevery few days for bunches
                        at the appro-priate stage of ripeness for cutting.Conventionally, the falling oi a few ripefruit to the ground is taken as the harvestsignal, and for
                        lhe taller palms this affordsproof of ripeness of a bunch that is other-wise difficult to see among the bases of thefronds. But before the bunch
                        reaches thefactory many more fruit are shed and so ayield loss is established </FONT></P>
                <P ALIGN="LEFT"><FONT SIZE="4">A sexual aberration<BR> Long ago, occasional abnormal fruit devel-opment was reported to occur in certain
                        oilpalms (3). In these, the rudimentary an-droecium did not remain as a circlet of tis-sue but instead enlarged and developed </FONT></TD>
                </TR><TR>
                <TD COLSPAN="3"><P ALIGN="CENTER">&amp;nbsp;</P>
                <P ALIGN="CENTER"><IMG SRC="Experimental%20Images/image/UWDB/Docu2HTML/S019GRY_4.bmp" WIDTH="794" HEIGHT="355"
                        BORDER="0"></TD>
                </TR>
                </TABLE>
                <P ALIGN="LEFT"><FONT SIZE="4">Figure 3 Longitudinal sections of mature fruit showing (at cell separation (abscissior~ at the base of the fruit
                        position 11 and (b)close-up of the sites of final separation at the rudimentary androecium or tepal bases (positions 2 and 31. (ct
                        Diagrammaticrepresentation of the sequence of potential positions for cell separation.</FONT></P>
                <P ALIGN="LEFT"><FONT SIZE="4"><BR> 174 </FONT></P>
        </BODY>
</HTML>
```

(b)

Fig. 8. (*Continued*).

generated through conversion based on the layer structure in Fig. 8(c).

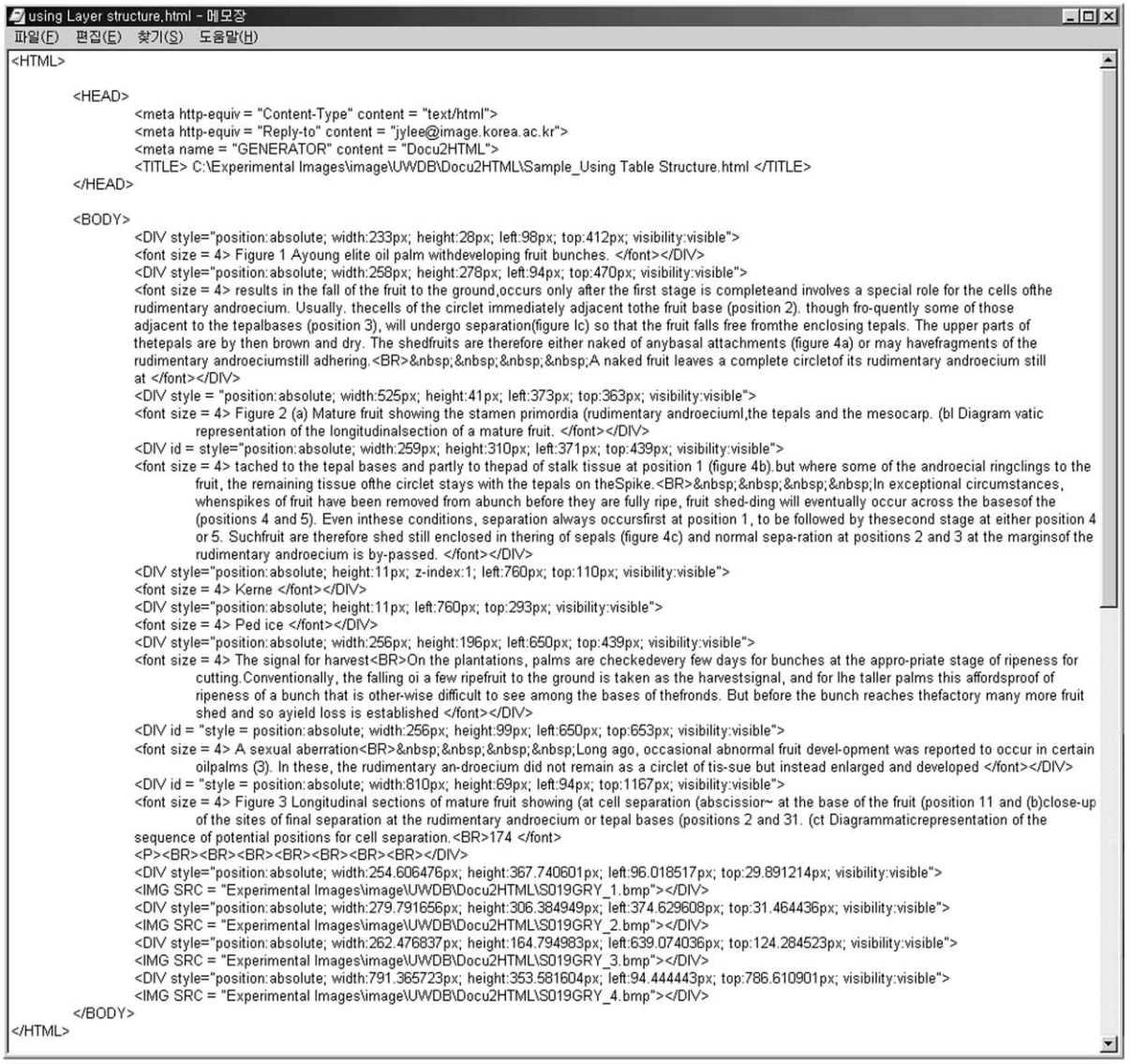
4. Generation of a structured hyperdocument based on logical structure analysis

In this section, we choose many related papers as input, and automatically generate a table of contents page by extracting section titles from the input images and arranging them with respect to their hierarchical relation.

Since HTML tags are very limited in availability and representation, it is difficult to fully express enough meaningful information of objects in a document image to HTML format. For that reason, the creation of a table of contents page is very worthwhile work, which provides a clear overview of the input document images and offers hyperlinks to the contents.

Fig. 9 shows the logical structure of a general technical paper. We generate the table of contents page of a technical paper that includes various objects and has a well-formatted structure.

To create a table of contents page, an initial logical structure analysis (or logical labeling) of the document images is required. It labels the objects in the document images with logical units (section number, page number,



using Layer structure.html - 메모장

파일(F) 편집(E) 찾기(S) 도움말(H)

```
<HTML>

        <HEAD>
                <meta http-equiv = "Content-Type" content = "text/html">
                <meta http-equiv = "Reply-to" content = "jylee@image.korea.ac.kr">
                <meta name = "GENERATOR" content = "Docu2HTML">
                <TITLE> C:\Experimental Images\image\UWDB\Docu2HTML\Sample_Using Table Structure.html </TITLE>
        </HEAD>

        <BODY>
                <DIV style="position:absolute; width:233px; height:28px; left:98px; top:412px; visibility:visible">
                <font size = 4> Figure 1 Ayoung elite oil palm withdeveloping fruit bunches. </font></DIV>
                <DIV style="position:absolute; width:258px; height:278px; left:94px; top:470px; visibility:visible">
                <font size = 4> results in the fall of the fruit to the ground,occurs only after the first stage is completeand involves a special role for the cells ofthe
                rudimentary androecium. Usually. thecells of the circlet immediately adjacent tothe fruit base (position 2). though fro-quently some of those
                adjacent to the tepalbases (position 3), will undergo separation(figure lc) so that the fruit falls free fromthe enclosing tepals. The upper parts of
                thetepals are by then brown and dry. The shedfruits are therefore either naked of anybasal attachments (figure 4a) or may havefragments of the
                rudimentary androeciumstill adhering.<BR>    A naked fruit leaves a complete circletof its rudimentary androecium still
                at </font></DIV>
                <DIV style = "position:absolute; width:525px; height:41px; left:373px; top:363px; visibility:visible">
                <font size = 4> Figure 2 (a) Mature fruit showing the stamen primordia (rudimentary androeciuml,the tepals and the mesocarp. (bl Diagram vatic
                        representation of the longitudinalsection of a mature fruit. </font></DIV>
                <DIV id = style="position:absolute; width:259px; height:310px; left:371px; top:439px; visibility:visible">
                <font size = 4> tached to the tepal bases and partly to thepad of stalk tissue at position 1 (figure 4b).but where some of the androecial ringclings to the
                        fruit, the remaining tissue ofthe circlet stays with the tepals on theSpike.<BR>    In exceptional circumstances,
                        whenspikes of fruit have been removed from abunch before they are fully ripe, fruit shed-ding will eventually occur across the basesof the
                        (positions 4 and 5). Even inthese conditions, separation always occursfirst at position 1, to be followed by thesecond stage at either position 4
                        or 5. Suchfruit are therefore shed still enclosed in thering of sepals (figure 4c) and normal sepa-ration at positions 2 and 3 at the marginsof the
                        rudimentary androecium is by-passed. </font></DIV>
                <DIV style="position:absolute; height:11px; z-index:1; left:760px; top:110px; visibility:visible">
                <font size = 4> Kerne </font></DIV>
                <DIV style="position:absolute; height:11px; left:760px; top:293px; visibility:visible">
                <font size = 4> Ped ice </font></DIV>
                <DIV style="position:absolute; width:256px; height:196px; left:650px; top:439px; visibility:visible">
                <font size = 4> The signal for harvest<BR>On the plantations, palms are checkedevery few days for bunches at the appro-priate stage of ripeness for
                        cutting.Conventionally, the falling oi a few ripefruit to the ground is taken as the harvestsignal, and for lhe taller palms this affordsproof of
                        ripeness of a bunch that is other-wise difficult to see among the bases of thefronds. But before the bunch reaches thefactory many more fruit
                        shed and so ayield loss is established </font></DIV>
                <DIV id = "style = position:absolute; width:256px; height:99px; left:650px; top:653px; visibility:visible">
                <font size = 4> A sexual aberration<BR>    Long ago, occasional abnormal fruit devel-opment was reported to occur in certain
                        oilpalms (3). In these, the rudimentary an-droecium did not remain as a circlet of tis-sue but instead enlarged and developed </font></DIV>
                <DIV id = "style = position:absolute; width:810px; height:69px; left:94px; top:1167px; visibility:visible">
                <font size = 4> Figure 3 Longitudinal sections of mature fruit showing (at cell separation (abscissior~ at the base of the fruit (position 11 and (b)close-up
                        of the sites of final separation at the rudimentary androecium or tepal bases (positions 2 and 31. (ct Diagrammaticrepresentation of the
                sequence of potential positions for cell separation.<BR>174 </font>
                <P><BR><BR><BR><BR><BR><BR><BR></DIV>
                <DIV style="position:absolute; width:254.606476px; height:367.740601px; left:96.018517px; top:29.891214px; visibility:visible">
                <IMG SRC = "Experimental Images\image\UWDB\Docu2HTML\S019GRY_1.bmp"></DIV>
                <DIV style="position:absolute; width:279.791656px; height:306.384949px; left:374.629608px; top:31.464436px; visibility:visible">
                <IMG SRC = "Experimental Images\image\UWDB\Docu2HTML\S019GRY_2.bmp"></DIV>
                <DIV style="position:absolute; width:262.476837px; height:164.794983px; left:639.074036px; top:124.284523px; visibility:visible">
                <IMG SRC = "Experimental Images\image\UWDB\Docu2HTML\S019GRY_3.bmp"></DIV>
                <DIV style="position:absolute; width:791.365723px; height:353.581604px; left:94.444443px; top:786.610901px; visibility:visible">
                <IMG SRC = "Experimental Images\image\UWDB\Docu2HTML\S019GRY_4.bmp"></DIV>
        </BODY>
</HTML>
```

(c)

Fig. 8. (*Continued*).

caption, header, footer, etc.) according to the role they play in the documents, and specifies the relationship among the units. we extract only section titles and page numbers among the logical units.

Fig. 10 shows the procedure for the generation of a structured table of contents hyperdocument.

4.1. Section title candidate extraction

As shown in Fig. 9, a general technical paper consists of N related pages, each with page number, header, footer, table, image, text body, section title, etc. It is natural that the first page contains the title and abstract of the technical paper, and the last page contains the list of references generally.

For the generation of a table of contents, first, we extract section title candidates from text objects, referring to text line types defined in Section 3.1.1. These text objects were already recognized by the character recognition engine. When current text line is of any type starting with the special pattern mentioned before and the previous line is of *EL* type, the current line is regarded as a section title candidate. Fig. 11 shows an example of a section title candidate and its special pattern.

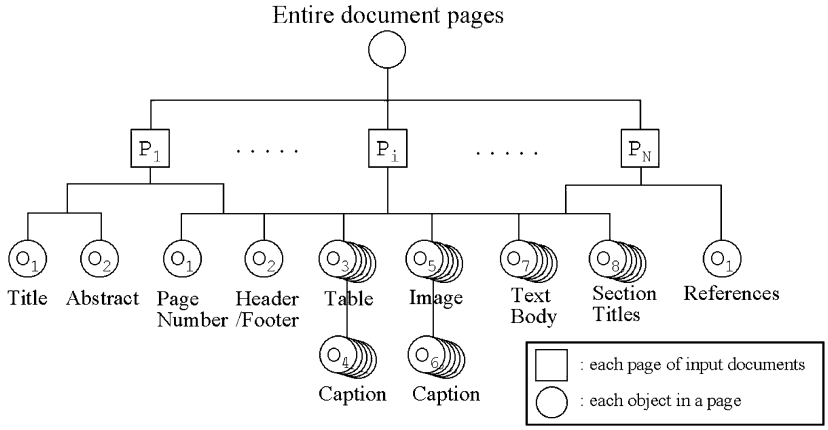


Fig. 9. Logical structure of a general technical paper.

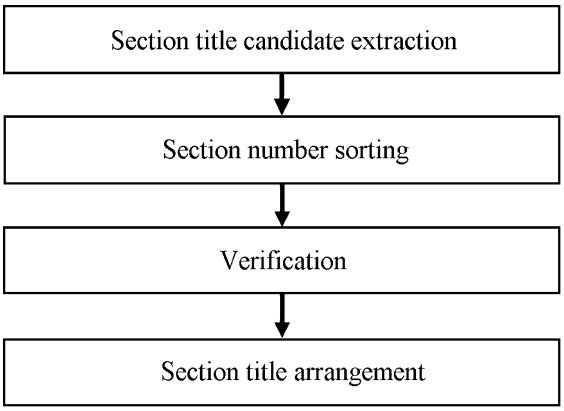


Fig. 10. A procedure for a structured ToC page generation.

Next, we choose a text object in each input document image as a page number if it satisfies the conditions given below:

- It is the topmost or bottommost text object in a document image.
- It consists of only one line.
- Its width is smaller than the value, $\alpha \times$ font size, where α is an arbitrary positive integer.

4.2. Section number sorting

In this section, we sort the extracted section title candidates by section number in ascending order at each level. To do so, we modify the pattern of a section number as shown in Fig. 12(b).

A sequence of non-negative numbers and symbols in a section title candidate represents a section number, followed by a series of ($-$ 1)'s which represents its section title text. A symbol is denoted as a dot or a space in the figure. In a section number, each symbol separates levels of a section number with the leftmost level being the highest level. Starting at the highest level, all section title candidates are sorted by section number in ascending order at each level.

4.3. Verification

Fig. 13 shows the hierarchical structure tree of the section numbers. The section numbers at each level are searched for any missing or inappropriate section numbers, which are then adjusted using the hierarchical tree. In addition, their suitability is verified. Since the section titles should have the same physical and logical information (font style, font size, page number, line space), such information is utilized in adjustment and verification. An example of the physical and logical information on section title candidates is shown in Fig. 14.

If there is a section title candidate with an improper section number, that is, a section number is out of sequence, the information on the candidate is compared with that of the candidates at the higher or lower level. If they match, the candidate is moved to the level and all the candidates at the level are checked again. If no match occurs, the candidate may be unsuitable and is thereby removed.

4.4. Section title arrangement

In this section, we arrange section titles hierarchically in a table of contents page. Each section title is sequentially inserted in the page with ⟨UL⟩ and ⟨/UL⟩ are added to the front and the end of the section title,

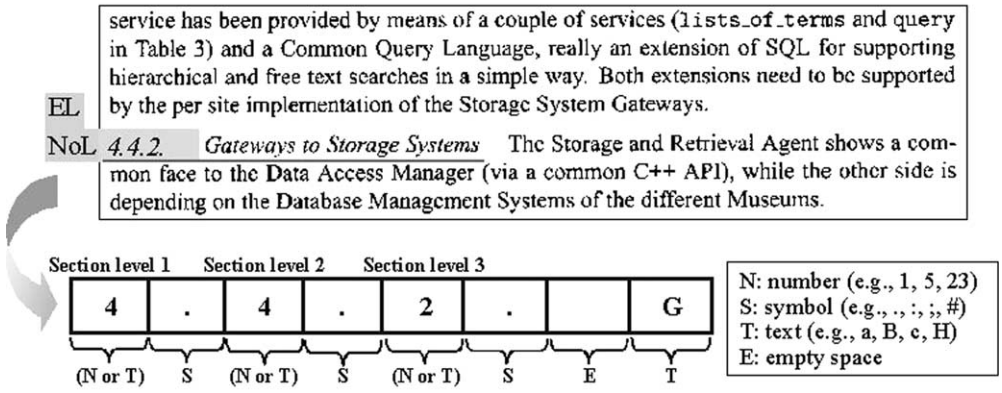


Fig. 11. An example of a section title candidate and its special pattern.

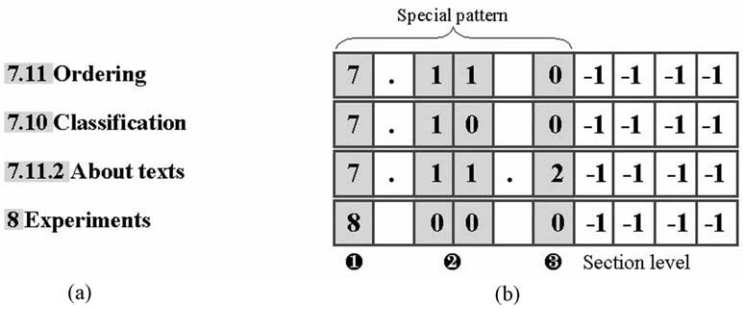


Fig. 12. Modification of the pattern for sorting. (a) Section title candidates. (b) The modified pattern.

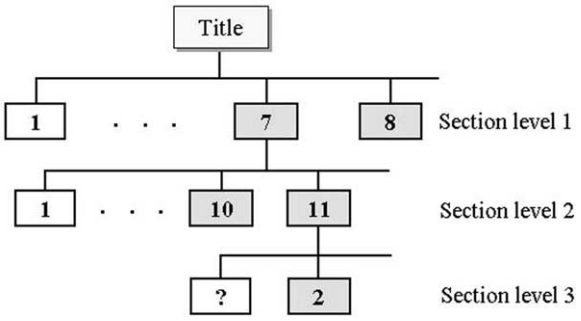


Fig. 13. Hierarchical structure of section numbers.

respectively, as many times as the number of the symbols (except spaces) in the pattern of the section number.

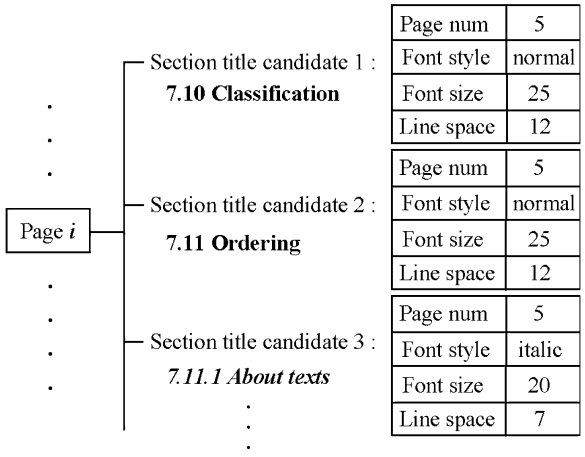


Fig. 14. Physical and logical information of section title candidates.

5. Experimental results and analysis

5.1. Experimental environment

The proposed methods were implemented on Pentium MMX 166 MHz PC. Experiments on HTML conversion were carried out with 300 document images taken from magazines, newspapers, books, scientific and technical journals, manuals, and UWDB (the database of University of Washington) [10]. Generation of a table of

contents page was tested on images of technical papers which are collected from proceedings of scientific conferences and journals.

5.2. Experimental results

Figs. 15–17 show the converted example images using the table and layer structure. As a result of using these two approaches on various document images, HTML documents are generated very similar to the original document images both in appearance and logic, as shown in Fig. 15.

In case of the conversion based on the table structure, the sizes of spaces between the objects in the converted HTML document may be considerably different from one another. This is because the objects have been merged for arrangement in a table format convertible into an HTML document. Besides, an object lying across columns cannot be represented, although a partial overlap of objects in a column can be avoided. Therefore, the conversion based on the table structure is adequate for conversion of formatted document images like general books and technical papers. On the other hand, in the case of using the layer structure, the conversion of complex and unformatted multi-column document images (i.e., documents without regular layout or structure, like magazines, advertisements, etc.), is performed well. However, when the font size or line space of a text object is incorrectly calculated, several objects may be overlapped and the spaces between the objects may also become larger or smaller than those of the objects in the original image.

As shown in Fig. 16, the document image is converted better using the table structure than using the layer structure. Note that in Fig. 16(c), text objects are overlapped when the layer structure used. The document image shown in Fig. 17 is converted better using the layer structure than using the table structure. Note that the size of space between objects is considerably different from one another, in Fig. 17(b) when the table structure used.

Fig. 18 shows a structured table of contents page created while converting the input document images. Section titles are arranged hierarchically. Each section title is hyperlinked to the beginning of the section and has its page number.

Table 3 shows the experimental result of extracting section titles from different kinds of paper images. For the table of contents generation, we experimented with objects of the documents which have a lot of section titles. N_c, N_{fn} and N_{fp} denote the number of section titles correctly extracted, false negatives, and false positives, respectively. In X/Y representation of N_c, X and Y denote the number of the correctly extracted section titles and the total number of the section titles in the input documents, respectively. As shown in Table 3, a table of contents page was generated with accuracy of about 90%.

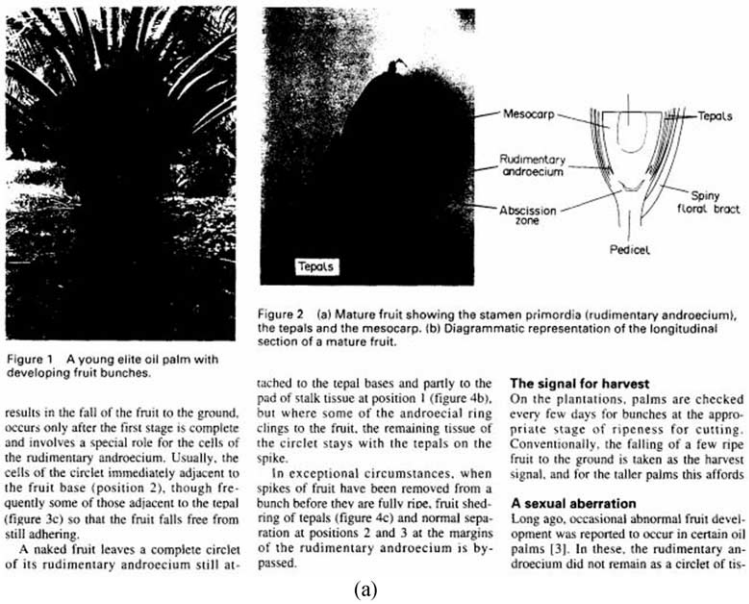
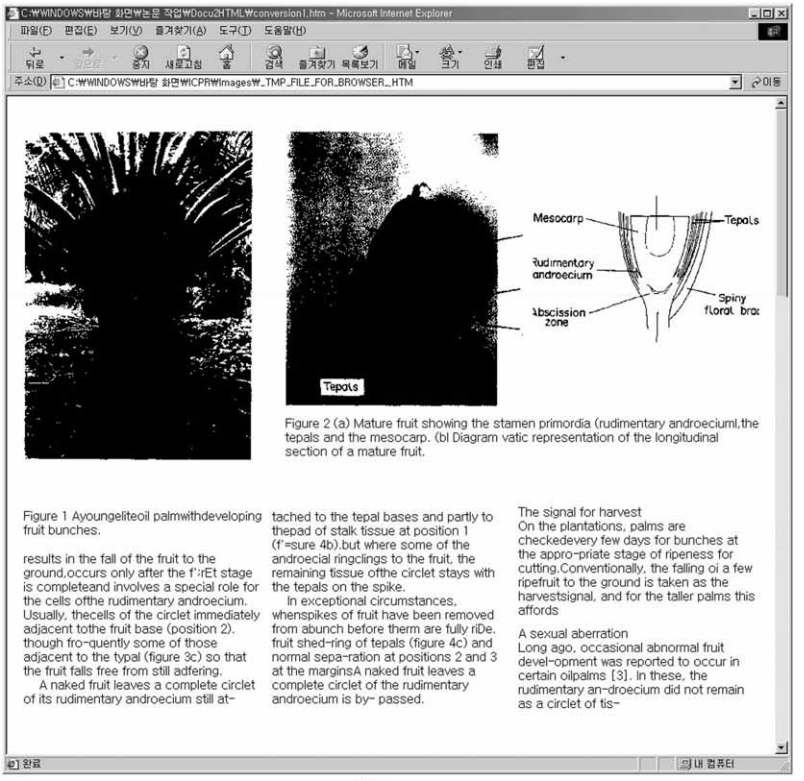
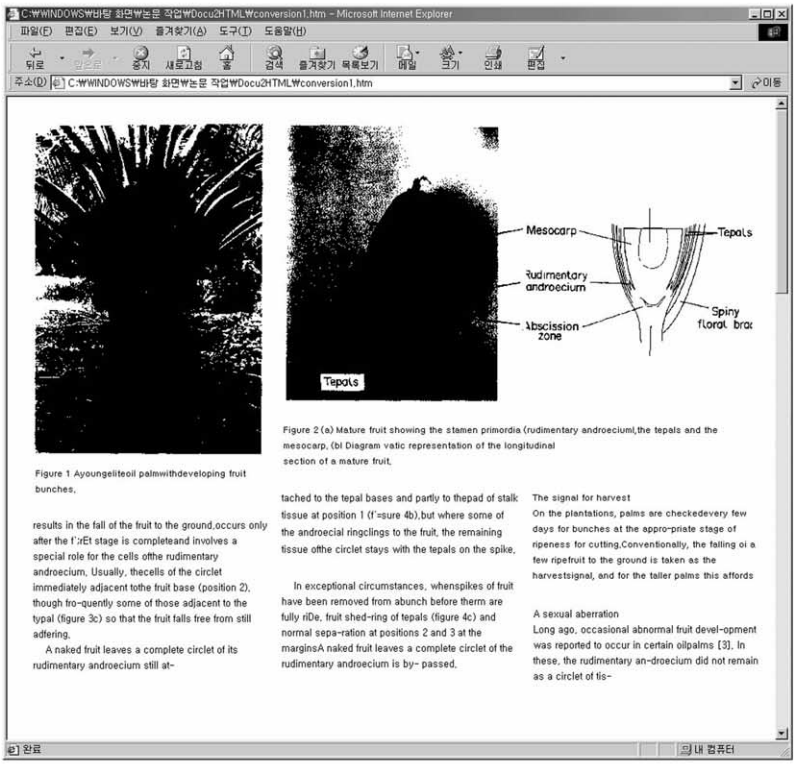


Fig. 15. Example 1 — conversion of multi-column document image. (a) Original document image. (b) The generated hyperdocument using table structure. (c) The generated hyperdocument using layer structure.



(b)



(c)

Fig. 15. (*Continued*).

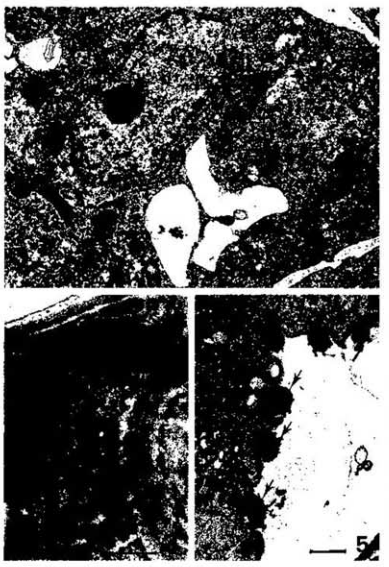


Fig. 3. Details of tapetal cell structure at a time corresponding to telophase II in MMCs. Mitochondria (*M*) contain a light matrix and conspicuous cristae, plastids (*P*), with dark stroma and dilated lamellae. Abundant cisternae of endoplasmic reticulum (*ER*) are seen near nucleus (*N*). Dictyosomes (*D*) are inactive.

Fig. 4. Accumulation of pro-Ubisch bodies (*arrow*) between plasmalemma (*Pl*) and tapetal cell wall (*TW*) at tetrad stage.

Fig. 5. Ubisch bodies (*arrows*) on surface of tapetal plasmalemma at young microspore stage. *Bars*: 1 μm

ent tectum formation (Fig. 6). Detailed descriptions of pollen wall synthesis during normal development are presented elsewhere (Majewska-Sawka et al. 1992), hence this report will center on the most important events and their relation with aberrant wall development around MS microspores.

Tapetum. During the tetrad period, the mitochondria show a further reduction in size (Table 1), and the plastids seem to present greater morphological variability. In addition to the forms observed during meiosis of MMCs (Fig. 3) another plastid type with more lightly stained stroma and long, narrow, irregularly arranged lamella appear. Figures of dividing plastids are seen frequently. Occasionally, concentric arrangements of ER are noted. In depressions formed between the plasmalemma and tapetal cell walls, large numbers of spherical electron-grey pro-Ubisch bodies accumulate (Fig. 4).

(a)

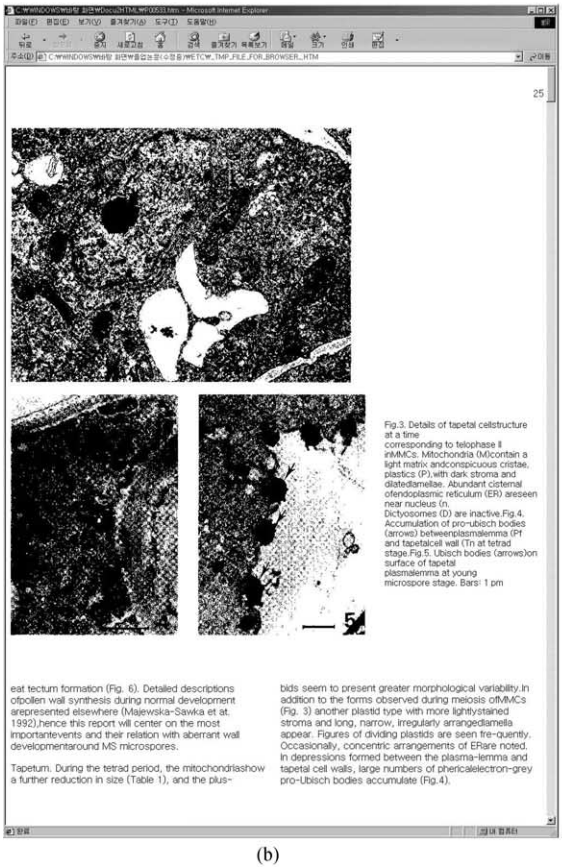


Fig.3. Details of tapetal cellstructure at a time corresponding to telophase II inMMCs. Mitochondria (M)contain a light matrix andconspicuous cristae, plastics (P),with dark stroma and dilatedlamellae. Abundant cisternal ofendoplasmic reticulum (ER) areseen near nucleus (n. Dictyosomes (D) are inactive.Fig.4. Accumulation of pro-ubisch bodies (arrows) betweenplasmalemma (Pf and tapetalcell wall (Tn at tetrad stage.Fig.5. Ubisch bodies (arrows)on surface of tapetal plasmalemma at young microspore stage. Bars: 1 pm

eat tectum formation (Fig. 6). Detailed descriptions ofpollen wall synthesis during normal development arepresented elsewhere (Majewska-Sawka et at. 1992),hence this report will center on the most importantevents and their relation with aberrant wall developmentaround MS microspores.

Tapetum. During the tetrad period, the mitochondriashow a further reduction in size (Table 1), and the plus-

bids seem to present greater morphological variability.In addition to the forms observed during meiosis ofMMCs (Fig. 3) another plastid type with more lightlystained stroma and long, narrow, irregularly arrangedlamella appear. Figures of dividing plastids are seen fre-quently. Occasionally, concentric arrangements of ERare noted. In depressions formed between the plasma-lemma and tapetal cell walls, large numbers of phericalelectron-grey pro-Ubisch bodies accumulate (Fig.4).

(b)

Fig.3. Details of tapetal cellstructure at a time corresponding to telophase II inMMCs. Mitochondria (M) contain a light matrix andconspicuous cristae, plastics (P),with dark stroma and dilatedlamellae. Abundant cisternal ofendoplasmic reticulum (ER) areseen near nucleus (n. Dictyosomes (D) are inactive.Fig.4. Accumulation of pro-ubisch bodies (arrows) betweenplasmalemma (Pf and tapetalcell wall (Tn at tetrad stage.Fig.5. Ubisch bodies (arrows)on surface of tapetal plasmalemma at young microspore stage. Bars: 1 pm

eat tectum formation (Fig. 6). Detailed descriptions ofpollen wall synthesis during normal development arepresented elsewhere (Majewska-Sawka et at. 1992),hence this report will center on the most importantevents and their relation with aberrant wall developmentaround MS microspores.

Tapetum. During the tetrad period, the mitochondriashow a further reduction in size (Table 1), and the plus-

bids seem to present greater morphological variability.In addition to the forms observed during meiosis ofMMCs (Fig. 3) another plastid type with more lightlystained stroma and long, narrow, irregularly arrangedlamella appear. Figures of dividing plastids are seen fre-quently. Occasionally, concentric arrangements of ERare noted. In depressions formed between the plasma-lemma and tapetal cell walls, large numbers of sphericalelectron-grey pro-Ubisch bodies accumulate (Fig.4).

(c)

Fig. 16. Example 2—conversion of multi-column document image. (a) Original document image. (b) Hyperdocument based on table structure. (c) Hyperdocument based on layer structure.

(a)

(b)

(c)

Fig. 17. Example 3 — conversion of multi-column document image. (a) Original document image. (b) Hyperdocument based on table structure. (c) Hyperdocument based on layer structure.

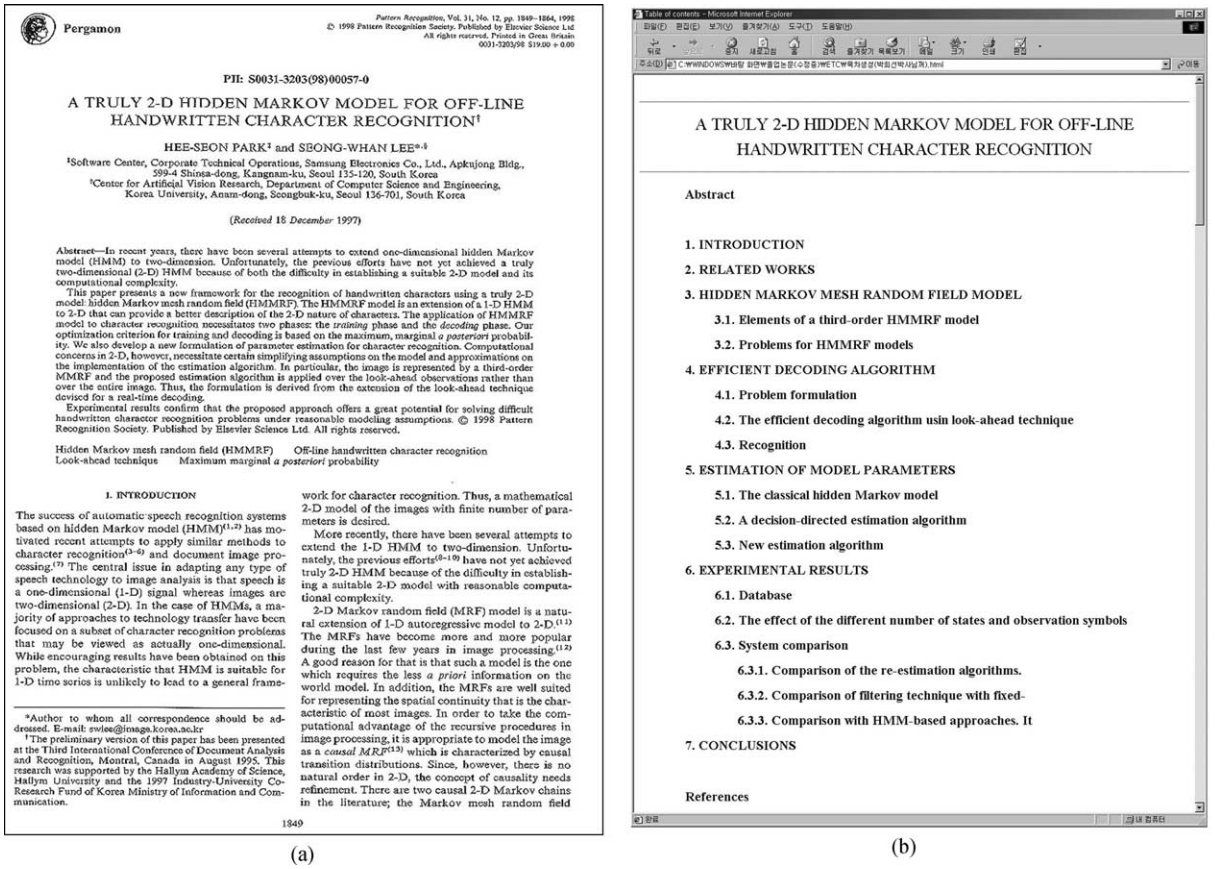


Fig. 18. An example of a structured table of contents page generation. (a) Input document images. (b) Table of contents page.

Table 3
Results of section title extraction

Papers		Section title extraction		
		N_{fn}	N_{fp}	N_c
Journal	PR[a]	20	13	134/158 (85%)
	PAMI[b]	19	11	142/158 (90%)
Conference	ICPR[c]	8	7	65/71 (92%)
	ICDAR[d]	7	8	65/73 (89%)

[a]Pattern Recognition, Vol. 30, No. 12, 1998, pp. 1849–1864, and Vol. 31, No. 4, 1998.
[b]IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 20, No. 12, 1998.
[c]Proc. of International Conference on Pattern Recognition, 1998, pp. 949–984.
[d]Proc. of International Conference on Document Analysis and Recognition, 1997, pp. 1–50.

6. Conclusions and further research

In this paper, we proposed two methods — one using the table structure and the other using the layer structure — converting multi-column document images into HTML documents and also proposed a method for generating a structured table of contents page by extracting the section titles from input document images.

For the conversion of each paper image into its hyper-document, the proposed conversion methods were tested on various kinds of complex multi-column document images. Experimental results revealed that the proposed methods performed well. However, they showed different performance on different types of document images. Hence, a scheme is needed to determine which one of the conversion methods should be applied for better performance when document images are given, for further research.

For the generation of a structured table of contents page, experiments were carried out on technical papers. Experimental results showed that, using the proposed

table of contents generation method, we could create a table of contents page by extracting the section titles from input document images without regard to the accuracy of character recognition.

The performance of the proposed methods still depends on the result of character recognition and the geometric layout analysis of input document images. Therefore, the methods must be improved to perform well regardless of the result of preprocessing of the conversion.

References

[1] T. Tanaka, S. Tsuruoka, Table form document understanding using node classification method and HTML document generation, Proceedings of the Third IAPR Workshop on Document Analysis Systems, Nagano, Japan, 1998, pp. 157–158.

[2] T.G. Kieninger, A. Dengel, A paper-to-HTML table converting system, Proceedings of the Third IAPR Workshop on Document Analysis Systems, Nagano, Japan, 1998, pp. 356–365.

[3] M. Worring, A.W.M. Smeulders, Content based internet access to paper documents, Int. J. Document Anal. Recognition 1 (4) (1999) 209–220.

[4] C. Faure, Preattentive reading and selective attention for document image analysis, Proceedings of the Fifth International Conference on Document Analysis and Recognition, Bangalore, India, 1999, pp. 577–580.

[5] J.L. Fisher, Logical structure descriptions of segmented document images, Proceedings of the First International Conference on Document Analysis and Recognition, Saint-Malo, France, 1991, pp. 302–310.

[6] C. Lin, Y. Niwa, S. Narita, Logical structure analysis of book document images using contents information, Proceedings of the Fourth International Conference on Document Analysis and Recognition, Ulm, Germany, 1997, pp. 1048–1054.

[7] Y. Ishitani, Logical structure analysis of document images based on emergent computation, Proceedings of the Fifth International Conference on Document Analysis and Recognition, Bangalore, India, 1999, pp. 189–192.

[8] T. Saitoh, M. Tachikawa, T. Yamaai, Document image segmentation and text area ordering, Proceedings of the Second International Conference on Document Analysis and Recognition, Tsukuba, Japan, 1993, pp. 323–329.

[9] T. Kochi, T. Saitoh, A layout-free method for extracting elements from document images, Proceedings of the Third IAPR Workshop on Document Analysis Systems, Nagano, Japan, 1998, pp. 336–345.

[10] I. Phillips, S. Chen, R. Haralick, CD-ROM document database standard, Proceedings of the Second International Conference on Document Analysis and Recognition, Tsukuba, Japan, 1993, pp. 478–483.

About the Author—JI-YEON LEE was born in Seoul, Korea, in 1975. She received the B.S. degree in Information Processing from Sangmyung University, Korea, in 1997 and received the M.S. degree in Computer Science and Engineering at Korea University, Seoul, Korea, in 2000.

She is currently working as a research engineer at Samsung Electronics, Co., Ltd. in Korea. Her research interests include multimedia, hyperdocument and document structure analysis.

About the Author—JEONG-SEON PARK received the B.S. and M.S. degrees in Computer Science from Chungbuk National University, Cheongju, Korea, in 1988 and 1992, respectively. She is currently working toward the Ph.D. degree in computer science and engineering at Korea University, Seoul, Korea.

From February 1994 to July 1996, she was a research engineer in S/W R&D center at Hyundai Electronics, Co., Ltd. in Korea and worked as an advanced research engineer at Hyundai Information Technology, Co., Ltd. in Korea from August 1996 to March 1999. She was the winner of the Annual Best Paper Award of the Korea Information Science Society in 1994. Her research interests include pattern recognition, image processing and computer vision.

About the Author—HYERAN BYUN received the B.S. and M.S. degrees in Mathematics from Yonsei University, Korea. She received her Ph.D. degree in Computer Science from Purdue University, West Lafayette, Indiana. She was an assistant professor in Hallym University, Chooncheon, Korea from 1994–1995. Since 1995, she has been an associate professor of Computer Science at Yonsei University, Korea. Her research interests include multimedia, computer vision, image processing, and pattern recognition.

About the Author—JONGSUB MOON received the B.S. degree and M.S. degree in Computer Science from Seoul National University, Korea in 1981 and 1983, respectively. Also, he received the Ph.D. degree in Computer Science from Illinois Institute of Technology, Illinois, U.S.A., in 1991. He worked at Gold Star Tele-electric research Institute as researcher between 1981 and 1985. After receiving the Ph.D. degree, he joined the Department of Information Engineering of Korea University, Korea as an assistant professor. Now he is an associate professor in the Department of Electric and Information Engineering of Korea University, Korea. His research interests include neural network, image processing, pattern matching and cognitive science.

About the Author—SEONG-WHAN LEE received his B.S. degree in Computer Science and Statistics from Seoul National University, Seoul, Korea, in 1984; and M.S. and Ph.D. degrees in Computer Science from KAIST in 1986 and 1989, respectively.

From February 1989 to February 1995, he was an Assistant Professor in the Department of Computer Science at Chungbuk National University, Cheongju, Korea. In March 1995, he joined the faculty of the Department of Computer Science and Engineering at Korea University, Seoul, Korea, as an Associate Professor, and now he is a Full Professor. Currently, Dr. Lee is the director of National Creative Research Initiative Center for Artificial Vision Research (CAVR) supported by the Korean Ministry of Science and Technology.

Dr. Lee was the winner of the Annual Best Paper Award of the Korea Information Science Society in 1986. He obtained the First Outstanding Young Researcher Award at the 2nd International Conference on Document Analysis and Recognition in 1993, and the First Distinguished Research Professor Award from Chungbuk National University in 1994. He also obtained the Outstanding Research Award from the Korea Information Science Society in 1996.

He has been the Co-Editor-in-Chief of the International Journal on Document Analysis and Recognition since 1998 and the Associate Editor of the Pattern Recognition Journal, the International Journal of Pattern Recognition and Artificial Intelligence, and the International Journal of Computer Processing of Oriental Languages since 1997.

He was the Program Co-Chair of the 6th International Workshop on Frontiers in Handwriting Recognition, the 2nd International Conference on Multimodal Interface, the 17th International Conference on Computer Processing of Oriental Languages, the 5th International Conference on Document Analysis and Recognition, and the 7th International Conference on Neural Information Processing. He was the Workshop Co-Chair of the 3rd International Workshop on Document Analysis Systems and the 1st IEEE International Workshop on Biologically Motivated Computer Vision. He served on the program committees of several well-known international conferences.

He is a fellow of International Association for Pattern Recognition, a senior member of the IEEE Computer Society and a life member of the Korea Information Science Society, the International Neural Network Society, and the Oriental Languages Computer Society.

His research interests include pattern recognition, computer vision and neural networks. He has more than 200 publications on these areas in international journals and conference proceedings, and authored five books.