目次情報を用いた書籍の文書画像構造解析

林 俊成[†] 成田 誠之助[††]

これまで，文字認識などのメディア変換や文書画像のレイアウト解析を中心に多くの文書画像処理技術の検討が行われてきた．一方，図書館の蔵書をハイパーテキスト化する場合，文書画像のレイアウト解析だけではなく，文書の論理構造分析にも注目すべきである．書籍の場合，目次は書籍の文書論理構造を最も忠実かつ簡潔に表しているものであり，この論理構造をベースに書籍の本文を分析した方が効率的であると考えられる．本論文は，目次情報を利用して文書画像を電子的な文書へ変換する新しい文書構造解析手法を提案する．これまで行われてきた手法は，事前に細部にわたり定義されたレイアウトモデルもしくはキーワードとのマッチングにより文書構造理解を行うものであった．これらの手法で問題となっているのは識別率の高いモデル作成法およびモデル構築の負担である．そこで本論文ではこれらの問題を解決するため，書籍それぞれの文書構造を簡潔かつ的確に表現している目次情報からモデルを作成し，これと文書画像とのマッチング処理により構造化処理を行う．これによりモデル作成の負担を軽減でき，かつ個々に付属した目次情報を利用するため汎用性も向上させることができる．実験の結果，書籍の論理構造である章節構造 99%，見出しセッション 94%，ヘッダ・フッタ・ページ番号構造 100%など，高い識別率を得ることができた．最後に，本方式に基づいて，解析した文書画像を HTML に変換する事例も紹介する．

Logical Structure Analysis of Book Document Image Using Contents Information

ChunChen Lin[†] and Seinosuke Narita[††]

Numerous studies have so far been carried out extensively for the analysis of document image structure with particular emphasis placed on media conversion and layout analysis. For the conversion of a collection of books in a library to the form of hypertext documents, the logical structure extraction technology is indispensable in addition to document layout analysis. The table contents of a book generally involves very concise and faithful information to represent the logical structure of the entire book document. That is to say, we can efficiently analyze the logical structure of a book by making full use of its contents pages. This paper is intended to propose a new approach for document logical structure analysis to convert document images and contents information into an electronic document. First, the contents page of a book are analyzed to acquire overall document logical structure. Thereafter, we are able to use this information to acquire the logical structure of the whole pages of the book by analyzing consecutive pages of a portion of the book. The test results demonstrate very high discrimination rates: up to 94% for the headline structure, 99% for chapter number, 100% for the head-foot structure.

1. はじめに

パーソナルコンピュータ，ワープロなどの普及にともない SGML，ODA，HTML などの電子化された文書を扱う環境が普及しつつあり，電子化文書の編集の容易さ，管理のしやすさ，電子媒体を扱うことによる簡便さなどから電子出版などもさかんになってきている．この利点を活かした電子化文書を新たに作成する場合は，その過程で適宜 SGML や ODA，HTML などの標準マークアップ言語を使用すればよい．しかし，新たに電子化文書を製作するには人的資源が圧倒的に不足する一方で，既存の印刷文書を効率良く電子化して再利用したいという要求も高い．特に膨大な蔵書を持つ図書館の場合，書籍を電子化することによって，効率良く蔵書を検索したり，相互に参照したりすることができる．さらに，書籍の文字データだけではなく図形，画像といったマルチメディア情報全体を計算機内で効率的に蓄積することも望まれている．

これを実現する技術が文書画像処理である．この文

† 東京外国語大学外国語学部
Faculty of Foreign Studies, Tokyo University of Foreign Studies
†† 早稲田大学理工学部
School of Science and Engineering, Waseda University

書画像処理は，単なる文字認識，領域の識別だけでなく，スキャナから読み込んだイメージから各種の情報，たとえば属性（表紙，目次，図など），章節，ページ番号を獲得し，文書の論理的構造まで認識し，効率良く文書を蓄積する処理技術である．文書画像処理手法に関しては，文字認識や図形認識などのメディア変換や文書画像のレイアウト構造解析を中心としてさかんに研究が行われている．文書画像のレイアウト構造解析の研究としては，レイアウトモデルから解析するもの[1)~5)]，キーワードから解析するもの[6)]などのモデルベースのもの，周辺分布特徴から解析するもの[7)]などがある．このようにレイアウトもしくはテキストから作成されたモデルとのマッチング手法による解析が主流であり，良い結果が得られている．このマッチング手法ではモデルの作成およびマッチングの方法が重要となってくる．このモデルは，作成の容易さ，記述性，汎用性，認識率といった点から評価されている．マッチング手法では，いかに認識率を落とさないようにするかということが重要である．

一方，論理構造ベースの検索を行うアプリケーションを想定した場合，文書画像のレイアウト解析よりも，文書全体の論理構造抽出に重点をおくべきである．これまで解析対象としては名刺，技術文書が主であり，連続するページ画像を持つ書籍の構造理解の研究はまだ十分になされていない．そうした中，土井ら[6)]は，技術文書 15,000 件，ビジネス文書 500 件から構造抽出規則を導き出し構造解析を行っている．この抽出規則はまず膨大な量のキーワード辞書を作成し，次にタイトル，著者，見出し，本文など抽出したい要素の名詞，固有名詞などと点，ハイフン，スペースの組合せパターンをすべて抽出してモデルを作成する．そして，実際の解析では解析したい文書から文字列を抽出し，膨大な量のキーワード辞書と照合することにより品詞その他を識別し，それらのパターンから文書種別，文書構造を解析している．しかし，こうした方法ではデータベース作成に多大な労力を必要としてしまう．

また，山田ら[8),9)]は土井らの手法を進め，連続する複数ページの文書画像を多段の章・節・段落構造を持つ論理構造化文書へと変換する方式について提案した．この手法では，ヘッダ・フッタ解析は各ページを文字認識した結果からページの初めと終わりの数文字を取り出し，あらかじめ用意しておいたページ番号のパターンとのマッチングによりページ番号を識別する．さらにヘッダ・フッタ識別では，ヘッダ・フッタがある範囲内では共通である性質を利用し，各ページの初めと終わりから数文字を取り出し，各ページに共通な文字列を見つけ識別を行っている．章節の見出し識別は，ページ画像内の矩形を矩形密度ホワイトスペースなどからブロック（カラム）に分け，各ブロックの初めの数文字を取り出し，連続した節番号の解析により，文書の論理構造を取り出している．確かにこの手法は，各論文誌や研究会予稿など種類ごとに同じフォーマットを持つものに対して，ブロックを用いたレイアウト解析および，章節番号解析を行う際に有効である．しかし，書籍解析に利用する場合，

(1) ブロック分けに用いたセパレーションの閾値は，書籍ごとに決める必要がある，
(2) 書籍によって，節番号がないあるいは連続していない場合，対応することができない，

など，多様なレイアウトを持つ書籍に対応することが難しいと考えられる．

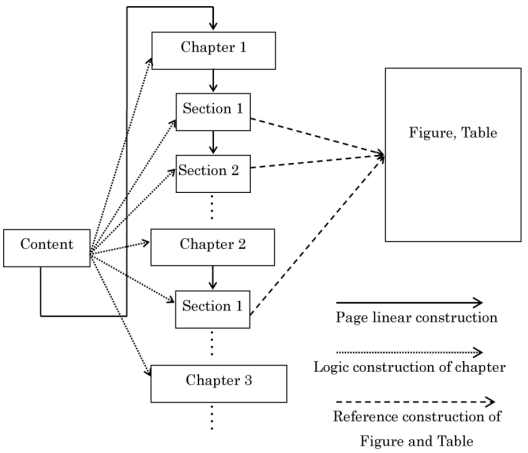

図 1 書籍論理構造
Fig. 1 The logic construction of book.

書籍の文書構造は，**図 1** に示すように，章節の論理構造，ページのような線形構造と本文から図表への参照構造からなる．目次は，書籍の文書論理構造を最も忠実かつ簡潔に表しているものであり，これを解析することによって，書籍の論理構造が得られる．しかも，ほとんどの目次は図表などの情報を含んでいないので，領域分割における解析をしやすく，この論理構造をベースに，本文文書画像を解析した方がより効率的な解析を行うことができると考えられる．本論文は，目次情報を利用した文書画像から電子的な文書へ変換する新しい文書論理構造解析手法を提案する．まず，目次ページを用いて見出し，見出し番号および各見出しのページ番号を分析し文書の論理構造を獲得する．これをベースに連続した複数の本文ページを解析し，文書の論理構造を取り出す．

以下，本論文では，2 章でシステム全体の処理概要，

3 章で目次ページ解析，4 章で本文レイアウト解析とマッチング処理，5 章で実験および実験結果について述べる．

2. 処理概要および基本処理

2.1 処 理 概 要

処理概要を**図 2** に示す．与えられた文書をスキャンして得た多ページにわたる画像のうち，目次画像と本文画像はあらかじめ指定する．そして，基本矩形・属性抽出処理で，画像から解析の基本単位となる基本矩形を抽出し，その属性抽出処理を行う．

次に目次ページ解析，本文ページ解析を行い，各要素に識別された構造化文書が出力される．目次ページ解析部では，書籍全体の論理構造を把握し，書籍の論理構造モデルを作成する．本文ページ解析部では，与えられた本文ページ画像で，本文や図表，ページ番号を識別する．次に目次画像から得られた論理構造にマッチングさせ，書籍全体の構造を識別し，最終的には HTML 形式で出力する．このように，構造解析は目次ページ解析，本文レイアウト解析，マッチング処理の 3 つに分けられる．

2.2 基本矩形の抽出（Basic Rectangle Extraction）

本システムでは全ページの共通処理として，見開き 2 ページを 1 画像として入力を行い，雑音除去処理などの前処理を行った後に，右 90 度に回転し，左右見開きのページにする．基本矩形抽出については次のように輪郭線追跡，外接矩形抽出，マージ処理の順で行う．

(1) ［輪郭線追跡］画像を走査し黒画素を見つけ，3×3 画素のマスクを使って，8 連結の輪郭線を検出し，その領域に接する最小矩形（外接矩形）を求める．

(2) ［外接矩形抽出］次に文字となる矩形を抽出するため，1 で抽出した矩形を重なりあっているものの統合を行う（**図 3** b）．

(3) ［マージ処理］さらに右射影方向にある閾値 N 以下で隣接する外接矩形の統合を行い，文字列矩形の抽出を行う（図 3 c）．

抽出された矩形を用いて，Y 方向のヒストグラムをとり，左右ページの矩形が存在する最大領域を認識し，X 方向の中心線から両側 N ポイントを書籍のとめ部分の雑音として除去する．

2.3 基本矩形属性

抽出された矩形から

- レイアウト属性
- テキスト属性

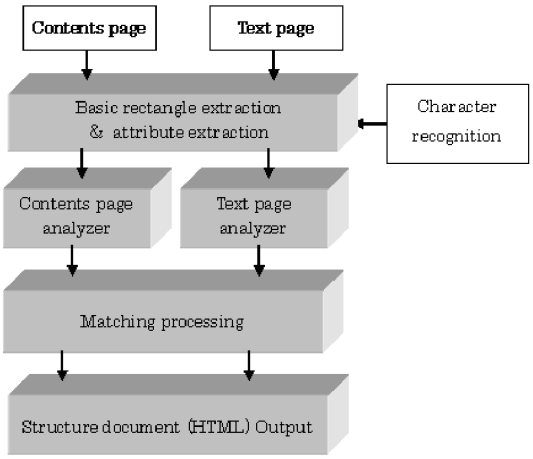

図 2 処理概要
Fig. 2 Outline of the processing.

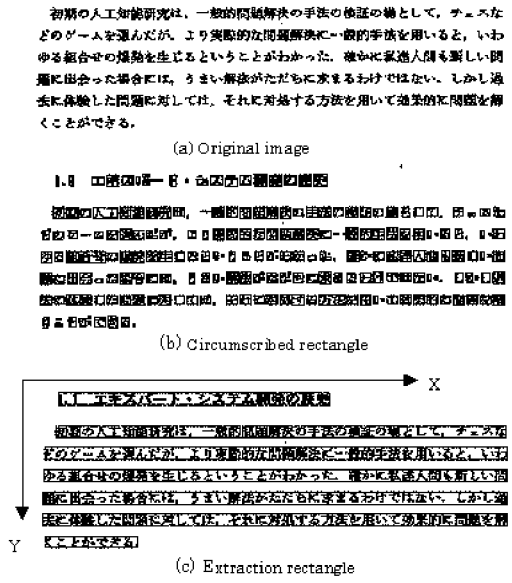

図 3 基本矩形抽出
Fig. 3 Basic rectangle extraction.

を得る．レイアウト属性は，**図 4** に示すように，個々の基本矩形の座標を (X_n0, Y_n0, X_n1, Y_n1) で表記することとし，**表 1** に示すように，幅（Width），高さ（Height），揃え（Indent），上下の行間（BtwnUp, BtwnDown）を検出する．揃えを抽出する際，画像の絶対座標では入力時の画像の置き方による誤差が大きいため，左右ページごとに全基本矩形が収まる外接矩形を求め，この座標を $(X0, Y0, X1, Y1)$ とし，各基本矩形の X_n0 から $X0$ までの変位を揃えとする．

テキスト属性は，抽出された基本矩形ごとに文字認識を行い，文字コードとして保存しておく．その際，

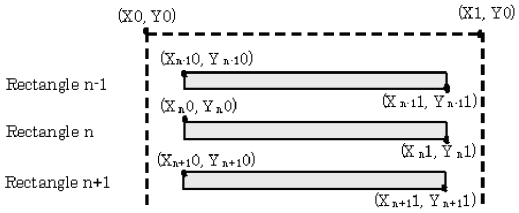

図 4 レイアウト属性
Fig. 4 Layout attributes.

表 1 基本矩形属性
Table 1 Basic rectangle attributes.

属性	項目	式
レイアウト属性	Height	Y_n1 - Y_n0
	Width	X_n1 - X_n0
	Indent	X_n1 - $X0$
	BtwnUp	Y_n0 - $Y_{n-1}1$
	BtwnDown	$Y_{n+1}0$ - Y_n1
テキスト属性	文字コード	
	文字種（数字，記号，文字）	

文字種（数字以外の文字，数字，記号）の識別も行う．

3. 目次ページ解析

3.1 処 理 概 要

目次ページ解析部では章節番号，見出し，ページ番号を抽出し書籍の階層構造を把握して，論理構造モデルを作成する．**図 5** に目次解析部の流れを示す．2 章で述べた基本矩形・属性抽出で得られた矩形から各行単位に文字種の組合せによってパターンを分類し，章節番号，見出し，ページ番号を識別する．その後，章節番号の連続性に矛盾がないか調べ補正を行う．

なお，前処理として，以下の条件に合致する矩形を見出しとページ番号間のセパレータとして除去する．

- X 方向に領域が N（本論文では 4 とした）ドット以下で連続している矩形群（点線）
- N（4 とした）ドット以下の高さで M 以上の長さを持つ矩形（直線）

3.2 章節番号識別

章節番号は行頭にあるものとし，各基本矩形のテキスト属性を用いて，先頭の数文字で以下の条件に一致するものを章節番号と識別する．

- `<Sec-Num> + [ <Separator> + <Sec-Num> ]`
  `<Sec-Num>`は正数，`<Separator>`は（. ,- ···）．また`[]` は繰返しを示す．
- `<Prefix-wd> + <Sec-Num> + <Postfix-wd>`
  `<Prefix-wd>`: 第，chapter，section，···（なし）
  `<Postfix-wd>`: 章，節，項 ···（なし）

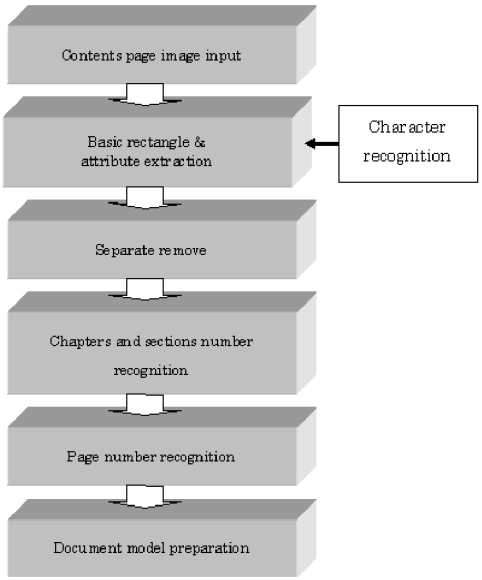

図 5 目次ページ解析処理概要
Fig. 5 Outline of analyzing process of contents pages.

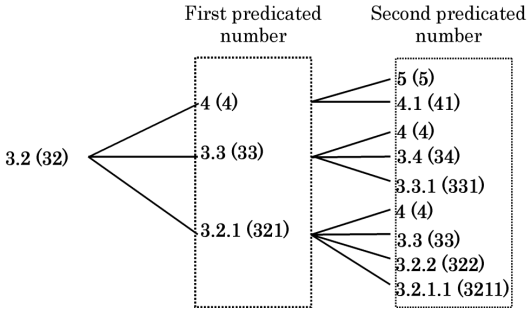

図 6 一次導出および二次導出番号
Fig. 6 First and second predicated number.

その後，文字の誤認識および見出しに数字が含まれている可能性があるので，全体から矛盾が生じないようにその補正を行う．一般に m レベルの階層にある節番号に対して，次に予想される節番号としては m+1 個の節番号が考えられる．ここではこれらを一次導出節番号といい，一次導出節番号から予想される節番号を二次導出節番号という（**図 6**）．また，文字認識による節番号の乱れは，散発的に起きることを想定して処理を行った．

(1) 各矩形より節番号開始パターンを検出し，解析の始点（SN_0）とする．
(2) 確定済みの最新の節番号部を SN_i，次候補を SN_{i+1}，次々候補を SN_{i+2} とする．
(3) SN_{i+1} が，SN_i の一次導出節番号のいずれかとマッチングが成功する場合，SN_{i+1} を最新の節番号と見なして (2) を繰り返す．

(4) SN_{i+1} が，SN_i の一次導出節番号のいずれともマッチングが成功せず，予測と違う場合，SN_{i+2} が二次導出節番号と比較し，マッチングが成功すれば SN_{i+2} を最新の節番号を見なして (2) へ戻る．このとき SN_{i+1} の訂正も行う．

(5) いずれの処理も失敗の場合 SN_{i+1} を最新の節番号と見なして (1) へ戻って，最後まで処理を続ける．

なお，本処理において，たとえば，“参考文献”，“索引”，あるいは，

1.2 UNIX の特徴.....22
・ 標準オペレーティング・システム.....22
・ ソースコードの配布.....23
… 中略 …
1.3 UNIX の応用分野.....35

のような節番号のない見出しの場合，処理を行わないが，4.5 節で述べる処理では，マッチングを行う．また，本処理方法は，章節番号の数字間のセパレータを無視しているので，山田ら[9]が指摘している “.” の誤認識，未検出処理の失敗をなくすことが可能である．

3.3 ページ番号および見出し識別

ページ番号は行末にあるものとし，各行末からある N 文字以下の数字列をページ番号として識別する．ページ番号の場合，章節番号のように，数字が連続するなどの規則がなく，数字が増加していくだけである．ここでは，文字の認識誤りなどによるエラーは，4.6 節で述べる見出しマッチング処理を行う際，上下の幅を持たせて検索を行い，修正を行う．

章節番号およびページ番号を抽出した後，残った矩形を見出しとして認識する．

3.4 見出し構造

各行をその要素の組合せによって次のいずれかに属するパターンとして分類する．

- 章節番号+見出し
- 見出し+ページ番号
- 章節番号+見出し+ページ番号

以上の解析により，章節番号がある見出し解析によって得られた階層レベルの見出しとし，**図 7** のような文書の見出し構造を得ることができる．なお，章節番号がないものは直前の見出しの階層と同じ階層とし，ページ番号のないものは，次の行のページ番号をこの行のページ番号とする．章節番号およびページ番号両方ともない行（たとえば，“目次” 行など）あるいは，数字だけの行（目次のページ番号など）は処理されない．また，本処理を終了した後，見出し構造リスト $HSList_i$ が生成され，各リストは

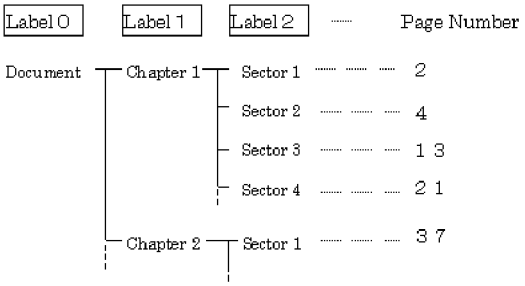

図 7 文書構造
Fig. 7 Document construction.

章節番号+見出し+ページ番号
あるいは
見出し+ページ番号
となる．

4. 本文レイアウト解析処理およびマッチング処理

4.1 目次情報を用いた本文解析の特徴およびその処理概要

1 章に述べたように，これまで文書画像処理が主に本文レイアウトもしくはテキストから作成されたモデルとのマッチングによる手法である．これらは，最初から決められた文書を解析したうえ，レイアウトベースかテキストベースかのモデルを作成し，いわゆるトップタウン方式で，名刺や技術文書，論文誌など同じフォーマットの文書が大量に存在する場合，非常に有効である．しかし，書籍の場合各書籍 1 冊ずつの行間，見出し・本文の文字サイズなどが違うため，全書籍に合うモデルを作成するには非常に困難である．一方，書籍は 3 章に述べたように，本文ページよりも構造が簡単な目次から書籍の論理構造が得られる．今までの手法では，本文ページの論理構造をいかに取り出すかに対し，本手法は，事前に書籍の論理構造を把握しているため，本文ページを解析する際，論理情報ではなく，いかに見出しではないものを取り除くかに重点を置いている．しかも，条件設定を厳しくし，見出しが除去されないようにし，その後，目次情報と本文情報をマッチングすることにより，見出し情報を取り出すことにした．その場合，作成されたモデルが簡単であり，より多くの文書に対応できることが利点である．

書籍の本文ページは，大別して見出し領域，本文領域，図形・写真領域，ヘッダ・フッタ領域，ページ番号領域から構成される．これらの領域は，罫線（Field Separator），または空白領域によって分類される．**図 8**

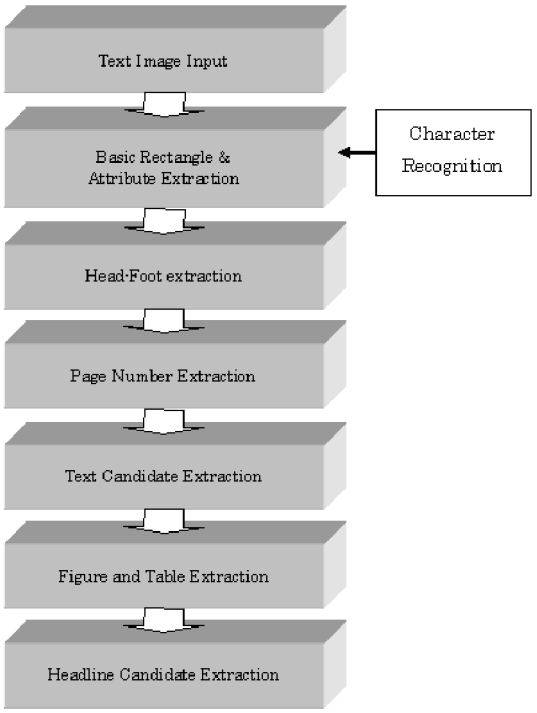

図 8 本文ページレイアウト解析処理概要
Fig. 8 Outline of analyzing process of text page layout.

に示すように，処理は，まず画像をスキャナから入力し，2 章で述べた基本矩形および矩形属性を抽出した後，各基本矩形は，高さが N 以下，幅が M 以上の場合，セパレータとして取り除く．また，全基本矩形の Y_n0 座標および高さが同じあるいは近い場合，1 つの基本矩形に統合する．上記の処理を終えた後に，ページごとに各基本矩形の Y_n0 順で各ページの本文リスト $PTList_i$ を作成する．各 $PTList_i$ のレイアウト属性およびテキスト属性を用いて，以下の処理フローに従って識別を行う．

4.2 ヘッダ・フッタの抽出

一般的に書籍の本文ページは，同一書籍であれば基本的にレイアウト構造が同じであり，**図 9** に示すように，つまり，上下左右の余白，ヘッダ・フッタ領域，ページ番号領域，本文領域は同じように配置されている．ここではこの性質を利用して，矩形領域からヘッダ・フッタ領域を判定する．ここで，ヘッダ・フッタは

- 存在する場合，ページ内の最上行および最下行にある，
- 同じポイント数で表記されている，
- 左右ページ x 方向について外側からの変位が等しい，

とする．

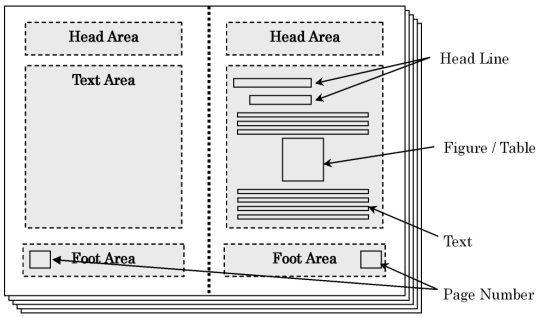

図 9 本文ページレイアウト構造
Fig. 9 Layout construction of text pages.

以上の特徴を用いて，次のような処理により，ヘッダ・フッタを抽出する．まず，各ページの $PTList_i$ の最上行，最下行の基本矩形群をヘッダ，フッタ候補としそれぞれ Hdc_i，Ftc_i とする．Hdc_i，Ftc_i に対して，基本矩形領域の外側からの変位（左ページの場合：X_a0-$X0$，右ページの場合：$X1$-X_a1）と矩形の幅 Y_a1-Y_a0 を調べ，グループ分けする．それぞれが，全ページ数の 80%以上存在すればそのヘッダ・フッタ矩形候補グループをヘッダ・フッタとして認識する．ヘッダ・フッタが存在しないページに対し空白を挿入して，それぞれヘッダグループ Hdg_i，フッタグループ Ftg_i とする．なお，現段階では，ページ番号も含まれている．

4.3 ページ番号の抽出

ページ番号はヘッダ・フッタと認識された矩形グループ Hdg_i，Ftg_i のテキスト属性を利用し，最も外側（とめ部分と逆側）の数字を抽出して，ページ番号候補群 PNc_i とする．そして，次のように最初のページのページ番号初期値を検出する．

(1) 添え字 i=1 とし，ページ番号初期値 PN_i に 1 を代入する．
(2) もし PN_i と PNc_i の数字が一致したらカウンタを 1 増加する．
(3) 添え字 i が N になるまで，PN_i を 1 増加し 2 に戻る．

このように（1）のページ番号初期値 PN_i に 1 から 10 を代入し，10 回計算し最もカウンタの数値が高いものをページ番号初期値とし，このページ番号初期値に沿って，全ページにページ番号をつける．

以上の処理を終了した後，ヘッダ・フッタグループ Hdg_i，Ftg_i の基本矩形を各ページの $PTList_i$ から除去する．

4.4 本文矩形識別

本研究では 1 段組を対象とし，書籍中に本文矩形が最も多く含まれているものとする．全 $PTList_i$ の

レイアウト属性の Height，BtwnUP，BtwnDown を用いてグループ分けし，最頻値を Text-Indent，Text-Height，Text-BtwnUp，Text-BtwnDown とする．次に全 PTList を対象に以下の条件により，本文領域を抽出する．

(1) 各基本矩形のレイアウト属性の`Indent<=Text-Indent` かつ `Height<=Text-Height` かつ `BtwnUP<=Text-BtwnUP` かつ `BtwnDown<=Text-BtwnDown`

(2) 各ページ先頭の基本矩形である場合，BtwnUP の条件を無視する．

(3) 各ページ最後の基本矩形である場合，Btwn-Down の条件を無視する．

以上の処理を終了した後，$PTList_i$ から本文領域と識別されるものを取り除く．なお，本処理は厳しい条件で本文領域を識別することにより，多数の本文を残してしまうが，マッチング処理により実際に見出しを識別することにした．

4.5 図表の識別

ここで，各ページの $PTList_i$ の高さが前節に求められた Text-Height の N 倍（本論文では N=4 とした）以上である矩形を図表候補とし，$PTList_i$ から削除した後，残ったすべての基本矩形を見出し候補 Hlc_i とする．

4.6 マッチング処理

マッチング処理では，目次解析部で得た論理構造と本文解析部の結果をマッチングし，書籍本文全体の最終的な構造化処理を行う．目次から作成された $HSList_i$ から 1 セットを抜き出し，章節番号と見出しの文字列を用いて，そのページ番号に示す本文ページの全 Hlc_i とマッチングを行う．マッチングに関しては，DP マッチングを用いて共通部分の文字を抽出して文字数を計算し，カウントされた文字数に両文字列の長い方の文字数で割って，一致率を計算する．一致率の計算式は，以下のとおりである．

$$rate(n) = \frac{\text{マッチングされた文字数}}{\text{長い文字列の方の文字数}} \qquad (1)$$

以上の処理で，最も $rate(n)$ が高かった Hlc を HSList の見出しとして識別する．ここで，目次解析でページ番号の誤認識が存在する場合を考え，一致率が N より低い場合（本論文では N=0.8 とする）は，$HSList_{i-1}$ に示したページ番号から $HSList_{i+1}$ に示したページ番号までの本文ページのすべての Hlc_i とマッチングさせ，最も $rate(n)$ が高い Hlc を見出しとして識別し，0.5 以下である場合，検出失敗とする．以上のように，全 $HSList_i$ に対して，マッチング処理を行い，Hlc_i から見出しを検出し，処理終了後残った Hlc_i を全部本文候補にする．

4.7 構造化文書 HTML 形式での出力

すでに述べたように構造化文書を記述する方法として ODA，SGML などがあるが，比較的タグ付けが容易で，実用的な面を考慮して HTML 形式で出力する．構造化文書のタグ付け規則は以下のとおり．

(1) 目次ページ

- ファイル名 filename，ページ番号 PN，レベル N の見出し Headline:
  ```
  <LI>
  <A HREF="(filename)-(PN).htm">
  <H(N+2)>(Headline)</H(N+2)></A>
  ```
  すべての見出しは`<LI>`タグが付けられ，レベルごとに`<UL> </UL>`タグでまとめられ，レベルが 1 下がるごとに入れ子を作成する．
- 見出しとして認識されなかった文字列：
  `<--!(文字列)-->`（コメントアウト）

(2) 本文ページ

- ページ番号左：
  ```
  <H1 ALIIGN = "LEFT">(PN)</H1>
  ```
- ページ番号右：
  ```
  <H1 ALIIGN = "RIGHT">(PN)</H1>
  ```
- レベル N の見出し Headline：
  ```
  <H(N)>(Headline)</H(N)>
  ```
- 段落：
  ```
  <P>
  ```
- 図表（1/4 に縮小）：
  ```
  <CENTER><IMG SRC="(filename)"
   VSPACE="30"></CENTER>
  ```

5. 実験結果および考察

5.1 実 験 概 要

本手法の有効性を示すため PC 上で処理の実行部を Microsoft Visual C++，GUI 部を Microsoft Visual Basic で実装し，以下のような実験を行った．3.4 節で述べた条件に適合する書籍 12 種類につき，2 章節階層のもの（A〜E）を 5 種類，3 章節階層のもの（F〜J）を 5 種類，および，章ごとに節番号が 1 からカウントされるもの（K），節番号が 1 冊を通じて加算されていくもの（L）など章節番号が連続されていないものを 2 種類用意する．1 種類ごとに目次ページを全文，本文ページを 41 ページまで読み込み，目次ページ全 59 ページ，本文ページ全 492 ページ，総数 551 ページで実験を行った．目次ページと本文ページを別々に解像度各 400 dpi/inch，しきい値 150 の 2 値化画像で

表 2 目次解析結果
Table 2 Results of contents page analysis.

		Total Item	Item	Include Mis-recognition	Recovered Chapter Num.
A	Head line	78	73/73	—	—
	Chapter Num.		71/71	1	1
	Page Num.		62/65	3	—
B	Head line	88	84/84	—	—
	Chapter Num.		81/81	2	2
	Page Num.		78/84	6	—
C	Head line	55	51/51	—	—
	Chapter Num.		47/47	1	1
	Page Num.		51/51	0	—
D	Head line	83	81/81	—	—
	Chapter Num.		79/79	1	1
	Page Num.		79/81	2	—
E	Head line	74	71/71	—	—
	Chapter Num.		59/59	4	4
	Page Num.		61/69	8	—
F	Head line	146	140/140	—	—
	Chapter Num.		133/133	7	7
	Page Num.		90/128	38	—
G	Head line	221	206/206	—	—
	Chapter Num.		126/126	1	1
	Page Num.		194/196	2	—
H	Head line	160	150/150	—	—
	Chapter Num.		149/150	10	9
	Page Num.		120/128	8	—
I	Head line	177	172/172	—	—
	Chapter Num.		164/165	12	11
	Page Num.		138/176	38	—
J	Head line	132	124/124	—	—
	Chapter Num.		122/124	11	9
	Page Num.		101/114	13	–
K	Head line	43	37/37	—	—
	Chapter Num.		—/37	2	—
	Page Num.		34/37	3	—
L	Head line	82	81/81	—	—
	Chapter Num.		—/81	4	—
	Page Num.		78/81	3	—
Total	Head line [%]	1467	100.0 (1270/1270)	—	—
	Chapter Num. [%]		99.6 (1031/1035)	56	46
	Page Num. [%]		89.2 (974/1092)	128	—

入力した．また，文字認識に関しては，日立マイコン社の MY-QREADER Pro を利用した．

5.2 目次解析実験結果

12 冊全部の目次解析の結果を**表 2** にまとめた．全項目数欄に示した数字は，目次に含まれた全項目の数であり，ヘッドラインの項目欄は，3.4 節で述べた処理されない項目を引いた数である．章節番号の項目欄の後ろの数字は，章節番号が存在する項目数であり，前の数字は，3.2 節で述べた補正処理を行った後に正しく識別された項目数である．ページ番号項目欄の後ろの番号は，全ページ番号の項目数であり，前の番号は文字認識の際正しく認識された数字の項目の数である．また，“補正された節番号” は，節番号の文字認識誤りに対し，3.2 節に説明した方法で正しく補正された数を示してある．なお，K，L は，章節番号が連続していないため，補正処理することができない．そのため K，L を含まない章節番号の総識別率は，99.61%（1031/1035）である．

5.3 本文ページレイアウト解析結果

本文レイアウト解析の実験結果を，**表 3** にまとめた．測定項目は，本文，見出しページ番号，ヘッダ・フッタ，および図表である．各項目は基本矩形を単位とし，表の第 1 項目は，各属性（見出し，本文など）の数であり，第 2，3 項目は，本文レイアウト解析を行った後，各属性に振り分けた誤りの数である．ここでいう第 1 種誤りとは，本文でありながら見出しに分

表 3 本文ページ解析結果
Table 3 Result of page analysis.

		Item number	Type I Error	Type II Error
タイプ A〜E	Text	4432	429	65
	Headline	76	0	494
	Page number	193	0	0
	Head-foot	193	0	0
	Picture & Table	63	0	0
タイプ F〜J	Text	3808	450	85
	Headline	152	0	535
	Page number	194	0	0
	Head-foot	194	0	0
	Picture & Table	83	0	0
タイプ K, L	Text	1379	95	30
	Headline	54	0	158
	Page number	80	0	0
	Head-foot	80	0	0
	Picture & Table	66	3	0

表 4 実験結果
Table 4 Result of the experiment.

	identification rate [%]
Page Number	100.00
Head/Foot	100.00
Chapter Number (Excluding K, L)	99.61
Text Headline	94.63

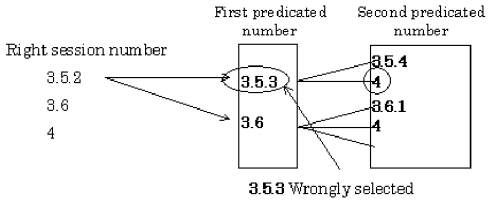

図 10 章節番号補正誤りの例
Fig. 10 An example of revision of session numbers.

類されたものを指し，第 2 種誤りとは，本文ではないのに，本文と識別されたものをさす．

5.4 マッチング処理の結果

マッチング処理を行った後の最終結果を，**表 4** にまとめてある．なお，本実験は図表のキャプションを識別していないため，これらのものが本文として識別される．

5.5 考　　察

5.5.1 目 次 解 析

章節番号解析においては，表 2 に示すように，一般的に，構造は K，L のような文書で書籍の論理構造が一番少なく，そして，文書 A〜E のような 2 章節階層数よりも文書 F〜J の 3 章節階層数の方が多い．章節番号の文字誤認識の数も，3 章節階層の方が多いが，単発的な文字の認識誤りでは，補正処理により，誤りが約 9 割以上吸収された．また，補正誤りの原因は，**図 10** に示すように，3.5.2 項→ 3.6 節→ 4 章に切り替えるとき，3.6 節に文字に誤認識が発生したが，3.5.3 項を選んでしまう選択エラーであり，もう 1 つは，連続した文字の認識誤りである．1 つ目の選択エラーに対しては，文字の認識誤りがあった数字を解析し，内容を参照しながら，節番号を正しく選択する必要がある．2 つ目の連続文字の認識誤りに対しては，より多くの候補（第三次導出番号など）を求め，その中から正しいものを選択するなど，アルゴリズム改良の余地があると思われる．

ページ番号について，第 F 番目と I 番目の文書の認識率が低い原因は，斜体文字が使われ，数字の 1 がほとんど 2 あるいは 7 と認識されてしまったためである．

5.5.2 本 文 解 析

この処理において，ヘッダ・フッタとページ番号の解析は成功したものの，本文を解析する際，本文でありながら見出しとして識別された第 1 種誤りが多く発生した．これらは，主に文書途中に挿入した図表に生じた行間の変化，あるいは，行に強調として使われるボールト字体による行高さの変化で生じたものである．見出し項目に本文あるいは，図表番号でありがなら，見出しとして識別された第 2 種認識誤りも発生する．本処理において，本文領域を抽出する際，識別条件を厳しくすることにより，多くの本文矩形が見出し候補になったが，これは，この段階で無理に識別せずにマッチング処理で確実に識別させるためである．識別条件を緩めてしまうと，見出しもこの段階で見逃してしまう恐れがあるため，テキストによるマッチング処理で最終的な見出し候補を識別する．

表 5 見出し識別実験結果および本文テキストを利用した場合の比較
Table 5 Result of headline distinction and comparison with text method.

	Using Content Method		Using Text Method	
Document Type	Type I Error	Type II Error	Type I Error	Type II Error
A	8.33%	0.00%	0.00%	16.67%
B	6.25%	0.00%	0.00%	0.00%
C	0.00%	0.00%	0.00%	0.00%
D	0.00%	0.00%	0.00%	6.25%
E	6.25%	0.00%	—	—
F	12.50%	0.00%	15.63%	21.88%
G	8.33%	0.00%	—	—
H	4.17%	0.00%	0.00%	16.67%
I	10.64%	0.00%	8.51%	27.66%
J	7.14%	0.00%	10.71%	42.86%
K	0.00%	0.00%	—	—
L	2.00%	0.00%	—	—
Average	5.47%	0.00%	4.36%	16.50%

5.5.3 マッチング処理

表 4 に示すように，マッチング処理を行った後，本文ページの 95%の見出しが識別された．また，5.5.1 項で述べたように，F と I 番目の文書のページ番号の誤認識が多い（1 がほとんど 2 や 7 に認識されてしまった）ものの，ページ番号が誤認識された見出しの多くは，ほとんど目次の $HSList_{i-1}$ のページ番号から $HSList_{i+1}$ のページ番号までに示した本文ページに存在したので，抽出することができた．また，抽出できない原因は，

(1) 短い見出し（たとえば，“概論”，“序説”）は，文字認識誤りにより一致率が低い，
(2) 目次ページ番号を認識する際，連続したページの認識誤りがある場合，本文見出しがスキャンしたページの中に存在しないことになってしまう，
(3) 見出しが 1 行だけであるのに，本文にある見出しが大きめなフォントを利用し，2 行になり，一致率が低くなってしまう，

である．(2) に関してはページ番号を無視し，全文マッチングを行う方法も考えられるが，各章に共通した “参考文献”，“この章のまとめ” などのような同じ見出しがあった場合，エラーの原因となる．また表 3 に示すように，本文の見出し候補数が多いので，処理時間がかかってしまう．文字認識率が解析の鍵であり，特に，数字の認識が非常に重要なポイントとなるので，目次解析をする際，章節番号，ページ番号をそれぞれ前もって位置を予想し，数字だけの文字認識機構で認識を行わせるなどの方法でアルゴリズムを改良する必要がある．

5.5.4 本文情報を用いた場合との比較

本手法と本文情報を用いて，見出し構造解析を行う場合と比較すると，**表 5** に示すように，本手法を用いた場合，第 1 種誤りが 5.47%，本文情報を用いた場合が 4.36%と若干高めだったが，第 2 種の誤りは発生しないのに対し，本文情報を用いた場合，16.5%である．本手法は論理構造を目次から取り出しているため，第 2 種の誤りが発生しないが，本文情報を用いた場合，番号の箇条書きが用いられ，しかも，各箇条のタイトルに大きめのフォントを利用している書籍の場合，ブロック構築の際に，別のブロックとして認識したため，連続した番号であるので，節番号解析部で第 2 種誤りが発生してしまう．また，タイプ K，L の場合，連続した節番号ではないので，解析することができないし，タイプ E，G の場合，章の最後に複数のページに渡って多数の参考文献があり，短い行による行間の変化で，節番号と思われた不連続の数字が発生し，節番号解析に失敗してしまう．

5.5.5 本手法の適用範囲

本手法は，目次の見出し情報を元にしているので，目次の各行が 3.4 節で述べた各見出しのパターンであれば，適用できると考えられる．しかし，以下のような文書に対しては，現在のところ対応することが難しいと考えられる．

(1) テキスト部分に複雑な図形飾りがついている場合．
(2) 目次において，2 行にわたった見出しがある場合．
(3) 目次において，1 行に 2 つの節番号がある場合．
(4) 本文ページ番号が連続していない場合．
(5) 論文，研究会予稿など，目次に章節番号が存在していないもの．

上記 (1) の場合，文字の誤認識によるマッチング処理ができない．(2) および (3) の場合は，1 つの見出し情報が 2 つの HSList になったり，2 つの見出し情

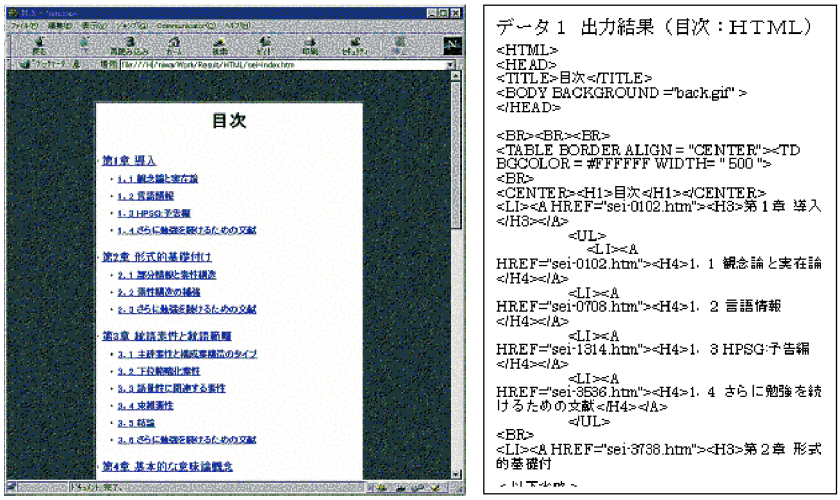

図 11 構造化文書 HTML の出力（目次）
Fig. 11 Constructed document HTML output (contents).

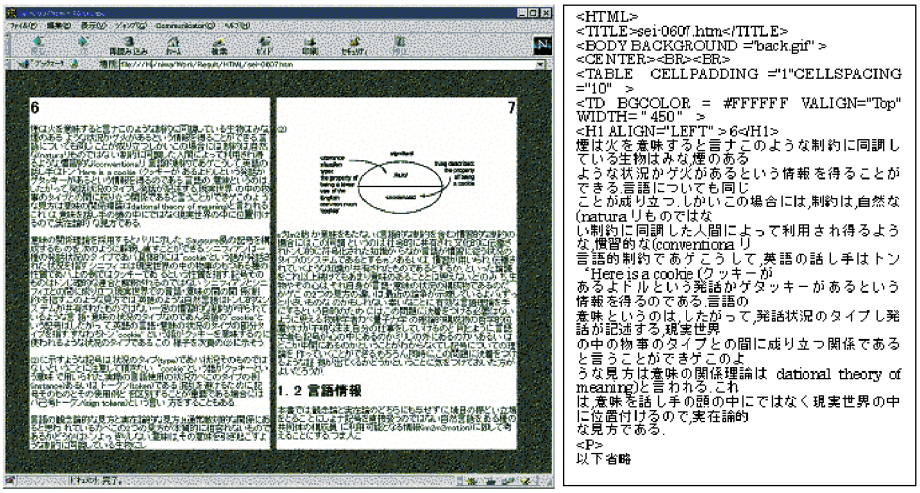

図 12 構造化文書 HTML の出力（本文）
Fig. 12 Constructed document HTML output (text).

報が 1 つの HSList になる．これらを解決するためには，目次画像情報を得る際にさらなる領域分割が必要である．(4) の場合は，ページ番号の解析ができないため，マッチング処理をすることができない．

5.6 HTML 文書

最後に，本手法により生成した HTML 文書を紹介する．本手法を用いることで，解析した文書を 4.7 節で説明した方法で**図 11**，**図 12** のような HTML 形式で表現することが可能であり，また，目次と本文の間にも，リンクを張ることが可能である．

6. ま と め

本論文は，書籍の論理構造を最も忠実かつ簡潔に表している目次を利用した，書籍の論理構造解析の手法について述べたものである．12 種類の書籍について実験を行い，かなり高い確率で識別することができた．今後の課題は数字の認識率を向上させるための方法，1 つの目次構造が 2 行である場合の対応などがあげられる．また，多様な書式を持つ書籍をすべて 100%の精度で解析することは難しいので，誤った識別を修正しやすくするためのユーザインタフェースの作成も重要なポイントとなる．

参 考 文 献

1) 山下晶夫，天野富夫：モデルに基づいた文書画像のレイアウト理解，電子情報通信学会論文誌，Vol.J75-D, No.10, pp.1673–1681 (1992).
2) 黄瀬浩一，馬場口登：レイアウトモデルに基づく文書構造解析，電子情報通信学会論文誌，Vol.J72-D, No.7, pp.1029–1039 (1993).
3) Floriana, E. and Donato, M.: A Knowledge-Based Approach to the Layout Analysis, *Proc. 4th Int. Conf. on Document Analysis and Recognition*, pp.466–471 (1995).
4) Liebowitz, S.T.: An Intelligent Document Understanding System, *Proc. 3th Int. Conf. on Document Analysis and Recognition*, pp.107–110 (1993).
5) Devashish, N.: Knowledge-Based Derivation of Document Logical Structure, *Proc. 4th Int. Conf. on Document Analysis and Recognition*, pp.472–475 (1995).
6) 土井美和子，福井美佳，山口浩司：文書構造抽出技法の開発，電子情報通信学会論文誌，Vol.J76-D, No.9, pp.2042–2052 (1993).
7) 秋山照雄，増田　功：周辺分布，線密度，外接矩形特徴を併用した文書画像の領域分割，電子情報通信学会論文誌，Vol.J69-D, No.8, pp.1187–1196 (1986).
8) 山田　満：文書画像の ODA 論理構造化文書への変換方式，電子情報通信学会論文誌，Vol.J76-D, No.11, pp.2274–2284 (1993).
9) 山田　満，宮里　勉，蓮池和夫：マルチメディア文書構造処理システム，画像電子学会誌，Vol.19, No.5, pp.286–295 (1990).

(平成 13 年 3 月 30 日受付)
(平成 14 年 9 月 5 日採録)

林　　俊成（正会員）

平成 5 年早稲田大学大学院修士課程修了．同年同大学院博士後期課程進学．平成 8 年同大学助手を経て，平成 10 年東京外国語大学に勤務．文字認識，文書画像構造解析，マルチメディア環境における学校教育，遠隔教育等の研究に従事．電子情報通信学会，教育工学会各会員．

成田誠之助

昭和 37 年早稲田大学大学院修士課程修了．昭和 37 年アメリカ・パデュー大学大学院留学（フルブライト留学）．昭和 38 年早稲田大学理工学部助手，以後，講師・助教授を経て，昭和 48 年教授，現在に至る．工学博士．分散計算機制御システム，並列処理，産業用ロボット制御，デジタル制御理論，CIM 等の研究に従事．現在は主として e-learning の研究を行っている．計測自動制御学会，電気学会，ロボット学会，IEEE 各会員．