定型化された一般文書画像からの指定要素抽出

A Layout-Free Method for Extracting Elements from Document Images

幸地　司*　　齋藤　高志*
Tsukasa KOCHI　　Takashi SAITOH

要　　旨

画像から得られるレイアウト特徴にもとづいた，定型化された書式を持つ一般文書からタイトルや著者などの書誌事項を抽出するための新しい方法を考案した．従来より文書の論理構造を認識して文書全体を構造化する試みがいくつか提案されているが，そのための変換コストや認識精度は十分なレベルとは言い難い．本論文では実際に必要な書誌事項だけを抽出する技術に着目し，そのために簡単に指定要素を定義できる方法と効果的なテンプレートマッチングの方法を提供する．多種多様な実データを用いた実験から，提案する手法は文書のレイアウト変動に柔軟に対応可能であり，さらに複数のサンプル文書を用いてテンプレートを更新することにより抽出精度は大幅に改善されることを示す．

ABSTRACT

An automatic document entry system is described which extracts textual information such as titles or authors from document images. The proposed system can analyze semi-formatted documents such as business letters, reports, or technical papers, which are quite different from checks or forms. Various attempts at generating SGML from a document image have not been success because of its complex construction of a model and a rule-base. We therefore focus on extracting some of the important layout elements by using easy operations of document-specific models. We also describe the learning method including a template modification and a multi-template strategy. Experimental results with various kinds of real data show the proposed method is able to withstand shift and noise of various kinds of documents.

* 画像システム事業本部 ソフトウェア技術開発センター ソフトウェア研究所
Software Research Center, Software Technology Development Center, Imaging System Business Group

1. 背景と目的

近年のインターネットやイントラネットの発展に伴って，文書の電子化や事務手続き等をすべてネットワーク化する動きが急速に加速しつつある．さらにはCD-R/WやDVD-ROM等大容量メディアの登場，およびメモリーや記憶装置の急激な価格低下などを背景としたコンピュータ周辺機器環境の進化によって，文書の電子化に対する考え方も大きく進歩してきた．

例えばJ.Hull[2]は，業務中に流通するあらゆる紙文書をコピー機を通してすべて画像としてデータベースに蓄積するようなシステムを紹介している．得られた画像はすべて文字認識処理されて，原画像あるいは縮小画像と共にユーザー毎に管理される仕組みである．ところが，従来の文字認識処理で生成された文書は単なるテキストの羅列であるので，元の紙文書が持つ論理的な高次元の情報を十分に活用することが出来なかった．このような状況下では，大量に処理される文書に対して人間の手で書誌事項を一つ一つ登録するのは非常に困難である．

これまで紙文書を自動的に電子化することは，文書の論理構造を認識して文書全体に章節構造や図表などを細かく定義したタグを埋め込むことを指すことが多かった．例えば，C.Wenzel[8]，M.Sharpe[9]，T.Watanabe[10]らは文書画像の要素間の関係と構造モデルとの整合性を調べ，該当する論理モデルの論理構造要素の属性をパラメータとして文書画像の要素の内容を認識する手法を提案している．ところがこれらの方法では，文書画像の要素間の関係を定義したモデルにおいて文書要素をノードおよび要素間の配置関係をリンクするようなグラフ構造を生成する必要があるので，モデルの作成や保守には非常に手間がかかる場合が多い．

多種多様な文書を扱うことが多いオフィスや電子図書館では，電子化の際に文書タイプごとに人手で詳細にモデルの設定を行うことは難しい．また前記に示したように従来技術の性能が不十分であることから，実際に文書を閲覧する段階では，未だに紙文書の優位性は揺るぎがたい．したがって，文書全体を詳細に構造化するよりも必要な書誌事項だけを正確に抽出するほうが得策だといえる．

本論文と同様に文書から必要な書誌事項だけを抽出するという観点からは，銀行チェックや各種申込書などを扱う定型帳票認識技術が挙げられる[5][6][7]．多くの研究機関でこの分野での研究開発を進めており，実用レベルのシステムもある．しかしながら，これら従来技術にも次のような致命的な問題が残されている．

・抽出項目が枠内に限定されている

・枠構造の変化や項目の位置変動に弱い

これらの欠点が多種多様な一般文書を電子化する際の大きな障害となっていた．上記の背景を踏まえて，本論文ではFig.1に示すような表形式ではない定型化された一般文書からタイトルや著者名など実際に必要となる書誌事項だけを抽出する技術に関し，従来技術では対処しきれなかった下記課題の解決を目標とする．

・ノイズやレイアウト変動に柔軟に対応できる

・多種多様な文書に容易に適応可能である

本論文は次のように構成される．第2章で提案するアルゴリズムとテンプレートについて述べ，第3章では多種多様な実データを用いた実験を通して，提案する手法が文書のレイアウト変動に対して柔軟に対応可能であることを示す．第4章は結びである．

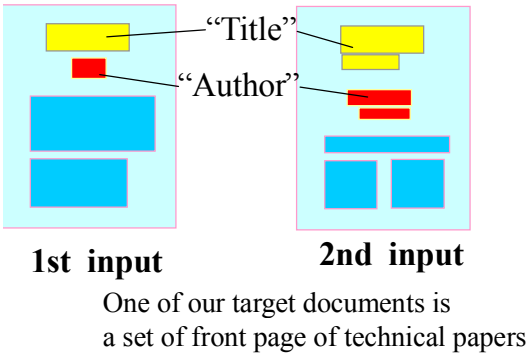

Fig.1　Examples of target documents.

2. 技術

本論文では定型化された書式を持つ一般文書からタイトルや著者などの書誌事項を精度良く抽出するための新しい方法を提案する．提案手法を搭載した文書登録システムの構成の一例をFig.2に示す．本方式は大きく2つの構成からなる．1つがFig.2の右側に示されたテンプレート作成・登録部，もう一方が左側に示された指定要素抽出部である．双方に共通するモジュールとして入力文書から必要なレイアウト特徴を

抽出するレイアウト特徴抽出部がある．各モジュールを簡単に紹介する．

1. レイアウト特徴抽出部
   文書をデジタル画像として入力して，領域識別，文字認識，フォント識別などの処理を経て前記文書のレイアウト特徴を抽出して次期モジュールに渡す．
2. テンプレート作成・登録部
   テンプレート用サンプル文書の上からGUI(Graphical User Interface)を用いて指定要素を設定してテンプレートを作成する．
3. 指定要素抽出部
   はじめに入力文書の種類を識別して最適なテンプレートを決定する．つづいて決定されたテンプレートを参照しながら指定要素を抽出して結果をHTML(Hyper Text Markup Language)形式などで保存する．

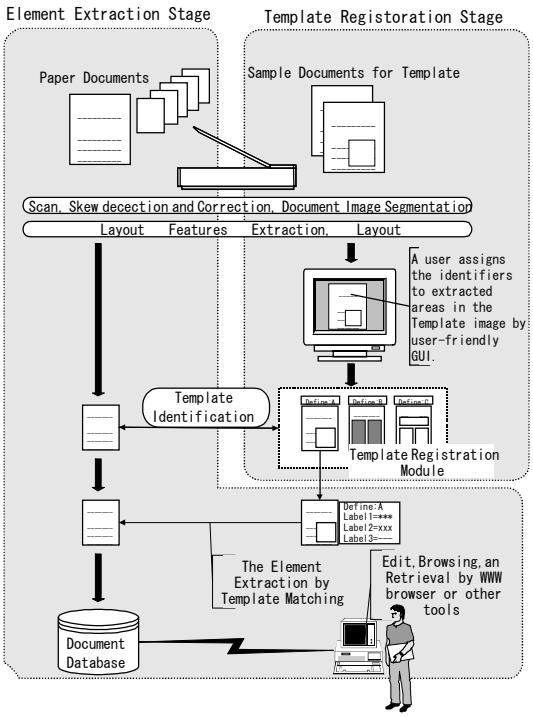

Fig.2　Overview of the system operation.

2-1　文書レイアウト特徴抽出

ここでは紙文書から提案手法で必要な文書レイアウト特徴を抽出する方法を説明する．抽出された特徴は文書特徴ベクトルあるいは基本文書データと呼ばれる簡単なリスト構造としてメモリあるいはファイルに保存される．文書特徴ベクトルの構造については次節で詳しく説明する．

テンプレートはサンプル用文書から作成された文書特徴ベクトルに，ユーザーが指定要素領域に要素名と簡単な属性を与えることによって作成される．はじめに紙文書から必要な特徴を抽出してテンプレートを作成するまでの処理の概要をFig.3を用いて説明する．

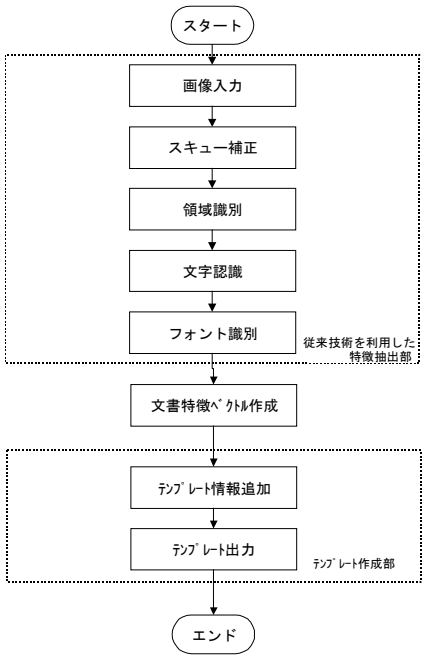

Fig.3　Flow diagram of extracting layout features from a document image.

文書をスキャナーから，あるいは画像ファイルからデジタル画像として入力する．入力文書に傾きがあればそれを補正して，文書の段落構成や図表および文字領域を識別して，行そして文字を切り出す．本論文ではキーワード照合など言語情報を用いた処理は一切行わない．したがって切り出された文字画像を認識する必要はないが，後行程での利用を配慮してこの段階で文書のすべての文字を認識する．これらの処理はリコー日本語活字OCRソフト「読取物語EX V3」に搭載されている技術を用いる．さらに阿部[4]の方法を用いて文字の認識と同時にフォントを識別する．以上がFig.3に示されたフローチャート上部に相当する流れである．これらの処理を終えた時点で，文書サイズや文字方向，切り出された領域/行毎に座標値や文字サイズなどのレイアウト特徴が得られる(Table 1)．本書では，前記抽出された入力文書全体に関する情報と領域/行のリスト構造をあわせて文書特徴ベクトルと呼ぶ．

Table 1　Required layout features of document images.

	レイアウト特徴の種類
文書全体	文書サイズ，文字方向，文字種
文字領域/行	座標，文字ｻｲｽﾞ，文字ﾌｫﾝﾄ，ｲﾝﾃﾞﾝﾄ，……

2-2　テンプレート作成

本論文で用いられるテンプレートは，前述の文書特徴ベクトルに必要とされる書誌事項の情報を与えたものである．本節ではFig.3の右下に示されたテンプレート作成部におけるテンプレートを作成するためのユーザーインターフェースを中心に，サンプル文書の入力からテンプレート出力までの流れをFig.4を用いて説明する．

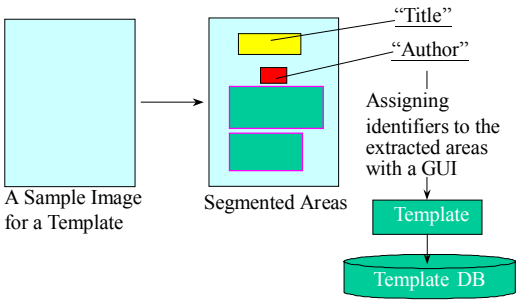

Fig.4　Easy operation for assigning a label name to a required element in the sample image.

はじめにテンプレートの基礎となるサンプル文書を任意に1枚選択して，前述の技術を用いてサンプル文書からレイアウト特徴を抽出する．Fig.4の中央は領域識別処理されたサンプル文書の各領域を色矩形で表示した画面である．

次いで，表示された文書の各領域の中から必要な書誌事項領域をマウスなどで選択することにより，指定要素設定用のダイアログが表示される．ここでユーザーが行うべき唯一の作業は，指定された書誌事項にラベル名を与えるだけである．この際，与えられたラベル名は単なる識別子としての役割しか持たないので，例えばラベル名として"Address"が与えられたとしても，住所辞書などを用いた知識処理は行わない．

もし必要ならば，ラベル名以外にも指定要素が持ち得る最大行数などの属性を指定することも可能であるが，テンプレートは基本的に指定要素のレイアウト情報のみ保持する．したがって提案するテンプレート作成手法では，第1節で紹介した従来技術のような文書要素間の相互依存関係を細かく定義するような面倒な手続きは一切不要である．

作成されたテンプレートはFig.5のようなタグつきコード形式として出力される．今回テンプレートをバイナリーデータではなくタグ付きテキストデータとして出力したのは，特徴抽出処理と要素抽出処理を分離独立化すること，およびテンプレート構造を単純化し，メンテナンス性を向上させるためなので，テンプレートを厳密なSGML(Standard Generalized Markup Language)形式では定義していない．

```
<!DOCTYPE page SYSTEM>
<page  type="Sample" number=1 width=595 height=841 cell_num=50 area_num=8 line_dir=0>
<cell id=19   label=" 開　始　日 " tbl=4 xs=85 ys=521 xe=85 ye=553 >
<area id=25 xs=85 ys=521 xe=85 ye=553 max=-1 ln=2 sz=21 ft=-1 col=-1 cp=-1 idt=3 kd=25 lkd=-1 >
<line id=25 xs=30 ys=182 xe=30 ye=194 len=19 cp=-1 sz=21 >　　（期日が年月日でない場合具体的に記載）
</line>
<line id=26 xs=30 ys=182 xe=30 ye=194 len=19 cp=-1 sz=21 ></line>
</area>
</cell>
<cell id=20 tbl=4 xs=85 ys=563 xe=85 ye=553 >
<area id=26 xs=85 ys=563 xe=85 ye=553 max=-1 ln=1 sz=21 ft=-1 col=-1 cp=-1 idt=3 kd=26 lkd=-1 >
<line id=27 xs=30 ys=197 xe=30 ye=194 len=13 cp=-1 sz=21 >　　終　了　日　end date</line>
</area>
</cell>
<cell id=21   label=" 終　了　日 " tbl=4 xs=85 ys=563 xe=85 ye=595 >
<area id=27 xs=85 ys=563 xe=85 ye=595 max=-1 ln=2 sz=21 ft=-1 col=-1 cp=-1 idt=3 kd=27 lkd=-1 >
<line id=28 xs=30 ys=197 xe=30 ye=208 len=19 cp=-1 sz=21 >　　（期日が年月日でない場合具体的に記載）
</line>
<line id=29 xs=30 ys=197 xe=30 ye=208 len=19 cp=-1 sz=21 ></line>
</area>
</cell>
```

Fig.5　Example of the template file.

2-3　指定要素抽出基本方式

2-3-1　指定要素と候補行とのマッチング距離

本節では指定要素抽出の基本方式を説明する．ここではFig.6を用いてテンプレートに指定された要素"DATE"に相当する項目をFig.6の右側文書から抽出する具体的な方法を説明する．テンプレートに指定された前記要素をE，入力文書の候補行Lとおき，それぞれを抽出されたレイアウト特徴を用いて次のようなm次元ベクトルで表す．

$$E = (f_1^E, f_2^E, f_3^E, \ldots, f_m^E) \quad \cdots\cdots\cdots\cdots \quad (2.1)$$

$$L = (f_1^L, f_2^L, f_3^L, \ldots, f_m^L) \cdots\cdots \quad (2.2)$$

このとき指定要素Eとその候補行Lとのマッチング距離$d(E,L)$をマハラノビス距離を用いて次の式で定義する．

$$d(E,L)^2 = \sum_{i=1}^{m} \frac{|f_i^E - f_i^L|^2}{v_i} \quad \cdots\cdots\cdots\cdots\cdots \quad (2.3)$$

ここでv_i(i=1,...,m)は各レイアウト特徴の分散であり，式(2.3)では約100枚のテスト文書を用いた予備実験で求めた値を利用する．この距離を用いて抽出された候補行の順位を決定する．

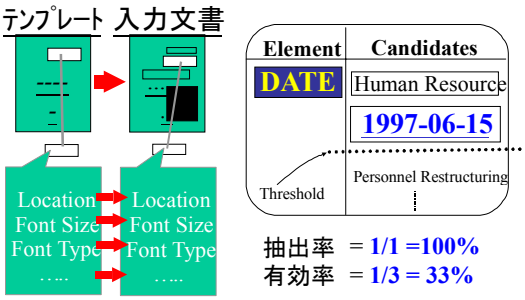

Fig.6 Simple matching strategy.

2-3-2 順番制約

提案する文書管理システムでは，キーワード抽出やキーワード検索が要求される．これらの機能を精度よく実現させるためには，抽出結果に多少のノイズが混入したとしても正解候補を漏れなく抽出するのが望ましいが，その結果一つの指定要素Eに対して複数の候補行が選択されてしまうのは避け難い．本研究では，定型化された文書の文書要素は一定の順番で配置されていると仮定している．最後にこの仮定を利用して次の方法で抽出精度の向上を図る．

入力文書のすべての行L_1, ..., L_nに対して，Eとのマッチング距離を求めたら，マッチング距離に関する適当なしきい値でEの候補を絞り込む(Fig.7)．前記で絞り込まれたEの候補行を，距離の順序で並べ替えてL'_1, L'_2, ...と書き直す．もし，Fig.7に示すように候補L'_1とL'_2のページ上における位置が逆転していれば，最後に並べ替える処理を加えてもよい．また，実際のページでは候補行L'_1,L'_2の間に，前記しきい値で足きりされた別の候補行L'_tが存在するかもしれない．この場合も，$L't$を救済して最終結果に加えてもよい．

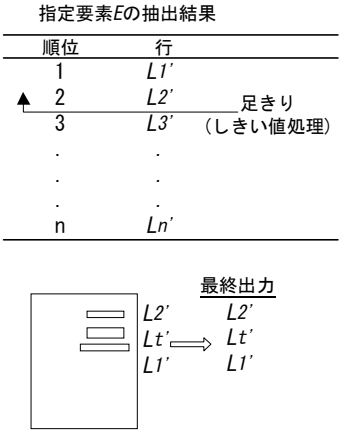

Fig.7 Post-processing of the template matching.

2-4 テンプレート自動更新による学習機構

2-4-1 学習機構の構成

これまで説明した指定要素抽出の基本方式では，1つのテンプレートは1枚のサンプル文書から作成されていた．我々のテンプレートはGUIを用いて指定要素を選択するだけで誰でも簡単に作成できることが最大の特徴である．さらにテンプレートの単純な構造によって，文書要素間の複雑な相互依存関係を原因とする認識過程の悪影響を回避することができた．しかしながら，1枚のサンプル文書だけで該文書全体を代表するには限界がある．例えば，技術論文の表紙ページから1行の表題と1行の著者項目を抽出対象要素としてテンプレートに指定したとする．ところが2枚目以降の文書では表題が2行以上存在するかもしれない．著者項目にいたっては共著者の数によっては3行以上存在することも珍しくない.このようなレイアウト変動を持った文書にも精度よく対応するには，何からの方法でレイアウト特徴の変動状態を抽出過程に反映させなければいけない．

本章では，複数のサンプル文書を用いて文書全体のばらつき具合いを検出する方法と，前記検出されたばらつき具合いにもとづいてテンプレートを更新して，要素抽出精度の向上をはかることを特徴とする学習機構を提案する．Fig.8に提案する学習機構全体の構成を示す．

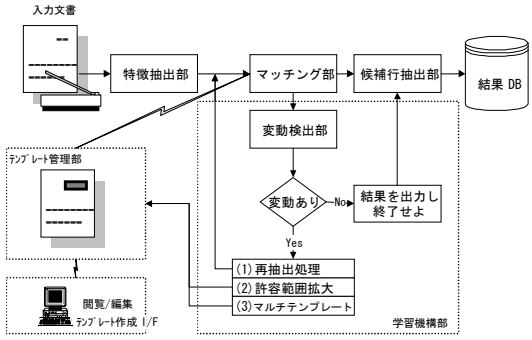

Fig.8 Outline of the learning system

Fig.8の左上から学習用サンプル文書を入力して，レイアウト特徴を抽出した後テンプレートマッチングにより指定要素を抽出してその結果をデータベースに格納する．提案する学習機構はFig.8の右半分に示された部分であり，そこではまずマッチング部から得られた情報を基にテンプレートに記述

された指定要素の変動量を検出する．もし一定値以上の指定要素の変動量が検出されたならば，変動の状態によって1."再抽出処理", 2."テンプレート更新", 3."マルチテンプレート"いずれかの対応を行う．逆に指定要素の変動量が検出されなかった場合は学習機構の処理は行わずに基本方式の流れに戻る．

2-4-2　レイアウト特徴変動量

本節ではある特定の書式を持つ1種類の文書群を対象とする．あらかじめ該文書の指定要素を定義したテンプレートは作成されているとする．提案する学習機構は，1枚づつ入力される処理対象文書から指定要素を抽出する際に，指定要素ごとに位置の変動や文字サイズおよびフォントなどのレイアウト特徴のばらつき具合いを自動的に検出する．本論文では，前記ばらつき具合いを表す指標として指定要素の変動量を定義してそれを効率よくテンプレートに反映して効果的な学習を試みる．指定要素の変動量の定義および検出方法をFig.9およびFig.10を用いて説明する．

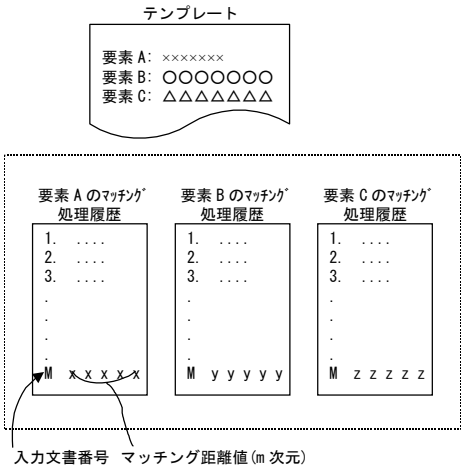

Fig.9　Processed data of the element extraction.

基本方式の流れに従って入力文書とテンプレートとのマッチングを行い，指定要素を抽出し，その抽出結果をデータベースに出力する．このとき指定要素毎にマッチング処理結果をFig.9に示すような履歴情報として保存する．マッチング処理結果とは，入力された文書番号毎に座標や文字サイズなどのレイアウト特徴を比較した距離値である．新文書は履歴の末尾に追加される．Fig.9の例ではテンプレートに3つの要素(A,B,C)が指定されていて，それぞれがテンプレートとは別に履歴情報を保持している．各要素のマッチング処理履歴の第1列は過去に処理してきた文書番号が記述されている．Fig.10はある指定要素Aのマッチング処理履歴情報の詳細である．

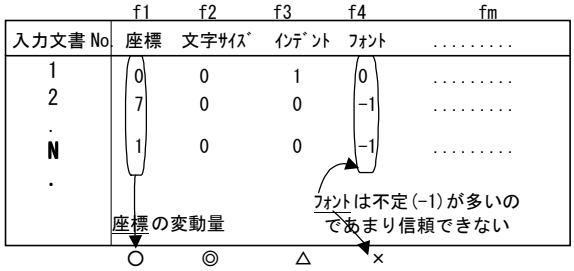

Fig.10 The details of layout variation of an element in processed data.

このように複数の文書を逐次的に求められた履歴情報の列データを参照すると，各指定要素毎に各レイアウト特徴のばらつき具合いが統計的に明らかになる．Fig.9のフォント特徴に関するマッチング処理の列を見ると，フォント特徴は過去に不定値(-1)を多く検出しているので要素Aに関してはフォント特徴はあまり信頼できないことが分かる．

レイアウト特徴空間の次元をmとしたとき，指定要素Eの変動量$V(E)$は，前記m個のレイアウト特徴のばらつき具合いから得られる値をそれぞれ適当に重みづけることによって求められる．例えば前述の論文表紙ページから表題(Title)を抽出する例では，論文毎の表題行数のばらつきにより表題以降の文書項目(著者など)の位置が大幅に変動する可能性があった.定型化された一般文書においても，このような位置の変動は比較的頻繁に発生すること，および位置の変動により正解候補の抽出精度が大きく低下することは経験的にわかっている.したがってレイアウト特徴の一つである「座標」は重要な特徴の一つであり，重みを強めるようにする．その他の各レイアウト特徴のばらつき具合いの定量化やそれらの重み付け等は，処理対象となる文書や実装形態に強く依存するのでここでは省略する．

2-4-3　テンプレート更新

1枚のサンプル文書から作成されたテンプレートでは，2枚目以降に入力された文書のレイアウト変動に精度良く対処するのは難しかった．この問題を解決するために，前節では複数のサンプル文書から逐次的に指定要素毎にマッチング処理履歴情報およびレイアウト特徴の変動量を検出する方法を提案した．ここではそれらの情報を用いて効率よく抽出精度

向上を図る手法を2つ提案する．

1つ目は，Fig.11に示すようにテンプレートに記述されたレイアウト特徴が取り得る範囲を適宜更新することである．

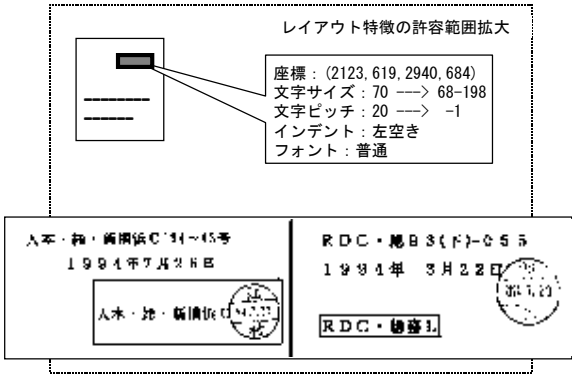

Fig.11 Modifying the template according to the rage of a layout feature.

本論文で提案するもう一つの学習方法は，1つのカテゴリを複数のマルチテンプレートでカバーするマルチテンプレート方式である．提案した基本方式では，はじめは必ずシングルテンプレートである．ところがFig.12の例では，同じ指定要素Eがページによってインデント特徴が異なる(EとE')．このような例に対処するためには，E'の存在範囲をカバーするテンプレートを新たに作成すればよい．Fig.13を用いてマルチテンプレート化の流れを説明する．Fig.13では，予め文書αからテンプレートT_1が作成されているとする．2枚目以降の文書βから指定要素Eを抽出する際，指定要素Eの変動量$V(E)$も同時に検出される．もし変動量$V(E)$があるしきい値よりも大きい場合に，かつ離散的な変動を持つレイアウト特徴が存在するならば，文書βから新たにテンプレートT_2を作成する．最後に既存テンプレートT_1とT_2を合成してマルチテンプレート構造を構築する．

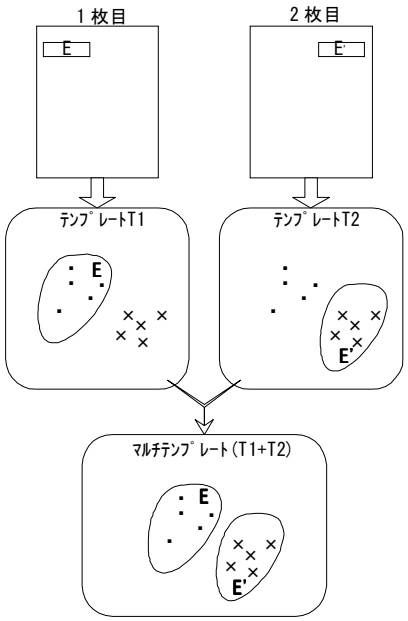

Fig.12 Outline of a multi-template strategy.

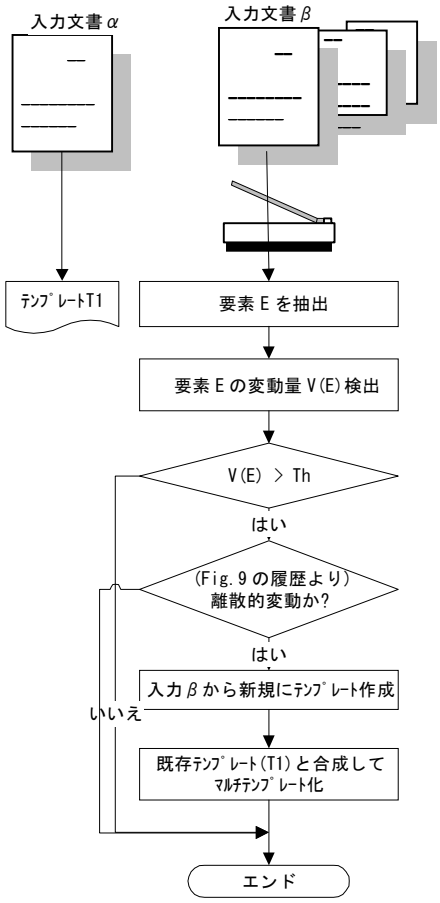

Fig.13 Flow diagram of a multi-template strategy.

Fig.14は，実際にマルチテンプレート化する際のユーザーインターフェースを示す図である．ページ上部に配置された要素(FROM:"リコピー","リコー")が偶数ページと奇数ページとで印刷される位置が異なる．この例のように，ある指定要素のレイアウト特徴の変動量が一定値以上検出されて，かつインデント幅(右，左，中央)のような離散的な性質を持つ変動が含まれている場合にはマルチテンプレートが有効である．

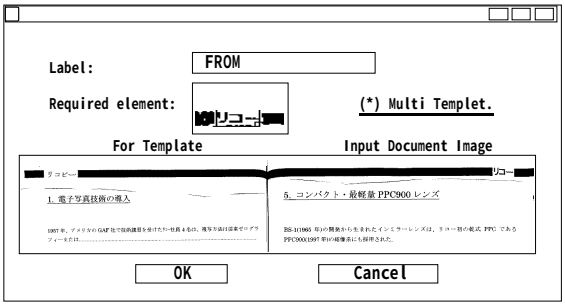

Fig.14 Introduction of a multi-template strategy.

3. 実験結果

本論文で提案された要素抽出方式の精度を測定するために，実際にオフィスで使用されている文書画像を用いて評価実験を行う．

3-1 テスト文書

実験では，通達文，論文表紙，雑誌，縦書き文書など17種類288枚の文書画像を用いた．またテスト文書全体を書式のレイアウト変動の大きさに応じて3つのセットに分類して，それぞれのセットに対する提案手法の適用度を測定する．分類された3セットは書式の変動具合に応じて，1.変動小，2.変動中，3.変動大と呼び，それぞれのサンプル数と指定要素の数はTable 2の通りである．

Table 2 Test images

	ｻﾝﾌﾟﾙ数	要素数
変動小	100	331
変動中	83	425
変動大	68	304
縦書き	37	74
計	288	2232

後半ではテンプレート更新による学習機構の効果を確かめるために，テスト文書を学習用と評価用に分割して実験を進めた．

3-2 テンプレート作成

はじめに1枚のサンプル文書から定型書式を持つ一般文書1種類を代表するテンプレートを作成する．この場合，サンプル文書の選び方によってその後の要素抽出精度が大きく影響される．もしテンプレート用サンプル文書だけが当該文書タイプ中で極端に他と異なるレイアウト構成を持つならば，2枚目以降の文書から精度良く指定要素を抽出することは難しい．今回の実験ではテンプレート用に任意に1枚の文書を選択した．実験用文書の1種類である(株)リコー社内通達文(Fig.15)を例にとってテンプレート作成の具体的な方法を説明する．

1) テンプレート用サンプル文書として複数の社内通達文の中から任意に1枚選択する．
2) 選ばれたサンプル文書が極端に他と異なるかどうかを確認する．ユーザーにテンプレート用に最適なサンプル文書の選択を強いるのは望ましくないので，ここでは明らかに他と異なる場合だけ再選択する．
3) 次の5つの要素を指定する．
   TYPE, DATE, TO, FROM, TITLE, ABSTRACT

同様にして他の文書タイプからもテンプレートを作成した．指定された要素は主にページ上部に印刷されたタイトル，日付，ID・記号などである．

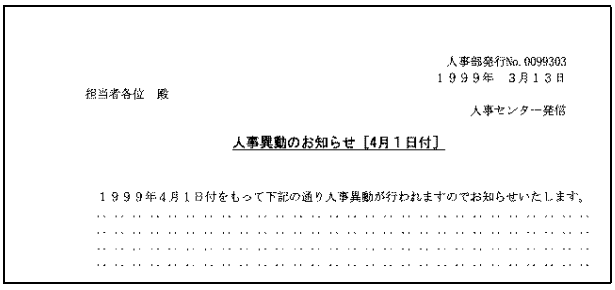

Fig.15 An example of test pages (business letter of Ricoh).

3-3　要素抽出実験

3-3-1　基本方式

はじめに17種類145枚のテスト画像を用いて指定要素抽出の基本方式の性能を評価した．その結果をTable 3の左側に示す．

Table 3　Experimental results.

	基本方式		学習後	
	抽出率	有効率	抽出率	有効率
変動小	98.2	67.1	98.2	72.9
変動中	99.2	51	99.2	56
変動大	85.3	43.6	100	44.8
縦書	100	47.7	100	65
計	95.6	55.5	99.2	61.4

ここで抽出率とは正解行が抽出候補の中に含まれている確率であり次の式で求められる．

$$\text{抽出率} = \frac{\text{抽出正解行}}{\text{正解行数}} \quad \cdots\cdots\cdots\cdots\cdots\cdots (3.1)$$

有効率とは抽出候補に含まれる正解行の割合であり次の式で求められる．

$$\text{有効率} = \frac{\text{抽出正解行}}{\text{全抽出行数}} \quad \cdots\cdots\cdots\cdots\cdots\cdots (3.2)$$

3-3-2　学習

テンプレート自動更新を特徴とする学習機構の効果を確かめるために17種類287枚のサンプル画像を学習用セット142枚，評価用セット145枚に分離して次の手順で評価実験を行う．

1)　学習セットを用いてマルチテンプレート化を含めたテンプレート自動更新
2)　前記更新されたテンプレートを用いて評価用セットで実験

学習セットを用いたテンプレート自動更新過程では，2種の文書でマルチテンプレート化が認められた．これらは，いずれも変動大セットに属する文書であり，新たに作成されたテンプレートの数は，文書1種類につき1〜2個であった．

評価用セットを用いた学習後の要素抽出実験の結果をTable 3の右側に示す．テンプレート更新の効果を示すために，Fig.16に基本法式での抽出結果とテンプレートを更新した場合の結果をグラフで示す．

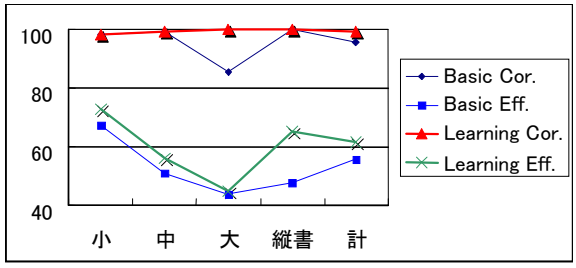

Fig.16 A comparison of experimental results between the basic algorithm and the learning method.

学習実験では有効率の低下を押さえつつ全体の抽出率が95.6%から99.2%へと向上した．特に変動大セットに対しては，抽出率が85.3%から100%へと大きく改善した．これは主にマルチテンプレート化によるものである．前出のFig.14に示された文書例では，全体で2つのテンプレートを持つマルチテンプレート構造が構築されて，入力ページ毎に最適なテンプレートを選択することにより精度よく抽出することが可能になった．

一方，テンプレートに記述されたレイアウト特徴の範囲を更新する方式でも，前記マルチテンプレート方式には及ばないが若干の正解抽出率向上が認められた．今後は，学習サンプルを大量に用いることにより更なる抽出精度の改善が可能であると予想される．

4. 結論

位置が固定された表形式文書からではなく，定型化された書式を持つ一般文書からタイトルや著者などの書誌事項を精度良く抽出するための新しい方法を考案した．多種多様な実データを用いた実験から，1枚のサンプルのみから作成されたテンプレートを用いる基本方式でも，同一書式内のレイアウト変動が中程度の文書に対しては文書のレイアウト変動に対して十分に追従できることを示した．さらに複数のサンプル文書を用いてテンプレートを更新する学習機構により，レイアウト変動が大きな文書に対しても抽出精度は大きく改善された．

本研究により，紙文書を自動入力する際にキー技術とな

る指定要素抽出技術が完成した．

5．今後の展開

これまで説明してきた指定要素抽出の流れでは，入力文書に対するテンプレートは予め人手で用意されていると仮定していた．ところが実際の利用形態では異なる文書がランダムに入力されて，かつ連続的な処理が求められる可能性が大きい．この際，人手に頼らずに入力文書を円滑に処理するためには，ランダムに入力される文書の種類を自動的に識別して最適なテンプレートを選択する技術が不可欠である．テンプレート自動選択技術は，筆者ら[14]によってある程度の見込みがつけられているが，今後も改良を進めていく必要がある．

また適応可能な書式の拡大を図ることも重要な課題の一つである．特に従来の表形式文書認識処理と融合して，表を含む文書から指定されたセルや一般の書誌事項を同時に抽出する技術は，今後実用的な文書管理システムを構築する上で早急に取り組むべき課題である．

参考文献

1) T.Kochi, T.Saitoh. A layout-free method for extracting elements from document images, Proceeding of Third IAPR Workshop on DAS, (1998), pp.336-345
2) J.Hull, et al.: The infinite memory multifunction machine(IM3), Proceeding of Third IAPR Workshop on DAS, (1998), pp.49-58.
3) T.Saitoh, et al.: Document Image Segmentation and Text Area Ordering, Proceedings of ICDAR, (1993), pp.323-329.
4) 阿部悌: 日本語活字フォントの識別,電子情報通信学会誌,(1997).
5) T.Watanabe, et al.: Extraction of data from preprinted forms, IEEE Trans. Pattern Analysis and Machine Intelligence, Vol.17, No.4, (1995), pp.432-445.
6) J.Yuan, et al.: Form items extraction by model matching', ICPR'96, (1996), pp.691-695.
7) H.Arai, K.Odaka: Information Acquisition and Storage of Forms in Document Processing, ICDAR, (1997), pp.164-169.
8) C.Wenzel: Supporting Information Extraction from Printed Documents by Lexico-Semantic Pattern Matching, ICDAR, (1997).
9) M.Sharpe, et al.: An Intelligent Document Understanding & Reproduction System", MVA'94, (1994), pp.267-271.
10) T.Watanabe, X.Huang: Automatic Acquisition of Layout Knowledge for Understanding Business Cards, ICDAR, (1997), pp.216-220.
11) H.Walischewski: Automatic Acquisition of Spatial Document Interpretation', ICDAR, (1997), pp.243-247.
12) C.Lin, et al.: Logical Structure Analysis of Book Document Images Using Contents Information, ICDAR, (1997), pp.1048-1054.
13) Y.Tang, et al.: Document Processing for Automatic Knowledge Acquisition, IEEE, Transaction on Knowledge and Data Engineering, Vol. 6, No.1, (1994), pp.3-31.
14) T.Kochi, T.Saitoh: User-defined template for identifying the document type and extracting elements from documents, Proceeding of Fifth International Conference on Document Analysis and Recognition, (1999).