修士学位論文

文書画像の領域分割に関する研究

東北大学大学院工学研究科 電気・通信工学専攻

塚田 仁志

目 次

第 1 章

序論

1.1 背景

現代は情報が氾濫する時代である．近年，膨大な量の情報を効率よく管理したいという観点から **“情報の電子化”** が注目を集めている．情報の電子化とは，様々な形態 (紙，音声，映像...) で存在する情報をコンピュータ上で処理できる形に変換することである．これにより情報の保存や検索を容易に行なうことができ，社会発展に大きく寄与することができる．

さまざまな形で発信される情報のなかで，新聞や雑誌や書類など，紙に印刷するという形で発信される情報は，その量も多く，また情報管理の需要も高い．印刷文書を電子化する単純な方法として，文書を光学スキャナで読み取とることにより得られた画像データを保管するという方法がある．この方法は容易に実現可能であるが，画像データによる保管であるためそのデータ量が莫大になってしまう点や情報検索ができないという点で有用性は低い．これに対し，印刷文書をコード化して保管する方法が有望視されている．現在印刷文書の電子化の大部分は，人手によりコンピュータに入力するという形がとられている．しかし前述したように，情報の電子化は膨大な量の情報を効率的に管理することが最大の目的である．人手による入力では時間と手間が多く必要となるため，これを自動的に行なうことが強く求められている．

印刷文書を電子化することを目指して，文字や図や表や写真など，文書を構成する要素を認識する処理の研究が行なわれている．文章領域に対して行なわれるは OCR(Optical Character Recognition) 技術である．この技術は古く 1960 年代から数多くの研究がなされており，近年では文字画像中にノイズが多く含まれる場合や，文字が変形している場合など，様々な障害に対しても安定して高精度の認識率を得る手法が確立している．図形領域や写真領域，見出し領域に対しては，画像処理の分野において，エッジ検出を行なうなどしてそこに何が描かれているかを認識する研究がなされている．また表領域に対しても単に表内の文字を認識するだけでなく，表を構成する要素間の意味的な関係を理解する処理が提案されている．

ここで挙げた認識処理は，文章や表や写真など個々の構成要素に対して行なわれる処理であり，実際に印刷文書の電子化を行うためには，与えられた文書のどの位置にどのような形で文書の構成要素が配置しているか，さらにそれらの構成要素間にどのような関係があるかという情報 (**文書構造** [1]) を獲得する必要がある．文章だけでなく写真や図や表などの領域が混在する文書の文書構造を解析し，得られた構成要素ごとに構成要素に応じた認識処理を適用するという一連の処理を実行するシステムを**文書認識システム**と呼ぶ．文書認識システムは構造解析部と認識部から成る (図 1.1)．入力文書から構成要素の領域のレイアウト情報を獲得するのが構造解析部で，その情報をもとにして各領域に対し認識

処理を行ない，文書全体の情報を理解してデータベースに構築するのが認識部である．

文書認識システムを高精度なものとして確立させるためには，先に述べた個々の要素を認識する処理 (OCR，表認識など) が高精度なものに発展すると共に，文書の構造を解析する処理も高精度なものになることが必要となる．文書の構造を解析する処理を高精度にするとは一言で言えば，様々なフォーマットの文書画像に対応する処理を確立するということである．近年，印刷技術の向上により，様々なフォーマットをもつ文書が新聞や雑誌等でみられるようになった．またそのような機能を備えた文書作成システムがワープロなどの形で市販されおり，一般の人でも簡単に自分好みのフォーマットで文書作成ができるようになった．

印刷文書の電子化の目的に膨大な量の情報を効率的に管理することがあることは前に述べたが，具体的な応用例としては次のようなものが考えられる．多くの本を電子化することによる電子図書館を実現することや目の見えない人を対象とした文書朗読システムや，点字翻訳システムなどを実現することなどである．これらの応用例を考えたとき，特定のフォーマットにしか対応できない文書認識システムでは実用面において問題がある．このようなことから，様々なフォーマットの文書に対して安定した処理が行なわれる文書認識システムが求められている．

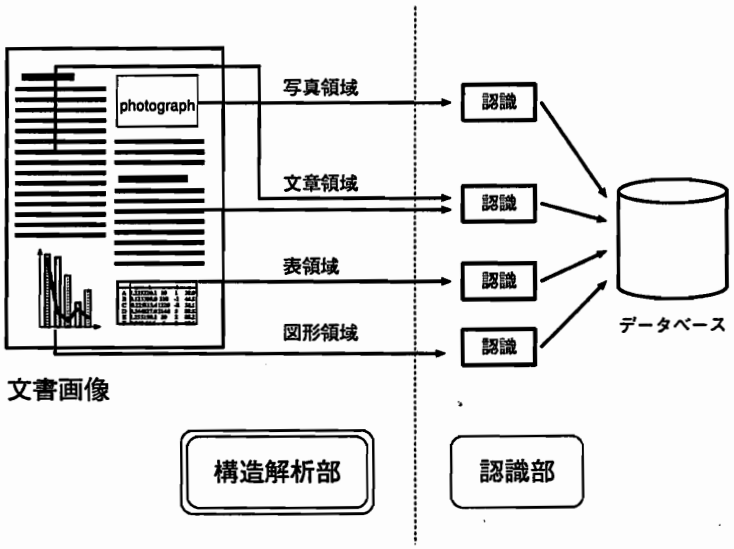


図 1.1 文書認識システム

1.2 本研究の目的

本研究では，様々なフォーマットの文書に対応することを目指し，文書認識システムのうち構造解析部に注目する．現在，文書認識システムを構成する際，様々なフォーマットの文書をターゲットとするにあたって最も大きな課題としてあげられるのが，**複雑な構造を持つ文書への対応**である．複雑な構造を具体的述べると，

- 構成要素の領域の形状が必ずしも矩形ではない
- 別々の構成要素が極めて接近

などである．このような構造を持つ文書では，領域の境界線を特定することが難しいためこれまで構造解析が困難とされてきた．複雑な構造の文書に対応する文書認識システムを構成するため，個々の文書に特化したシステムを構成するという方法が用いられている．しかしこの方法では様々な構造の文書に対応することができないので，本研究の対象外とする．

ここで，人間が文書を読むことについて若干考察する．人間はこれまでに見たことのないようなフォーマットの文書が与えられた場合，極めて特別な例を除いてはその文書構造を理解することが可能であると思われる．このような文書構造を理解する過程においては，未知フォーマットの文書を一定の経験則により理解していると考えられる．このことから，大多数の文書に共通する基本的なルール，つまり汎用性の高いルールだけを用いることにより複雑な構造をもつ文書にも対応できるのではないかという考えに至る．よって本研究の目的は，複雑な構造の文書にも対応できる文書認識システムを開発することとする．またシステム構成において用いられるルールは，汎用性の高いルールに限定することとする．

1.3 本論文の構成

本論文の構成は以下のとおりである．

第 1章 本研究の背景と研究の目的を述べる．

第 2章 文書認識システムの概要を述べ，研究の方針について考察する．

第 3章 文字行の形状や相互位置関係の類似性に注目し，文字行候補を段階的にグループ化して文章ブロックを生成する手法を提案する．

第 4章 文字のならびの規則性を基に，表領域の一部であると思わる表候補小領域を形成して，それらを統合することにより，文書画像から表領域を抽出する手法を提案する．

第 5章 本論文全体をまとめ，さらに今後の課題について述べる．

第 2 章

文書認識システム

本章では，文書認識システムの概要について触れる．また本研究の対象としている構造解析処理において現在用いられている手法を検討する．それらを踏まえ，文書認識システムを構成する際に，どのようなアプローチが有効か，文書画像のどのような特徴に注目するのが有効か，などの考察を行い研究の方針を決定する．

2.1 文書認識システムの概要

第 1章で述べたように，文書認識システムは構造解析部と認識部から成る．構造解析部が理解した文書構造を認識部に与えて処理が進行するわけであるが，ここで構造解析部が認識部に与える文書構造の表現手段を導入する必要が生じる．文書の構造を表現するものとして，木構造を用いて表現する**幾何構造**と**論理構造** [1] が広く用いられている．木構造を用いることにより，要素の包含関係が表現される．両構造共，木のルートノードは対象の文書画像全体 (ページ) を表わす．

幾何構造とは，文書を表面上の特徴から逐次分割することにより得られた要素を表わすものと定められている．幾何構造により，各構成要素の形状や位置関係が階層的に示される図 2.2の入力文書から獲得した幾何構造の例が同図 (b) により示されている．この例では，一つのカラムによるの領域群が一つのノードにより表現され，また下に接する別のカラムを子ノードとして表現している．

論理構造とは，文書を人間の意味理解という観点から逐次分割することにより得られた要素を表わすものと定められている．論理構造により各構成要素のもつ意味や，他の構成要素との意味的な関係を階層的に示される．よって論理構造による表現では，各構成要素は “表題”，“アブストラクト”，“副表題”，“段落” などのラベルをもつことになる．論理構造の例 (図 2.2(c)) では，title と subtitle の関係や，1 つの subtitle に含まれる paragraph 群との関係などが表わされている．

与えられた文書からその幾何構造を抽出する処理を**文書解析**，文書解析によって得られた幾何構造を論理構造にマッピングする処理を**文書理解**と呼ぶ (図 2.1)．本研究では構造解析部のうち，文書解析の部分を取りあげる．

2.2 従来手法の検討

構造解析部においてなされる処理で，中心的なものが**領域分割**である．領域分割とは，与えられた文書を 文章，表題，表，図などの領域に分割する処理で，文書解析においては重要な処理である．領域分割を行なう具体的な手法は多く提案されているが，それらは

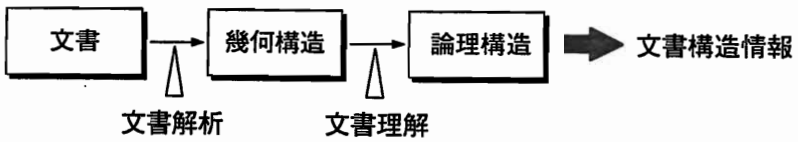


図 2.1 構造解析部

モデルベース型アプローチ，top-down 型アプローチならびに bottom-up 型アプローチに大別することができる．モデルベース型アプローチは文書の構造記述モデルと，入力文書画像とのマッチングにより領域分割を行なうものである．また top-down 型アプローチは，ページが大きい要素から小さい要素へと分割されていき，その過程において幾何構造を獲得するというものである．例えば，ページが文章のカラムに，文章のカラムが段落に，段落が文字行にというようにある．逆に bottom-up 型アプローチでは 小さい要素を統合して大きい要素が形成される．例えば黒画素連結成分が文字に，文字が単語に，単語が文字行に，文字行が段落にという形である．

図 2.3は，3 種の手法がどのような特性を持つかを図示したものである．以下でそれぞれのアプローチについて検討する．

2.2.1 モデルベース型アプローチ

画像処理により構造解析を行うのではなく，文書の構造記述言語や，木構造モデル [2] や，文字列をベクトル化する方法 [3] を導入して，参照モデルとの比較を行ない文書の構造解析をおこなう手法が提案されているここで用いられる参照モデルはどのようなものであるかというと，例えば，“文書のこの位置にこの形状の文字行があったら，それはヘッダである” というような規則を記述したものである．これらの手法の参照モデルは，対象としている文書をしぼって形成されている．よって，そのような特定の文書を扱う場合には高精度な処理がなされており，実用性は十分であるといえる．

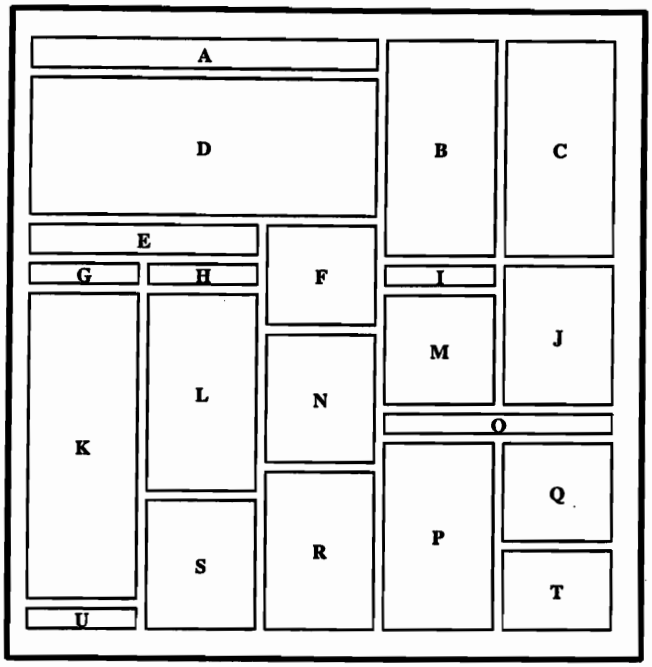
参照モデルを用いる手法の共通の問題点として，これらの手法で用いるルールは，個々の文書に依存する部分が大きく，対象文書が制限されてしまうため，様々な文書を処理する必要がある場合は膨大な量の参照モデルをあらかじめ準備する必要があるという点が挙げられる．よってモデルベース型のアプローチは個々の文書への依存性が高いという特性をもつ (図 2.3)．

2.2.2 top-down 型アプローチ

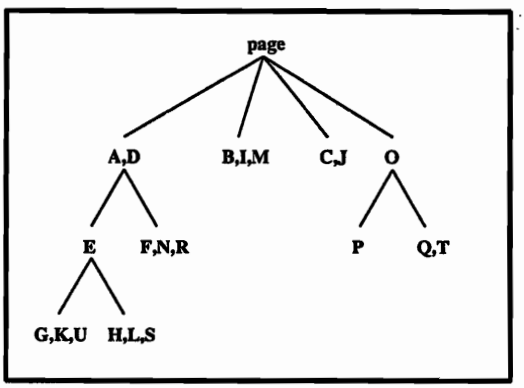
top-down 型アプローチとして代表的なものが，黒画素の射影ヒストグラムを用いる方法 [4] である．黒画素の射影ヒストグラムを用いる方法 [4] である．$f(x,y)$ を文書画像を二値画像により表現したものとし，R を 文書画像の範囲とする．水平方向，垂直方向の射影ヒストグラムは，それぞれつぎのように定義される．

$$p_{HORIZ}(t) = \int_R f(x,t)dx \tag{2.1}$$

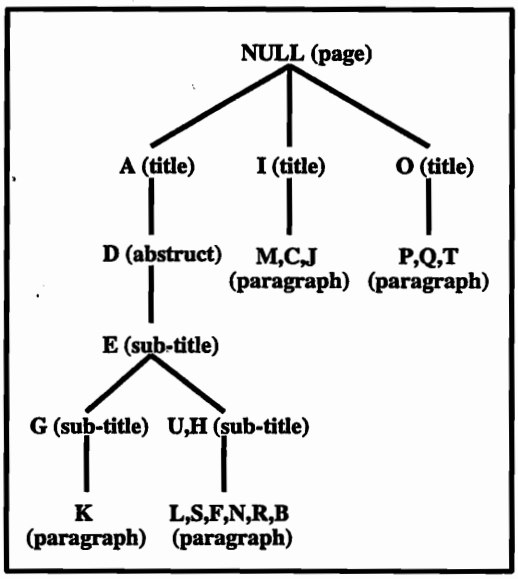
$$p_{VERT}(t) = \int_R f(t,y)dy \tag{2.2}$$



(a) 文書



(b) 幾何構造



(c) 論理構造

図 2.2 文書構造

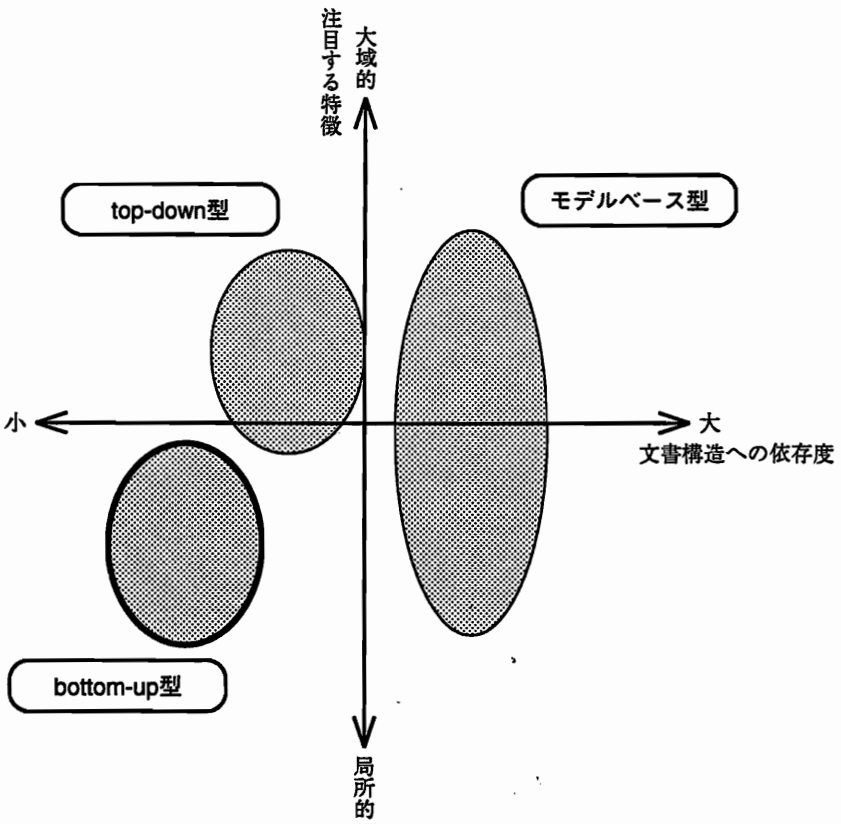


図 2.3 3アプローチの特性

射影ヒストグラムにおいて，0 が一定の長さ以上連続する部分を領域の境界とみなして分割する，という処理を水平垂直の両方向について再帰的におこなうことにより領域分割を行うものである．射影ヒストグラムによる領域分割の処理例を図に示す．また，文書画像の白画素領域に注目して領域分割を行なう手法 [5] もある．これは黒画素に接する空白領域の極大矩形 (白矩形) を抽出し，白矩形の面積とアスペクト比 (矩形の長辺と短辺の比) から領域分割要素として適する白矩形を選びそれらをもとに領域分割を行なうというものである．この手法では，空白領域を抽出したときに，それが領域分割要素として適当かどうかの判断方法を確立することが困難である点が問題としてあげられおり，その問題にれに対するため空白領域に対し “有効面積” を計算してより正確な判断を行う手法が提案されている [6].

ここで述べた 2 つの手法の特長は，文書の大域的な構造に注目して top-down 的に処理をすすめることで，処理速度が速いものになっているという点である．しかしこれらの top-down 型アプローチは，文書を構成する要素の領域形状が全て矩形であり，文書画像を水平方向あるいは垂直方向の分割をくりかえすことにより完全に領域分割できる **Manhattan レイアウト**にしか対応できないという問題がある．

2.2.3 bottom-up 型アプローチ

bottom-up 型アプローチとしては，外接矩形の周辺分布をもとに解析する方法 [7] や k-近傍をもとに解析する方法 [8] などがある．特に O'Gorman の手法 [8] では，黒画素の局所的な特徴に注目して統合する方法で，任意形状の領域分割に対応する方法である．

bottom-up 型アプローチについて詳しくは第 3章で述べるが，任意形状の領域への対応が可能であるという点において，対象が Manhattan レイアウトに限られる top-down 型アプローチより文書構造への依存性の小さいことがいえる (図 2.3).

2.3 本研究の方針

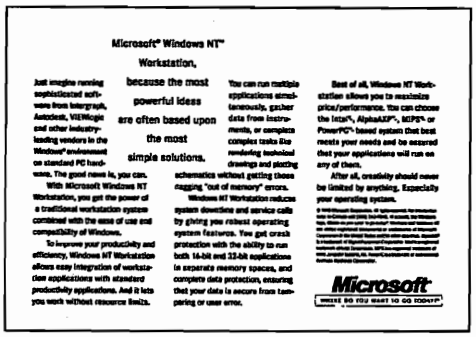
任意の形状の領域に対応するためには，領域の形状に依存しないルールが必要となる．大域的なルールは領域の形状に依存する部分が大きく，ここで用いるのは不適当といえる．よって，大域的でないルールとして局所的なルールを考えてみる．ここでいう “局所的” とは，隣接する外接矩形の間あるいは隣接する文字行の間程度の範囲を意味するものとする．このようなルールを用いた場合，領域の形状を仮定せずに画素の局所的特徴から領域を形成していくので，ルールが領域の形状に依存する部分が極めて少ないと思われる．したがって，局所的なルールを導入することにより汎用性の高いルールによる処理が実現できると考えられる．また，2.2節での検討も考えあわせると，様々な構造の文書に対応するには bottom-up 型のアプローチが望ましいといえる．

よって本研究では

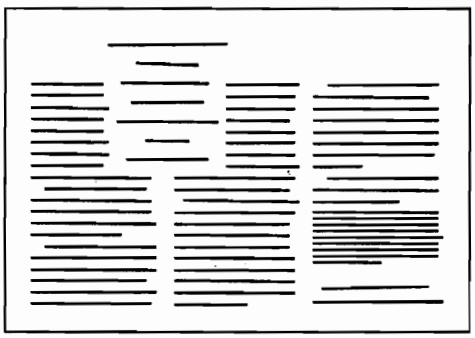
- bottom-up 型アプローチ
- 局所的特徴に注目

という方針で処理を構築することとする．

文書の領域分割を行なう際に，文書の意味解析 (たとえば文章領域なら，何が書かれているかということ) を行ない，その情報を基にして処理を行なうことにより有効な処理



(a)



(b)

図 2.4　文書画像 (a) とその文字行 (b)

が実現できるのではないかと考えられるが，ここでは領域分割を行なう際に意味解析をどのように用いることが望ましいかを考察する．図 2.4は比較的複雑な構造をもつ文書画像と，その文字行のみとを示したものであるが，我々は文書の内容を読むことなしに，文字行群だけを見てこの文書の領域分割を行うことができるであろう．人間が文書を読む場合，最初からいきなり文字を認識して意味解析による文書構造理解を行なっていると考えるよりも，文字のならびなどから視覚的にある程度まとまりの良い部分を一つの領域とするというような目安によって領域情報を獲得し，その目安だけではっきりしない部分は意味を解析することで文書構造を理解していると考えるほうが自然であろう．他方，文書を作成する立場から考えると，内容を読まなければ領域を理解することができないような構造を用いるのは極めて稀な場合であると考えられる．よって文書解析の初期の段階においては，大部分の文書はその幾何的な特徴のみに注目することで十分な処理が期待できる．また，意味解析を伴なう処理を行なった場合，文書の構成言語ごとに対応する処理を構築する必要があり，一般に文字認識処理では多くの実行時間が必要となる．よって本研究では意味の理解までは立ち入らず，文書画像の局所的な幾何特徴に注目したルールにより領域分割処理を構築することとする．

2.4 まとめ

本章では，文書認識システムの概要を述べ，様々な構造の文書に対応するためには構造解析部の改善が必要であり，構造解析部で不可欠となる領域分割を本研究で取りあげることとした．また領域分割で提案されている手法について，それらを 3 種に大分類し，それれの特性を検討した結果，“bottom-up 型アプローチ”，“文書画像の局所的特徴に注目” という本研究の方針を決定した．また，構造解析では初期の段階から意味解析を行なう必要はなく，大部分の文書画像は，その幾何特徴から文書構造を獲得することが可能であることを述べた．

第 3 章

文章領域抽出処理

文章領域は文書画像の中で最も重要な意味を持ち，また複雑な構造が多く見られる領域である．本章では文書画像から文章領域を特定する処理を提案する．

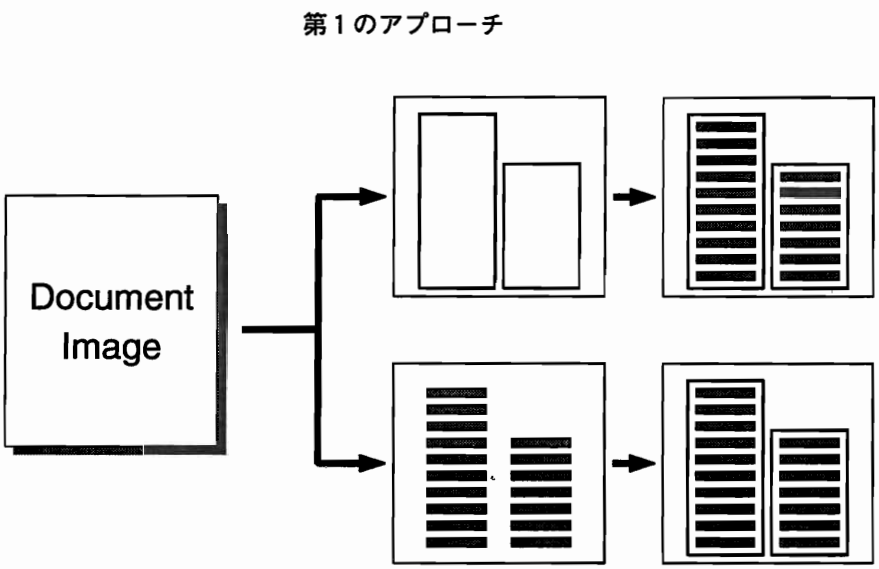


図 3.1　2 つのアプローチ

3.1　研究の背景

文書画像の領域分割処理に，複雑な構造への対応が課題となっていることは第 1 章で述べた．

局所的な特徴を用いて，文書画像から文章領域を抽出しようとした場合，2 種類のアプローチが考えられる (図 3.1参照)．第 1 のアプローチは，文書画像から先に文章領域を概略的に抽出し，その中で文字行や文字の大きさを推定してそれらを抽出し，文章領域を獲得するというものである．秋山らの手法 [7] は射影分布や罫線をもとに概略的に領域分割する．そこで得られた領域で射影分布や，crossing counts(走査線上で，白画素が黒

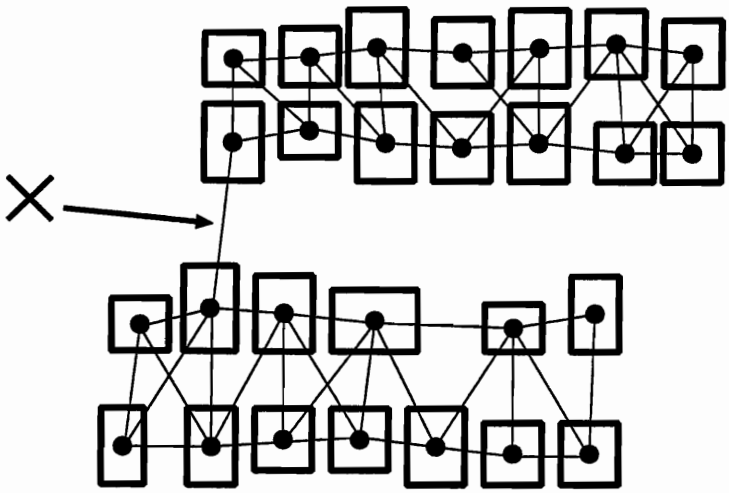


図 3.2 nearest neighbor を用いる方法

画素に変化する回数) 等を用いて文章領域を推定する．推定された文章領域内の外接矩形の中から代表的なものを選び，これを起点として外接矩形の周辺分布をもとに外接矩形を結合して文字行を形成するというものである．この手法では，概略的な文章領域抽出を top-down 的に行なっており，任意形状の文章領域には対応できない．O'Gorman の手法 [8] は，黒画素の連結成分を最小要素として，要素の中心間のユークリッド距離の近いものから k 番目までを結合する *k*-近傍という概念を導入している．ページ内の全ての要素についての *k*-近傍を求めることにより，要素間の "距離" と "角度" の分布を知ることができる．O'Gorman はこの分布を *docstrum* と名付けている．*docstrum* は文書画像内の文字行の諸性質 (文字行間の距離や文字行内の空白) を求めるために有用な情報となる．実際の文書画像では，ページ内の文字行が同じ全て同じ文字行間距離や文字行内空白を持つとは限らないなど，局所的に文字行の性質の異なる場合があり，この問題に対応するため *k*-近傍を結合して得られる領域をひとつの局所領域とみなす．そして局所領域内で *docstrum* を求めることにより文字行の性質を知ることができ，領域毎の性質に応じた処理が可能となる．しかしこの手法では要素を連結する条件において，距離に関する重要なしきい値を最適な値に設定しなければならない．特に，図 3.3のように異なる領域が接近している場合や構成要素が混み入って配置されている場合は，領域を統合するかしないかのしきい値を最適にするのが困難であるという問題が指摘されている (図 3.2)．他方，第 2 のアプローチは，先に文字行を抽出し，それらをグループ化して文章領域構成するものである．このアプローチの場合，文字行の形状についての複数の特徴量を用いて統合判断を行なうことができる点や，もし一部で不正な統合が行なわれたとしても，後の処理にその不正統合を救済させる余地を残すことができるという点などから処理の安定が期待される．また，人間が文書を読む場合，図や写真などの混在する文書中から文字が集中していて似たような大きさの文字や文字行が配置している部分を単一の文章領域ととらえていると考えられる．したがって，文字行を組みあわせて文章領域を生成するアプローチが人間が作成した文書に適しているということができ，以上の理由から第 2 のアプローチを採用することとする．

文字行を基にして文章領域を抽出する第 2 のアプローチによる手法として提案されて

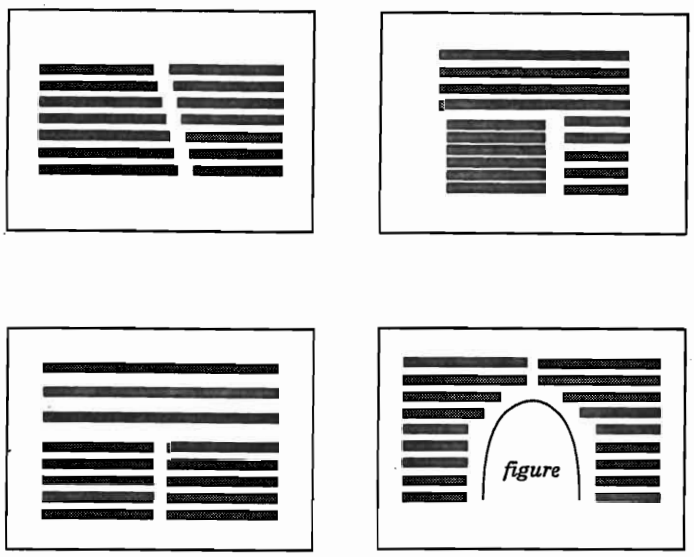
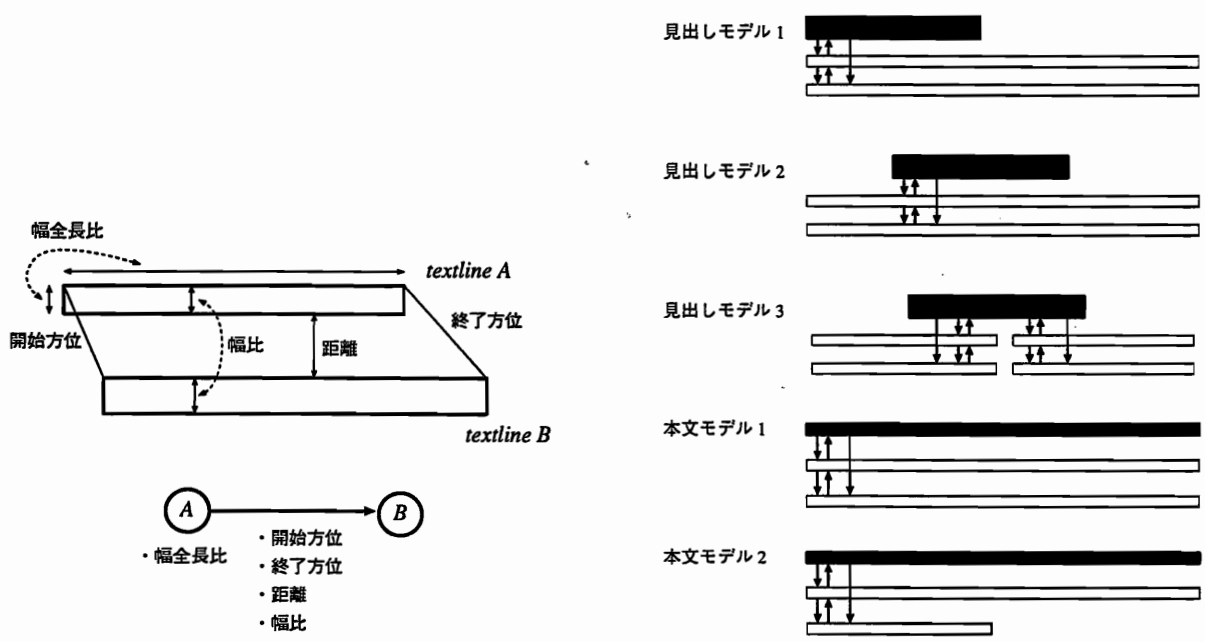


図 3.3　様々な文章領域



(a) グラフ表現　　(b) 内部参照モデル

図 3.4　グラフモデルによる構造解析

いるものに，距離が一定値以下である文字行同士を統合する処理を繰り返して行なうことにより文章領域を形成するという手法 [9][10] がある．これらの方法は，統合しようとする文字行との間で距離についての情報しか用いていないため，前述した第 2 のアプローチとしての長所が生かされておらず，図 3.3のような複雑な構造の文書への対応が困難である．

文字行の形状も考慮にいれた手法が，角谷らによって報告されている [11](図 3.4)．角谷らの手法は，文書画像中の文字行の形状や位置関係などをグラフにより表現する．文字行をグラフの節，文字行間の関係をグラフの辺とする．グラフの節は文字行の “長さ:高さの比”，グラフの辺は対象とする文字行との間の，“開始方位”，“終了方位”，“高さ比”，“距離” によって構成される．入力文書から得られたグラフを，内部参照モデル (図 3.4(b)) に変換する際のコストを計算し，最も少ない変換コストで変換される内部参照モデルに一致するとみなして，文書の領域分割を行うものである．内部参照モデルは，利用者が与える．これにより容易にルールの追加や修正を行なうことができ，汎用性が期待されている．報告では処理を見出し領域の抽出に限定しており，この手法を文章領域抽出まで拡張することが考えられるが，様々な形状の文章領域や，後に述べる分離文字行などに対応することをめざした場合，内部参照モデルの数が非常に多くなるため実現は困難であると考えられる．

また文字行の高さと距離についてのヒストグラムを基に，文字行をクラス分割する方法で文章領域を抽出する方法が平山によって発表されている [12]．文字行のクラスタ分割は，ページ中の文字行の高さのヒストグラムを用いて行なう．形成されたクラスタ毎の文字行の距離 (行間距離) についてのヒストグラムから距離のしきい値を求め，これを用いて文字行をマージして文章領域を抽出するという方法である．この手法では，高さがほぼ一致する文字行クラスで単一の距離しきい値を設けているため，1 ページ中に，文字の大きさが同じで行間距離が異なる文章領域が存在した場合両方の文章領域を抽出することが困難になるという問題がある．また，文書画像全体の文字行についてのヒストグラムを用いているため，比較的少数の文字行から形成されている文章領域がある場合，それが単一のクラスタとして正しく分割されるかどうか微妙であるという問題もある．これらの 2 つの問題に対応するためには，文字行を局所的ルールにより統合して，徐々に文章領域に近づける方法が有効であると考えられる．

ここで紹介した，文字行の形状を考慮に入れて文字行統合を行なう方法として提案されている手法 [11][12] では，システムへの入力データとして用いる文字行群はすべて正しく文書画像から抽出されたものであることを仮定している．しかし実際文書画像から文字行抽出を行なおうとした場合，後述するような問題が生じることを考慮に入れなければならない．

3.2 提案手法の概要

文書画像から文章領域生成までの処理の概要を図 3.6に示す．文書画像からの文字行抽出処理には，頑健・高速な文字行抽出法として提案されている区分直線連結法 [13] を用いる．文書画像から直接文字行抽出処理を行なう場合，抽出しようとする文字行の長さが未知であるため，

- 文字行間の過接合
- 分離文字行

図 3.5　過接合 (a) と分離文字行 (b) の例

などの生じることが問題となっている．(図 3.5参照)．過接合とは文章領域が接近している場合などに一直線上に存在する別々の文章領域に属する文字行が単一の文字行として誤抽出されることである．過接合は特に多段組構造をもつ文書において多く生じる．文字行抽出を行なった結果得られた，過接合の生じた文字行を含む文字行候補群を統合して文章領域を抽出しようとすると本来複数の段組で構成されている文章領域が，単一の文章領域として抽出されてしまうという大きな問題が発生する．また分離文字行とは文字行内の句読点による空白や単語間の空白などより 1 本の文字行が分離して別々の文字行として誤抽出されることである．そのような誤った文字行抽出が行なわれた場合でも正確に文章領域抽出処理を行なう必要がある．以下，文書画像から抽出された文字行群で，上に述べた過接合や分離文字行を含む可能性のあるものを**文字行候補**と呼ぶ

過接合に対応するものが，過接合切断部である．この処理は周辺の文字行の端点の配置や，空白部の配置から過接合と思われる部分を切断するものである．分離文字行に対しては，文字行を統合する段階で対応することが有効と考えられるので文章領域生成処理部の中で対応する．

3.2.1　文字行パラメータ算出

まず区分直線連結法により抽出された文字行候補から，図 3.7に示すような文字行のパラメータの算出を行う．このうち，(1) 傾き (slope)，(2) 長さ (length)，(3) 高さ (height) は個々の文字行の形状についての特性の評価を表す．また，(4) 距離 (distance)，(5) 重なり (overlap) は文字行の相互位置関係の評価を意味する．

3.2.2　過接合切断部

前述したように過接合は特に多段組の構造を持つ文書中の接近した文章領域間において生じることが多い．

文書を生成する規則から考えて，各文字行候補の左端および右端は文章領域の境界と一致する可能性がある．また，文書画像から抽出された文字候補行の中に過接合を生じた文字行が含まれている場合，過接合は異なる文章領域に属する文字行間で生じるので，その位置は境界に一致する．このことを考慮すると，文字行候補内におけるある程度以上の長さの空白部分も境界に一致する可能性がある．ここで，

- 文字行候補の両端点

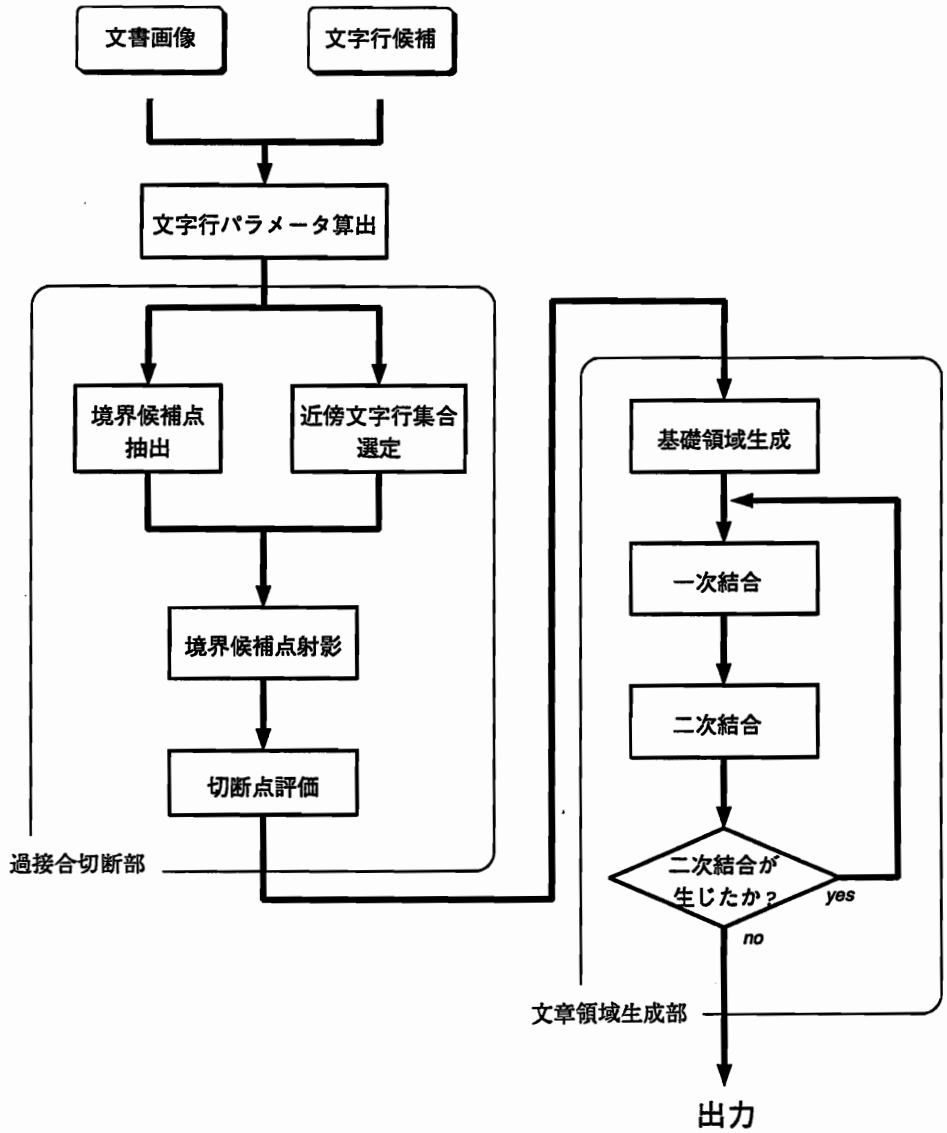


図 3.6　文章領域抽出処理のながれ

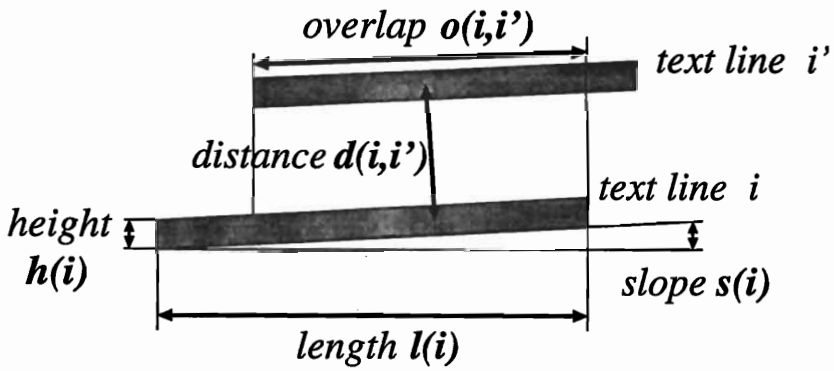


図 3.7　文字行がもつパラメータ

- 文字行候補中のある程度以上の長さの空白部分

を**境界候補点**と呼ぶ．境界候補点の集合の中からいくつかを用いることにより文章領域の境界を特定することができる．図 3.8に境界候補点をもとに文章領域の境界を推定する概念図を示す．

簡単な構造の文書の場合は，領域の境界線は水平あるいは垂直方向の直線と仮定できる．しかし，本手法は任意形状の領域分割を目指しており，境界線が必ずしもそのような直線であると仮定することはできない．また，境界線が直線ではなく湾曲している場合にも対応する必要がある．湾曲している線は局所的にみれば線分の結合したものと近似できるので，局所的な 境界候補点の分布で過接合の位置を推定することにより，そのような境界に対応することが可能となる．本処理部では，局所領域を設定するものとして**近傍文字行集合**を導入する．近傍文字行集合は全文字行候補ひとつひとつに対して定義される．文字行候補はその近傍文字行集合に属する文字行の境界候補点から境界が存在すると思われる位置が推定される．さらに文字行自身の画素の分布も考慮に入れた上で，境界と推定された位置に過接合が生じていると判断された場合その位置で 2 つの文字行に分割される．

以下では，文書画像から抽出された文字行候補の全体を $\mathcal{S}$，過接合切断部により過接合が切断された文字行候補の全体を $\mathcal{S}^*$ とする．

3.2.3　文章領域生成部

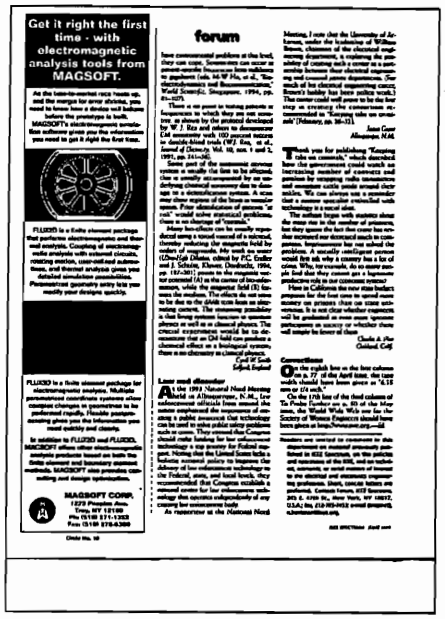
文章領域生成部では，**同一文章領域内には，形状や相互位置関係の類似する文字行が配置されている**という経験則を用い，文字行を結合して文章領域に相当する文字行グループを生成する．この経験則は大多数の文書に共通に成立している基本的なルールと考えられ，汎用性が期待される．文字行を結合するルールを導入するわけであるが，それらのルールは以下に示す原則を基に生成する．

原則 3.1 同一文章領域内の文字行は，傾きが一致する

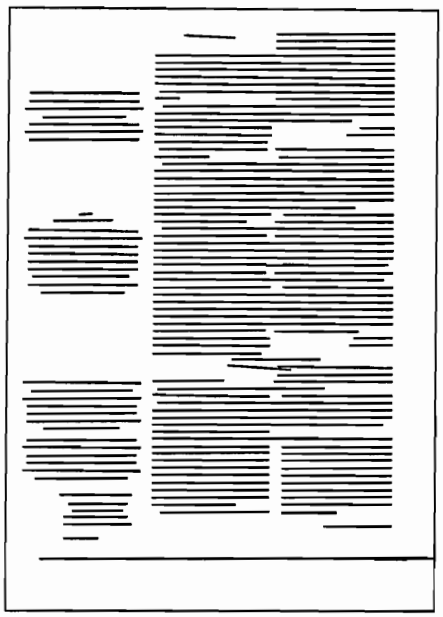
原則 3.2 同一文章領域内の文字行は，長さが一致する

原則 3.3 同一文章領域内の文字行は，高さが一致する

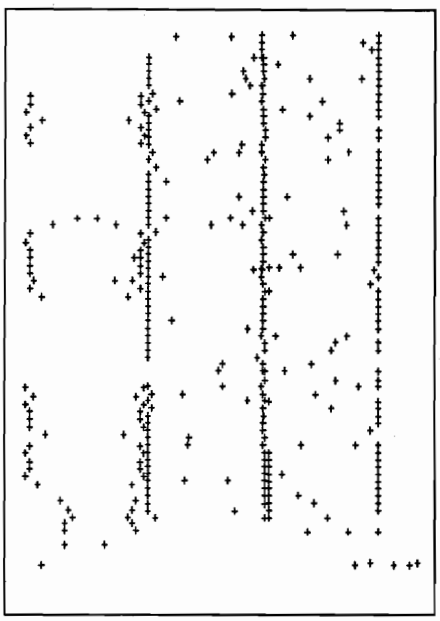
原則 3.4 同一文章領域内の文字行は，行間距離が一致する



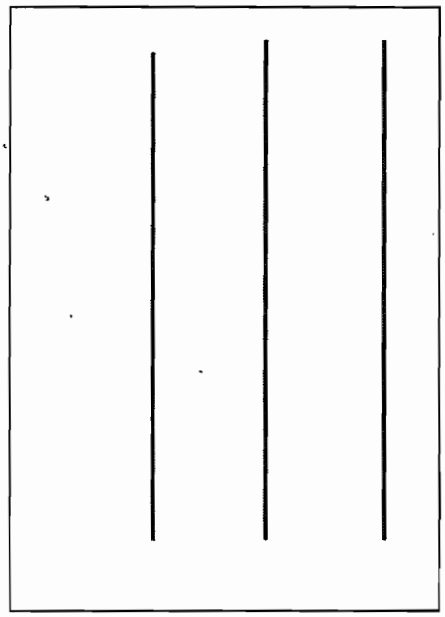
(a) 文書画像



(b) 抽出された文字行候補



(c) 境界候補点



(d) 境界の推定

図 3.8　境界候補点による境界推定の概念図

これにより前述した経験則を反映したルールとなることが期待される．

まず，入力された文字行候補に対し，形状が酷似していることや相互位置関係などから明かに同一の文章領域に属すると思われる文字行候補同士を結合し，**基礎領域**を生成する．基礎領域に対し，ルールによる結合を繰り返すことにより文章領域に相当する領域を得る．文字行から文章領域形成の結合処理途上にある領域を**中間領域**と呼ぶ．基礎領域，中間領域は共に文字行の集合であり，基礎領域が中間領域の初期値となる．中間領域の全体を $\mathcal{M}$ で表わす．このとき$\omega \in \mathcal{M}$ について，$\omega \subset \mathcal{S}^*$ である．

中間領域の結合は，文字行の持つパラメータから求められる中間領域の形状類似性と相互位置関係についての尺度を評価することにより，同一文章領域に属すると思われるもの同士を結合する．結合には，単純にパラメータを評価することによる一次結合と，同一文章領域ありながら一次結合だけでは結合されなかった可能性が高い中間領域同士を結合する二次結合とをくりかえし適用する．二次結合は，任意形状の文章領域，センタリング，分離文字行，段落の最後で文字行が短くなっている場合，などに対応することを目指している．

一次結合，二次結合を結合が生じなくなるまで繰り返し適用し，結合が生じなくなった段階で形成された中間領域が文章領域に相当すると判断して出力する．

3.3　文字行パラメータ算出

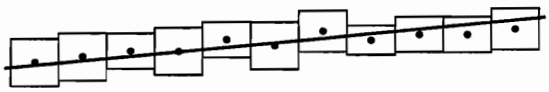


図 3.9　文字行の傾き

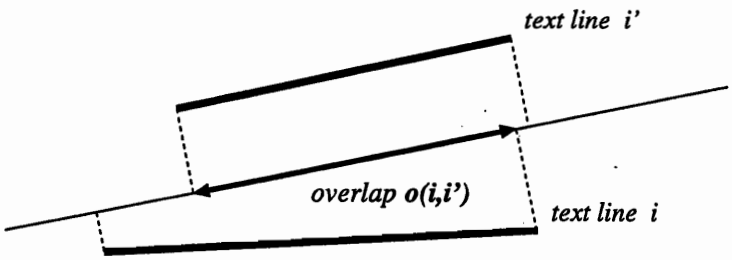


図 3.10　文字行間の重なり

区分直線連結法では，文字行を基本矩形と呼ばれる矩形の集合という形で表現している．これら基本矩形の集合から，図 3.7に示す各パラメータを算出する．まず文字行内の各基本矩形について中心点を求め，それらを結ぶ直線を最小二乗法で近似することにより文字行の傾きを求める (図 3.9参照)．ここで近似により得られた直線は文字行の芯線 (中心線) となる．文字行の高さは基本矩形の高さの平均値，文字行の長さは左端と右端の基本矩形の座標の差とする．他の文字行との相互位置関係を示すパラメータである文字行間の距離は，各々の文字行の芯線を平行に近似し，それら芯線間のユークリッド距離とする．

また文字行間の重なりは，両文字行の傾きの平均値を傾きとする直線上に，その垂直な方向に両文字行を射影したときの重なり部分の長さとする (図 3.10)．以降で示す条件式では，文字行候補 i, i' について，$s(i), l(i), h(i)$ をそれぞれ文字行 i の傾き，長さ，高さとし，$d(i,i'), o(i,i')$ をそれぞれ文字行 i, i' の 距離，重なりとする．

3.4 過接合切断部

3.4.1 境界候補点の抽出

文字行候補画像から，その行方向と垂直な方向の射影ヒストグラムを求めることにより空白部分を検出する．文字行候補 $i \in \mathcal{S}$において，検出された空白部分の長さ l_sが

$$l_s \geq K_0 \cdot h(i) \tag{3.1}$$

である場合，その空白部分の中心位置を境界候補点とする．本実験では， $K_0 = 1.2$ とした．また，ノイズによる空白部分の検知もれを防ぐためあらかじめ RLSA(Run-Length Smoothing Argolithm)[4] により射影ヒストグラムのスムージングを行う．

3.4.2 近傍文字行集合の選定

文字行 $i \in \mathcal{S}$について，その近傍文字行集合 $\mathcal{F}(i)$ を以下のように定義する．

$$\mathcal{F}(i) \equiv \left\{ i' \in \mathcal{S} \;\middle|\; \begin{array}{rrcl} & d(i,i') & \leq & K_1 \cdot h(i) \\ and & |s(i) - s(i')| & \leq & K_2 \\ and & \frac{\max(h(i),h(i'))}{\min(h(i),h(i'))} & \leq & K_3 \end{array} \right\} \tag{3.2}$$

条件はそれぞれ距離，高さ，傾きについての制限を設けている．距離についての制限により，局所的な範囲を設定している．また，高さや傾きについての制限を用いることにより，本文領域にある文字行候補の近傍文字行集合に見出しや表題が含まれることなど，別の領域の文字行候補が近傍文字行集合に含まれることを防いでいる．

しきい値の選定については，それぞれ予備実験から $K_1 = 5.0$， $K_2 = 3.0[°]$， $K_3 = 1.5$ とした．距離に関するしきい値 $K_1 = 5.0$ は注目している文字行候補から，上下それぞれ 2 ～ 3 行程度までを近傍文字行集合に含めることを目安にした値である．

3.4.3 境界候補点の射影

注目している文字行候補 i の長さと同じ長さを持つアキュムレータに，その近傍文字行集合 $\mathcal{F}(i)$ の境界候補点および端点を射影する (図 3.11参照)．過接合が生じている部分の空白部分 1 つは正しく抽出された文字行の端点 2 つに対応する．よって空白部分により得られた境界候補点は値 2 を，文字行端点は値 1 をアキュムレータに加えるものとする．これにより境界候補点射影分布を得る．

3.4.4 切断候補点の評価

境界候補点の射影により得られた境界候補点射影分布上では境界候補点の密集している部分が，文章領域の境界であることが推測される．まず，境界候補点射影分布を平滑化する．平滑化は，全幅 3 の移動平均演算子と全幅 12 の移動平均演算演算子を組合わせて

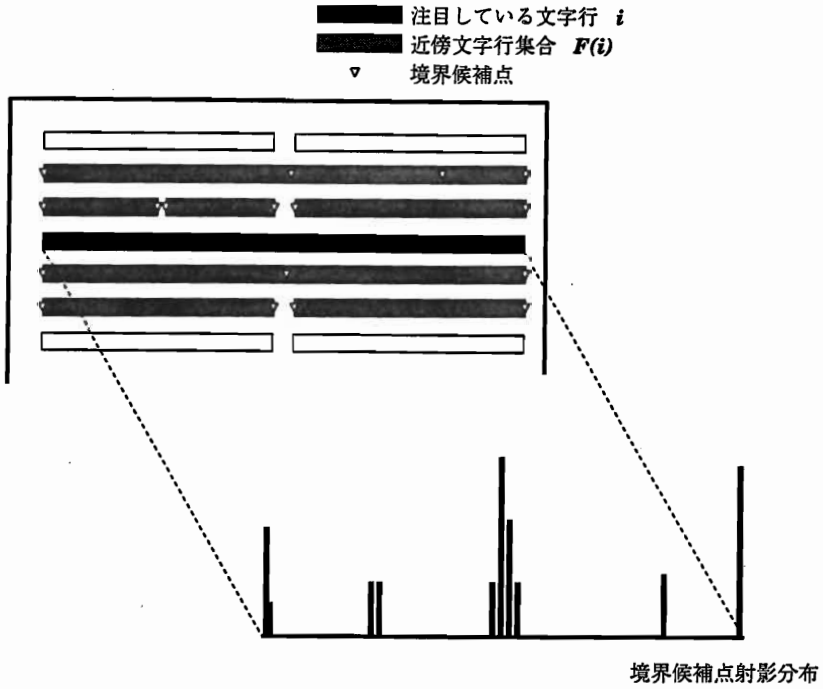
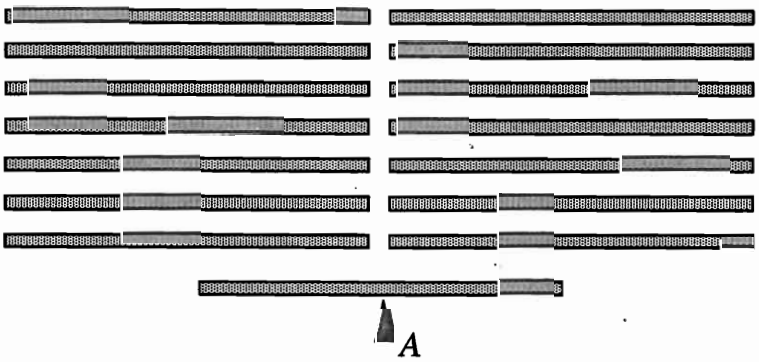
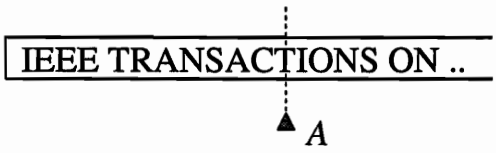


図 3.11　境界候補点の射影



(a)



(b)

図 3.12　切断候補点の評価

作用することにより行なう．境界候補点の密集している部分は，平滑化した境界候補点射影分布において高いピークをもつ部分と予想されるので，一定以上の高さをもつピークを検出し，その位置を切断候補点とする．

この切断候補点は近傍文字行集合 $\mathcal{F}(i)$ の情報のみから導出されたものである．そこで，切断候補点とされた位置に，注目している文字行候補 i 自身の空白があるかどうかを確認する必要がある．切断候補点とされた位置に空白があればそれを最終的な出力とし，ない場合はその切断候補点は却下される．

なぜこのような処理が必要かというと，例えば図 3.12(a) のような行配置があったとすると，図中 A 位置が切断候補点とされることが充分考えられる．しかし当然この位置で 2 つの文字行に分割してはならない．このよう誤りを防ぐために切断候補点の評価処理が必要となる．

3.5　文章領域生成部

3.5.1　基礎領域生成

文字行 $i \in \mathcal{S}^*$ と文字行 $i' \in \mathcal{S}^*$ の形状が類似しているとみなす条件を次に示す．

$$|s(i) - s(i')| \leq T_1 \tag{3.3}$$

$$Ratio(l(i), l(i')) \leq T_2 \tag{3.4}$$

$$Ratio(h(i), h(i')) \leq T_3 \tag{3.5}$$

$$\left(\text{但し，}\quad Ratio(x, y) \equiv \frac{\max(x, y)}{\min(x, y)} \geq 1\right)$$

これらの条件は, それぞれ原則 3.1，原則 3.2，原則 3.3 から導入されたものである．各定数の値についてであるが，理想的には，$T_1 = 0, T_2 = 1, T_3 = 1$ となるが，$s(i)$ や $h(i)$ などの各パラメータは区分直線連結法の基本矩形から算出したものであり，正しい値との間に誤差が考えられる．よってその誤差を考慮して予備実験により値を設定する．

相互位置関係の条件は次の条件とした．

$$\frac{o(i, i')}{\min(l(i), l(i'))} \geq T_4 \tag{3.6}$$

式 3.6の左辺は，両文字行の重なり部分の長さを，それぞれの文字行の長さのうち，短いほうの長さで正規化したものである．一方の文字行が他方の文字行を完全に覆っている場合左辺は 1 を示し，また両文字行に重なり部分が全く無い場合は 0 を示す．この条件により両文字行の水平方向のズレを制限している．式 3.3〜式 3.6 の条件をみたし，文字行間距離 $d(i, i')$ が最も小さいという位置関係にある文字行同士は同一の文章領域に属するとみなす．ここでは，そのような条件を見たす文字行群で，文字行の集合である基礎領域を構成する (図 3.13参照)．基礎領域は中間領域の初期値である．

基礎領域の検討

ここで得られた基礎領域は形状が類似し，かつ文字行間の距離が最近接関係にあるという条件で生成されている．つまり，基礎領域を生成する段階では，最初に挙げた原則のうち原則 3.4を用いることはできない．そのため，図に示すように行間距離が異なる文字

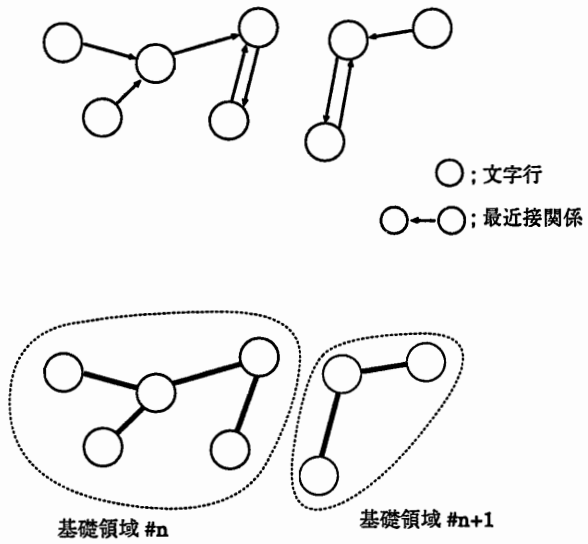


図 3.13　基礎領域生成

行が基礎領域内に含まれる可能性がある．この問題に対応するため，基礎領域が生成された段階で，領域に属する文字行間距離についての平均値および分散値から明かに文字行間距離が異なる文字行を，その基礎領域から除去し，除去された文字行は，単独で基礎領域を構成することとする．

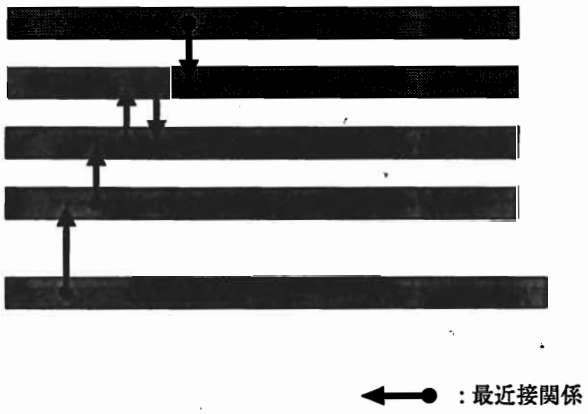


図 3.14　基礎領域生成の再確認

3.5.2　一次結合

中間領域の形状の類似性は，各々の中間領域内に含まれる個々の文字行の形状の平均で評価する．$S(\omega), L(\omega), H(\omega)$ をそれぞれ中間領域 $\omega \in \mathcal{M}$ に属する文字行の傾き，長さ，高さの平均値をとする．$N(\omega)$ を ω に属する文字行の数として，

$$S(\omega) \equiv \frac{1}{N(\omega)} \sum_{i \in \omega} s(i) \tag{3.7}$$

$$L(\omega) \equiv \frac{1}{N(\omega)} \sum_{i \in \omega} l(i) \tag{3.8}$$

$$H(\omega) \equiv \frac{1}{N(\omega)} \sum_{i \in \omega} h(i) \tag{3.9}$$

となる．これらの条件は，式 3.3～式 3.5を中間領域に拡張したものであるから，原則 3.1，原則 3.3，原則 3.2 から導入されたことになる．また原則 3.4から，両中間領域内の文字行間距離 $G(\omega)$ が類似していることも形状類似の条件に加える．文字行間距離は，同一中間領域内において，距離が最も近い文字行との間の距離とする．

$$G(\omega) = \frac{1}{N(\omega)} \sum_{i \in \omega} \min_{j \in \omega - \{i\}} d(i,j) \tag{3.10}$$

中間領域 ω と ω' の形状類似に基づく一次結合条件を以下にしめす．

$$|S(\omega) - S(\omega')| \leq T_5 \tag{3.11}$$

$$Ratio(L(\omega), L(\omega')) \leq T_6 \tag{3.12}$$

$$Ratio(H(\omega), H(\omega')) \leq T_7 \tag{3.13}$$

$$Ratio(G(\omega), G(\omega')) \leq T_8 \tag{3.14}$$

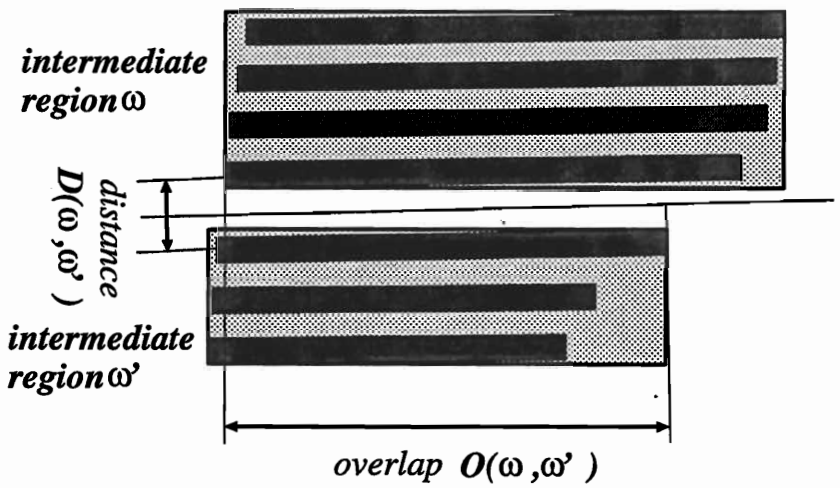


図 3.15　中間領域の相互位置関係

中間領域の相互位置関係に用いるパラメータを図 3.15に示す．相互位置関係に関する結合条件は，重なりついては基礎領域の生成のときの条件と同様のものを定義する．中間領域間の距離は，両者の間で最も接近している文字行間の距離とする．距離についての条件では，単に距離が近い，遠いというだけの判定は行なわない．2 個の中間領域が同一文章領域に属するのであれば原則 3.4より，両中間領域間の距離はそれらが属する文章領域内の文字行間距離に一致するはずである．従って，同一文章領域に属する中間領域間の距離は，各々の中間領域内の文字行間距離と一致することが期待される．

中間領域の相互位置関係についての一次結合条件を以下のように定める．中間領域領域 ω, ω'間の距離を $D(\omega, \omega')$，重なり部分の長さを $O(\omega, \omega')$ とする．

$$\frac{O(\omega, \omega')}{\min(L(\omega), L(\omega'))} \geq T_9 \tag{3.15}$$

$$\max(Ratio(G(\omega), D(\omega, \omega')), Ratio(G(\omega'), D(\omega, \omega'))) \leq T_{10} \tag{3.16}$$

式 3.15は式 3.6 と同様に中間領域の水平方向のズレを制限するルールで，式 3.16は前述したように原則 3.4 から導入されたルールである．

式 3.11～式 3.16に示した形状類似，相互位置関係の条件を満す中間領域同士を結合し，あらたな中間領域を形成する．

3.5.3　二次結合

二次結合は

- 分離文字行
- センタリング
- 段落の最後で文字行が短くなっている場合
- 任意形状の文章領域

などに対応することを目指している．これらの構造を持つ文書では，一次結合だけでは文章領域を正確に特定することが不可能となる．そのような構造をもつ文書に対し本処理

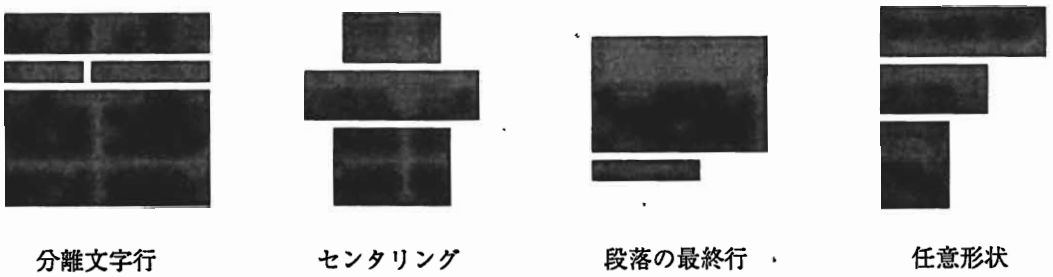


図 3.16　一次結合終了時の中間領域

を適用した場合，一次結合が終了した段階では中間領域はそれぞれ図 3.16のような状態であると考えられる．これらの中間領域を結合することをめざして二次結合の条件を考察する．

二次結合の結合条件として，一次結合の結合条件の一部を '緩和' したものを用いる．具体的には，一次結合の条件である式 3.11～式 3.16のうち，長さに関する条件と重なり (水平方向のズレ) の条件を緩和する．式 3.12の定数 T_6をより大きくし，式 3.15の定数 T_9をより小さくする．これにより以下の条件式が導かれる．

$$|S(\omega) - S(\omega')| \leq T_5 \tag{3.17}$$

$$Ratio(L(\omega), L(\omega')) \leq T_6' \quad (T_6' > T_6) \tag{3.18}$$

$$Ratio(H(\omega), H(\omega')) \leq T_7 \tag{3.19}$$

$$Ratio(G(\omega), G(\omega')) \leq T_8 \tag{3.20}$$

$$\frac{O(\omega, \omega')}{\min(L(\omega), L(\omega'))} \geq T_9' \quad (T_9' < T_9) \tag{3.21}$$

$$\max(Ratio(L(\omega), D(\omega, \omega')), Ratio(L(\omega'), D(\omega, \omega'))) \leq T_{10} \tag{3.22}$$

しかしながら，この条件で二次結合を実行すると，望ましくない結合の発生する恐れが十分にある．例えば，図 3.17に示すような場合である．この例に示した 3 個の中間領域は，別々の文章領域に属すると思われるが，ここで導入した条件式を適用すると 3 個の中間領域が同一の中間領域に統合される． この問題に対応するため，新たなルールを導入する．

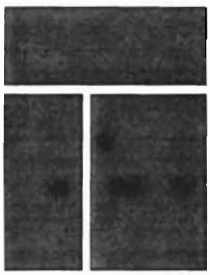


図 3.17　二次結合が望ましくない中間領域の例

図 3.16に示した例の中間領域群は同一文章領域に属すると考えられるのに対し，図 3.17に示した例は同一文章領域に属するとは考えにくい．このことを考慮にいれ，行なおうとしている二次結合が望ましいかそうでないかの判断を以下の手順で行なう．

二次結合判定アルゴリズム

STEP 1

中間領域ωとの二次結合条件をみたす中間領域の集合 $\mathcal{A}_\omega = \{\omega_0, \omega_1, \cdots\}$ を求める

STEP 2

ωとの結合スコアが最も高い$\omega^{OPT} \in \mathcal{A}_\omega$ を求める．結合スコア $S_c(\omega, \omega')$ は以下の式で計算される．

$$S_c(\omega, \omega') = \frac{\min(L(\omega), L(\omega'))}{O(\omega, \omega')} \tag{3.23}$$

STEP 3

$\forall x \in \mathcal{A}_\omega - \{\omega^{OPT}\}$ が，二次結合の重なり条件，

$$\frac{O(x, \omega^{OPT})}{\min(L(x), L(\omega^{OPT}))} \geq T_9'$$

を満たすならω と ω^{OPT} の結合を許可し，満たさないならその結合を不許可とする．

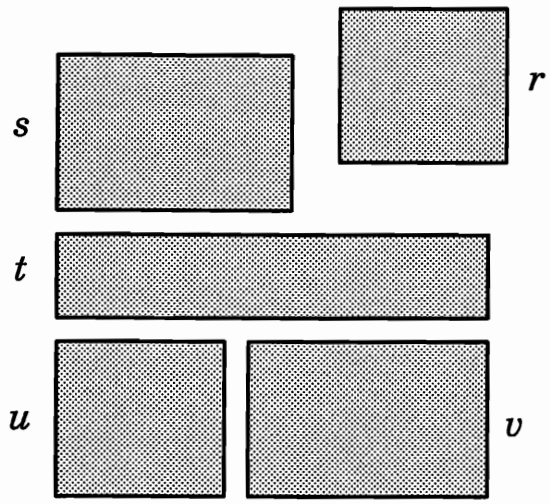


図 3.18　二次結合判定処理の例

二次結合判断処理を，図 3.18に示す r, s, t, u, v の中間領域を例に説明する．まず注目する中間領域 ω として，t を選ぶ．t との二次結合条件を満たす中間領域の集合として，$\mathcal{A}_t = \{s, u, v\}$ が決まる (STEP 1)．$\mathcal{A}_t$ のなかで，t との結合スコアが最も高いものとして，$t^{OPT} = v$ が選ばれる (STEP 2)．そして，$\forall x \in \mathcal{A}_t - \{t^{OPT}\}$ と t^{OPT} との間，つまり今の場合，s, u と vとの間において二次結合の重なり条件が確認される．s と vとの間で条件が満されたとしても，u と v との間で条件が満されることはない．よって，t と v の二次結合は許可されない (STEP 3)．

二次結合を行なったあと，新しく生成された中間領域の各パラメータを計算し，再び一次結合—二次結合を行なう．そして二次結合が生じなくなった時点で形成された中間領域が文章領域に相当すると判断し，結果として出力する．

3.6　予備実験

前述したように，本処理では文字行のパラメータを区分直線連結法の基本矩形より算出しているため，正しい値との誤差が考えられる．その誤差によって文字行の中間領域の類似性が正当に評価されないおそれがあることを考慮しなければならない．

そこでパラメータに文字行の類似性が反映されていることを確認し，同時に各結合条件の定数の目安をつけるため予備実験を行なった．調査の対象としたパラメータは “文字行の高さ”，“文字行間距離”，“文字行の傾き” である．その他のパラメータについては誤差が極めて小さいと考えられたので実験方法は，一つの文章領域画像に対して文字行抽出をおこない，そこで得られた文字行の各パラメータを算出する．そして，同一文章領域内の他の文字行のパラメータとの比較を行ない，どの程度の違いがあるかを計算するというものである．パラメータの比較は，“文字行の高さ”，“文字行間距離” については両者の

比，“文字行の傾き”については両者の差により計算される．予備実験で用いた文章領域は 30 個で，含まれていた文字行は全部で 462 個であった．図 3.19に予備実験で用いた文章領域画像と抽出された文字行の例，図 3.20に予備実験の結果を示す．

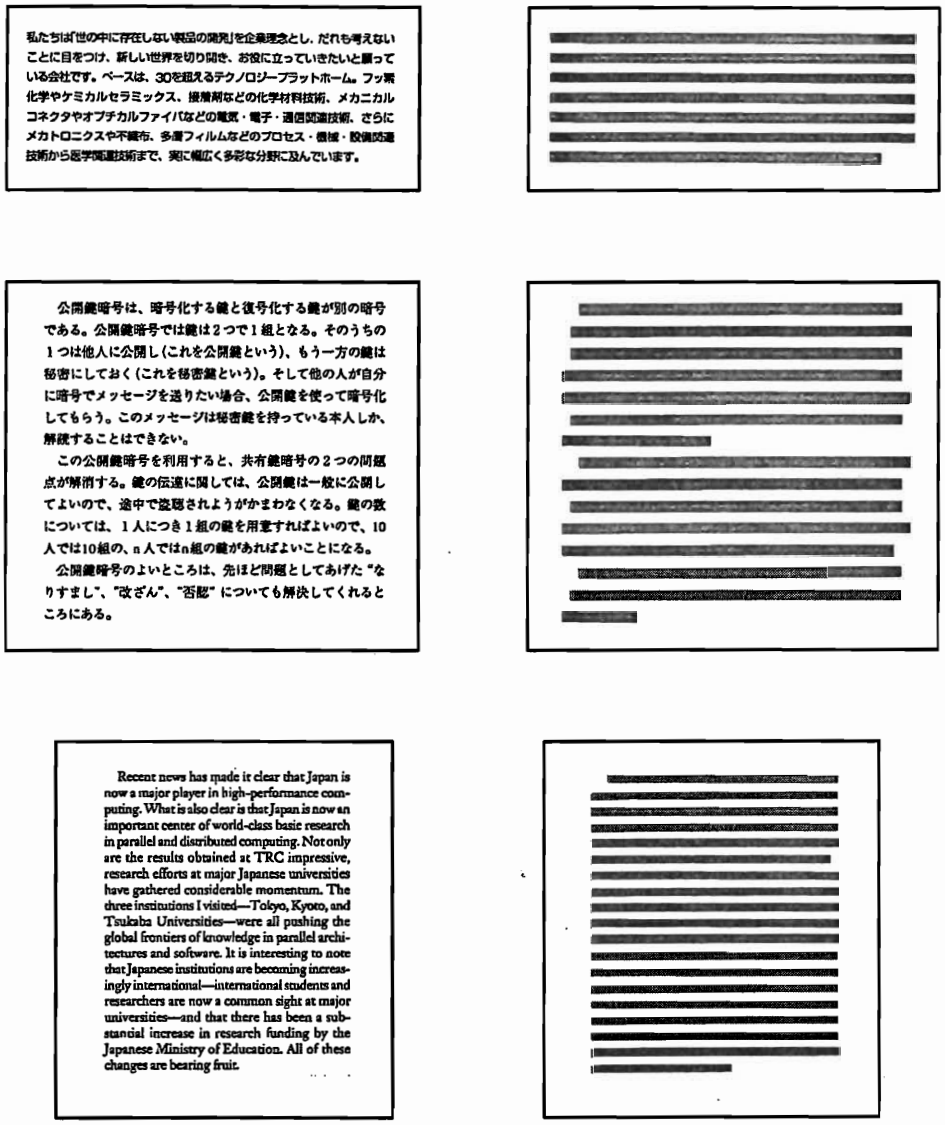
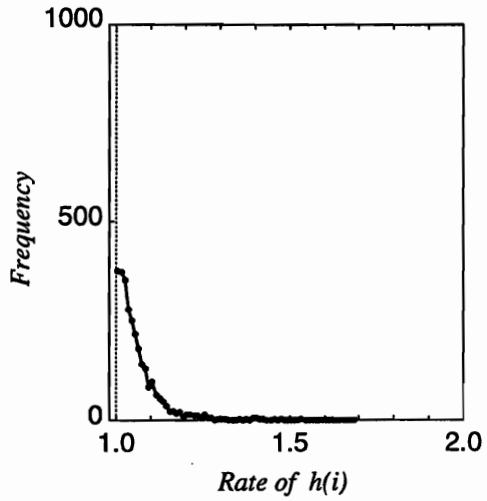
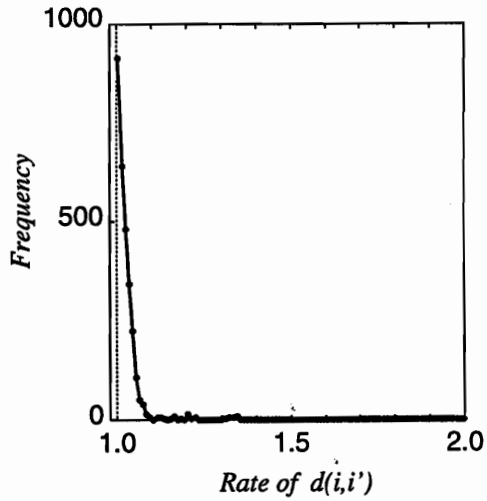


図 3.19　予備実験に用いた文章領域の例

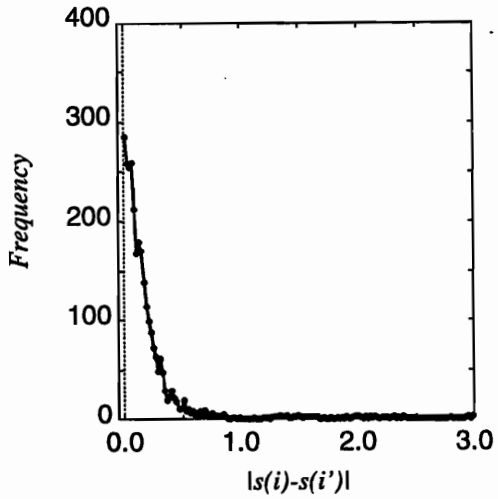
予備実験の結果から，3 種のパラメータ共，理想的な値で最も高い度数を得ることができ，文字行の類似性が文字行パラメータに反映されていることが確認された．また，結合条件の定数の目安として，高さ，文字行間距離についてはパラメータ比 が 1.3，傾きについてはパラメータ差が 1.5°を設定すればよいことがわかった．



(a) 高さの比



(b) 行間距離の比



(c) 傾きの差

図 3.20　予備実験の結果

3.7 適用例

3.7.1 過接合切断部

過接合切断部によって，文字行抽出処理において生じた過接合が切断された例を図 3.21 に示す．また，処理例の図 (a) 中の矢印で示した行 A について，その近傍文字行集合と境界候補点の射影の様子を同図 (c) に示す．過接合の生じた部分で境界候補点射影分布が高いピークをもつことが確認できる．この例では，35 箇所の過接合を正しく切断することができた．

3.7.2 文章領域生成部

文章領域生成部の処理例を図 3.22(a)，(b) に示す．(b) は文章領域中に図が入りこんでおり，文章領域が複雑に配置された文書画像で，文章領域の特定に成功した例である．

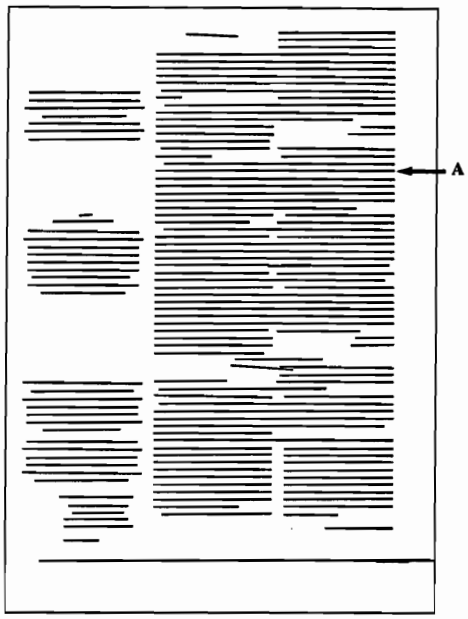
また，処理の過程を追った例を図 3.23に示す．それぞれ太枠で強調して示した部分に注目し，順を追って処理を確認する．抽出された文字行のなかで形状が酷似するもの同士で基礎領域が形成される (図中 A)．結合条件を満たす中間領域が一次結合によって結合され新たな中間領域が形成される (図中 B)．段落の最後で短くなっている文字行を二次結合によって結合する (図中 C)．この処理を繰り返し，最終結果を得る．図 3.23の文書画像では，別の文章領域の文字行との間隔のほうが同じ文章領域内の他の文字行との間隔よりも近い位置に配置されている部分がある．図 3.24の，A と B や C と D などである．このような場合，従来の近傍の黒画素と統合する方法や，近傍の文字行を統合する方法では対応できなかった．しかし今回提案した手法では，文字行の形状類似性を考慮して領域形成を行うことにより文章領域生成に成功している．

また二段組の文章領域間に，故意にノイズを加えた文書画像に対して処理を行なった例を，図 3.25に示す．文章領域間のノイズのため，A で示す文字行候補の過接合が過接合切断処理部によって切断することができなかったものである．このような文字行候補に対して文章領域生成部により生成された文章ブロックを見てみると，文字行候補 A だけがどちらの文章領域にも含まれず独立した形になっている．文字行候補 A 以外の文字行候補はそれぞれ正しい文章領域に分離している．従来の手法では図に示した文字行候補群が全て単一の文章領域として抽出さると考えられるのに対し，本手法では文字行候補 A 以外は全て正しく分割されているということで，後の処理 (意味理解を伴なうような処理) によって文字行 A が切断される可能性を残している．よって，異なる領域間の誤結合の面において処理の安定性が確認された．

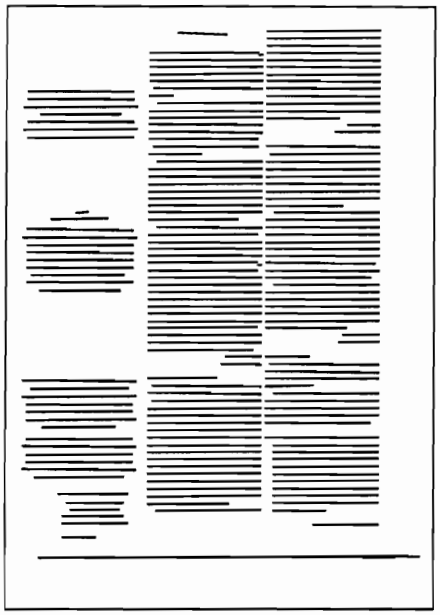
3.8 評価実験

本論文で提案した文章領域抽出処理，表領域抽出処理の有効性を確認するため評価実験をおこなった．

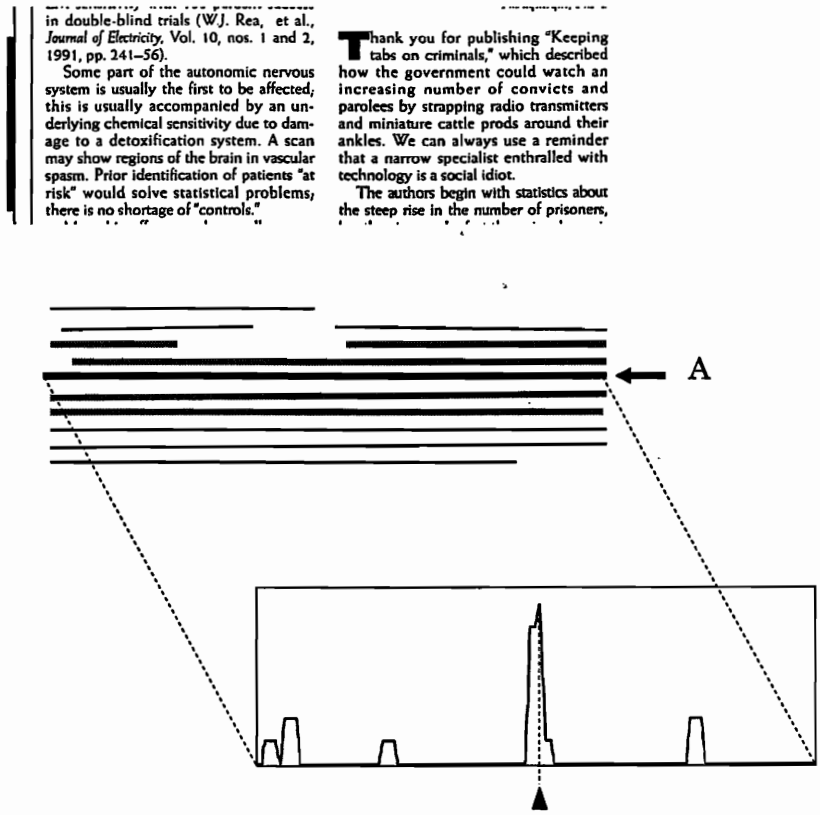
表 3.1に示す 80 ページの未知文書を用いて，文章領域抽出処理の実験を行なった．イメージスキャナにより，200[dot/inch] の解像度で読み込んだ文書画像に対し，あらかじめ区分直線連結法により文字行抽出処理を行なったものを本処理への入力データとした．各条件式の定数は，予備実験の結果を参考に $T_1 = T_5 = 1.5°$，$T_2 = T_3 = T_6 = T_7 = T_8 = T_{10} = 1.3$，$T_4 = T_9 = 0.8$，$T_6' = 5 \times T_6$，$T_9' = 0.3 \times T_9$ とした．



(a) 抽出された文字行候補

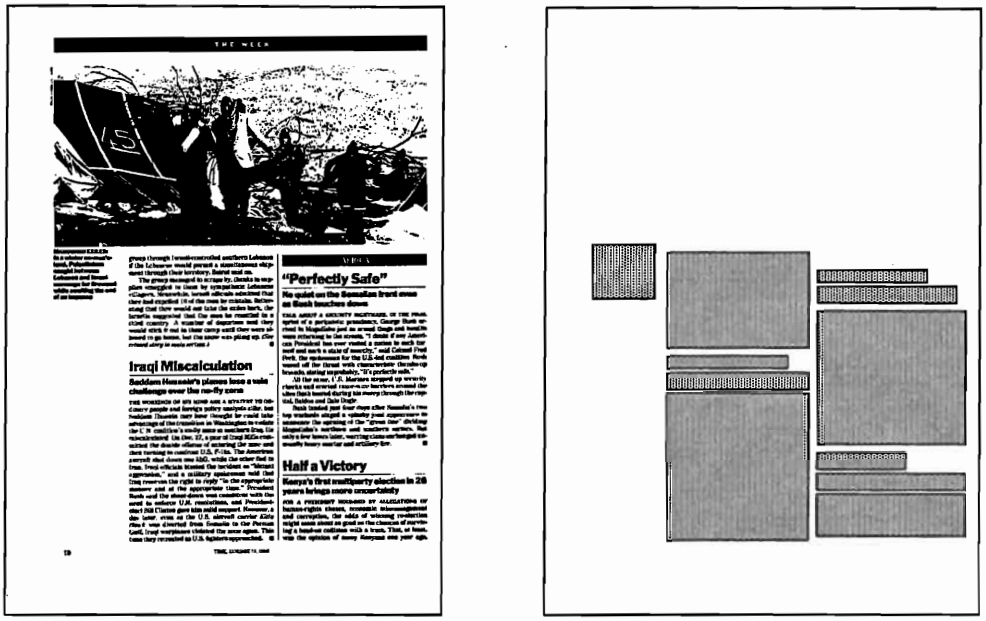


(b) 過接合切断処理後

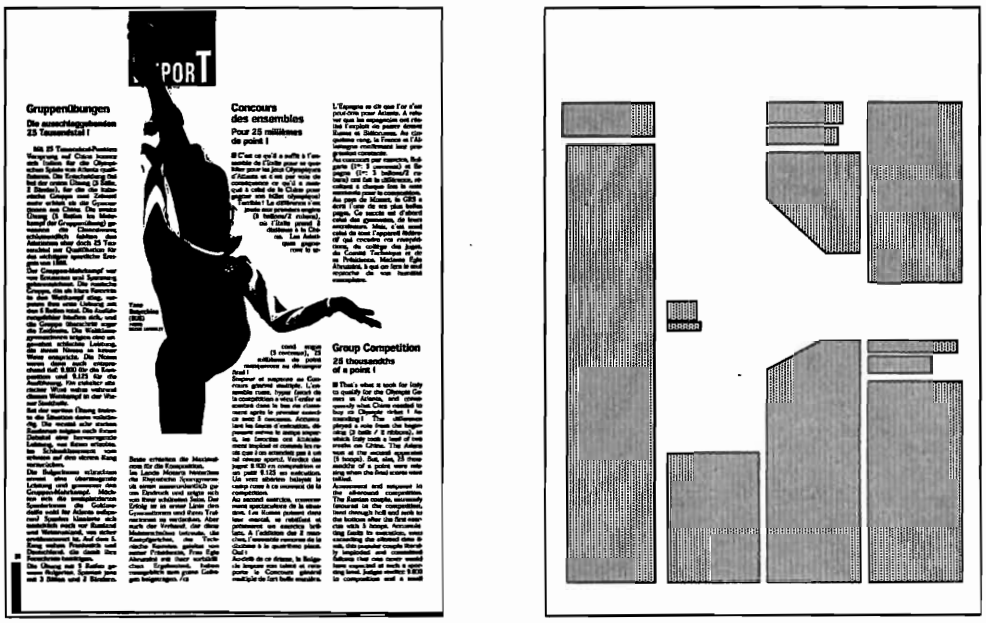


(c) 過接合切断の様子

図 3.21　過接合切断処理部の処理例

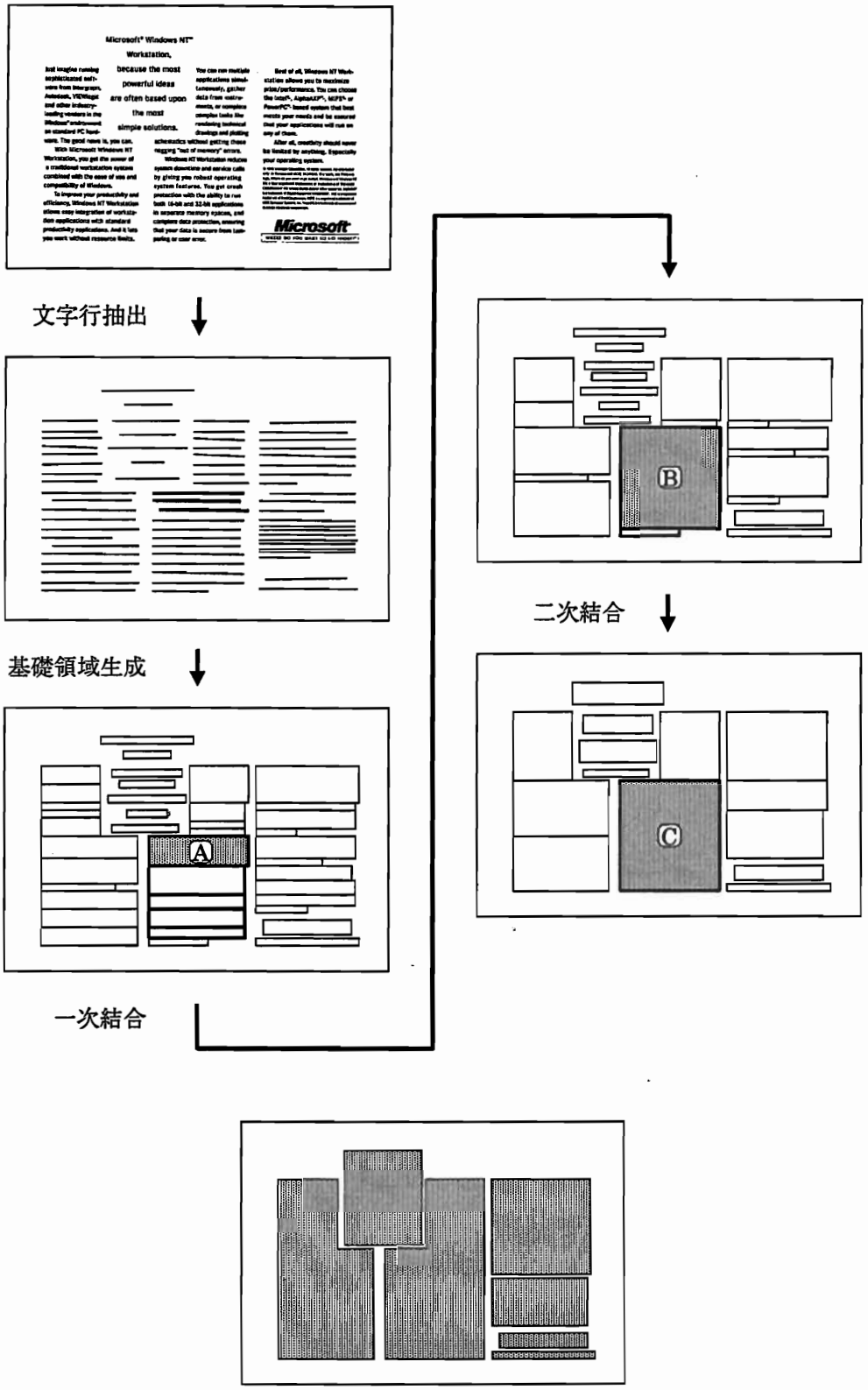


(a)



(b)

図 3.22　文章領域生成処理例



最終結果

図 3.23　文章領域生成部の処理過程例

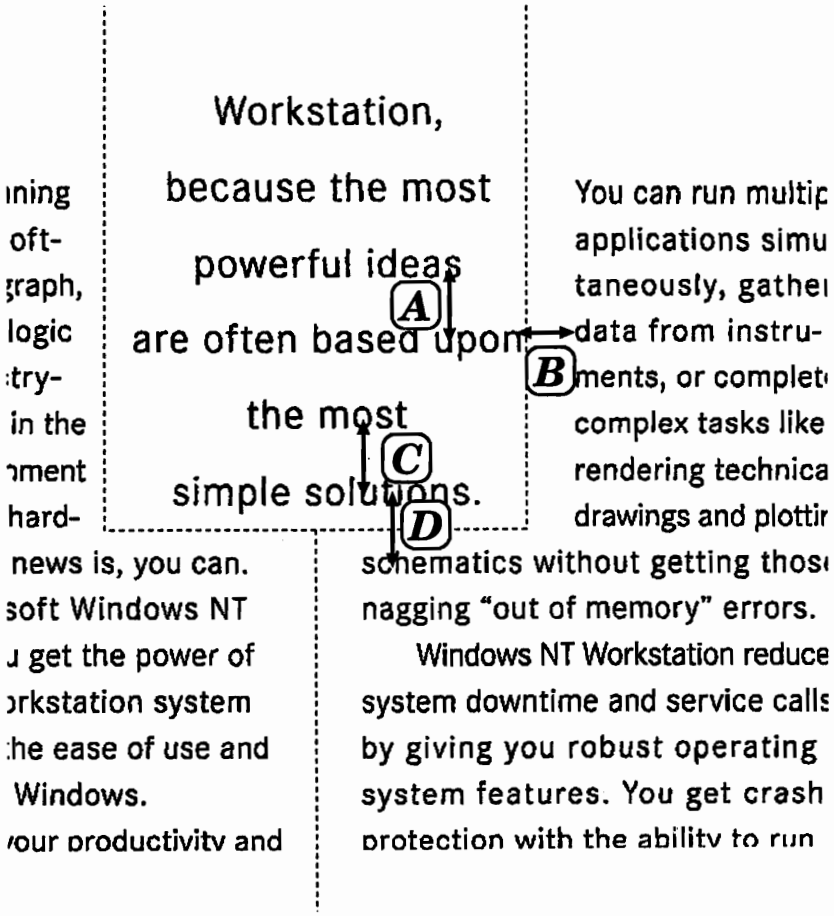
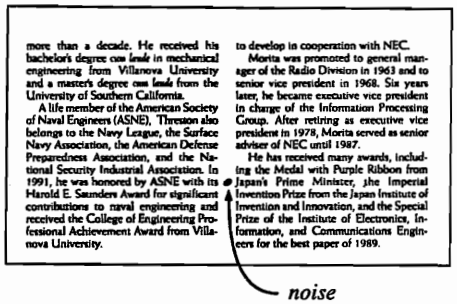


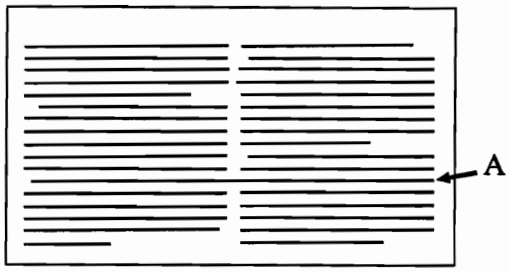
図 3.24　適用例の部分拡大

内容	サイズ	使用言語	枚数
雑誌い	B5	日本語	10
雑誌ろ	B5	日本語	10
商品カタログ	A4	日本語	10
学会予稿集	B5	日本語	10
学会論文誌	A4	英語	10
雑誌 A	A4	英語	10
雑誌 B	A4	英語	10
雑誌 C	A4	英語	10
雑誌カ	A4	韓国語	10

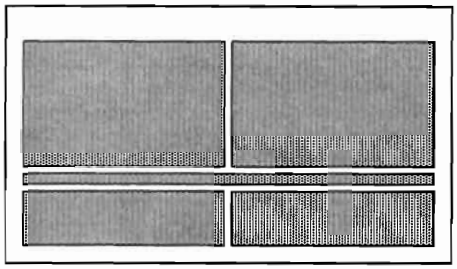
表 3.1　文章領域抽出実験に用いた未知文書



(a) ノイズを含む文書画像



(b) 過接合切断処理後



(c) 文章領域抽出後

図 3.25　切断が不可能な過接合に対する処理

全文章領域	476
抽出に成功した文章領域	460
抽出に失敗した文章領域	16

表 3.2　文章領域抽出実験の結果

評価実験の結果を表 3.2に示す．入力文書 80 ページに含まれていた全文章領域の約 97% に相当する文章領域の抽出に成功した．失敗は，本来他の文章領域に属するべき文字行候補を誤って結合したり，単一の文章領域が複数の文章領域に分割されたものなどである．抽出に失敗した部分の例を図 3.26に示すこの例においては，図中に Bで示した “カセットプレーヤー” の文字行はその上の文章領域とは分離されるべきである．しかし実際には同一の文章領域に含まれてしまった．これは，Bで示した文字行と A で示した文字行との文字行間距離が実際よりも若干小さく評価されてしまったことが原因である．本手法では，傾きや高さなどの文字行パラメータを求める際には区分直線連結法における基本矩形から算出し，それを基に文字行間距離などを求めている．区分直線連結法における基本矩形は，黒画素をある程度 “ぼかした” ものであるので，これらの算出されたパラメータと実際の値とのあいだで誤差が生じ，それが文字行間距離にも影響したのである．処理時間が増加することをいとわないならば，パラメータの算出を高精度な方法にすることによってこのような失敗を減少させることが可能であると思われる．

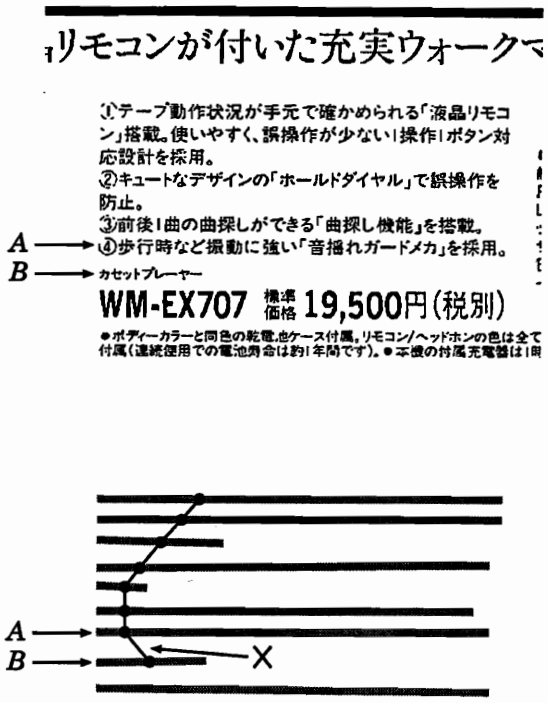


図 3.26　文章領域抽出処理の失敗例

3.9　まとめ

本章では，任意形状の文章領域抽出を目指した文章領域生成法を具体的に説明した．本手法では，文書画像から直接文字行抽出を行う場合に問題となっている過接合や分離文字行に対応するものである．

過接合に対応するために，文章領域生成部の前処理として過接合切断部を設けている．過接合切断部では文字行内の空白部分や文字行の両端点を境界候補点とし，この分布から文章領域の境界線を推定している．この推定は，近傍文字行集合を導入することにより局所的に行なっているため境界線に傾斜や湾曲のある場合にも正しい推定が可能となっている．

文章領域生成部では，文字行の形状や相互位置関係の類似性についての経験則から文字行や中間領域を結合するルールを導入した．中間領域の結合には，結合ルールをそのまま適用する一次結合と，任意形状の文章領域や分離文字行などに対応することを目指した二次結合を設けた．文字行のグループ化を徐々に拡張していくことで，local から global に至る bottom-up 的アプローチとなっている．結合ルールは傾きや高さなど複数の特徴から導入されているので，単に近距離にある文字行を結合する手法 [9][10] に比べ，異なる文章領域が接近している場合に頑健な処理が行なわれるという特徴がある．

第 4 章

表領域抽出処理

表領域は文字あるいは文字行が集中しているという点で，文章領域に似た特性を持ち，それら 2 領域が混同されないことが必要である．本章では，そのような問題を考慮に入れ，文書画像から表領域を特定する手法を提案する．

4.1 研究の背景

第 1章で述べたように，文書認識システムの認識部においては各領域に応じた認識処理が行なわれる．文章領域を認識する処理は，構造解析部において得られた文字行を単純に文字認識にてコード化すればよい．表領域は文字が密集しているという点で文章領域と似た構造を持つ．しかしながら，表領域に対して文章領域と同様に，単純に文字認識処理を行うだけでは表文書がもつ情報をすべて電子化したことにはならない．文章領域が文字の一次元的な系列により表現されるのに対し，表領域は水平方向ならびに垂直方向の文字との関係，つまり二次元的な情報が重要な意味をもつことになる．よって表を認識するということは，表を構成する項目を文字認識し，さらに表文書のもつ二次元な情報もあわせて表領域をコード化するということになる．

表形式文書を認識する表認識手法はこれまでに数々提案されいる．例えば，印刷時や文書画像読み取り時に生じる罫線のかすれやとぎれに対応する手法 [14] や，罫線が実線ではなく点線，破線などの場合に対応する手法 [15] や，表内の文字の罫線への接触に対応する手法 [16] などである．しかしここで挙げた [14][15][16] の手法は表領域を構成する文字要素 (これを項目と呼ぶ) がすべて実線あるいは点線，破線等の罫線によって区切られていることを前提としている．しかし実際の印刷書中には罫線の一部あるいは全部が省略された表も多く見られ，そのような表形式文書に対応する認識部が必要となる．

罫線が省略された場合でも項目を抽出し，認識する表認識手法が提案されている．糸乗の手法 [17] は文字の集まりを一つのブロックとしてその並びを利用することにより，罫線の省略された表文書中の項目を特定する手法を提案した．この手法は罫線の省略に対応できるものの，表中の項目の省略には対応できなかった．項目の省略や包含があったり，項目の位置ずれがあったりするような表画像に柔軟に対応することを目指した認識手法が提案されている．金津らの手法 [18] は，表の中で水平方向に並んだ文字ブロックの集合をを “行” とみなし表画像を行分割する．そして隣接する行の間で，あらかじめ定めておいた対応付けモデルを用いて文字ブロックの対応づけを行なって表構造を理解するというものである．また，平山の手法 [19] では表画像を文字ブロックのならびから垂直方向に分割しカラムを形成する．そして隣接するカラム間で文字ブロックの対応付けを決める際に DP マッチングを適用して最適な対応付けを求め，表認識を行なうものである．これらの

手法では，表内の項目を隣接する項目との間でどのように対応付けするかが大きな課題となっている．現段階においては罫線や項目の省略に完全に対応する手法として確立したものでないが，今後，さらに高精度な手法に発展することが期待される．

ここで表認識の前段階の処理として必要となる，文書画像中の表領域を特定する手法について従来どのような手法が用いられているかを考察する．田畑らのの手法 [15] では，表領域が罫線で囲まれるという仮定のもとに，文書画像の射影ヒストグラム中で 0 の無い連続部分を表領域と見なすものである．また saitoh らの手法 [10] では罫線抽出処理で文書画像から罫線を抽出しておき，水平方向の 2 個の罫線が一定距離以下にあり，その間に 1 個以上の垂直方向の罫線がある領域を表領域と見なすものである．これらの手法は表領域中の罫線の存在を前提としている．

前述したように，実存する表の中には罫線が全て省略されているものみられ，そのような表に対応する表認識の研究がすすめられている．しかしながら，表領域の抽出処理で現在提案されているものは表中に罫線が存在することを前提としており，罫線省略のある表を文書画像中から抽出して認識するという一連の処理には対応できない．そこで本章では罫線省略のある表領域を文書画像から抽出する手法を提案する．

4.2 提案手法の概要

本章で提案する表領域抽出処理のながれ図を図 4.1に示す．本手法は罫線省略のある表領域の抽出をめざしている．罫線省略の全くない表画像と全ての罫線が省略された表画像について，それぞれの外接矩形表現を比較すると，両者とも文字により形成された外接矩形の形状や配置のみで表画像であることが推定できる．そこで本手法は，あえて罫線の情報を用いずに文書画像から表領域を抽出する処理を目指す．本処理は，bottom-up 的アプローチにより，文字が規則的に配置されていると推定される局所領域を統合し表領域を生成するものである．表領域に関する知識として，以下の原則を導入する．

原則 4.1 比較的短い文字行が配置している

原則 4.2 項目の重心は水平方向，垂直方向に規則的に配置している

次に処理の概略を説明する．まず，局所的な範囲を特定するものとして，**表候補小領域**を形成し，それぞれの表候補小領域について，文字のならびを評価する**特徴点**を抽出する．そして隣接する表候補小領域との間で，表の特性を表す指標として導入する**類似度**と**特徴点密度**を計算する．類似度ならびに特徴点密度から，同一の表領域に属すると思われる表候補小領域を統合する．表候補小領域の統合が終了した段階で，表候補小領域の再生成を行なう．これは，表中の項目に省略がある場合に対応することを目指す処理である．表候補小領域の再生成が生じたら表候補小領域の統合をもう一度行ない，再生成が生じなくなるまで繰りかえす．そして統合された表候補小領域により特定される部分を表領域であると判断し，結果として出力する．

4.3 表領域抽出処理

4.3.1 前処理

与えられた文書画像に対し，黒画素連結成分の外接矩形を求める．ここで得られた外接矩形のうち，以下の条件を満たさないものは表を構成する文字による外接矩形とは考え

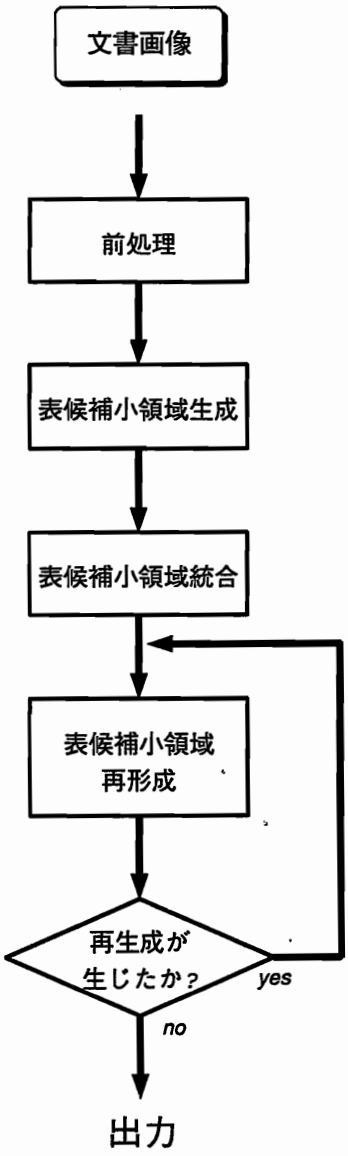


図 4.1　表領域抽出処理のながれ

られにくいので除去する．この条件により，大きするぎる文字や，罫線や，ノイズによる外接矩形が除去されることが期待される．

$$\begin{cases} Ratio(l_b(m), h_b(m)) & < & 3 \\ l_b(m) \cdot h_b(m) & < & 800 \\ l_b(m) \cdot h_b(m) & > & 10 \end{cases} \tag{4.1}$$

ただし,$l_b(m), h_b(m)$ はそれぞれ外接矩形 m の水平，垂直方向の辺の長さを表す．また，各条件式の定数は，200[dot/inch] においての画素数である．

4.3.2　表候補小領域の形成

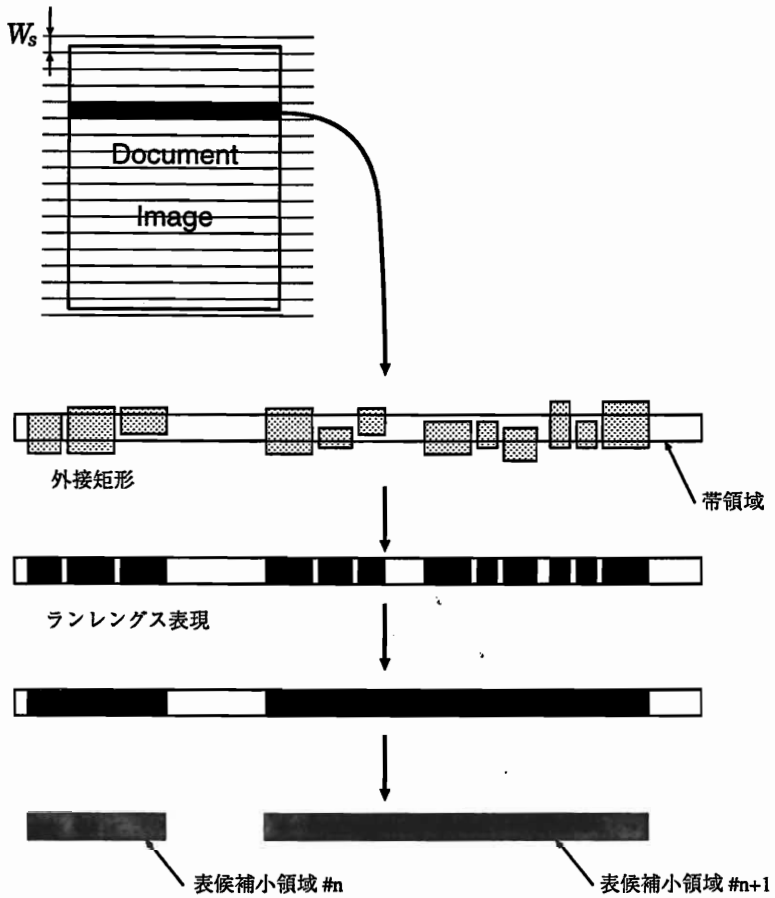


図 4.2　表候補小領域の形成

図 4.2に表候補小領域の形成の様子を示す．まず文書画像を一定間隔 W_sの水平方向スリットにより分割し帯領域を形成する．各帯領域について，領域内の外接矩形の配置をもとに表候補小領域を形成する．各帯領域内で外接矩形のある部分を 1，ない部分を 0 とすることにより，1 と 0 のランレングス表現にする．このランレングス表現に対し，定数 C_t で RLSA によるスムージングを行う．その結果得られた黒ラン 1 つに対し 1 つの表候補小領域を形成する．

4.3.3 特徴点抽出

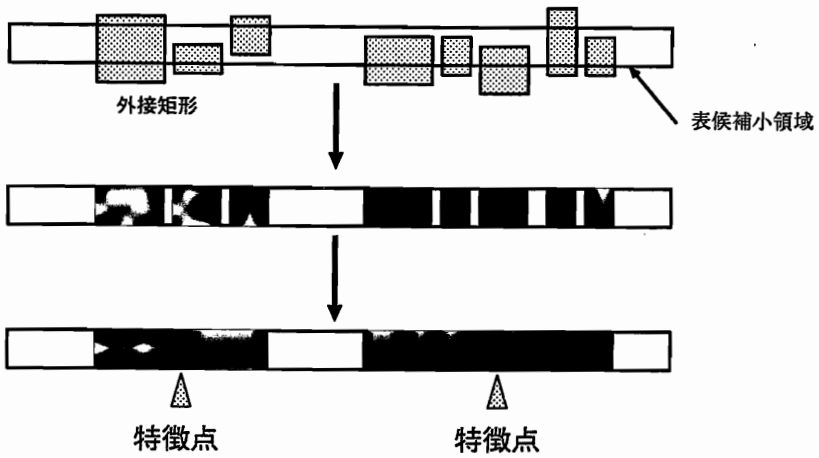


図 4.3 特徴点抽出

各々の表候補小領域に対し，表候補小領域の形成の際と同様の方法より領域内の外接矩形から得た 1，0 のランを，一定値 C_c で RLSA によりスムージングする．ここで外接矩形から得た 1，0 のラン系列を，一定値でスムージングしたものを得たが，このラン系列の黒ラン 1 つは表の項目 1 つに対応することが期待される．最初に導入した原則 4.2から，表領域を評価するためには項目の重心によりならびを評価することが有用であるといえる．したがって，黒ランの中心点を特徴点とすることにより特徴点は項目の重心となることが期待され，表領域の特性を評価する基として，有効な意味を持つと思われる．

4.3.4 表候補小領域の統合

概要

上下で接している表候補小領域間で，特徴点の配置から統合を判断する．図 4.4(a) の例で示すように，表候補小領域 A, B の特徴点 $\mathcal{C}(A) = \{\alpha_1, \alpha_2, \cdots \alpha_I\}$ と $\mathcal{C}(B) = \{\beta_1, \beta_2, \cdots \beta_J\}$ のそれぞれの対応を決める．特徴点の対応付けは，特徴点間の距離 $d_c(\alpha_i, \beta_j)$ を定義して，$\sum d_c$ が最小となるよう決定する．対応付けの決定には DP マッチング [20] を適用する．DP マッチングは，1 次元パターンについて，動的計画法により二つのパターンの要素間の対応付け (整列化) を能率良くを行ない，それに基づいてパターン間の類似度を計算するもので，従来から文字や音声のパターン認識などに適用されている．DP マッチングの特長としては，

- 最適な対応付けを，少ない計算手順で求めることができる
- 対称性がある

などが挙げられる．

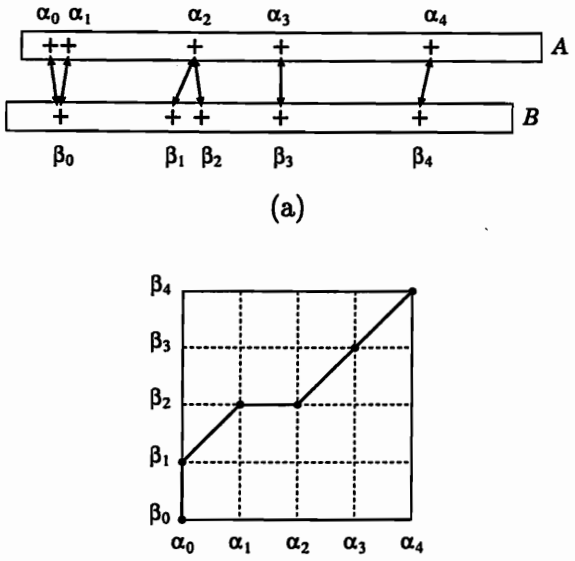


図 4.4　DP マッチングによる特徴点の対応付け

DP マッチングによる特徴点系列の対応付け

$\mathcal{C}(A)$, と $\mathcal{C}(B)$ をの比較を行う際に，DP マッチングにより特徴点の対応付けを決定する方法について述べる．$p_k = (i(k), j(k))$ が$\alpha_{i(k)}$と$\beta_{j(k)}$とを対応させることを表わすと，二つのパターンの対応は $p_k = (i(k), j(k))$ の系列$\tilde{p} = \{p_0, p_1, \cdots, p_K\}$ で表わすことができる．これを p_0 から p_Kへの経路と呼ぶことにする．経路は二次元ベクトルの系列であるから，平面上に表わすことができる．図 4.4(b) は同図 (a) のように行なった対応付けを，平面上の経路で表わしたものである．p_0 から p_Kへの経路の全体を，$P((0,0),(I,J))$ と表わす．ここで，経路 $\tilde{p}$ により，二つのパターンの $\mathcal{C}(A)$ と $\mathcal{C}(B)$ を対応付けたときの距離 $D_t(A,B;\tilde{p})$ を次のように定義する．

$$D_t(A,B;\tilde{p}) \equiv \sum_{k=0}^{K} d_c(\alpha_{i(k)}, \beta_{j(k)}) \tag{4.2}$$

だたし,$d_c(\alpha_{i(k)}, \beta_{j(k)})$ は $x_c(\alpha)$ を特徴点αの x 座標値として，以下のように定義する．

$$d_c(\alpha_{i(k)}, \beta_{j(k)}) \equiv |x_c(\alpha_{i(k)}) - x_c(\beta_{j(k)})| \tag{4.3}$$

このことから，$\mathcal{C}(A)$ と $\mathcal{C}(B)$ 間の距離を最小にする対応付けを求めることは，p_0から p_K に至る最適な経路を選ぶ問題に変型することができる．したがって，表候補小領域 A, B の類似度 $D_t(A,B)$ は

$$D_t(A,B) \equiv \min\{D_t(A,B;\tilde{p}) | \tilde{p} \in P((0,0),(I,J))\} \tag{4.4}$$

とおくことができる．

次に $D_t(A,B)$ を計算する方法を考える．まず経路 $\tilde{p} \in P((0,0),(I,J))$ に対して，

$$g(i,j;\tilde{p}) \equiv \sum_{k=0}^{K} d_c(\alpha_{i(k)}, \beta_{j(k)}) \tag{4.5}$$

とおく. そして

$$g(i,j) \equiv \min\{g(i,j;\tilde{p})|\tilde{p} \in P((0,0),(I,J))\} \tag{4.6}$$

と定義すると,

$$g(i,0) = d_c(\alpha_0,\beta_0) + d_c(\alpha_1,\beta_0) + \cdots d_c(\alpha_i,\beta_0) \tag{4.7}$$

$$g(0,j) = d_c(\alpha_0,\beta_0) + d_c(\alpha_0,\beta_1) + \cdots d_c(\alpha_0,\beta_j) \tag{4.8}$$

$$d_c(I,J) = D_t(A,B) \tag{4.9}$$

が導かれる. 一般に有限集合 Sが, その部分集合の和集合として,

$$S = S_1 \cup \cdots \cup S_M \tag{4.10}$$

と表わされるとき, S上の実数値関数 f に対して,

$$\min\{f(x)|x \in S\} = \min\{\min\{f(x)|x \in S_1\}, \cdots, \min\{f(x)|x \in S_M\}\} \tag{4.11}$$

が成立する. よって,

$$W_1(i,j) = \{p_0,\cdots,p_L,(i,j)|(p_0,\cdots,p_L) \in P((0,0),(i-1,j))\} \tag{4.12}$$

$$W_2(i,j) = \{p_0,\cdots p_L,(i,j)|(p_0,\cdots p_L) \in P((0,0),(i-1,j-1))\} \tag{4.13}$$

$$W_3(i,j) = \{p_0,\cdots p_L,(i,j)|(p_0,\cdots p_L) \in P((0,0),(i,j-1))\} \tag{4.14}$$

とおくと, $P((0,0),(i,j)) = W_1(i,j) \cup W_2(i,j) \cup W_3(i,j)$ であるから, 以下の式が導かれる.

$$\begin{aligned} g(i,j) &= \min\{g(i,j;\tilde{p})|\tilde{p} \in P((0,0),(i,j))\} \\ &= \min\left\{\begin{array}{l} \min\{g(i,j;\tilde{p})|\tilde{p} \in W_1(i,j)\}, \\ \min\{g(i,j;\tilde{p})|\tilde{p} \in W_2(i,j)\}, \\ \min\{g(i,j;\tilde{p})|\tilde{p} \in W_3(i,j)\} \end{array}\right\} \\ &= \min\left\{\begin{array}{l} g(i,j-1) + d_c(\alpha_i,\beta_j), \\ g(i-1,j-1) + d_c(\alpha_i,\beta_j), \\ g(i-1,j) + d_c(\alpha_i,\beta_j) \end{array}\right\} \\ &= d_c(\alpha_i,\beta_j) + \min\left\{\begin{array}{l} g(i-1,j), \\ g(i-1,j-1), \\ g(i-1,j) \end{array}\right\} \end{aligned} \tag{4.15}$$

これにより $g(I,J)$ が計算することができ, $D_t(A,B)$ が求められる.

類似度と特徴点密度

表候補小領域を統合する再に，各々の表候補小領域の‘表領域らしさ’を評価する指標として 2 つの指標を導入する．

原則 4.2から，表候補小領域の特徴点配置が類似していることを評価する指標として，A, B 間の類似度 $S_t(A, B)$ を次のように定義する．

$$S_t(A, B) \equiv \frac{D_t(A, B)}{K} \tag{4.16}$$

表候補小領域間の類似度は，対応付けた際の各特徴点間の距離の和 $D_t(A, B)$ を対応付けの個数 K により正規化したもので与える．

また，原則 4.1から表候補小領域の面積は大きいが特徴点の個数が少ないものは表の一部とはみなされにくく，逆に小さい面積の表候補小領域内に多くの特徴点が含まれるものは表の一部の可能性が高いと考えられる．A, B の**特徴点密度** $L_t(A, B)$ を導入し，以下のように定義する．

$$L_t(A, B) \equiv \frac{l_t(A), l_t(B)}{2 \cdot K} \tag{4.17}$$

ただし，$l_t(A)$ は 表候補小領域 A の水平方向の長さを示す．この指標を統合条件に用いることにより，例えば多段組の構造を持つ文章領域おいて，特徴点の配置が類似していても表領域の一部とは考えられにくい表候補小領域の統合を防ぐことが期待される．

4.3.5 表候補小領域の再形成

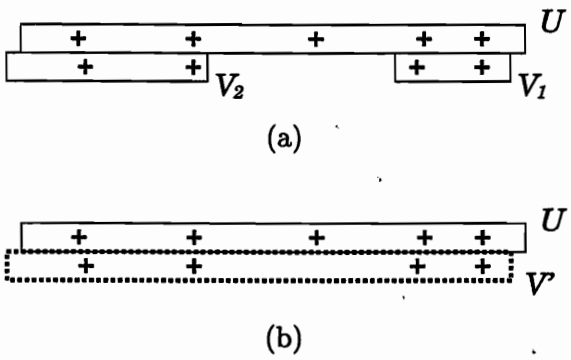


図 4.5 表候補小領域の再形成

様々な表のなかには，罫線だけでなく項目が省略されているものがありそれらに対応する必要があることはすでに述べた．では表中の項目に省略がある場合，本処理にどのような影響があるかを考察する．表候補小領域の特徴点は表中の項目に対応することが期待されるので，項目に省略があった場合は，その項目に対応する特徴点が欠落すると思われる．しかし，表候補小領域内で特徴点が欠落するだけではなく，同一の表領域内にありながら項目の省略部分による分離により別々の表候補小領域を形成する場合も考えられる (図 4.5(a))．このような場合，表領域を水平方向に分割する形で形成されている表候補小領域群 (図 4.5(a) における V_1, V_2) は，特徴点の配置から単一の表候補小領域に統合されることが望ましい．

そこで，ある表候補小領域の上辺あるいは下辺に接する表候補小領域が 2 つ以上ある場合，それらの表候補小領域の統合 (これを表候補小領域の再形成と呼ぶ) を行うかどうかを判断する対象とする．この処理を，図 4.5を例に説明する．まず，上辺あるいは下辺に接する複数の表候補小領域 V_1, V_2 を，仮に統合して 1 つの表候補小領域 V'とする．V'には V_1, V_2の特徴点がそのままコピーされる．ここで仮に形成された表候補小領域と，複数の表候補小領域に接していた表候補小領域 Uとの間で表候補小領域の統合条件を満すかどうかを確認する．もし条件を満すなら V_1, V_2を削除し，新たに V'を形成し，さらに U と V' 統合する．

項目の省略は，ある程度連続する場合も考えられる．よって，表候補小領域の再形成は再形成が生じなくなるまで繰り返し実行する．

4.4 文章領域抽出処理との統合

これまでで述べた表領域抽出処理は，複雑な構造の文書画像からの表領域抽出が困難であるという問題点がある．これは，表領域形成の基本単位となる表候補小領域を，黒ランを RLSA でスムージングするという方法により生成しているため，スムージング定数の選び方によっては，表領域の一部分と文章領域の一部分が同一の表候補小領域に含まれるということが生じるのである．複雑な構造をもつ文書画像は，その複雑さの主たる原因は文章領域が複雑であることと考えられる．そこで，表領域抽出処理に第3章で提案した文章領域抽出処理を統合することにより，複雑な構造をもつ文書画像に対応する表領域抽出処理が期待される．

そこで，入力された文書画像に対し，文章領域抽出処理により文章領域であると判断された部分を文書画像から除去する．これにより複雑な構造がある程度解消され，前述したような，文章領域と表領域とが同一の表候補小領域に含まれるという問題が解決される．

4.5 予備実験

4.5.1 概要

本章で表候補小領域の統合判断のために導入した 2 つの指標 (類似度，特徴点密度) について，それらが表領域と文章領域を区別するのに有効であるかどうかを確認する予備実験を行なった．実験方法は，表領域のみの画像と文章領域のみの画像について，本処理の特徴点抽出までを行ない，上接あるいは下接関係にある表候補小領域間において，2 つの指標を算出しその分布を調査するというものである．実験に用いたのは，表領域画像 20 個，文章領域画像 40 個である．なお文書をスキャナで読み取る際の解像度は 200[dot/inch] である．

4.5.2 結果

予備実験の結果を図 4.6に示す．両軸の数値は，200[dot/inch] における値を示している．この結果から，表領域は文章領域に比較して類似度，特徴点密度共に小さい値をとることが確認された．文章領域の多くは 2 つの指標とも大きい値をとっている．一部，多段組構造をもつ文章領域は，類似度が小さい値をとるものもあるが，特徴点密度が大きい値をもっているため，表領域との区分は可能である．

2 つの指標を用いることにより図 4.6(a) 中の直線 Xで表領域と文章ブロックの特性をほぼ分離できることが確認された. この結果から, 表候補小領域 A, Bを統合する条件を以下のように導入した.

$$L_t(A,B) < -2 \times S_t(A,B) + 100 \tag{4.18}$$

4.6 評価実験

本章で提案した表領域抽出処理の有効性を確認するため評価実験を行なった. 論文誌 (日本語) から表領域を含む 20 ページを選び, それらをイメージスキャナにより 200[dot/inch] の解像度で読み込み, 実験の入力文書画像とした.

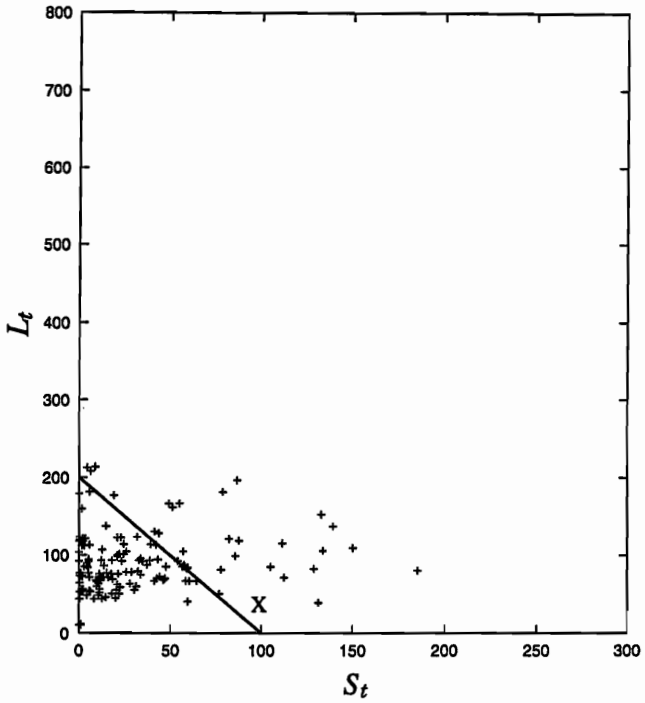
水平方向スリットの間隔 W_sは, 200[dot/inch] において, 40[dot] とした. これは標準的な大きさの文字行 (11[pt])1〜2 行程度が含まれることを想定した値である. RLSA の定数は, C_t =120[dot], C_t =16[dot] とした. また, 表候補小領域の統合条件は, 予備実験から得られた式 4.18の条件を用いた.

以上の条件の下で評価実験を行なったところ, 入力文書画像 20 ページの全てにおいてそれぞれの表領域をほぼ特定することに成功した. 処理例を図 4.7に示す. 図 4.7(a) の文書画像に対し, 表候補小領域の形成ならびに特徴点抽出を行なった結果が同図 (b) である. それぞれの表候補小領域中で, 表の項目に対応する位置で特徴点が得られていることが確認される. 最終結果が同図 (c) である.

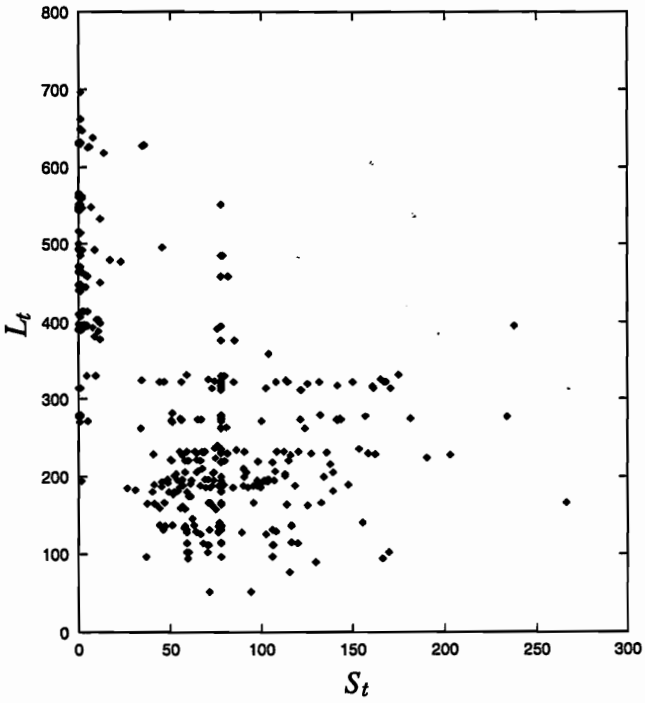
4.7 まとめ

本章では, 罫線の省略された表領域を抽出することをめざして, 文書画像の罫線の情報を用いることなく, 文字のならびから表領域を特定する手法を提案した. 表領域の抽出は, 局所領域である表候補小領域を設定し, これを統合するという bottom-up 型のアプローチで行なわれる. 表候補小領域について, 類似度と特徴点密度という 2 つの指標を評価し, 表領域の特性をもつならばその表候補小領域を統合する. 予備実験の結果, この 2 つの指標により表領域の特性が表われることが確認された. この指標からの統合によって得られた領域が表領域であると判断して結果とする. また, 罫線だけでなく項目が省略されていることを考慮にいれた処理も導入している. この処理は項目の省略によって複数に分離したと思われる表候補小領域群について表候補小領域の再形成をおこない統合処理を行うというものである. さらに, 表候補小領域を形成する際に表領域と文章領域にまたがるような領域が形成されるのを防止するため, 文章領抽出処理を基に文章領域を除去した後に表候補小領域の形成を行なう.

提案した手法に対し 20 ページの表領域を含む文書画像で評価実験を行なったところ, すべての文書画像で表領域の位置をほぼ特定することが確認された.



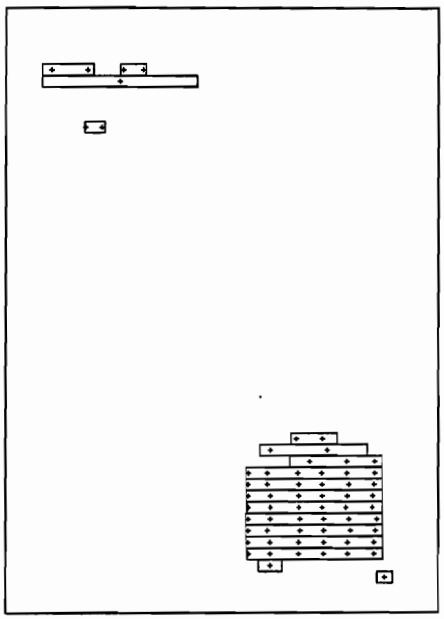
(a) 表領域



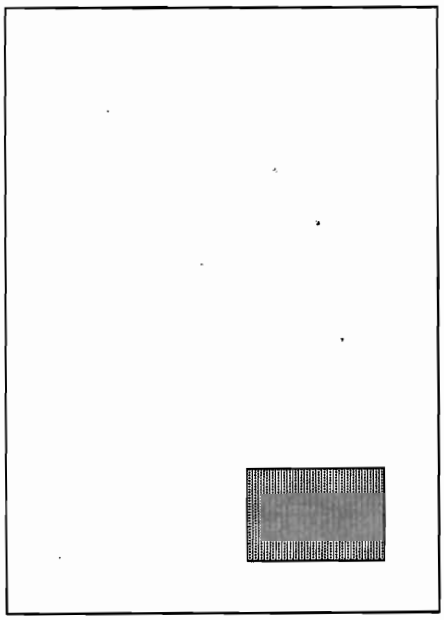
(b) 文章領域

図 4.6　予備実験の結果

(a) 文書画像



(b) 表候補小領域と特徴点



(c) 抽出結果

図 4.7　表領域抽出処理の処理例

第 5 章

結論

5.1 本論文のまとめ

本論文での成果を以下にまとめる.

まず第 1章, 第 2章では情報の電子化の必要性について触れ, その中で印刷文書を電子化することが求められていることを述べた. そして, 印刷文書を電子化する文書認識システムの全体像と, システム構成において, 現在大きな課題となっていることが複雑な構造の文書への対応であることを述べ, そのような課題に対応する文書認識システムの構成を本研究の目的とした. 研究の対象としては, 構造解析部で不可欠となる領域分割処理を取りあげた. そして文書画像の構造解析に現在用いられている手法をグループ分けして, それぞれの手法の特性について考察し, 本研究で構成する領域分割手法の方針を文書画像の局所的な幾何特徴に注目する bottom-up 型アプローチとすることを決定した.

第 3章では, 文書のなかで最も大きな位置を占める文章領域の抽出処理を提案した. 前半では文書画像から直接文字行抽出を行なう際にしばしば問題となっている文字行の過接合に対応する手法として, 過接合切断部を導入した. 複雑な構造をもつ文章領域においては, 領域を区分する要素がかならずしも直線で表現できるとは限らず, 曲線である場合も見られる. そのような文書画像に対応するため, 抽出された文字行候補およびその空白部分から領域の境界部分であることが推測される境界候補点を抽出する. そして, 局所的な範囲を特定するために定めた近傍文字行集合における境界候補点の配置から, 文字行候補中で過接合の生じている部分を推定する. また同章後半で導入した文章領域抽出部は, 抽出された文字行候補に対し, 文章領域に相当するグループを生成するというものである. この処理は, 多くの文書画像に共通する基本的なルールとして, 文書中の文字行の配置についての経験則によって, 文字行から文章領域を生成するルールを導入している. 本手法は文字行を統合して文章領域を生成しているので, 任意形状の文章領域を抽出することができる. また文字行の形状や相互位置関係の類似性に着目することにより, 異なる領域が接近する場合も両文章領域分離が可能となっている. さらに本手法は, 文書画像から直接文字行抽出を行なった際に問題となる, 分離文字行や過接合に対応するものとなっているという点が従来手法と異なるところである. 提案手法に対し評価実験を行なったところ, 約 95%の文章領域抽出に成功し, 本手法の有効性が確認された.

第 4章では文章領域につづいて大きな位置を占める表領域抽出処理を提案した. 従来, 表領域は罫線により成り立っているというのが大前提であったが, 罫線の省略に対応する認識処理が提案されている中, それに対応する表領域特定処理が必要であるという観点から, 罫線の情報を用いずに文字のならびから表領域を推定する処理を提案した. 局所領域として導入した表候補小領域間で, '表らしさ' を評価する類似度と特徴点密度という 2 つの指標を導入た. そして予備実験により 2 つの指標が表領域の特性を評価する指標として

有効であることが確認された．この結果をもとに表候補小領域の統合条件を設定し，評価実験を行なったところ，本手法の有効性が確認された．

図領域や写真領域についてであるが，文章領域や表領域が特定された後に，それらを文書画像から除去することにより得られる文書画像は，図領域や写真領域のみの十分に簡単な構造の文書画像となっている．このような文書画像から図領域や写真領域を領域分割することは周辺分布や線密度特徴 [7] 等を用いて容易に実現可能であると考えられる．

5.2 今後の課題

今後の課題として，縦書と横書の混在する文書への対応が挙げられる．本論文で提案した文章領域抽出処理は文章領域生成の際に文字行の傾きを考慮している．よって，縦書と横書の混在する文書からの文字行抽出処理がある程度正確に行なわれるならば，そのような文書に対応できる文章領域抽出処理手法である．文字行抽出処理においては，以前から“文字間の距離は文字行間の距離よりも小さい”とい前提が用いられていたが，この前提は必ずしも成立せず，このようなことから縦書と横書の混在する文書からの文字行抽出処理は困難な問題となっている．

文章領域抽出処理との統合処理によりこの問題が解決されるのではないかと思われる．例えば以下のような方法が考えられる．正立した文書画像と，90°回転させた文書画像のそれぞれについて単一方向の文字行抽出処理を行ない，それぞれ文章領域抽出処理を行なう．そしてある領域について，両方向の文字行候補のうちより安定した文章領域が生成された方をその領域の文字行の方向とする，というものである．

縦書と横書の混在する文書の認識は今後有効な手法が期待される．

参考文献

[1] Y. Y. Tang, C. D. Yan and C. Y. Suen: "Document processing for automatic knowledge acquisition", IEEE Trans. Knowledge& Data Eng., **6**, 1, pp. 3–21 (1994).

[2] 山下, 天野："モデルに基づいた文書画像のレイアウト理解", 信学論, **J75–D–II**, 10, pp. 1673–1681 (1992).

[3] 山岡, 岩根, 岩城："パターン分類手法に基づく文書画像の構造解析", 信学論, **J79–D–II**, 5, pp. 756–764 (1996).

[4] K. Wong, R. Casey and F. Wahl: "Document analysis system", IBM J.Research and Development, **6**, 6, pp. 647–656 (1982).

[5] H. Baird, S. Jones and S. Fortune: "Image segmentaion using shape-directed covers ", Proc. 10th Int'l Conf. Pattern Recognition (ICPR), IEEE CS Press, pp. 820–825 (1990).

[6] 秋吉, 黄瀬, 高松, 福永："空白の構造に基づく文書画像の領域分割", 信学技報, PRU94–101 (1995).

[7] 秋山, 増田："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", 信学論, **J69–D**, 8, pp. 1187–1196 (1986).

[8] L. O'Gorman: "The document spectrum for page layout analysis", IEEE Trans. Pattern Anal.& Mach. Intell., **15**, 11, pp. 1162–1173 (1993).

[9] S. Tsujimoto and H. Asada: "Major components of a complete text reading system ", Proc. IEEE, **80**, 7, pp. 1133–1149 (1992).

[10] T. Saitoh, T. Yamaai and M. Tashikawa: "Document image segmentation and layout analysys", IEEE Trans. Inf.& Syst., **E77–D**, 7, pp. 778–784 (1994).

[11] 角谷, 木村, 下平, 奥村："矩形レイアウトモデルに基づく文書画像の領域識別", 信学技報, PRU93–82 (1993).

[12] 平山："複雑なカラムをもつ文書イメージの領域分割法", 信学論, **J79–D–II**, 11, pp. 1790–1799 (1996).

[13] 後藤, 阿曽："文字行の局所的な直線性を利用した頑健・高速な文字行抽出法", 信学論, **J78–D–II**, 3, pp. 465–473 (1995).

[14] 佐藤, 井上, 鳥生：“文書入力のための表構造の認識”, 昭 63 信学会春季全大, D–232 (1988).

[15] 田畑, 鶴岡, 木村, 三宅：“表の構造理解のための罫線抽出と領域分け”, 信学技報, PRU90–73 (1990).

[16] O. HORI and D. S. DOERMANN: “Table-form structure analysys based on box-driven reasoning”, IEICE Trans., **E79–D**, 5, pp. 542–547 (1996).

[17] 糸乗：“文字ブロックの並びを考慮した表構造の認識”, 1993 信学会春季全大, D–410 (1993).

[18] 金津, 阿曽：“表の意味的特徴を考慮した表構造認識手法”, 平 6 電気関係東北連大, 2D12 (1994).

[19] 平山：“DP マッチングを用いた表形式データの解析方法”, 1995 信学会ソサイティ大, D–202 (1995).

[20] 上坂, 尾関：“パターン認識と学習のアルゴリズム”, 文一総合出版社 (1990).

学会発表

1. 塚田 仁志，後藤 英昭，阿曽 弘具: “多段組の文書画像における文字行抽出処理の高精度化”, 平成 7 年度 電気関係学会東北支部連合大会 **2I21** (1995–08).
2. 塚田 仁志，後藤英昭，阿曽 弘具: “類似文字行の結合による文章ブロックの抽出”, 1996 年 電子情報通信学会総合大会 **D–457** (1996–03).
3. 塚田 仁志，後藤 英昭，阿曽弘具: “文字行候補からの文章ブロック生成アルゴリズム”, 信学技報 **PRMU96–106** (1996–12).

謝辞

東北大学工学部 阿曽弘具教授，ならびに東北大学情報処理教育センター助手 後藤英昭博士には3年間にわたり御指導いだたきました．両先生には最大の敬意をもって感謝致します．

東北大学工学部 伊藤貴康教授，同 西関隆夫教授には本論文をまとめるにあたり貴重な御意見をいただきました．また，東北大学工学部 大町真一郎博士，黒岩丈介博士，森大毅氏には日頃から御討論をしていただき，貴重な御助言をいただきました．阿曽研究室の方々には，快適な計算機環境を整えていただいたほか，プログラミング上での問題解決に御協力いただきました．皆様に感謝申しあげます．

最後に，私の大学生活を理解し支えてくれた両親に感謝します．

修士学位論文 本審査会 OHP 資料

文書画像の領域分割に関する研究

1997 年 2 月 18 日

東北大学大学院工学研究科 電気・通信工学専攻
塚田 仁志

–第 1 章– 序論

背景

膨大な情報を効率的に管理→ 印刷文書の自動電子化

図, 写真等の混在する文書の電子化
··· 文書認識システム

文書認識システムにおける課題

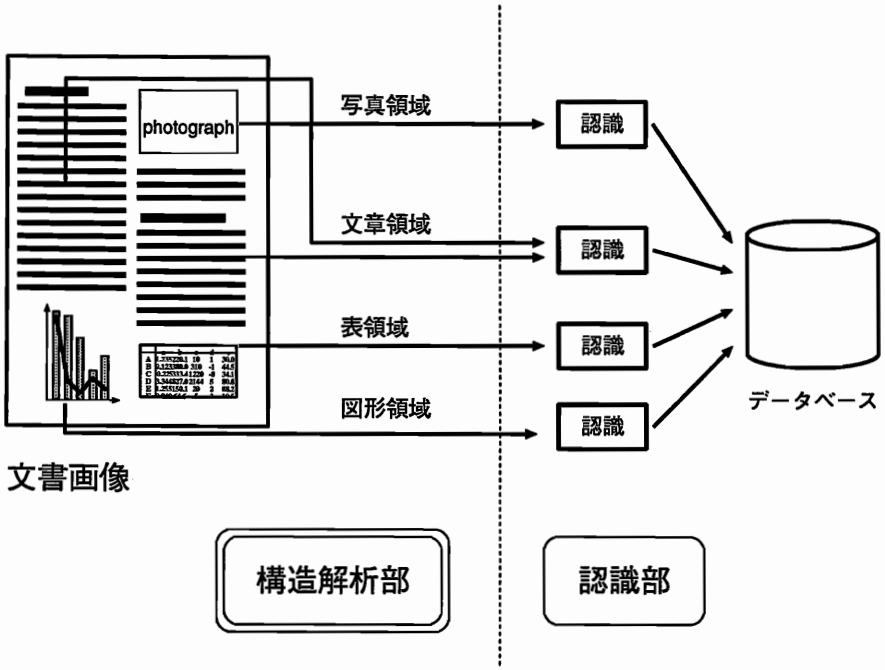
<u>複雑な構造の文書への対応</u>

「複雑な構造」
- 領域の形状が任意
- 異なる領域が接近

本研究の目的

汎用性の高いルールのみを用いて, 様々な構造の文書に対応できる文書認識システムを開発する.

文書認識システムの概要



構造解析部：

文書レイアウトを理解して要素抽出

認識部：

抽出された各要素を認識

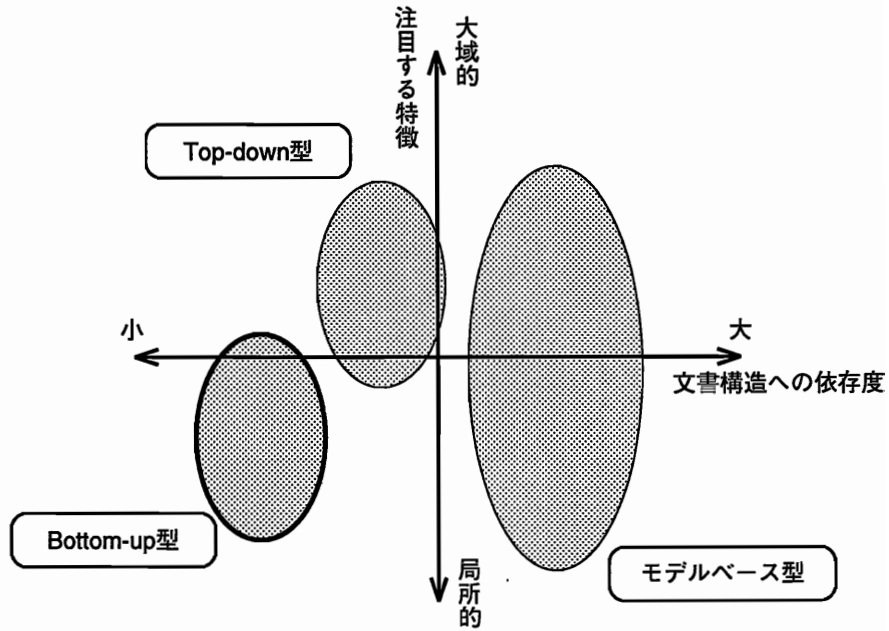
複雑な構造の文書への対応→ **構造解析部の改善**

構造解析で不可欠な処理

... 領域分割

領域分割手法

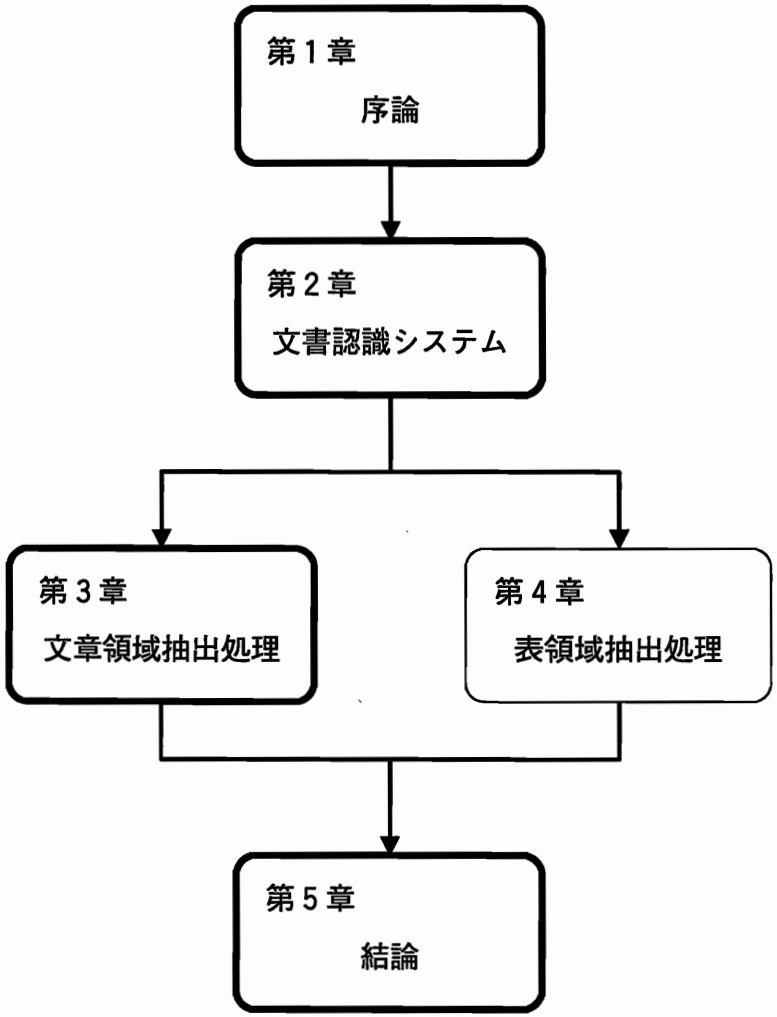
- Top-down 型
- Bottom-up 型
- モデルベース型



本研究の方針:

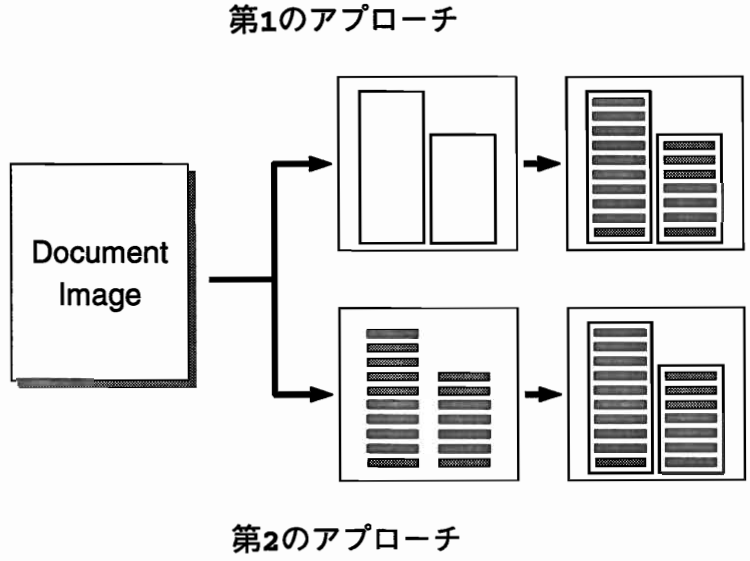
- **Bottom-up 型アプローチ**
- **局所的特徴**

本論文の構成



–第 3 章–　文章領域抽出処理

2 つの Bottom-up 型アプローチ



第 1 のアプローチ... 異なる領域が接近する場合, しきい値の最適化が困難

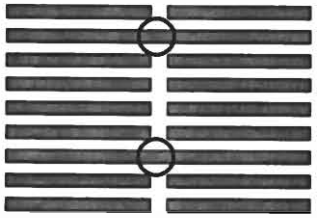
⟶ 第 2 のアプローチを適用

文字行抽出における問題

文書画像から Bottom-up 的に文字行抽出
文字行の長さがわからないため，次の問題が生じる

過接合

行方向に文字行が誤統合

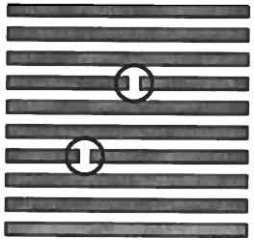


$\Longrightarrow$「過接合切断処理」により対応

近傍にある文字行の端点や空白部分の配置をもとに，文字行内の過接合の位置を推定

分離文字行

行内空白による文字行の分離



$\Longrightarrow$ 文章領域抽出処理の中で対応

提案手法の概要

同一文章領域内には，形状や相互位置関係の類似する文字行が配置されている

この経験則に基づき，文字行を結合

↓

文章領域に相当するグループを生成

結合ルール:

同一文章領域内の文字行

- 傾きが一致
- 高さが一致
- 長さが一致
- 行間距離が一致

中間領域: 文章領域生成途上における文字行の集合
基礎領域: 中間領域の初期値

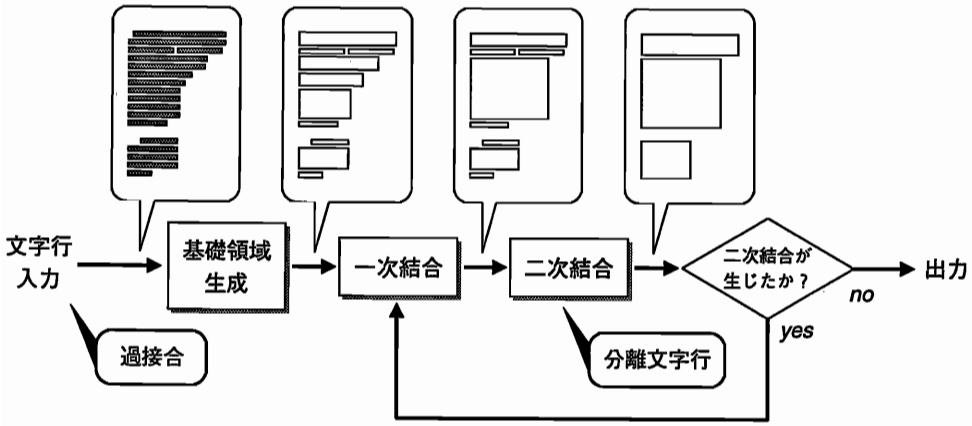
中間領域の結合

一次結合:

結合ルールによる結合

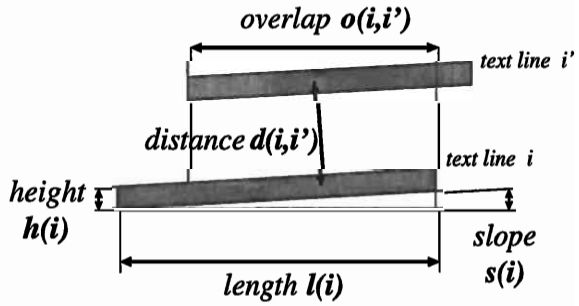
二次結合:

同一文章領域にありながら一次結合では結合されなかったと思われる中間領域を結合



文字行抽出 ··· 「区分直線連結法」(後藤ら,1995)

基礎領域の生成



文字行 i, i' について,

$s(i)$	i の傾き
$l(i)$	i の長さ
$h(i)$	i の高さ
$o(i,i')$	i と i' の重なり

形状類似に関する条件:

$$\begin{aligned} |s(i)-s(i')| &\leq T_1 \quad \text{(傾き一致)} \\ Ratio(l(i),l(i')) &\leq T_2 \quad \text{(長さ一致)} \\ Ratio(h(i),h(i')) &\leq T_3 \quad \text{(高さ一致)} \end{aligned}$$

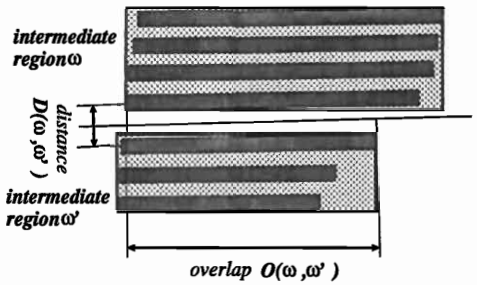
$$\left(\text{但し } Ratio(x,y) \equiv \frac{\max(x,y)}{\min(x,y)} \geq 1 \quad (x,y \geq 0)\right)$$

相互位置関係に関する条件:

$$\frac{o(i,i')}{\min(l(i),l(i'))} \geq T_4$$

各条件をみたし，文字行間距離が最近接関係にあるもの同士で**基礎領域**を生成

中間領域の一次結合



中間領域ω, ω' について,

$S(\omega)$	ωに属する文字行の傾き平均値	
$L(\omega)$	〃	長さ平均値
$H(\omega)$	〃	高さ平均値
$G(\omega)$	〃	文字行間距離平均値
$O(\omega,\omega')$	ωとω'の重なり	
$D(\omega,\omega')$	ωとω'の距離	

形状類似に関する条件:

$$
\begin{aligned}
|S(\omega)-S(\omega')| &\leq T_5 && \text{(傾き一致)}\\
Ratio(L(\omega),L(\omega')) &\leq T_6 && \text{(長さ一致)}\\
Ratio(H(\omega),H(\omega')) &\leq T_7 && \text{(高さ一致)}\\
Ratio(G(\omega),G(\omega')) &\leq T_8 && \text{(行間距離一致)}
\end{aligned}
$$

相互位置関係に関する条件:

$$\frac{O(\omega,\omega')}{\min(L(\omega),L(\omega'))} \geq T_9$$

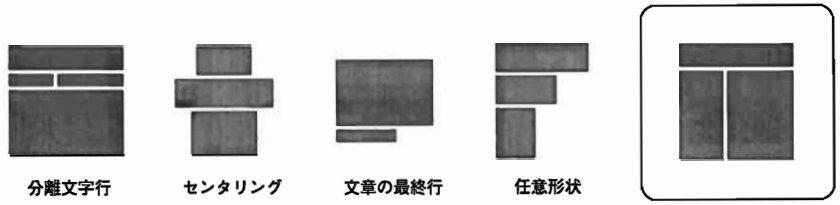
$$\max(Ratio(G(\omega),D(\omega,\omega')),\ Ratio(G(\omega'),D(\omega,\omega'))) \leq T_{10} \quad \text{(行間距離一致)}$$

中間領域の二次結合

- 分離文字行
- センタリング
- 文章の最終行
- 任意形状の文章領域

などに対応

一次結合終了時



二次結合条件:

- 一次結合条件のうち,「長さの条件」「重なりの条件」を緩和

$$Ratio(L(\omega),L(\omega')) \leq T_6' \qquad (T_6' \geq T_6)$$

$$\frac{O(\omega,\omega')}{\min(L(\omega),L(\omega'))} \geq T_9' \qquad (T_9' \leq T_9)$$

- 新たなルールの導入 → 不正な結合を防止
  周辺の中間領域も考慮

評価実験

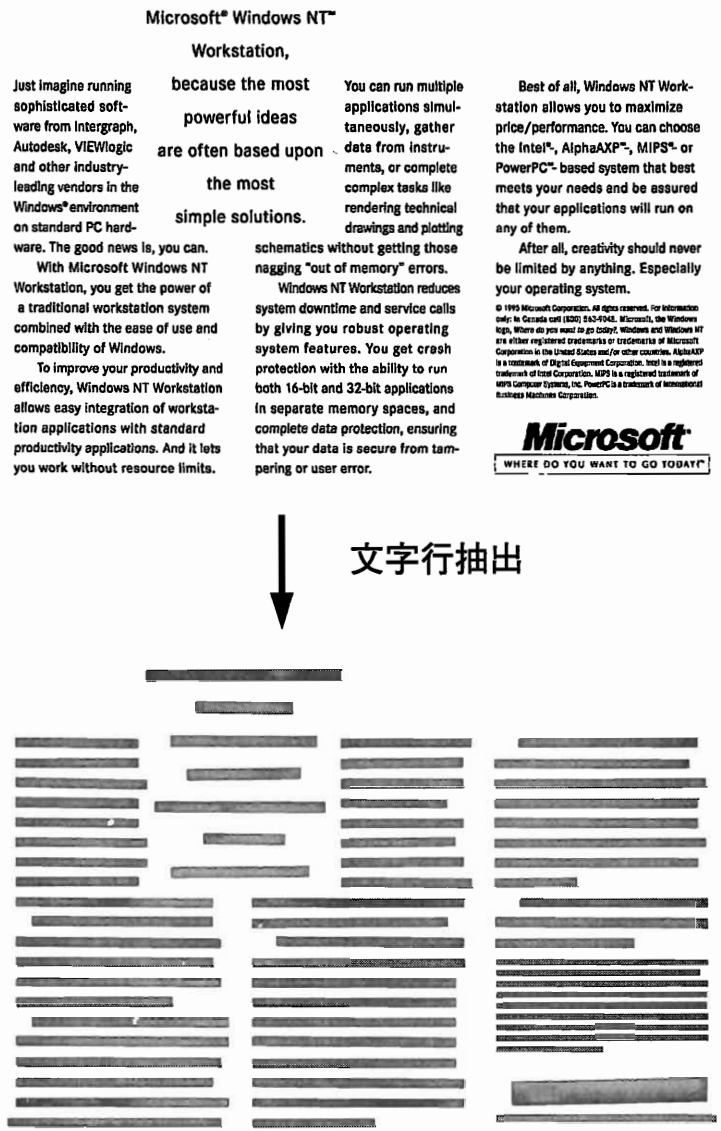
提案手法の有効性を確認するため，文章領域抽出処理の実験を行った

内容	サイズ	使用言語	枚数
雑誌い	B5	日本語	10
雑誌ろ	B5	日本語	10
商品カタログ	A4	日本語	10
学会予稿集	B5	日本語	10
学会論文誌	A4	英語	10
雑誌 A	A4	英語	10
雑誌 B	A4	英語	10
雑誌 C	A4	英語	10
雑誌カ	A4	韓国語	10

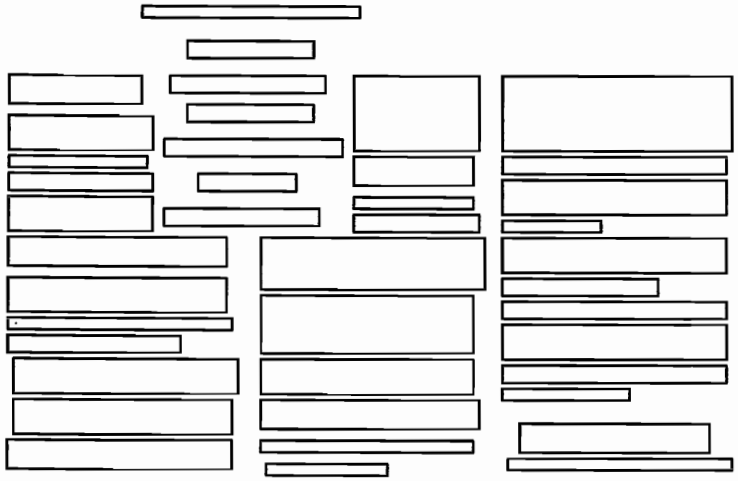
各条件式の定数は次のとおり

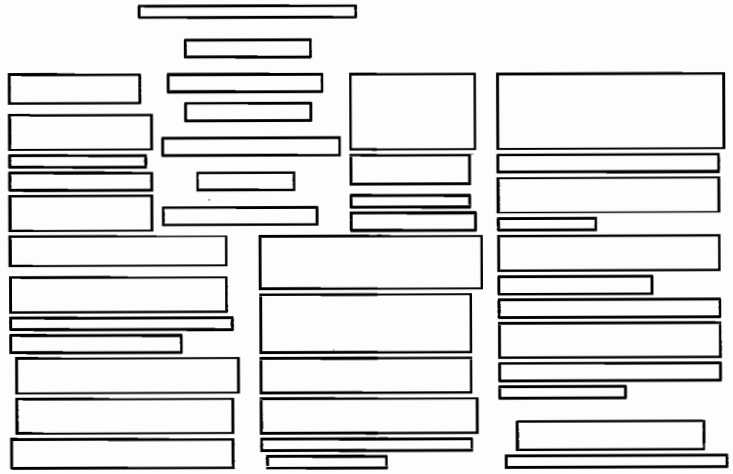
$T_1 = T_5 = 1.5°,$
$T_2 = T_3 = T_6 = T_7 = T_8 = T_{10} = 1.3,$
$T_4 = T_9 = 0.8,$
$T_6' = 5 \times T_6,$
$T_9' = 0.3 \times T_9$

適用例

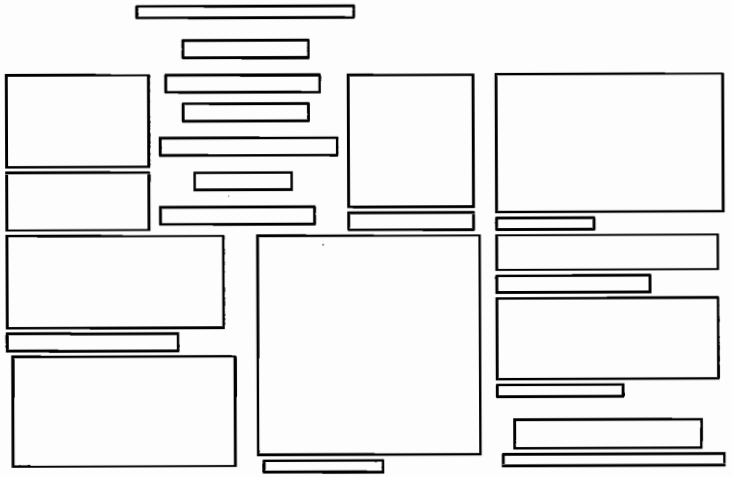


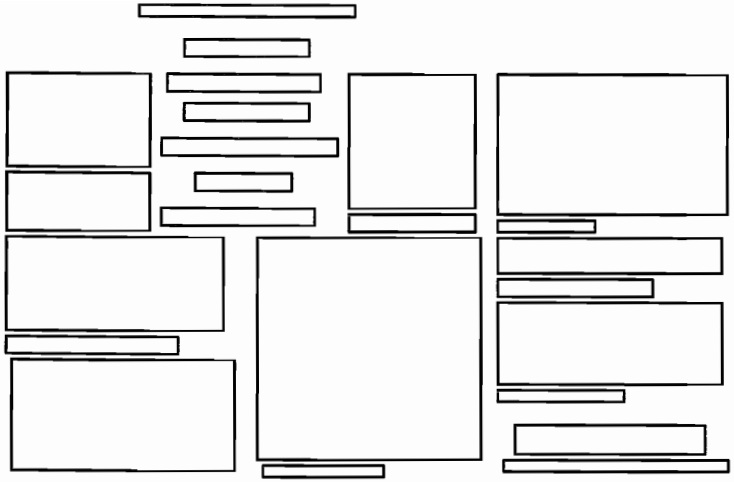
基礎領域生成



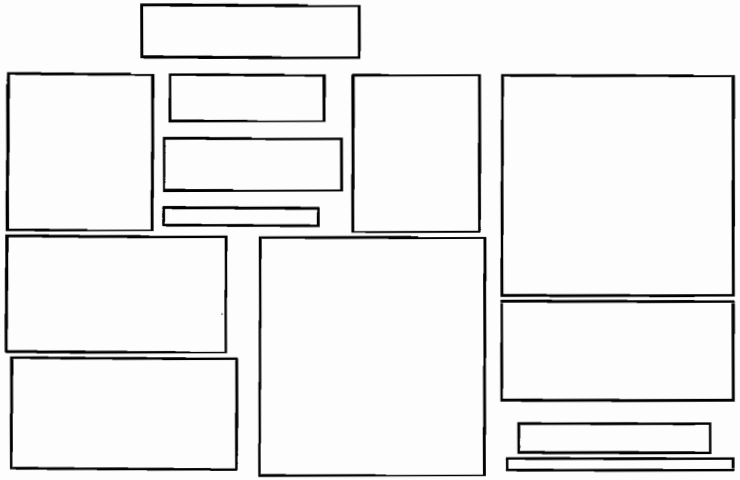


一次結合





二次結合



最終結果

Microsoft® Windows NT™

Workstation,

because the most
powerful ideas
are often based upon
the most
simple solutions.

Just imagine running sophisticated software from Intergraph, Autodesk, VIEWlogic and other industry-leading vendors in the Windows® environment on standard PC hardware. The good news is, you can.

With Microsoft Windows NT Workstation, you get the power of a traditional workstation system combined with the ease of use and compatibility of Windows.

To improve your productivity and efficiency, Windows NT Workstation allows easy integration of workstation applications with standard productivity applications. And it lets you work without resource limits. You can run multiple applications simultaneously, gather data from instruments, or complete complex tasks like rendering technical drawings and plotting schematics without getting those nagging "out of memory" errors.

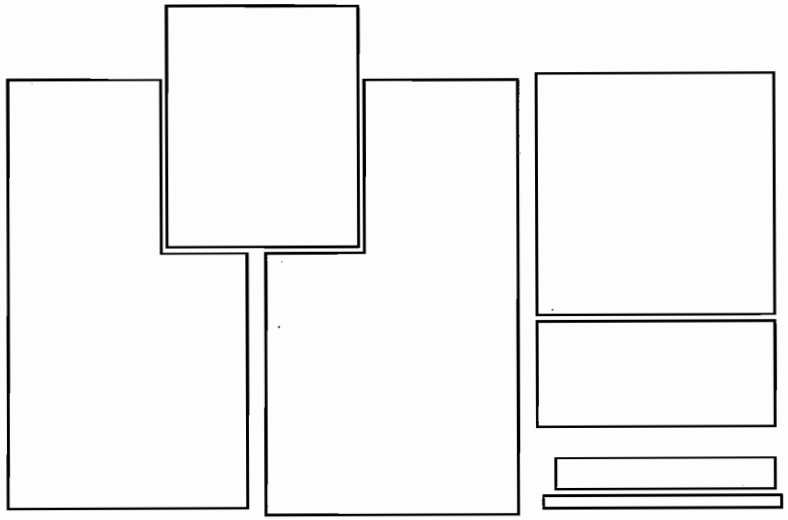
Windows NT Workstation reduces system downtime and service calls by giving you robust operating system features. You get crash protection with the ability to run both 16-bit and 32-bit applications in separate memory spaces, and complete data protection, ensuring that your data is secure from tampering or user error.

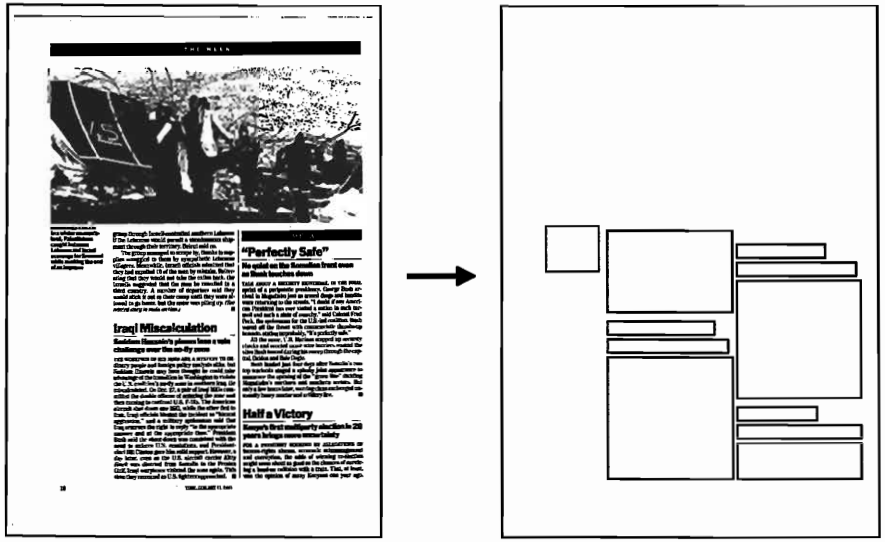
Best of all, Windows NT Workstation allows you to maximize price/performance. You can choose the Intel®-, AlphaAXP™-, MIPS®- or PowerPC™- based system that best meets your needs and be assured that your applications will run on any of them.

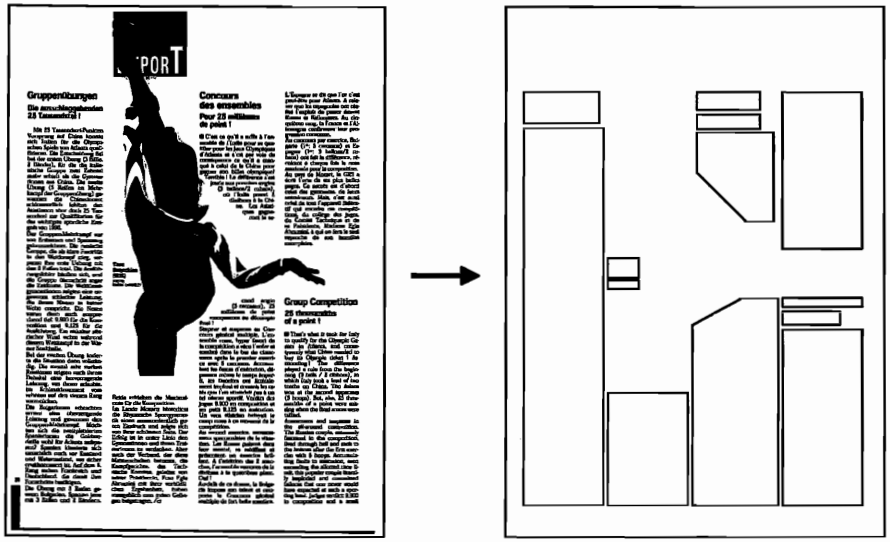
After all, creativity should never be limited by anything. Especially your operating system.

Microsoft

WHERE DO YOU WANT TO GO TODAY?™







結果

全文章領域数	476 個
抽出に成功	460 個

約 97 % の文章領域抽出に成功

→ 提案手法の有効性を確認

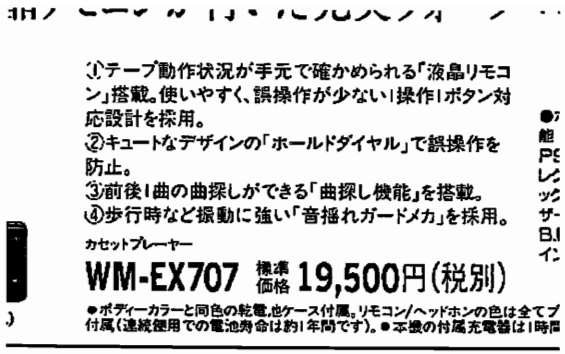
処理時間 ··· 平均 8.33 秒 (Sparc Station 10, 50MHz)

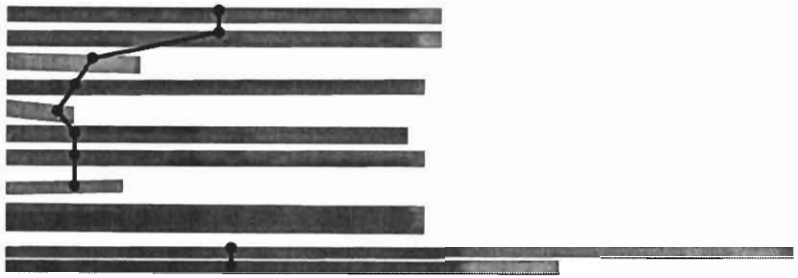
第 3 章のまとめ

- 文字行の形状や相互位置関係に注目した文章領域抽出処理を提案
- 評価実験により有効性を確認

提案手法の特徴

- 複雑な構造の文書に対応
- 文字行の過接合や分離文字行に対応
- 上下対称のルール





–第 4 章–　表領域抽出処理

背景

近年の表認識処理 ･･･ 罫線の省略に対応

↓

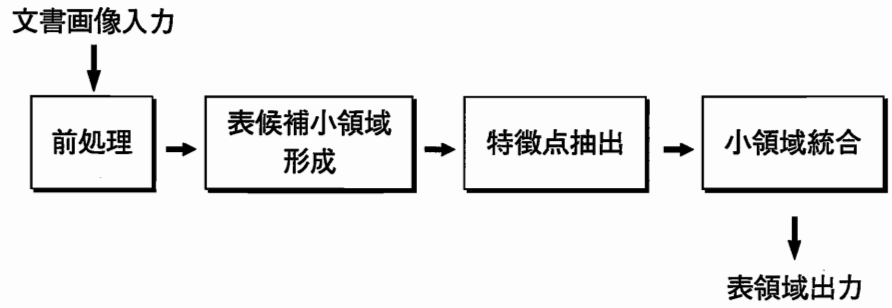
罫線の存在を前提としない表領域抽出処理が必要

提案手法の概要

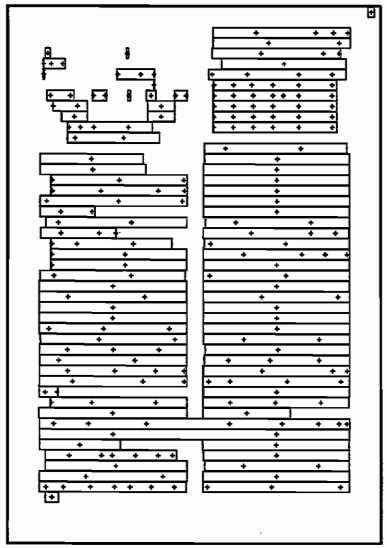
表領域についての知識

- 比較的短い文字行
- 表を構成する項目の重心が規則的に配置

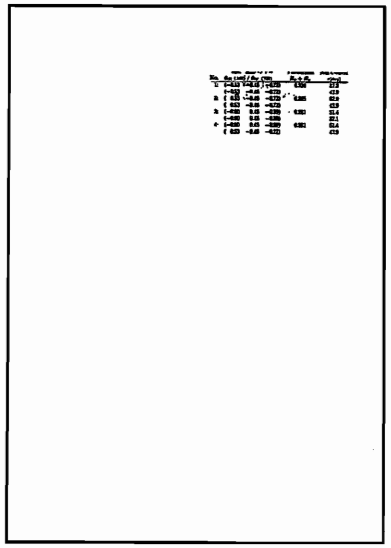
処理のながれ



(a) 入力文書画像



(b) 表候補小領域と特徴点



(c) 表領域抽出結果

–第 5 章– 結論

まとめ

- 複雑な構造をもつ文書に対応する文章領域抽出法を提案した (第 3 章)
- 罫線や項目の省略に対応する表領域抽出法を提案した (第 4 章)

今後の課題

- 縦書横書混在文書への対応