修士学位論文

高度文字画像抽出アルゴリズムに関する研究

東北大学工学研究科 電気・通信工学専攻

平山 理継

目 次

図 目 次

表目次

第1章

序論

1.1 本研究の背景

近年における高度情報化社会の発展は、パーソナルコンピュータ・ワードプロセッサ・コピー機・ファクシミリ等の紙を消費するOA機器の普及を促進してきた。ワードプロセッサ・パーソナルコンピュータはもとより、コピー機やファクシミリ等も家庭を対象に販売されるようになり、広く一般家庭にも見られるようになりつつある。

これらの機器のなかでも特に、ワードプロセッサ・デスクトップパブリッシングなどの文書作成のための計算機支援環境の発展は著しい。

文字は人間社会において情報の交換や記録の媒体として重要な役割を果たしてきた。グーテンベルグの活版印刷術の発明以後、人間社会における文字の使用は飛躍的に増大し、近年の計算機時代に至っての文字情報の氾濫とも言える時代を迎えるまでになっている。人間の書いた文字または印刷された文字を機械によって識別させ、コード情報に変換させることによって、計算機などの入力情報を高速に作りだそうという着想は、1930年代の特許申請に始まり、多くの人々の興味と関心の的になった。特に、計算機の出現した1950年代には、実用的な文字認識装置の開発が試みられた。また文字認識の問題は、さらに広いパターン認識の研究の一環として、また実用的にも最も効果の大きい研究課題として1960年代から1970年代を通じて精力的に研究開発が行われてきた。今後も文字を中心とした情報のデータベース化のニーズは益々高まっていくことが想像できる。

文書というものは、原稿用紙のように文字のみのものも存在するが、論文・報告書や

雑誌等のように通常は文字と図・表・写真の領域が混在している場合が多い (図 1.1)。さらにはポスターのように、写真の上に文字が重畳しているものも存在する。他の画像と比べた文書画像の大きな特徴は、信号的にも意味的にも異なる領域が混在、または重畳しているという点である。

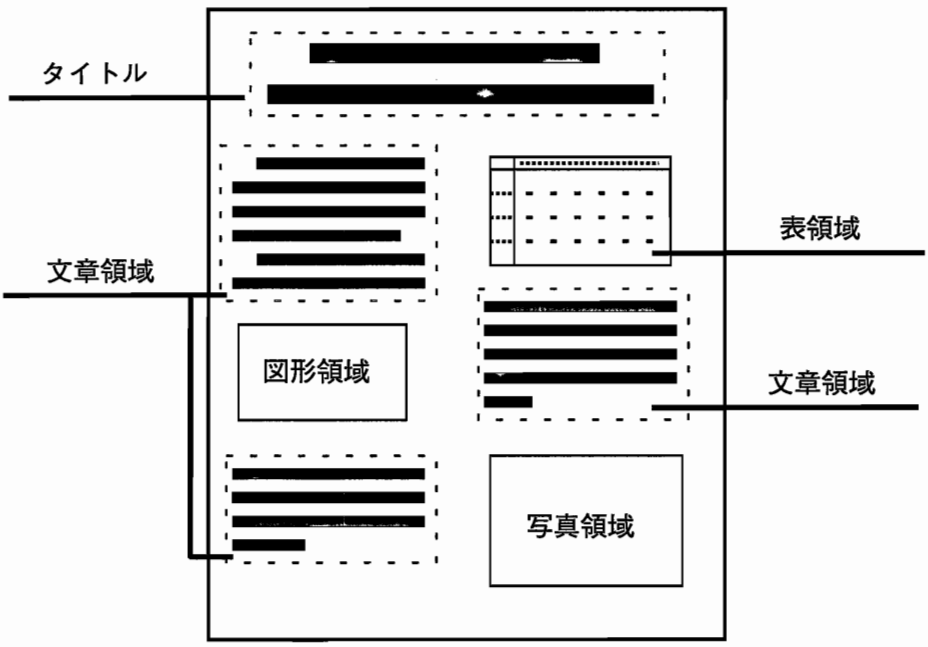


図 1.1: 文書画像の構成の例

文書画像処理は、文書処理・画像理解・文字認識・言語理解だけでなく、マルチメディア処理・人工知能・データベース・ヒューマンインタフェース等の分野とも関係が深い総合的な分野である (図 1.2)。従って文書画像を解析・理解し電子化してデータとして蓄えることは有益かつ要求が高い。実際に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ういった種類の文書構成要素が配置されているか、さらにそれらの構成要素間の関係はどういうものなのか、といった情報を取得する必要がある。文章・写真・図・表領域の混在するような文書を解析し各構成要素の特定を行うことをレイアウト解析と呼ぶ。さらに、解析した各構成要素を認識系にかけることで文書のデータベースを構築できる。

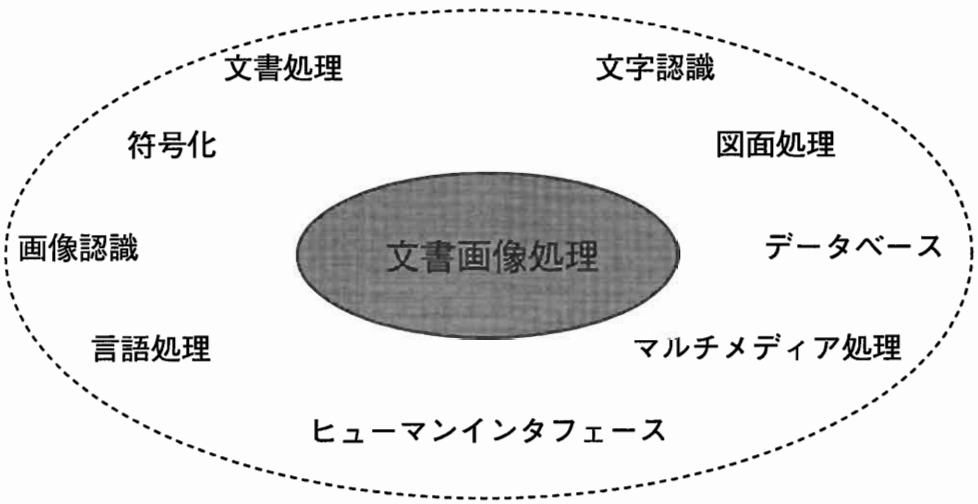


図 1.2: 文書画像処理の位置付け

1.2　本研究の目的

1.1 節で述べたように文書画像処理には多岐に渡る処理群が含まれている。文書画像に限らず画像というものには文字が含まれている場合が多い。この文字情報は、例えば画像データベース、あるいは文書検索システム等を構築する際にキー情報として利用できる。こうした応用のためには画像中より文字領域の抽出を行わねばならない。

抽出された文字画像は OCR(Optical Character Recognition) 装置により文字コードに変換されて記憶装置に蓄えられていく。この際 OCR 装置への入力となる文字画像領域は解像度こそ高いものが必要とされる[1] が、画素当たりの濃淡値は白黒の二段階のパターンで充分である。

濃淡の文書画像からの文字抽出へのアプローチとしては大きく二つに分けることができる。一つは、文字画像のエッジに着目するもので、例としては松尾らの手法 [1] や上羽らの手法 [2][3] が挙げられる。しかしこの手法は比較的大きな文字に対しては効果的であ

[1]漢字を認識させる場合、各文字が最低でも 64×64 dot が必要であり、文字の大きさを 4 mm 角 (約 11 ポイント) とすると 400 dpi の解像度が必要になる。

るが、小さい文字に対してはエッジ部が不安定であるため抽出が困難である。この欠点を指摘し、二つめのアプローチにあたる手法の例としてカラークラスタリングによる手法[4][5]が提案された。この手法により、30×30ドット程度の小さな文字でも、個々の文字パタンを背景から分離できるようになった。これらはカラーの文書画像に対しての文字抽出アプローチであるが、鎌田ら[6]はカラー文書画像の場合、色ノイズが大きい画像が多く、文字抽出の制度が低下する欠点があるので、グレースケールの文書のほうが精度面では良いという指摘をし、現時点ではグレースケール文書画像に対する文字抽出手法を開発し、将来の課題としてこの手法をカラー文書へも拡張することが良いということを発表している。

現実に存在する文書画像においては背景が複雑であったり、シェーディングの影響による濃度むらが生じたり、あるいは文字が文書中の配置によって画像中においてすべてが同一レベルの濃淡値で表現されていない場合がある。このような場合、単一のしきい値処理を用いて二値パターンを得ようとする[7]と文字領域を正確に抽出するのが困難となる。このような文書からの文字の抽出では、単一のしきい値を用いて処理するのではなく、複数のしきい値により文字の画素と背景の画素を分離し、そこから文字領域を抽出してくるといった方法が有効であると考えられている。

本研究においては、**文字領域抽出**に対するアプローチとして、グレースケール文書画像の多値しきい値による手法をとることにした。多値しきい値処理を採用した理由の一つは、カラークラスタリングより単純であるが、処理は似ているので、小さな文字の抽出に有効であると考えられるからである。二つ目の理由は、多値しきい値処理の方がより基本的なので、まず基本から詳しく調べるべきであると考えたからである。本研究の目的は、複雑な背景を持つ、または多種類の濃淡レベルで代表されるような文字から構成されるような文書画像からも文字領域を高精度に抽出できるアルゴリズムを開発することである。

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　本研究の背景及び目的を述べる。

第2章　文書画像処理の概要に関して述べる。

第 3 章　濃淡文書画像に対し、濃度ヒストグラム解析によりその文書画像がいくつの代表濃度で構成されるかを検出し、代表濃度により各画素にラベルを割り当てる手法について述べる。

第 4 章　第 3 章 で述べた手法でラベル付けされた文書画像中から、各ラベルを黒画素とした二値画像を抽出し、連結する黒画素成分に対する外接矩形を求め、その文字列性 (局所的直線性) を検証することで、文字列領域の抽出を行う手法について述べる。

第 5 章　本論文の結論及び今後の課題について述べる。

第 2 章

文書画像処理

2.1 はじめに

文書画像処理の研究は、ファクシミリの符号化及び伝送画像の画質の改善が目的となって盛んになった背景がある。近年においては計算機の処理能力の向上も手伝い、音声処理も含めたマルチメディア統合システムを目指す動きや、文書画像に含まれる文字を認識し、その意味を理解してデータベースを自動的に作成するシステム開発を目指した研究も盛んになっている。

第 1 章 でも述べたように文書画像処理という技術はさまざまな分野と関係がある総合的な分野である。

コピー機やファクシミリ等からも分かるように、文書を画像としてとらえると、紙に書かれたものは内容に関わりなく扱うことができるようになる。文書画像処理とは、このような紙に書かれた文書を画像として取り込み、画像中の情報 –文字など– を計算機処理の対象とする処理技術である。

本章では文書画像処理の研究の概要について述べていく。

2.2 文書画像処理の概要

文書画像の特徴は、第 1 章 でも述べたように何らかの意味で意味の異なる領域が混在、または重畳している点である。文書画像は、図 2.1 のように示すように階層構造を持っており、階層の上部から下に向って構成要素を追及していくような処理をトップダウン的処理と呼ぶ。逆に最小とする構成要素から、ある手がかりを頼りに大局構造を追及する処理をボトムアップ的処理と呼ぶ。

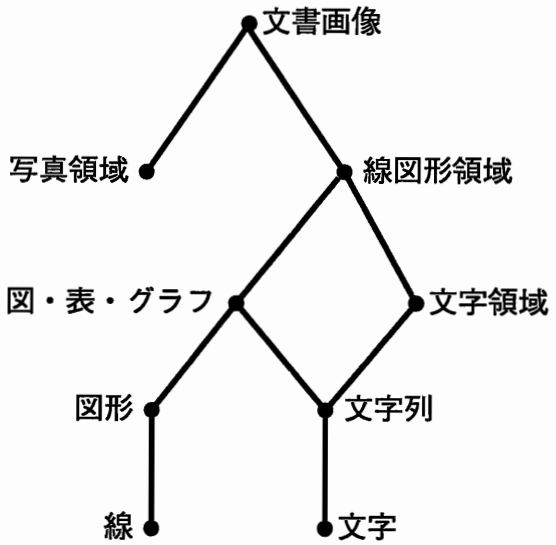


図 2.1: 文書画像の構成要素の関係

文書画像処理は、図 2.2 に示すように、領域の識別処理と結果の統合、システム化に特徴がある。識別された個々の領域を認識する処理は、文字認識・画像認識・図面認識として個別に研究されている。

文書画像を入力するときは、各領域に対して、解像度と量子化の問題を考えなければならない。また、文字・図・表・グラフ等の線図領域は、細かい解像度が必要であるが、画素当たりの濃淡値の白黒の二段階で充分である。文字を、カラー 24 bit の濃淡値で入力したり、写真領域を高解像度 (400 dpi 以上) で入力してもほとんど意味がないが、入力の段階で情報が失われてしまうと後の処理では対処することができない。そのため入力段階では、文書一枚をカラー画像、400 dpi としておき、何らかの処理を施すことによってデータ量を削減していく方法が有効であると考えられる。

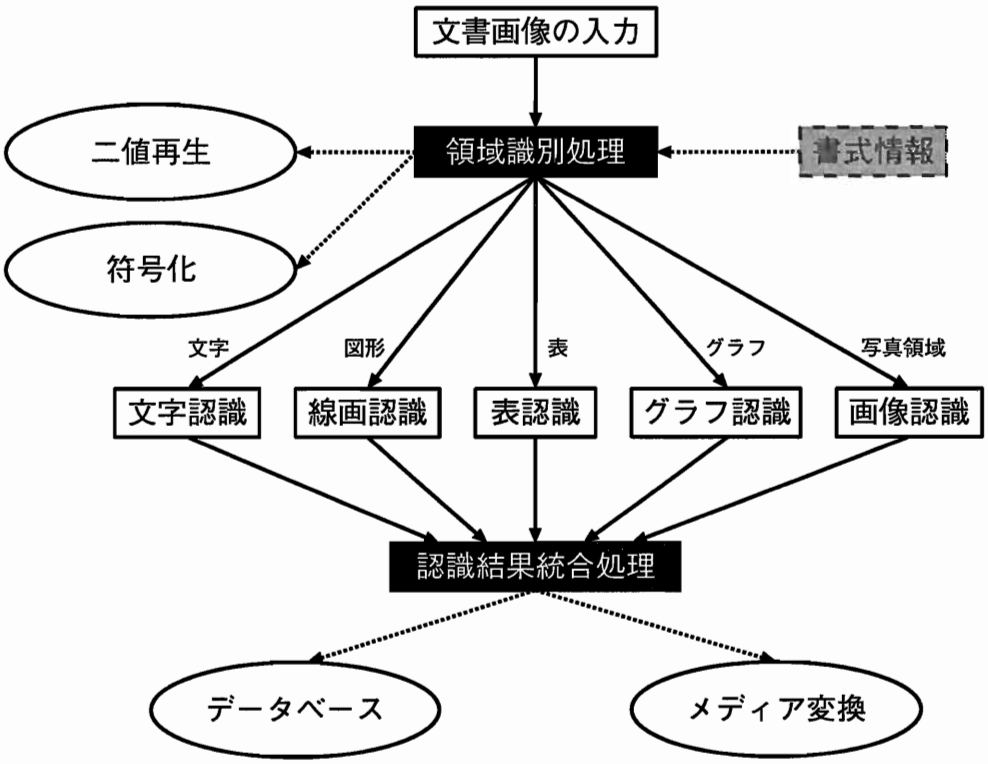


図 2.2: 文書画像処理の概要

文書画像には、性質の異なる領域が混在しているので、まず、領域を識別する処理を行う。各領域の識別のためには、文書画像中に存在する各々の領域がどのようなものであるかを定義しなければならない。

しかし、文字・図・表・グラフ等の領域は元々、“意味的に定義されている” という点に注意しなければならない。すなわち、

- **文字として読めるものが文字**
- **図として意味をもつものが図**

なのである。しかしこのような定義では処理ができないので、信号レベルの術語でこれらの領域を定義する必要がある。これを、**信号レベルのモデル**と呼ぶ。例を挙げると、“エッジが明確に存在する領域が文字領域である” や、“局所的にコントラストが低い領域が写真領域である” 等である。しかし直感的にも分かるように、このような定義では意味的に定義された文字領域は写真領域を完全には識別することはできない。

領域の識別誤りがどのような影響を与えるかという問題は、文書画像処理の目的と深く関わってくる。そこで今、文書画像処理の目的を次のように大まかに 4 つに分けて考える。

1. ファクシミリ等で伝送するときのデータ量の圧縮
2. コピー機やファクシミリで行われる二値再生処理
3. 情報再利用のためのデータベース化
4. マルチメディア情報システムでのメディア変換

目的がデータ圧縮や画像の再生等の場合は、信号的な領域識別ができれば充分であり、領域識別結果が意味的に正しいかどうかは問題ではない。例えば、画像を二値化するときに、局所的な領域を調べて、ある信号的な性質を持っていれば固定しきい値で二値化し、そうでなければディザマトリックス[1]で二値化する。このような処理は、文字領域は固定

[1]二値化手法の一つであるディザ法では、$N \times N$ 画素を 1 つの単位として考え、それに対応する $N \times N$ のしきい値マトリックス (ディザマトリックス) を作り、このディザマトリックス D_N を一種のマスクとして原画像に重ね合せ、各画素の濃度 $f(x,y)$ と対応するしきい値 T_{xy} とを比較して二値化する手法である。

しきい値、写真領域はディザマトリックスというように意味的な領域に正しく対応させる必要はないので、信号レベルのモデルで十分に対処できる。

目的が文書画像の意味理解である場合の領域識別は、意味的に正しい結果が必要である。しかし、意味的に定義されている領域を信号レベルのモデルで記述することはほとんど不可能なので、領域を誤りなく行うことはできない。従って、フィードバック機構がないと、領域識別誤りが後の処理に致命的な問題を引き起こす可能性がある。

文書画像処理の研究において、初期の頃は、信号レベルのモデルのみを用いて文書画像を領域識別し、データ量の圧縮や画像の再生を目的としたものが多かった。これらの研究は、コピー機やファクシミリ等で、ある程度の実用化ができた。

しかし、データの圧縮率をさらに高めるためには、人間が必要とする情報のみを抽出し、それらを伝送することを考えなければならない。計算機の処理能力の向上は、処理のコストと通信のコストの格差を広げていくので、さらなるデータ圧縮処理が必要となる。また、処理目的の 3. や 4. を目指した研究では、意味的な処理が入ってこないと自ずと限界が生じてくる。

近年では、意味処理を積極的に盛り込んで行く方向の研究も進められている。具体的には、文字領域を切り出すだけでなく、文字を認識して、文書画像を理解するシステムの構築を目指す。計算機環境の発達により、文字認識がソフトウェアでも高速にできるようになり、道具として使えるようになってきたことが、この方向の研究に大きく貢献している。文字を認識して、文書を理解しようとすると、言語情報の利用も重要になり、文字の切り出し等の前処理だけでなく、単語情報を利用した後処理も含めて、総合的な文字認識システムをその中に含まざるを得なくなる。

本研究は、2. 及び 3. の目的に位置し、今後の意味処理を含めたシステムを構築する際の、認識処理の前処理に相当する。

2.3　信号レベルのモデルを用いた文書画像処理

信号レベルの処理は、信号レベルのモデルに基づいて行われるので、処理結果の善し悪しは、モデルの善し悪しに起因する。研究の中心は、信号レベルのモデルの提案であるが、汎用的な信号レベルのモデルを定義することは、ほぼ不可能である。

そのため、文書画像処理の研究のなかで、意味的な領域識別が必要な場面で信号レベ

ルのモデルを利用しているものは、人間による介入、すなわち、対話修正を仮定せざるを得ない。この意味では、これらの研究は人間のための道具の作成を目指しているので実用性が高い。

2.3.1　写真領域と線図領域の識別

写真領域と線図領域を識別する信号レベルのモデルとし、次のようなものがある。

(1) 線のモデルを定義し、そのモデルに合わない領域を写真領域とする

(2) 局所領域の濃淡値の最大と最小の差が小さい領域は写真領域である

(3) 写真領域を二値で擬似的に表現した網点領域には、ピーク点が規則正しく現れる

これらのモデルに基づいた研究では、特定の実験データに対して、写真領域が識別できたとする結論が多い。写真領域を識別して、その後に何をするかによって、識別誤りが及ぼす影響が異る。識別された領域を、それぞれ認識したり、それぞれに適応した処理を行う場合は、この段階での誤りが致命的となる。

2.3.2　文字領域と図・表・グラフ領域の識別

線図領域と濃淡領域は、信号的な性質がかなり異なるので、これらの領域を識別する信号モデルは比較的単純に記述できる。しかし、文字領域と図・表・グラフ領域は両方とも線から構成されているので、信号レベルのモデルの定義が困難である。

図・表・グラフ領域と文字領域を識別する信号レベルのモデルとしては、次のようなものが提案されている。

(1) 線の密度が高い領域は文字領域である

(2) 文字領域には文字が規則正しく並んでいる

これらのモデルにより、大雑把な分離はできるが、ここでも結果をどう利用するかが明確でないので、本当の評価はできない。

2.3.3　文字の切り出し

一般に文字領域は、多くの文字から構成されているので、まず、文字領域からの文字列の切り出しを行う。

文字列の切り出し処理の代表的な手法は、黒画素を縦横方向に投影し、射影ヒストグラムの谷を検出するものである。画像が傾いて入力されている場合は、投影する方向や対象とする領域の大きさを工夫しなければならない。

次に、文字列から個々の文字を切り出す処理を行う。文字を信号的に定義しようとすると、以下のようなものが挙げられる。

(1) 日本語の文字は形が正方形である

(2) 文字は直線上に並んでいる

(2) 連続する文字間の空白は、文字列間の空白より狭い

これらの定義に従って、連結する黒画素の外接矩形を正方形になるように統合する手法 [8]、外接矩形の重心をハフ変換[2]し、文字列の方向を求めて文字を切り出す方法 [9]、文字列の垂直方向に投影をとる方法 [10] や文字行や列方向に膨張させる方法 [11] 等が提案されている。

2.3.4　文書画像のファイリングシステム

光ディスクが実用化された頃に、多くのメーカが光ディスクによる文書のファイリングシステム (図 2.3) を発表した。光ディスクファイリングシステムには、多くの文書が電子的に蓄積できるので、事務所のスペースが有効に利用できるというものであった。実際に、多くの文書が電子化できたが、その検索は不便なことが多く、紙のような手軽さで検索することはできていない。

電子的に記録され蓄積された文書画像を修正するために、いろいろな方式が提案されている。トレーシングペーパ等の透明用紙を原画像に重ねてそちらに編集構成記号を記入して編集構成を行うシステム [12] や、人間がやっているように、原画像に赤ペンで訂正を入れて、その指示に従って原稿を構成するシステム [13] 等がある。これらのシステム

[2]ハフ変換は、パラメータで表現できる図形 (例えば直線・円・楕円・放物線) を画像中から検出するための方法である。一般的には直線の検出に対し使われる。

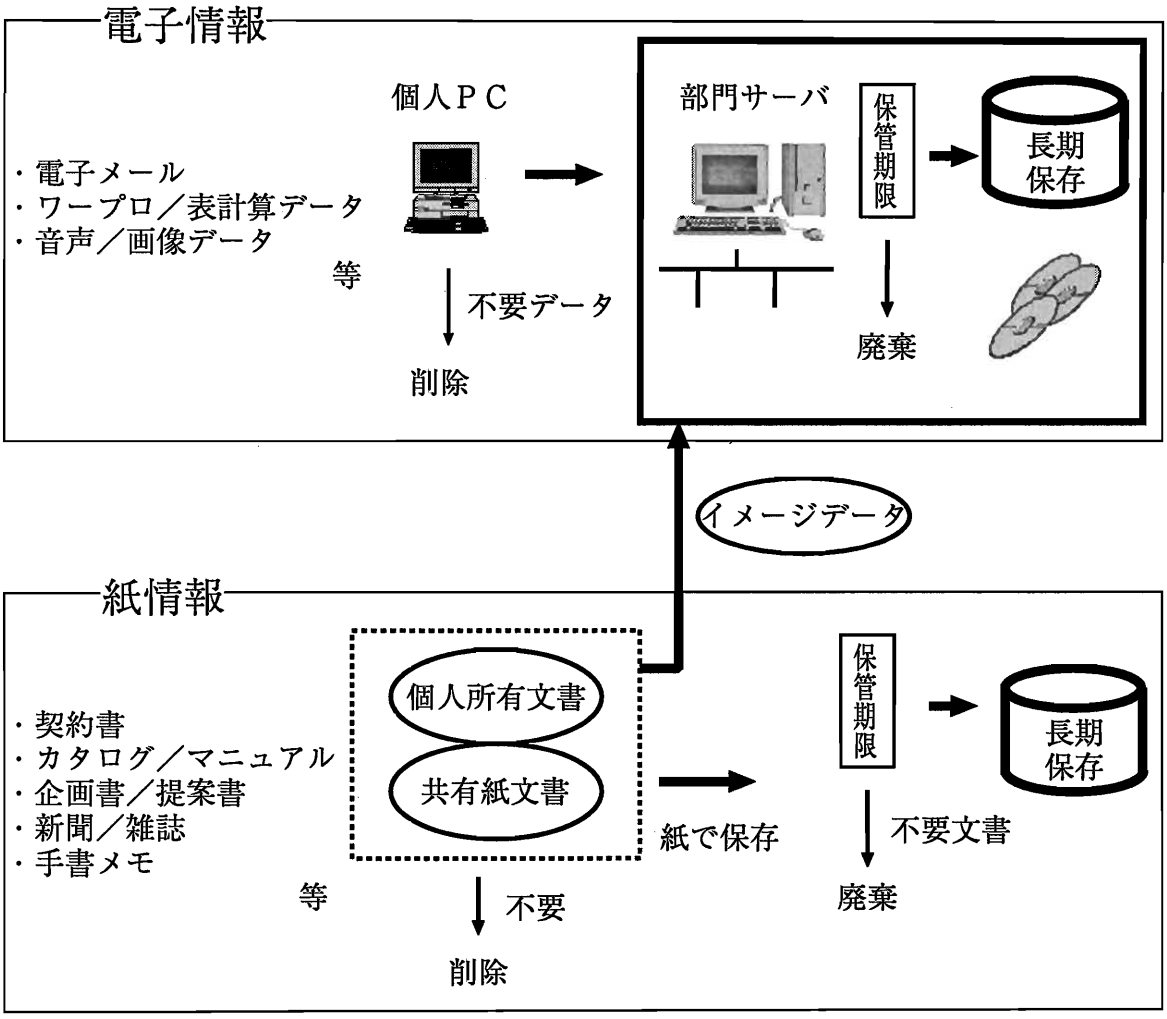


図 2.3: 文書ファイリングシステム

では、書き加えられた指示と原画像にある線を区別する必要から、別用紙や赤色を用いている。

信号的な処理だけで文書画像をファイリングしようとする方向で開発されたシステムは、人間が検索キーを与えなければならないこと、一覧性がないことなど、まだ、人間にとっては本当に使いやすい道具とはなっていない。そのため、意味的な処理を加えて実用化が目指されている。

2.4　まとめ

本章では、文書画像処理の概要を述べ、文書画像処理の研究の代表的なものについても言及した。さらに、文書画像処理の目的の大まかな分類をし、本研究がそれらの中のどの目的に基づいているかを述べた。

文書画像処理技術とは、紙に書かれた文書から電子的なメディアへの変換技術であるとも言える。電子的なネットワーク環境は、急速に全世界的に広まってきているので、紙と電子文書の相互変換が手軽にできるようなシステムへの期待は強い。

第 3 章

文書画像のラベル付け

3.1 はじめに

濃淡の文書画像中における文字・文字列を抽出する問題では代表的なものとしては判別分析に基づく手法 [7] が挙げられる。しかし、このような単一なしきい値で分離する方法では、文字領域部に対応する濃度レベル値が複数存在するような文書にはうまく対処できない。図 3.1 では濃度ヒストグラム中に 3 つのピークが存在し、濃度レベルの低いほうから順に文書画像中ではそれぞれ、黒い文字、灰色の文字、背景・白抜き文字に対応している。この文書画像に対し、濃度ヒストグラムを基に単一のしきい値で二値化した場合、黒・灰色・白の文字すべてを文字領域として背景と分離することができない。これに対し、文書画像をいくつかの部分領域に分割した上で、各小領域毎にしきい値を決定し、各領域間でそれらをなだらかに結ぶ方法 [14][15] が提案された。しかし、各小領域内で文字に対する濃度レベルが一種類であるとは限らないため、このような局所的な二値化処理では、文字と背景に分離できないことがある。さらにこのような単一しきい値処理では、明るい画素を背景部、暗い画素を文字部というように決めることが多いが、実際にはどちらが背景部でどちらが文字部であるかが分からない。必ずしも背景は文字色より明るいとは限らず、例えば “白抜き文字” のようなものも存在するからである。

本章では、上述のような複雑な文書画像からも高精度に文字領域の抽出を行うための多値しきい値処理による濃淡文書画像のラベル付け手法を詳しく説明する。本研究で扱う “複雑な文書画像” とは、画像中の背景や文字が複数の濃度 (色) で描かれているものを

指す。

通常、文書中の見出しなどの文字領域というのは、読み手に “読んでもらいたい” ものであるから、文書作成者は、読み手にとって目につきやすいようにして作成されていることが多いと言える。また、非常にデザイン的に凝ったものを除けば、一つの文字行の中の文字は等しい色で書かれている。さらに、同様の理由から背景とのコントラストも高くされているのが普通である。よって本研究では、文字自体の濃度の変化が非常に大きいものや、グラデーション文字、コントラストの著しく低い文字は対象としない。

レ

3.2　提案手法の概要

3.1 節で述べたような問題に適応させるため、我々の提案する方法においては、文書画像の各小領域における濃度ヒストグラムの解析を頼りに代表色 (濃度) とその数を決定し、それにより小領域内の各画素にラベルを割り当てることで小領域内の**セグメンテーション**を行う。ここで、セグメンテーションとは、画像データを局所的な特徴 (明るさ・色・テクスチャ等) の一様な部分画像に分割する手法のことで、領域分割 (region segmentation) とも呼ばれる。小領域毎にこの処理を行うのは、全体の濃度ヒストグラムを求めた場合、ある文字領域に相当する濃度レベルの画素群が、この文字領域以外の領域で濃度レベルに近い濃度の画素から構成される部分 (背景や絵等) に吸収され、その結果、濃度ヒストグラム中において文字領域に対応するピークを代表色の濃度値として検出できない可能性があるためである。各小領域内のセグメンテーションをした後、全てのラベル間において、同一の文字列及び背景部を形成していると思われるもの同士を結合していく。

最終的に得られた各ラベルにおいて文字・及び文字列を形成しているかの検証を行い、文字列を構成していると判断されたラベルを取り出していく。文字列の抽出については第 4 章 で述べる。

3.3　小領域内ラベル付け

今、濃淡画像を座標 (x, y) における画素を $P(x, y)$ 、その濃度を $f(x, y)$、その濃度ヒストグラムを濃度レベル D の関数 $H(D)$ で定義する。

原画像

濃度ヒストグラム

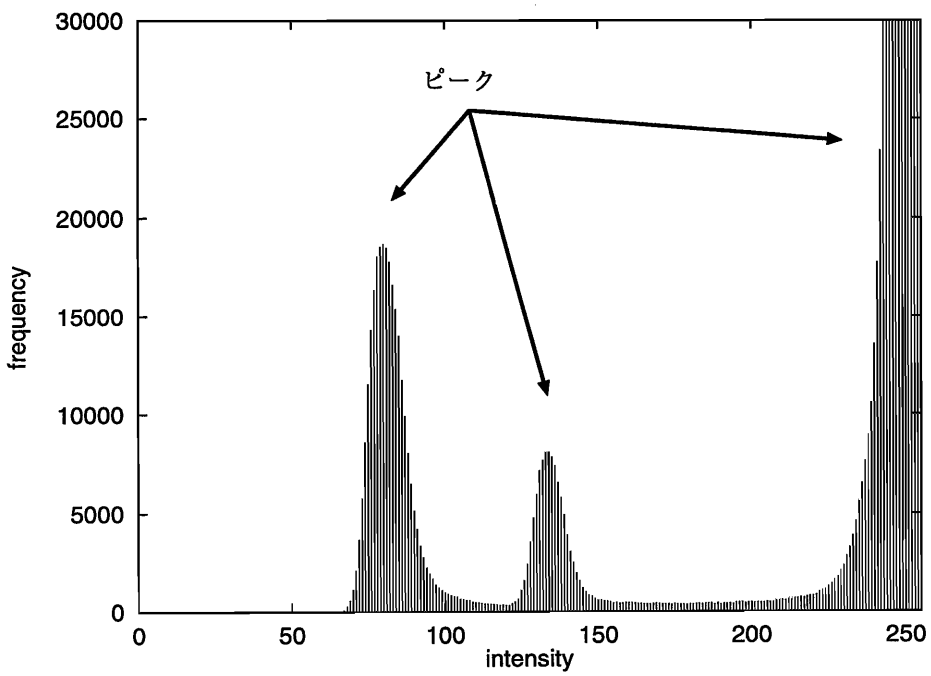


図 3.1: 単一しきい値処理ではうまく文字領域を抽出できない例

$$H(D) = \sum_{P(x,y) \in F(D)} 1 \tag{3.1}$$

ここで $F(a)$ は濃度値が a であるような画素 $P(x,y)$ の集合で、$F(a) = \{P(x,y) | f(x,y) = a\}$ である。

今、この画像 $f(x,y)$ を $N \times N$ ピクセルの正方形領域 R^i に分割し、隣接する小領域同士は $N/10$ ピクセル分の重なりをもたせるように配置する (図 3.2)。

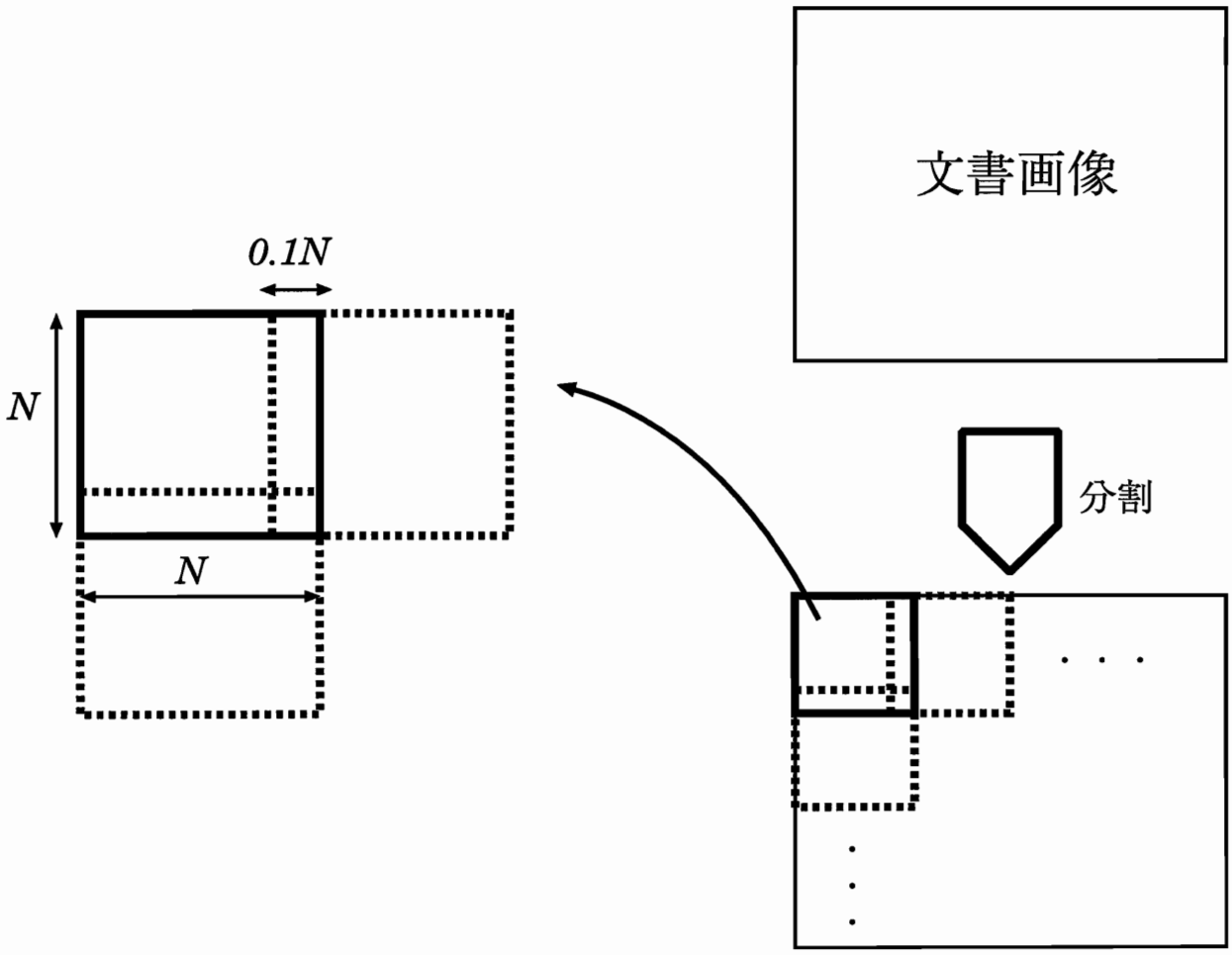


図 3.2: 文書画像の小領域への分割

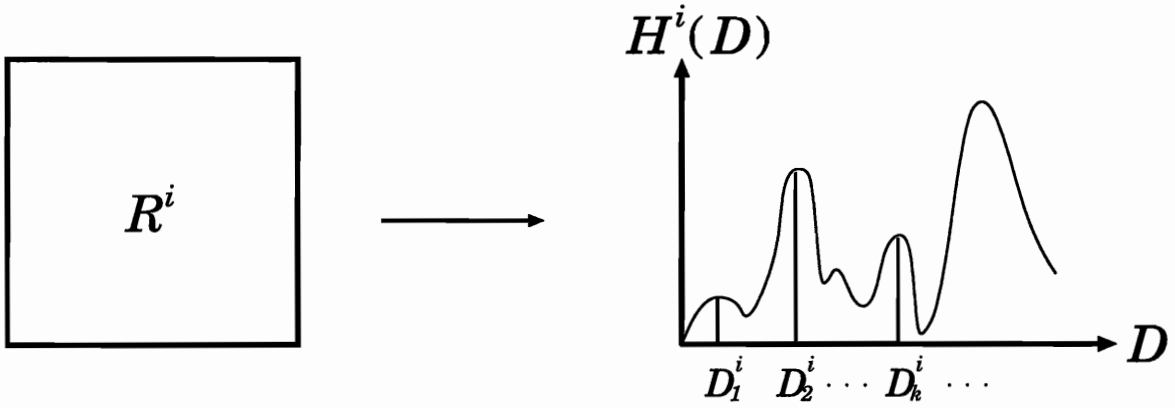


図 3.3: 小領域から濃度ヒストグラム作成

この各小領域 R^i における濃度ヒストグラムを 濃度 D を変数に持つ関数 $H^i(D)$ で表したとき、この関数 $H^i(D)$ から極大値を示す濃度 D_k^i を検出する。ただし、$H^i(D)$ から直接極大値検出を行うとノイズの影響などにより、本来存在しないはずの D_k^i が得られる可能性がある。従って $H^i(D)$ に対してスムージングを施した後のスムージングヒストグラム $Hs^i(D)$ から検出することにする。D_k^i を小領域 R^i における k 番目の代表濃度と呼ぶ。

3.3.1 移動平均オペレータによるスムージング法

ノイズ除去スムージング法の一つとして、移動平均オペレータによるものがある。画像処理の分野でよく利用されるノイズ除去フィルタの最も基本的な技術の一つ [16] である。局所平均フィルタとも呼ばれ、中心画素を含む局所領域の平均濃度を中心画素の濃度とするものである。

原画像を $f(x,y)$ 、処理後の画像を $g(x,y)$ とすれば、このフィルタは

$$g(x,y) = \frac{1}{m \cdot n} \sum_{i=1}^{m} \sum_{j=1}^{n} f\left(x - \left[\frac{m-1}{2}\right] - 1, y - \left[\frac{n-1}{2}\right] - 1\right) \tag{3.2}$$

の式で表わすことができる。ここで、m,n はスムージングするマトリクスサイズを与え、[ ] はガウス記号である。

このフィルタは最も単純なスムージング法であり、濃淡画像を、より滑らかな画像へ変換する効果がある。今、この手法を濃度ヒストグラムのスムージングへ応用することを考える。二次元の画像と違い、濃度ヒストグラムは一次元で表されるため、移動平均オペレータによって処理する場合、その一点に関して前後の点との平均をとることになる。本研究においては、前後 3 点との合計 7 点の平均を処理後の値とすることにした。その様子を信号モデルを用いて図 3.4 に示した。

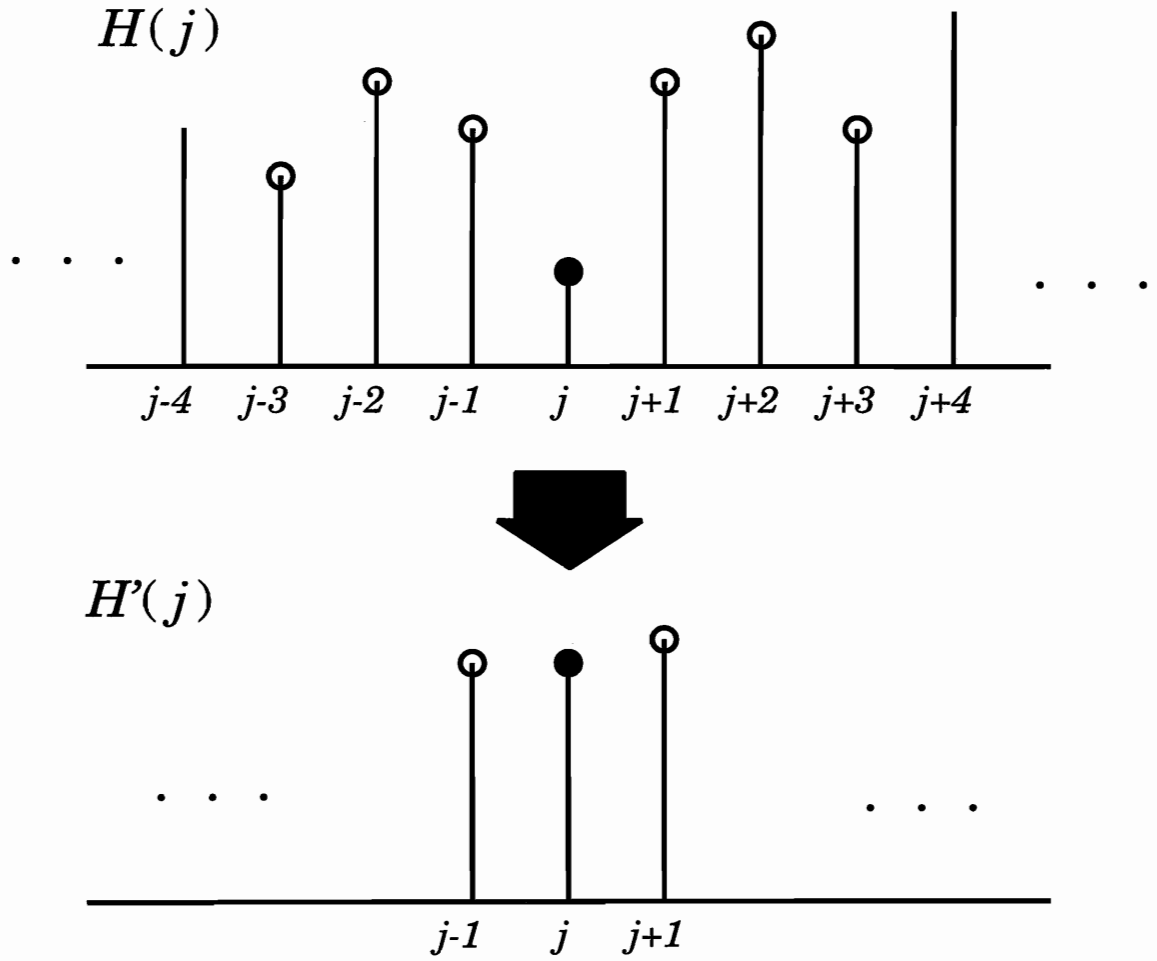


図 3.4: 移動平均オペレータによるスムージング

入力信号を $H(j)$ 、処理後の出力信号を $H'(j)$ としたとき、出力信号 $H'(j)$ は次式

$$H'(j) = \frac{1}{7}\left(\sum_{j-3}^{i=j+3} H(i)\right) \tag{3.3}$$

で定義される。

3.3.2 ヒステリシススムージング法

小領域内で代表色を選出する際、濃度ヒストグラムを解析してピークの検出を行うが、このとき、濃度むら等による小振幅成分をも抽出するという問題がある。濃度ヒストグラムにヒステリシススムージングを施すことにより、不必要な代表色選出を抑えることができる。

画像中の雑音は、エッジ等による大きな濃淡変化と比べ、一般に小さな濃淡レベルのゆらぎとして現われる。これを除去するための処理として、比較的効果が明確でよく使われるものにヒステリシススムージング [16][17] がある。図 3.5 に小振幅成分除去の様子を示す。

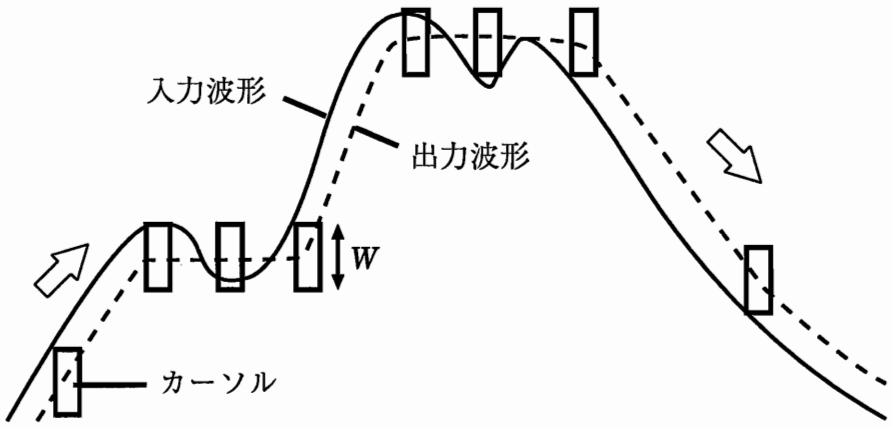


図 3.5: ヒステリシススムージングによる小振幅成分の除去

この処理を次式 (3.4) により離散のモデルに対応させる。

$$H_o(D) = \begin{cases} H_o(D-1) + \frac{W}{2} & if \quad \left(H_i(D) \leq H_o(D-1) - \frac{W}{2}\right) \\ H_o(D-1) - \frac{W}{2} & if \quad \left(H_i(D) \geq H_o(D-1) + \frac{W}{2}\right) \\ H_o(D-1) & if \quad \left(H_o(D-1) - \frac{W}{2} < H_i(D) < H_o(D-1) + \frac{W}{2}\right) \end{cases} \tag{3.4}$$

但し、$H_i(D), H_o(D)$ は入・出力波形で W はカーソルの全幅である。

濃度ヒストグラムに対し、移動平均オペレータによるスムージングとヒステリシススムージング法によるスムージングを施した例を図 3.6, 3.7 に示す。この図より、元の濃度ヒストグラムがスムージング処理により、滑らかになり、小振幅成分が除去されていることが分かる。

上述 2 種類のスムージング処理を施した小領域内濃度ヒストグラムより、ピーク検出を行って、代表濃度 D_k^i を決定する。続いて、各画素 $P(x,y) \in R^i$ に関してその濃度値が $f(x,y)$ であるときに $|f(x,y) - D_k^i|$ が最小となる D_k^i を見つけ、$P(x,y)$ を集合 L_k^i に属させる。但し、L_k^i は小領域 R^i 内においてラベル k が割り当てられた画素の集合である。

3.3.3 代表色の統合処理

濃度ヒストグラムの解析により得られた代表濃度 D_k^i により、小領域内で各画素にラベル付けを行った。

一つの文字に対しては一種類のラベルが付与されることが望ましい。しかし、この時点では、一つの文字に対して複数のラベルが割り当てられる可能性がある。そこで、小領域内のコントラストを考慮しながら、ラベル同士の併合を行う (図 3.8)。

小領域 R^i 内での任意の 2 つの代表色 D_s^i, D_t^i において

$$\left|D_s^i - D_t^i\right| \leq \max(\alpha, 0.2d) \tag{3.5}$$

を満たすならば

$$L_s^i \cup L_t^i \tag{3.6}$$

によりラベルの併合を行う。但し、α は併合条件に関する定数、d は図 3.8 に示したように、最低の濃度と最高の濃度の差である。

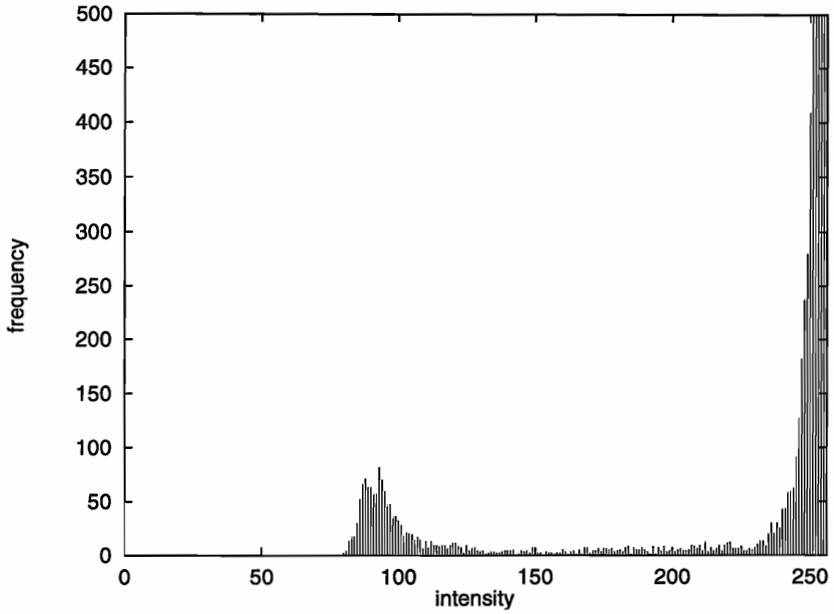


図 3.6: 未処理の濃度ヒストグラム

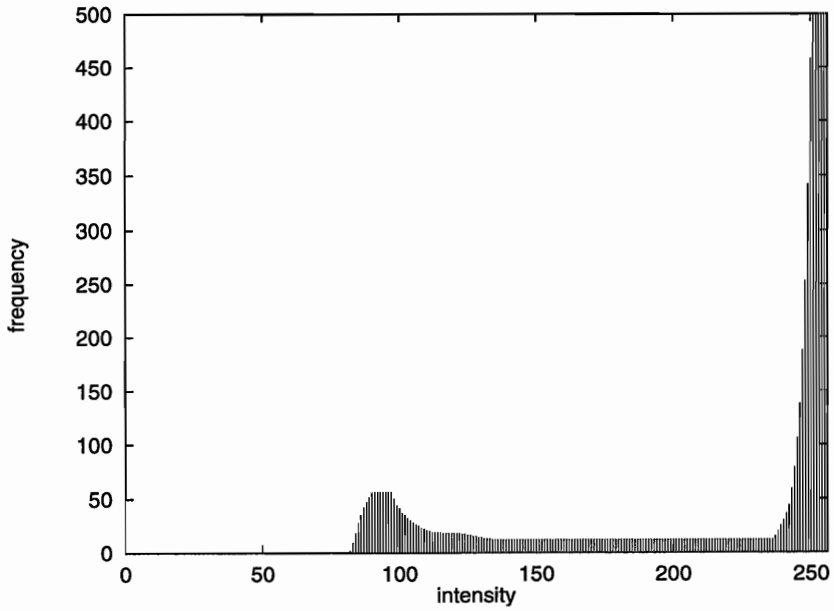


図 3.7: スムージング後の濃度ヒストグラム

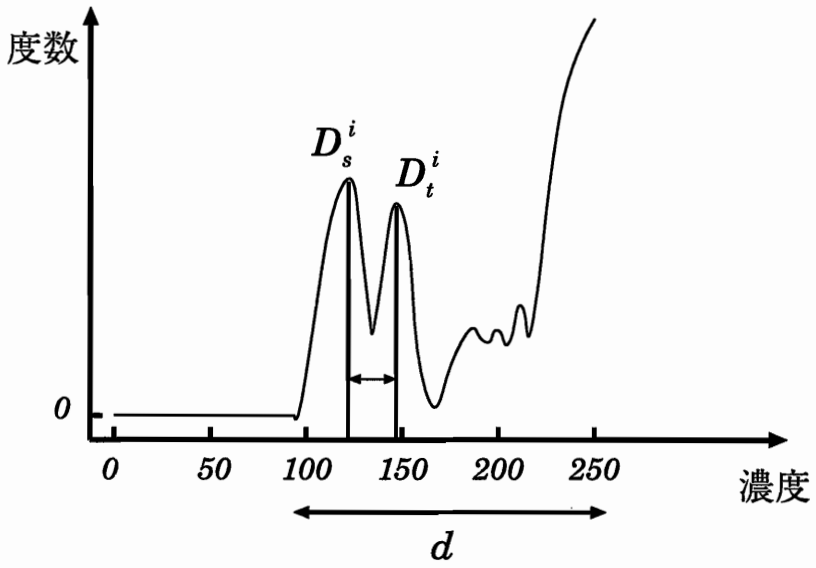


図 3.8: 小領域 R^i 内での代表色の併合

3.4　小領域間でのラベルの統合

3.4.1　オーバーラップ部のラベル選択

まず第 1 段階として隣接する小領域間で重なりを持った部分に関しての処理を行う。

この部分をあらためて R_g としこの領域を部分領域と呼ぶことにする。この部分領域 R_g 内に存在する画素 $P(x,y)$ は 2 つあるいは 4 つの小領域に同時に属している。すなわちこの画素には 2 種類ないし 4 種類のラベルが割り当てられていることになる。

このような重なりを持つ部分領域 R_g 内の各画素 $P(x,y)$ に対し、採用するべきラベルを決めるため、以下の式 (3.7) を判断基準とする。

$$\min_{R^i \supset R_g} \left(\left| f(x,y) - D^i(x,y) \right| \right) \tag{3.7}$$

但し、$D^i(x,y)$ は領域 R^i において画素 $P(x,y)$ に割り当てられた代表濃度を表す。

(3.7) 式を満足する小領域 R^i で画素 $P(x,y)$ に割り当てられたラベルのみを採用し、これ以外の小領域にて割り当てられたラベルは、この画素に関しては破棄する。

3.4.2 全小領域間のラベル統合

全ての小領域内に存在する画素に対するラベル付けが終わってから、隣接する小領域間でのラベルの統合を行う。

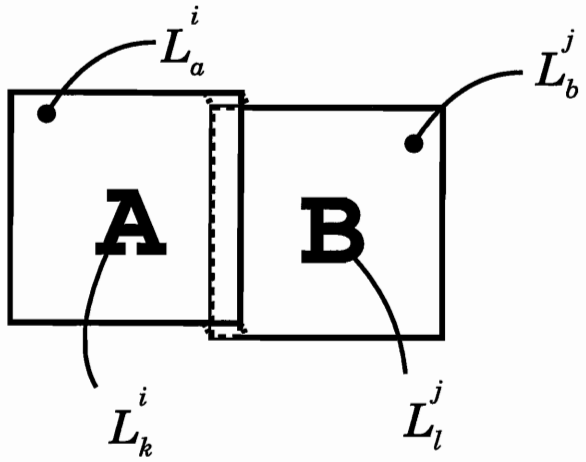


図 3.9: すべての小領域 R^i 間での代表色の併合

各小領域においてラベル k が割り当てられている代表濃度を D_k^i とするとき、

$$\left|D_k^i - D_l^j\right| \leq \theta \tag{3.8}$$

を満たすならば $L_k^i \cup L_l^j$ によりラベルをマージしていく。ここで θ はラベルの併合条件を与えるしきい値である。

このラベル併合処理により、対象となる文書画像のラベル総数 M が決まる。

3.5 評価実験

本章で提案した文書画像のラベル付け処理の有効性を確認するため、各種濃淡文書画像に対して評価実験を行った。実験に用いたサンプル及び実験条件を表 3.1 に示す。実験に用いたサンプル文書画像のうちの代表的な例を図 3.18 ～ 図 3.21 に示す。

使用サンプル	400 dpi 濃淡画像 (8bit/pixel)
小領域サイズ (N)	50, 70, 100, 150, 200 (dot)
カーソル幅 (W)	20
マージ定数 (α)	10
ラベル統合しきい値 (θ)	10

表 3.1: 実験条件・定数

表 3.1 定数 W の値は、濃度ヒストグラムにおいて画素の頻度が 20 程度におさまる小振幅成分がある代表色を表すものではなく濃度むらなどによる揺らぎによるものである事がほとんどであった事実に基く。また、濃度ヒストグラム中で、一つの文字に対する画素は、濃度値で 20 から 30 程度の広がりを示すことがあるため、小領域内での代表色同士の統合の際の最小しきい値を与える定数 α を 10 と定めた。同様の理由から、小領域間でラベルを統合する際の、代表濃度差に関するしきい値 θ を 10 とした。

各小領域サイズ毎のラベル付けの結果を表 3.2 にまとめた。ここで、○及び×の評価の基準は、文書画像中に存在する、ある意味をもった一連の文字領域に対して同じラベルが割り当てられているか否かを、目視による判断である。小領域サイズ N を、小さくしていくと濃度ヒストグラムからの代表濃度選定の際、濃度ヒストグラムを構成するための画素数が少なすぎるために、本来出るべきピークが濃度ヒストグラム中に現れず、実際には 2 色代表色が選出されるべきであるはずなのに 1 色しか抽出できないことがあった。そのため文字がつぶれてしまうような結果になる場合がある (図 3.10)。逆に N を大きくしていくと、文字が大きく、線幅が太いようなものにはほぼ問題なくラベル付けを行えるが、小さく線幅の狭いような文字の存在する領域だと、文字そのものの内部での濃度のばらつきの影響が大きいため、背景を構成する画素の数が増えるに従い、やはり濃度ヒストグラム中で背景部の画素が成す部分に埋もれるケースがある。それにより、文字に対して背景と同じラベルが割り当てられることになる。図 3.11 がこの場合の例であるが、下線部の文字に対して正しくラベルが割り当てられていない。従って、小領域サイズ N には、その画像の解像度によって最適なサイズ (の範囲) が存在する。実験から $N = 100$ dot から 150 dot 程度に設定すれば良好にラベル付けが成されることが分かった。

ラベル付け実験の結果例を図 3.12 ～ 図 3.16 に示す。図 3.12 は本章のはじめに述べ

サンプル No.	N:50	N:70	N:100	N:150	N:200
1	○	○	○	○	○
2	×	×	○	○	○
3	×	○	○	○	○
4	○	○	○	○	○
5	×	×	○	○	○
6	×	×	○	○	×
7	×	○	○	○	○
8	×	○	○	○	○
9	×	○	○	○	○
10	×	×	○	○	○
11	○	○	○	○	○
12	×	×	○	○	○
13	○	○	○	○	○
14	○	○	○	○	○
15	○	○	○	○	○
16	×	×	○	○	○
17	○	○	○	○	○
18	○	○	○	○	○
19	×	○	○	○	○
20	○	○	○	○	○
21	×	○	○	○	○
22	○	○	○	○	○
23	○	○	○	○	×
24	○	○	○	○	○
25	○	○	○	○	×

表 3.2: 各小領域サイズにおけるラベル付け実験結果

図 3.10: ラベル付けのうまくいかない例 (N 小)

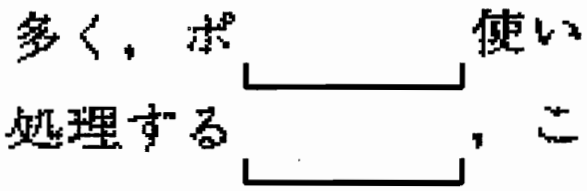


図 3.11: ラベル付けのうまくいかない例 (N 大)

た、単一のしきい値による処理では対処できない文書画像であり、本章で提案したラベル付け手法による処理を行った結果、図 3.13 に示すように正しくラベル付けされた。本実験結果より、全小領域間で統合されて得られた、文書画像全体に対するラベル数は 3 となり、それぞれ文字 (黒・灰色・白抜き) にラベルが割り当てられていることが分かる。同様に、図 3.14, 図 3.15 も総ラベル数 3 となったもので、各文字に対してのラベル付けがなされている。図 3.16 の原画像に対しては、総ラベル数は 4 が得られたが、それらのうちで、文字の部分に対応するラベル 1 種類のみを示した (3.16)。これは、比較的文字のサイズの小さいもの (約 10 pt) に対して、ラベル付けが成功した結果を示している。但し、上記の例は、いずれも小領域サイズ N を 100 dot として実験を行ったものである。

3.6　まとめ

本章では、単一しきい値処理では対処できないような複雑な濃淡文書画像からの文字領域抽出を行うためのラベル付け手法を提案した。

本手法では、濃淡文書画像を小領域に分割した後に、各小領域毎に濃度ヒストグラム

図 3.12: 原画像 (No.11)

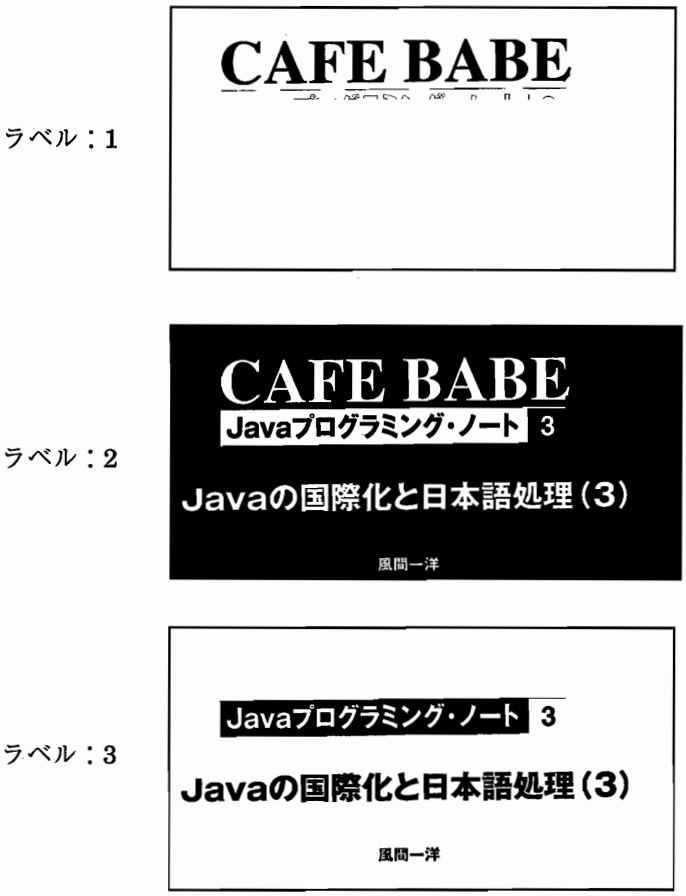
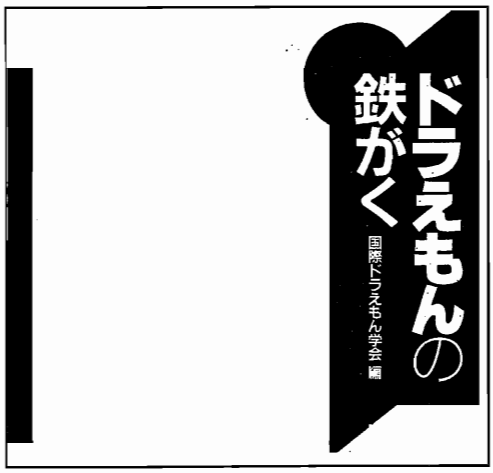


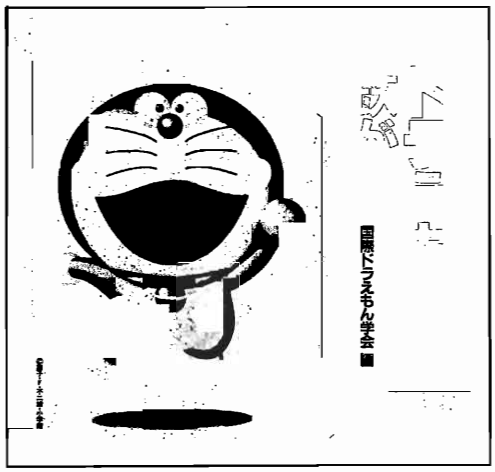
図 3.13: ラベル付け結果

原画像

ラベル：1



ラベル：2



ラベル：3

図 3.14: ラベル付け結果 (No.3)

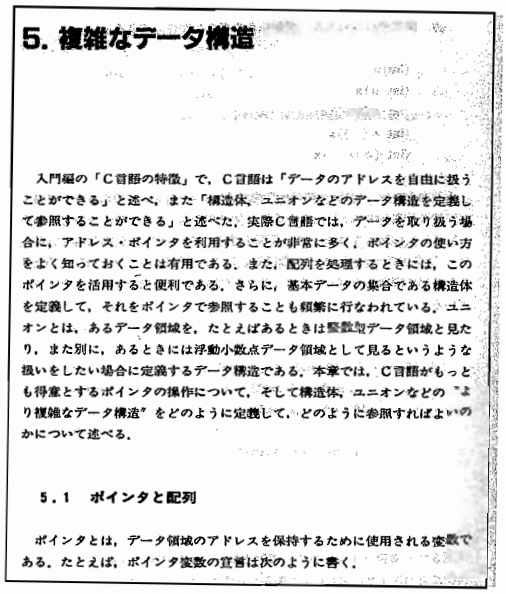
原画像

ラベル：1

ラベル：2

ラベル：3

図 3.15: ラベル付け結果 (No.17)



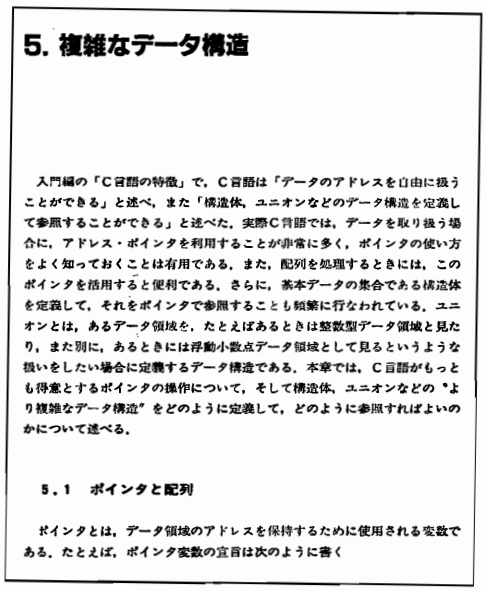
5. 複雑なデータ構造

入門編の「C言語の特徴」で，C言語は「データのアドレスを自由に扱うことができる」と述べ，また「構造体，ユニオンなどのデータ構造を定義して参照することができる」と述べた．実際C言語では，データを取り扱う場合に，アドレス・ポインタを利用することが非常に多く，ポインタの使い方をよく知っておくことは有用である．また，配列を処理するときには，このポインタを活用すると便利である．さらに，基本データの集合である構造体を定義して，それをポインタで参照することも頻繁に行なわれている．ユニオンとは，あるデータ領域を，たとえばあるときは整数型データ領域と見たり，また別に，あるときには浮動小数点データ領域として見るというような扱いをしたい場合に定義するデータ構造である．本章では，C言語がもっとも得意とするポインタの操作について，そして構造体，ユニオンなどの"より複雑なデータ構造"をどのように定義して，どのように参照すればよいのかについて述べる．

5.1　ポインタと配列

ポインタとは，データ領域のアドレスを保持するために使用される変数である．たとえば，ポインタ変数の宣言は次のように書く．

図 3.16: 原画像 (No.6)



5. 複雑なデータ構造

入門編の「C言語の特徴」で，C言語は「データのアドレスを自由に扱うことができる」と述べ，また「構造体，ユニオンなどのデータ構造を定義して参照することができる」と述べた．実際C言語では，データを取り扱う場合に，アドレス・ポインタを利用することが非常に多く，ポインタの使い方をよく知っておくことは有用である．また，配列を処理するときには，このポインタを活用すると便利である．さらに，基本データの集合である構造体を定義して，それをポインタで参照することも頻繁に行なわれている．ユニオンとは，あるデータ領域を，たとえばあるときは整数型データ領域と見たり，また別に，あるときには浮動小数点データ領域として見るというような扱いをしたい場合に定義するデータ構造である．本章では，C言語がもっとも得意とするポインタの操作について，そして構造体，ユニオンなどの"より複雑なデータ構造"をどのように定義して，どのように参照すればよいのかについて述べる．

5.1　ポインタと配列

ポインタとは，データ領域のアドレスを保持するために使用される変数である．たとえば，ポインタ変数の宣言は次のように書く

図 3.17: ラベル付け結果

を解析して代表色とその数を決定し、小領域内のセグメンテーションを行う。それらをすべての小領域間において併合処理をすることで文書画像全体のセグメンテーション結果とした。

小領域サイズを与える N の値を $50, 70, 100, 150, 200$ dot として実験を行い、N が大きすぎても小さすぎても適切にラベル付けができない例を示し、$N = 100$〜150 dot 程度において、対象文書の文字領域に対し良好にラベル付けが行われることを確認した。

図 3.18: サンプル文書画像 No.1

図 3.19: サンプル文書画像 No.3

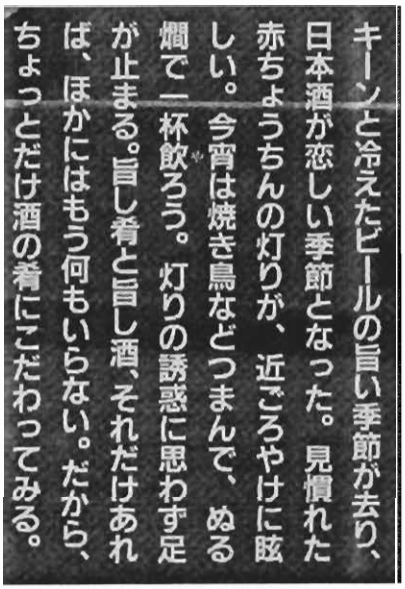


図 3.20: サンプル文書画像 No.14

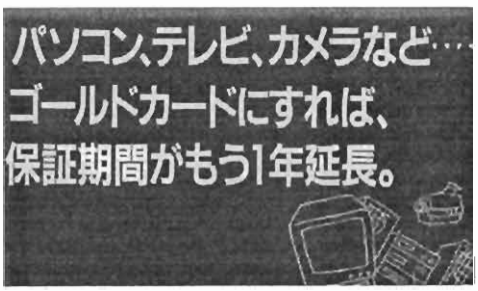


図 3.21: サンプル文書画像 No.15

第 4 章

文字列の抽出

4.1 はじめに

文書画像中の文字列の領域を抽出することが重要な意味を持つことは既に、第 1 章 でも述べた。濃淡文書画像からの文字領域抽出を考えた場合、第 3 章 で述べた手法などで得られた二値画像から抽出することになる。しかし、この二値画像には、文字パターンだけでなく、図や写真等の領域も含まれる。文書画像に対して、文字認識処理を適用する場合、この二値画像から文字領域のみを抽出する必要がある。

文字領域抽出に関する研究としては、局所的に領域分割し文字の等色性に頼る方法 [18][19] や、数学モルフォロジーを用い、細長線領域を文字領域として抽出する方法 [20] 等が挙げられるが、前者は情景画像中の非常に性質の限られた文字 (看板文字) のみを扱い、小さい文字はほとんど考慮されていない。後者は文字以外の細長い線領域の対処が困難という問題がある。また、制約充足の概念に基づいて、日本語印字文字の特徴を反映するコスト関数の最小化によって文字領域の抽出を行う方法 [21] [22] が提案されており、フォーマットに依存しない知識のみを利用するもので、文書のフォーマットに対する自由度が大きい。しかし、処理の対象とする文書画像としてノイズ等の文字以外の要素がほとんど含まれていないものを考えている。

本研究では、“文書中の文字列は縦書き及び横書きで書かれており、局所的に見れば直線性を有している” という、広く一般の文書画像に当てはまる、文字列の幾何学的特性に着目した。本章では、この特性を利用した、二値の文書画像からの文字列画像抽出法につ

いて述べる。

4.2 提案手法の概要

濃淡の文書画像から、文字の領域を抽出するために、第3章 で提案したラベル付け処理を適用する。それによって得られる、ラベル付けされた文書画像から文字列を抽出する。ラベル付けされた文書画像に対し、各ラベルの画素集合において、空間的にどの部分が背景あるいは文字領域を構成しているのかを自動的に判別させることにより、文字領域を背景と分離することが目標である。

まず、最終的なラベルの集合の数 (M 種類) だけ二値画像 $f_B^{(m)}(x,y)$ に分離する (図4.1)。但し、$m = 1, 2, ..., M$ である。

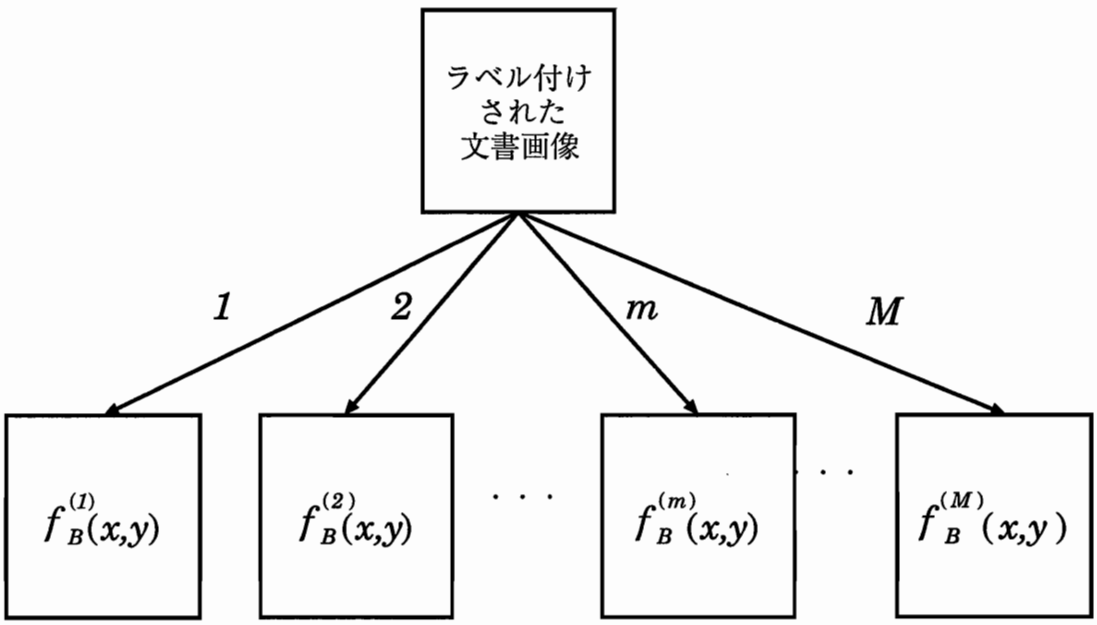


図 4.1: ラベル毎に二値画像へ分離

ラベル毎に分離して得られた二値画像 $f_B^{(m)}(x,y)$ から、前述の文字列の特性を考慮して、文字領域の抽出を行う。

この後の処理の流れは以下のとおりである。

1. 各 m について二値画像 $f_B^{(m)}(x,y)$ に対して外接矩形 $B_u^{(m)}$ を作成する
2. 外接矩形の列に対して文字列 (直線) 性の検証を行い、条件を満足した外接矩形 $B_u^{(m)}$ を集合 $S^{(m)}$ に追加 ($S^{(m)}$ は文字列を構成すると考えられる矩形の集合)
3. 二値画像 $f_B^{(m)}(x,y)$ から $S^{(m)}$ に属する外接矩形の内部の黒画素を文字列画像として抽出する

それぞれの段階の処理について、次節より詳しく説明する。

4.3 文字列画像抽出処理

4.3.1 外接矩形の作成

はじめに、第 3 章 で述べた方法によって得られたラベル集合 L_k^i (併合されていくつかの集合に分類されている) にあらためて通し番号をつけて、集合 $L_1, L_2, ..., L_m, ...L_M$ と置き換える。

それぞれの m に対し、$P(x,y) \in L_m$ となる画素を黒画素とした m 種類の二値画像 $f_B^{(m)}(x,y)$ を得る。$f_B(x,y)^{(m)}$ の黒画素の連結成分に対する外接矩形 (Bounding Box) を作成する。外接矩形とは黒画素の連結成分を囲む最小の長方形である。

4.3.2 フィルタリング

二値画像中には、黒画素の連結成分として文字やその一部だけでなく、ノイズ・背景・絵やその一部なども含まれる。そのため、外接矩形 $B_u^{(m)}$ から次節で述べる文字列 (直線) 性の検証を基に文字列を構成している外接矩形を抽出しようとした際、文字列部分のものでないものも多く抽出する可能性がある。その上、考慮する外接矩形の組み合わせ数も膨大となり効率的ではない。そこで、外接矩形のうち、文字及び文字を構成しているとは考えられにくいものを排除しておくというフィルタリング処理を施す。これにより、外接矩形の組合せ数を減らすことができる。

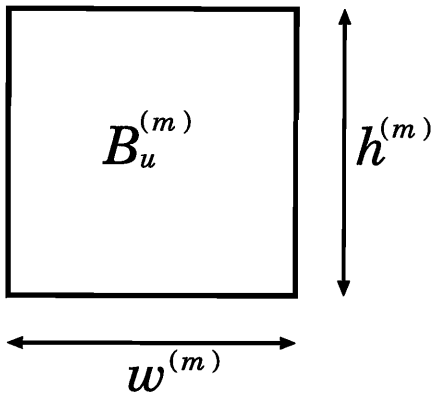


図 4.2: 外接矩形

この排除の基準は次のものである。図 4.2 のように、外接矩形 $B_u^{(m)}$ の幅を w、高さを h とおく。

$$T_{min} \leq w_u^{(m)} \leq T_{max} \tag{4.1}$$

$$T_{min} \leq h_u^{(m)} \leq T_{max} \tag{4.2}$$

$$\frac{\max(w_u^{(m)}, h_u^{(m)})}{\min(w_u^{(m)}, h_u^{(m)})} \leq R \tag{4.3}$$

式 (4.1),(4.2) は、幅・高さの制限を、式 (4.3) は縦横比に関する制限を与える。但し T_{min}, T_{max} は一辺の長さの最小・最大を与える定数で、$R(\geq 1)$ は縦横比の上限を与える定数である。

条件 (4.1),(4.2),(4.3) 式をすべて満足する外接矩形 $B_u^{(m)}$ を文字及び文字の一部に対する外接矩形であるとみなし、以下の検証の対象とする。

図 4.3、 図 4.4 にフィルタリング処理の様子を示す。

4.3.3 文字列候補領域の選定

横書き・縦書きの文字列の幾何的な配置関係を利用して、文字列を構成しているはずの外接矩形の選定を行う。各 $f_B^{(m)}(x, y)$ に対応する外接矩形 $B_u^{(m)}$ の左上の座標を $(x_u^{(m)}, y_u^{(m)})$

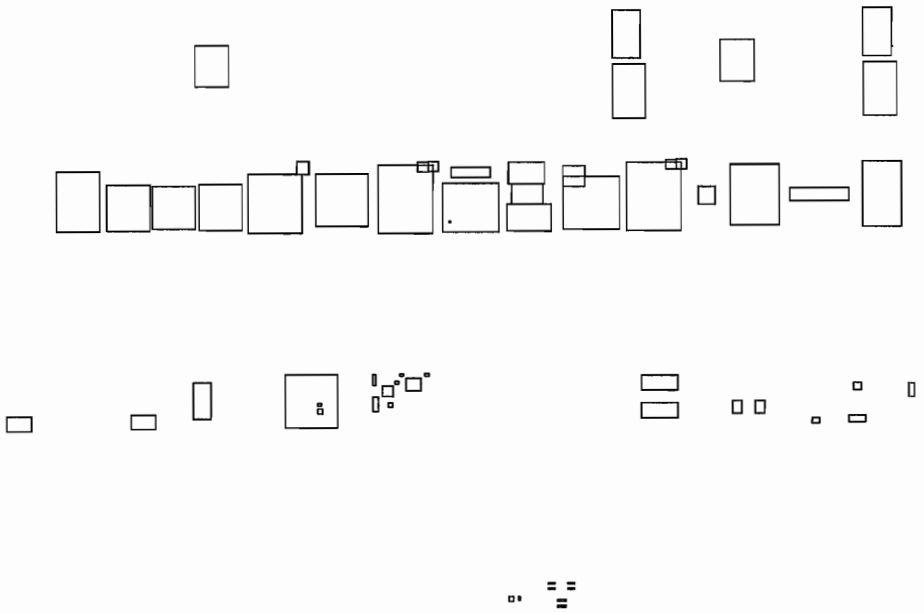


図 4.3: フィルタリング前

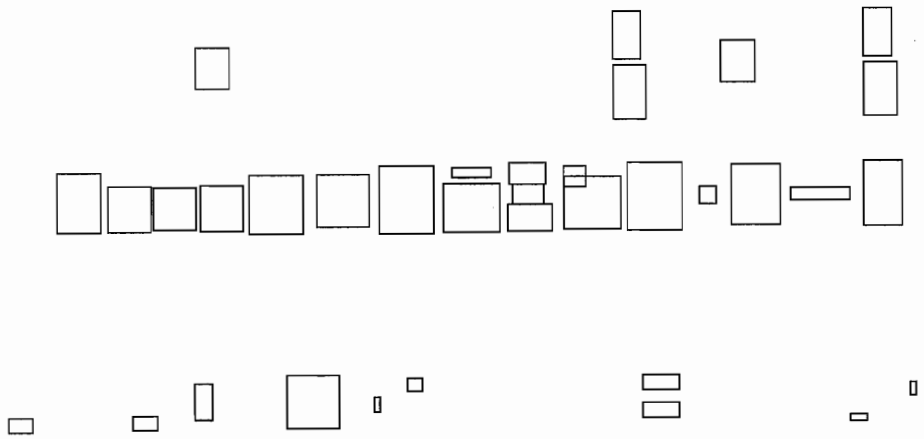


図 4.4: フィルタリング後

とおく (図 4.5)

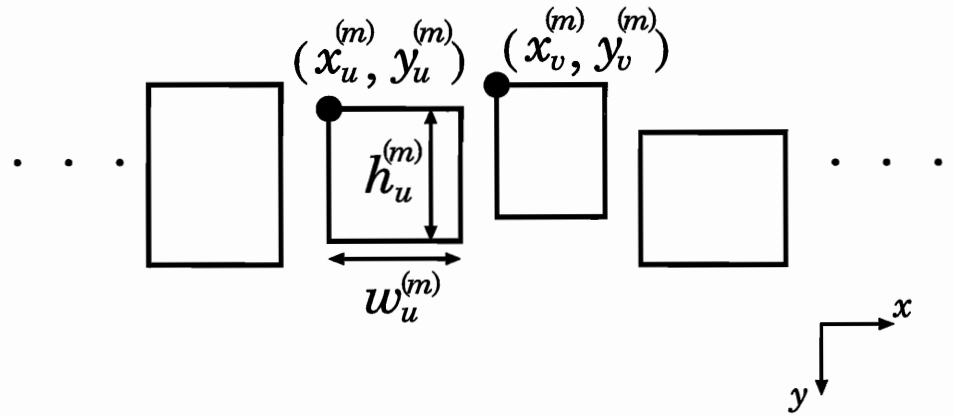


図 4.5: 文字列の直線性の検証

- **横書きの文字列の直線性の条件**
  以下の 1. または 2. の条件のいずれかを満足すれば $B_u^{(m)} \in S^{(m)}, B_v^{(m)} \in S^{(m)}$ とする。

  1.

$$\left|x_u^{(m)} - x_v^{(m)}\right| < T_1 w_u^{(m)} \tag{4.4}$$
$$\& \quad \left|y_u^{(m)} - y_v^{(m)}\right| < T_2 h_u^{(m)} \tag{4.5}$$
$$\& \quad \left|(y_u^{(m)} + h_u^{(m)}) - (y_v^{(m)} + h_v^{(m)})\right| < T_2 h_u^{(m)} \tag{4.6}$$

  2.

$$\left|x_u^{(m)} - x_v^{(m)}\right| < T_1 w_u^{(m)} \tag{4.7}$$
$$\& \quad y_u^{(m)} \leq y_v^{(m)} \tag{4.8}$$
$$\& \quad y_u^{(m)} + h_u^{(m)} \geq y_v^{(m)} + h_v^{(m)} \tag{4.9}$$

T_1, T_2 は定数、$S^{(m)}$ は $B^{(m)}$ における文字列を形成していると判定した矩形の集合である。一般に同一の文字列の中に存在する文字は、大きさがほぼ等しく、近接して配置さ

れている。この性質を満たすような矩形を抽出するための条件が、上記の条件 1. である。しかし、条件 1. では "—" や "ー" 等の文字が拾いきれないことがある。これらの文字や、文字の一部を抽出するための条件が、条件 2. である。

同様にして縦書きの文字列に関しても文字列 (直線) 性の検証を行う。次の条件を用いる。

- **縦書きの文字列の直線性の条件**

1.

$$\left|y_u^{(m)} - y_v^{(m)}\right| < T_1 h_u^{(m)} \tag{4.10}$$
$$\& \quad \left|x_u^{(m)} - x_v^{(m)}\right| < T_2 w_u^{(m)} \tag{4.11}$$
$$\& \quad \left|(x_u^{(m)} + w_u^{(m)}) - (x_v^{(m)} + w_v^{(m)})\right| < T_2 w_u^{(m)} \tag{4.12}$$

2.

$$\left|y_u^{(m)} - y_v^{(m)}\right| < T_1 h_u^{(m)} \tag{4.13}$$
$$\& \quad x_u^{(m)} \leq x_v^{(m)} \tag{4.14}$$
$$\& \quad x_u^{(m)} + w_u^{(m)} \geq x_v^{(m)} + w_v^{(m)} \tag{4.15}$$

以上の処理により、横書き・縦書きの文字列を構成しているとみなされた外接矩形の集合 $S^{(m)}$ ができる。この $S^{(m)}$ に属しているすべての外接矩形 $B_u^{(m)}$ の内部の黒画素を $f_B^{(m)}(x,y)$ から文字列画像とする。但し、黒画素の抽出の際は、濁点や半濁点を抽出し損なうのを防ぐ目的で、対象となる外接矩形 $B_u^{(m)}$ は、上下方向に $0.1h_u^{(m)}$、左右方向に $0.1w_u^{(m)}$ ずつ、それぞれ拡大しておくものとする。

4.4 パラメータの決定

一般に、文書画像中には、様々な大きさの文字 (数 pt ～) が含まれる。前節までに述べてきたアルゴリズム中で示したパラメータは、設定値に応じて、文字の抽出精度に大きな影響を与える。従って、抽出の対象とする文字の大きさや形状を考慮した値に設定する必要がある。

対象とする文書画像のサイズは A5 から A4 程度とした。このとき、文書画像中に存在する文字、文字の一部 (分離文字) の大きさや、文字列内での文字と文字の間隔、分離文字の構成要素同士の距離から、諸定数を決定する。

4.4.1 フィルタリングに関する定数

取り扱う文書画像中に存在する文字のサイズは、約 8 pt から 48 pt 程度である。画像の解像度を 400 dpi としたときの文字の大きさ (dot) の例を図 4.6 に示す[1]。

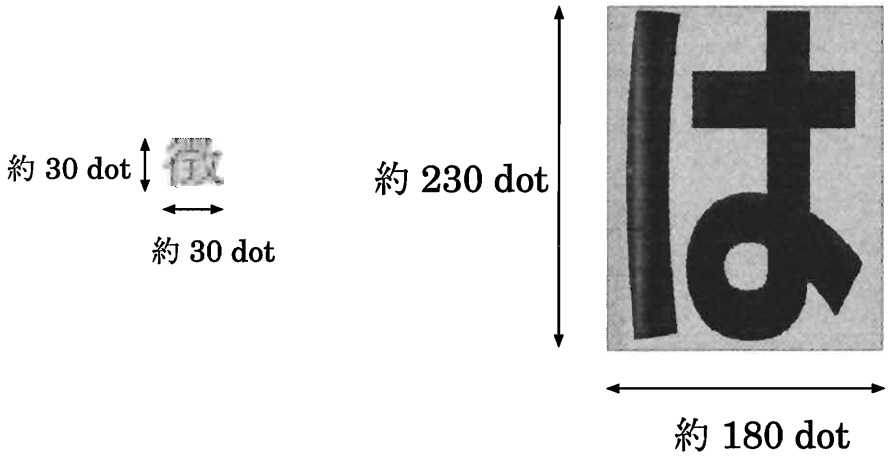


図 4.6: 文字の大きさ

図中の小さいほうの文字は、約 8 pt で、約 30(dot)×30(dot) となっている。分離文字を構成する文字の一部分は一辺が、一文字の一辺の長さの 1/3 程度のものが大半である。大きい文字は約 40 pt で、約 230(dot)×180(dot) である。一文字に外接する矩形を見たとき、フォントによっては、このようにやや縦長な文字となっているものもあるが、基本的には正方形である。48 pt の文字であれば一辺のサイズは約 300 dot となる。また、図 4.6 の例において、文字に対する縦横比を計算すると 230(dot)/180(dot)≃1.28 となる。図 4.6 のような縦長な部分文字でも縦横比はせいぜい 4 程度である。

以上の検証により、文字及び文字の一部のサイズに関する定数 T_{min} は 10(=30/3) dot、T_{max} は 300 dot 程度を、縦横比 R は 5 程度を考慮すれば良いことが分かる。

[1]但し、表示の都合上縮尺を変えてあるので、見た目の両者の文字の大きさの比は実際とは異なる。

4.4.2 文字列性に関する定数

解像度 400 dpi の文書画像中に含まれる文字列の画像の例を図 4.7 に示した。上側の文字列が約 8 pt で、下側の文字列は約 25 pt である。

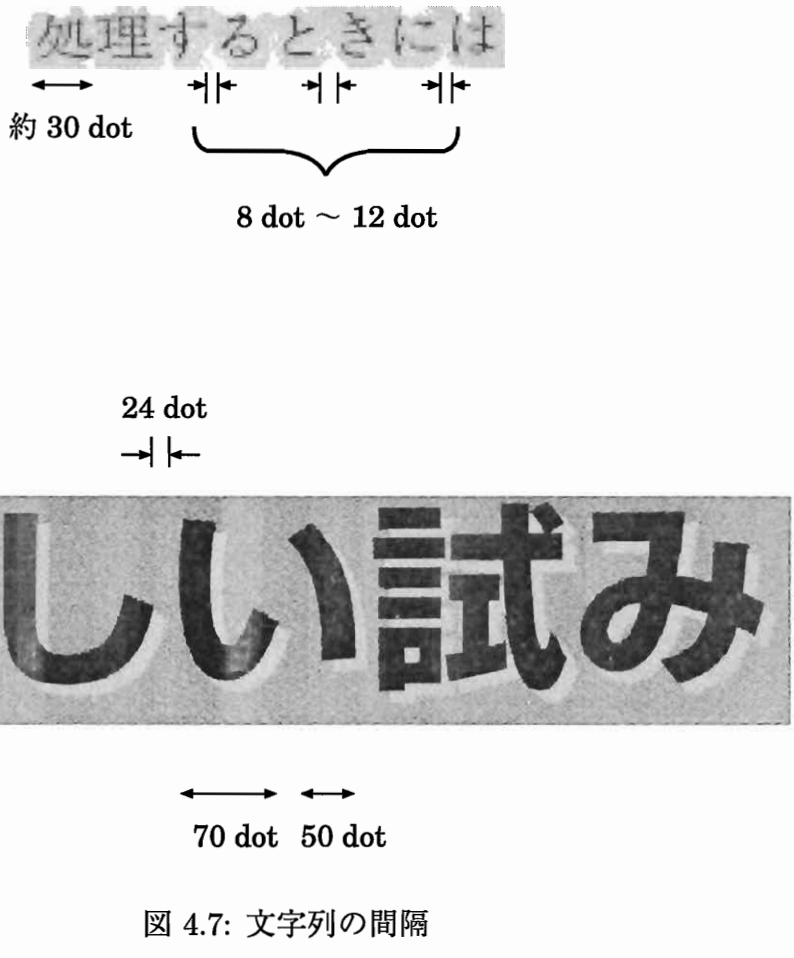


図 4.7: 文字列の間隔

上側の文字列において、文字と文字の距離 (左端と左端との距離) は、およそ 30+10=40 dot であり、左の文字の幅を $w_1(=30)$ とすれば、$40 < 1.5w_1$ である。下側の文字列では約 175(=150+25) で、同様に左の文字の幅を $w_2(=150)$ とすれば、$175 < 1.2w_2$ である。また、文字内部における分離文字同士の距離は通常、文字同士のそれと比較して同じか小さい傾向にある。また、横書きの文字列の場合、“ー” や “—” 等の例を除けば、上端・下端の座標 (y 軸方向) に関し、隣合う文字との差は、小さい方の文字の高さを h とした際

に $h/3$ 程度より小さい。

以上を考慮すれば、隣り合う文字や文字の一部との距離が文字・文字列を構成するかどうかの基準を与える定数 T_1 は 1.5 程度と与えるのが妥当である。また、文字列の直線性に関しては、多少の傾き (横書きなら水平方向に対し ±5% 程度) を考慮に入れて、$\pm h/2.5$ 程度の範囲を同一文字列内の文字と判断するのが適当であると考えられる。すなわち、定数 T_2 は、0.4=(1/2.5) 程度が妥当であろう。

4.5 評価実験

第 3 章 の 3.5 節で用いた 25 文書を用いて、$N = 100$ dot の場合について、文字列領域抽出手法の評価実験を行った。4.4 節のパラメータに関する検討から、諸定数は表 4.1 に示したものとした。

最小文字条件 (T_{min})	10 (dot)
最大文字条件 (T_{max})	400 (dot)
縦横比の上限 (R)	5
T_1	1.5
T_2	0.5

表 4.1: 文字列抽出実験での定数

実験結果から総文字数、抽出成功文字数、抽出率をまとめたものが表 4.2 である。具体的な文字列領域の画像の抽出例を図 4.8〜 図 4.14 に示す。

総文字数	2130
抽出成功文字数	2008
抽出率	94.3%

表 4.2: 文字列抽出実験結果

表 4.2 から、実験では、文字列抽出処理を行った全文書中の総文字数 2130 文字から 2008 文字 (抽出成功率 94.3%) の抽出に成功した。

図 4.8 は上から順に、文字領域抽出前二値画像、文字列を構成していると判断された外接矩形、文字領域抽出結果である。すべての文字が抽出できている。図 4.9 では、下部にある “ももたろうでんてつセブン” の文字列以外は抽出に成功している。この文字列が抽出できなかった原因は、文字が小さすぎたためにフィルタリング処理によって、除外されたためである。図 4.10 では、文書中に図を含んだ例であるが、文字の領域の抽出に成功している。図 4.11 の上段の例では、文字の領域を全て抽出できているが、文字と判断された外接矩形を 10% 拡大してから、その内部の黒画素を取り出す処理のために、文字の近傍にあった文字ではない部分も抽出してしまっている。中段では、“B” の内部の黒画素の塊が抽出されている。意味処理を行っていないので、このような例では、文字に対する矩形かどうかの判断ができない。下段の例で “Java プログラミング・ノート” の右に隣接する “3” が抽出されていない。これは、本手法では文字列性の検証に基づいて文字領域の抽出を行っているため、単独で存在し、一つの連結成分構成される文字は抽出できない。また、図 4.11 と似たような例が、図 4.13 である。文字ではない領域なのに抽出されている部分の理由は、図 4.11 で述べたものと同様である。図 4.12 では、ほぼ全ての文字領域が抽出できているが、ルビの部分が正確に抽出できていない。外接矩形を 10% 拡大してから内部の黒画素を取り出すという処理でもカバーできなかったからである。

文字列の局所的な直線性に着目して、文字列の抽出を行う処理であるため、図 4.14 のように文字列全体では曲線であるような文字列も抽出することができた。しかし、この例のように曲率半径が大きい文字列に対しては適応可能であるが、曲率半径が小さい文字列など、縦・横書き以外の任意方向の文字列抽出処理も今後の検討課題に挙げられる。

抽出に失敗した例を図 4.15 に示す。矢印の流れの順に、二値画像→外接矩形→フィルタリング後、文字列 (直線) 性の検証によって選出された矩形→ (上下・左右方向に拡大した) 選出矩形内部の画像の抽出結果、となっている。「照」の “火 (れっか)” の部分の画像が未抽出である。同様に、抽出できなかった文字の大半は、小さい文字 (10 pt 未満) の分離文字、特に上述 “火 (れっか)” や “濁点” の欠落であった。フィルタリング処理の時点で、文字列候補から除外され、さらに、文字列を構成するとみなされた矩形を上下・左右方向に拡張しても、その内部に存在しなかったために抽出できなかったものである。

このような問題点を解決する案として、ボトムアップ的な処理を、数ステップで行うものが考えられる。すなわち、フィルタリングの条件を緩くしておき、非常に近接する矩

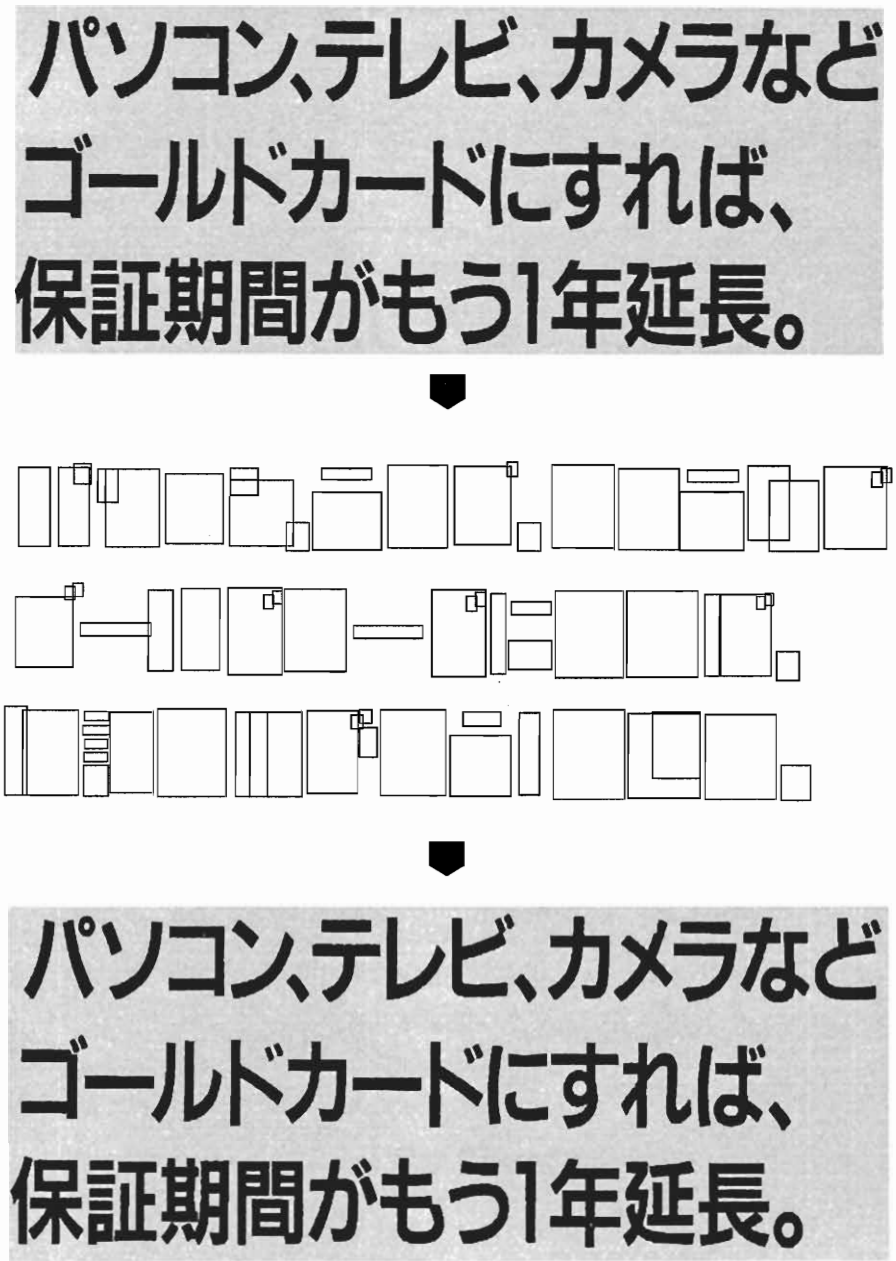


図 4.8: 文字列抽出結果例 (No.15)

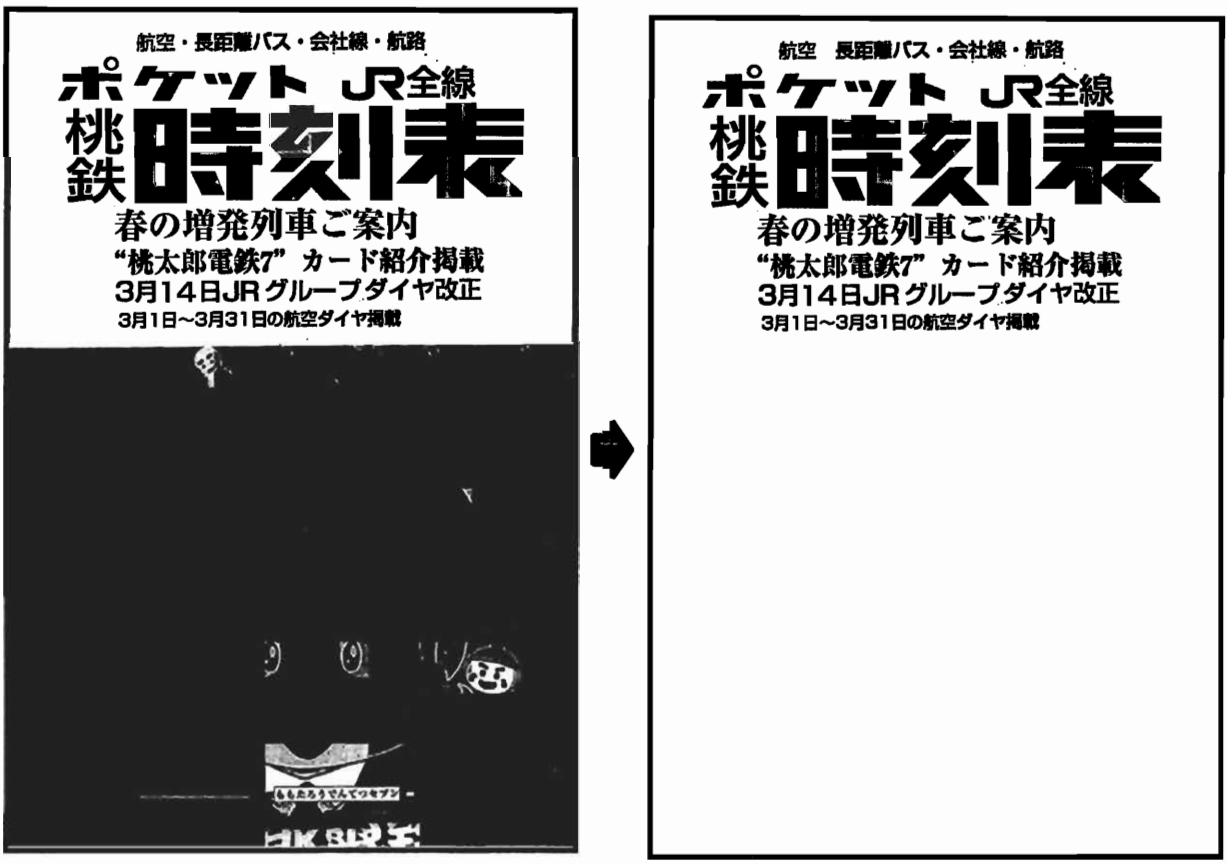


図 4.9: 文字列抽出結果例 (No.2)

文字列領域抽出

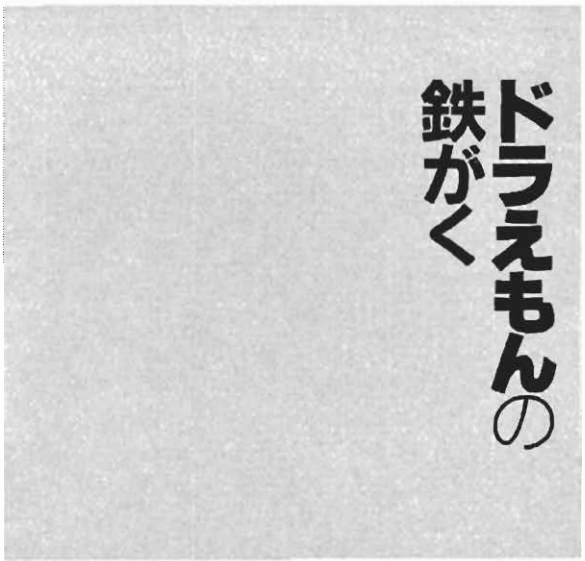


図 4.10: 文字列抽出結果例 (No.3)

図 4.11: 文字列抽出結果例 (No.11)

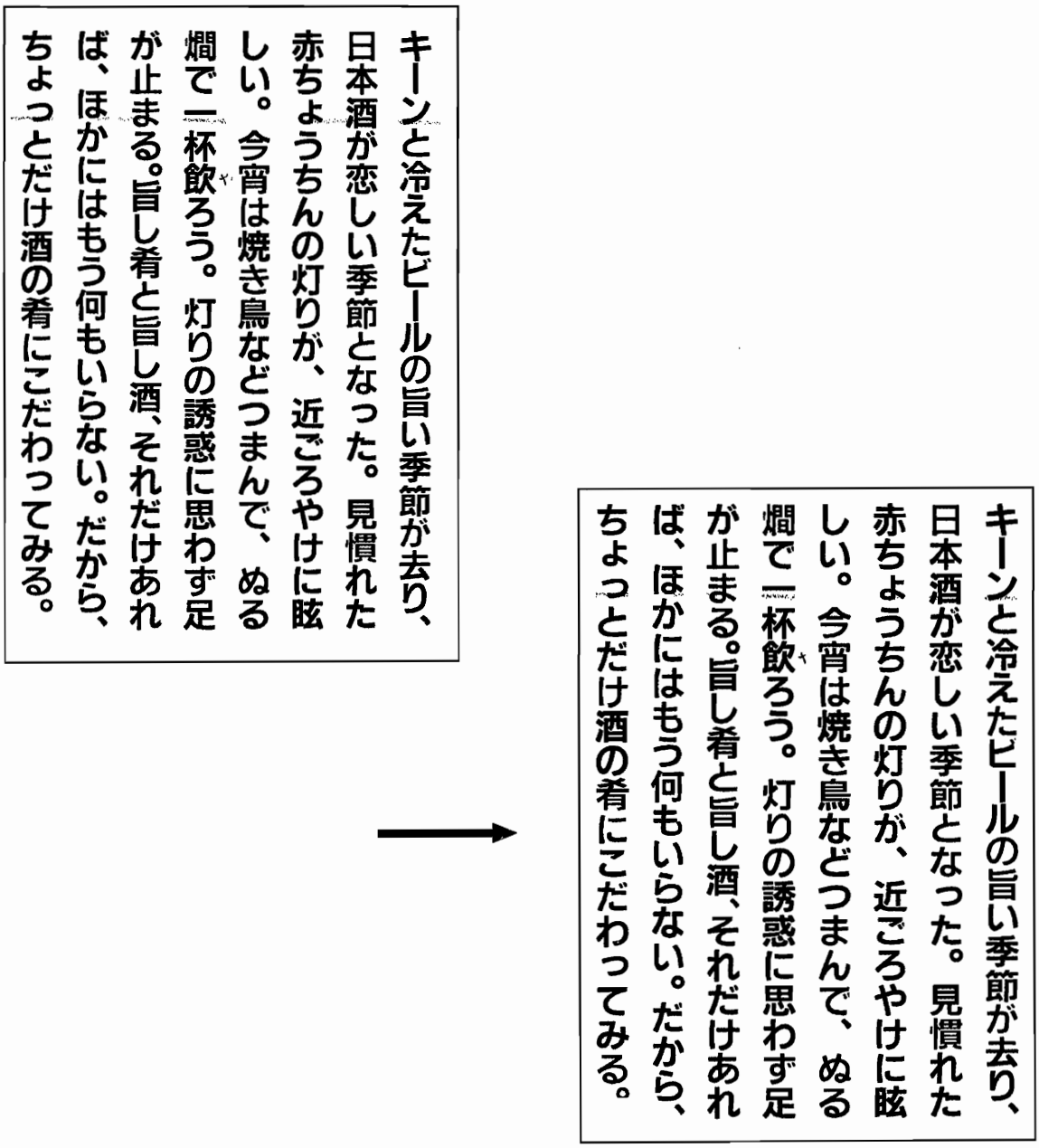


図 4.12: 文字列抽出結果例 (No.14)

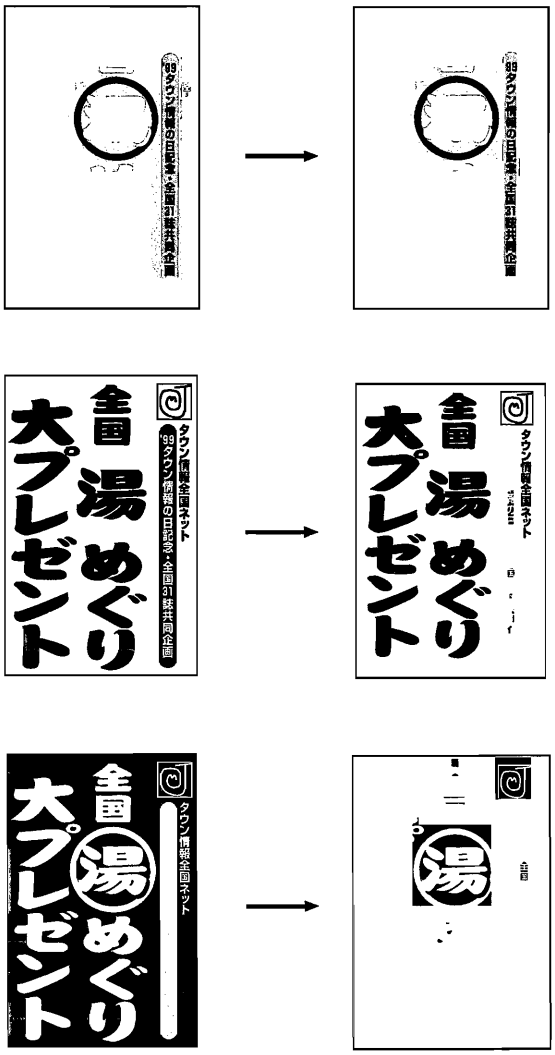


図 4.13: 文字列抽出結果例 (No.17)

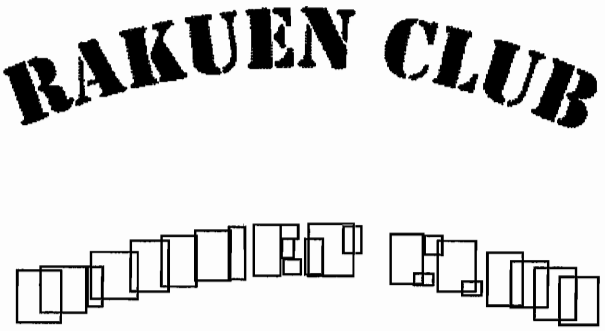


図 4.14: 文字列抽出結果例 (No.22)

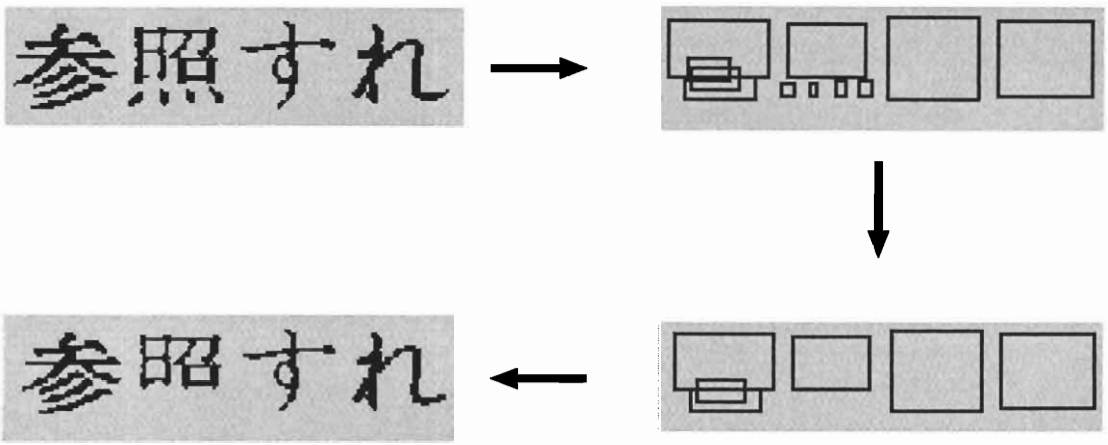


図 4.15: 抽出失敗例

形同士をまとめて一つの矩形にする。さらに、まとめた矩形について縦横比等の条件から検証対象にするか否かの判定を行い、文字・文字列を構成している矩形を探す操作を繰り返し行う、という処理である。この処理を実現することができれば、分離文字の一部欠落を減らす効果を期待できる。

また、“—” のような文字についても抽出しきれないものがあった。外接矩形の縦横比が R よりも大きく、文字に対する矩形とみなされなかったことがその理由である。文字か否かの判断が難しいため、今後の課題に挙げられる。

4.6 まとめ

本章では、二値画像からの文字列領域抽出手法を提案した。

本手法では、文書中の文字列は縦書き及び横書きであるという、多くの文書に共通する性質に基づいた、文字列領域抽出手法である。つまり、外接矩形間の縦・横方向への局所的な直線性を検証することで、文字列を構成している外接矩形を選定し、その部分の文字画像を抽出する。実験により約 94.3% の文字列部分の画像の抽出に成功た。10 pt 未満の分離文字に関してはさらに検討の余地があるが、それ以上のサイズの文字 (～48 pt) に関しては、実験結果からほぼ抽出可能であり、本手法の有効性を確認できた。

第 5 章

結論

5.1 本論文のまとめ

第 1 章 では本研究の背景として、近年の情報化社会における情報のデータベース化のニーズについて述べた。その中でも文書に関して取り上げ、文書画像処理のひとつの**文字領域抽出**の必要性を論じ、過去の研究とその問題点に触れた。単一しきい値処理における問題点を解決するための手段として多値しきい値による手法を選択し、高精度な文字領域画像の抽出を本研究の目的とした。

研究の対象となる文書画像における制約としては、濃淡画像 (8bit/pixel) の文書画像で、文字は等色であるものを仮定し、グラデーション文字や背景と文字のコントラストが著しく低いものは対象外とした。

第 2 章 においては、文書画像処理の概要をまとめた。これにより文書画像処理と、それにおける本研究の位置付けを示した。

第 3 章 では、濃淡文書画像に対して、濃度ヒストグラムの解析によるラベル付け法を提案した。

本手法では、画像の局所領域における濃度ヒストグラムから自動的に代表色 (濃度) を決定し、それにより局所領域内の画素のラベル付けをし、隣接する小領域間で似た濃度のラベルを統合する。これにより、複雑な背景を持つ文書画像中の文字を背景と分離 (セグメンテーション) することができる。

実験により実際に文字に対するラベル付けが良好になされることを確認した。

第4章では、第3章でラベル付けされた画像に対する文字列画像の抽出法を提案した。

本手法ではまず、各ラベルに対応する二値画像を生成し、それぞれの二値画像に対して外接矩形を作成する。その外接矩形同士の**文字列(直線)性**を考慮して、幾何的な配置に基づいたルールにより文字列を抽出する。

実験により約94.3%の文字画像抽出率を得、本手法の有効性を確認した。また、抽出に失敗した例に関して原因を考察し、改善法を検討した。

5.2 今後の課題

今後の課題としては、まず文書画像のラベル付けについては、濃度ヒストグラム解析法の精錬により、本研究では対象としなかったような、さらに小さい文字(数pt程度)や、画像の品質の低いような文書画像に対してもラベル付けができるような頑健性の追及が挙げられる。ここで言う品質の低い画像とは、例えば、紙質が悪くノイズの多く含まれる画像や、同一文字中で濃淡の変化のあるディザリングの施されたようなもの等を指す。

文字列画像抽出アルゴリズムについては、本研究では限定方向(縦・横)の文字列性のみの検証により文字列領域を抽出したが、実際の文書には任意方向の文字列が存在する可能性がある。従って、任意方向の文字列領域の抽出を考慮した手法の考案が必要である。

参考文献

[1] 松尾，梅田：“濃淡及び色情報による情景画像からの文字列抽出”
電子情報通信学会技術報告，PRU92-121，1992.

[2] 上羽，武田，岡田：“等色線処理によるカラー画像からの文字領域の抽出”
電子情報通信学会技術報告，PRU94-28，1994.9

[3] 上羽，武田，岡田：“等色線処理によるカラー画像からの文字列領域の抽出”
画像電子学会誌，第 25 巻，第 4 号，1996.

[4] 仙田，美濃，池田：“文字列の単色性に着目したカラー画像からの文字パタン抽出法”
電子情報通信学会技術報告，PRU94-29，1994.9

[5] 長谷，丸山，松下，米田，酒井：“カラー文書画像中の文字領域抽出のための領域分割方式”
電子情報通信学会技術報告，PRMU97-217，1998.2

[6] 鎌田，藤本：“低解像度テキスト画像の高速かつ高精度な 2 値化方式”
電子情報通信学会技術報告，PRMU98-165，1998.12

[7] 大津：“判別および最小 2 乗規準に基づく自動しきい値選定法”
電子通信学会論文誌，J63-D，No.4，1980.4

[8] 豊田，野口，西村：“日本語印刷文書における文字の切り出し– 新聞自動読み取りへの応用–”
情報処理学会論文誌，24，4，pp.481-487，1983.7

[9] L.A.Fletcher , R.Kasturi : "A Robust Algorithm for Text String Separation from Mixed Text/Graphics Images"
IEEE Trans. Pattern Analysis and Machine Intelligence , Vol.10 , No.6 , Nov. 1998.

[10] 秋山 , 内藤 , 増田："非接触文字優先切出しによる印刷物からの文字切出し法"
電子情報通信学会論文誌 , J67-D , No.10 , 1984.4

[11] 中村 , 氏家 , 岡本 , 南："ミックスモード通信のための文字領域の抽出アルゴリズム"
電子情報通信学会論文誌 , J67-D , No.11 , 1984.11

[12] 末永："文書・画像の編集と作成"
電子情報通信学会論文誌 , J68-D , No.4 , 1984.4

[13] 林 , 美濃 , 坂井："色マーク手書き記入による図面の自動編集・構成"
情報通処理学会論文誌 , 26 , 4 , pp740-747 , 1985.7

[14] 塩："情景中文字の検出のための動的 2 値化処理法"
電子情報通信学会論文誌 , J71-D , No.5 , 1988.5

[15] 大谷 , 塩："情景画像からの文字パターン抽出と認識"
電子情報通信学会論文誌 , J71-D , No.6 , 1988.6

[16] 高木 , 下田："画像解析ハンドブック"
東京大学出版会 , 1991.

[17] R.W.Ehrich : "A symmetric hysteresis smoothing algorithm that preserves principal features"
CGIP , vol.8 , pp.121-126 , 1978.

[18] 滝沢 , 仙田 , 美濃 , 池田："動画像からの看板文字パターン列の抽出"
電子情報通信学会技術報告 , IE94-133 , PRU94-133 , 1995.3

[19] 劉 , 山村 , 大西 , 杉江 :" シーン内の文字列領域抽出について"
電子情報通信学会論文誌 , J81-D-II , No.4 , 1998.4

[20] L.Gu , N.Tanaka , R.M.Haralick , T.Kaneko : "The Extraction of Characters from Scene Image Using Mathematical morphology"
IAPR Workshop on Machine Vision Applications , November.12-14 , 1996.

[21] K.Gyohten , N.Babaguchi , T.Kitahashi : "Constraint Satisfaction Approach to Extraction of Japanese Character Regions from Unformatted Document Image"
IEICE TRANS. INF. & SYST. , VOL.E78-D , No.4 , APRIL , 1995.

[22] K.Gyohten , T.Sumiya , N.Babaguchi , K.Hakusho , T.Kitahashi : "A Multi-Agent Based Method for Extractiong Characters and Character Strings"
IEICE TRANS. INF. & SYST. , VOL.E79-D , No.5 , MAY , 1996.

研究業績

1. “文書画像のラベル付け法と文字抽出法”
   平山理継, 後藤英昭, 阿曽弘具
   電子情報通信学会 情報・システムソサエティ大会, D-12-13(1998.9)

2. “微少文字パタン抽出のための多値しきい値処理の改良と後処理”
   後藤英昭, 平山理継, 阿曽弘具
   電子情報通信学会 総合大会, 1999.3 (発表予定)

謝辞

本研究を進めるにあたり、全般的な御指導を賜りました東北大学大学院工学研究科 阿曽弘具教授に深く感謝致します。

本論文をまとめるにあたって、貴重な御意見を頂いた、東北大学大学院工学研究科 内田龍男教授、東北大学情報科学研究科西関隆夫教授に大変感謝致します。

また、御指導・御意見を賜わった東北大学情報処理教育センター助手 後藤英昭氏、東北大学大学院工学研究科助手 大町真一郎氏、同 森大毅氏に心より感謝致します。

最後に、多岐に渡ってお世話になった阿曽研究室の皆様に感謝致します。

修士学位論文本審査 ohp 資料

高度文字画像抽出アルゴリズムに関する研究

東北大学大学院工学研究科 電気・通信工学専攻

平山 理継

- 第 1 章 - 序論

— 背景 —

高度情報化社会の発展による OA 機器普及の促進

計算機支援環境 ― 作成
計算機支援環境 ― 利用(読む)

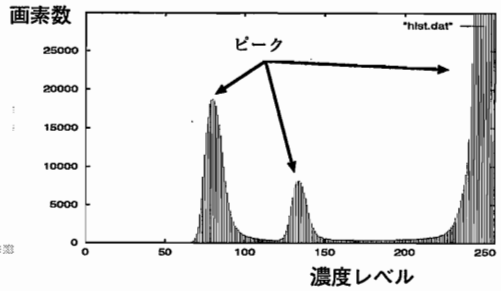
文書画像処理

紙に書かれた文書を画像として

そのまま計算機で扱う処理技術

文字領域抽出

単一なしきい値処理では対応できない文書も多い



要因

- 局所的な明るさの違い
- 複雑な背景

文字抽出へのアプローチ

- **エッジ情報を利用するアプローチ**

  “等色線処理によるカラー画像からの文字領域の抽出” (1994, 上羽ら)

  ・小さい文字に対応できない

- **カラークラスタリングによるアプローチ**

  “文字列の単色性に着目したカラー画像からの文字パタン抽出法” (1994, 仙田ら)

  ・セグメンテーションまでしか行っていない

単一しきい値ではなく**多値しきい値による方法**

本研究の目的

濃淡文書画像を多値しきい値処理によるラベル付けをして文字領域の抽出を高精度に行う

本論文の構成

第 1 章 序論

第 2 章 文書画像処理

第 3 章 文書画像のラベル付け

第 4 章 文字列の抽出

第 5 章 結論

- 第 3 章 - 文書画像のラベル付け

— はじめに —

仮定：ある一連の意味を持った文字領域は等色

— 本研究の方針 —

- 濃淡の文書画像にラベル付けを行う (第 3 章)
- 文字・文字列を形成しているラベルを取り出す (第 4 章)

提案手法の概要

1．文書画像をいくつかの小領域に分割

2．各小領域内で濃度ヒストグラム解析

↓

代表色から各画素にラベルを割当てる

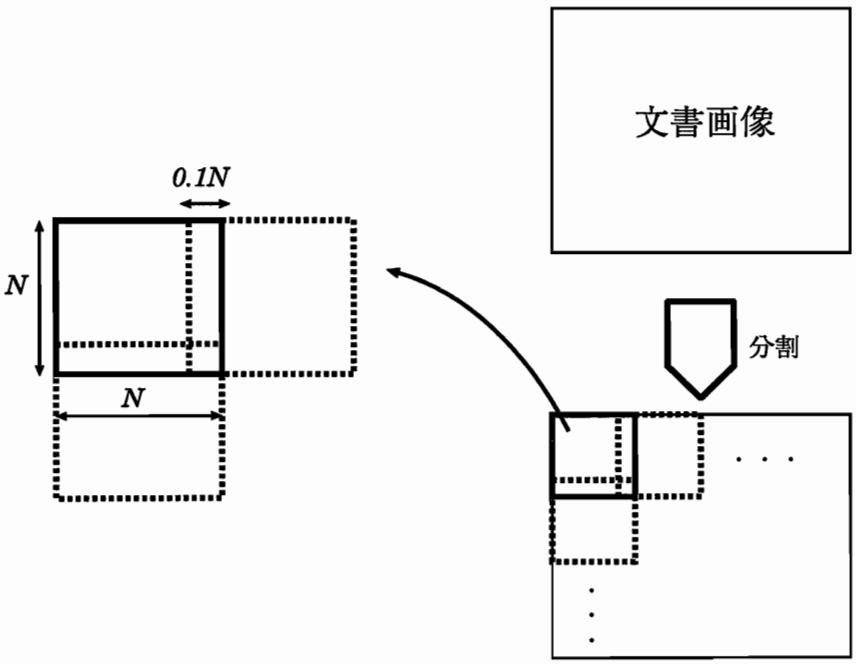
3．全小領域間でラベルを統合する

ラベル付け：

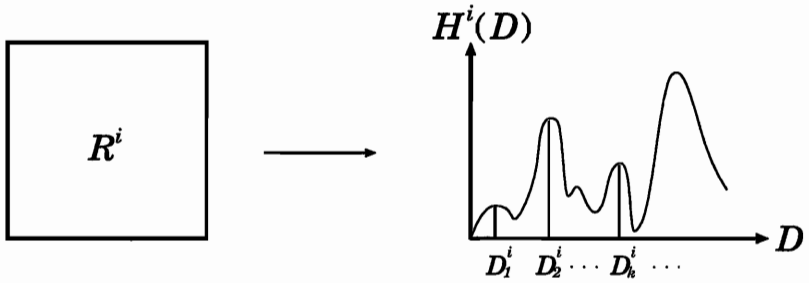
各画素に対して代表色に応じて同じラベルを割り当てること
(画素の濃度が最も近い代表色により)

セグメンテーション：

画像を局所的な特徴の一様な部分画像に分割すること



文書画像を $N \times N$ pixel の小領域へと分割する

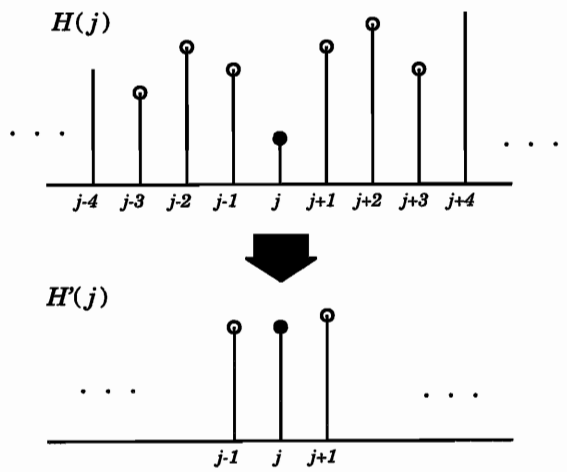


各小領域 R^i から濃度ヒストグラム $H^i(D)$ を得る

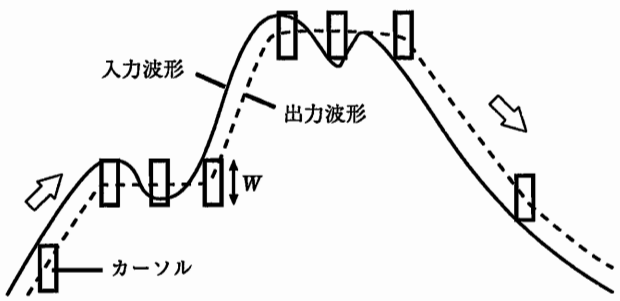
- 濃度ヒストグラム解析により代表濃度 D_k^i を決定
- 濃度が D_k^i で代表される画素に対し、ラベル k を割り当てる
- 同じラベルを持つものをラベルの集合 L_k^i に属させる

濃度ヒストグラムのスムージング

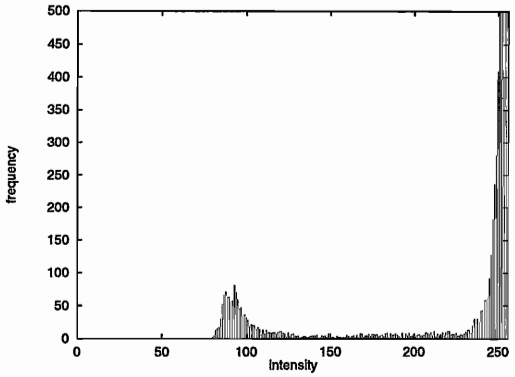
2 段階のスムージング



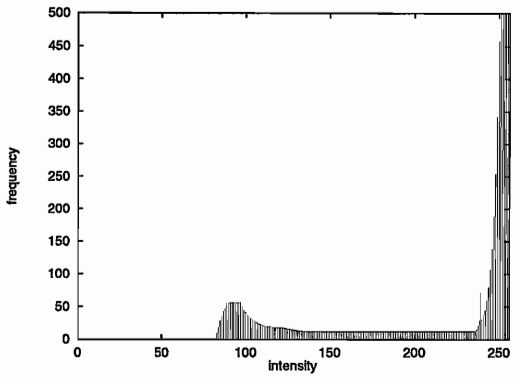
移動平均オペレータによるスムージング



ヒステリシススムージングによる小振幅除去



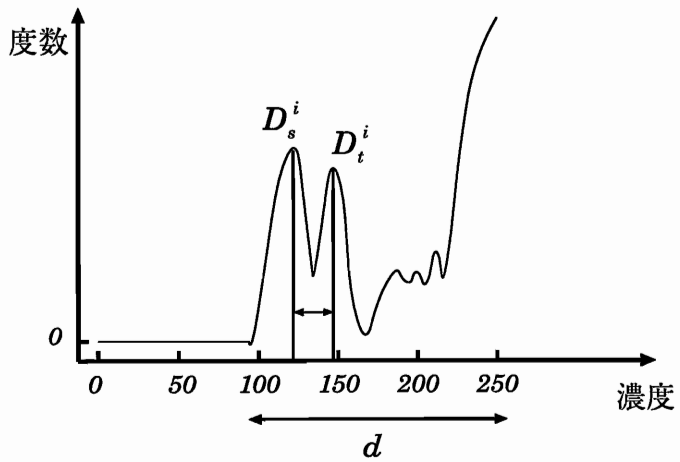
濃度ヒストグラム (処理前)



濃度ヒストグラム (処理後)

スムージングを施した例

小領域内での代表色併合



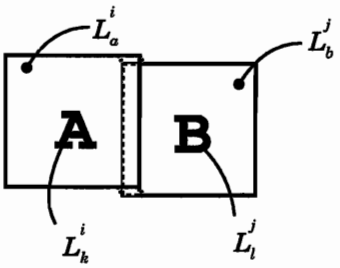
$$\left|D_s^i - D_t^i\right| \leq \max(\alpha, 0.2d)$$

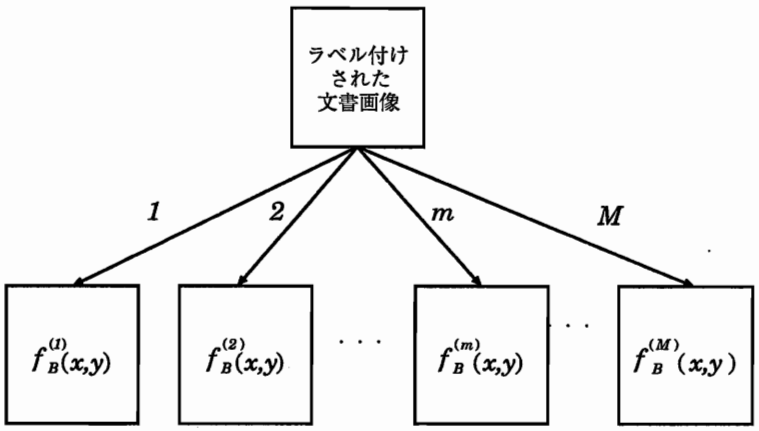
を満たすならば

$$L_s^i \cup L_t^i$$

全小領域間のラベル併合

$\left|D_k^i - D_l^j\right| \leq \theta$ を満たすならば $L_k^i \cup L_l^i$ を繰り返す
θ ：併合条件を決める定数





各ラベルを黒画素とする二値画像を m 種類得る

実験

各種文書画像 (25 枚) に対してラベル付け実験を行った

— 条件・定数 —

使用サンプル	400dpi 濃淡画像 (8bit/pixel)
小領域サイズ (N)	50, 70, 100, 150, 200 (dot)
カーソル幅 (W)	20
マージ定数 (α)	10
ラベル統合しきい値 (θ)	10

ラベル付け結果

文書 No.	N:50	N:70	N:100	N:150	N:200	文書 No.	N:50	N:70	N:100	N:150	N:200
1	○	○	○	○	○	14	○	○	○	○	○
2	×	×	○	○	○	15	○	○	○	○	○
3	×	○	○	○	○	16	×	×	○	○	○
4	○	○	○	○	○	17	○	○	○	○	○
5	×	×	○	○	○	18	○	○	○	○	○
6	×	×	○	○	×	19	×	○	○	○	○
7	×	○	○	○	○	20	○	○	○	○	○
8	×	○	○	○	○	21	×	○	○	○	○
9	×	○	○	○	○	22	○	○	○	○	○
10	×	×	○	○	○	23	○	○	○	○	×
11	○	○	○	○	○	24	○	○	○	○	○
12	×	×	○	○	○	25	○	○	○	○	×
13	○	○	○	○	○						

- 小領域サイズ N → 大

  小さい文字に対するピークの検出ができない

- 小領域サイズ N → 小

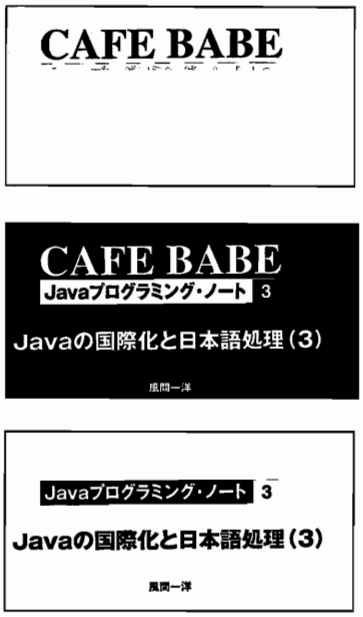
  複数あるべき代表濃度をその数だけ決定できない

サンプル画像

サンプル 1

サンプル 3

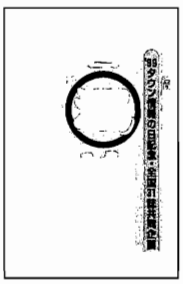
ラベル付け結果



サンプル 11 のラベル付け結果

ラベル付け結果

原画像



ラベル：1

ラベル：2

ラベル：3

サンプル 17 のラベル付け結果

第 3 章のまとめ

小領域サイズを与える N は 100dot から 150dot 程度が適当

- 複雑な背景を持つような文書画像にも適応できる
  ラベル付け法を提案した
- 実験から $N = 100, 150$ dot の時、文字領域に対して
  同じラベルが割り当てられる事を確認した

- 第 4 章 - 文字列の抽出

— はじめに —

二値化された画像中には文字以外のものも含まれる
→ 文字部分の抽出処理が必要 (文字列 (直線) 性に着目)

— 本研究の方針 —

各 m 種の $f_B^{(m)}(x,y)$ に対して外接矩形 $B_u^{(m)}$ を作成

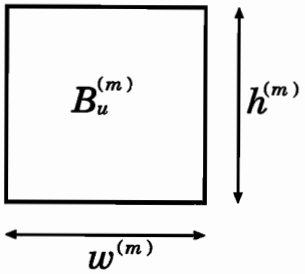
文字列性の検証を行い、条件を満足した $B_u^{(m)}$ を文字列構成矩形集合 S に追加する

$f_B^{(m)}(x,y)$ から S の対応する部分を文字列画像として抽出する

文字列抽出アルゴリズム

文字条件

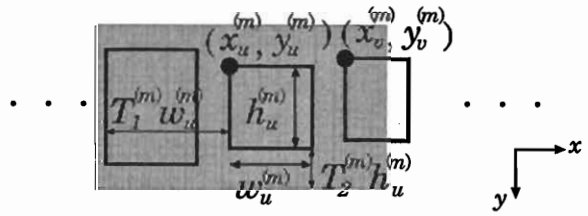
$$T_{min} \leq w \leq T_{max}$$
$$T_{min} \leq h \leq T_{max}$$
$$\frac{\max(w,h)}{\min(w,h)} \leq R$$



T_{min}, T_{max} ：文字サイズの最小・最大の定数

R ：縦横比の上限を与える定数

フィルタリングにより文字を成していると考えられる矩形 $B_u^{(m)}$ のみを考慮



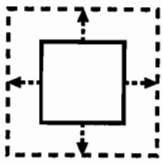
矩形間での文字列性検証

- 横書きの文字列

1. $|x_u - x_v| < T_1 w_u$ & $|y_u - y_v| < T_2 h_u$ & $|(y_u + h_u) - (y_v + h_v)| < T_2 h_u$
2. $|x_u - x_v| < T_1 w_u$ & $y_u \leq y_v$ & $y_u + h_u \geq y_v + h_v$

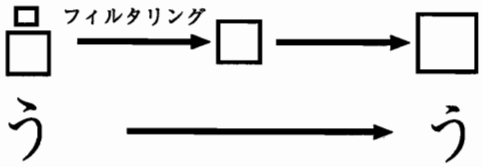
T_1, T_2 ：文字列性判断のための定数

1. or 2. が満足されたら $B_u^{(m)} \in S, B_v^{(m)} \in S$



$B_u^{(m)} \in S$ なる矩形内部の画像を取り出す

$B_u^{(m)}$ は、上下左右に幅・高さをそれぞれ 10% 分拡大する



文字・文字列領域部の切り出し

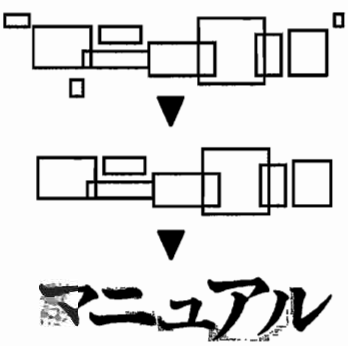
評価実験

第 3 章の実験でラベル付けした 25 文書について文字列抽出実験を行った

定数：$T_{min} = 10, T_{max} = 400, R = 5, T_1 = 1.5, T_2 = 0.5$

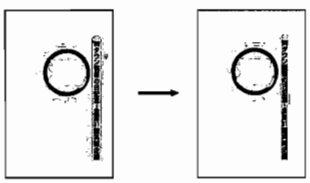
— 結果 —

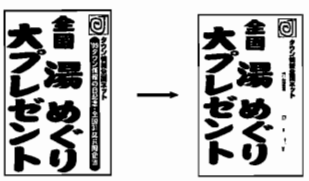
総文字数	2130
抽出成功文字数	2008
抽出率	94.3%

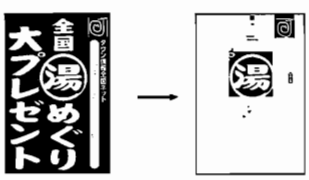


文字列領域抽出

サンプル 3 の
文字列領域抽出結果







サンプル 17 の
文字列領域抽出結果

第 4 章のまとめ

- 二値文書画像中より幾何的な特性を利用して、
  縦書き及び横書き対応の文字列抽出法を提案した
- 実験により約 94.3% の抽出率が得られた

- 第 5 章 - 結論

— まとめ —

- 濃淡文書画像に対するラベル付け法を提案した
- ラベル付け画像からの文字列の抽出法を提案した

— 今後の課題 —

- 濃度ヒストグラム解析法の再検討によるさらなる高精度化
- 任意方向の文字列の抽出