制約充足に基づく文書画像からの文字領域抽出に関する研究

1995年12月

行天　啓二

謝辞

本論文は, 大阪大学産業科学研究所 北橋 忠宏 教授 の御指導の下に筆者が同大学大学院工学研究科（通信工学専攻）在学中に行なった研究の成果をまとめたものである. 博士後期課程において大阪大学産業科学研究所知能システム科学部門複合知能メディア研究分野（北橋研究室）に所属して以来, 本研究を遂行しその成果をまとめるにあたり, 北橋 忠宏 先生 には終始懇切丁寧なる御指導, 御鞭撻を賜った. ここに深甚なる感謝の意を表する次第である.

大阪大学工学部通信工学教室 故 手塚 慶一 名誉教授 には, 筆者が同大学工学部において通信網工学講座配属の際に研究者としての姿勢をお教え頂き, 博士前期課程在学中多大なる御薫陶を賜った. ここに心から感謝申し上げる.

本論文をまとめるにあたり, 大阪大学工学部通信工学教室 池田 博昌 教授 には有益な御教示, 御助言を賜った. また, 大阪大学工学部通信工学教室 長谷川 晃 教授, 倉薗 貞夫 教授, 森永 規彦 教授, 前田 肇 教授 には同大学における研究の機会を与えて頂くと共に, 講義を通じて通信工学一般および各専門分野に関し様々な御指導, 御教示を賜った. ここに衷心より御礼申し上げる.

本研究の全過程を通じ, 直接御指導頂き, 叱咤激励と共に終始有益なる御助言を頂いた大阪大学産業科学研究所 馬場口 登 助教授 に厚く御礼申し上げる.

大阪大学産業科学研究所北橋研究室に所属して以来, 同研究室助手 角所 考 博士には本研究の細部に至るまで昼夜をおかず熱心な御討論, 的確なアドバイスを頂いた. また, 近畿大学理工学部 淡 誠一郎 講師, 北橋研究室助手 高野 敦子 先生, 松下電工株式会社 顧 海松 博士には御厚意溢れる御助言と御支援を頂いた. ここに深く御礼申し上げる.

筆者が博士前期課程において在籍していた通信網工学講座において, 種々の面で御協力を頂いた大阪大学言語文化研究科 中西 暉 教授, 同大学工学部通信工学教室 岡田 博美 助教授, 同教室 山本 幹 講師, 同教室助手 戸出 英樹 先生, 同教室 後藤 嘉代子 技官, 並びに, 大阪大学工学部情報システム工学教室 大川 剛直 講師 に御礼申し上げる.

筆者が現在属する北橋研究室において, 芦田 昌也 氏, 李 明浩 氏には同級生として色々とお世話になった. また, 筆者の研究仲間であった田中 清 氏（現, NTT）, 上田 俊弘

氏（現, JR 東海）, 太田 裕樹 氏（現, 大和銀行）, 隅谷 倫子 さん（現, 凸版印刷）にはミーティングなどを通じ, 本研究を遂行する上での様々なヒント, アイデアを頂いた. ここに深謝申し上げる.

最後に, 李 仁浩 氏 をはじめとする北橋研究室の諸兄には公私にわたり様々な面でお世話になった. ここに記して感謝の意を表する次第である.

内容梗概

本論文は, 著者が大阪大学大学院工学研究科（通信工学専攻）在学中に行なった制約充足に基づく文書画像からの文字領域抽出に関する研究の成果をまとめたものであり, 次の５章から構成されている.

第１章は緒論であり, 本研究の背景となる文書画像解析一般に関して概観すると共に, 本研究の目的および位置付けについて述べる.

第２章では, 文書画像解析における文書画像からの文字領域抽出問題について議論する. まず, これまでに提案されてきた文字領域抽出法を概観し, 従来手法のアプローチを明らかにする. 概して, 多くの従来手法は書式に依存した処理形態を有していた. 具体的には, 書式のある文書画像を対象とし, 書式に関する知識を利用して文字領域を抽出するというアプローチに基づいていた. これに対し, 任意の文書への適用を考えた場合, 書式の種類や有無に全く依存しない文字領域抽出処理が要求される. この場合, 書式に依存しない知識のみを用いて文字領域を抽出しなければならない. 以上の観点に基づき, 書式に依存しない一般的な知識としてどのようなものを挙げることができるかについて検討する.

第３章では, 前章で挙げた書式に依存しない一般的知識を利用することを目指し, 制約関数の最小化により文字領域を抽出する手法について論じる. 通常, 与えられた複数の知識を全て満たすような解を獲得するためには, 各知識を逐次的に考慮することにより所望の解を絞り込むというアプローチがとられる. しかし, 所望の解が全ての知識を完全に満たしているという保証がない場合, 全ての知識を適度に満たすような解を獲得することが理想的である. このような問題解決を実現する一アプローチとして, 制約充足に基づいた処理がある. 具体的には, 全ての知識を所望の解に対する制約と捉え, 全ての制約を一つの制約関数に反映させることを考える. 制約関数については, その最小解が全ての知識を適度に満たす所望の解になるように構成する. その結果, 制約関数の最小化により, 全ての知識を適度に満たすような所望の解を獲得することができる. ここでは, 以上に述べた制約充足の概念を文字領域抽出問題に応用している. すなわち, 一般的知識を制約として捉えた上で, 全ての制約を反映した制約関数を定義する. 制約関数を最小化することにより, 全ての一般的知識を適度に満足する所望の文字

領域を抽出することができる.

第4章では, 前章で提案した制約関数の最小化による文字領域抽出法を拡張することにより, 文字領域抽出過程において文字列の構造について考慮した, より精度の高い文字領域抽出法について論じる. 前章で論じた文字領域抽出法は, 一般的知識として各文字の概略的な形状や局所的な文字配置に関する知識のみを利用していたため, 文字列としての構造が不適切な文字領域抽出結果が得られるという欠点を有していた. この問題に対処するためには, 文書中の各文字列構造を把握し, その構造を考慮した上で文字領域を抽出するような処理が必要になると推察される. ところが, 文書中には様々な特性を有する文字列が混在する. そのため, 各文字列固有の構造を文字領域抽出過程において考慮するためには, 各文字列ごとに個別の処理が施されるような処理形態が要求される. そこで, このような処理を実現する枠組として, 分散処理の一形態であるマルチエージェントシステムに着目し, マルチエージェントシステムを導入した文字領域抽出法について検討する. 本システム中では, 処理単位となるエージェントは基本的に文書中の各文字列を担当し, 前章で提案した制約関数の最小化による手法を用いて自分が担当する文字列から文字を抽出する.

第5章は結論であり, 本研究で得られた成果を総括すると共に, その意義, 及び今後の課題について述べる.

目次

第 1 章

緒論

人類の文化的行動を支える要因の一つとして, 人間相互のコミュニケーションを挙げることができる. コミュニケーションとは, 各人が有する意思・思想を相互に伝達するものであり, 主に言語による対話や伝送情報が記述された紙などを媒体にすることにより実現される. 特に, 紙を媒体にしたコミュニケーションでは, 言語情報を文字により記述したり, 言語では表現することができない各種情報を図や表などにより表現することが可能となるため, 言語による対話以上の効率的かつ大容量の情報伝送が実現される [篠田 92]. また, 通信媒体としての紙は, 人間にとって極めて便利な特性を有している. 例えば, 紙は空間的な情報記録媒体であるため, 紙上に散在する各種情報の相互参照が可能となる. また, 紙には音声などのような時間軸がないため, 紙中の情報を利用者側のペースで獲得することができるという利点がある. さらに, 紙はそれ自体が情報伝送媒体であると同時に情報蓄積媒体ともなりうる. 情報が記述された紙をそのままの形で蓄積しておくことができる. 以上のような様々な利点を背景にして, 紙を媒体にした情報伝送は, 古来から人間社会において極めて重要な位置を占めている. このように, 情報伝送あるいは情報蓄積という目的の下に用いられる各種情報が記述された紙媒体を文書と呼ぶ.

近年, 文書による情報伝送に拍車がかかりつつあり, 人間社会に様々な文書が氾濫している. この背景には, 高度情報化社会の発展に伴う各種ＯＡ機器の普及があると考えられる. ワードプロセッサやデスクトップパブリッシングなど文書作成のための計算機支援環境が急速に発展し, 企業内ではもちろんのこと, 一般の個人ユーザでさえも, 容易に様々な文書を作成できるようになった. その結果, 極めて重要な情報から, 日常的な情報まで, あらゆる情報が文書化の対象となり, 各種文書の氾濫に結び付いてし

まっている. 人間社会における文書の氾濫に対し, 情報の受け手側がその情報の洪水に対処しきれなくなりつつあることが, 現在, 社会問題になっている.

このような文書の氾濫に対し, 文書情報の受け手側の支援を計算機に実行させることができないかという要求が高まりつつある. すなわち, 計算機に文書を「読む」機能を持たせることにより, これまでの「計算機→文書」という一方通行の情報の流れに対し, 「文書→計算機」という情報の流れを作り出すことはできないかという問題である. このような技術は一般に文書画像解析と呼ばれ, これまで様々な角度から検討されてきた [美濃 93].

一概に文書を「読む」機能といえども, そのような機能を実現するために必要となる処理は, 目的に応じて様々なものがある. どのような情報を, 文書中から「読む」, すなわち, 獲得するのかによって, 施すべき処理は変わってくる. ただし, いずれの目的においても, まず与えられた文書から, 文章, 図, 表, グラフ, 写真などの各領域を抽出し, その後に各領域に対して目的に応じた処理を施すという処理形態がとられることが一般的である. この意味で, 文書画像中の各種領域の領域抽出処理は文書画像解析における根幹の技術であると位置付けることができる. 以下では, 各目的に応じた文書画像解析の処理形態を概説し, 文書画像解析における領域抽出問題について論じる.

文書画像解析は, 主に以下のような目的の下に研究されている [美濃 93].

- ファクシミリ通信における文書画像のデータ量圧縮
- コピーやファクシミリなどで行なわれる文書画像の二値再生処理
- 文書のデータベース化
- マルチメディア情報システムにおける文書情報のメディア変換

ファクシミリ通信などにおける文書画像解析に関する研究では, いかに効率的に文書中の情報を圧縮するべきかが主眼とされてきた. このような処理系では, 文書画像中の各領域の抽出後, 各領域それぞれの信号的特性を考慮して, 最も効率的に文書中の情報を圧縮することが目標となる. 例えば, 文章, 図, 表, グラフなどが記述されている領域については, 画素あたりの濃淡値は二値で十分であるが細かい解像度が必要である. 特に, 文字が記述されている文章領域については, 人間が文字を認識するために必要な最低限の解像度で読み取られなければならない. 逆に, 写真が存在する領域について

は, 解像度は粗くても画素あたりの濃淡値は多値である方が望ましい. 以上のような要求に則り, 各領域の情報を各領域に最も合った形で圧縮することにより, 効率的なデータ量圧縮が実現される.

コピーやファクシミリなどで利用される二値再生処理を目的とした文書画像解析においても, 先のデータ量圧縮に関する文書画像解析と同様の処理形態がとられる. 文章, 図, 表, グラフ, 写真などの領域が抽出された後, 各領域の信号的な特性が把握され, その特性に応じて最も適切な二値再生処理が施される. これにより, 文書画像が効率的に二値化される.

さて, 以上のようなデータ量圧縮, もしくは二値再生処理を目的とした文書画像解析における文書画像の領域抽出問題について考えてみる. ここでの領域抽出の目的は, 文書中の文章, 図, 表, グラフなどの各領域を抽出し, 各領域の信号的特性に応じた処理を適用することにある. この場合, 抽出された各領域に要求されるのは領域内における信号的特性の共通性であり, 各領域における意味的な共通性は必要ではない. したがって, 文書画像の領域抽出処理が厳密に正確である必要はなく, 領域抽出結果が文章, 図, 表, グラフ, 写真などに代表される各領域と完全に一致しないことも許される. 以上のような理由により, データ量圧縮や二値再生処理を目的とした文書画像解析においては, 文書画像の領域抽出問題は極めて簡単になる. 実際, データ量圧縮や二値再生処理を目的としたこの種の研究は, ファクシミリやコピー機などで実用化されている [中村 1 84].

次に, 文書のデータベース化における文書画像解析の位置付けについて考えてみる. この場合, 文書中の意味的な情報を, 文書画像解析によりデータベース中に構造的に蓄積することが目的となる. 具体的には, まず文書画像から文章, 図, 表, グラフ, 写真などの各領域を抽出し, 各領域に様々な認識処理を施すことによりデータベースに蓄積すべき各種情報を獲得する. その後, 獲得した情報はどの領域に属し, 何を表現しているのかなどに関する様々な付加的情報を与え, データベース中に構造的に蓄積する.

このような, 文書のデータベース化を指向した文書画像解析では, 文書画像の領域抽出処理に正確さが要求される. 文書中の各領域を正確に抽出しなければ, その後の認識処理により得られる情報が誤ったものとなるためである. 文書画像中の各領域をいかに正確に抽出するかについては極めて困難な問題であり, これまでに様々な検討がされてきた. ただし, 文書データベース構築を目的とした文書画像解析では, その処理対

象となりうる文書の種類や書式に関する情報が既知であるという前提を置いたとしても問題はない. 以上のような背景により, 対象文書の種類や書式に関する先験的知識を積極的に用いた極めて正確な領域抽出法がこれまでに数多く検討されてきた. 現に, この種の技術は既に確立されつつあり, 新聞 [秋山 86, 駱 92], 学術書 [中村 1 85, 山田 93], 名刺 [黄瀬 89], 帳票 [成瀬 92], 図書目録カード [長谷 87] などに代表される, 各種文書のデータベース構築に応用されている.

最後に, マルチメディア情報システムにおける文書情報のメディア変換を目的とする文書画像解析について考えてみる. マルチメディア情報システムとは, 音声, 画像, 図形など, 形式が異なる情報を組み合わせて扱う情報システムの総称である. 次世代の情報通信を担う新たな技術として, 近年, 盛んに研究及び開発が進められている. マルチメディア情報システムでは, ユーザにとって最も便利な媒体を用いて, ユーザにとって最も判り易い形で文書中の情報を提示することが目的の一つとなっている. このような情報提示法は, 文書中の情報についても同様に有効である. 例えば, 文書中の文字情報を文書画像中から獲得し, 音声情報に変換してユーザに提示する一連の処理や, 文書中の必要な文章のみをユーザにとって最も読み易い書式で表示するような処理は, 文書を扱うマルチメディア情報システムにおいて極めて重要な要素技術になりうると考えられる. 特に, 文字情報の音声情報への変換のように, ユーザにとって最も親しみ易い形で情報を提示するために, その情報が記述されている媒体を変換する技術をメディア変換と呼ぶ. マルチメディア情報システムにおいて, 文書中に記述されている情報のメディアを変換するためには, 文書中の意味的情報を正確に獲得することができなければならない. したがって, 文書データベース構築の場合と同様に, 高性能の文書画像解析が要求される [北橋 90].

ところが, 両者には前提の相違点がある. 文書のデータベース化の場合とは異なり, マルチメディアシステム上で文書画像を扱う際に, 扱うことができる文書の種類を限定することは許されるべきではない. ユーザにとって便利なシステムを構築するためには, ユーザが有する様々な文書を制限なく扱うことができなければならない. 任意の文書中に記述されている情報を正確に獲得し, メディアを変換してユーザに提示することができるような技術が必須である. よって, マルチメディア情報システム上では, 任意の文書に適用可能な文書画像解析, すなわち汎用的文書画像解析が必要になるものと思われる.

しかし, 任意の文書に適用可能な文書画像解析, 特に, 任意の文書から, 文章, 図, 表, グラフ, 写真などの各種領域を抽出する手法はこれまでにほとんど提案されていない. その理由の一つは, これまでに汎用的文書画像解析の必要性が議論されてこなかった点にある. また, もう一つの理由として, 文書の種類や書式などに関する先験的知識を用いずに正確に各領域を抽出することは極めて困難な問題であるという点を挙げることができる. しかし, 前述のようなマルチメディア情報システムの必要性が叫ばれている現在, 汎用的文書画像解析に関する研究は今後盛んに議論されていかなければならないテーマであると言える.

汎用的文書画像解析を確立する上では, 様々な要素技術を開発していかなければならない. 特に, 文書中の主たる情報は文章中の文字により記述されている点を考慮すると, 文書中から文字領域を抽出し認識する技術は極めて重要であると位置付けることができる. このうち, 文字認識に関してはこれまでの様々な検討により, 印刷文字に対してはほぼ100%に近い認識率が達成されている. これに対し, 任意の文書からの文字領域抽出に関する研究についてはまだ検討され始めたばかりの段階である.

以上のような背景に基づき, 本論文では, 任意の文書への適用を指向した汎用的文字領域抽出法について検討する. 第2章では, これまでに提案されてきた文字領域抽出法を概観し, その研究動向を踏まえた上で任意の文書への適応を指向した文字領域抽出法がとるべきアプローチについて検討する. その結果として, 文字領域抽出のためには, 文書の種類や書式に依存しない知識を用いなければならない点を示唆し, そのような知識として, 文字形状及び文字配置に関する知識を具体的に列挙する. 第3章では, 第2章で挙げた文字形状及び文字配置に関する知識を利用した文字領域抽出法について論じる [行天 93b, Gyohten 93, Gyohten 95a]. ここでは, 与えられた知識を最も効果的に利用することができるアプローチとして制約充足の概念に着目し, 文字領域抽出問題を制約関数と呼ばれる関数の最小化問題に置き換えることにより, 制約充足に基づいた文字領域抽出法を論ずる. 第4章では, 第3章で触れた文字領域抽出法を改良し, 文字形状及び文字配置に関する知識ばかりではなく, 文書中の各文字列の構造についても文字領域抽出の際に考慮することが可能な手法について論じる [行天 94, 隅谷 95, Gyohten 95b, 行天 95, Gyohten 96]. 本手法は, 第3章の手法に分散処理環境の一種であるマルチエージェントシステム [奥乃 94] の概念を導入することにより, 各文字列固有の文字列構造を考慮した処理を, 文字列ごとに施すことができる点

を特徴とする．最後に第5章では，本研究で得られた成果を総括すると共に，その意義，及び今後の課題について述べる．

第 2 章

文書画像解析における文字領域抽出

2.1 緒言

文書画像からの文字領域抽出は文書画像解析における重要な要素技術であり, これまでに様々な角度から検討されてきた. 文字領域抽出の対象となりうる文書として, 様々なものが想定される. 特に近年, 新聞, 学術書などの文書データベース構築を目的とした文書画像解析の必要性を背景として, このような書式のある文書からの文字領域抽出問題について盛んに議論されている.

これに対し, ポスターやチラシなど, いわゆる書式のない文書からの文字領域抽出については, これまでにあまり議論されてこなかった. 本研究の目的である, 任意の文書に適用可能な文字領域抽出法を実現する上では, 書式のある文書への適用はもちろんのこと, 書式のない文書への適用も可能となる手法を考えていかなければならない. この場合, どのようなアプローチにより, どのような知識を用いて, 書式のない文書から文字領域を抽出することができるかについてが, 議論の中心となる.

そこで本章では, 文書画像解析における文字領域抽出の位置付けを明らかにした後, これまでに議論されてきた文字領域抽出のアプローチについて概観する. その上で, 任意の文書への適用を指向した汎用的文字領域抽出法を実現するためには, どのようなアプローチが必要になるか, また, 任意の文書から文字領域を抽出するためには, どのような知識が必要になるかについて検討し, それらの知識を具体的に示す.

2.2 文書画像解析

文書中に存在する情報は, 図 2.1に示す様々な構成要素が文書中に混在, または重畳して配置されることにより表現される. 文書画像解析とは, 計算機により, これらの構成要素が存在する領域を同定し, 各領域に対して目的に応じた処理を施すことにより, 文書画像中から何らかの情報を得る一連の処理を指す. 特に, 文書中の構成要素の中でも, 図 2.2に示すような文章, 図, 表, グラフ, 写真は, 文書中に含まれている情報の一単位としてまとまりが良い. したがって, 文書画像解析は, 図 2.3に示すような処理形態を基にして論じられることが多い [美濃 93].

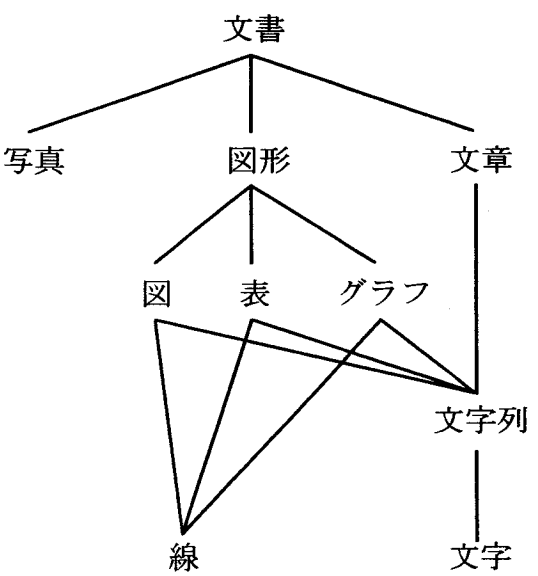
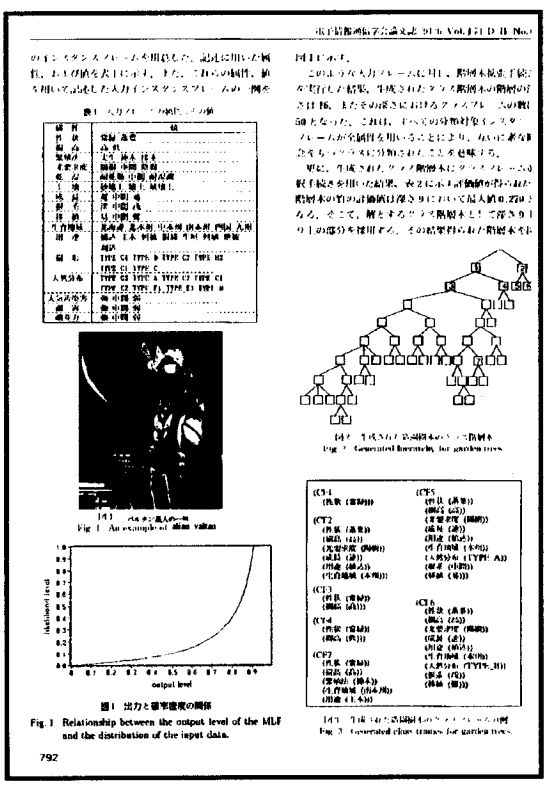


図 2.1 文書中の構成要素

次に, 文書画像解析における文字領域抽出問題について考えてみる. 文字領域抽出とは, 図 2.4(a) に示すような各領域を文書画像中から抽出する処理を指す. 各領域は, 文書中の文字の外接矩形であり, 各領域に文字認識処理を施すことにより, 文字により記述されている意味的な情報を明らかにすることができる. 文字領域抽出処理及び文字認識処理による意味的情報の獲得が必要となるのは, 文書データベース構築やメディア変換のために文書画像解析を用いる場合などである.

では, 各文字領域はどのようにして抽出すれば良いのであろうか. 図 2.3では, 文書



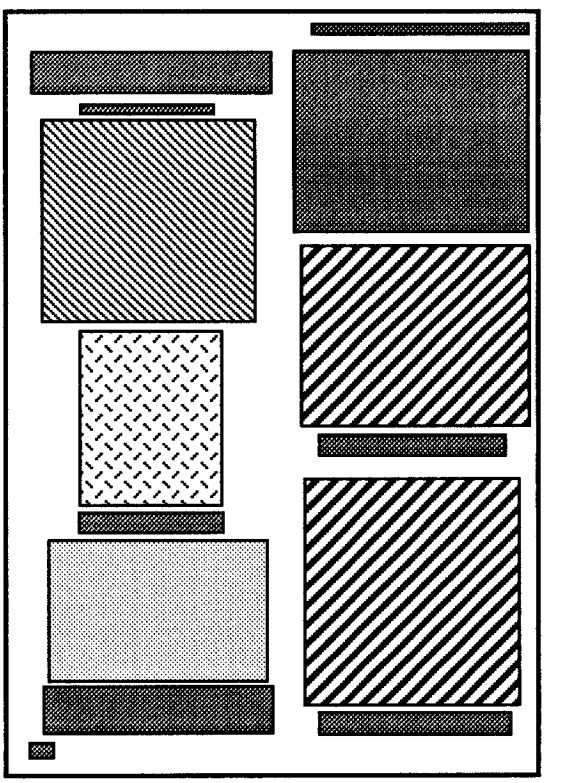


図 2.2 文書画像の一例

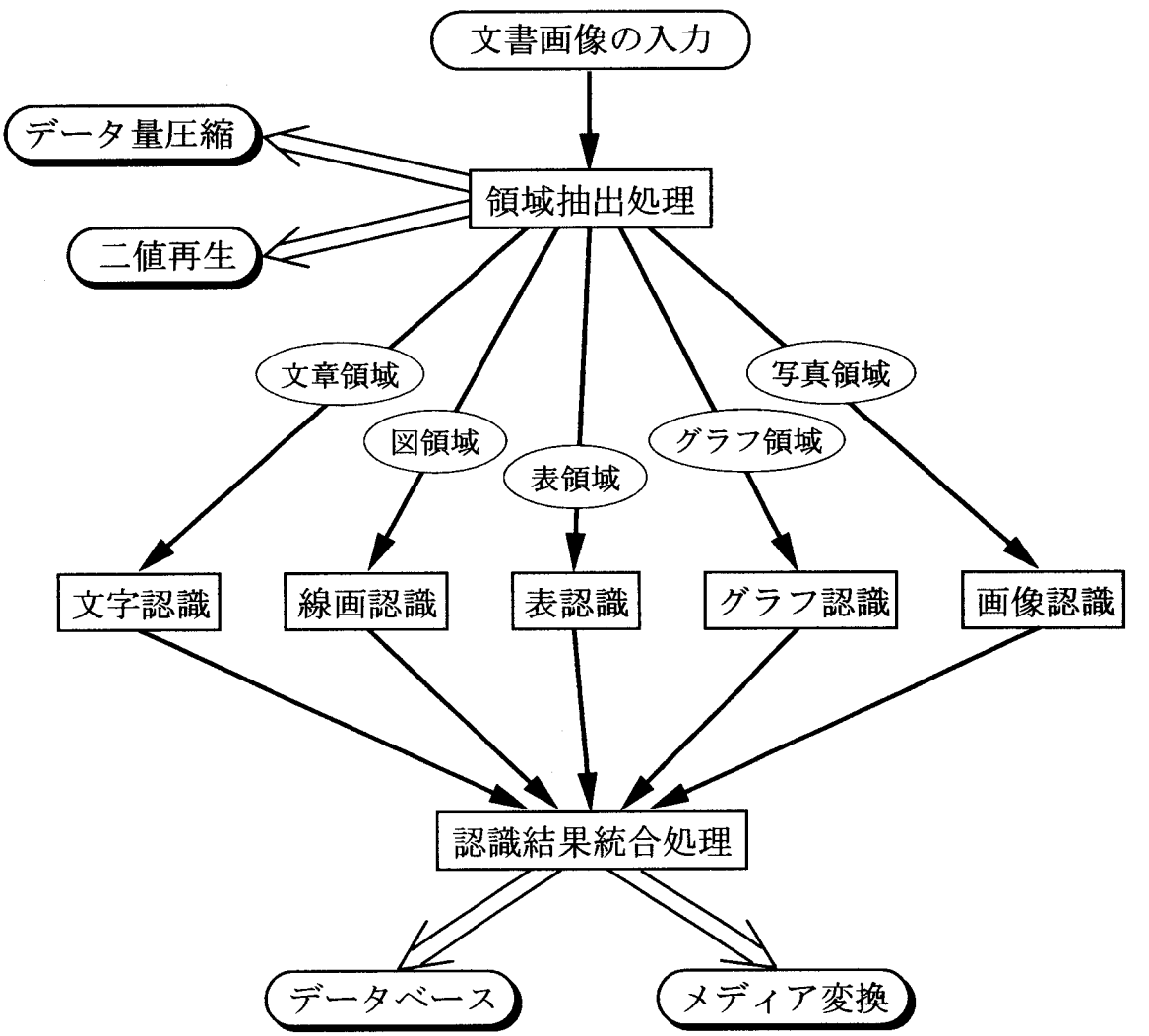
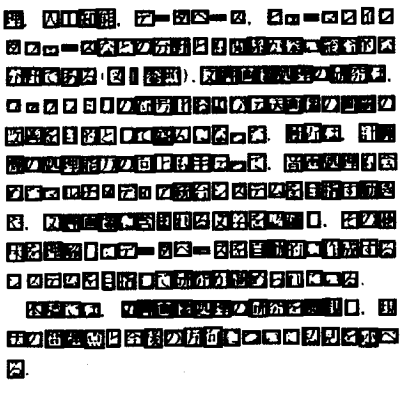
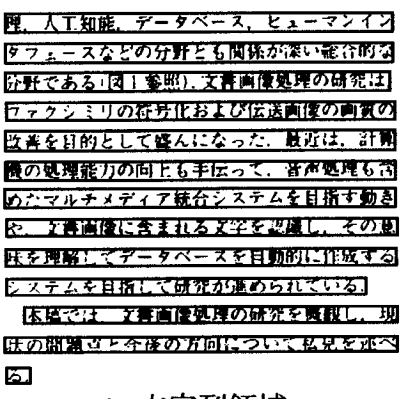

文書画像の入力
データ量圧縮
領域抽出処理
二値再生
文章領域
図領域
表領域
グラフ領域
写真領域
文字認識
線画認識
表認識
グラフ認識
画像認識
認識結果統合処理
データベース
メディア変換

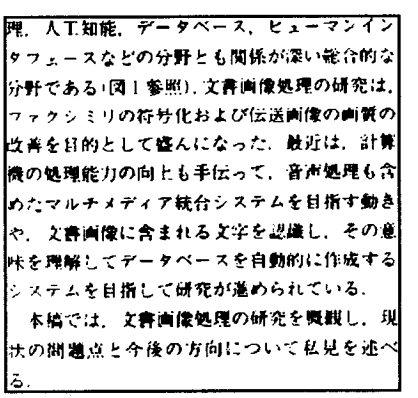
図 2.3 文書画像解析



(a) 文字領域



(b) 文字列領域



(c) 文章領域

図 2.4 文字に関わる各種領域

画像から文章領域を抽出し, その中に含まれている各文字を認識するという一連の処理が示されている. しかし, 実際に文字認識処理を施すためには, 図 2.4(c) に示すような文章領域だけではなく, 図 2.4(a) のような文字領域, また, 図 2.4(b) のような文字列領域も明らかにされていなければならない. 文字領域, 文字列領域, 文章領域が抽出されることによって, はじめて, 文字認識処理を施すことができ, 語や文の意味, そして, 文章中の意味的情報が明らかになる. このように, 文字領域, 文字列領域, 文章領域は互いに密接な関連がある. よって, 文字領域抽出問題を考える場合においては, 文字列領域抽出, 文章領域抽出問題についても考えなければならない. まず, どの領域が文書画像から抽出され, 次にどの領域が抽出され, 最後にどの領域が抽出されるのかに関しては, 各手法により様々である. そこで, 次節では, 文字領域抽出に関する従来手法のアプローチについて概観する.

2.3 従来の文字領域抽出法

文書画像は, ある一定の書式に則って文章, 図, 表, グラフ, 写真などが記述されている文書と, 特に決まった書式がなく各種情報が比較的自由に記述されている文書とに大別できる. 書式を有する文書中の文字は, 書式に則って規則的に配置されている. したがって, 書式に関する知識を積極的に用いることにより, 比較的容易に文字領域を抽出することができる. これに対し, 書式を持たない文書に関しては, 文字配置に関する規則のようなものはほとんどない. この種の文書から文字領域を抽出する際には, 書式に関する知識を一切用いることができない. このため, 書式のない文書からの文字領域抽出は極めて困難であり, これまでに様々なアプローチが試みられてきた.

以上の観点から, 文字領域抽出に関する従来手法のアプローチを分類すると, 図 2.5 のようになる. 以下本節では, 従来の文字抽出法を, 書式のある文書を対象にした手法と書式のない文書を対象にした手法とに大別して概説する.

2.3.1 書式のある文書を対象とした手法

書式のある文書の例として, 新聞, 学術書, 公文書などを挙げることができる. これらの文書に関しては, 文書中に, 各種情報を機能的に記述するために書式の概念が導入されている. その文書特有の書式に則って文書中の構成要素を配置していくことによ

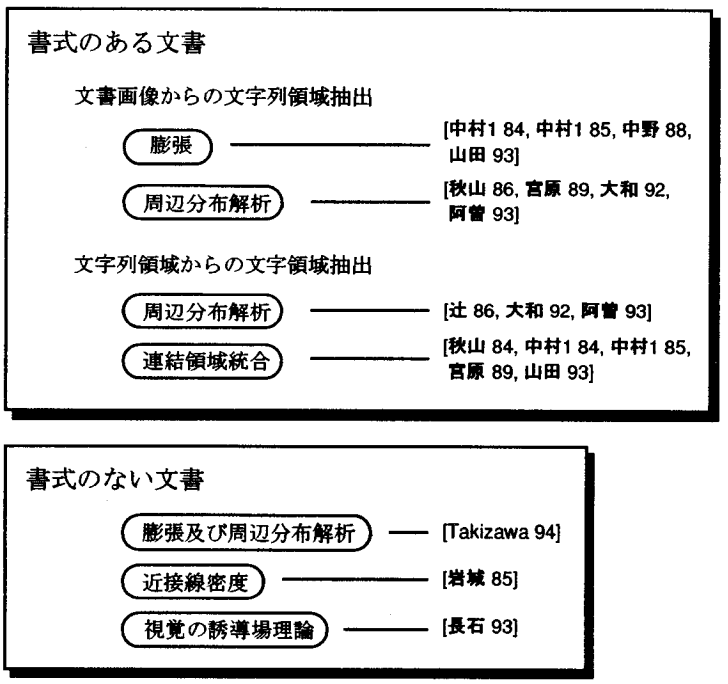


図 2.5 従来の文字領域抽出法

り, 整理された形で, 各種情報が記述されている.

このような文書から文字領域を抽出する場合, 書式に関する知識を積極的に用いることが有効であると考えられる. すなわち, 書式に関する知識を用いることにより, 文書中にどのような構成要素がどのように配置されているかを推定し, その推定結果に基づいて文字領域を探し出していくアプローチである.

日本語印刷文字により記述された書式のある文書の場合, 文字領域, 及び文字領域に関わる文字列領域, 文章領域について言及した書式に関する知識として, 以下のようなものを挙げることができる. これらの知識は, 日本語印刷文字に限らず, 中国の漢字や韓国のハングル語においても満足される.

1. 文字領域

    1-(a). 文字領域はほぼ正方形である.

2. 文字列領域

    2-(a). 文字列領域は方形である.

    2-(b). 文字列領域中の文字サイズは同一である.

    2-(c). 文字列領域中の文字間隔は一定である.

    2-(d). 文字列領域中の文字配置は直線的である.

    2-(e). 文字列領域中の隣接文字は極めて近接している.

3. 文書領域

    3-(a). 文章領域は方形である.

    3-(b). 文章領域中の文字列は等間隔かつ平行に配置されている.

    3-(c). 文章領域中の文字列は全て同一の文字サイズ, 文字間隔を有する.

これらの他にも, 各種文書の書式に応じた様々な知識が存在する. しかし, ここに挙げた知識は, 書式に関する知識の中でも基本的かつ本質的なものであり, これらを利用することにより書式のある文書から文字領域を抽出することは十分可能である. 概して, 文書中から文字領域を抽出する場合, 上記の知識を用いて文書画像中から文字列領域を抽出し, 次に, 文字列領域から文字領域を抽出することが多い. また, 文字列領域抽出後, 文字列領域の規則的な配置を基にして文章領域を明らかにすることが一般的である. 以下では, 文書画像中から文字領域を抽出する手順の過程である, 文書画像中からの文字列領域抽出, 及び, 文字列領域からの文字領域抽出について触れる.

文書画像中からの文字列領域抽出 文書画像中から文字列領域を抽出する場合, 膨張を用いた手法, あるいは周辺分布を用いた手法が用いられることが多い. 以下に, 両手法について概説する.

膨張を用いた手法は, 知識 2-(e)を考慮した手法である. すなわち, 膨張処理により文書画像中の全ての連結領域を拡大させ, 近接する文字間の間隔を埋めることにより生成される長細い連結領域を, 文字列領域として抽出するというアプローチである [田村 85]. 膨張を用いた手法は, 文字以外の文書構成要素と融合しない限り, 図表や写真などの文字列に関わりのない領域の影響を受けない. この意味で, 膨張を用いた文字列領域抽出法は, 書式を有するあらゆる文書に適用可能であると考えられる. この特性により, 膨張を用いた文字列領域抽出法は, 多くの文書画像解析の研究において用いられている [中村 1 84, 中村 1 85, 中野 88, 山田 93]. ただし, どの程度の膨張を施すかについては, 実験的検討により決定される場合が多い.

周辺分布を用いた手法は, 知識 2-(a), 3-(b), 3-(c)などを考慮した手法である. すなわち, 文書中の文字列配置の規則性に着目し, 図 2.6に示すような文書画像の周辺分布を解析することにより文字列領域を明らかにするアプローチをとる [田村 85]. 例えば図 2.6の場合, 文章領域中の文字列は横書きであるため, 垂直方向の周辺分布は, 文字列配置の規則性を反映した特徴的な分布を持つ. この分布を基にして, 文書画像中から文字列領域を容易に抽出することができる. 周辺分布を用いた手法は, その処理が極めて容易であることから, 多くの文書画像解析の研究において用いられている [秋山 86, 宮原 89, 大和 92, 阿曽 93]. しかし, 例えば新聞や論文誌などのように, 文書中に図表や写真など余計な領域が混在している場合, 上述の周辺分布の規則性が曖昧になってしまう. したがって, 周辺分布により文字列領域を抽出する場合, 与えられた文書中に文字や文字列のみしか存在しないという前提が与えられていることが条件となる.

文字列領域からの文字領域抽出 文字列領域から文字領域を抽出する場合, 周辺分布を用いた手法, あるいは連結領域の統合に基づく手法が用いられることが多い. 以下に, 両手法について概説する.

周辺分布を用いた手法は, 知識 2-(b), 2-(c), 2-(d)などを考慮した手法である. すなわち, 文字列領域中の文字配置の規則性に着目し, 文字列領域の周辺分布を解析することにより, 文字列領域を文字領域ごとに分離することができる. 周辺分布を用いた文字領域抽出は, 文字列領域抽出の場合と同様, 多くの文書画像解析の研究において用いられている [辻 86, 大和 92, 阿曽 93] . しかし, 文字列領域抽出の場合とは異なり, 文字領

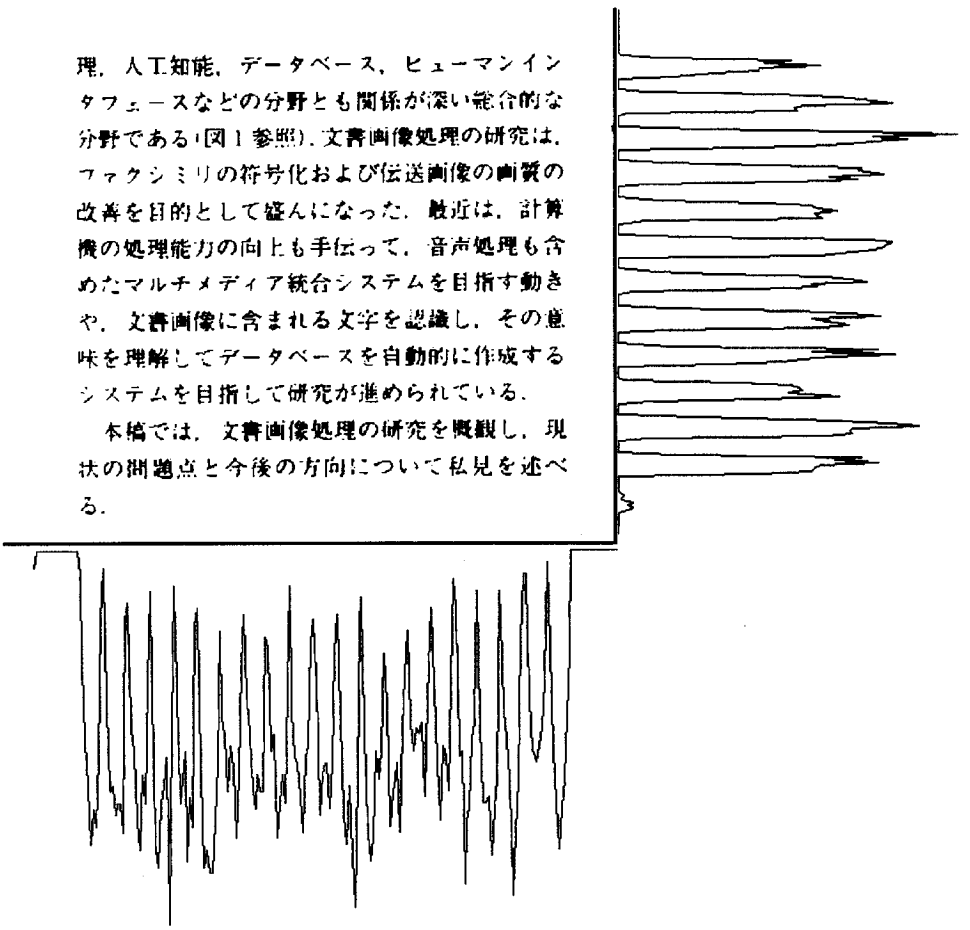


図 2.6 文書画像の周辺分布

域抽出のための周辺分布解析は容易ではない．例えば，“い” のように複数の連結領域から構成される文字を横軸に投影することにより得られる周辺分布は，文字配置の規則性を反映しない分布を示す．以上の理由から，周辺分布を用いた文字領域抽出では，周辺分布の複雑な解析を要する．

連結領域の統合に基づく手法は，知識 1-(a), 2-(b), 2-(c), 2-(d)などに基づく手法である．具体的には，まず一文字分の文字サイズを文字列領域の形状から推定し，その大きさに相当する正方形に収まるように連結領域をグループ化していくことにより，各文字領域を抽出するアプローチをとる．この種の手法も，文書画像解析の一要素技術として数多く利用されている [秋山 84, 中村 1 84, 中村 1 85, 宮原 89, 山田 93]．なお，連

結領域のグルーピング法は, 各文字領域抽出法により様々である.

以上の文字列領域抽出法, 文字領域抽出法は, 書式のある文書全てにおいて満たされる上述の知識のみを用いた, 一般的な手法である. 名刺や図書目録カードなど, 特殊な書式を有する文書を対象とした文書画像解析に関する研究では, その特殊な書式に関する知識を積極的に用い, さらに正確に文字領域を抽出する場合が多い [長谷 87, 黄瀬 89].

2.3.2 書式のない文書を対象とした手法

書式のない文書の例として, ポスターやチラシなどを挙げることができる. この種の文書中の各構成要素に関しては, デザイン優先の配置がなされており, 特定の書式は存在しない. したがって, 書式のある文書を対象とした場合とは異なり, 文字領域抽出の際に, 書式に関する先験的知識を利用することはできない. 以上の理由から, 書式のない文書を対象とした文字領域抽出法は, これまであまり提案されてこなかった. 以下では, 従来に提案された, 書式のない文書を対象とした文字領域抽出法を紹介する.

Takizawa らは, 書式のある文書に関する知識である 2-(a), 2-(b), 2-(c), 2-(d), 2-(e) などの知識に基づき, 近接する連結領域の統合により文字列領域を抽出し, 抽出された文字列領域から各文字領域を分離する手法を提案している [Takizawa 94]. この手法は, 先の膨張を用いた文字列領域抽出及び周辺分布を用いた文字領域抽出を統合し, 拡張した手法であり, 任意の方向に配置された文字列を有する文書画像に対しても適用可能となっている. ただし, 知識 2-(d), 2-(e)を満たさないような文字列, すなわち, 湾曲し, かつ隣接文字間の間隔が大きいような文字列を有する文書画像に適用することはできない.

岩城らは, 「文字は短線分の密集により構成されている」という知識のみを用いて, 文字領域を文書画像から直接抽出する手法を提案している [岩城 85]. この手法では, 着目する点の近傍の線密度を調べる特徴量である近接線密度を利用する. その値を文書画像全体に関して調べることにより, 文字と図形の分離を図っている. 岩城らの手法は, 用いている知識が単純なものであるため, 高い汎用性を有していると言える. しかし, 文字列領域及び文章領域を得ることはできない. また, パラメータ依存性が大きく, 適切なパラメータ値を選ばなければならないという欠点も併せ持っている.

長石は, 視覚の誘導場理論を利用することにより, 書式の定まっていない文書中の

手書き文字領域を抽出する手法を提案している [長石 93]. 視覚の誘導場理論とは, 文字間の距離が短くなると個々の文字を識別しにくくなる現象を, 文字の周りの「場」により説明したものである. この手法では, 手書き文字における視覚の誘導場の分布状態を分析することにより, 各々の文字領域を抽出する. 長石の手法は, 先の書式のある文書に関する知識を一切用いておらず, 極めて汎用性の高い手法であると言える. しかし, 先の岩城らの手法と同様に, 単に文字領域のみを抽出することが目標であり, 文字列領域及び文字領域を得ることはできない.

上述の手法の他にも, 書式のない文書からのアルファベットの抽出を目的とした手法が, いくつか提案されている [Fletcher 88, O'Gorman 93, Hönes 93]. しかし, これらの手法の全てが, 「一つの文字は一つの連結領域により構成されており, かつ, 文字列中の隣接する文字は極めて近接している」という知識に基づいている. 日本語印刷文字により記述された書式のない文書の場合, このような知識が満足されるとは限らない. したがって, これらの手法を日本語印刷文字の場合にそのまま応用することはできない.

任意の文書に適用可能な, 汎用性のある文字領域抽出を実現するためには, 文書の種類や書式の有無などに全く依存しないような知識のみを手がかりにして, 文字領域を抽出するというアプローチが必要となる [行天 93a]. これに対し, 従来の多くの文字領域抽出法において用いられてきた知識には, 書式を限定する何らかの要素が含まれていた. また, 手法の汎用性を目指した文字領域抽出法でも, 性能の面で, それぞれ問題点を抱えていた. 以上の点から, 真の意味で汎用性のある文字領域抽出法は, 未だに提案されていないと言っても過言ではない.

2.4 文字領域抽出のための一般的知識

真に汎用性のある文字領域抽出法を確立するためには, まず, 文字領域に関する, 文書の書式に全く依存しない知識を明らかにしなければならない. このような知識を, 以後, 一般的知識と呼ぶ. 以下では, 日本語印刷文字により記述された文書を対象とし, 図 2.7に示すような一般的知識について論じる.

一般に日本語印刷文字の形状は多種多様である. しかし, そのほとんどは, ある正方形内に短線分を密集させることによって描かれているという特徴を有する. これよ

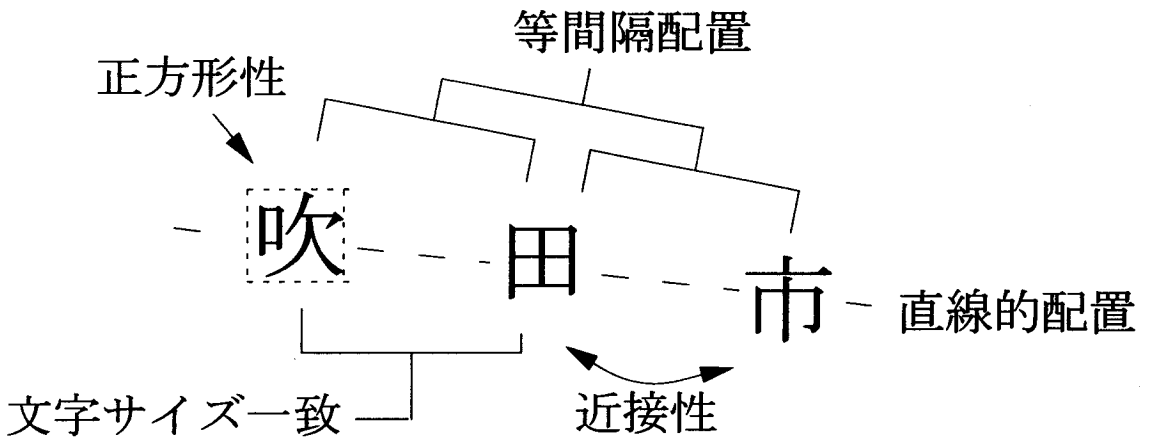


図 2.7 文字形状特徴及び文字配置特徴

り, 文字形状に関する一般的知識として, 以下のものを挙げることができる.

知識 1 (正方形性) 文字領域の形状, すなわち, 文字の外接矩形の形状はほぼ正方形である.

また, 日本語印刷文字により情報が記述されている文書中では, 複数個の文字集合により一つの文字列が形成され, 意味・内容を表す一単位となる. したがって, 文字配置に関する一般的知識を考えた場合, その一般的知識は文字列内の文字配置を基にして論ずることができる. 今, 文字列中の一文字に着目した場合, その文字の存在による周りへの影響は, せいぜい両隣の二文字に限定されると考えても不自然ではない. そこで, 文字列中の文字配置は, 隣接する二文字, あるいは三文字の文字配置に関する規則性により特徴付けられるものと考える. 以上の観点より, 文字列中の隣接する二文字に関する文字配置特徴について言及した一般的知識, また, 三文字に関する文字配置特徴について言及した一般的知識を, 以下に列挙する.

知識 2 (近接性) 文字列領域中の隣接文字は比較的近接している.

知識 3 (文字サイズ一致) 文字列領域中の隣接文字の文字サイズはほぼ同一である.

知識 4 (直線的配置) 文字列領域中に連続して配置されている三文字はほぼ直線的に並んでいる.

知識 5 (等間隔配置) 文字列領域中に連続して配置されている三文字はほぼ等間隔な二つの空白部を形成する.

以上に列挙した一般的知識は文字形状及び文字配置に関する日本語印刷文字の特徴について言及したものであり, 文書の種類や書式の有無に関わらず, ほとんど全ての文書において満たされる知識である. また, 日本語印刷文字に限らず, これらの知識は中国の漢字や韓国のハングル語などにおいても満足される. ただし, 文書中の文字が, これらの一般的知識を全て満足しているとは限らない. 例えば, 漢字の "一" は正方形性を全く満たしていない. また, 湾曲していたり文字間隔が大きい文字列中に存在する文字は, 直線的配置や近接性が成り立たない. このように, いくつかの一般的知識が満足されていないような文字は, 文書中に多数存在する.

2.5 結言

本章では, 文書画像解析における文字領域抽出問題, 及び, 文字領域抽出に関する従来手法のアプローチについて論じ, その議論の結果を基に, 任意の文書に適用可能な汎用的文字領域抽出法の設計方針を示した. 本章の議論は, 以下の二点に集約される.

- 文字領域抽出に関する多くの従来手法は, 対象文書に何らかの制限を与えることにより問題を簡略化したものがほとんどであり, 従来手法を単純に拡張するだけでは, 任意の文書に適用可能な汎用的文字領域抽出法を構築することは困難である.
- 汎用的文字領域抽出法では, 文書の書式の種類や有無などに全く依存しない一般的知識のみを用いて, 文字領域を抽出しなければならない.

そして, 書式に全く依存しない一般的知識として, 文字形状及び文字配置に関する一般的知識を具体的に列挙した. 次章以降では, これらの知識を利用した文字領域抽出法について議論する.

第 3 章

制約関数の最小化による文字領域抽出法

3.1 緒言

前章では, 文字領域抽出問題において任意の文書から文字領域を抽出するためには, 文書の種類や書式の有無に全く依存しない一般的知識のみを利用しなければならないと結論付けた. そして, 一般的知識として, 文字形状及び文字配置に関する一般的知識を具体的に示した.

では, これらの一般的知識を文字領域抽出の際に利用するためには, どのようなアプローチが有効であるかという点について考えてみる. 多くの画像処理手法においてある特定の領域を画像中から抽出する場合, まず所望の領域に関するいくつかの知識が予め与えられていることが前提となる. そして, これらの知識を逐次的に利用することにより, 所望の領域を段階的に絞り込んでいくというアプローチがとられる. このようなアプローチは, 与えられた全ての知識が所望の領域において完全に満たされていることが保証されている場合に有効である. しかし, 前章で列挙した一般的知識については, 全ての文字がこれらの知識を完全に満たしているとは限らないという曖昧性がある. この場合, 全ての一般的知識を完全に満たす領域を文書画像中から探し出すのではなく, 全ての一般的知識を総合的に考慮し, これらの知識を適度に満たすような領域を文字領域として抽出することが理想的である. このように, 複数の知識を総括的に扱う問題解決の一つのアプローチとして, 制約充足に基づいた処理がある. 具体的には, 全ての知識を所望の解に対する制約と捉え, 全ての制約を一つの制約関数に反映させることを考える. 制約関数については, その最小解が全ての制約を適度に満たす所望の解になるように構成する. その結果, 一つの制約関数を扱うことにより全ての知識を総

合的に考慮することができ, 制約関数の最小化により, 全ての知識を適度に満たすような所望の解を獲得することができる.

そこで本章では, 以上に述べた制約充足の概念に着目し, 制約関数の最小化による文字領域抽出法 CSCE (Constraint Satisfaction based Character Extractor) について述べる [行天 93b, Gyohten 93, Gyohten 95a]. CSCE では, 文字形状及び文字配置に関する一般的知識を制約として捉えた上で, 全ての制約を反映した制約関数を定義し, その最小化により日本語印刷文字の文字領域を抽出する. 制約関数を定義する際には, 制約関数が定義される空間について考えなければならない. また, 一般的知識を制約関数内で扱うためには, 文字形状及び文字配置に関する特徴を数量化した特徴量などを定義しなければならない. この観点より, まず制約関数が定義されるべき空間として, 文字領域抽出問題の問題空間について述べる. 次に, 文字形状特徴及び文字配置特徴の特徴量について論じる. 以上の問題空間, 特徴量について言及した上で文字領域抽出のための制約関数を定義し, その最小解を導く手続きについて論じる. 最後に, 書式のある文書及び書式のない文書を対象とした文字領域抽出実験により, CSCE の有効性を明らかにする.

3.2 画像認識／理解における制約関数最小化問題

制約関数の最小化による問題解決は, 最適化と呼ばれる分野の範疇に入るものである [Tagliarini 91, Rose 93]. この分野では, 対象とする問題に対する最適な解決策を求めるための定式化と手法を扱う. すなわち, 解決すべき問題を「与えられた拘束条件の下で, 評価関数を最小（または最大）にする解を見い出す」という最適化問題の形に記述し, 数学的に厳密なアルゴリズムを用いて解くというアプローチをとる [茨木 93]. 最適化問題と制約関数の最小化問題は本質的に同一のものであり, 最適化問題と制約関数の最小化問題を比較した場合, 最適化問題における評価関数が制約関数に相当する. 拘束条件の有無については, 各問題に依存する. ここでは問題の簡単のため, 拘束条件が与えられていない制約関数最小化問題について論ずるものとする.

問題解決手法として制約関数最小化のアプローチを導入する利点は, 所望の解に関する様々な知識を, 制約関数を介して総括的に扱うことができる点にある. 例えば, 解に関する様々な知識が与えられているが, その全ての知識が所望の解において満足され

ているとは限らないという問題を考える．そのような問題では，与えられた全ての知識が完全に満たされるような解を導出することは不可能であり，全ての知識がある程度総合的に満足されているような解を導くことが望ましい．そこで，制約関数の最小化による問題解決の考え方を導入し，所望の解に関する各種知識の総合的な満足度を制約関数により評価するものとする．制約関数を最小にする解は，与えられた全ての知識が満足されているような所望の解を表現するものとする．すると，ある問題が与えられた場合，その問題を基にして制約関数を導き，その最小解を求めることにより，全ての知識が適度に満たされるような解を得ることが可能になる．この種の問題解決は，近年，画像認識／理解の分野において数多く検討されており，画像からのエッジ抽出 [Tan 91]，領域分割問題 [Toborg 91]，画像中の物体の輪郭線抽出 [Kass 88] などの他，様々な方面に応用され，成果を挙げている．

画像認識／理解における諸問題を制約関数の最小化問題として考える場合，いかにして所与の問題を数学的に記述するか，制約関数をどのようにして定義するか，また，制約関数の最小解をどのようにして導くかという三つの点が問題となる．以下では，画像認識／理解における制約関数最小化問題において上述の問題点がどのように解決されているかについて論ずる．

画像認識／理解に関する様々な問題を数学的な問題として扱うためには，与えられた問題の解となりうる様々な候補解を，パラメータにより表現することができなければならない．複数のパラメータの様々な値を組み合わせることにより，これから導くべき解を全て網羅することができるような表現形式が必要となる．この種のパラメータ集合は，複数のパラメータの値域により張られる一種の空間を表現しているという意味から，問題空間と呼ばれる．問題空間中の一点，すなわち，各パラメータの値をそれぞれ決定することにより，与えられた問題に対するある一つの候補解を表現することができる．具体的な問題空間の構成方法は問題により様々である．一般に，与えられた画像からこれから導くべき解を記述することができる何らかのモデルを考え，そのモデルの状態変化を反映する様々なパラメータの集合を，問題空間と解釈することが多い．

通常，制約関数は，このようなモデル中のパラメータにより構成される問題空間上に定義される．制約関数は，所望の解に関する様々な知識の満足度を評価するものであり，その値の大小により，様々な候補解それぞれがどの程度所与の知識を満足しているかを評価することができる．ところが，制約関数を具体的にどのように定義しなければ

ならないかという点に関する一般的な方式はない. 制約関数の定義法は, 問題それぞれに応じて様々である. ただし, 制約関数は, 所望の解に関する知識を制約として反映した関数と, 解の妥当性に関する知識を制約として反映した関数とを組み合わせることにより構成されている場合が多い. これらの関数の呼び名については色々なものがあるが, 本論文では, 前者を目的関数, 後者をペナルティ関数と呼ぶことにする. 目的関数は, その最小解が, 与えられた知識を満足するような解に相当するように構成される. すなわち, 目的関数の最小値を与える解が, モデルを介して, 所望の知識を満たすような解を表現する. ペナルティ関数は, 問題空間中のパラメータが, 問題の目的に沿った妥当な解を表現している場合に, 最小値をとるように構成されている. ただし, 解の妥当性に関するこのような制約については, ペナルティ関数内に反映させる場合の他に, 問題空間を表現するモデル中に組み込んでしまう場合や, 制約関数の最小化アルゴリズム中に組み込んでしまう場合, あるいは, 最適化問題における拘束条件として解釈される場合もある. 制約関数は, このような目的関数とペナルティ関数の線形和により構成されることが一般的である.

最後に, 定義された制約関数をどのようにして最小化するかについて論じる. 通常, 画像認識／理解などを扱う制約関数は線形でない場合が多いため, 図 3.1に示すように, いくつかの「山」や「谷」を持つものと考えられる. このような非線形関数から最小値, すなわち大域的最適解を導くことは極めて困難である. そこで, 画像認識／理解における制約関数最小化問題では, 局所的最適解は大域的最適解に類似する解を表現するであろうという前提を置く. そして, 解として妥当な局所的最適解をいかに効率的に見つけるかを中心的な課題としている. 非線形関数の最小化手法としては, 最急降下法, ニュートン法, 準ニュートン法, 共役勾配法, シミュレーテッド・アニーリング法などが多く用いられる [茨木 93, 嘉納 87]. これらは, 拘束条件のない最適化問題における代表的な最小化手法である. これらの最小化手法のアルゴリズム中に, 問題依存の様々な知識を導入することにより, 効率的に局所的最小解を導くことが一般的である.

3.3 問題空間

本節では, 文字領域抽出問題を制約関数の最小化問題として捉えた場合に必要となる, 制約関数が定義される問題空間について議論する. まず, 文字領域抽出問題の問題

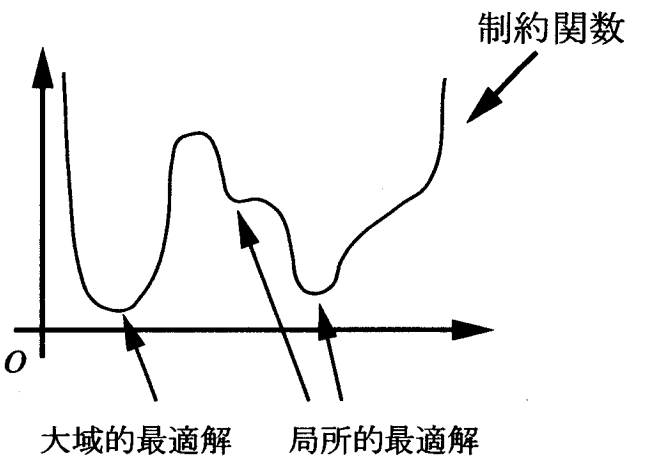


図 3.1 制約関数における大域的最適解と局所的最適解

空間を定義するために, 抽出すべき文字領域を記述するモデルについて論ずる. 二値により表現されている文書画像中の文字領域は, 画像中の連結領域を文字ごとにグルーピングしたものと等価であると考えることができる. この観点の下に, 画像中の連結領域と文字との所属関係を表現する手段として, 図 3.2に示すようなモデルを考えることができる. これを領域モデルと呼ぶことにする. CSCE では, 領域モデル中のパラメータにより構成される空間を, 問題空間としている. 領域モデルは, ノード群と, それらを結ぶリンクにより構成されている. C_jは下層の j番目のノードを表し, V_iは上層の i番目のノードを表す. また, $\mathcal{C}$は下層のノードの集合, $\mathcal{V}$は上層のノードの集合を表すものとする. ノード C_jは画像中の連結領域, ノード V_iは日本語印刷文字が存在していると思われる領域を表している. V_iにより表現される領域を, 以後, 文字候補領域と呼ぶ. 具体的に, 文字候補領域の形状が画像中においてどのように形成されるかについては後述する. また, 連結領域 C_j, 文字候補領域 V_iと記述した場合, それぞれのノードに対応する領域を表しているものとする. ノード V_iに対しては, 0から1の値を有するパラメータ p_{V_i}を与える. これは, 文字候補領域 V_iが本当に文字領域であるかに関する確からしさを表す.

領域モデル中の全てのノードがリンクにより密に結ばれている場合, リンク数の爆発の恐れがある. よって, ノード V_iは, 下層のノード $\mathcal{C}$のうち, 距離的に近いノードとのみリンクにより結ばれている. ノード V_iとリンクにより結ばれている $\mathcal{C}$内のノード

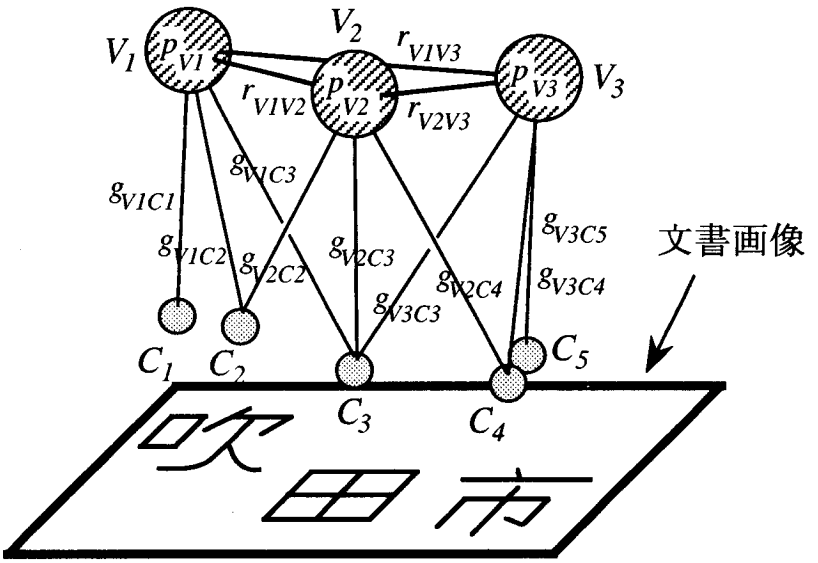


図 3.2 領域モデル

の集合を, $\mathcal{C}_{V_i}$ と記述する. パラメータ $g_{V_iC_j}$ は, 0 または 1 の値を有し, ノード V_i と C_j 間のリンクの強さを表す. 連結領域 C_j が文字候補領域 V_i に含まれている場合, $g_{V_iC_j}$ は 1 となり, そうでない場合は 0 となる. さらにノード V_i は, 上層のノード $\mathcal{V}$ のうち, 距離的に近いノードとのみリンクにより結ばれている. ノード V_i とリンクにより結ばれている $\mathcal{V}$ 内のノードの集合を $\mathcal{V}_{V_i}$ と記述する. ノード V_i, V_j $(i < j)$ 間のリンクは, 0 から 1 の値を有するパラメータ $r_{V_iV_j}$ を持つ. このパラメータは, 文字候補領域 V_i と V_j が文字列領域中で隣接する確からしさを表している. 領域モデル中では, 二つのノード間に複数のリンクが存在することはない. また, ある一つのノードに両端が継るようなリンクも存在しない.

パラメータ $p_{V_i}, g_{V_iC_j}, r_{V_iV_j}$ の集合は, 領域モデルの状態を表現する. すなわち, これらのパラメータ値が, 文字領域の形状や文字列領域中での隣接関係を領域モデル内に記述する. そこで, これらのパラメータを, 問題空間を形成する変数と解釈する. これにより得られる問題空間は, 以下のように定義することができる.

$$P = \{p_{V_i} | V_i \in \mathcal{V}\}$$

$$G = \{g_{V_iC_j} | V_i \in \mathcal{V}, C_j \in \mathcal{C}_{V_i}\}$$

$$R = \{r_{V_iV_j} | V_i \in \mathcal{V}, V_j \in \mathcal{V}_{V_i}, i < j\}$$

さらに, 領域モデル中の各ノードが, どのようにして連結領域や文字候補領域を表現しているかについて述べる. 下層のノード C_jは, 連結領域に外接する二つの矩形の座標情報を有している. これらの矩形は, 図 3.3に示すように, 一方が xy軸方向に, もう一方は xy軸を 45 度傾けた uv軸方向に沿っている. ノード C_jは, 各矩形の四隅の座標値を有しており, 合計八点の座標値により, 連結領域の概略的な形状を表現している.

ノード V_iは, 日本語印刷文字が存在するであろう文字候補領域を表現している. パラメータ $g_{V_iC_j}$の定義より, $g_{V_iC_j}$=1 であるならば, 文字候補領域 V_iは, 連結領域 C_jを含んでいなければならない. そこで, 文字候補領域は, 図 3.4に示す二つの矩形の重複部分により与えられる. 一方の矩形は, $g_{V_iC_j}$=1 でノード V_iと継る $\mathcal{C}$中の全てのノードが表現する連結領域を囲む, xy軸方向に沿った最小の外接矩形である. また, もう一方の矩形は, uv軸方向に沿った同様の矩形である.

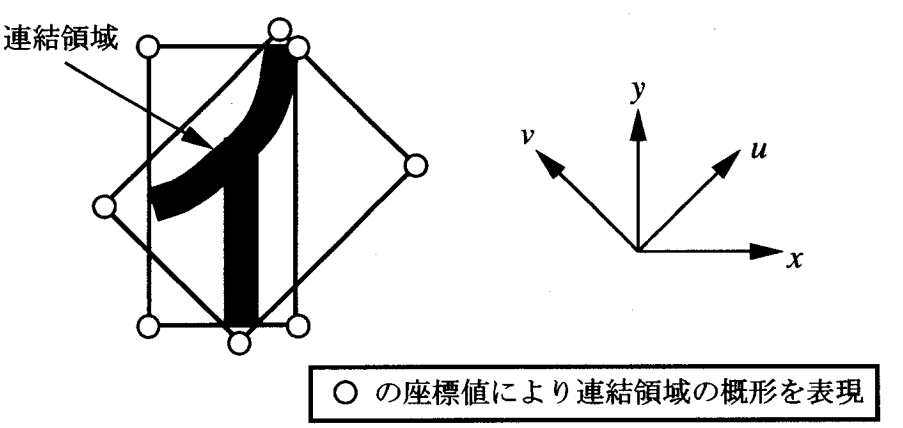


図 3.3 連結領域の表現

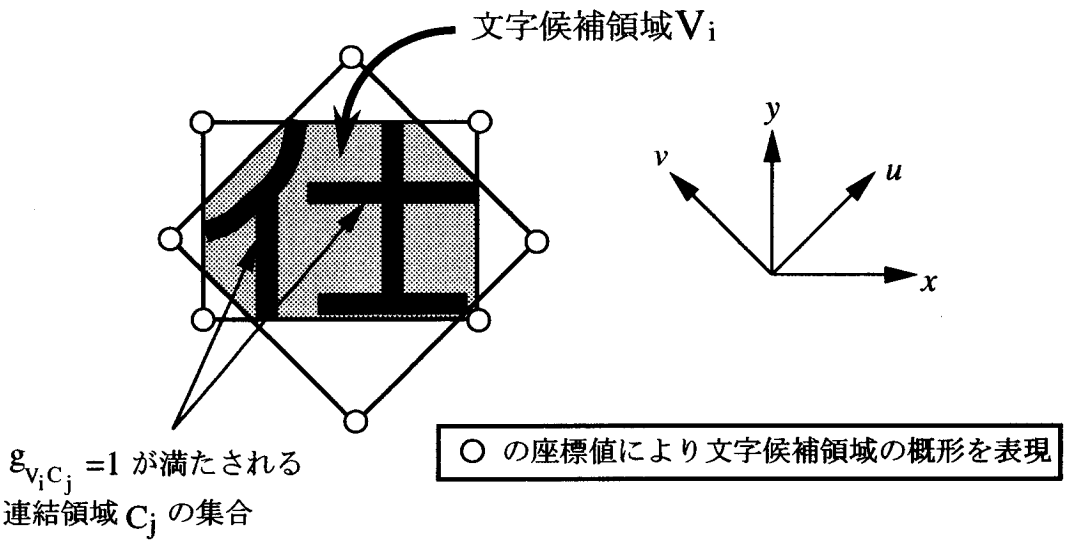


図 3.4 文字候補領域の表現

3.4 文字形状特徴及び文字配置特徴の特徴量

文字領域抽出問題では, 前章で挙げた文字形状及び文字配置に関する一般的知識の満足度を制約関数により評価することが要求される. この場合, 一般的知識が言及している文字形状及び文字配置に関する特徴がどの程度満たされているかを数量的に表現し, その数量化の結果を制約関数に反映することが必要となる. そこで本節では, 文字形状及び文字配置に関する特徴の満足度を数量的に表現した尺度, すなわち, 特徴量について論ずる. これらの特徴は文字候補領域が満たすべきものである. したがって, 特徴量は, 領域モデル中のノード集合 $\mathcal{V}$中のノードに対して定義される.

特徴量を定義する前に, 文字候補領域のサイズについて考える. 文字候補領域 V_i のサイズは, 以下のように定義される.

$$s_{V_i}(G) = \min\{s^{xy}_{V_i}(G),\ s^{uv}_{V_i}(G)\} \tag{3.1}$$

ここで, $s^{xy}_{V_i}(G)$ と $s^{uv}_{V_i}(G)$ はそれぞれ、ノード V_i の文字候補領域の形状を決定する二つの矩形の面積を表す. $s^{xy}_{V_i}(G)$ と $s^{uv}_{V_i}(G)$ はそれぞれ, xy軸方向に沿った矩形の面積と uv 軸方向に沿った矩形の面積に対応する.

以下では, 各特徴の特徴量を定義する. 前章で触れたように, 日本語印刷文字の形状の特殊性を表す特徴として, 正方形性を考えることができる. この特徴の特徴量は, ノード集合 $\mathcal{V}$中の各ノードに対して定義される. ノード V_iの正方形性の特徴量 $q_{V_i}(G)$ は, 以下のように定義される.

$$q_{V_i}(G) = \begin{cases} q^{xy}_{V_i}(G) & s^{xy}_{V_i}(G) \leq s^{uv}_{V_i}(G) \\ q^{uv}_{V_i}(G) & \text{otherwise} \end{cases} \qquad (0 \leq q_{V_i}(G) \leq 1) \tag{3.2}$$

ここで, $q^{xy}_{V_i}(G)$ と $q^{uv}_{V_i}(G)$ はそれぞれ, xy軸方向と uv軸方向に沿った, 文字候補領域の形状を定める矩形に対応し, それぞれの矩形の長い方の辺に対する短い方の辺の割合を表している. 正方形性の特徴量 $q_{V_i}(G)$ は, 二つの矩形のうちの小さい方が正方形に近いほど, 大きな値をとる.

文字配置の規則性に着目した特徴として, 近接性, 文字サイズの一致, 直線的配置, 等間隔配置を挙げることができる. このうち, 近接性と文字サイズの一致に関しては, 二つの文字間の関係を言及した特徴である. したがって, これらの特徴の特徴量は, 二つのノード V_i, V_jについて定義される. 同様の理由から, 直線的配置と等間隔配置の特徴量は, 三つのノード V_i, V_j, V_kについて定義される.

ノード V_i, V_jに関する近接性の特徴量 $c_{V_iV_j}(G)$ は, 以下のように定義される.

$$c_{V_iV_j}(G) = \frac{s_{V_i}(G) + s_{V_j}(G)}{d_{V_iV_j}(G)} \qquad (c_{V_iV_j}(G) \geq 0) \tag{3.3}$$

ここで, $d_{V_iV_j}(G)$ は, 二つの文字候補領域 V_i, V_j間の中心間のユークリッド距離を表す. 文字候補領域 V_i, V_j間の距離 $d_{V_iV_j}(G)$ が小さくなった場合, すなわち, これらの文字候補領域が近接性を満たすほど, 近接性の特徴量 $c_{V_iV_j}(G)$ の値は大きくなる.

ノード V_i, V_jに関する文字サイズ一致の特徴量 $m_{V_iV_j}(G)$ は, 以下のように定義される.

$$m_{V_iV_j}(G) = \mathrm{rat}\{s_{V_i}(G),\ s_{V_j}(G)\} \qquad (0 \leq m_{V_iV_j}(G) \leq 1) \tag{3.4}$$

ここで, $\mathrm{rat}\{\cdot,\cdot\}$ は, 二つの要素のうち, 大きい方の要素に対する小さい方の要素の割合を表す. 文字候補領域 V_i, V_jのサイズがほぼ等しい場合, 文字サイズの一致に関する特徴量 $m_{V_iV_j}(G)$ の値は大きくなる.

ノード V_i, V_j, V_kに関する直線的配置の特徴量 $l_{V_iV_jV_k}(G)$ は, 以下のように定義される.

$$l_{V_iV_jV_k}(G) = \frac{1}{\pi}\left|[\theta_{V_iV_jV_k}(G) + \pi]_{2\pi}\right| \qquad (0 \leq l_{V_iV_jV_k}(G) \leq 1) \tag{3.5}$$

ここで, $[\cdot]_{2\pi}$は角度を$-\pi$からπに限定する（例えば, $[-11\pi/4]_{2\pi} = -3\pi/4$）. $\theta_{V_iV_jV_k}(G)$ は, 文字候補領域 V_i, V_j, V_kをこの順で結ぶ二つのベクトルが成す角, すなわち曲率を表す. これらの文字候補領域が直線的に配置されていた場合, 直線的配置に関する特徴量 $l_{V_iV_jV_k}(G)$ の値は大きくなる.

ノード V_i, V_j, V_kに関する等間隔配置の特徴量 $e_{V_iV_jV_k}(G)$ は, 以下のように定義される.

$$e_{V_iV_jV_k}(G) = \mathrm{rat}\{d_{V_iV_j}(G),\ d_{V_jV_k}(G)\} \qquad (0 \leq e_{V_iV_jV_k}(G) \leq 1) \tag{3.6}$$

文字候補領域 V_i, V_j, V_kが等間隔に配置されている場合, 等間隔配置に関する特徴量 $e_{V_iV_jV_k}(G)$ の値は大きくなる.

ただし, 特徴量 $c_{V_iV_j}(G)$, $m_{V_iV_j}(G)$ については, ノード V_i, V_j間にはリンクが存在していなければならない. すなわち, $V_i \in \mathcal{V}_{V_j}$かつ $V_j \in \mathcal{V}_{V_i}$でなければならない. 同様に, 特徴量 $l_{V_iV_jV_k}(G)$, $e_{V_iV_jV_k}(G)$ については, ノード V_i, V_j間, 及び, ノード V_j, V_k間にリンクが存在していなければならない.

3.5 制約関数

以下では, 文字領域抽出のための制約関数について述べる. 前述のように, 制約関数には所望の解に関する知識と解の妥当性に関する知識が制約として与えられる. 所望の解に関する知識は目的関数内で考慮され, 解の妥当性に関する知識はペナルティ関数内で考慮されることが一般的である. 文字領域抽出問題の場合, 所望の解に関する知識は, 前章で挙げた文字形状及び文字配置に関する一般的知識である. これらの一般的知識がどの程度満足されているかについては, 前節で定義した各種特徴量を用いることにより評価することができる. よって, ここで用いる制約関数内の目的関数は, 文字形状特徴及び文字配置特徴の特徴量により一般的知識の充足度を評価する. 一方, 解の妥当性に関する知識として, 文書中の文字領域に関する以下の四つの知識を考える. これらの知識が満たされている文字領域は, 文字として妥当であると解釈することができる.

(i). 一つの連結領域は複数の文字領域に属さない.
(ii). 一つの文字領域は有限個の連結領域により構成される.
(iii). 文字列領域は分岐しない.
(iv). 文字列領域は文字領域により形成される.

以上の四つの知識に加え, 領域モデル中のパラメータ値に関する知識を解の妥当性に関する知識と考え, ペナルティ関数内で評価するものとする. 以下に, 目的関数とペナルティ関数を定義する.

目的関数の定義では, 文字形状及び文字配置に関する一般的知識を, 制約としてどのように目的関数内に反映すれば良いかが問題となる. ここでは, 文字形状及び文字配置に関する特徴の特徴量と, 問題空間中の変数との関係に着目した. 文字候補領域 V_iが正方形性を満たしている場合, この文字候補領域は文字らしいと解釈することができる. したがって, 特徴量 $q_{V_i}(G)$ が十分大きな値を有する場合, 変数 $p_{V_i} = 1$ とならなければならない. さらに, 複数の文字候補領域が, 文字配置に関する特徴を満たす形で配置されている場合, 各々の文字候補領域は, 文字列領域中の各文字をぞれぞれ表現しているものと考えられる. したがって, 特徴量 $c_{V_iV_j}(G)$, $m_{V_iV_j}(G)$ が十分大きな値を有する場合は $r_{V_iV_j} = 1$, 特徴量 $l_{V_iV_jV_k}(G)$, $e_{V_iV_jV_k}(G)$ が十分大きな値を有する場合は $r_{V_iV_j}r_{V_jV_k} = 1$ でなければならない. 以上より, 以下のような目的関数を考えることができる.

$$F_{1(P,G)} = \sum_{V_i \in \mathcal{V}} E(p_{V_i}, J_q(q_{V_i}(G))) \tag{3.7}$$

$$F_{2(G,R)} = \sum_{V_i \in \mathcal{V}} \sum_{\substack{V_j \in \mathcal{V}_{V_i} \\ i<j}} E(r_{V_iV_j}, J_c(c_{V_iV_j}(G)) J_m(m_{V_iV_j}(G))) \tag{3.8}$$

$$F_{3(G,R)} = \sum_{V_j \in \mathcal{V}} \sum_{V_i \in \mathcal{V}_{V_j}} \sum_{\substack{V_k \in \mathcal{V}_{V_j} \\ i<k}} E(r_{V_iV_j} r_{V_jV_k}, J_l(l_{V_iV_jV_k}(G)) J_e(e_{V_iV_jV_k}(G))) \tag{3.9}$$

$J_x(x \in \{q, c, m, l, e\})$ は各特徴量のしきい関数であり, 以下のように定義される.

$$J_x(z) = \begin{cases} 0 & z < t_x^{min} \\ \frac{z - t_x^{min}}{2(t_x^{mid} - t_x^{min})} & t_x^{min} \leq z < t_x^{mid} \\ \frac{z - t_x^{mid}}{2(t_x^{max} - t_x^{mid})} + \frac{1}{2} & t_x^{mid} \leq z < t_x^{max} \\ 1 & z \geq t_x^{max} \end{cases} \tag{3.10}$$

ここで, t_x^{min} と t_x^{max} は, 各特徴量の上限と下限を表す. t_x^{mid} は, 各特徴のしきい値を表す. 関数 E は, 以下のように定義される.

$$E(a,b) = (-b+1)a + \frac{1}{2}(1-a) \tag{3.11}$$

上述の三つの目的関数は, 変数と特徴量との関係を明らかにするための関数である. 各々の関数は, 変数の値と特徴量の値の関係が適切である場合に最小となる. また, これらの目的関数は, 各々の特徴が強制的に満たされた場合, すなわち, 各々の特徴量の値が大きくなるような状態変化が起こった場合も, より小さな値をとる. これにより, 目的関数の最小化により, 文字候補領域が特徴を満たすように変形され, 最終的に所望の文字領域が得られることになる.

ペナルティ関数は, 空間 P, G, R 中の変数値が, 文字としてもっともらしい文字領域抽出結果を表現している場合にのみ, 0 となる. ペナルティ関数は, 文字として妥当な文字領域抽出結果を獲得するためには不可欠な関数であり, 目的関数と同様, 制約関数内に組み込まれている.

領域モデル内では, 空間 P, G, R 中の各変数は 0 から 1 までの間の値を有する. これは, 領域モデル内で絶対に満足されていなければならない. これを解の妥当性に関する知識と捉え, 以下のように定義されるペナルティ関数により制約関数に反映する.

$$F_{d(P,G,R)} = \sum_{V_i \in \mathcal{V}} \sum_{C_j \in \mathcal{C}_{V_i}} H(g_{V_i C_j}) + \sum_{V_i \in \mathcal{V}} H(p_{V_i}) + \sum_{V_i \in \mathcal{V}} \sum_{\substack{V_j \in \mathcal{V}_{V_i} \\ i<j}} H(r_{V_i V_j}) \tag{3.12}$$

ただし, 関数 H は以下のように定義される.

$$H(z) = \begin{cases} 0 & 0 \le z \le 1 \\ M & \text{otherwise} \end{cases} \tag{3.13}$$

ここで, M は十分大きな値である. 関数 F_d は, 問題空間 P, G, R 内の全ての変数値が 0 から 1 までの範囲内にある場合のみ, 0 となる.

文字候補領域と連結領域間の関係については, 先に挙げた知識のうち, (i)と (ii)を考えなければならない. すなわち, 文書中の連結領域は, 二つ以上の文字候補領域に属するべきではない. また, 文字候補領域を構成する連結領域の数は, 有限であるべきことである. この二点より, 以下のようなペナルティ関数を考えることができる.

$$F_{g1(G)} = \sum_{C_j \in \mathcal{C}} U(\sum_{V_i \in \mathcal{V}_{C_j}} g_{V_i C_j}, 1) \tag{3.14}$$

$$F_{g2(G)} = \sum_{V_i \in \mathcal{V}} U(\sum_{C_j \in \mathcal{C}_{V_i}} g_{V_i C_j}, A) \tag{3.15}$$

ただし, 関数 Uは以下のように定義される.

$$U(z,n) = \begin{cases} z^2 & z < 0 \\ 0 & 0 \leq z \leq n \\ (z-n)^2 & z > n \end{cases} \tag{3.16}$$

A は, 文字を構成する連結領域数の上限を表す. 集合 $\mathcal{V}_{C_j}$は, 以下のように定義される.

$$\mathcal{V}_{C_j} = \{V_i | V_i \in \mathcal{V}, C_j \in \mathcal{C}_{V_i}\}$$

関数 F_{g1}は, 文字候補領域と連結領域間の制約のうち, 前者の制約が満たされている場合, すなわち, $\sum_{V_i \in \mathcal{V}_{C_j}} g_{V_i C_j}$が 0 もしくは 1 である場合に, 0 となる. 関数 F_{g2}は, 後者の制約, すなわち, $\sum_{C_j \in \mathcal{C}_{V_i}} g_{V_i C_j}$の値が有限である場合に, 0 となる.

文字列に関する制約としては, 先述の知識のうち (iii)と (iv)を考えなければならない. すなわち, 文字列には分岐がないため, ある文字領域が文字列領域中で三つ以上の文字と隣接することはない. また, 文字列領域を成している文字候補領域は, 本当の文字領域であると考えることができるという知識である. この二点より, 以下のようなペナルティ関数を考えることができる.

$$F_{r(R)} = \sum_{V_i \in \mathcal{V}} U(\sum_{V_j \in \mathcal{V}_{V_i}} r_{V_i V_j}, 2) \tag{3.17}$$

$$F_{p(P,R)} = \sum_{V_i \in \mathcal{V}} \sum_{\substack{V_j \in \mathcal{V}_{V_i} \\ i<j}} \{r_{V_i V_j}(1 - p_{V_i} p_{V_j})\} \tag{3.18}$$

関数 F_rは, 前者の制約が満たされている場合に 0 となる. この制約は, 全てのノード V_iにおいて, 不等式 $0 \leq \sum_{V_j \in \mathcal{V}_{V_i}} r_{V_i V_j} \leq 2$ が満たされていなければならないことを表す. 関数 F_pは, 後者の制約が満たされている場合に 0 となる. この制約は, 変数 $r_{V_i V_j}$が 1 である場合, 変数 p_{V_i}と p_{V_j}が 1 でなければならないことを意味する. この制約は,

$r_{V_iV_j}(1-p_{V_i}p_{V_j})=0$ により表される.

以上に定義した三つの目的関数と五つのペナルティ関数の線形和により, 以下に示すような制約関数を定義する.

$$\begin{aligned} F_{const(P,G,R)} = \quad & S_{obj}(S_1F_{1(P,G)}+S_2F_{2(G,R)}+S_3F_{3(G,R)}) \\ & +S_{pen}(S_dF_{d(P,G,R)}+S_{g1}F_{g1(G)}+S_{g2}F_{g2(G)}+S_rF_{r(R)}+S_pF_{p(P,R)}) \end{aligned} \tag{3.19}$$

ここで, S_{obj}, S_{pen}, S_1, S_2, S_3, S_d, S_{g1}, S_{g2}, S_r, S_pは, 各目的関数, ペナルティ関数の係数である.

3.6 文字領域抽出手続き

文字領域抽出手続きでは, 制約関数の値が小さくなるように, 領域モデル内の変数値を調整する. その結果, 最終の平衡状態において, 所望の文字領域を表現する最適の変数値を得ることができる. 制約関数の最小化については, 3.2節で触れたように様々な最小化手法を想定することができる. しかし, 非線形関数を対象とした最小化手法の大半では, その非線形関数を偏微分することが必要となる. ここで用いている制約関数は複雑な構造を有しているため, その偏微分を導くことは困難である. そこでここでは, 関数の偏微分などを全く必要としない単純なアルゴリズムにより構成されるシミュレーテッドアニーリング法 [Kirkpatrick 83] を基本として, 文字領域抽出手続きを構成している. ただし, CSCE では, 文字を効率的に抽出するため, 本来のシミュレーテッドアニーリング法を若干修正している. 制約関数の最小化の中途段階では, ノード集合 $\mathcal{V}$ 内に, いくつかの意味のない, もしくは冗長なノードが形成される. ここで言う意味のないノードとは, ノード集合 $\mathcal{C}$ 中のノードとのリンクを持たないノードのことを表す. また, 冗長なノードとは, お互いに重複する文字候補領域を表現するノードのことを表す. 本アルゴリズムには, 関数の最小化の過程でこれらのノードを除去する処理が加えられている. また, 領域モデルの状態変化に応じて, 領域モデル中のリンクは周期的に張り替えられる.

文字領域抽出手順は, 以下のように与えられる. ただし, Tは文字領域抽出過程で用いられる変数, T_0, μ, Dはそれぞれシステムのパラメータを表す. また, $\{P_\tau\}$, $\{G_\tau\}$,

$\{R_\tau\}$ は各空間上の状態変化列であり, $[u]_+ = \max\{0, u\}$ である.

1. 領域モデル中の下層と上層のノード集合 $\mathcal{C} = \{C_j | j = 1, \ldots, M\}$ $\mathcal{V} = \{V_i | i = 1, \ldots, M\}$ を生成する. 両集合中の各ノードは, 文書画像中の連結領域を表す. M は, 連結領域の数である.

2. ノード集合 $\mathcal{V}$ 中の各ノード V_i を, ノード集合 $\mathcal{C}_{V_i}$ 中の全てのノード, また, ノード集合 $\mathcal{V}_{V_i}$ 中の全てノードとリンクにより結ぶ. また, 領域モデル中のパラメータの初期値を, $p_{V_i} = J_q(q_{V_i}(G_0))$, $g_{V_iC_j} = \delta_{ij}$, $r_{V_iV_j} = J_c(c_{V_iV_j}(G_0))J_m(m_{V_iV_j}(G_0))$ とする. ただし, δ_{ij} はクローネッカーのデルタである.

3. T の初期値を T_0 とする. またカウンター値をそれぞれ $\eta = 0$, $\tau = 0$ とする.

4. 以下の 4a から 4d までの処理を, $(P_\tau, G_\tau, R_\tau) = (P_{\tau-\mu}, G_{\tau-\mu}, R_{\tau-\mu})$ となるまで繰り返す.

   (a) 以下の処理を μ 回繰り返す.

      i. 現在の状態での $F_{const(P_\tau, G_\tau, R_\tau)}$ の値を算出する.

      ii. 状態空間から任意に選んだ変数の値を変化させることにより仮の状態を生成し, 仮の状態での $F_{const(P', G', R')}$ の値を算出する.

      iii. $\exp(-[F_{const(P', G', R')} - F_{const(P_\tau, G_\tau, R_\tau)}]_+ / T)$ の確率で $(P_{\tau+1}, G_{\tau+1}, R_{\tau+1}) = (P', G', R')$ とし, それ以外の場合は $(P_{\tau+1}, G_{\tau+1}, R_{\tau+1}) = (P_\tau, G_\tau, R_\tau)$ とする.

      iv. $\tau \leftarrow \tau + 1$

      v. 以下の二つの処理により, $\mathcal{V}$ 内の冗長なノードを除去する.

         A. 重複する文字候補領域を表現する二つのノードが存在する場合, 一方を除去する.

         B. 長期に渡って $\sum_{C_j \in \mathcal{C}_{V_i}} g_{V_iC_j} = 0$ である場合, ノード V_i を除去する.

   (b) $T \leftarrow DT$ により T の値を下げる.

   (c) $\eta \bmod 10 = 0$ である場合, 以下の処理により領域モデル中に新たなリンクを生成する.

i. 各ノード V_iについて, $\mathcal{V}_{V_i}$, $\mathcal{C}_{V_i}$を再生成する.

ii. 各ノード V_iについて, 新たに $\mathcal{V}_{V_i}$, $\mathcal{C}_{V_i}$に加えられたノードと V_i間にリンクを張り, リンク上の変数に初期値を与える.

(d) $\eta \leftarrow \eta + 1$

3.7 実験結果

CSCE を, C 言語を用いて, SUN SPARC station 10 Model 30 上に構築した. CSCE 内で用いられているパラメータの値は, 表 3.1の通りである. これらの値は, 実験的に得た値であり, 全ての実験において共通して用いたものである.

表 3.1 CSCE のパラメータ値

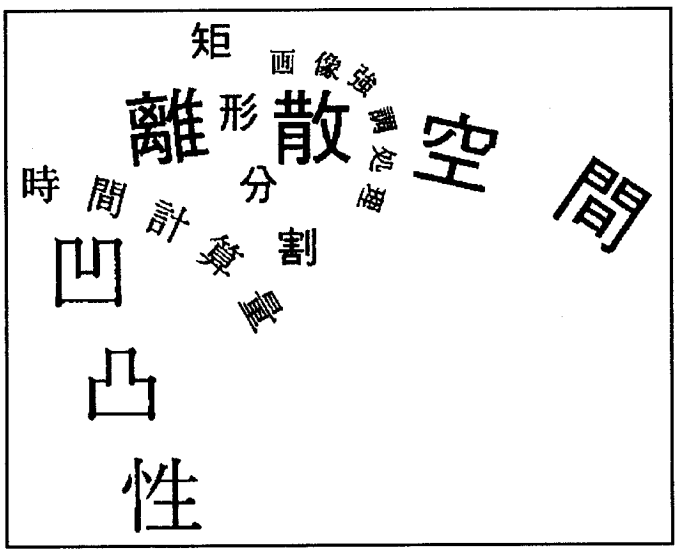
式 (3.7)–(3.9) 中のしきい値
$t_q^{min} = 0.0 \quad t_q^{mid} = 0.5 \quad t_q^{max} = 1.0$
$t_c^{min} = 0.0 \quad t_c^{mid} = 0.7 \quad t_c^{max} = 2.0$
$t_m^{min} = 0.0 \quad t_m^{mid} = 0.5 \quad t_m^{max} = 1.0$
$t_l^{min} = 0.0 \quad t_l^{mid} = 0.7 \quad t_l^{max} = 1.0$
$t_e^{min} = 0.0 \quad t_e^{mid} = 0.7 \quad t_e^{max} = 1.0$
式 (3.15) 中のパラメータ値
$A = 10$
式 (3.20) の係数値
$S_{obj} = S_{pen} = 1/2 \quad S_1 = S_2 = S_3 = 1/3$
$S_d = S_{g1} = S_{g2} = S_r = S_p = 1/5$
文字領域抽出手続き中のパラメータ値
T_0 = 初期状態から一変数値を変化させることによる制約関数の平均変化量の 0.015 倍
μ = 領域モデル中のパラメータ数の 10 倍
D = 0.9

表 3.2 書式なし文書の仕様

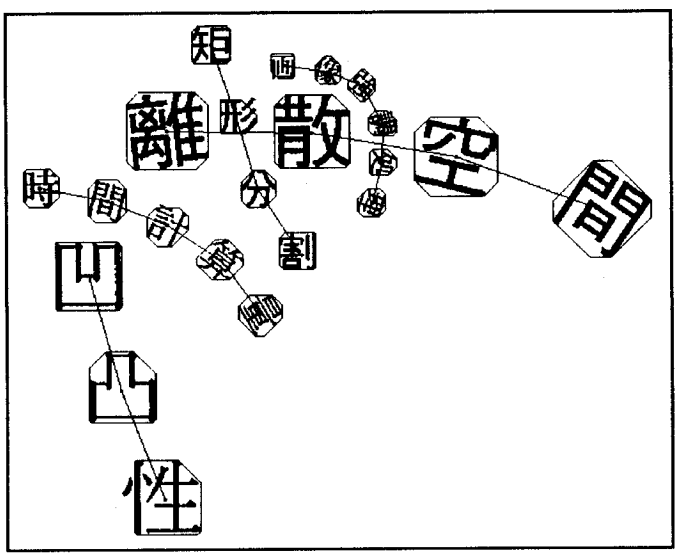
画像番号	1	2	3	4	5	6	7	8	9	10
日本語印刷文字数	21	26	22	26	20	29	32	28	27	26
文字列数	4	7	5	7	6	6	7	6	7	6
湾曲している文字列の数	4	7	4	2	1	2	5	4	2	3
直線的な文字列の数	0	0	1	5	5	4	2	2	5	3
二文字列の交差数	1	0	2	2	0	0	3	2	2	1

本実験では, 70 枚の書式あり文書と 10 枚の書式なし文書を, 実験対象として用いた. 書式あり文書は, 文書画像データベース JEIDA'93 から得た様々な文書画像の部分画像である. 実験に用いた文書の種類は, 小説, 教科書, 学術書, 電話帳, 名簿, マニュアル, ワープロ文例, 特許, 実用書など, 様々な書式を有する書式あり文書である. 文書画像中には, 汚れやかすれがほとんど含まれていない. また, 文字列は, 全て直線的に文字が配置されており, 垂直もしくは水平の文字列方向を有している. ただし, 異なる文字列方向を有する文字列は混在していない. 実験で用いた書式のある文書画像 70 文書のうち, 46 文書が水平方向の文字列を有し, 24 文書が垂直方向の文字列を有している. また, 8 文書には, 異なる文字サイズを有する文字列が混在している. 書式なし文書は自作であり, 互いに交差するような文字列も含んでいる. 書式なし文書の仕様は, 表 3.2 の通りである. 書式なし文書中の各文字列は, 一様な文字サイズ, 文字間隔を有している. ただし, 各文字列の文字サイズ, 文字列方向, 曲率は, 任意に決定したものである.

図 3.5(a) は, 実験において用いた書式なし文書画像の一例である. 表 3.2中では, 3 番の文書に相当する. 図 3.5(b) は, CSCE 適用後の文字領域抽出結果である. 図中の多角形は, 最終的に抽出された文字領域である. すなわち, 領域モデルの最終的な平衡状態において, ノード集合 $\mathcal{V}$中のノードのうち $p_{V_i} \geq 0.5$ であるノードが表している文字候補領域である. 文字候補領域の中心間を結ぶ直線は, 領域モデル中において $r_{V_iV_j} \geq 0.5$ となるリンクを図示したものである. 図 3.5(b) の結果では, 日本語印刷文字上に, 正しく文字候補領域が構成されていることが判る. また, 文字列領域中の文字候補領域のうち, 互いに隣接するものは, リンクにより正しく結ばれていることが判る.



(a) 原画像



(b) 文字領域抽出結果

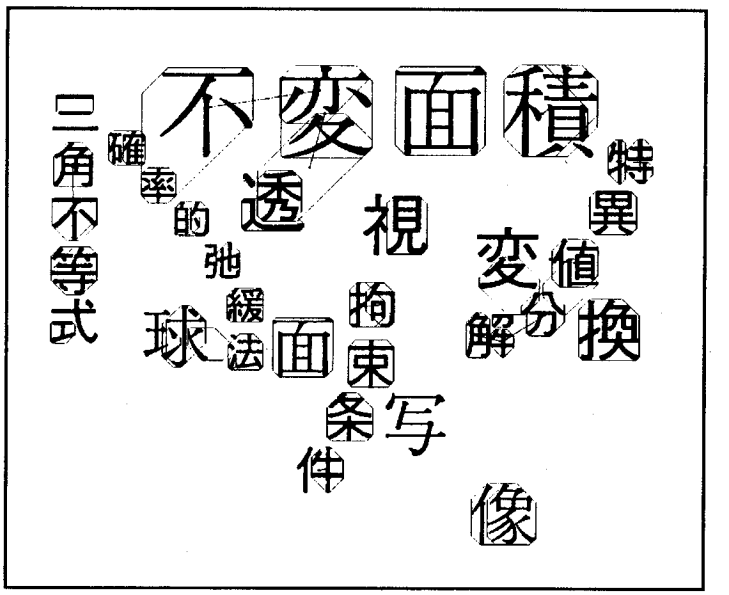
図 3.5 CSCE による実験結果

これは, CSCE により, 文字領域はもちろんのこと文字列領域も正しく抽出することができることを意味する. また, CSCE は, 文書の書式などに関する知識を一切用いていない. そのため, 書式のない文書に対しても適用可能であることが, 図 3.5(b) に示す結果より判る.

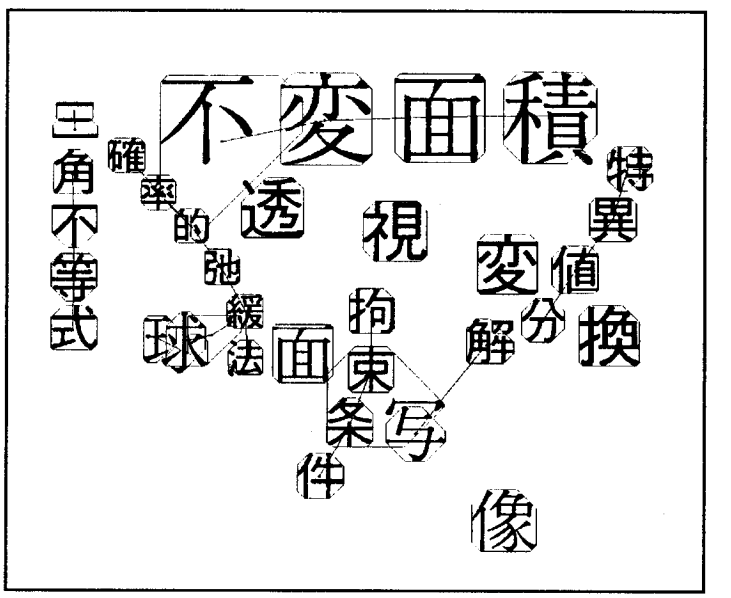
図 3.6は, 文字領域抽出過程における文字候補領域の変化を表している. 実験対象は, 表 3.2中の 7 番の文書である. 文字領域抽出過程において, 誤った形状を有する文字候補領域が, 次第に修正されていく様子が判る. この傾向は, 文字列中の文字候補領域において顕著に現れている. また, 誤って抽出されている文字列領域が, 文字領域抽出過程で削除されている様子も判る. 文字列領域の削除については, 誤って抽出された文字列領域が正しく抽出されている文字候補領域と重複している場合に見受けられる. 以上の結果は, 文字形状に関する知識と文字配置に関する知識がお互いの不足部分を相補う形で考慮されることにより, 文字領域が抽出されていることを表す. すなわち, CSCE では, 文字形状及び文字配置に関する一般的知識が有効に利用されていると言える.

表 3.3(a) に, 本実験の実験結果を示す. 提案手法では, $p_{V_i} \geq 0.5$ を満たすノード V_iにより表される文字候補領域が, 抽出された文字領域に相当する. この文字候補領域を, 文書中の文字上に正しく生成することができた場合, 正しく文字領域を抽出することができたと解釈する. 日本語印刷文字の抽出率は, 表 3.3のように定義する. また, 我々の手法は, 前述の文字形状及び文字配置に関する一般的知識のみしか利用していないため, 文字候補領域は日本語印刷文字上ばかりではなく, アルファベットや記号など, 一般的知識を満たす他の文字上に形成される場合もある. そこで, 正当率は表 3.3 のように定義する.

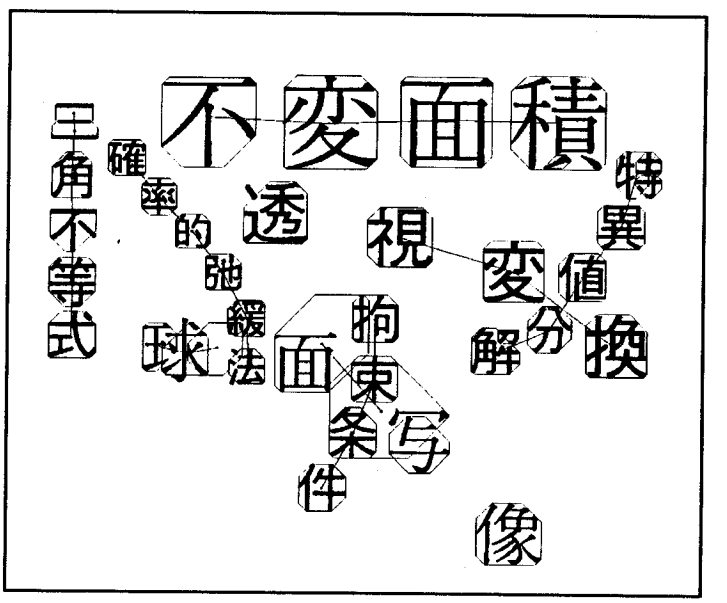
CSCE の有効性を示すため, 比較実験も同時に行なった. ここでの比較手法は, 書式のない文書に対しても適用可能な手法でなければならないので, 文書の書式に関する知識は一切用いられてはならない. 本実験では, 文書の構造に関する知識や, 文字列の属性値の一様性に関する知識を一切用いない, ボトムアップ的アプローチに基づく手法を, 比較手法として用いた. 手法の概要は以下の通りである. まず, 文書中の連結領域に対応する文字候補領域を生成する. 文字候補領域は, その周辺に存在する他の文字候補領域を統合しながら拡大していく. この際, 統合の対象となるのは, 距離 $h \cdot s/2$ 以内に存在する文字候補領域である. ただし, s は文字候補領域のサイズであ



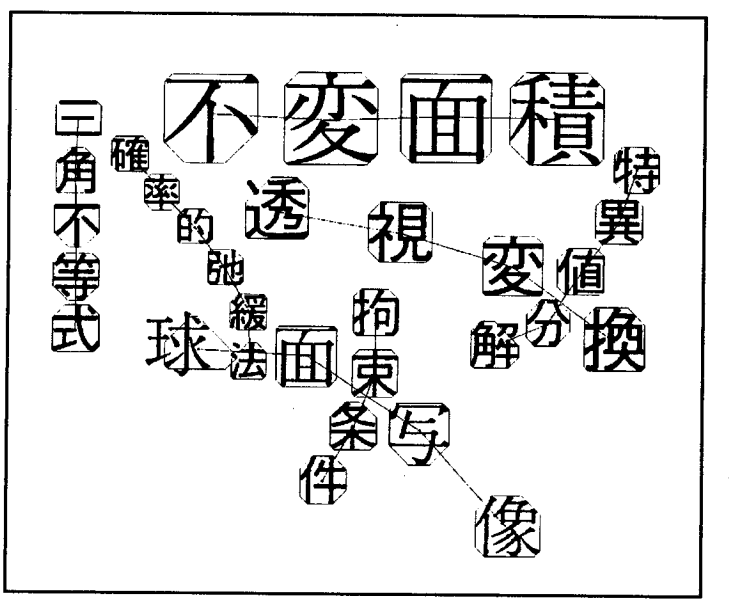
(a) η =0の時の文字候補領域



(b) η =7の時の文字候補領域



(c) η =10の時の文字候補領域



(d) η =20の時の文字候補領域

図 3.6 文字領域抽出過程における文字候補領域の状態変化

表 3.3 CSCE による文字領域抽出性能評価 (a)CSCE による結果 (b) 比較手法による結果

	書式あり		書式なし	
	(a)	(b)	(a)	(b)
日本語印刷文字数：N_j	3043	3043	257	257
文字として抽出された文字候補領域数：N_e	3664	4218	279	329
正しく抽出された日本語印刷文字領域数：N_{cj}	2893	2516	234	208
正しく抽出された印刷文字領域数：N_{cp}	3297	3055	234	208
抽出率：$R_e = N_{cj}/N_j$	95.1%	82.7%	91.1%	80.9%
正答率：$R_a = N_{cp}/N_e$	90.0%	72.4%	83.9%	63.2%

り，$s = \min\{s^{xy}, s^{uv}\}$ により定義される．s^{xy}, s^{uv}は文字候補領域に外接する二つの矩形の面積である．これらの矩形はそれぞれxy軸方向，uv軸方向に沿ったものである．文字候補領域の拡大は，統合が起こらなくなるまで再帰的に続けられる．以上の処理により最終的に残った文字候補領域を，抽出された文字領域と解釈する．本実験では，パラメータ $h = 1.5$ とした．これは，最も良好な抽出結果が得られた際のパラメータ値である．比較実験の結果を表 3.3(b) に示す．

表 3.3(a) より，交差する文字列を含むような書式のない文書画像に CSCE を適用しても，書式のある文書画像に適用した場合と大差のない精度で文字領域を抽出することができることを確認した．その際，表 3.1に示すパラメータ値を調整する必要も，全くない．この結果より，湾曲した文字列を含むような文書画像に対して全く対処できなかった従来手法と異なり，CSCE は，文書の種類に依存せず，極めて安定した性能を有していると言える．また，CSCE と比較手法の文字領域抽出性能を比較しても，CSCE が抽出率，正答率両者に関して良好な性能を有していることが判る．以上より，CSCE は，あらゆる文書に適用可能な，汎用的な文字領域抽出法であると言える．

表 3.4に，日本語印刷文字の文字列領域の抽出結果を示す．ここでは，文字列領域中で隣接する二文字の組を，文字列領域の一単位として扱うことにする．抽出率及び正当率は，表 3.4のように定義される．CSCE では，両端で $p_{V_i} \geq 0.5$ を有する文字候補領

表 3.4 CSCE による文字列領域抽出性能評価

	書式あり	書式なし
文字列数：N_l	2529	196
抽出された文字列領域数：N_{le}	2522	190
正しく抽出された文字列領域数：N_{lc}	2302	169
抽出率：$R_{le} = N_{lc}/N_l$	91.0%	86.2%
正答率：$R_{la} = N_{lc}/N_{le}$	91.3%	88.9%

域と継っており，かつ，$r_{V_iV_j} \geq 0.5$ を有するリンクを，抽出された文字列領域と解釈する．表 3.4より明らかなように，CSCE は，任意の文書画像から，文字列領域を精度良く抽出することもできる．文字列領域抽出に関しては，本来，本研究の目的ではない．しかし，図 3.5に示すような複雑な文字列領域を抽出できる手法がこれまでに提案されていない点を考えると，文字領域と同時に文字列領域も抽出できるという特徴は，非常に意義深いものであると考える．

図 3.7は，対象文書画像中の文字数と，文字領域抽出の計算時間の関係を図示したものである．本実験で用いた対象画像は，書式のある文書の一部を，様々な文字数で切りとったものである．図 3.7より明らかにように，文字数が増えるに従い，計算時間が増加していることが判る．これは，制約関数の最小化の際に，繰り返し処理による最小化手法を用いていることによる．この問題に対しては，例えば，並列処理による制約関数の最小化手法を導入することにより対処することができるのではないかと考えている．

以下では，CSCE と従来の文字領域抽出法を，その文字領域抽出性能と計算時間の面から比較検討する．従来の文字領域抽出法としては，前章で触れたように書式のある文書を対象とした手法と書式のない文書を対象とした手法に大別される．このうち，書式のない文書を対象とした研究では，抽出率や正答率などによる文字領域抽出性能の数量的な評価がなされていないため，CSCE との性能比較は困難である [岩城 85, 長石 93, Takizawa 94]．そこでここでは，書式のある文書を対象とした文字領域抽出に関する従来手法として大和らの研究 [大和 92] をとり上げ，これらと CSCE とについて比較する．

文字領域抽出性能については，我々の手法の文字領域抽出率が約 95%であるのに対

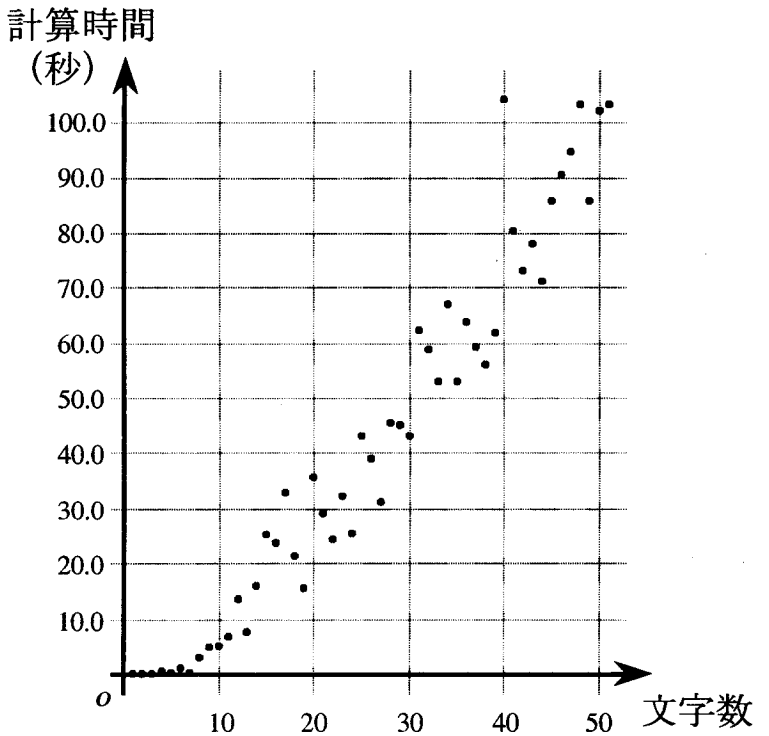
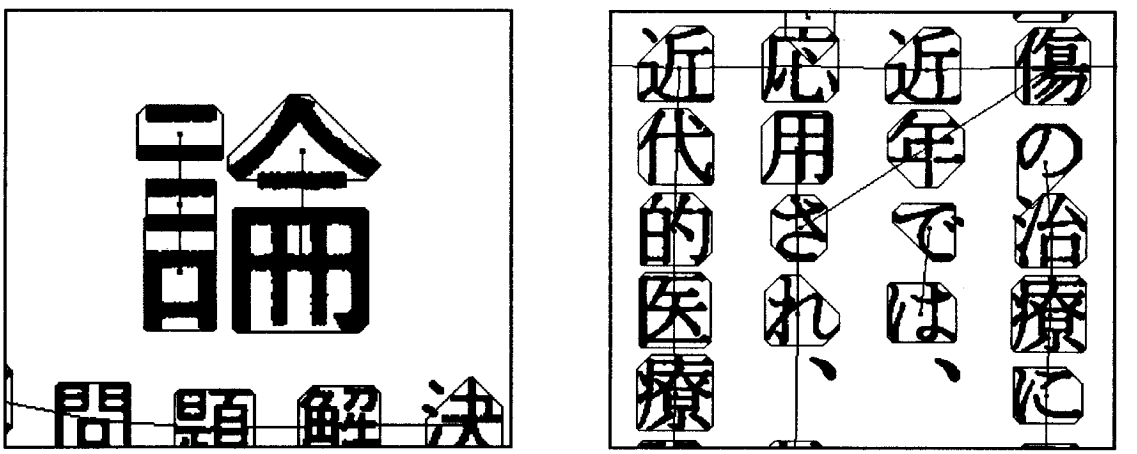


図 3.7 文字数と計算時間の関係

し, 大和らの手法では約 99%であった. この 4%の性能差は無視することができない程度のものである. 大和らの手法に限らず, 書式のある文書を対象とした多くの文字領域抽出手法が, 文字領域抽出率 99%を達成している. この点を考慮すると, 処理対象の文書の種類が既知である環境下では, CSCE は従来手法に比べはるかに劣っていることが判る. また, 従来手法が近年ようやく文書データベース構築の要素技術として脚光を浴びてきた面を考慮すると, CSCE は, 書式の有無に関わらず文字領域抽出率 99%以上の性能を実現しなければ, マルチメディア情報システムにおける要素技術となりえないことを示唆している.

計算時間の面から CSCE と従来手法を比較する. 我々の手法は, 図 3.7に示すような計算時間を要する. 単純に計算すると, 一文字あたりに要する計算時間は平均約 2 秒である. これに対し, 大和らの手法では一文字当たりの文字領域抽出のための計算時間は約 0.2 ミリ秒である. なお, 大和らの手法で用いた計算機は PC-9801VX21 である. この比較からも明らかなように, CSCE は文字領域抽出に多大な計算時間を要するた



(a) 文字領域中の文字列　　(b) 誤った方向に向いた文字列

図 3.8 誤抽出例

め, 現段階では実用向きではない. この計算時間が CSCE 実用化における最大の問題点である. 前述のように, 制約関数の最小化に計算時間の大半を要している点を考慮すると, 最小化手法の改良が急務であると考える.

最後に, CSCE による誤抽出例について考察する. CSCE では, 一般的知識として文字形状及び文字配置に関する知識を用いることにより文字領域を抽出している. これらの一般的知識は, せいぜい三つの文字に関する特徴について言及した局所的な知識である. よって, CSCE では, 文書の全体的な構造から考えると極めて不自然な誤抽出結果が得られる場合もある. 例えば, CSCE における典型的な誤抽出例として図 3.8 のような誤りを挙げることができる. 図 3.8(a) では, 文字中の局所領域に, 誤った文字列領域が形成されている. 図 3.8(b) では, いくつかの文字列領域が誤った方向に向いた形で抽出されている. 以上の文字領域抽出結果は明らかに誤りである. しかし, CSCE で用いている文字形状及び文字配置に関する一般的知識があまりにも局所的な文書構造にのみ言及しているため, 図 3.8に示す誤抽出結果は, 一般的知識をある程度満たしてしまっている. これらの誤抽出結果は制約関数の局所解であり, 制約関数の最小化によるアプローチでは回避することができない. 以上のような問題に対処するためには, 文書のより大局的構造を考慮した新たな処理の枠組が必要になると推察される. 例え

ば, 図 3.8に示す結果は一般的知識を適度に満足しているが, 文字列の構造に着目した場合, 妥当な結果であるとは言い難い. そこで, 文字列として妥当な構造を有する文字領域抽出結果が常に得られるようにCSCEの枠組を変更することにより, 図3.8に見受けられる誤りを修正することができると考えられる.

3.8 結言

本章では, 文字形状及び文字配置に関する一般的知識のみを用いて文字領域を抽出するためには制約充足のアプローチが有効であろうという観点に基づき, 制約関数の最小化による文字領域抽出法 CSCE について述べた. CSCE を, 書式の有無に関わりなく, あらゆる種類の文書に適用した結果, 文字領域抽出に関しては抽出率 91%以上, 正答率 83%以上の性能を確認した. また, 文字列領域抽出に関しては抽出率 86%以上, 正答率に関しては 88%以上の性能を確認した. この際, 書式の有無による手法の性能に, 大きな差異は見受けられなかった. 従来までに, 書式のない文書を扱った文字領域抽出法及び文字列領域抽出法がほとんど提案されていない現状を考慮すると, 上記の結果は良好なものであると考えることができる.

CSCE の特徴を列挙すると, 以下の通りである.

- 文字領域抽出のために用いている知識は文書の書式に全く依存しないため, 手法としての汎用性を有する.
- 所望の文字領域に関する全ての知識を制約として制約関数に与えることにより, 与えられた全ての知識の総合的に満足するような領域を, 文字領域として獲得することができる.
- 文字領域抽出と同時に, 文字列領域も抽出可能である.

一方, 問題点として, 多大な計算時間を要する点, また, 文書の全体的な構造から考えると不自然な文字領域抽出結果が得られる傾向があるという点を挙げることができる. これは, 制約関数の最小化というアプローチの特性, また, CSCE で用いられている知識が局所的であるという点に起因するものと思われる.

第 4 章

マルチエージェントシステムに基づく文字領域抽出法

4.1 緒言

前章では, 制約関数の最小化による文字領域抽出法 CSCE について述べた. CSCE は, 文書の種類や書式の有無などに全く依存しない知識のみを用いて文字領域を抽出するため, 任意の文書に適用可能であり, 手法としての汎用性を有している. このような汎用性は, 従来の文字領域抽出法にはない CSCE 固有の特徴である. しかしその一方, CSCE が用いている知識が不十分であるため, 場合によっては誤った文字領域が抽出されてしまう. 具体的には, 文字列として極めて不自然な構造を有する文字列領域が抽出されてしまい, その誤りが文字領域抽出にも悪影響を及ぼすというものである.

そこで, 前章の最後で指摘したように, 文字領域抽出結果が文字列として妥当な構造を有するように CSCE の処理の枠組を改良することができないかという点について検討する. すなわち, CSCE の処理形態を変更し, CSCE では明確に把握することができなかった文字列領域の構造を積極的に考慮することにより, 文字領域抽出精度の向上を試みる. この際に注意しなければならないのは, CSCE の汎用性を維持させなければならない点である. 例えば, 前章の実験で用いた書式のない文書画像のように, 様々な文字列が複雑に入り組んだ文書画像に対しても柔軟に対処できるような処理形態が要求される. 以上の点を考慮すると, CSCE の新たな処理形態では, 文書中に混在する様々な文字列の位置及びその構造を的確に把握することができ, かつ, 各文字列の構造に最も適した文字領域抽出処理を文字列ごとに適用することができなければならない.

そこで本章では, 上述のような処理の枠組として分散処理の一形態であるマルチエージェントシステムに着目し, マルチエージェントシステムに基づく文字領域抽出法 COCE (COordinative Character Extractor) について議論する [行天 94, 隅谷 95, Gyohten 95b, 行天 95, Gyohten 96]. マルチエージェントシステムとは, エージェントと呼ばれる自律的に動作する複数のプロセスがお互いに協力することにより, 与えられた問題を解決する処理形態である [奥乃 94]. 文字領域抽出問題の場合, 各エージェントは文書画像中の一つの文字列領域を担当することが望ましい. すなわち, エージェントは任意の文書の様々な文字列配置に対してその担当領域を柔軟に変化させ, 自分が担当すべき文字列領域を文書画像中から正確に探し出す. そして, 自分が担当している文字列領域固有の文字列構造を考慮に入れた上で, 文字列領域から文字領域を抽出することにより, CSCE 以上の精度で文字領域を抽出することが可能になると思われる. そこで, このようなマルチエージェントシステムを導入した場合, COCE の全体像がどのようになるかについて, まず最初に論ずる. その後, COCE 中で各エージェントが実行すべき処理について検討する. 具体的には, 各エージェントが文字列構造を考慮した場合, エージェントの文字領域抽出, 及び, 担当領域の変更としてどのような処理を施すことが望ましいかについて具体的に論ずる. 最後に, 書式のある文書及び書式のない文書を対象とした文字領域抽出実験, 特に, 前章で論じた CSCE との比較実験を通じて, COCE の有効性を明らかにする.

4.2 マルチエージェントシステム

マルチエージェントシステムは, 人工知能の一分野である分散人工知能において研究されてきたシステムであり, これまでに様々な検討がなされてきた. 分散人工知能の分野におけるマルチエージェントシステムの研究の目標は, 独自の知識に基づいて自律的に行動するエージェント間の協調動作の原理を, 理論的, もしくは実験的に明らかにすることにあった [石田 92, 桑原 93].

近年, マルチエージェントシステムを, 画像認識／理解 [松山 92, 角 94, 中村 2 94], ロボティクス [國吉 95] の分野に応用しようとする研究が盛んになりつつある. 特に, 画像認識／理解の分野では, 黒板モデルと呼ばれる分散並列的な問題解決の枠組が提案されて以降, マルチエージェントシステムを用いて, 従来の画像認識／理解における

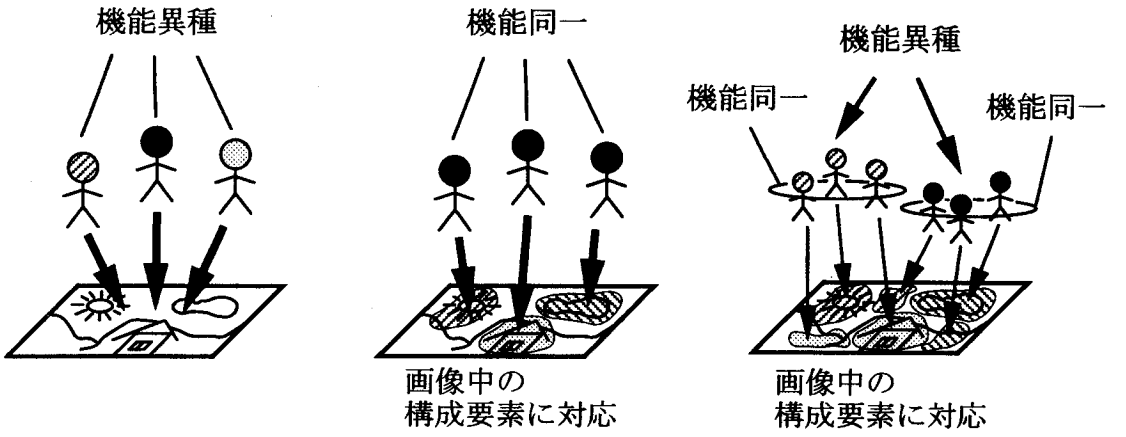


図 4.1 画像認識／理解におけるマルチエージェントシステムの方式

様々な手法のボトルネックに対処することを目指した研究が数多くなされてきた.

そこで本節では, 従来の画像認識／理解の分野における, マルチエージェントシステムを利用した様々な研究について概観する. 画像認識／理解にマルチエージェントシステムを導入するためには, 各エージェントが何を対象として何を実行するのかについて考えなければならない. 従来の手法は, 各エージェントの処理対象, 及び, 各エージェントの機能の差に基づき, 図4.1に示す三つの方式に大別できると考えられる [行天 95]. 以下に, 各方式について詳述する.

I) 機能分散方式 個々のエージェントの機能が異なり, かつ, 全てのエージェントは画像全体を処理対象とする方式である. この方式では, 各エージェントは複数の異なる機能を持つ画像処理アルゴリズムからなる. 各エージェントは並列・非同期に動作し, 個々の機能を入力画像に対して適用する. エージェントが各自の処理を協調的に進め, 個々の結果を統合することにより, 処理結果の精度を高めるという点が, 本方式の狙いである. このような方式を採用した研究として, ステレオ対応問題を扱った VIC-Stereo[Watanabe 90, Ohta 91] などがある.

II) 空間分散方式 個々のエージェントの機能が同一であり, かつ, エージェントの処理対象が画像中の各構成要素, すなわち, 部分画像である方式である. この方式では, エージェント間の協調により構成要素間の空間的位置関係などを把握し, 最終的に画像全体の大局的構造をエージェント単位で記述することが目的となる. この方式では, 画像中の構成要素が任意に配置されている場合でもエージェント間の協調により適応的に対処できるという処理の柔軟性が認められる. このような方式の例として, 領域分割問題 [和田 95] や地図中の等高線抽出問題 [Shimada 95] を扱った研究がある.

III) 機能／空間分散方式 先の二つを統合した方式, すなわち, 各エージェントの処理対象は画像中の各構成要素であり, かつ, 各エージェントの機能は処理対象の種類により異なる方式である. この方式では, 各エージェントは, 各自が担当する構成要素を抽出するのに最も適切な処理を施し, かつ, エージェント間の協調により構成要素の位置関係を把握することを狙っている. この方式の例として, シーン中のオブジェクト抽出問題を扱った SIGMA[松山 85] や PAFE[中村 2 93] などがある.

ここで, 上述のように分類したマルチエージェントシステムと文字領域抽出問題との関連について検討する. 本章で検討する文字領域抽出法において重要な点は, 文書画像中の各文字列領域の構造に最も適した文字領域抽出処理を, 文字列ごとに適用することである. マルチエージェントシステムの枠組で考えた場合, 各エージェントが基本的に一つの文字列領域を担当すると仮定する. そして, エージェントは自分が担当する文字列領域の構造を把握した上で, その領域に最も適した文字領域抽出処理を適用することにより, 担当する文字列領域から文字領域を抽出する. また, エージェント間の何らかの協調により文字列領域間の空間的位置関係を把握し, その結果を文字領域抽出精度の向上に役立たせるというアプローチが最も自然であると考えられる. 以上より, ここでは空間分散方式のマルチエージェントシステムを用いて文字領域抽出問題を扱うことが適当であると考えられる.

4.3 システムの概要

本節では, 空間分散方式のマルチエージェントシステムを用いた文字領域抽出法 COCE の概要について述べる. COCE では, 図 4.2に示すように, 各エージェントは文

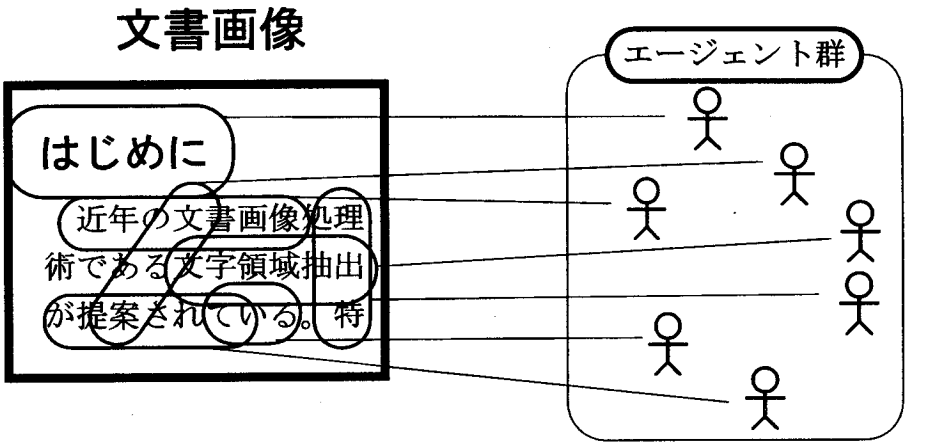


図 4.2 COCE におけるマルチエージェントシステム

字列領域の一部分に相当すると思われる領域を担当するものとする. このような処理形態により, エージェントは自分が担当する文字列領域に最も適した処理を施すことができる. また, 一つの文字列領域を担当しなければならないという前提以外, エージェントの担当領域に関する制約はない. よって, 任意の文書の文字列配置に対しても, エージェントは適応的にその担当領域を変化させながら対処することができる.

4.4 エージェントの処理

各エージェントが実行すべき処理は, 基本的には文字領域抽出処理である. ただし, 図4.2のようなシステムを導入することにより, 各エージェントは文書中の各文字列の位置や構造を把握することができる. これにより, エージェントはさらに以下のような処理を実行することができる.

a). 拡張 CSCE による文字領域抽出

b). 文字領域抽出結果の検証・修正

c). 担当領域拡大

a)に挙げた拡張 CSCE とは, 前章で論じた文字領域抽出法 CSCE の機能を拡張したものである. 拡張 CSCE は, 従来の CSCE の機能に加え, 文字列領域の構造も考慮し

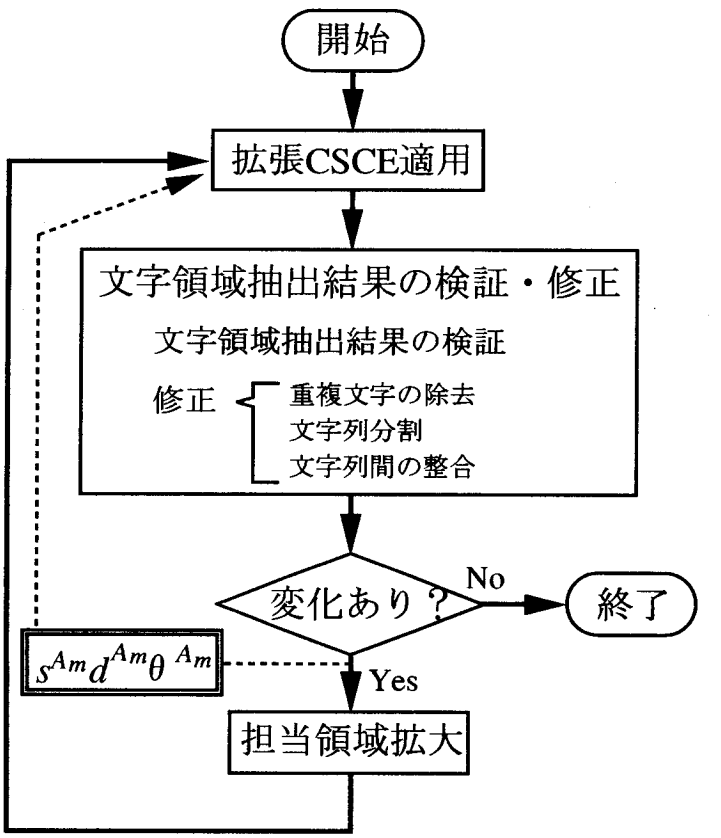


図 4.3 エージェントの処理

て文字領域を抽出する手法である. COCE では各エージェントが各文字列の文字列構造を把握することができる. そのため, エージェントは拡張 CSCE を用いてさらに正確に文字領域を抽出することができる.

また, COCE ではエージェント単位で文字列領域を把握することができるため, 文字列領域間の空間的位置関係を明らかにすることができる. そのため, b) に示す処理により, 各エージェントの文字領域抽出結果が, 文字列として妥当な構造を有しているかを検証することができる. また, 場合に応じた修正処理を文字列ごとに個別に施すことができる.

c)は, エージェントの担当領域を拡大することにより, 未抽出の文字列領域を文書画像中から見つけ出す処理である. 文書中には様々な文字列が混在してる. そのため, 文書画像中から何の手がかりもなしに文字列領域を抽出することは極めて困難である.

ところがCOCEでは, 各エージェントが担当する文字列領域の位置により, ある程度の文字列配置を推察することができる. 例えば, あるエージェントが担当する文字列領域の両端には, 未抽出の文字列領域が存在する可能性が高い. したがって, エージェントの担当領域を文字列方向に伸長することにより, 未抽出の文字列領域を見つけ出すことができると考えられる.

COCE中の各エージェントは, 上述の三つの処理を実行するものとする. その処理フローは図4.3の通りである. 各エージェントは, 拡張CSCEにより, 担当領域中から文字領域を抽出する. その後, エージェントは, 自分の文字領域抽出結果が文字または文字列として妥当であるかを検証する. 妥当でないと判断した場合はその原因を明らかにし, 抽出結果を強制的に修正する. この検証・修正は, エージェント個人でなされる場合もあれば, 他エージェントの助けを借りてなされる場合もある. 以上の処理を, 担当領域を文字列方向に拡張しながら, 文字領域抽出結果に変化がなくなるまで続ける. 最終的に, 各エージェントの担当領域が一つの文字列領域と同等になり, その文字列領域中の文字領域抽出結果も同時に得られる. 以下, 図4.3に示す各々の処理について詳述する.

4.4.1 文字領域抽出

各エージェントは, 図4.4に示すように自分の担当領域に対して領域モデルを生成し, 拡張CSCEを適用する. 拡張CSCEは, 前章で述べたCSCEを拡張したものであり, 新たに, 各文字列の構造に適した文字領域抽出が可能となっている. 具体的には, 各文字列における属性値である図4.5に示す文字サイズ, 文字間隔, 曲率は文字列中において一様であるという観点より, これらの属性値に対応する三つの目的関数F_s, F_a, F_uを追加している.

$$F_{s(G,s^{Am})} = \sum_{V_i \in \mathcal{V}} \sum_{C_j \in \mathcal{C}_{V_i}} \{ g_{V_iC_j} \oplus z_{V_iC_j(G,s^{Am})} \} \tag{4.1}$$

$$F_{a(G,R,d^{Am})} = \sum_{V_i \in \mathcal{V}} \sum_{\substack{V_j \in \mathcal{V}_{V_i} \\ i<j}} E(r_{V_iV_j}, J_a(\mathrm{rat}\{d_{V_iV_j}(G), d^{Am}\})) \tag{4.2}$$

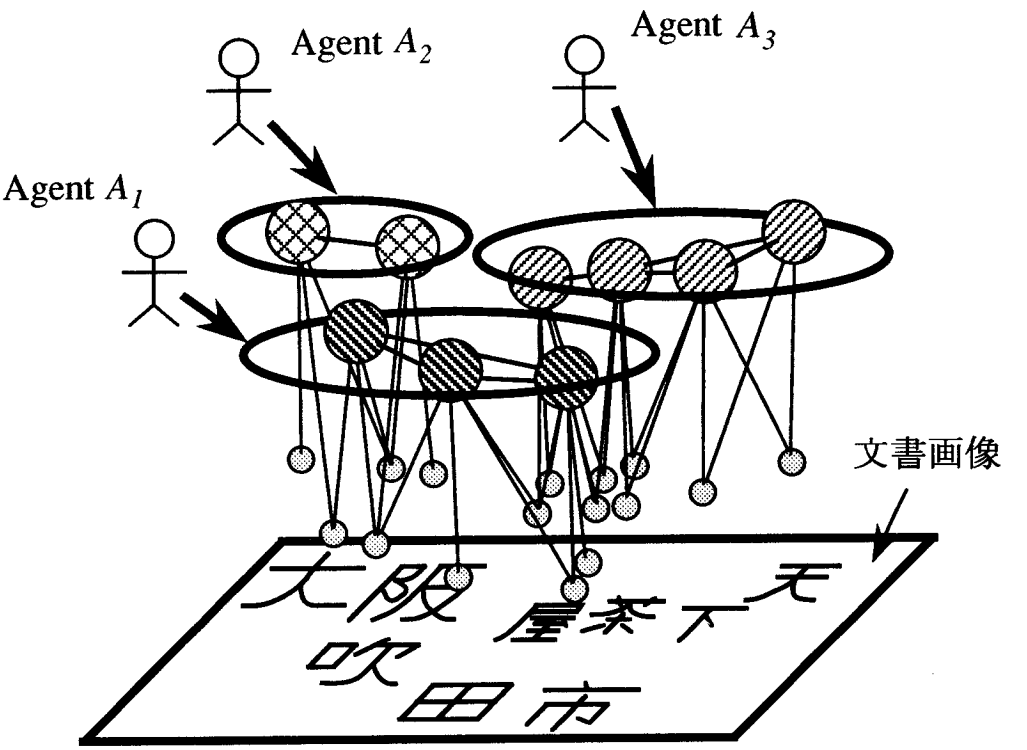


図 4.4 COCE における領域モデル

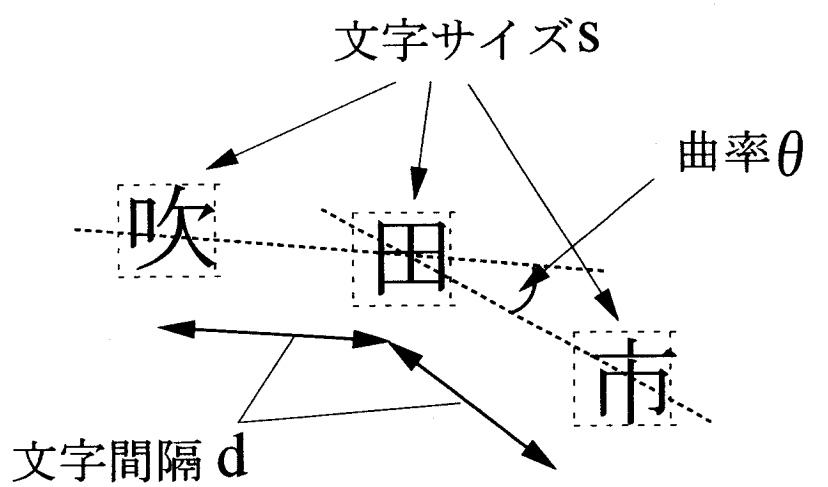


図 4.5 文字列の属性値

$$F_{u(G,R,\theta^{A_m})} = \sum_{V_j \in \mathcal{V}} \sum_{V_i \in \mathcal{V}_{V_j}} \sum_{\substack{V_k \in \mathcal{V}_{V_j} \\ i<k}} E(r_{V_iV_j} r_{V_jV_k}, J_u(|[|\theta_{V_iV_jV_k(G)} - \theta^{A_m} + \pi]_{2\pi}|/\pi)) \quad (4.3)$$

関数 $z_{V_iC_j(G,s^{A_m})}$ は, 連結領域 C_jが, 文字サイズ s^{A_m}の文字候補領域 V_iに含まれるかを表す関数であり, 含まれている場合は1, そうでない場合は0である. $\oplus$は排他的論理和を表す. $J_x(x \in \{a,u\})$, $\mathrm{rat}\{\cdot,\cdot\}$ $[\cdot]_{2\pi}$は, 前章の制約関数の定義の際に用いたものと同じである. s^{A_m}, a^{A_m}, θ^{A_m}はそれぞれ, エージェント A_mが担当している文字列の文字サイズ, 文字間隔, 曲率の平均 である.

目的関数 F_s, F_a, F_uは, 文字領域抽出結果中の文字列の属性値が, s^{A_m}, a^{A_m}, θ^{A_m}と一致する場合, 最小となる. これらの属性値については, 制約関数を最小化する際に予め与えておく必要がある. ここでは, 図4.3に示すように, 一手順前の文字領域抽出結果から算出されフィードバックされたものを与える.

エージェント A_mが実行する拡張CSCEでは, 上で触れた目的関数 F_s, F_a, F_uに加え, 前章で述べたCSCE中で用いられた目的関数 F_1, F_2, F_3, 及び, ペナルティ関数 F_d, F_{g1}, F_{g2}, F_r, F_pを用いて以下のように制約関数を定義する. この制約関数を最小化することにより, 従来のCSCEで用いていた一般的知識に加え, 文字列中の属性値の一様性も考慮して文字領域を抽出することができる.

$$\begin{aligned} F_{const(P,G,R,s^{A_m},d^{A_m},\theta^{A_m})} = \quad & S_1' F_{1(P,G)} + S_2' F_{2(G,R)} + S_3' F_{3(G,R)} \\ & + S_s' F_{s(G,s^{A_m})} + S_a' F_{a(G,R,d^{A_m})} + S_u' F_{u(G,R,\theta^{A_m})} \\ & + S_d' F_{d(P,G,R)} + S_{g1}' F_{g1(G)} + S_{g2}' F_{g2(G)} + S_r' F_{r(R)} + S_p' F_{p(P,R)} \end{aligned} \quad (4.4)$$

以後, エージェント A_mが文字として抽出した領域の集合を, $\mathcal{V}_{A_m}^L$ と表記する. すなわち, $\mathcal{V}_{A_m}^L = \{V_i | V_i \in \mathcal{V}_{A_m}, p_{V_i} \geq 0.5\}$ ($\mathcal{V}_{A_m}$は, A_mが有する領域モデルにより表現される文字候補領域の集合) である. また, エージェントにより, 文字として抽出された領域を, 仮文字領域と呼ぶことにする.

なお, COCEの初期段階では, 文字列属性の値を算出することはできない. この場合, 各エージェントは従来のCSCEを利用して文字領域を抽出する.

4.4.2 文字領域抽出結果の検証・修正

拡張 CSCE による文字領域抽出後, エージェントは文字領域抽出結果が文字／文字列として妥当な構造を有しているかを検証する. 具体的には, 拡張 CSCE により得られた仮文字領域が文字／文字列に関する最低限の条件を満たしているかを調べる. この条件を, 文字／文字列条件と呼ぶ. 文字／文字列条件が満たされていない場合, エージェントはその原因を明らかにし, 条件が満たされるように仮文字領域を強制的に修正する. 以上のような検証・修正処理により, エージェントの文字領域抽出結果が, 常に文字／文字列条件を満足することが保証される. 以下に, 検証・修正処理について述べる.

A) 文字領域抽出結果の検証 まず, 文字／文字列条件を定義するための準備として, 仮文字領域の集合 $\mathcal{V}^L$ 上の二項関係を定義する. ただし, $\mathcal{V}^L = \bigcup_{A_m \in \mathcal{A}} \mathcal{V}^L_{A_m}$ ($\mathcal{A}$ はエージェントの集合) であり, これは, 全エージェントの文字領域抽出結果を表す.

定義 1 (重複関係) 仮文字領域集合 $\mathcal{V}^L$ 上の二項関係$\bowtie$を

$$V_i \bowtie V_j \Longleftrightarrow g_{V_i C_k} = g_{V_j C_k} = 1 \text{ を満たす } C_k \text{が存在する}$$

と定義する. $V_i \bowtie V_j$であるとき. 仮文字領域 V_iと V_jは重複していることを表す.

定義 2 (隣接関係) 仮文字領域集合 $\mathcal{V}^L$ 上の二項関係$\sim$を

$$V_i \sim V_j \Longleftrightarrow r_{V_i V_j} \geq 0.5 \text{ を満たす } r_{V_i V_j} \text{が存在する}$$

と定義する. $V_i \sim V_j$であるとき. 仮文字領域 V_iと V_jは文字列領域中で隣接していることを表す.

これらの二項関係を用い, 以下のような文字／文字列条件を定義する. 文字条件は, 仮文字領域 $V_i (\in \mathcal{V}^L)$ に対して, 文字列条件は, 仮文字領域の集合 $\mathcal{V}^{L'} (\subset \mathcal{V}^L)$ に対して課せられる条件である.

定義 3 (文字条件) 仮文字領域 $V_i (\in \mathcal{V}^L)$ に関する文字条件は,

$$V_i \text{は文字条件を満たす} \Longleftrightarrow V_i \bowtie V_j \text{を満たす} V_j \in \mathcal{V}^L \text{が存在しない}$$

と定義される．本条件は，文字は互いに重複してはならないことを表す．

定義 4 (文字列条件) 仮文字領域集合 $\mathcal{V}^{L'}(\subset \mathcal{V}^L)$ に関する文字列条件は，

$\mathcal{V}^{L'}$は文字列条件を満たす $\Longleftrightarrow$ $\mathcal{V}^{L'}$は $\mathcal{V}^L$上の同値関係 $\sim^*$ による同値類の一つである

と定義される．本条件は，文字列は隣接する文字の数珠つなぎにより構成されなければならないことを表す．ただし，$\sim^*$は，$\sim^* = I \cup \bigcup_{n=1}^{\infty} \sim^n$で定義される$\sim$の反射的推移的閉包である．$I$は$\mathcal{V}^L$上の恒等関係である．$\sim^*$は反射的，推移的，対称的であるため，$\mathcal{V}^L$上の同値関係となる．ある仮文字領域 $V_i(\in \mathcal{V}^L)$ に対し，$[V_i] = \{V_j | V_j \in \mathcal{V}^L, V_i \sim^* V_j\}$ により定義される $\mathcal{V}^L$の部分集合 $[V_i]$ を，$\sim^*$による V_iの属する同値類と呼ぶ [小野 94]．

エージェント A_mは，自分の文字領域抽出結果 $\mathcal{V}^L_{A_m}$が文字列条件を満たしているか，また，担当する全ての仮文字列領域 $V_i(\in \mathcal{V}^L_{A_m})$ が文字条件を満たしているかを調べることにより，文字領域抽出結果の妥当性を検証する．

いま，文字／文字列条件検証の結果，エージェント A_mによる文字領域抽出結果 $\mathcal{V}^L_{A_m}$ が文字／文字列条件を満足していないと仮定する．文字／文字列条件が満足されない状況として，以下のいずれかが考えられる．ただし，$\mathcal{V}^L_{A_m}/\sim^*$は，$\mathcal{V}^L_{A_m}$の$\sim^*$による商集合，すなわち，$\mathcal{V}^L_{A_m}$上の同値関係$\sim^*$による同値類全体の集合を表す．また，$|A|$ は集合 A の要素の数を表す．

1. 文字条件が満たされていない．

   (a) $V_i \bowtie V_j$を満たす，$V_i, V_j \in \mathcal{V}^L_{A_m}$が存在する．すなわち，お互いに重複する仮文字領域がエージェント内に存在する．

   (b) $V_i \bowtie V_j$を満たす，$V_i \in \mathcal{V}^L_{A_m}$, $V_j \in \mathcal{V}^L_{A_n}$ $(m \neq n)$ が存在する．すなわち，お互いに重複する仮文字領域がエージェント間に存在する．

2. 文字列条件が満たされていない†．

   (a) $|\mathcal{V}^L_{A_m}/\sim^*| \neq 1$ となっている．すなわち，エージェント A_mの文字領域抽出結果内に複数の文字列領域が存在する．

†本来ならば，文字列条件を満足しないもう一つの状況として，「$V_i \sim^* V_j$を満たす，$V_i \in \mathcal{V}^L_{A_m}$, $V_j \in \mathcal{V}^L_{A_n}$ $(m \neq n)$ が存在する」を考えることができる．しかし，COCE では，異なるエージェント間に跨る隣接関係は定義されないので，このような状況はありえない．

COCE では, 1aに対しては重複文字の除去, 1bに対しては文字列間の整合, 2aに対しては文字列分割を適用することにより, 文字領域抽出結果 $\mathcal{V}_{A_m}^L$ を修正する.

B) 重複文字の除去 エージェント A_mの文字領域抽出結果 $\mathcal{V}_{A_m}^L$ 中に, 互いに重複する仮文字領域が存在する時に, 重複文字の除去処理が適用される. 本来, 二つの文字が重複することはありえないので, お互いに重複する仮文字領域の少なくとも一方は, 誤った文字を表現していると考えられる. 本処理では, 互いに重複する仮文字領域のうち, 誤っていると考えられる方を, 強制的に $\mathcal{V}_{A_m}^L$ から除去する. 具体的には, より多くの仮文字領域と重複し, かつ, より外接矩形の正方形性が満たされていないものから順番に除去していく.

C) 文字列分割 エージェント A_mの文字領域抽出結果 $\mathcal{V}_{A_m}^L$ 中に複数の文字列領域が含まれている場合に, 文字列分割処理が適用される. 本処理は, 一つのエージェント内に存在する複数の文字列領域を分離し, それぞれ別のエージェントに割り振ることにより, 強制的に文字列条件を満足させる. 具体的には, 文字領域抽出結果 $\mathcal{V}_{A_m}^L$ 内の余分な文字列領域の数だけエージェントを新たに生成し, それぞれのエージェントに $\mathcal{V}_{A_m}^L$ 内の余分な文字列領域を担当させる. 新たに生成されたエージェントは, 自分に割り振られた文字列領域を各自の文字領域抽出結果と解釈して, 以後の処理を継続する.

D) 文字列間の整合 文字列間の整合処理は, エージェント A_mが有する仮文字領域と, エージェント A_nが担当する仮文字領域とが重複している場合に適用される. 本処理では, まず, 重複の原因を明らかにし, その後に, 各原因に応じた修正を施す.

エージェント間の仮文字領域重複の原因として, 以下の二つの状況が考えられる.

原因 1 エージェント A_m, A_nが一つの文字列領域に割り当てられている.

原因 2 エージェント A_m, A_nのうち少なくとも一方が誤った文字列領域を担当している.

エージェント A_mと A_nが担当する文字列がお互いに類似し, かつ, 滑らかに継っている場合, 両エージェントは元々同一の文字列領域を分担していると考えることができる. よって, 重複の原因は原因 1によるものであると判断する. これ以外の場合は, 原因 2が

原因であると判断する．重複がいずれの原因によるものかを判断するために，ここでは類似度と平滑度と呼ばれる二つの尺度を導入している．その定義は，以下の通りである．

$$\mathrm{Sim}(A_m, A_n) = \mathrm{rat}\{s^{A_m}, s^{A_n}\} \cdot \mathrm{rat}\{a^{A_m}, a^{A_n}\} \cdot |[\theta^{A_m} - \theta^{A_n} + \pi]_{2\pi}|/2\pi \tag{4.5}$$

$$\mathrm{Smooth}(A_m, A_n) = |[|\phi_{(A_m,A_n)}| - \theta^{A_n} + \pi]_{2\pi}| \cdot |[|\phi_{(A_m,A_n)}| - \theta^{A_m} + \pi]_{2\pi}|/4\pi^2 \tag{4.6}$$

ただし，$\phi_{(A_m,A_n)}$ は，両文字列領域の重複部における曲率である．類似度は，エージェント A_m, A_nが担当する文字列の類似度を表す．類似度は，両文字列が似たような文字サイズ，文字間隔，曲率を有している場合に大きな値をとる．平滑度は，両文字列領域がどの程度平滑に継っているかを表す．平滑度は，両文字列領域が重複部分で滑らかに継っている場合に大きな値をとる．両尺度に対するしきい値 T_{Sim}, T_{Smooth}を考え，両尺度の値がしきい値以上である場合，重複の原因は原因1によるものとし，そうでない場合は，原因2によるものとする．

重複の原因が原因1にある場合，エージェント A_m, A_nのうちの一方を除去し，残りのエージェントに，両者の文字領域抽出結果 $\mathcal{V}^L_{A_m} \cup \mathcal{V}^L_{A_n}$を割り当てる．この処理は，両エージェントの文字領域抽出結果を統合したものを一つの文字列領域と解釈し，一つのエージェントに割り当てることに相当する．

重複の原因が原因2にある場合，両文字列領域のうち文字列らしくない方に属する重複仮文字領域を除去することにより，強制的に文字条件を満足させる．文字列らしさについては，文字列領域中の重複仮文字領域の集合と非重複仮文字領域の集合との類似度により判断する．この類似度が，しきい値 T_{Sim}以上である場合，文字列領域は文字列らしいと解釈する．

4.4.3 担当領域拡大

拡張 CSCE による文字領域抽出，結果の検証・修正を終えたエージェントは，自分が担当すべき文字列領域の未抽出部分を探し出すために，担当領域をさらに拡大する．具体的には，エージェント A_mは，自分の文字領域抽出結果 $\mathcal{V}^L_{A_m}$ のうち，お互い距離が最も離れている仮文字領域 V_i, V_jを求める．V_iと V_jの位置から，文字領域抽出結果 $\mathcal{V}^L_{A_m}$ の文字列方向及び両端を明らかにし，それを基にして，A_mの担当領域を，両端から文字

列方向に伸長する.

4.5 実験結果

COCE を, C 言語を用いて, SUN SPARC station 10 Model 30 上に構築した. 各エージェントは, TSS プロセスとして実装した. なお, 本実験で実験対象として用いた文書は, 前章で用いた 70 枚の書式あり文書及び 10 枚の書式なし文書である. COCE 内で用いられるパラメータの値は, 表 4.1の通りである. これらの値は, 実験的に得た値であり, 全ての実験において共通して用いたものである.

表 4.1 COCE のパラメータ値

式 (4.2), (4.3) 中のしきい値
$t_a^{min} = 0.0 \quad t_a^{mid} = 0.7 \quad t_a^{max} = 1.0$
$t_u^{min} = 0.0 \quad t_u^{mid} = 0.7 \quad t_u^{max} = 1.0$
式 (4.5) の係数値
$S_1' = S_2' = S_3' = 3/23$
$S_s' = S_a' = S_u' = 3/23$
$S_d' = S_{g1}' = S_{g2}' = S_r' = S_p' = 1/23$
文字列間の整合における式 (4.5), (4.6) のしきい値
$T_{Sim} = 0.75 \quad T_{Smooth} = 0.7$

図 4.6は, 書式のない文書からの, 文字領域及び文字列領域の抽出結果である. 各ウィンドウは, 各エージェントの担当領域を表す. 各ウィンドウ中の多角形は, 最終的に残った仮文字領域, すなわち, 抽出された文字領域を表す. また, 各文字領域間を結ぶ直線は, 文字領域間の隣接関係を表す. すなわち, この直線により結ばれている文字領域は, 文字列領域中で隣接していることを表す. 図 4.6より明らかなように, 本実験で用いられた文書画像は, 文字列が互いに入り組んだ, 複雑な構造を有している. しかし, 各エージェントは適切に文字列領域を探し出すことができ, また, 各文字列領域中から正しく文字領域を抽出することができたことが判る.

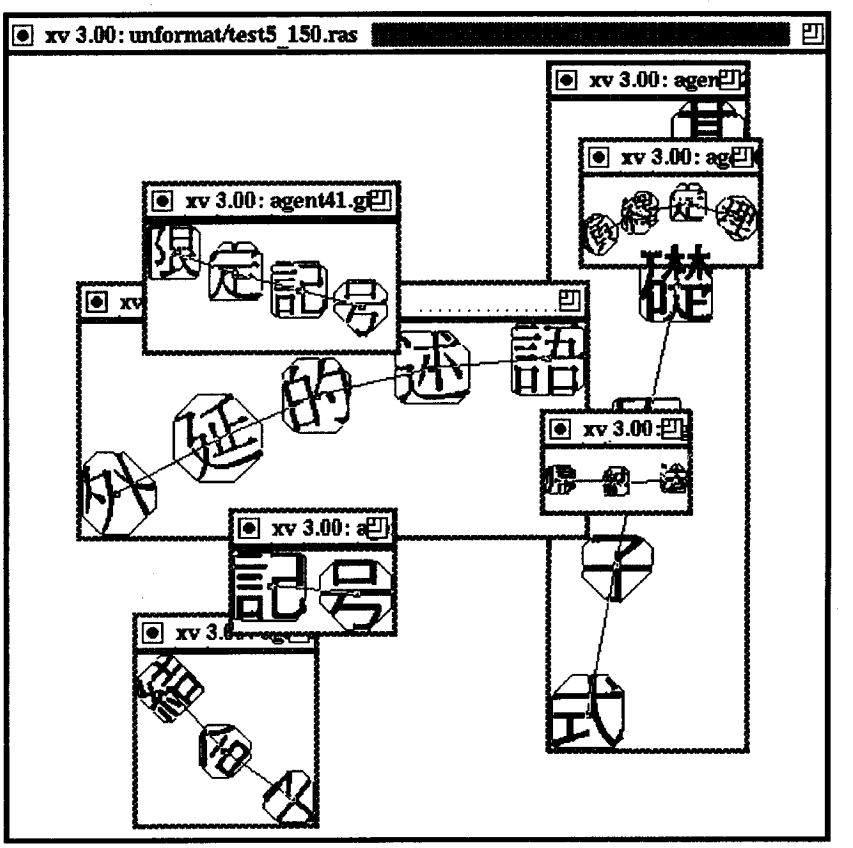


図 4.6 COCE による実験結果

表 4.2 COCE による文字領域抽出性能評価 (a)COCE による結果 (b)CSCE による結果

	書式あり		書式なし	
	(a)	(b)	(a)	(b)
日本語印刷文字数：N_j	3043	3043	257	257
文字として抽出された文字候補領域数：N_e	3484	3664	257	279
正しく抽出された日本語印刷文字領域数：N_{cj}	2959	2893	247	234
正しく抽出された印刷文字領域数：N_{cp}	3359	3297	247	234
抽出率：$R_e = N_{cj}/N_j$	97.2%	95.1%	96.1%	91.1%
正答率：$R_a = N_{cp}/N_e$	96.4%	90.0%	96.1%	83.9%

表 4.3 COCE による文字列領域抽出性能評価 (a)COCE による結果 (b)CSCE による結果

	書式あり		書式なし	
	(a)	(b)	(a)	(b)
文字列数：N_l	2529	2529	196	196
抽出された文字列領域数：N_{le}	2573	2522	192	190
正しく抽出された文字列領域数：N_{lc}	2412	2302	183	169
抽出率：$R_{le} = N_{lc}/N_l$	95.4%	91.0%	93.4%	86.2%
正答率：$R_{la} = N_{lc}/N_{le}$	93.7%	91.3%	95.3%	88.9%

表 4.2(a) と表 4.3(a) は, COCE による文字領域及び文字列領域の抽出結果である. 比較のため, 前章で議論した CSCE による結果も, 表 4.2(b) と表 4.3(b) にそれぞれ示す. 性能評価の基準は, 前章で述べた実験と同様, 抽出率と正答率である. 抽出率は, 文書中の文字（文字列）領域の総数に対する, 正しく抽出することができた文字（文字列）領域の数の割合を表す. また, 正答率は, 抽出された文字（文字列）領域の総数に対する, 正しく抽出された文字（文字列）領域の数の割合を表す. 正答率は, 文書中の文字（文字列）領域をどれくらい抽出することができたかを表す指標と解釈することができる. また, 正答率は, 文書中の文字（文字列）領域をどれくらい正確に抽出することができたかを表す指標と解釈することができる. なお, 前章での実験と同様に, 文字列領域抽出の性能評価においては, 文字列領域の単位は各文字列領域ではなく, 文字列領域中で隣接する一組の文字領域とした. 表 4.2と表 4.3により, 以下のような点が明らかになったと考える.

- 文字領域抽出及び文字列領域抽出において, COCE は, CSCE に比べ, 良好な性能を示した.
- 文字領域抽出において, COCE は CSCE に比べ, 抽出率で 2%-5%, 正答率で 6%-12%, 良好な性能を示した. 特に, 正答率は大幅に改善された.
- 文字列領域抽出において, COCE は CSCE に比べ, 抽出率で 4%-7%, 正答率で 2%-6%良好な性能を示した.

以上の結果は, CSCE の汎用性が COCE においても維持されているという点, 及び, COCE が各文字列の構造を考慮して適切に文字領域を抽出することができた点を実証するものであると考えている. また, COCE において正答率が大幅に改善された点については, 文字領域抽出結果の検証・修正処理による所が大きいと考えている. すなわち, 文字として, あるいは文字列として明らかに誤っているような文字領域抽出結果を, 容易に修正することができる機能を付加することにより, 手法としての信頼性が向上したものと考える. このような検証・修正処理は, これまでの CSCE になかった機能である.

図 4.7は, 本実験で用いた対象文書画像中の連結成分数と, COCE 及び CSCE による文字領域抽出の計算時間の関係を図示したものである. COCE 中では, 各エージェ

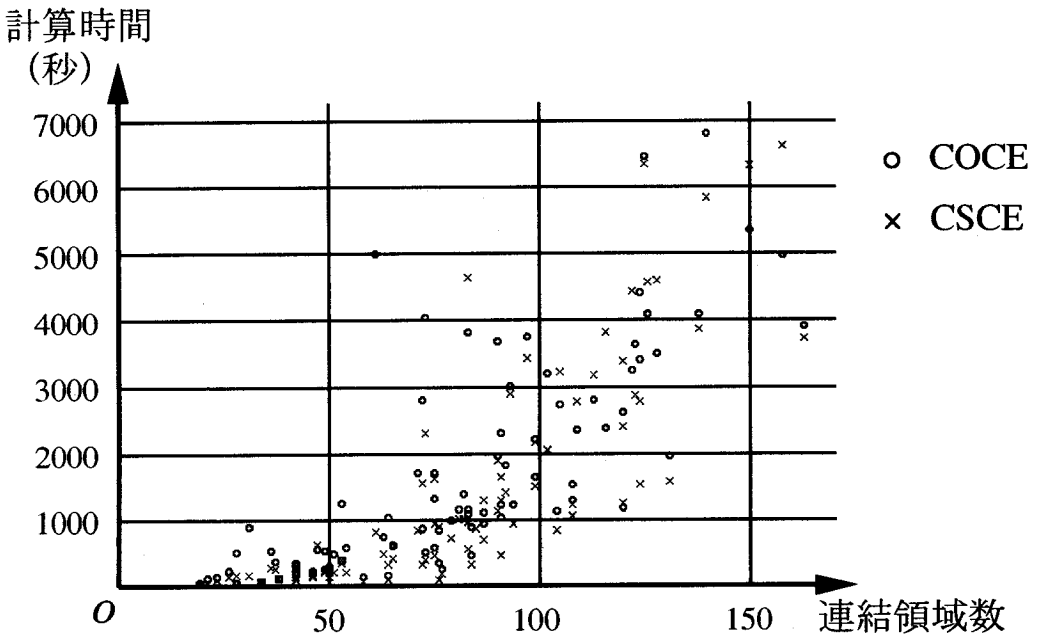


図 4.7 連結領域数と計算時間の関係

ントは基本的に CSCE を実行する. そのため, CSCE と同様に, COCE の処理時間が多大であるという点は否めない. ところが, CSCE と COCE の処理時間を比較した結果, COCE の処理時間は CSCE のそれに比べて, 平均 1.58 倍であった. COCE 中には CSCE を実行するエージェントが複数存在する点, また, COCE は一台の計算機上に実装されている点を考慮すると, この処理時間の差異はわずかなものであると捉えることができる. 通常, 複数の処理プロセスが存在するような処理環境を想定した場合, お互いの処理プロセス間の整合性をどのように維持するかが問題となる. しかし, COCE では, エージェント間の整合性維持のために要した時間が多大であったような形跡は見受けられない. この結果は, COCE の良好な収束性を示すものであると考える.

ここで, 前章と同様に, COCE と大和らの文字領域抽出手法 [大和 92] とを比較する. 文字領域抽出性能については COCE は約 97%の抽出率を示しており, 大和らの手法の抽出率 99%に大きく近付いている. これは, COCE のマルチエージェントシステムのアプローチが有効に働いたことを示している. また, 計算時間については, COCE は CSCE と同様に, 文字領域抽出に多大な時間を要する. しかし, 前述のように, COCE

とCSCEの間には計算時間に大きな差はなかった．これは，COCEにおいて導入したマルチエージェントシステムのアプローチが，計算時間増大の要因ではないことを表す．今後COCEの計算時間を改善するためには，各エージェントが実行する拡張CSCEにおける制約関数の最小化手法を改良しなければならないという点は，前章で論じたCSCE改良に関する検討と同様である．

最後に，誤抽出結果について考える．実験で用いた文書中には，日本語印刷文字の他に，アルファベット，句読点，数式など，様々な構成要素が含まれている．これに対し，COCEは，文書中には日本語印刷文字のみしか含まれていないという前提に基づいている．そのため，日本語印刷文字ではない構成要素を，誤った文字として抽出してしまう傾向があった．さらに，書式のある文書中で，平行に並んでいる文字列間の距離が，文字列中の隣接文字間の距離に比べて短い場合，COCEは誤った文字列領域を抽出してしまうことがあった．このような誤抽出結果は，COCEの枠組内では，もはや回避することはできない問題である．この問題に対処するためには，COCEに文字認識機能を持たせ，文章中の語や文などの文脈情報に関する知識を積極的に利用することが必要であると思われる．また，もう一つの問題点として，あるエージェントが文書中の文字列領域を探し出すことに失敗した場合，その文字列領域中の文字領域を抽出することが不可能になってしまうという点を挙げることができる．これは，各文字列領域が一つのエージェントの管轄下にあるため，あるエージェントが文字領域抽出に失敗した場合，そのエージェントの管轄下にある文字列領域を抽出すること，また，その文字列領域中の文字領域を抽出するエージェントがいなくなってしまう点に問題がある．この問題に対処するためには，多くの冗長なエージェントを用意し，誤った挙動をするエージェントを補佐させることが重要であると考える．ある文字列領域に対して複数のエージェントを割り当て，その文字列領域中の文字領域抽出結果に関していくつかの選択肢を用意できるような処理形態が必要であると思われる．

4.6 結言

本章では，前章で述べた文字領域抽出法CSCEにマルチエージェントシステムを導入することにより，文書中の各文字列固有の文字列構造を考慮した文字領域抽出法COCEについて述べた．前章と同様，COCEをあらゆる種類の文書に適用した結果，文

字領域抽出に関しては抽出率及び正答率 96%以上, 文字列領域抽出に関しては抽出率及び正答率 93%以上の性能を確認した. この結果は, CSCE の性能を大幅に上回るものであり, COCE の有効性を確認することができたと考えている.

COCE は, 基本的には CSCE の制約充足の概念に基づく手法であるため, CSCE と同様の特徴を有する. ただし, マルチエージェントシステムを導入することにより, 新たに以下のような特徴が加わっている.

- 各エージェントが, 与えられた文書の種類に応じて適応的に担当領域を変化させることにより, CSCE の汎用性が保存されている.
- 各エージェントが一つの文字列領域を個別に処理することにより文字列の属性値について考慮することができ, その結果, 文字領域抽出及び文字列領域抽出性能が CSCE に比べ大幅に改善されている.
- 文字領域結果が文字／文字列として妥当な構造を有しているかを各エージェントが検証し, 場合に応じて修正を施すことにより, 文字領域抽出の信頼性が CSCE に比べ大幅に改善されている.

第 5 章

結論

近年盛んに検討されているマルチメディア情報システムを構築する上では, 汎用的文書画像解析が必須である. 特に, 現在未検討であり, かつ, 汎用的文書画像解析の根幹となりうる要素技術は, 任意の文書からの文字領域抽出であると考えられる. そこで本論文では, 制約充足に基づく文書画像からの文字領域抽出法について議論し, 任意の文書からの文字領域抽出において有効であると思われるアプローチを具体的に示した. 以下に, 本研究で得られた諸結果をまとめる.

まず, 従来の文字領域抽出法を概観し, 従来手法のアプローチ及びその問題点などについて述べた. 従来手法の文字領域抽出法のほとんどは, 扱うことができる文書の種類を限定しているか, あるいは, 性能の面で問題がある. このような従来手法のアプローチを汎用的文字領域抽出法に応用することは不適切である点を指摘した. その上で, 真の意味で任意の文書に適用可能な汎用的文字領域抽出法を実現するためには, 文書の種類や書式の有無などに全く依存しない一般的な知識のみを用いて文字領域を抽出することが必要になる点を示唆した. そして, このような一般的知識として, 文字形状及び文字配置に関する一般的知識を具体的に列挙した.

次に, 文字形状及び文字配置に関する一般的知識を用いた文字領域抽出法について論じた. 一般的知識の曖昧性を考慮すると, 文字領域抽出の際に一般的知識を利用するためにはこれらの知識を総合的に扱わなければならない. 各知識の不十分点を, お互いに補うような形で知識活用がなされなければならないと考えられる. このような知識活用を実現するためには制約充足に基づいたアプローチが有効であろうという観点から, 制約関数の最小化による文字領域抽出法について議論した. 本手法を様々な文書に適用した結果, 全ての一般的知識が満たされているような文字領域を抽出することが

できた．また，文字列領域も同時に抽出することができることを明らかにした．さらに，文書の種類や書式の有無が，文字領域抽出性能に影響を与えていない点を確認した．以上の結果は，制約関数の最小化による文字領域抽出法が汎用的文字領域抽出の有効なアプローチになりうることを示唆していると考えている．

最後に，先に述べた制約関数の最小化による文字領域抽出法の拡張手法として，文字形状及び文字配置に関する一般的知識ばかりではなく，文字列の構造も考慮した文字領域抽出法について論じた．本手法は，分散処理環境の一種であるマルチエージェントシステムを導入することにより各文字列の構造に関する特性を考慮して文字領域を抽出することができる．また，抽出された文字領域は，文字／文字列として妥当な構造を常に有することが保証されている．先の実験と同様に，本手法を様々な文書画像に適用した．その結果より，制約関数の最小化による文字領域抽出法の汎用性を維持したまま，文字領域抽出及び文字列領域抽出に関する性能が全て向上していることを確認した．特に，手法の信頼性が大幅に向上していることを明らかにした．以上の結果は，マルチエージェントシステムが有する適応的な処理形態が，汎用的文字領域抽出における処理の枠組として有効であることを実証していると考える．

以上，本論文で論じた文字領域抽出法の特徴を総括的に述べると，以下の三点に集約される．

- 文書の種類や書式に依存しない，文字に関する一般的な知識のみを利用しているため，任意の文書からの文字領域抽出が可能である．
- 制約関数の最小化というアプローチを用いることにより，一般的知識を総合的に満たすような文字領域を文書画像中から抽出することができる．
- 各エージェントが文字列領域を担当するようなマルチエージェントシステムを導入することにより，文字列として適切な構造を有している文字領域及び文字列領域を文書画像中から抽出することができる．

以上の特徴は，今後汎用的な文字領域抽出法，ひいては汎用的文書画像解析における様々な問題点を解決する糸口になるのではないかと考えている．最後に，本研究に関する今後の課題を挙げる．

第一に，計算時間の問題を挙げることができる．本論文で論じた手法では，繰り返し処理による最小化手法により制約関数の最小解を導いている．このような最小化手法

は最小解の導出に多大な時間を要するという欠点を有しており，文字領域抽出法の実用化という観点から考えた場合，現実的な手法であるとは言い難い．この問題に対処するためには二つのアプローチがあるのではないかと考えている．一つは，場合に応じて一般的知識の数を減らしたり，適当な前処理部を付加したりすることである．本論文で論じた文字領域抽出法は，真の意味での汎用性を目指すために，文書の種類などに全く依存しない一般的知識全てを制約として制約関数に反映している．しかし，本手法が用いられる状況によっては，前処理を施すことにより容易に文字の候補となるべき領域を絞り込むことが可能になると考えられる．また，全ての一般的知識を総合的に考慮する必要がない状況も考えることができる．その場合，一般的知識ではなく，状況依存の知識をまず用いて文書中の文字領域となるべき候補領域を絞り込む．その後に，必要となる一般的知識のみを反映した制約関数を用いて文字領域を抽出することにより，計算時間の短縮を図ることができると思われる．以上の改良は，本論文で論じた手法が用いられるマルチメディア情報システムに完全に依存する．本論文は，汎用的文字領域抽出法がとるべきアプローチについて論じたものである．したがって，以上のような状況依存の改良手法に関する詳細な検討はここでは省略する．もう一つは，関数の最小化手法そのものの高速化である．近年，繰り返し処理による最小化手法を並列計算機上に構築することにより，計算時間を大幅に短縮する研究が進められている [Ingber 92]．本論文で述べた手法についてもそのような技術を導入することにより，文字領域抽出の計算時間を短縮することができるものと思われる．

第二に，言語情報の利用を挙げることができる．本論文で論じた手法は，文字の図形的特徴のみを考慮して文字領域を抽出するものであった．このようなアプローチでも良好な文字領域抽出結果を獲得することができることは，本論文での実験より明らかである．しかし，本論文で論じた手法による文字領域抽出のほとんどの誤抽出例は，一般的知識をある程度良好に満たしてしまっている．このような誤りを，図形的特徴のみを考慮して除去することは極めて困難であると推察される．今後，本論文で議論した文字領域抽出法をさらに改良するためには，抽出する文字の意味的な情報，すなわち，言語情報も考慮しなければならないと考えられる．具体的には，文字領域抽出後，各文字領域に対する文字認識処理による結果と，文書中の語や文などの文脈情報に関する知識を照合し，その結果を基にしたフィードバックにより，より精度の高い文字領域抽出を実現するというアプローチが考えられる [佐々木 94]．

最後に, 本研究の背景である汎用的文書画像解析技術の確立を挙げることができる. この課題は文書画像解析において最も重要かつ最も困難な問題であり, 今後多くの研究者により様々な検討がなされなければならない. 本論文で提唱したアプローチは, このような汎用的文書画像解析を実現する際に何らかの参考になるのではないかと考えている. 例えば, 汎用的文書画像解析では, やはり文書の種類や書式の有無などに依存しないような一般的な知識のみを用いて文書中の各構成要素を抽出しなければならない. いわゆる一般的知識を用いなければならないという点では, 本論文で論じた文字領域抽出法と通じるものがある. この観点を鑑みれば, 制約充足に基づいた領域抽出法, すなわち, 制約関数の最小化に基づく領域抽出法が, 一般的知識を利用していく上で有効なアプローチとなりうるのではないかと考える. また, 汎用的文書画像解析において利用すべき一般的知識は, 文書画像中の様々な構成要素について個別に言及したものであると予想される. このような知識を扱う処理形態として, マルチエージェントシステムは有効なのではないかと考える. 各エージェントに文書中の各構成要素を担当させ, その構成要素に関する知識をエージェントの処理の際に利用させることにより, 様々な知識を, システム中に容易に導入することができると推察される. ただし, 以上のような構成を有する汎用的文書画像解析においては未検討の問題も存在する. その一例として, 文書画像構成要素の階層性に準じたマルチエージェントシステムに関する問題を挙げることができる. 第2章で触れたように, 文書中の各構成要素の関係は階層性を有する. このような階層性を考慮した文書画像解析をマルチエージェントシステムにより実現するためには, 各構成要素間の階層関係をそのまま反映したようなマルチエージェントシステムを導入する必要がある. しかし, 階層性を有するマルチエージェントシステムについては本論文では言及していない. この点は, 今後検討していかなければならない課題である.

文書は人間の生活に密着した情報伝送媒体であり, かつ, 情報記録媒体である. 当然, 計算機によりその情報処理を自動化しようとすることも自然な流れであると考える. しかし, 従来の文書画像解析に関する研究が処理対象となる文書の種類を限定したものばかりであるという現状を振り返ると, 真の意味で計算機に文書を「読ませる」ような研究については, まだその検討が始まったばかりの段階であると言わざるおえない. 第1章で述べたように, 近い将来, マルチメディア情報システムに関する検討が盛んになるにつれ, 汎用的な文書画像解析技術の重要性はますます高まっていくであろ

う. 本研究の成果が, これらの技術を実現する際の一助となれば筆者の最も幸いとするところである.

参考文献

[秋山 84] 秋山 照雄, 内藤 誠一郎, 増田 功 : "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", 電子通信学会論文誌 (D), vol. J67-D, no. 10, pp. 1194–1201 (1984).

[秋山 86] 秋山 照雄, 増田 功 : "周辺分布, 線密度, 外接矩形特徴を併用した文書画像の領域分割", 電子通信学会論文誌 (D), vol. J69-D, no. 8, pp. 1187–1196 (1986).

[阿曽 93] 阿曽 弘具, 大町 真一郎, 木村 正行, 勝山 裕 : "高速高精度知的認識システム SEIUN", 電子情報通信学会論文誌 (D-II), vol. J76-D-II, no. 3, pp. 474–484 (1993).

[石田 92] 石田 亨, 桑原 和宏 : "分散人工知能 (1):協調問題解決", 人工知能学会誌, vol. 7, no. 6, pp. 945–954 (1992).

[茨木 93] 茨木 俊秀, 福島 雅夫 : "最適化の手法", 共立出版 (1993).

[岩城 85] 岩城 修, 久保田 一成, 荒川 弘熙 : "近接線密度法による文字・図形分離抽出", 電子通信学会論文誌 (D), vol. J68-D, no. 4, pp. 821–828 (1985).

[奥乃 94] 奥乃 博 : "マルチエージェントと協調計算 III", 近代科学社 (1994).

[小野 94] 小野 寛晰 : "情報代数", 共立出版 (1994).

[嘉納 87] 嘉納 秀明 : "システムの最適理論と最適化", コロナ社 (1987).

[黄瀬 89] 黄瀬 浩一, 杉山 淳一, 馬場口 登, 手塚 慶一 : “レイアウトモデルに基づく文書構造解析”, 電子情報通信学会論文誌 (D-II), vol. J72-D-II, no. 7, pp. 1029–1039 (1989).

[北橋 90] 北橋 忠宏, 安部 憲広, 馬場口 登: “文書画像の知的コミュニケーションのための画像の記号化と論理的意味”, 電子情報通信学会技術研究報告, HC90-19 (1990).

[行天 93a] 行天 啓二, 馬場口 登 : “制約充足型文字領域抽出の基礎検討”, 電子情報通信学会技術研究報告, PRU92-119 (1993).

[行天 93b] 行天 啓二, 馬場口 登, 北橋 忠宏 : “コスト最小化による 2 次元画像からの文字領域抽出”, 情報処理学会第 47 回全国大会 2-135 (1993).

[行天 94] 行天 啓二, 馬場口 登, 北橋 忠宏 : “文書画像からの文字領域の協調的抽出法”, 情報処理学会第 49 回全国大会 2-209 (1994).

[行天 95] 行天 啓二, 馬場口 登, 角所 考, 北橋 忠宏 : “文書画像からの文字切り出しのためのマルチエージェントシステム”, 電子情報通信学会技術研究報告, PRU95-152 (1995).

[國吉 95] 國吉 康夫 : “マルチロボットにおける観察に基づく協調”, 電子情報通信学会技術研究報告, PRU95-156 (1995).

[桑原 93] 桑原 和宏, 石田 亨 : “分散人工知能 (2):交渉と均衡化”, 人工知能学会誌, vol. 8, no. 1, pp. 17–25 (1992).

[佐々木 94] 佐々木 寛, 城風 敏彦, 岩根 典之, 木下 哲男 : “DAI 的手法による文字切り出し方式の一検討”, 電子情報通信学会技術研究報告, AI93-79 (1994).

[篠田 92] 篠田 浩一郎: “形象と文明”. 白水社 (1992).

[角 94] 角 保志 : “画像理解システムにおける機能分散的協調処理”, 人工知能学会誌, vol. 9, no. 5, pp. 631–636 (1994).

[隅谷 95] 隅谷 倫子, 行天 啓二, 角所 考, 馬場口 登, 北橋 忠宏 : "マルチエージェントシステムを利用した文字領域抽出法", 1995年電子情報通信学会総合大会 D-574 (1995).

[田村 85] 田村 秀行 : "コンピュータ画像処理入門", 総研出版 (1985).

[辻 86] 辻 善丈, 浅井 紘 : "分散最小基準に基づく適応型文字分離方式", 電子通信学会論文誌 (D), vol. J68-D, no. 8, pp. 1497–1504 (1986).

[長石 93] 長石 道博 : "視覚の誘導場による手書き文字切出し", 電子情報通信学会論文誌 (D-II), vol. J76-D-II, no. 9, pp. 1948–1956 (1993).

[中野 88] 中野 康明, 藤澤 浩道 : "自動ファイリングのための文書理解の一方式", 電子情報通信学会論文誌 (D), vol. J71-D, no. 10, pp. 2050–2058 (1988).

[中村 1 84] 中村 納, 氏家 誠, 岡本 教佳, 南 敏 : "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", 電子通信学会論文誌 (D), vol. J67-D, no. 11, pp. 1277–1284 (1984).

[中村 1 85] 中村 納, 鈴木 弘之, 南 敏 : "横書き日本語文書における個別文字の抽出", 電子通信学会論文誌 (D), vol. J68-D, no. 11, pp. 1899–1909 (1985).

[中村 2 93] 中村 裕一, 長尾 真 : "並列探索による画像特徴抽出－ PAFE における並列オブジェクトによる探索－", 人工知能学会誌, vol. 8, no. 1, pp. 65–78 (1993).

[中村 2 94] 中村 裕一 : "分散協調処理による空間的構造の抽出", 人工知能学会誌, vol. 9, no. 5, pp. 637–643 (1994).

[成瀬 92] 成瀬 博之, 渡辺 豊英, 駱 琴, 杉江 昇 : "枠罫線を情報を用いた帳票文書の構造認識" 電子情報通信学会論文誌 (D-II), vol. J75-D-II, no. 8, pp. 1372–1385 (1992).

[長谷 87] 長谷 博行, 米田 政明, 酒井 充, 吉田 順作 : "図書目録カードの自動項目分類について", 電子情報通信学会論文誌 (D), vol. J70-D, no. 8, pp. 1579–1588 (1987).

[松山 85] 松山 隆司, ビンセント ハング : "画像理解システム SIGMA", 情報処理学会論文誌, vol. 26, no. 5, pp. 877–889 (1985).

[松山 92] 松山 隆司: "分散協調処理による画像理解", 計測と制御, vol. 31, no. 11, pp. 1149–1154 (1992).

[美濃 93] 美濃 導彦: "文書画像処理の現状と動向", 電子情報通信学会誌, vol. 76, no. 5, pp. 502–509 (1993).

[宮原 89] 宮原 末治, 木村 義政, 豊田 充, 宮田 一人 : "部分パターンによる可変ピッチ文書からの文字切出しと認識", 電子情報通信学会論文誌 (D-II), vol. J72-D-II, no. 6, pp. 846–854 (1989).

[山田 93] 山田 満 : "文書画像の ODA 論理構造化文書への変換方式", 電子情報通信学会論文誌 (D-II), vol. J76-D-II no. 11 pp. 2274–2284 (1993).

[大和 92] 大和 一晴, 宮脇 冨士夫, 畑 豊, 賀 先楓 : "日本語印刷文書の読取りシステム", 電子情報通信学会論文誌 (D-II), vol. J75-D-II, no. 2, pp. 257–266 (1992).

[駱 92] 駱 琴, 渡邉 豊英, 杉江 昇 : "ルールベースの適用による日本語新聞誌紙面の構造認識", 電子情報通信学会論文誌 (D-II), vol. J75-D-II, no. 9, pp. 1514–1525 (1992).

[和田 95] 和田 俊和, 野村 圭弘, 松山 隆司 : "分散協調処理による画像の領域分割法", 情報処理学会論文誌, vol. 36, no. 4, pp. 879–891 (1995).

[Fletcher 88] L.A.Fletcher and R.K.Kasturi : "A Robust Algorithm for Text String Separation from Mixed Text/Graphics Images",

IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 10, no. 6, pp. 910–918 (1988).

[Gyohten 93] K.Gyohten, N.Babaguchi and T.Kitahashi : "Region Extraction by Constraint Satisfaction", Proceedings of Asian Conference on Computer Vision '93, pp. 846–849 (1993).

[Gyohten 95a] K.Gyohten, N.Babaguchi and T.Kitahashi : "Constraint Satisfaction Approach to Extraction of Japanese Character Regions from Unformatted Document Image", IEICE Transactions on Information and Systems, vol. E78-D, no. 4, pp. 466–475 (1995).

[Gyohten 95b] K.Gyohten, T.Sumiya, N.Babaguchi, K.Kakusho and T.Kitahashi : "Extracting Characters and Character Lines in Multi-Agent Scheme", Proceedings of the Third International Conference on Document Analysis and Recognition, pp. 305–308 (1995).

[Gyohten 96] K.Gyohten, T.Sumiya, N.Babaguchi, K.Kakusho and T.Kitahashi : "A Multi-Agent Based Method for Extracting Characters and Character Strings", IEICE Transactions on Information and Systems, vol. E79-D, no. 5 (1996).

[Hönes 93] F.Hönes and J.Lichter : "Text String Extraction within Mixed-Mode Documents", Proceedings of the Second International Conference on Document Analysis and Recognition, pp. 655–659 (1993).

[Ingber 92] L.Ingber and B.E.Rosen : "Genetic Algorithms and Very Fast Simulated Reannealing: A Comparison", Mathematical and Computer Modeling, vol. 16, no. 11, pp. 87–100 (1992).

[Kass 88] M.Kass, A.Witkin and D.Terzopoulos : "Snakes: Active Contour Models", International Journal of Computer Vision, vol. 1, no. 4, pp. 321–331 (1988).

[Kirkpatrick 83] S.Kirkpatrick, C.D.Gelatt and M.P.Vecchi : "Optimization by Simulated Annealing", Science, vol. 220, no. 4598, pp. 671–680 (1983).

[O'Gorman 93] L.O'Gorman : "The Document Spectrum for Page Layout Analysis", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 15, no. 11, pp. 1162–1173 (1993).

[Ohta 91] Y.Ohta, M.Watanabe and Y.Sumi : "Approaches to Parallel Computer Vision", IEICE Transactions, vol. E74, no. 2, pp. 417–426 (1991).

[Rose 93] K.Rose, E.Gurewitz and G.C.Fox : "Constrained Clustering as an Optimization Method", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 15, no. 8, pp. 785–794 (1993).

[Shimada 95] S.Shimada, K.Maruyama, A.Matsumoto and K.Hiraki : "Agent-based Parallel Recognition Method of Contour Lines", Proceedings of the Third International Conference on Document Analysis and Recognition, pp. 154–157 (1995).

[Tagliarini 91] G.A.Tagliarini, J.F.Christ and E.W.Page : "Optimization Using Neural Networks", IEEE Transactions on Computers, vol. 40, no. 12, pp. 1347–1358 (1991).

[Takizawa 94] K.Takizawa, D.Arita, M.Minoh and K.Ikeda : "Extraction of Inclined Character Strings from Unformed Document Images Using the Confidence Value of a Character Recognizer", IE-

ICE Transactions on Information and Systems, vol. E77-D, no. 7, pp. 839–845 (1994).

[Tan 91] H.L.Tan, S.B.Gelfand and E.J.Delp : "A Cost Minimization Approach to Edge Detection Using Simulated Annealing", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 14, no. 1, pp. 3–18 (1991).

[Toborg 91] S.T.Toborg and K.Hwang : "Cooperative Vision Integration Through Data-Parallel Neural Computations", IEEE Transactions on Computers, vol. 40, no. 12, pp. 1368–1379 (1991).

[Watanabe 90] M.Watanabe and Y.Ohta : "Cooperative Integration of Multiple Stereo Algorithms", Proceedings of the Third International Conference on Computer Vision, pp. 476–480 (1990).