千葉工業大学

博士学位論文

形態を指向した非線形画像処理に関する研究

令和元年 9 月

糸井 清晃

論文要旨

本論文では，形状の処理を目的とした画像処理に関して幾つかの手法を提案する．画像処理は，処理結果として出力される情報の種類によって，大きく二つに分けられる．画像が出力される処理を（狭義の）画像処理と呼び，一方，画像を表現するものの，画像形式ではない情報が出力される処理を画像解析（計測・認識）と呼ぶ．狭義の画像処理の一部は，画像を時間的または空間的に変化する色情報のディジタル信号と捉え，フィルタ処理として，もしくは，画素値の並びを行列と捉えれば行列演算として処理を実現することが可能である．

一方，被写体がなにであるかが意味を持つような処理は困難な場合が多い．このような場合には，被写体の形状を考慮した処理が必要になる．例えば，文書画像の処理であれば，文書を構成する主たる要素である文字の形状が重要であり，顔画像処理であれば，顔を構成する要素の形状およびその位置関係が重要である．このような処理を実現する方法には，画像が持つ形状の特徴を人間の知識に基づいて解釈し，対象となる画像または所望する画像処理に特化したアルゴリズムを構築して処理する方法と，機械学習によって，コンピュータを複数の事例を用いて学習させ，未知の入力を処理する方法がある．

本研究では，文書画像と顔画像に関する形状を考慮した画像処理について検討する．文書画像に関しては，紙媒体の文書を電子化する際に有用となる処理について検討している．文書を電子化しテキストデータに変換する処理は，文書を保存する際の省スペース化及び情報検索の効率化の面で重要である．

本論文では，文書を文字認識してテキストデータ化する際に障害となる，見出し文字列の背景に付けられた「地紋」と呼ばれる飾りを除去する方法と，印刷文書に書き込まれた注釈などから有益な情報を得るために，手書き文字を抽出する方法を提案している．

前者では，多数の見出し文字列を調査して形状の特徴によって 3 種類に分類し，分類されたそれぞれの地紋を除去するのに特化したアルゴリズムを検討して，個々のアルゴリズムを組み合わせることで全体のアルゴリズムを構成している．

後者では，形状を除去することができるモフォロジカルフィルタの一つであるクロージングフィルタ用いるが，その性能を向上させるために，フィルタを構成するダイレーションを Maxout 関数によって拡張し，Maxout フィルタネットワークを構成する．このフィルタネットワークを 3 層のニューラルネットワークとみなし，その係数を多数の事例によって学習させることによって，手書き文字だけを抽出できるようにしている．

学習方法は，従来の「確率的勾配降下法」「ミニバッチ学習」に加えて，「ミニバッチ＋最大値学習」を提案している．最大値学習とは，ニューラルネットワークの入力と出力を比較し，大きい方を改めて出力とする学習方法である．以上の三つの方法を用いて学習を行い，その性能を比較している．

学習のための事例となるデータは，手書きの注釈が書き込まれた印刷文書画像を入力データ，それに対応する手書きの注釈のみの画像を「理想手書き文字」と呼び，教師データとしている．

顔画像処理では，テレビ電話やテレビ会議システムなどの通信において，話者の表情を分析することによって低ビットレート化を実現する知的符号化を提案している．提案手法では，送信側のカメラによって動画として撮影された話者の表情を分析した結果を極少ない数の数値として送信し，受信側で受信した表情の分析結果の数値から送信話者の表情に似た表情を合成してディスプレイに表示する．

表情の分析では，話者の特徴的な表情を選んで基本表情とし，これらを階層型ニューラルネットワークで学習させておく．このニューラルネットワークによって，カメラで撮影された話者の任意の表情と基本表情との類似度を計算する．表情の合成は，表情の分析結果である類似度を基に計算される割合によって，基本表情をモーフィングを用いて合成することによって実現する．

提案した手法に関して実験を行い，次のように評価し，その有効性を確認している．

地紋の除去では，処理結果である地紋が除去された見出し文字列を目視と文字認識率によって評価している．目視による評価では，処理結果を 4 段階で評価し，概ね良好に除去できることを確認している．文字認識率による評価では，目視による評価それぞれに属する処理結果ごとに認識実験を行い，認識率が目視評価と同じ傾向となることを確認している．また，目視により良好と評価されたものの認識率は，元々地紋のない見出し文字の認識率とほぼ同等であることも確認している．

手書き文字の抽出では，目視と SNR によって評価している．SNR の平均は，確率的勾配降下法，ミニバッチ学習，ミニバッチ＋最大値学習の順に大きくなっており，提案した最大値学習の効果を確認している．また，目視による評価では，理想手書き文字と比較して遜色ない抽出結果になっていることを確認している．更に，従来の印刷文書からの手書き文字抽出手法では困難であった，印刷文字と手書き文字が重なっている場合の処理も可能となったことを確認している．

知的符号化では，受信側で出力される合成表情が，概ね送信側のカメラで撮影した話者

の表情の変化に追従して変化していることを確認している．また，提案手法による顔画像の送信に必要なデータ構造（ビット数）に関する検討も行い，画像のまま送信する場合に比べて，極超低ビットレートを実現できることを確認している．

目次

第 1 章　序論

1.1　背景

コンピュータ技術の向上と共に発展した画像処理は，人間社会を便利で快適なものにするための自動かつ知的なシステムの構築に貢献している．工業分野においては，ロボット制御やプリント基板をはじめとする各種部品の検査を可能としている．医療分野においては，CT や MRI など生体情報イメージングやその解析による画像診断を実現している．交通分野では，自動運転や高度道路交通システム（ITS）に関連する技術として期待されている．また，人間と機械のコミュニケーションのためのヒューマンマシンインタフェースの構築における重要な技術要素となっている．

画像処理は，その処理結果として出力される情報の種類によって，大きく二つに分類することができる [1]．画像が出力されるような処理を，（狭義の）画像処理と呼び，一方，画像を表現するものではあるが，画像形式とは異なる情報が出力される処理を画像解析（計測・認識）と呼ばれ，パターン認識，メディア理解，コンピュータビジョン [2] などの研究分野の基礎となる技術であり，冒頭述べた各分野への応用に大きく貢献していると言える．

狭義の画像処理は一般的に，画像を空間的に変化する色情報のディジタル信号として捉えることによって，フィルタ処理として実現することが可能であるし，また，画像を，個々の画素値を要素に持つ行列として捉えれば行列演算を用いて実現することが可能である．

このような処理は，画像を数値の並びとして扱うのが主体であり，その画像の被写体が持つ特徴は考慮されないので，被写体がなにであるかが意味を持つような処理は困難な場合が多い．このような場合には，被写体の形状を考慮した処理が必要になる．例えば，文書画像の処理であれば，文書を構成する主たる要素である文字の形状が重要であり，形状の特徴から文字を区別することによって文字認識が実現されており，他要素（図表など）の形状の特徴と比較することによって文書の領域分割が実現されている．また，顔画像処

理であれば，顔を構成する要素の形状が重要であり，その特徴から，画像中の顔を検出する処理や，顔による個人識別が実現されている．

これらの例では，主に画像解析（計測・認識）が用いられており，文書画像の例では，認識した文字を表すコードや文書のレイアウト構造などの情報が，また，顔画像の例では，検出した顔の位置や識別した人物に対応する ID などが出力される．このような画像解析（計測・認識）を実現する方法には，画像が持つ形状の特徴を人間の知識に基づいて解釈し，対象となる画像または所望する画像処理に特化したアルゴリズムを構築して処理する方法と，機械学習によって，コンピュータを複数の事例を用いて学習させ，未知の入力を処理する方法がある．

一方，文書画像処理や顔画像処理は共に歴史のある研究分野であり，その取り扱う対象である文書や顔は，我々の日常において，コミュニケーションもしくは情報伝達の重要な要素である．これらの画像を，やはり既に我々の日常に欠かせなくなった，パソコンをはじめとする電子機器において，さらに効率的に処理できるようにすることは意義のあることであると考える．そこで，本研究では，文書画像処理と顔画像処理における形態を指向した非線形画像処理に関して，1.2 節，1.3 節のような課題を検討した．

1.2 文書画像処理

文書画像処理は，紙メディア上のに記されたコンピュータには直接読み取れない情報を，コンピュータ上で取り扱うための研究分野であり，総じて画像として入力された文書から，コンテンツを抽出することが主な目的である [3]．文書を保存する際の省スペース化，情報の再利用及び検索の効率化の観点から進められている，紙媒体文書の電子化などは代表的な例といえる．

電子化の過程においては，まず画像として入力された文書を，テキスト・図表・写真など性質の異なるブロックに分割し，更にテキストブロックは，文字列ブロックから文字ブロックへ，また，図表ブロックは図表本体とキャプションへ，などのように細分化され，ツリー状にレイアウト構造が解析されていく．分割されたブロックの内，文字ブロックは，それらを文字認識してテキストデータに変換することにより，保存時のデータ量の削減や検索の効率化を実現している [4] [5] [6]．加えて，レイアウト構造に対して，論理構造が決定される．論理構造とは，タイトル，著者名，本文などのツリー構造である．本文は更に，章や節に分けられる．電子化後の文書保存において，レイアウト構造および論理

構造による分類は重要なものである．

一方でペーパレス化が推進されており，電子書籍による出版物を目にする機会は増えており，パソコン上で作成された資料を印刷することなく，電子ファイルのまま関係者に配布し，各自が所有するパソコンやタブレットなどの端末上で使用する機会も増えている．しかしながら，紙媒体による出版は未だに健在であり，紙資料も取り回しが楽であったり，注釈やメモなどの記入すること自体は，パソコンやタブレット上で行うより手軽であるなどの点で需要がある．既存の文書も含め，紙媒体の文書の電子化は今後も必要とされると考えられる．

本研究では，文書を電子化する際に有用となる二つの処理，見出し文字列の地紋除去と手書き文字の抽出に関して検討した．

1.2.1 見出し文字列の地紋除去

文書画像処理に関する手法の一つ目は，文書検索のキーワードとなり得る単語を有する見出しの文字認識の際に障害となる「地紋」と呼ばれる背景の飾りを除去する手法である．

紙媒体の文書の電子化において重要な技術要素である文字認識に関しては，各研究機関において様々な研究がなされてきた結果，現在では雑音のない活字であれば 100% 近い認識が可能となっている．

ところが，新聞や雑誌の見出しには文書検索する際のキーワードとなりうる単語が含まれているるにもかかわらず，文字の背景に「地紋」と呼ばれる，見出しであることを強調のための飾りが施されている場合が多く，そのため，活字であるにもかかわらず，文字認識が困難となることがある．このような背景に地紋のある見出し文字の復元及び認識に関しては，次のような研究が報告されている．

林らは，背景に地紋のある見出しをイメージスキャナで読み込む際，読み込み輝度を変えて 2 回読み込み，セルラニューラルネットによるノイズ除去を行う手法 [7] を提案しているが，見出し文字列の片側のみに地紋が存在している場合には除去できないという問題点が指摘されている．酒匂らは，模様の識別に有効なテクスチャ解析を用いた手法 [8] を提案しているが，処理コストがかかるという点や，テクスチャとしての特徴が明確でない場合には適応できない，という問題が指摘されている．岡本らは，文字より地紋の方が細いという性質を用い，膨張・収縮処理によって除去する手法 [9] を提案している．これは，収縮処理で文字部分が削られるが，膨張処理によって元に戻るため，文字より細い地紋の

みを除去でき，良好な結果が得られる．しかし，地紋と見出し文字の太さが拮抗しているか，逆転している場合には，除去が困難である．また，Liang らは，地紋の周期性を用いて推定・抽出し，原画像との モフォロジー演算を行うことで，除去する手法 [10] を提案しており，萩田らによって，背景雑音 (地紋) のある文字自体を認識するという試みもなされている [11][12]．これら除去手法は，いずれも地紋の繰り返し模様という性質に着目したものであり，性質の異なる地紋には対応できないので，これらを例外的に処理する必要が生じる．

これらのことから，一意な処理では，精度の高い地紋除去は困難であることがわかる．これを解決するためには，あらかじめ地紋をその形状の特徴に従って分類しておき，それぞれに適切な除去アルゴリズムを構築し，それらを適切に選択して適用することが望ましいと考えられる．しかし，このような地紋の分類はなされていない．

本研究では，多くの新聞に掲載されている地紋付き見出しを調査し，形状の特徴によって，3 種類に分類できることを示し，その分類結果に従い，それぞれの地紋に対する単純な除去アルゴリズムおよび，これらのアルゴリズムを効果的に組み合わせた，全体のアルゴリズム構成を提案する．提案手法による処理結果である，地紋が除去された見出し文字列の評価は，人間による目視と文字認識率によって行う．目視による評価では，処理結果を 4 段階で評価する．また，文字認識率による評価では，目視による評価それぞれに属する処理結果ごとに認識実験を行い，文字認識率によると目視による評価の傾向を確認する．本手法は，文献 [13] に対応しており，本論文では， **第 3 章**において詳述する．

1.2.2 手書き文字の抽出

文書画像処理に関する手法の二つ目は，紙媒体の文書に手書きされた注釈の抽出である．

紙媒体の文書には，利用者による書き込みが存在することがあるが，書き込まれるのは注釈などの有用な情報である場合が多く，文書の中から手書き文字を抽出することは，情報の再利用という観点からも，有益なことであると考えられる．手書きによって注釈が記入された印刷物などの文書画像から注釈部分を抽出することによって，注釈のテキストデータ化，編集が容易に実現できるようになり，注釈情報の有効活用が可能となる．また，注釈部分を除去することにより，原文書の復元も可能となる．

手書き文字の抽出もしくは手書き文字と印刷文字の判別を実現する方法に関しては，いくつかの手法が提案されているが [14][15][16][17]，いずれの手法も，文書画像の局所特徴

量を抽出し，その特徴の差異を活用して領域分割手法によって処理を実現しているので，手書き文字と印刷文書の活字が重なっている場合は，抽出が困難である．Nakai らは，この重なりの問題に対応したいるが [15]，原文書が必要な方法となっている．また，Umair らの方法 [17] は，特殊なカメラを必要とする方法であり，すでにスキャンされた文書画像に用いることはできない．

また，印刷物の活字を，前節の地紋ととらえ，これを除去する方法を検討すればよいとも考えられるが，活字と手書き文字が混在している状態からは，地紋のようにカテゴリに分類できるような特徴を抽出するのは困難である．そこで，このような特徴抽出を人間の知識に依存して抽出するような手法ではなく，機械学習に基づいた手法を検討する必要があると考えられる．

さて，本研究で対象としている文書画像は，グレースケール画像であるので，手書き文字及び印刷文書の文字のいずれも画像上の輝度面において窪みとして現れる．したがって，手書き文字と印刷文書の文字が混在している文書画像から，手書き文字のみを抽出するためには，印刷文書による輝度の窪みのみを識別して埋め，かつ，手書き文字による輝度の窪みを極力保存する必要がある．

2.1.2 節に示したとおり，画像の輝度面上の突起および窪みを，その形状によって除去する方法としてモフォロジカルフィルタが知られており，その基本処理である，ダイレーション（膨張）とエロージョン（浸食）を組み合わせることによって，例えば，輝度の突起を削ることができるオープニングや輝度の窪みを埋めることができるクロージングのようなフィルタを構成することができる．また，構造要素を変更することによって，様々な形状を除去することができるので，多種の画像処理が実現されている [18][19]．

さらに，このモフォロジカルフィルタの性能を向上させるために，エロージョンおよびダイレーションを，それぞれ Maxout 活性化関数で拡張した，Maxout フィルタネットワークが提案されており [20]，ニューラルネットワークにおける，Maxout 活性化関数 [21] によって構成された，畳み込み型ネットワークの一層がダイレーションの拡張に相当することを示し，モフォロジカルフィルタの一つであるクロージングフィルタを拡張し，雑音除去性能を向上させている．また，ネットワークのパラメータをクロージングフィルタから与えることで良好な結果を得ている [20]．

そこで本研究では，印刷文書の輝度の窪みのみを識別して除去するために，モフォロジカルクロージングフィルタを拡張することで得られた，Maxout フィルタネットワークを用いた文書画像中の手書き文字の抽出方法を提案する．

手書き文字の抽出処理では，手書き文字を構成する黒画素の画素値と紙部分を構成する白画素の画素値を保存したまま，活字を構成する黒画素の画素値のみを紙の色である白に変化させる．したがってすべての画素において，手書き文字と活字が混在する文書画像の画素値と手書き文字のみの画像の画素値は等しいか，後者の方が大きくなる．そこで，各層間の重み付き結合のみ考慮した単純な層構造に加え，層構造ネットワークの出力画像と入力画像の各画素を比較して，大きい方を出力するネットワーク構造を提案する．本論文では，このネットワーク構造を有するフィルタを「最大値学習フィルタ」と呼び，その学習を「最大値学習」と呼ぶことにする．

実験に用いる学習方法には，2.2.4 節で示した従来の確率的勾配降下法やミニバッチ学習に加えて，本研究で提案する最大値学習フィルタをミニバッチ学習によって学習させる「ミニバッチ＋最大値学習」を採用する．

ミニバッチに含まれる学習サンプルの選択方法としては，複数の記入者に対応するために，記入者ごとの学習サンプルをまとめてミニバッチを構成する．学習のための事例となる学習サンプルは，手書きの注釈が書き込まれた印刷文書画像を入力データ，それに対応する手書きの注釈のみの画像を「理想手書き文字」と呼び，これを教師データとする．以上の三つの方法を用いて学習を行い，その性能を比較する．

実験では，使用されるフォントの種類が少ない学術論文を印刷文章の対象として，Maxout フィルタネットワークの学習及び手書き文字の抽出を行い，目視と SNR によって評価する．SNR の平均は，確率的勾配降下法，ミニバッチ学習，ミニバッチ＋最大値学習によるネットワークによる処理結果を比較し，提案した最大値学習の効果を確認する．また，目視による評価では，理想手書き文字と比較して遜色ない抽出結果になっていることを確認する．更に，従来の印刷文書からの手書き文字抽出手法では困難であった，印刷文字と手書き文字が重なっている場合の処理に関しても評価する．本手法は，文献 [22] に対応しており，本論文では， **第 4 章**において詳述する．

1.3 顔画像処理

顔画像処理には，画像中からの顔の検出及びその追跡，顔認識による人物同定，表情の推定といった，コンピュータに顔を理解させる研究分野，顔の各種補正処理，表情合成などの分野がある．我々人間が，顔によって相手を判別したり，コミュニケーションの際にその表情から相手の感情などを読み取っていることからも顔がもたらす情報は重要である

と言える．

コミュニケーションにおいて，相手の個人の特徴だけでなく，その表情によって心理状態など，声だけでは伝わりにくい様々な情報を伝えるという役割を担っている．遠隔地の人間とこのようなコミュニケーションをとるために提案されたテレビ電話・テレビ会議システムにおいては，映像が伝えるべき重要な情報は，話者の表情ということになる．ところが，映像の圧縮技術が進歩した現在においても，やはり，映像情報の伝達は，コストのかかる処理であり，圧縮率を高めれば，画質の劣化を招き，画質を優先させれば低圧縮率となる，いわゆる圧縮率と画質のトレードオフの関係が存在している．

しかし，テレビ電話・テレビ会議システムなどにおいては，話者がどういう表情をしているかがわかることが重要であるので，送信側の話者の表情を分析し，いくつかの数値で表現し，その数値を逐次送信することができ，受信側では，受信された分析結果の数値によって送信話者の表情を合成することによって，これらのシステムにおける低ビットレート通信が可能になる [23].

このような通信は「知的通信」と呼ばれ，それによって送受される情報の符号化を「知的符号化」と呼ばれる．映像を不規則信号と見做し，波形レベルでの統計的な冗長を削減するような符号化とは異なり，画像分析やパターン認識の技術を利用した符号化方式であり，対象となる被写体の形状に関するモデルを送信側，受信側の共通知識として保持しておき，その変形情報のみが伝送され，受信側で変形情報を用いてモデルを変形・合成させる．送信側において，映像の分析を行い，受信側において合成を行うことから，「分析合成符号化」とも呼ばれる．

本研究では，ニューラルネットワークによる表情の分析とモーフィングを用いた表情の合成を基礎とした，顔画像の知的符号化手法を提案する [24]．送信側の表情分析では，話者の特徴的な表情を選んで基本表情とし，これらを階層型ニューラルネットワークで学習させておく．このニューラルネットワークによって，カメラで撮影された話者の任意の表情と基本表情との類似度を計算し，その結果を送信する．受信側では，送信話者の表情の分析結果である類似度を基に，モーフィングを用いて基本表情を合成する．

実験によって，受信側で出力される合成表情が，概ね送信側のカメラで撮影した話者の表情の変化に追従して変化することを確認する．また，提案手法による顔画像の送信に必要なデータ構造（ビット数）に関する検討も行い，画像のまま送信する場合に比べて，極超低ビットレートを実現できることを確認する．本手法は，文献 [24] に対応しており，本論文では， **第 5 章**において詳述する．

第 2 章　ニューラルネットワークと非線形画像処理フィルタ

2.1　非線形画像処理フィルタ

非線形画像処理フィルタとは，画像の領域に基づく濃淡変換を実現する空間フィルタのうち，入力データとフィルタ重み係数の畳み込み演算では実現できないもののことである．つまり，出力画像 f' の座標 $\boldsymbol{z} = (z_1, z_2)$（ただし，$z_1, z_2 \in \mathbb{Z}$）における画素値が，入力画像 f の座標 $\boldsymbol{z}$ の画素値 $f_{\boldsymbol{z}}$ と，フィルタ重み係数 $w_{\boldsymbol{y}}$ 及びその $\boldsymbol{z}$ との相対座標集合 $\mathcal{S}$ によって，

$$f'_{\boldsymbol{z}} = \sum_{y \in \mathcal{S}} w_{\boldsymbol{y}} f_{\boldsymbol{z}+\boldsymbol{y}} \tag{2.1}$$

のように定義され，$w_{\boldsymbol{y}}$ に与える値によって様々な処理ができる線形画像処理フィルタとは異なり，領域 $\boldsymbol{z} + \boldsymbol{y}$ 内の画素値の順序統計に基づいて画素を選択し出力するフィルタである．これらのうち，領域内の画素値を大きい順（もしくは小さい順）に並べたときの中央値を出力するのが「メディアンフィルタ」であり，領域内の画素値の最大値もしくは最小値を出力するのが「モフォロジカルフィルタ」における「ダイレーション」と「エロージョン」である．これらのフィルタに関して，それぞれ 2.1.1 節，2.1.2 節において詳細を述べる．

2.1.1　メディアンフィルタ

メディアンフィルタの処理手順を図 2.1 に示す．まず，(a) において青く囲まれた注目画素及びその周辺の画素（この例では赤く囲まれた 3×3 画素として説明する）を抽出する．抽出された 9 画素の画素値をソート処理する（図には，昇順，降順とも示した）．並び替えられた画素値の先頭から 5 番目の画素値「14」を出力する．原画像の注目画素の画

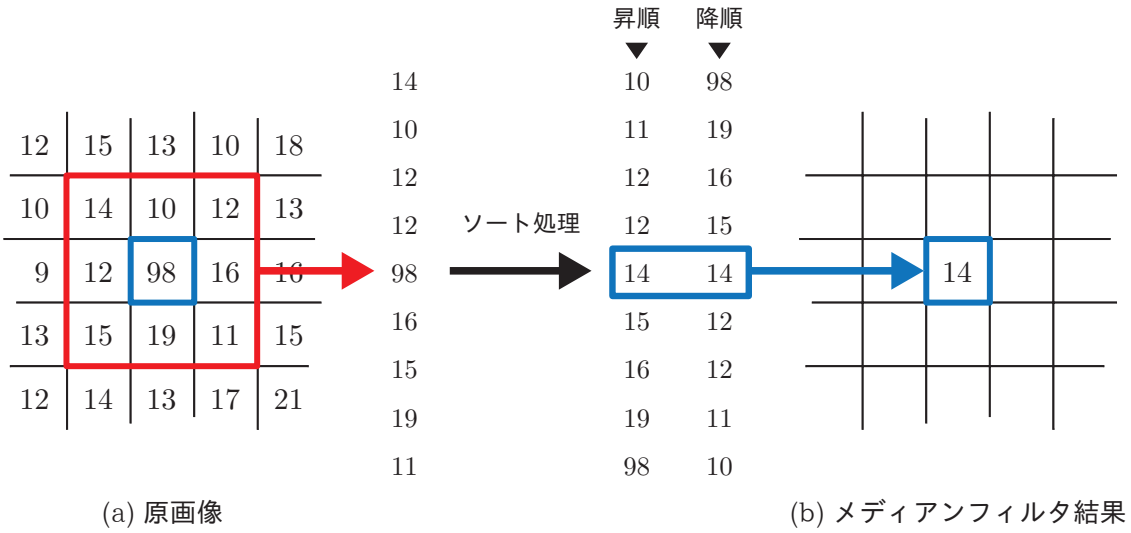


図 2.1　メディアンフィルタの処理手順

素値「98」に対して，出力値が「14」であることから，メディアンフィルタによって，周囲の画素値に比べて突出した画素値を取り除くことができることがわかる．

図 2.2 は，原画像 (a) に対しインパルス性雑音を重畳した画像 (b) をメディアンフィルタによって処理した結果 (c) である．比較のために示した線形画像処理フィルタに属する移動平均フィルタによって処理した結果 (d) と比較すると，(d) では，雑音が薄く残ってしまっているうえに，平均化処理の影響でエッジが保存されずぼけてしまっているが，(c) では，雑音は除去され，エッジは保存されていることが分かる．なお，この例では，両フィルタとも，座標 $\boldsymbol{z}$ の画素とその周辺の 3×3 画素領域内の 9 画素を用いて処理しており，これは，線形画像処理フィルタにおいては，式 (2.1) の $\mathcal{S}$ を

$$\mathcal{S} = \left\{ \begin{array}{ccc} (-1,-1) & (0,-1) & (1,-1) \\ (-1,0) & (0,0) & (1,0) \\ (-1,1) & (0,1) & (1,1) \end{array} \right\} \tag{2.2}$$

とした場合に相当する．

(a) 原画像

(b) インパルス性雑音重畳画像

(c) メディアンフィルタ

(d) 移動平均フィルタ

図 2.2　メディアンフィルタの処理例

2.1.2　モフォロジカルフィルタ

モフォロジカルフィルタは，画像の輝度面上の突起および窪みを，その形状によって除去するフィルタとして知られている．基本となる処理，ダイレーション（膨張）とエロージョン（浸食）を組み合わせることによって，画像中に存在する輝度の形状を除去するフィルタを構成することができる．

画像 f の座標 $\boldsymbol{z}$ における画素値を $f_{\boldsymbol{z}}$ とすると，ダイレーションは

$$d_s \circ f_{\boldsymbol{z}} = \bigvee_{\boldsymbol{y} \in \mathcal{S}} f_{\boldsymbol{z}+\boldsymbol{y}} + s_{\boldsymbol{y}} \tag{2.3}$$

と定義され，エロージョンは

$$e_s \circ f_{\boldsymbol{z}} = \bigwedge_{\boldsymbol{y} \in \mathcal{S}} f_{\boldsymbol{z}-\boldsymbol{y}} - s_{-y} \tag{2.4}$$

と定義される．ここで，$\bigvee$ は数値の集合の最大値を，$\bigwedge$ は最小値をそれぞれ表す．$\mathcal{S}$ は構造要素の座標の集合を，$s_{\boldsymbol{y}}$ は構造要素の各座標におけるバイアスを示す．ダイレーションとエロージョンの処理手順を図 2.3 に示す．(a) の青く囲まれた注目画素を中心に，構造要素によって決定される $\boldsymbol{y} \in \mathcal{S}$ の領域内に存在する画素に対して，ダイレーションでは最大画素値が，エロージョンでは最小画素値が選択されている．

ダイレーションとエロージョンを組み合わせることで構成できるフィルタに，輝度の突起を削ることができるオープニングフィルタや輝度の窪みを埋めることができるクロージングフィルタがある．オープニングフィルタは，エロージョンを適用することで得られた画像に対してダイレーションを適用する処理であり，

$$o_s \circ f_{\boldsymbol{z}} = d_s \circ e_s \circ f_{\boldsymbol{z}} \tag{2.5}$$

と定義される．クロージングフィルタは，ダイレーションを適用することで得られた画像に対してエロージョンを適用する処理であり，

$$c_s \circ f_{\boldsymbol{z}} = e_s \circ d_s \circ f_{\boldsymbol{z}} \tag{2.6}$$

と定義される．

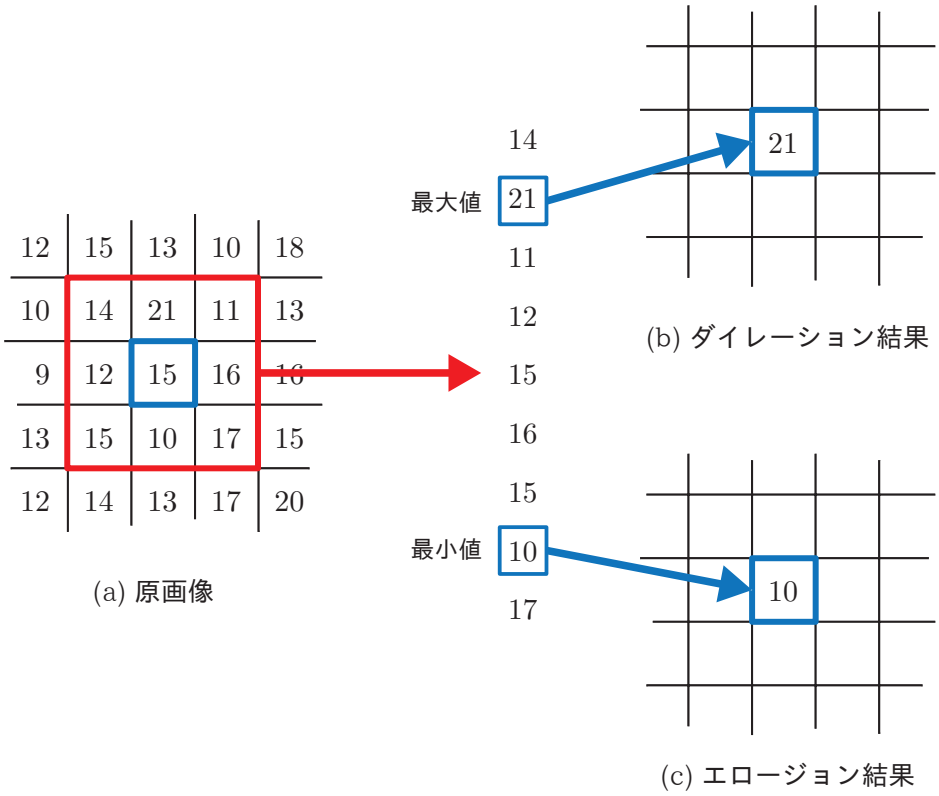


図 2.3　ダイレーション・エロージョンの処理手順

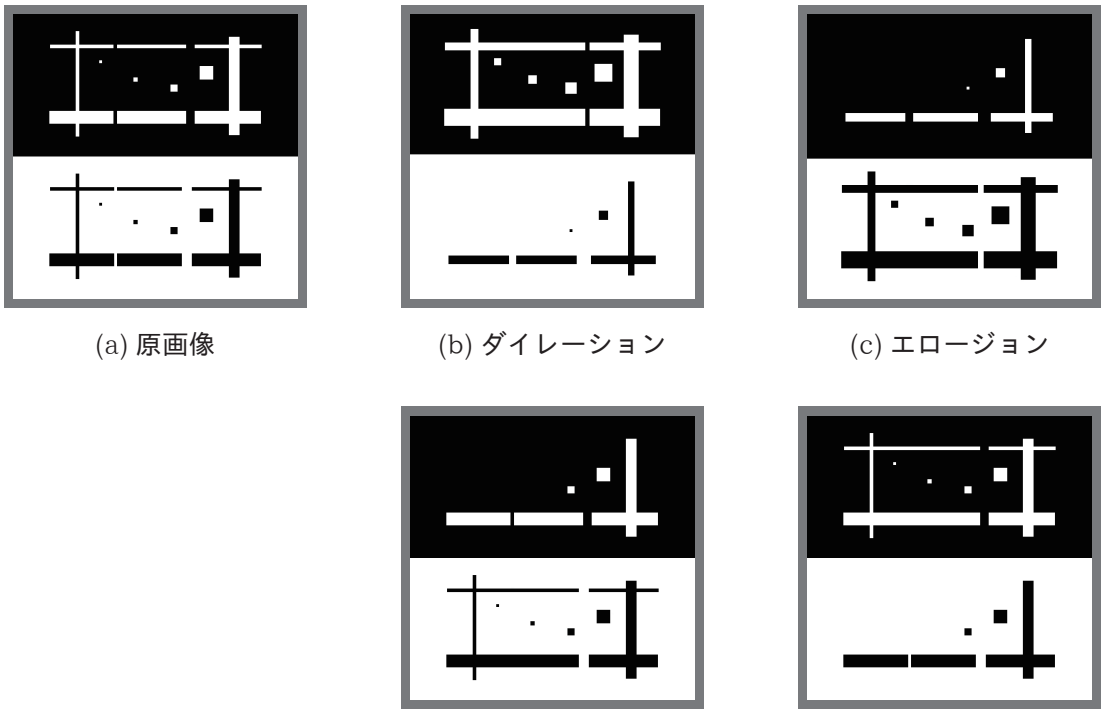
ここで，構造要素として

$$\mathcal{S} = \left\{ \begin{array}{ccc} (-3,-3) & \cdots & (3,-3) \\ \vdots & (0,0) & \vdots \\ (-3,3) & \cdots & (3,3) \end{array} \right\} \tag{2.7}$$

$$s_{\boldsymbol{y}} = 0, \forall \boldsymbol{y} \in \mathcal{S}$$

を使用した場合のモフォロジー画像処理結果の例を図 2.4 に示す．原画像は，上半分が黒の背景に，白い太さの異なる線と大きさの異なる点が描かれており，線には，細くまたは太く途切れている個所がある．下半分は，上半分を階調反転した画像である．

ダイレーションの結果 (b) では，白い線は太く，点は大きくなっており，細い黒線や小さい黒点は消えてしまっている．また，白い線の途切れがつながっている．これは，原画像において黒画素であっても，その周囲 7×7 画素に白画素が存在すれば，その画素値が最大値として出力されるためである．



(a) 原画像　(b) ダイレーション　(c) エロージョン

(d) オープニング　(e) クロージング

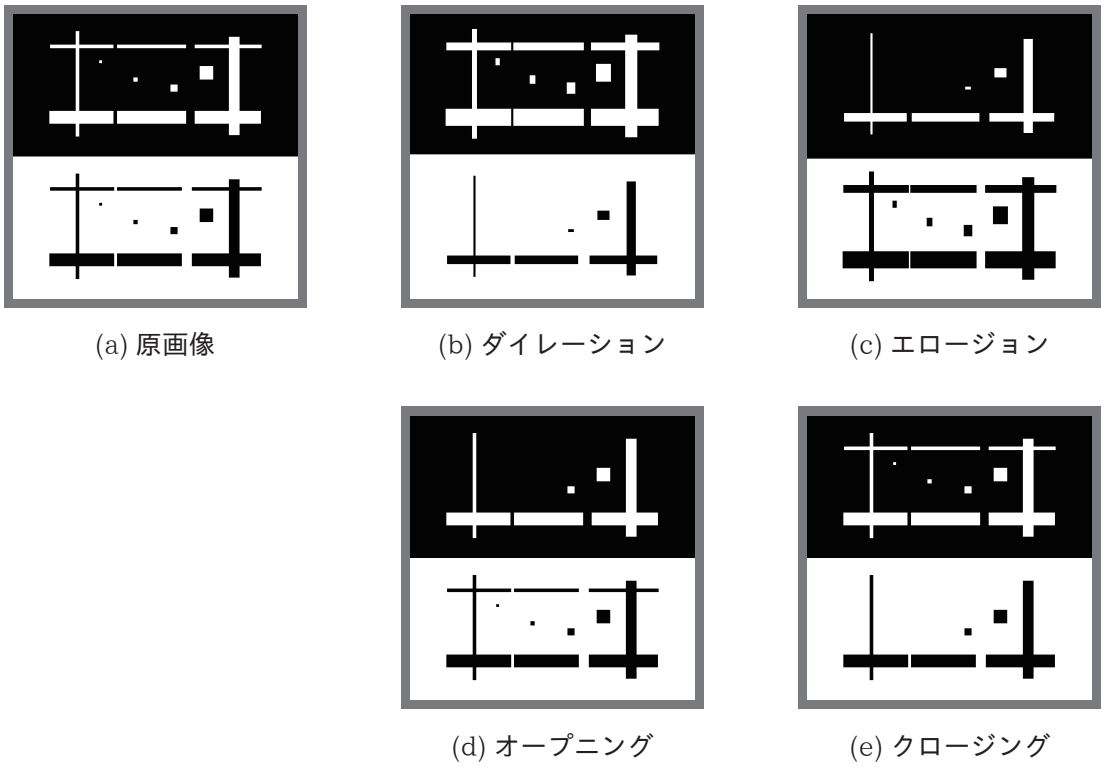
図 2.4　モフォロジー画像処理の例（7 × 7 構造要素）

エロージョンの結果 (c) では，ダイレーションとは逆に最小値を出力するので，白い線や点と黒い線や点の関係がダイレーションとは逆になっている．

エロージョン，ダイレーションの順に処理されるオープニングの結果 (d) では，エロージョンを適用した際に，消えてしまった白い線や点は復元されないが，その他の部分は復元される．また，エロージョンでつながった黒線の途切れも復元されずつながったままになっている．

クロージングの結果 (e) では，オープニングとは逆の処理が施されるので，白黒の関係がオープニングの結果とは逆の関係になっている．

ところで，モフォロジカルフィルターでは，構造要素を変更することによって，様々な



(a) 原画像　(b) ダイレーション　(c) エロージョン

(d) オープニング　(e) クロージング

図 2.5　モフォロジー画像処理の例（3 × 7 構造要素）

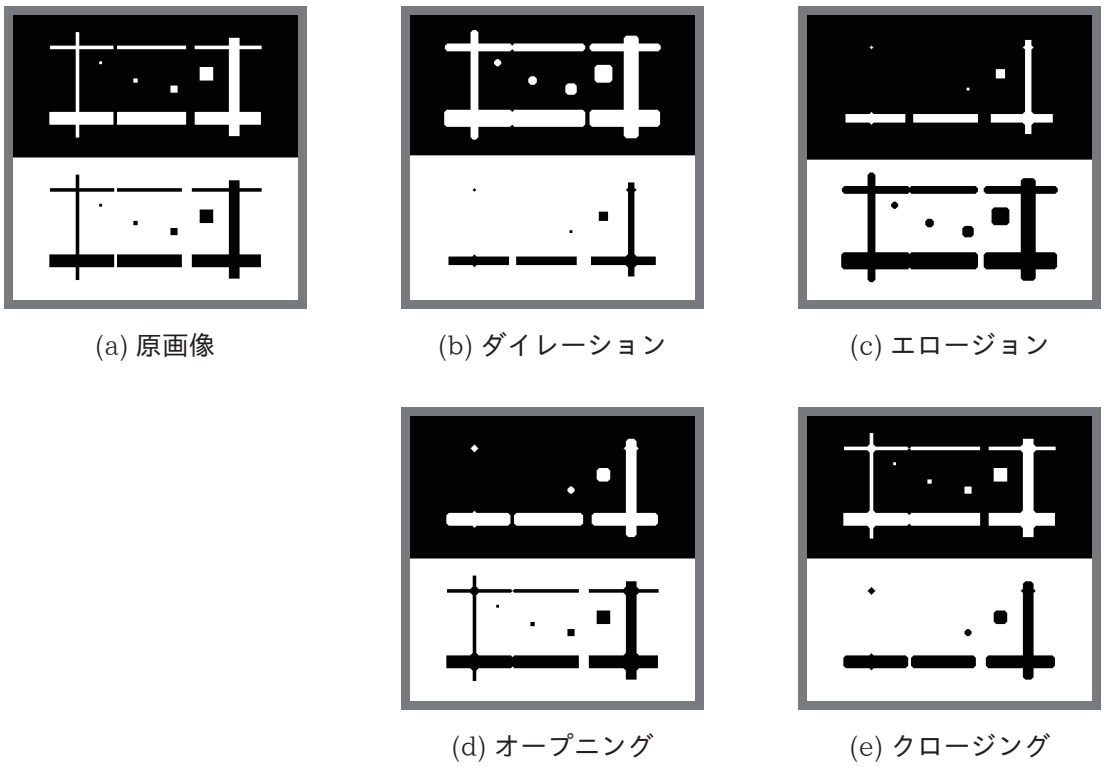
形状を除去することができることが知られている．例えば，構造要素として

$$\mathcal{S} = \left\{ \begin{array}{ccc} (-1,-3) & \cdots & (1,-3) \\ \vdots & (0,0) & \vdots \\ (-1,3) & \cdots & (1,3) \end{array} \right\} \tag{2.8}$$

$$s_{\boldsymbol{y}} = 0, \forall \boldsymbol{y} \in \mathcal{S}$$

を使用した場合の処理結果の例を図 2.5 に示す．式 (2.7) の構造要素では 7 × 7 画素の正方形の範囲で処理されたが，式 (2.8) の構造要素では 3 × 7 画素の縦長長方形の範囲で処理される．

図 2.4 に示した処理結果と比較して特徴的な差異としては，ダイレーション，エロージョンによって消えてしまった細い線のうち，左の縦線が消えずに残ること，途切れた線が途切れたままになっていること，正方形として描かれている点の縦横比が変わっていることが挙げられる．線の途切れが維持されているため，オープニングの結果 (d) における下半分とクロージングの結果 (e) における上半分は，それぞれ原画像の下半分と上半分を



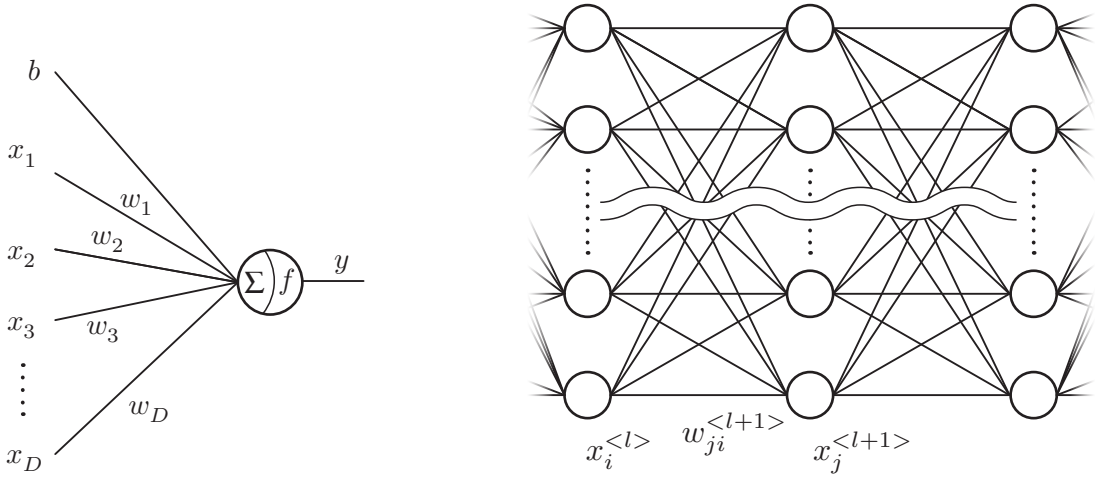
(a) 原画像　(b) ダイレーション　(c) エロージョン

(d) オープニング　(e) クロージング

図 2.6　モフォロジー画像処理の例（菱形構造要素）

復元している．構造要素が縦長になったことで，上下方向への処理の影響は同じでも，左右方向への処理の影響が減少たため，縦長の領域は維持されやすくなったといえる．

別の例として，菱形構造要素を使用した場合の処理結果の例を図 2.6 に示す．線や点の角であった部分に菱形の特徴が現れていることがわかる．



(a) ニューロンユニット　　(b) 全結合型ネットワーク

図 2.7　ニューラルネットワーク概要

2.2　ニューラルネットワーク

ニューラルネットワークは，脳の神経細胞（ニューロン）のネットワークを数理モデル化した情報処理の仕組みであり，画像処理の分野においては，画像に含まれる対象（人，物）の検知，など主に認識処理に応用されている．脳のネットワークは，ニューロンから伸びる樹状突起と索軸によるニューロン同士の相互接続を介した電気信号による情報伝達によって構成されている．本節では，階層型ニューラルネットワークに関して，ニューロンの接続形態に相当するネットワーク構造（全結合型，畳み込み型），ニューロン間の電気信号の制御に相当する活性化関数（シグモイド関数，正規化線形関数，Maxout 関数），およびニューラルネットワークの学習方法としてバックプロパゲーション学習に関して述べる [25].

2.2.1　ネットワーク構造

ニューロンユニットの概略を図 2.7(a) に示す．ユニットは，別のユニットからの出力 $x_1, x_2, \ldots, x_D$ に，重み $w_1, w_2, \ldots, w_D$ をそれぞれ掛けた値の和にバイアス b を加えた値を入力とし，その値より求められる活性化関数 $f(x)$ の値を出力する．(b) に示す階層型で全結合型ネットワークの場合には，別のユニットとは，前の層の全ユニットである．

(b) では，l 層ユニット i に出力が，$l+1$ 層のユニット j に，重み $w_{ji}^{<l+1>}$ で結合している様子を示している．このようなニューラルネットワークのある層の N_U 個のユニットに D 次元のデータ

$$\boldsymbol{x} = (x_1, x_2, \ldots, x_D)^\top \tag{2.9}$$

が入力されるとする．ここで，$\top$ は転置を表す．$\boldsymbol{x}$ の各要素は，前の層のそれぞれのユニットの出力値と考えてよい．各入力の各ユニットに対する重みを $N_U \times D$ の行列

$$\boldsymbol{W} = \begin{bmatrix} w_{11} & \cdots & w_{1D} \\ & \ddots & \\ \vdots & w_{ji} & \vdots \\ & \ddots & \\ w_{N_U 1} & \cdots & w_{N_U D} \end{bmatrix}, \tag{2.10}$$

バイアスを N_U 次元の列ベクトル

$$\boldsymbol{b} = (b_1, b_2, \ldots, b_{N_U})^\top \tag{2.11}$$

とすれば，入力の重み付き和は，N_U 次元の列ベクトル

$$\boldsymbol{u} = \boldsymbol{W}\boldsymbol{x} + \boldsymbol{b} \tag{2.12}$$

である．各ユニットで $\boldsymbol{u}$ は，活性化関数 $f(\boldsymbol{u})$ によって処理され，N_U 個のユニットからの N_U 次元ベクトルの出力

$$\boldsymbol{y} = f(\boldsymbol{u}) = f(\boldsymbol{W}\boldsymbol{x} + \boldsymbol{b}) \tag{2.13}$$

を得る．

次に，前後の層との繋がりに関して述べる．式 (2.13) の出力 $\boldsymbol{y}$ は，次の層への入力となるので，$\boldsymbol{x}$ を l 層への入力，l 層の出力である $\boldsymbol{y}$ を $l+1$ 層への入力として，$\boldsymbol{x}^{<l+1>}$ と書き直すと，

$$\boldsymbol{x}^{<l+1>} = f(\boldsymbol{W}^{<l+1>}\boldsymbol{x}^{<l>} + \boldsymbol{b}^{<l+1>}) \tag{2.14}$$

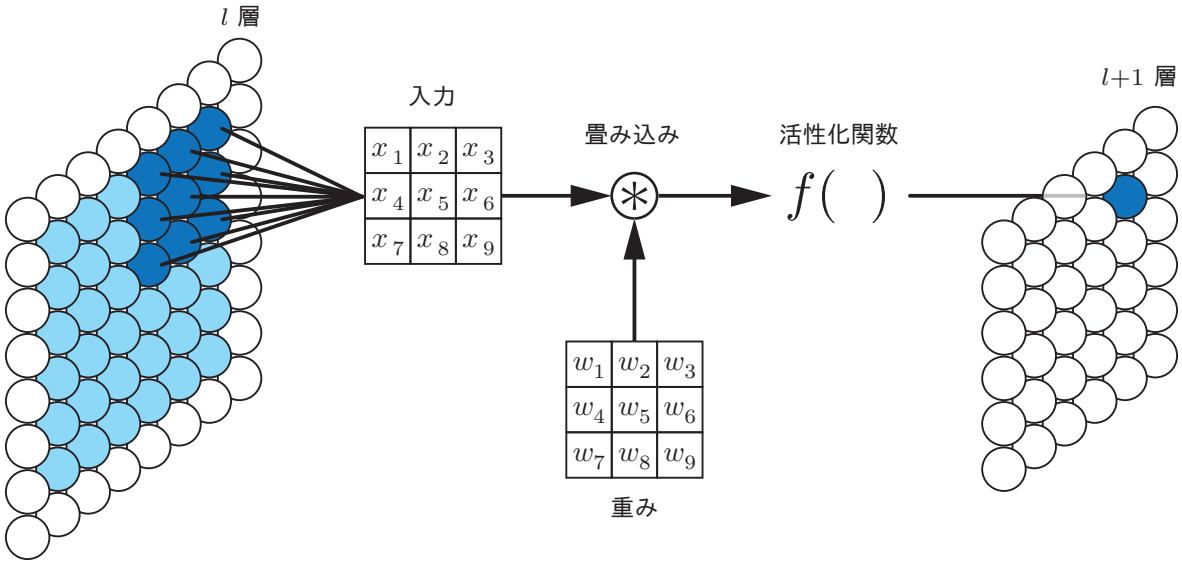


図 2.8　畳み込み層

または，成分を表示して，

$$x_j^{<l+1>} = f(\sum_i w_{ji}^{<l+1>} x_i^{<l>} + b_i^{<l+1>}) \tag{2.15}$$

となる．式 (2.15) において，i は l 層のユニット番号，j は $l+1$ 層のユニット番号である．$w_{ji}^{<l>}$ は l 層のユニット i の出力が，$l+1$ 層のユニット j へ入力される際の重みである．L 層のニューラルネットワークの順伝搬では，式 (2.13) または式 (2.15) を $l=1$ から $l=L$ まで繰り返す．

次に，畳み込み型ネットワークについて述べる．これは，図 2.8 に示す畳み込み層を有するネットワークである．畳み込み層では，$l+1$ 層のユニットは，前の層である l 層のユニットの内，特定のユニットのみに結合されている．図では，青丸で示した 3×3 ユニットであり，これらのユニットの出力値 $(x_1, x_2, \ldots, x_9)$ が入力として，重み $(w_1, w_2, \ldots, w_9)$ と畳み込みされ，その結果が活性化関数で処理されて出力される．この，l 層の 3×3 ユニットは，縦横方向に 1 ユニットずつスライドさせながら選ばれ，それに対応する $l+1$ 層のユニット値が計算される．

この処理は，l 層の各ユニットを，画像の画素に置き換えて考えると，式 (2.1) で示したフィルタ処理と等価である．つまり，画像を扱うニューラルネットにおいて，畳み込み層とはフィルタ処理によって画像の特徴を抽出する役割を担っている層であり，重み自体も学習によって決定されるので，所望のネットワーク出力を得るのに必要な画像特徴を抽出

するフィルタが自動的に決定される特徴がある．

図では，重みは 3×3 の 1 組を示してあるが，これを複数組設定することで，特徴抽出のフィルタを増やすことができることも特徴である．この場合には，$l+1$ 層のユニット数も図のような 6×6 ユニットの組が複数必要になる．

畳み込み層の処理では，画像のフィルタ処理の際に問題になる縁の処理と同じ問題が発生する．図の l 層の 8×8 ユニットの内，白く示したユニットを中心とする 3×3 ユニットを処理しようとする場合に，対応するユニットが存在しないため，このままでは処理できないことになるので，図では，それを除いた水色で示す 6×6 ユニットのみ考慮している．

これに対応するには，l 層の周囲にもユニットが存在していて何らかの出力があると仮定して処理することによって，$l+1$ 層を 8×8 ユニットで処理できるようにする．このような細工をパディングといい，画像のフィルタ処理でも使用されている．l 層の周囲に仮定したユニットの出力は，0 を設定したり，l 層の実ユニットと同じユニットが上下左右で折り返して存在していると仮定して設定したりする．0 を設定するのが簡単であるが，画像処理として見た場合には，周辺部の出力値が小さくなってしまう恐れがあるので，後者のように，何らかの値を設定するほうが良いと言える．

2.2.2　活性化関数

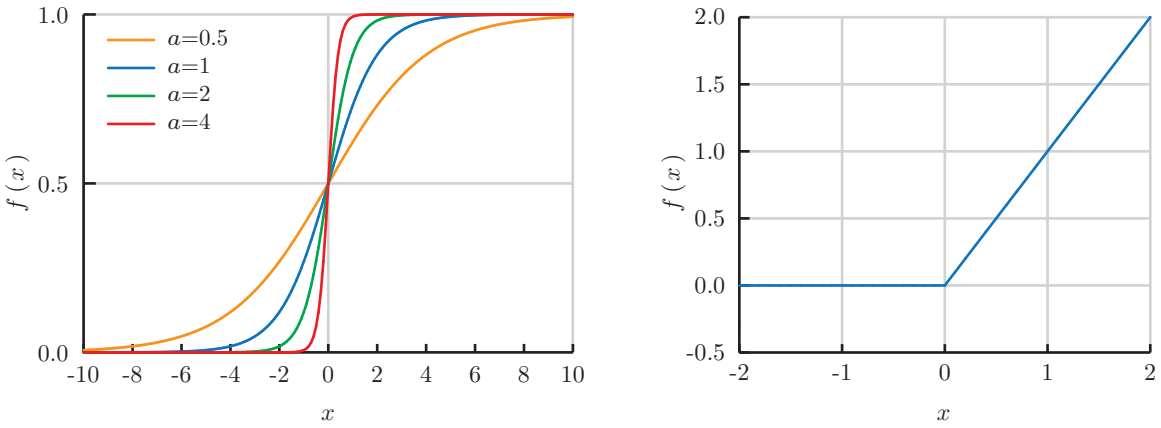
活性化関数は，2.2.1 節で述べたように，別のニューロンからの入力の重み付き和に対応した出力をする関数である．活性化関数として一般に使用されるのは，単調増加する非線形関数である．本節では，その例を幾つか示す．

シグモイド関数は，バックプロパゲーション学習を用いてユニット結合重みを更新する際に，微分可能な関数であることが必要となることから導入された関数である．入力を x とすると，

$$f(x) = \frac{1}{1+e^{-ax}} \tag{2.16}$$

のように表され，その形状は図 2.9(a) に示すとおりである．ここで a はゲインと呼ばれ，変化させると図のように関数の形状が変化し，大きくするほど階段関数のような形状に近づく．$a = 1$ の場合を，標準シグモイド関数と呼ぶ．

正規化線形関数は，深層学習の分野で盛んに用いられており，0 を出力するか，入力値



(a) シグモイド関数　　(b) 正規化線形関数

図 2.9　活性化関数の例

をそのまま出力するかの単純な関数であるため計算量は少なく，学習の進みが早く，学習結果も良好である場合が多いという特徴がある．入力を x とすると，

$$f(x) = \bigvee\{x, 0\} \tag{2.17}$$

のように表され，図 2.9(b) のような形状をしている．ここで $\bigvee$ は，{} 内の数値集合の最大値を意味しており，この場合は，x と 0 の大きいほうが関数値となる．正規化線形関数を活性化関数に持つニューロンユニットを ReLU（Rectified Linear Unit，正規化線形ユニット）と呼ぶ．

Maxout 関数は，ニューラルネットワークの各ニューロンユニットへの入力を，N 次元ベクトル $\boldsymbol{x} = (x_1, x_2 \cdots x_N)$ で表した場合，出力 $f(\boldsymbol{x})$ は，K 組の荷重パラメータ $w_{n,k}$ とバイアス b_k を用いて，

$$f(\boldsymbol{x}) = \bigvee_{k=1\cdots K} \left\{ \left(\sum_{n=1}^{N} w_{n,k} x_n \right) + b_k \right\} \tag{2.18}$$

によって表される．最大値関数によって，Maxout 関数のエピグラフは N 次元空間における K 個の超平面のエピグラフの積集合となり，図 2.10 に示すように，任意の凸関数を K 個の超平面で近似することができる．ReLU などと較べると，学習するパラメータの数が K 倍となる反面，自由度が高くなり，種々の問題へ適用されている [21]．また，バックプロパゲーション学習の際に，いずれかの線形和の出力が現れるために，ReLU と比較

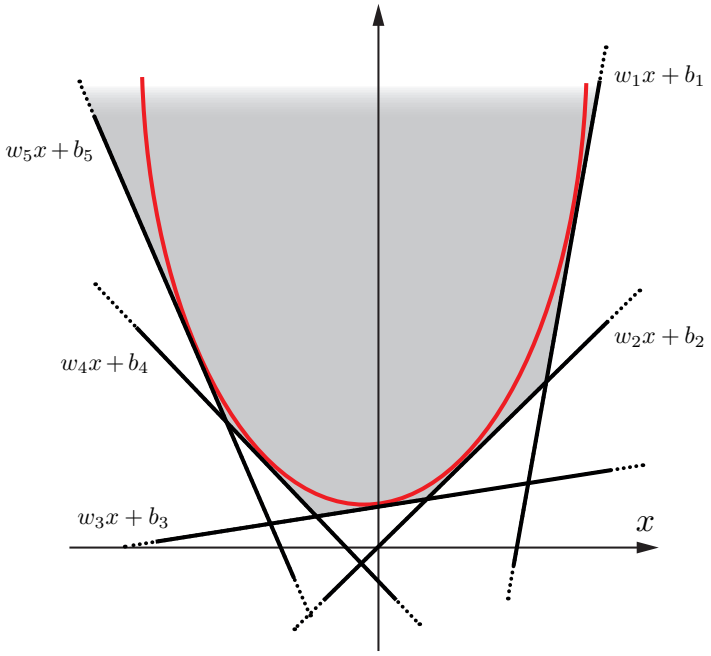


図 2.10　Maxout の例（$K=5, N=1$）

して勾配消失が起こりにくい点も利点の一つである．

2.2.3　バックプロパゲーション学習

ニューラルネットワークの入力層への学習データ入力 $\boldsymbol{x}_n$ $(n=1,\ldots,N_T)$ に対する出力を $\boldsymbol{y}_n$，所望する出力値である教師データを $\boldsymbol{t}_n$ とし，N_T 個の出力値の誤差を

$$E(\boldsymbol{W}) = \frac{1}{2}\sum_{n=1}^{N_T}(\boldsymbol{y}_n - \boldsymbol{t}_n)^2 \tag{2.19}$$

と定義する．これを，誤差関数または損失関数と呼ぶ．$\boldsymbol{W}$ はニューラルネットワークの全重みであり，$\boldsymbol{y}_n$ は $\boldsymbol{W}$ に依存する．ニューラルネットワークの学習とは，$\boldsymbol{W}$ を適切な値にして，損失関数の値 $E(\boldsymbol{W})$ を小さくすることである．これを実現する方法として，勾配法と呼ばれる反復計算がある．損失関数 $E(\boldsymbol{W})$ の勾配（偏微分値）を用い，k 回目の反復の重みを $\boldsymbol{W}_k$ として，

$$\boldsymbol{W}_{k+1} = \boldsymbol{W}_k - \varepsilon\frac{\partial E(\boldsymbol{W}_k)}{\partial \boldsymbol{W}_k} \tag{2.20}$$

のように $\boldsymbol{W}_k$ を反復更新する．ここで，ε を学習率と呼び，1 回の反復での更新量を決定するパラメータである．

重みを，式 (2.20) によって更新する際に，それぞれの重みによる損失関数の偏微分を求める必要があるが，ネットワークの計算は，式 (2.15) からもわかるように，$\boldsymbol{x}$ を介した，活性化関数の入れ子になるため，計算が困難である．この問題を解決したのがバックプロパゲーション法（誤差逆伝搬法）である．

以下ではバックプロパゲーション法について述べるが，説明の簡略化のため，重み $\boldsymbol{W}$ とバイアス $\boldsymbol{b}$ をまとめた表記を採用する．まず，入力ベクトル $\boldsymbol{x}$ は，x_i を x_{i+1} で置き換え，先頭に改めて $x_1(=1)$ を追加して，

$$\boldsymbol{x} = (x_1, x_2, \ldots, x_{D+1})^\top \tag{2.21}$$

とする．これによって，常に 1 を出力するユニットが追加されたことになる．一方，$b_1, \ldots, b_i, \ldots, b_{N_U}$ を $w_{11}, \ldots, w_{j1}, \ldots, w_{N_U 1}$ で置き換え，$\boldsymbol{W}$ の元々の要素 w_{ji} を w_{ji+1} で置き換えることによって，改めて

$$\boldsymbol{W} = \begin{bmatrix} w_{11} & w_{12} & \cdots & w_{1D+1} \\ \vdots & & \ddots & \\ w_{j1} & \vdots & w_{ji} & \vdots \\ \vdots & & \ddots & \\ w_{N_U 1} & w_{N_U 2} & \cdots & w_{N_U D+1} \end{bmatrix} \tag{2.22}$$

とする．式 (2.21)，式 (2.22) より，式 (2.13) は，

$$\boldsymbol{y} = f(\boldsymbol{u}) = f(\boldsymbol{W}\boldsymbol{x}) \tag{2.23}$$

となる．これを，式 (2.14)，式 (2.15) のように層の繋がりを考慮した記述に直すと，

$$\boldsymbol{x}^{<l+1>} = f\left(\boldsymbol{u}^{<l+1>}\right) = f\left(\boldsymbol{W}^{<l+1>}\boldsymbol{x}^{<l>}\right) \tag{2.24}$$

または，ベクトル，行列の要素表示を用いて，

$$x_j^{<l+1>} = f\left(u_j^{<l+1>}\right) = f\left(\sum_i w_{ji}^{<l+1>} x_i^{<l>}\right) \tag{2.25}$$

となる．以上により，バイアス $\boldsymbol{b}$ を学習する代わりに，常に 1 を出力するユニットと次層の各ユニット間の重みを学習することに置き換えたことになる．

ここから，具体的に損失関数の重みによる微分を求めていく．式 (2.19) の損失関数は，1 組の学習データと教師データを考え E とする．$\boldsymbol{y}_n$，$\boldsymbol{t}_n$ から n を削除し，これらベクトル表記を要素表記にして要素番号 j を付し，

$$E = \frac{1}{2}\sum_{j=1}^{N_L}(y_j - t_j)^2 \tag{2.26}$$

とする．ここで，N_L は，出力層のユニット数である．出力層に関して，

$$y_j = f\left(u_j^{<L>}\right) = f\left(\sum_{i=1}^{N_{L-1}} w_{ji}^{<L>} x_i^{<L-1>}\right) \tag{2.27}$$

であるので，

$$\begin{aligned} E &= \frac{1}{2}\sum_{j=1}^{N_L}\left(f\left(u_j^{<L>}\right) - t_j\right)^2 \\ &= \frac{1}{2}\sum_{j=1}^{N_L}\left(f\left(\sum_{i=1}^{N_{L-1}} w_{ji}^{<L>} x_i^{<L-1>}\right) - t_j\right)^2 \end{aligned} \tag{2.28}$$

と表せる．微分の連鎖律を考慮すれば，

$$\frac{\partial E}{\partial w_{ji}^{<L>}} = \frac{\partial E}{\partial u_j^{<L>}} \cdot \frac{\partial u_j^{<L>}}{\partial w_{ji}^{<L>}} \tag{2.29}$$

となる．この式で，右辺の第 1 項が E の $u_j^{<L>}$ 微分であり，第 2 項が，

$$u_j^{<L>} = \sum_{i=1}^{N_{L-1}} w_{ji}^{<L>} x_i^{<L-1>} \tag{2.30}$$

の $w_{ji}^{<L>}$ 微分であるので，

$$\frac{\partial E}{\partial w_{ji}^{<L>}} = \left\{ f\left(u_j^{<L>}\right) - t_j \right\} \cdot \frac{df(u_j^{<L>})}{du_j^{<L>}} \cdot x_i^{<L-1>}$$
$$= \{y_j - t_j\} \cdot \frac{df(u_j^{<L>})}{du_j^{<L>}} \cdot x_i^{<L-1>} \tag{2.31}$$

となる．これは，ユニット j の出力と教師データの差，ユニット j への入力値における活性化関数の微分値，前の層のユニット i の出力値の積であり，何れの値も既知である．

次に，中間層での損失関数の偏微分を考える．ここでも連鎖律を考慮して，

$$\frac{\partial E}{\partial w_{ji}^{<l>}} = \frac{\partial E}{\partial u_j^{<l>}} \cdot \frac{\partial u_j^{<l>}}{\partial w_{ji}^{<l>}} = \frac{\partial E}{\partial u_j^{<l>}} \cdot x_j^{<l-1>} \tag{2.32}$$

となる．この式の右辺の第 2 項は，l 層のユニット j の出力なので求めることができる．第 1 項に関しては，$u_j^{<l>}$ が変化すると，ユニット j からの出力 $x_j^{<l>}$ が，$l+1$ 層のユニット k の重み付き和 $u_k^{<l+1>}$ を変化させることによって，E の変化が生じることを考慮すれば，

$$\frac{\partial E}{\partial u_j^{<l>}} = \sum_{k=1}^{N_l} \frac{\partial E}{\partial u_k^{<l+1>}} \cdot \frac{\partial u_k^{<l+1>}}{\partial u_j^{<l>}} \tag{2.33}$$

となる．ここで，順伝搬では，

$$u_k^{<l+1>} = \sum_{k=1}^{N_l} w_{kj}^{<l+1>} x_j^{<l>} = \sum_{k=1}^{N_l} w_{kj}^{<l+1>} f(u_j^{<l>}) \tag{2.34}$$

であることを考慮すると，

$$\frac{\partial u_k^{<l+1>}}{\partial u_j^{<l>}} = \frac{df(u_j^{<l>})}{du_j^{<l>}} \cdot w_{kj}^{<l+1>} \tag{2.35}$$

のように微分が求められる．これを，式 (2.33) に代入すると

$$\frac{\partial E}{\partial u_j^{<l>}} = \sum_{k=1}^{N_l} \frac{\partial E}{\partial u_k^{<l+1>}} \cdot \frac{df(u_j^{<l>})}{du_j^{<l>}} \cdot w_{kj}^{<l+1>} \tag{2.36}$$

となる．この式で，右辺の 2 項目はユニット j への重み付き和入力の微分，3 項目は現状

の $l+1$ 層の重み値であるので既知である．1 項目は，出力層での，$\frac{\partial E}{\partial u_j^{<L>}}$ から順に計算できるので，左辺も計算可能である．この計算結果より，式 (2.32) も計算可能であり，損失関数 E の重み $w_{ji}^{<l>}$ による偏微分値を計算することができ，式 (2.20) によって，重みの更新が可能となる．

2.2.4　損失関数と学習サンプル

前節では，式 (2.19) を損失関数として定義した．これは，N_T 個の学習サンプル全てを用いてニューラルネットワークの出力誤差を計算し，重み $\boldsymbol{W}$ を更新している．このような方法を，「バッチ学習」と呼んでいる．これに対し，N_T 個の学習サンプルの中からランダムに選んだ一つの学習サンプルを用いて重みを更新する手法を「確率的勾配降下法」と呼び，バッチ学習と比べて次のような特徴がある．

バッチ学習は，1 回の反復において全ての学習サンプルを用いているために，サンプル数が増えれば，学習の計算コストも増えることになるが，確率的勾配降下法では一つのサンプルを用いるため，計算コストは常に一定である．また，バッチ学習における損失関数は常に式 (2.19) であるため，局所的な極小解に陥ると，そこから抜け出すことは困難であるが，確率的勾配降下法では，1 回の反復ごとにサンプルがランダムに選ばれることから，毎回異なる損失関数となるため，局所的な極小解に陥るリスクを低減することができる．しかし，確率的勾配降下法では，ある更新で使用される学習サンプルに外れ値などノイズがあると，不正な方向へ重みを更新してしまうことがある．

そこで，全学習サンプルよりは少ない複数のサンプルの集合（ミニバッチ）を用いて 1 回の更新を行う「ミニバッチ学習」が提案されている．t 回目の更新におけるミニバッチを $\mathcal{D}_t$，これに含まれるサンプル数 (ミニバッチサイズ) を $N_t = |\mathcal{D}_t|$，個々の学習サンプルによる誤差を $E_n(\boldsymbol{W})$ とすれば，t 回目の更新に用いられる損失関数 $E_t(\boldsymbol{W})$ は，

$$E_t(\boldsymbol{W}) = \frac{1}{N_t} \sum_{n \in \mathcal{D}_t} E_n(\boldsymbol{W}) \tag{2.37}$$

となる．誤差の合計を N_t で割って正規化しているのは，ミニバッチサイズを変更したときに，学習係数を調節する必要をなくすためである．この正規化の効果と，ミニバッチに含まれるサンプルをランダムに選ぶことによって，確率的勾配降下法の効果をある程度維持しながら，外れ値の影響も緩和できるのがミニバッチ学習の特徴である．

第 3 章　文字認識のための見出し文字列の地紋除去

本章では，1.2.1 節に示した，文書検索のキーワードとなり得る単語を有する見出しの文字認識を実行する際に障害となる背景飾りの「地紋」を除去する方法について述べる．文字認識の性能が向上しているとはいえ，背景に地紋のある見出しの文字認識は困難である．本研究では，多くの新聞に掲載されている地紋付き見出しを調査し，形状の特徴によって，妥当と考えられる 3 種類に分類できることを示す．その分類結果に従い，それぞれの地紋に対して，人間の知識に基づいた簡単な除去アルゴリズムと，それらのアルゴリズムを効果的に組み合わせた，除去アルゴリズム全体の構成を提案する．

処理結果である地紋が除去された見出し文字列の評価は，人間による目視と文字認識率によって行う．目視による評価では，処理結果を 4 段階で評価する．また，文字認識率による評価では，目視による評価それぞれに属する処理結果ごとに認識実験を行い，両評価の関連を確認する．

本章の構成は次の通りである．3.1 節において，地紋を形状の特徴によって分類する．3.2 節では，分類された地紋それぞれに対する除去手法を検討するとともに，見出し文字の白黒判別および地紋の判別方法も検討し，統合した処理の流れによって地紋除去を実行した結果を 3.3 節でまとめる．

3.1　地紋の分類

地紋の種類とその特徴を明確にするため，まずは，数紙の新聞から地紋の付いた見出し 100 サンプル程度を抜き出して調査した．その結果，地紋は大きく以下の 3 種類に分類できると仮定した．

1. 薄い網線・網点による地紋
2. グラデーションを有する地紋

(a) 網線網点地紋　(b) グラデーション地紋　(c) 片側地紋　(d) その他

図 3.1　地紋の分類

3. 文字の片側のみ存在する地紋

それぞれの地紋の例を，図 3.1(a)〜(c) に示す．また，本論文ではこれ以降，1〜3 に分類された地紋を，それぞれ「網線網点地紋」,「グラデーション地紋」,「片側地紋」と記述する．

この分類の妥当性を検証するために，更に主要な新聞社 4 社（読売，朝日，毎日，日経）の連続 10 日分の新聞に掲載された地紋付き見出し 1000 サンプルを抽出して調査し，上記 3 種類への分別を行った．結果，網線網点地紋に分類されたものが 65.8 %，グラデーション地紋に分類されたものが 13.8 %，片側地紋に分類されたものが 11.3 %，その他に分類されたものが 9.1 % であることがわかった．ここで，その他に分類されたものは，図 3.1(d) に示すように構造が複雑であり，絵や写真に重なっている場合が多く，形がまちまちなものが多い．

また，書体に関しては，本文や 3 種類に分類された時紋付き見出しに多く見られる「明朝」「ゴシック」とは異なったものが含まれている場合が多く，影付けがされるなど地紋

が無くても文字認識は困難と考えられる．しかし，これらの地紋のほとんどが，「社説」や「投書欄」などの記事の欄につけられたタイトルを強調するためのものであり，それら記事の内容を具体的に表す見出しは，これとは別に存在している．従って，本研究では，「網線網点地紋」，「グラデーション地紋」，「片側地紋」に分類される地紋のみに限定した除去手法を提案する．

3.2　地紋除去手法の検討

提案する地紋除去手法の基本方針は，三つに分類された地紋の形状の性質を考慮し，自動的にこれらを除去することである．そこで，まずは市販の画像処理系ソフトウェアのフィルタ機能などを用いて手作業によって，どのような手順を経てそれぞれの地紋を除去できるか検討したところ，以下のようになった．

1. 網線網点地紋に分類されるものは，図 3.2 に示す例のように，ぼかし処理の後に 2 値化することよって除去可能である．従って，入力された見出し画像に応じて適切な 2 値化閾値を自動決定することが検討項目となる．
2. グラデーション地紋に分類されるものは，グラデーションの濃淡を網点の大きさによる疑似中間調で表現しており，図 3.3 に示す例のように，網線網点地紋と同様の方法で除去できるものも存在するが，一般には除去できない（図 3.5 左側の例）．
3. 片側地紋に分類されるもののうち，図 3.4 に示す例のように，地紋部分が網線もしくは網点で構成されている場合には，網線網点地紋と同様の方法で除去できるが，大多数はべた塗りに近いものであり，除去は困難である（図 3.5 右側の例）．

この後の各節では，提案手法を構成する各地紋の除去ユニットの詳細について述べる．入力画像は，地紋のついた新聞の見出しをイメージスキャナを用いて，解像度 150 dpi，256 階調のグレースケール画像としてコンピュータに読み込んだものを用いた．150 dpi という解像度は，解像度を変えた読み込み実験を行った結果から，文字と地紋を分離している「文字の縁取り」がつぶれないと考えられる最小限界値から設定した．

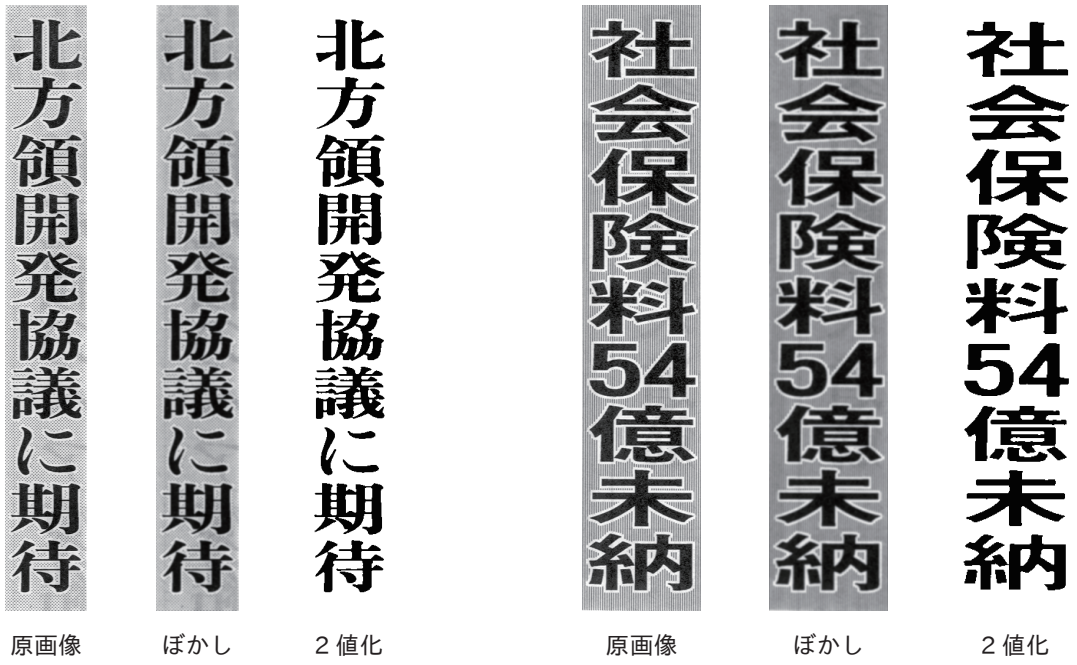


図 3.2　手作業による網線網点地紋の除去

図 3.3　ぼかし処理と 2 値化によって除去できるグラデーション地紋の例

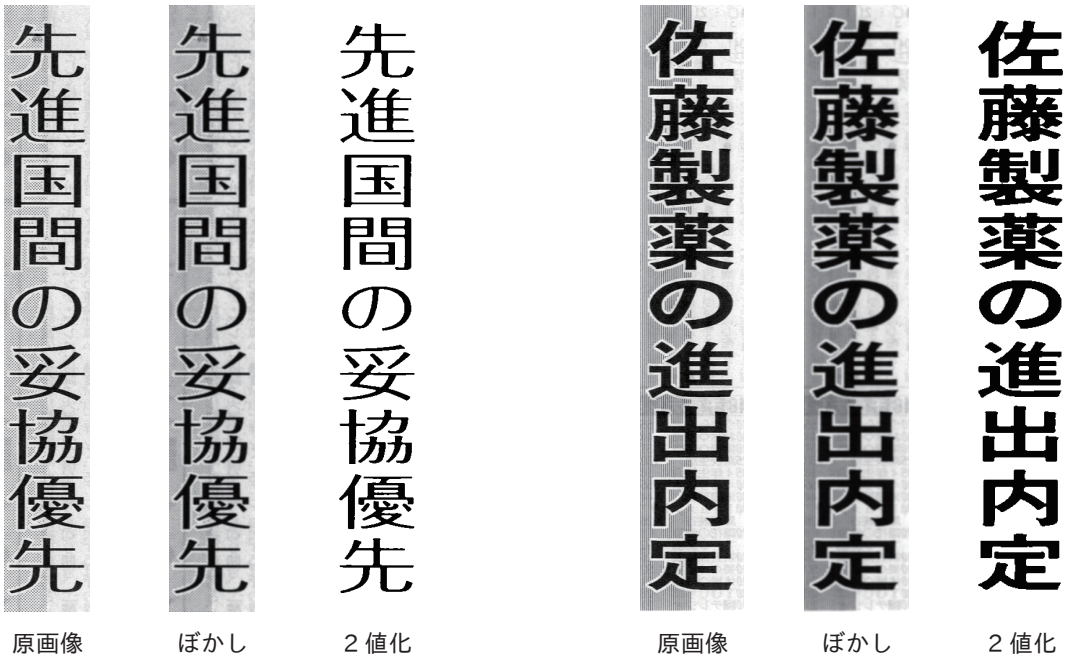


図 3.4　ぼかし処理と 2 値化によって除去できる片側地紋の例

図 3.5　ぼかし処理と 2 値化によって除去できないグラデーション地紋，片側地紋の例

3.2.1　網線網点地紋の除去

入力された見出し画像から網線網点地紋を除去する手法について述べる．この段階では，網線網点地紋の大部分と，グラデーション地紋及び片側地紋の一部が除去される．具体的な手法は，移動平均フィルタによってぼかした後，2 値化を行う．移動平均フィルタは，平均化の能力により，周辺に比べて極端に大きいまたは小さい画素値をノイズとして除去するフィルタとして利用される．これを利用して，白黒問わず細い線や小さな点を薄くすることができる．ここでは，式 (2.1) において，$w_{\boldsymbol{y}} = 1/9, \forall \boldsymbol{y} \in \mathcal{S}$ とした

$$f'_{\boldsymbol{z}} = \sum_{y \in \mathcal{S}} \frac{1}{9} f_{\boldsymbol{z}+\boldsymbol{y}} \tag{3.1}$$

を用いた．ここで，$\mathcal{S}$ には，式 (2.2) のように 3×3 の係数を用いている．

2 値化の閾値は，入力された見出し画像に従って自動的に決定できるのが望ましい．また，ぼかし処理後の画像の各画素が，閾値によって，文字を構成する画素とそれ以外（地紋と背景）を構成する画素の二つのクラスに明確に分離できるのが理想である．そこで，大津によって提案された判別分析法 [26]

$$\begin{aligned}
\sigma_B^2(t) &= \frac{\displaystyle\sum_{k=k_0}^{t-1} n_k(\overline{f_0} - \overline{f})^2 + \sum_{k=t}^{N} n_k(\overline{f_1} - \overline{f})^2}{\displaystyle\sum_{k=k_0}^{N} n_k} \\
\sigma_I^2(t) &= \frac{\displaystyle\sum_{k=k_0}^{t-1} n_k(k - \overline{f_0})^2 + \sum_{k=t}^{N} n_k(k - \overline{f_1})^2}{\displaystyle\sum_{k=k_0}^{N} n_k} \\
F_0(t) &= \frac{\sigma_B^2(t)}{\sigma_I^2(t)}
\end{aligned} \tag{3.2}$$

を用いて $F_0(t)$ を最大にする t を求めることによって設定することとした．ここで，$\overline{f_0}$ は，画素値が 0〜$t-1$ の画素の平均画素値，$\overline{f_1}$ は，画素値が t〜N の画素の平均画素値，$\overline{f}$ は全画素の平均画素値，n_k は，入力画像から求めた画素値ヒストグラムにおいて画素値

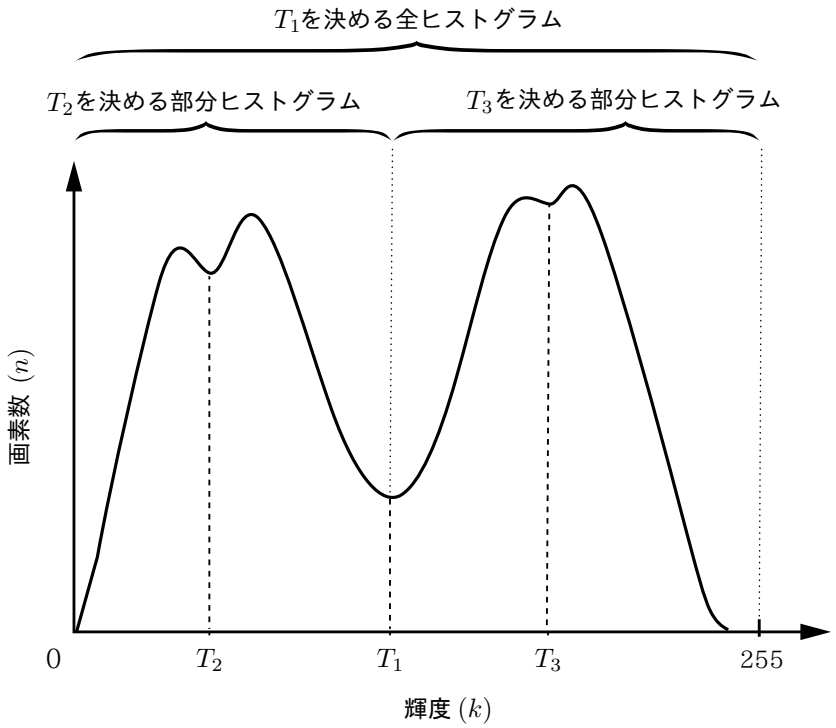
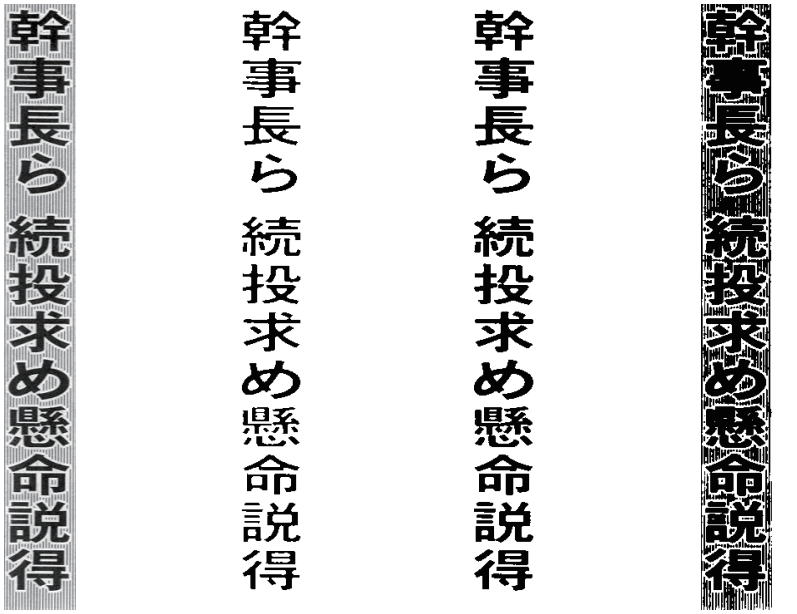


図 3.6　2 値化閾値の決定

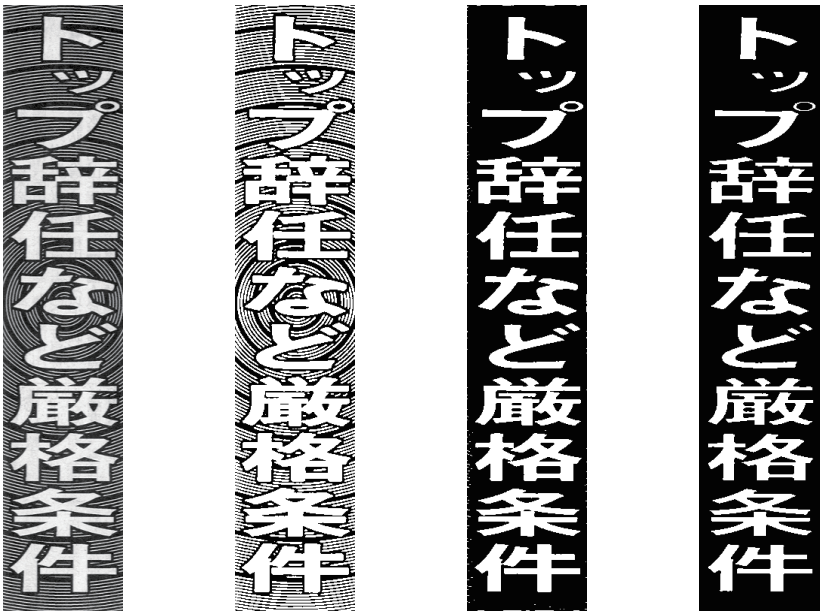
k を持つ画素の数である．ただし，判別分析法によって決定した閾値をそのまま用いて 2 値化する場合，常に文字とそれ以外の画素値が明確に分離できるとは限らず，地紋が残ってしまう恐れがある．これを回避するため，本研究では，閾値を判別分析法によって決定した値より，文字の色に近い値に設定する手法を提案する．

図 3.6 に示すように三つの閾値を設定して検討した．一つ目の閾値は，ヒストグラム全体に判別分析法を適用して求めた $t = T_1$，残り二つは，0 から $T_1 - 1$ までと，T_1 から 255 までのそれぞれの部分的ヒストグラムに対して判別分析法を適用して求めた $t = T_2$ および $t = T_3$ である．式 (3.2) を用いて計算する際，T_1 を求める場合は $k_0 = 0, N = 255$，T_2 を求める場合は $k_0 = 0, N = T_1$，T_3 を求める場合は $k_0 = T_1, N = 255$ をそれぞれ設定する．この三つの閾値を用いて処理を行った結果の例を図 3.7 に示す．図の最も左が原画像，続く三つが，移動平均フィルタ処理後に三つの閾値 (T_2，T_1，T_3) で 2 値化した結果を示している．



原画像　閾値 69　閾値 124　閾値 194

(a) 黒文字処理結果の例



原画像　閾値 88　閾値 155　閾値 208

(b) 白文字処理結果の例

図 3.7　網線網点地紋除去予備実験の結果例

表 3.1　予備実験結果

文字色	良好に除去された割合 [%]		
	T_2	T_1	T_3
白	0.0	33.3	95.8
黒	90.4	28.6	0.0

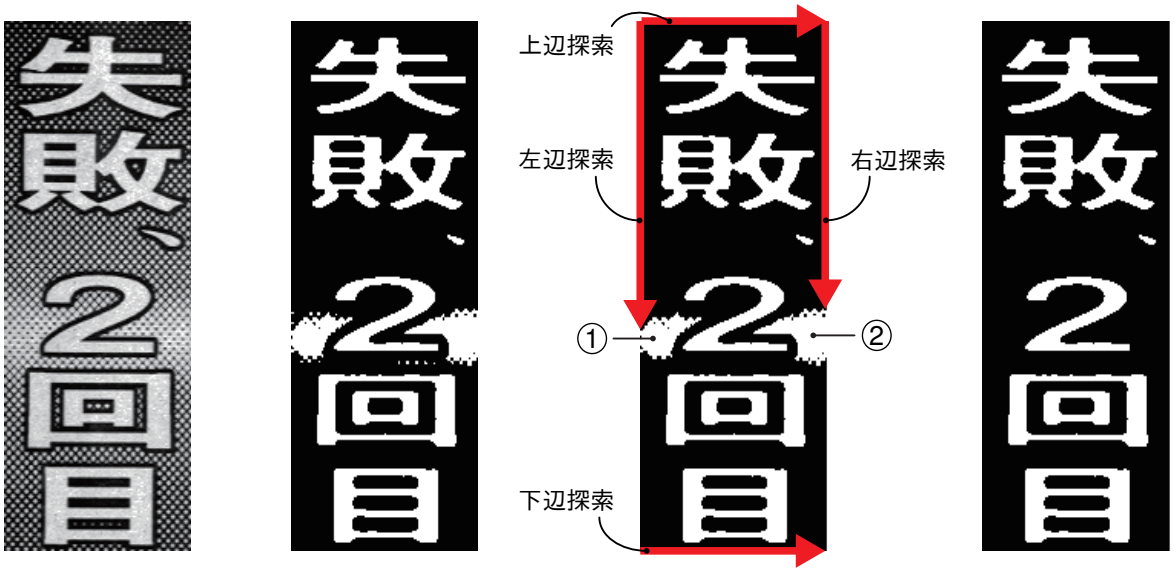
網線網点地紋と分類されたもの 100 種類 (黒文字:41，白文字:59) を対象とし，三つの閾値を用いて 2 値化する予備実験を行った結果を表 3.1 に示す．それぞれの閾値に対して，地紋が完全に除去されており，かつ，文字部分の欠けがない結果を良好と判断し，その割合を示している．表より，文字が黒の場合は，閾値 T_2 で，白の場合は閾値 T_3 で 2 値化すれば，ほとんどの地紋に対して良好な結果が得られることが確認できる．すなわち，3.2.4 節で述べる文字の白黒判別結果から，黒文字ならば T_2，白文字ならば T_3 を用いることで，2 値化の閾値の自動決定が可能となる．

3.2.2　グラデーション地紋の除去

グラデーション地紋の除去手順を図 3.8 に示す．まず，原画像 (a) を判別分析法による閾値を用い 2 値化した見出し画像 (b) に対し，白領域で面積が 9 画素以下のものを除去する．9 画素としたのは，150 dpi の見出し画像では，文字を構成する領域のうち最小である句読点・濁点であっても，9 画素を下回ることはあり得ないためである．この時点で，残った白領域は文字領域と，(c) の①，②に見られる，小領域として除去されなかった地紋領域である．

次に，3.2.4 節で述べたように，文字領域は画像の縁に接することはなく，地紋領域は接触するという性質を用いて，図中に赤い矢印で示した画像の縁をスキャンし，白画素を見つけたら，そこから一続きの白領域を黒で塗りつぶし，(d) に示す結果が象を得る．図の例では，左右の縁をスキャンすると地紋の白領域がみつかる．

前述のとおり，この処理はすでに地紋が除去されている見出しも対象となる．その場合でも，本処理では，文字そのものには影響は無いので，逆にノイズのないさらに良好な結果が期待できる．また，片側地紋と識別されるべきものがグラデーション地紋と判断されてしまったものでも，片側にのみ存在する白領域は，画像の端に接触しているため，画像の縁に接触する領域の除去によってある程度除去可能である．



(a) 原画像　(b) ぼかし+2値化画像　(c) 地紋領域探索　(d) 結果画像

図 3.8　グラデーション地紋の除去

3.2.3　片側地紋の除去

文字の片側のみに存在する地紋の除去について述べる．このような地紋は，ほとんどのペイント系ソフトウェアに実装されている塗りつぶし処理を用いて図 3.9(a) のように，背景の白領域を黒く塗りつぶすことで除去できる場合もある．図中の破線は画像の領域と紙面の境界を表し，赤い×のある白領域と一続きの領域を塗りつぶすとする．同図 (b) のように，背景領域が文字に囲まれ，一続きの領域になっていないような場合には，赤丸で示したように，塗り残しが発生してしまう．したがって，注目している白領域が背景領域か文字領域かを区別して塗りつぶす必要がある．

これらの除去手法は，林らが提案しているが [7]，“くにがまえ” などのように他の文字要素を完全に囲んでしまう場合には，正常に除去できない．また，除去に失敗した場合，文字はほとんど読みとれなくなってしまう．そこで，“くにがまえ” の問題も含めて，安定して片側地紋を除去できる方法を検討するため，改めて見出しの性質について考察した結果，“文字領域同士・背景領域同士が隣り合わない” という性質を見いだすことができた．

本研究では，この性質を利用した以下の手法を提案する．図 3.10(a) の画像を例に手順を説明する．(a) は前処理として入力画像を判別分析法より求めた閾値によって 2 値化した見出し画像に対して，左上の画素から各ラインを右へスキャンし白領域に対してラベリ

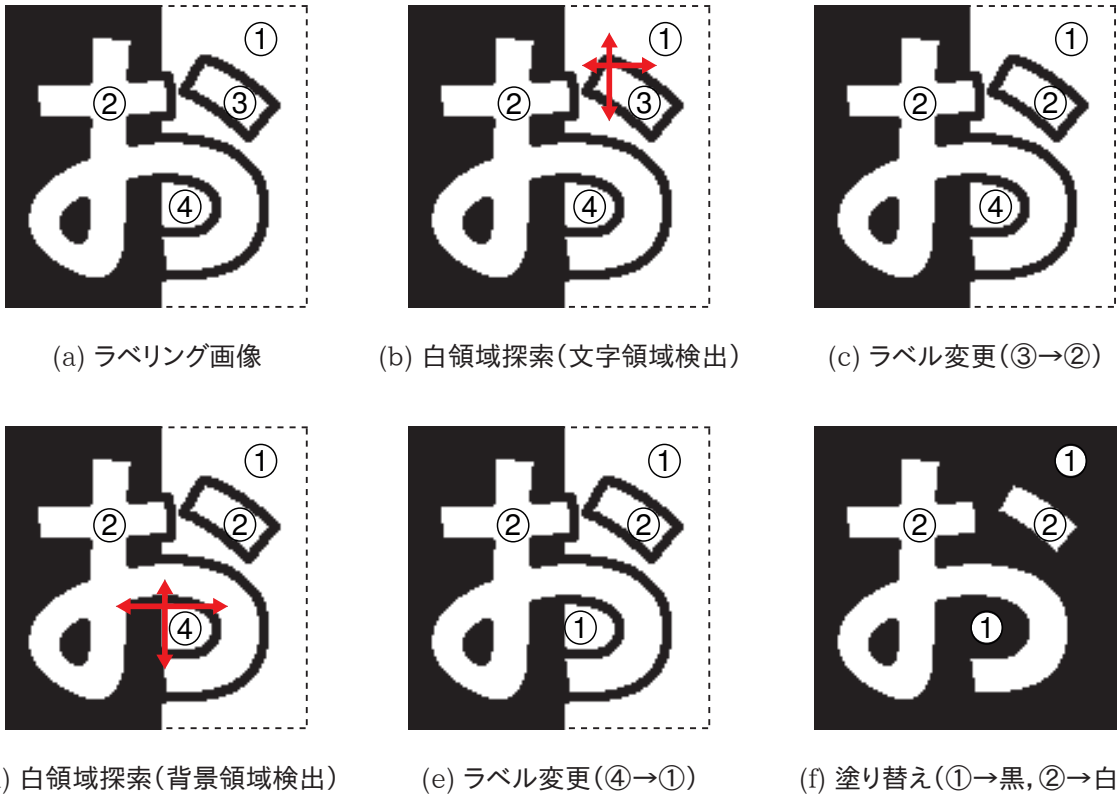
(a) 塗りつぶし処理によって除去可能な場合

(b) 塗り残しが発生する場合

図 3.9　塗りつぶし処理による除去

ングを行い，そのラベルを①〜④としたものである．ラベル①領域は，画像の縁 (同図中の破線) に接するので，必ず背景領域である．ラベル②領域は，必ず文字領域である．なぜなら，ラベル①領域以外で白の背景となりうる領域は，ラベル④領域のように，文字のみ，または，文字と片側黒領域によって囲まれた領域であるため，背景領域を囲む文字領域が常に先にラベリングされることになるからである．この状態から，③以上のラベルを，次のように処理する．

1. 画像を左上からスキャンし，①，②以外のラベルの付いた画素を検出したら，その画素から上下左右の 4 方向に，ラベル①または②の付いた画素を探索する — この画素を含む領域が文字領域ならば，周りにはラベル①領域が存在し，背景領域ならば，ラベル②領域が存在するはずである．
2. 4 方向を探索して検出したラベルをチェックし，ラベル①よりラベル②の方が多ければ，そのラベル領域は背景領域と判断し，ラベルを①に変更する — (b) に示すとおり，ラベル③領域は 4 方向すべてにラベル①の領域が存在しているので，文字領域であると判断し，(c) のようにラベルを②に変更する．

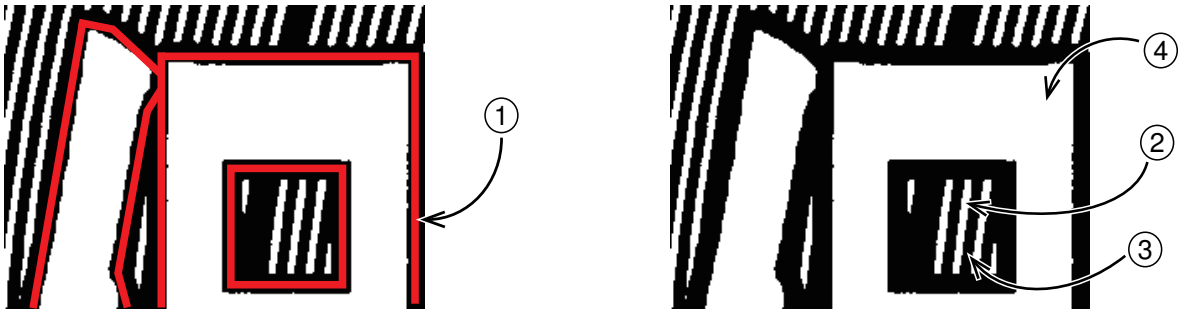


(a) ラベリング画像　(b) 白領域探索（文字領域検出）　(c) ラベル変更（③→②）

(d) 白領域探索（背景領域検出）　(e) ラベル変更（④→①）　(f) 塗り替え（①→黒，②→白）

図 3.10　片側地紋の除去

3. ラベル②よりラベル①の方が多ければ，文字領域と判断しラベルを②に変更する
— ラベル④領域は，(d) に示すように，4 方向のすべてにラベル②の領域が存在しているので，背景領域と判断し，(e) のようにラベルを①に変更する．
4. 1.〜3. の処理を③以上のラベルがなくなるまで繰り返し，画像上に存在するのが黒領域とラベル①，②領域のみにする．
5. ラベル①領域を黒，ラベル②領域を白に変更し，(f) のように文字を白領域として抽出する．



(a) 文字の縁取り　　(b) 地紋領域

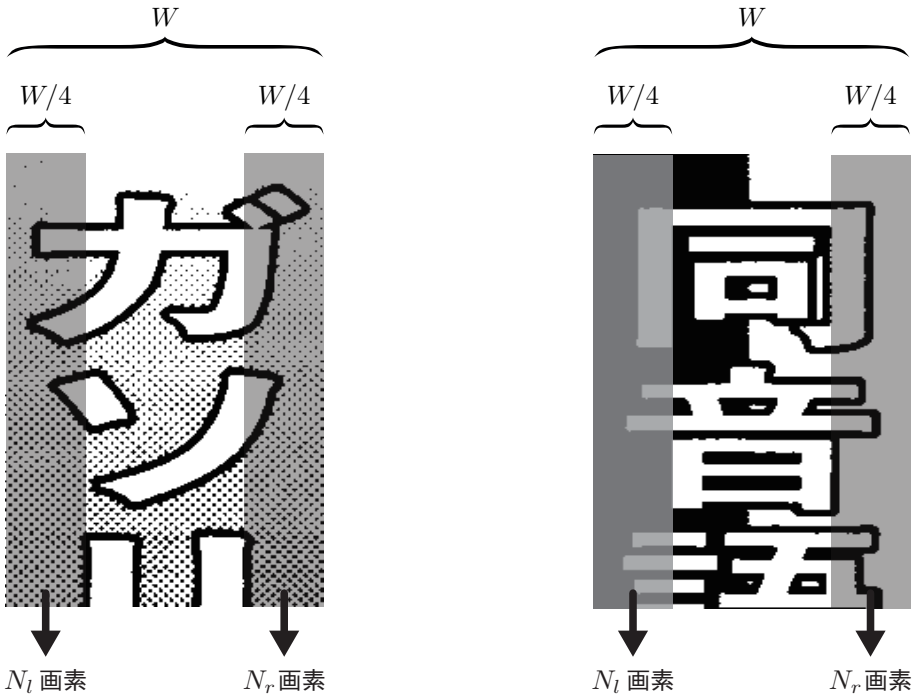
図 3.11　見出しの構造

3.2.4　文字の白黒判別

本研究では，見出し文字列の画像を入力して，そこに付された地紋を除去して出力するまでを想定している．しかし，最終的なシステムでは，この後に見出し文字列を文字認識して出力することになるが，見出し文字の色は白の場合と黒の場合があるので，文字認識処理が想定する文字の色に統一しておく必要がある．そのため，次のような手順で見出し文字の白黒を判別する．

見出し文字の白黒を判別する際に重要となる要素は，それを読む人間に対し文字領域と地紋の区別を明確にするために文字のまわりに設置されている，図 3.11(a) の①部分のような縁取りである．縁取りが存在していることで文字領域は画像の縁には接触しない（図では，わかりやすいように赤で示している）．これに対して，地紋を構成する領域の白または黒の領域は画像の縁に接触するものとしないものとが存在するが，(b) の②地紋白領域，③地紋黒領域，④文字領域）に示すように，接触しない領域の面積は文字領域の面積に比べて小さいこのことから，画像の縁と接しない最大の領域が，白 (黒) ならば白 (黒) 文字と判断できる．

実際の処理では，各々の白または黒の領域に対して，白黒以外の色で塗りつぶし処理を施し，その過程で，x 座標，y 座標いずれかが 0 である画素に到達したら，この領域は無視し，そうでない場合は，塗りつぶした画素数を面積として残す．全ての領域に対して処理が終了したら，面積によって白黒を判別するが，このとき面積が上位三つの領域の白黒を調べ，白黒どちらの領域が多いかによって判断することによって精度を向上させている．



(a) グラデーション地紋　　(b) 片側地紋

図 3.12　左右の白画素計数

3.2.5　グラデーション地紋と片側地紋の判別

網線網点地紋とグラデーション地紋，片側地紋の一部は，3.2.1 節と 3.2.4 節で述べた処理を用いることによって除去できる．次に，見出し文字列に残ったのがグラデーション地紋か片側地紋かを判別して，別々の手法で除去する．以後，処理の一貫性を保つため，縦書き見出しを基本として，横方向画素数が縦方向画素数より大きくなる横書き見出しは，転置し縦書きに変換し，除去処理終了後にもう一度転置して横書きに戻す．また，3.2.4 節の方法によって，文字の色の判別もできているので，文字の色に関しては，白を基本として，文字の色が黒である場合は階調反転を行う．

グラデーション地紋と片側地紋を判別するために用いた知識は以下のものである．この知識は，本研究における地紋の分類に適用されるものであり，その性質が大きく異なる地紋があった場合は，この知識も改めて検討する必要がある．

1. 文字を構成する白画素は，画像の左右半分にほぼ同数存在する．

2. グラデーション地紋を構成する白画素は，画像の左右半分にほぼ同数存在する．
3. 片側地紋を構成する白画素は，画像の左右で分布が大幅に偏っている．

この知識をもとに，以下のようにして地紋を判別する．まず，図 3.12 に示すように，図中灰色で示した，画像の左右の縁から内側へ向かって，画像の横方向画素数 W の 4 分の 1 の領域内の白画素数を数える（左側計数 N_l，右側計数 N_r）．ここで，W の 4 分の 1 領域とした理由は，片側地紋で，地紋ありの部分と地紋無しの部分の白黒の境目が，常に画像の中心線上にあるわけではなく時には大きく左右にずれている場合があり，左右半々の領域をカウントすると，白画素数の偏りが小さくなり誤判別を起こしやすくなるからである．

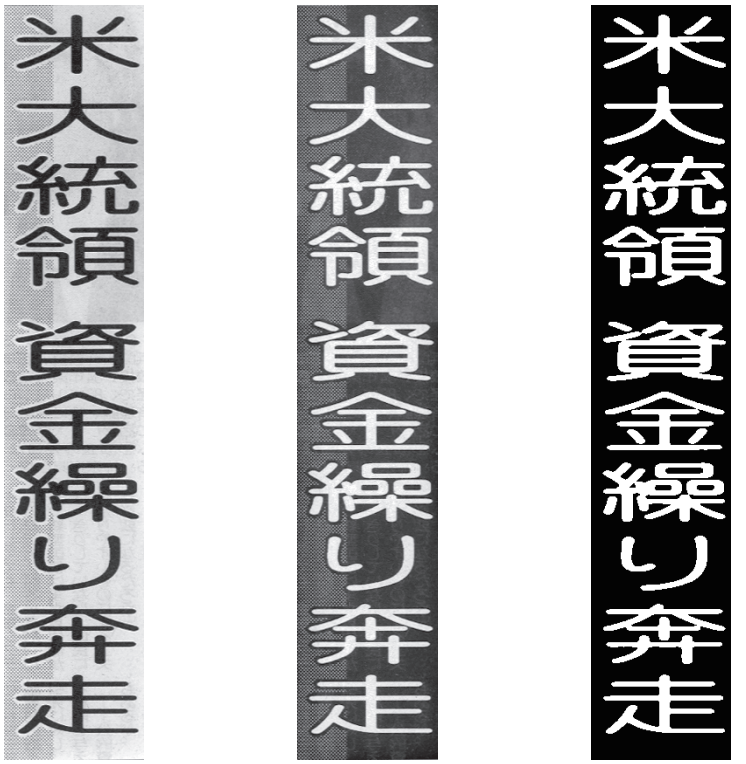
計数した二つの領域の白画素数を比べて，多い方が少ない方の 2 倍以上であれば片側地紋と判断する．図の例で，(a) では見た目にも N_l と N_r の差は少ないと考えられ，(b) では明らかに N_r の方が大きいと考えられる．従って，(a) はグラデーション地紋，(b) は片側地紋と判別されることになる．判別後，それぞれの地紋に対する除去処理に移る．

特殊な例として，文字の並び方向に対して垂直な方向に疑似濃淡が変化しているものがあった．これらは，グラデーションの程度によってグラデーション地紋か，片側地紋のどちらになるかが異なるが，最終的に良好に地紋が除去されれば，どちらと判別されたかは問題としないこととした．グラデーション地紋と判別された例を図 3.13 に示す．

また，片側地紋に分類されているもののグラデーション地紋と判別される場合がある．これは，図 3.14 に示すように，地紋部分が網線網点で構成されているため，ぼかし処理と 2 値化によって除去できてしまったためである．この場合は，地紋を判別するための知識「文字を構成する白画素は，画像の左右半分にほぼ同数存在する」によってグラデーション地紋と判別される．この処理では，グラデーション地紋と片側地紋は識別しているが，3.2.1 節で述べた処理で，地紋が除去できたかどうかは判断していない．

原画像　　階調反転　　ぼかし＋2 値化

図 3.13　グラデーション地紋が片側地紋と判別される例



原画像　　階調反転　　ぼかし＋2 値化

図 3.14　片側地紋がグラデーション地紋と判別される例

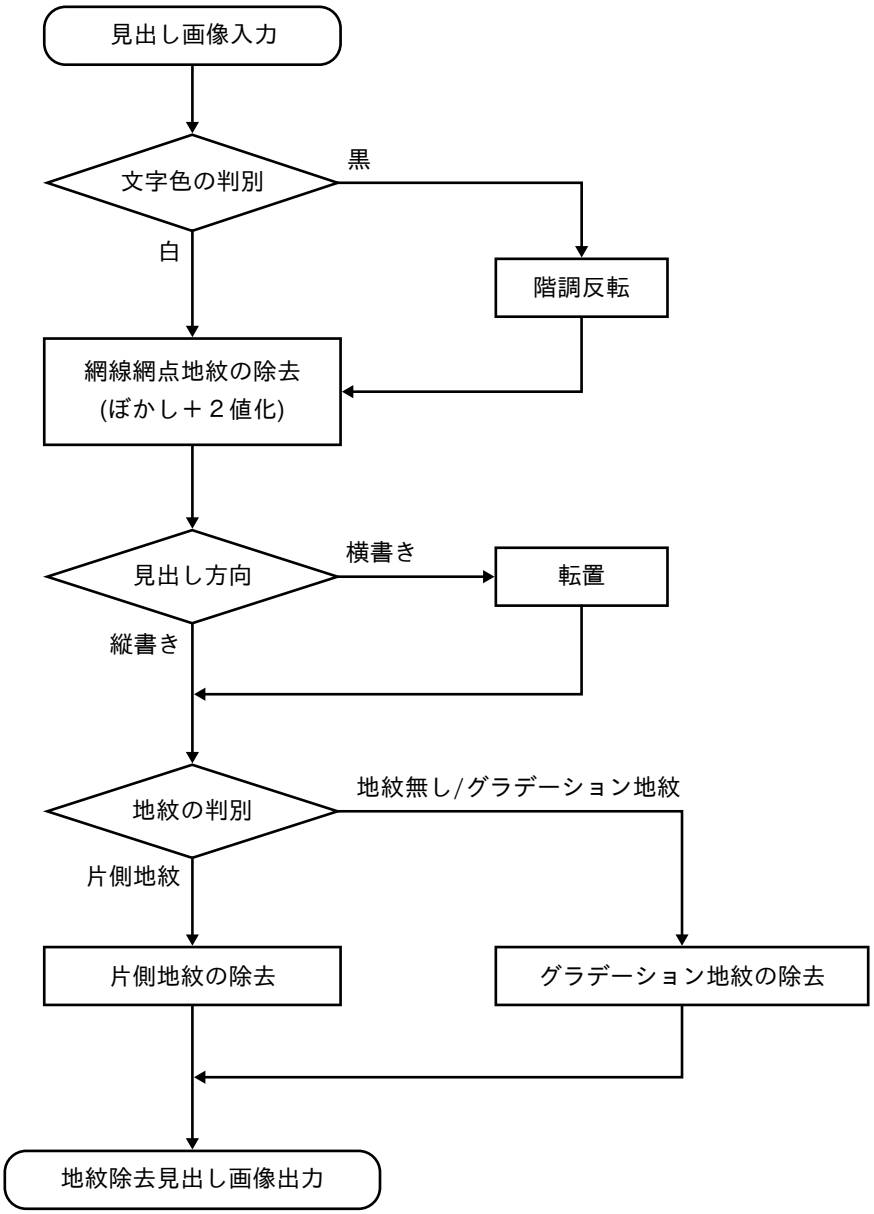


図 3.15　地紋除去処理全体の流れ

3.3　地紋除去結果

前節までに述べてきた各地紋除去手法を用いた処理全体の流れは，図 3.15 に示す通りである．これを，C 言語による計算機プログラムとして実装し，多数の地紋付き見出しを処理した結果について述べる．入力画像は，新聞から切り出した見出しを，3.2 節で述べたように，スキャナを用い 150 dpi のグレースケール画像として読み込んだものである．

表 3.2 処理結果の評価

地紋	白黒判別 [%]	評価の割合 [%]			
		A	B	C	D
網線網点	100.0	90.6	7.3	2.1	0.0
グラデーション	100.0	77.0	12.5	10.5	0.0
片側	100.0	76.4	22.3	1.3	0.0
全体	100.0	81.4	14.0	4.6	0.0

地紋の内訳は，網線網点地紋 が 140 個，グラデーション地紋が 90 個，片側地紋が 80 個であり，合計 310 個である．文字の白黒は，それぞれの地紋について半分ずつである．除去結果は，人間による目視で以下のように評価した．地紋が完全に除去できており，文字の乱れがないものを “A”，地紋の残りが少し目立つ，または，文字の乱れがほんの少しあるものを “B”，地紋の残りがある，または，文字の乱れがあるものを “C”，ほとんど除去されていない，または，文字が読めない程乱れているものを “D” とした．更に，文字認識を行うのにほとんど支障がないと考えられる “A”，“B” 評価をまとめて “良好”，逆に “C”，“D” 評価を “不良” と表現する．目視による主観評価であり，特に “B” と “C” の評価の違いは，文字領域と比較した地紋の残りや文字の欠けの大きさで判断するしかないが，この評価が正しいか否かの定量的な判断は，3.4 節に示す文字認識の結果に委ねる．

白黒判別と処理結果の評価を表 3.2 にまとめた．文字の白黒はすべて正しく判定された．しかし，これは，「文字と地紋を分離している縁取りがつぶれない程度」の解像度として設定した 150 dpi での結果である．そこで，実際の利用環境で，75 dpi で入力された場合の影響について実験を行った．縁取りのつぶれは増加したが，本手法は，3.2.4 節で述べたように，最大領域だけでなく上位三つの領域を用いて判定しているので，判定結果に影響は見られなかった．

以降では，各除去ユニットの処理結果について述べる．処理結果の画像には，いずれも，良好なもの（A，B 評価）と不良なもの（C 評価）を示している．

網線網点地紋の除去結果の例を図 3.16 に示す．97.9% が良好な除去結果となった．B 評価のものは，(b) に示すように，多少文字部分に欠けが生じているが，特に支障無く読むことができる．不良の原因は，(c) に示すように，黒文字で明朝体の場合，横棒がかなり細いため，ぼかし処理と 2 値化処理によって，本来つながっているべき個所が途切れてしまうという現象が起きたためである．

グラデーション地紋の除去結果の例を図 3.17 に示す．89.5% が良好な結果となった．

(a) A 評価 (b) B 評価 (c) C 評価

図 3.16 網線網点地紋除去結果の例

B 評価のものは，多少雑音が残っているが文字自体には問題は無く，支障なく読むことができる．不良の原因には，(c) に示すように，文字と地紋を本来繋がっているべき地紋が 2 値化の段階で途切れてしまい，塗りつぶされずに大きな雑音として残ってしまったためである．

片側地紋の除去結果の例を図 3.18 に示す．98.7% が良好な結果となった．(B) は，多少地紋が雑音として残っている程度である．不良の原因は，実際，新聞や雑誌の見出しなどでは，文字間が狭いまたは，文字が複雑であるため背景が省略されていたり，ぼかし処理と 2 値化の際につぶれてしまうので，縁取りを境にして文字を構成する領域が隣り合ってしまい，処理の前提となる，“文字領域同士・背景領域同士は隣り合わない” という性質が満たされなくなったためである (同図右)．

横書き見出しの処理結果を図 3.19 に示す．横書き見出しは，処理の一貫性のために，転置によって縦書きに変換して処理した後に，横書きに戻しているが，その場合でも処理は可能であった．上のサンプルは A 評価，下は B 評価である．なお，横書き見出しは縦

(a) A 評価　(b) B 評価　(c) C 評価

図 3.17　グラデーション地紋除去結果の例（左：原画像，右：処理結果）

書き見出しに比べてサンプル数が少ないので，両者に関する地紋除去能力の差異については確認していない．

以上の除去結果をまとめると，合計 310 サンプルの見出しに対して処理した結果，全体の 81.4% については，完全に地紋を除去できた．良好な除去が行われたのは，全体の 95.4% となった．

林らの手法 [7] においては，60 サンプルの見出しに対し 51 サンプル（85%）において完全に文字のみを抽出できたとしている．本手法の A 評価によって比較すると，若干性能が高いと考えられるが，林らの手法では，失敗した場合には，ほとんど地紋が除去されていないか，目視でもほとんど文字が読めなくなっているが，本手法では，完全な結果でなくても，ほとんどの地紋が除去されており，文字が欠けている場合でも，目視によってほとんどが読める程度になっている．また，岡本らの手法 [9] では，文字単位で評価しており，292 文字中 240 文字（82.2%）が復元されたとしている．本手法の評価は見出し単位の評価であり，一部分の地紋の残りや数文字の欠けでも評価が下がってしまっている．

(a) A 評価　　(b) B 評価　　(c) C 評価

図 3.18　片側地紋除去結果の例

表 3.3　文字認識結果

	無地見出し	評価			
		A	B	C	D
認識率 [%]	90.0	89.0	86.0	79.0	–

しかし，個々の文字を確認すると，B 評価や C 評価の中には，A 評価に含まれる文字と同等な結果が多く含まれており，本手法が有効であると考えられる．

3.4　文字認識結果

処理結果に対し文字認識を行った結果について述べる．認識には，加重方向指数ヒストグラム法を用いた市販の OCR ソフトを採用した．まず，ランダムに選んだ無地見出し中の文字を認識させ，本研究環境下での OCR ソフトの認識率を求めた．その結果，100 文字中 90 文字を認識し，認識率 90.0% であった．同ソフトの仕様では認識率 99% となっ

図 3.19　横書き見出しの結果

ていたが，見出しのような大きな文字では，分割文字を別々の文字として認識してしまったり，書体が精密に再現されてしまうため，認識率はむしろ低下してしまった．

次に，本手法によって地紋を除去された見出しから，(A)(B)(C) の評価別に，それぞれ 100 文字をランダムに選択して認識させた．その結果を**表 3.3** に示す．

人間による評価が良いものほど認識率が高くなっていることが分かる．(A)(B) では，ほぼ OCR ソフトの性能通り，すなわち，無地見出しの認識率 (90%) に近い値が得られた．(C) の様に見た目の除去結果が悪くても，比較的高い認識率が得られることが判明した．

第 4 章　文書上の注釈情報活用のための手書き文字の抽出

本章では，1.2.2 節で示した，紙媒体の文書に書き込まれた手書きの注釈を抽出する方法について述べる．紙媒体の文書には，その手軽さ故，注釈やメモが記入されることがある．注釈には有用な情報がある場合が多く，文書の中からこれを抽出することは，情報の再利用という観点からも，有益なことであると考えられる．

本研究では，印刷文書の輝度の窪みのみを選択的に除去するために，モフォロジカルフィルタのダイレーションとエロージョンをニューラルネットワークにおける Maxout 関数で置き換え，モフォロジカルクロージングフィルタを拡張して得られた Maxout フィルタネットワークを構成する．このネットワークを事例によって学習させたフィルタネットワークを用いて文書画像中の手書き文字の抽出方法を提案する．

これに加え，すべての画素において，手書き文字と活字が混在する文書画像の画素値と手書き文字のみの画像の画素値は等しいか，後者の方が大きくなるのが理想であるので，ネットワークの入出力を比較し，大きい方を出力する「最大値学習」というネットワーク構成も提案する．学習には，確率的勾配降下法，ミニバッチ学習，ミニバッチ＋最大値学習を用い，これらを比較する．

実験では，学術論文など，使用されるフォントの種類が少ない印刷文章を対象として，Maxout フィルタネットワークの学習及び手書き文字の抽出を行い，目視と SNR によって評価し，提案した最大値学習の効果を確認する．目視による評価では，理想手書き文字と比較して遜色ない抽出結果になっているかを確認する．また，従来の研究による印刷文書からの手書き文字抽出手法では困難であった，印刷文字と手書き文字が重なっている場合の抽出処理結果も評価する．

本章の構成は，次の通りである．4.1 節で，モフォロジカルフィルタと，その Maxout 関数による拡張，及び手書き文字を抽出するための Maxout フィルタネットワーク構成を提案し，その学習方法，及び使用する学習サンプルの作成方法を 4.2 節で説明する．学習が終了した Maxout フィルタネットワークを用いた手書き文字の抽出結果とその評価に

ついて，4.3 節で述べる．

4.1 Maxout フィルタネットワーク

本節では，2.1.2 節に示したモフォロジー画像処理の基本処理であるダイレーションとエロージョンの関係について述べ，それらが 2.2.2 節で示した Maxout 関数の特殊な場合であることを示し，この関係から，文献 [20] で提案されているように，クロージングフィルタを Maxout フィルタネットワークへ拡張する．

4.1.1 モフォロジカルフィルタと Maxout 関数による拡張

モフォロジー画像処理に関しては，2.1.2 節に述べたとおり，基本処理であるダイレーションは，式 (2.3) のように定義され，エロージョンは，式 (2.4) のように定義されている．両者の関係は

$$e_s \circ f_{\boldsymbol{z}} = -d_{s^*} \circ (-f)_{\boldsymbol{z}} \tag{4.1}$$

と表すことができる．ここで，s^* は，構造要素 s に対して原点を中心に点対称の構造要素であり，

$$s^*_{(p,q)} = -s_{(-p,-q)} \tag{4.2}$$

と表される．式 (2.3) と式 (4.1) から，式 (2.4) は

$$e_s \circ f_{\boldsymbol{z}} = -\bigvee_{\boldsymbol{y}\in\mathcal{S}} (-f)_{\boldsymbol{z}+\boldsymbol{y}} + s_{\boldsymbol{y}} \tag{4.3}$$

と書き換えることができる．

座標 $\boldsymbol{z}$ におけるダイレーション $d_s \circ f_{\boldsymbol{z}}$ は入力画像の画素の集合 $\{f_{\boldsymbol{z}+\boldsymbol{y}}\}_{\boldsymbol{y}\in\mathcal{S}}$ を用いて計算されるので，これを Maxout 関数の入力 $\boldsymbol{x}$ とすると，

$$\hat{d}_w \circ f_{\boldsymbol{z}} = \bigvee_{k=1\cdots K} \left\{ \left(\sum_{\boldsymbol{y}\in\mathcal{S}} w_{\boldsymbol{y},k} f_{\boldsymbol{z}+\boldsymbol{y}} \right) + b_k \right\} \tag{4.4}$$

のように Maxout 関数によって構成される非線形フィルタを定義することができる．ここで，$w_{\boldsymbol{y},k}$ は座標 $\boldsymbol{y}$ に対応する k 番目の荷重パラメータである．Maxout 関数が任意の凸関数を近似することができることから，このフィルタを凸フィルタと呼ぶ．この凸フィルタにおいて，k によって異なる $\boldsymbol{y}$ のときに $w_{\boldsymbol{y},k}$ が 1 となり，その他ではすべて 0 になる場合，バイアスを構造要素に持つダイレーションに一致する．

また，出力の符号を反転させたフィルタを凹フィルタと呼び，

$$\hat{e}_v \circ f_{\boldsymbol{z}} = - \bigvee_{k=1\cdots K} \left\{ \left(\sum_{\boldsymbol{y}\in\mathcal{S}} v_{\boldsymbol{y},k} f_{\boldsymbol{z}+\boldsymbol{y}} \right) + a_k \right\} \tag{4.5}$$

と表す．k によって異なる $\boldsymbol{y}$ のときに $v_{\boldsymbol{y},k}$ が -1 となり，その他ではすべて 0 になる場合，エロージョンに一致する．

モフォロジカルフィルタは，2.1.2 節において述べたとおり，ダイレーションとエロージョンを組み合わせることで構成される．オープニングとクロージングを構成するダイレーションとエロージョンを，それぞれ凸フィルタと凹フィルタで置き換えると，オープニングに対応する凹凸フィルタ

$$\hat{o}_{v,w} \circ f_{\boldsymbol{z}} = \hat{d}_w \circ \hat{e}_v \circ f_{\boldsymbol{z}} \tag{4.6}$$

及び，クロージングに対応する凸凹フィルタ

$$\hat{c}_{v,w} \circ f_{\boldsymbol{z}} = \hat{e}_v \circ \hat{d}_w \circ f_{\boldsymbol{z}} \tag{4.7}$$

を定義することができる．ここで，w, v は，それぞれ $w_{\boldsymbol{y},k}, v_{\boldsymbol{y},k}$ の略記である．

凹凸フィルタ，及び凸凹フィルタは，活性化関数として Maxout 関数を用いた 3 層畳み込み型ニューラルネットワークとみなすことができる．凸凹フィルタに関して，図 4.1 に示す．凹凸フィルタに関しては，式 (4.6) より，$\hat{d}_w$ と $\hat{e}_v$ を入れ替えた図になる．

文書画像からの手書き文字の抽出は，手書き文字を構成する黒画素の画素値を維持したまま，印刷文字を構成する黒画素のみを紙の色である白に変化させる処理，つまり，選択的に輝度の窪みを除去する処理であるので，基本となるのは，式 (4.7) で定義される凸凹フィルタである．そこで，凸凹フィルタの入力 f_z として，手書きの注釈が記入された文書画像，理想的な出力として，入力に記入されたものと同じ手書きの注釈のみの画像の複数の事例を用いて，荷重パラメータ v, w およびバイアス a, b を学習させることによって，

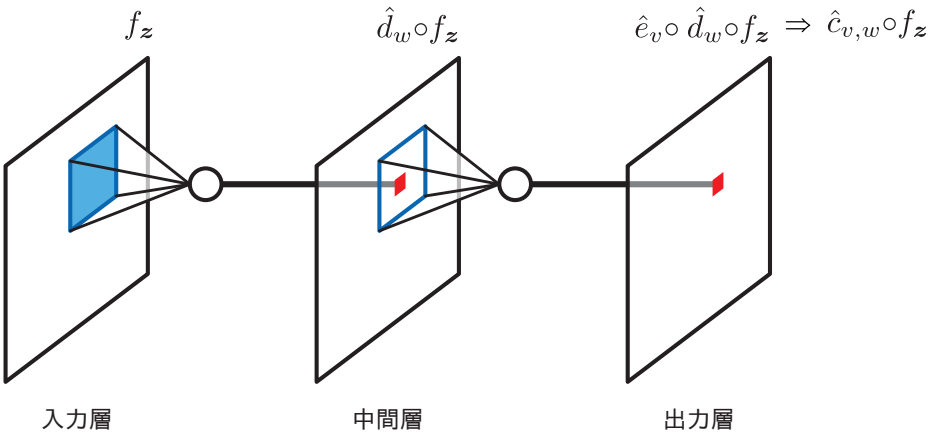


図 4.1　凸凹フィルタ

手書き文字抽出へ適用することができる．

4.1.2　手書き文字抽出のためのネットワーク

手書き文字を抽出する処理は，印刷文字を構成する黒画素のみを除去する処理であるので，印刷文字の黒が紙面の白へ変化するのみであることが理想となる．つまり，すべての画素において，入力画像となる手書きの注釈が記入された文書画像の画素値より，出力画像となる手書き文字のみの画像の画素値の方が大きくなるようにネットワークを構成すべきである．そこで，本研究では，この特徴を考慮し，図 4.2 に示すように，画像 $f_{\boldsymbol{z}}$ 入力に対する凸凹フィルタの出力 $\hat{c}_{v,w} \circ f_{\boldsymbol{z}}$ を，入力画像 $f_{\boldsymbol{z}}$ と比較して，大きい数値

$$r \circ f_{\boldsymbol{z}} = \bigvee \{f_{\boldsymbol{z}}, \hat{c}_{v,w} \circ f_{\boldsymbol{z}}\} \tag{4.8}$$

を出力するネットワーク構成を提案する．

深層ネットワークの研究において，入力画像と理想出力画像の差分を出力するようにネットワークを学習させることで，収束速度の改善，性能の向上を図る残差学習 [27] [28] が提案されている．式 (4.8) で示すネットワーク構造も，入力画像と理想画像の不一致のみを学習させることで，手書き文字の抽出精度の向上が期待できる．以後，本論文では式 (4.8) に示した構造を，最大値を学習する意味で，残差学習に対応させて「最大値学習」と呼ぶ．

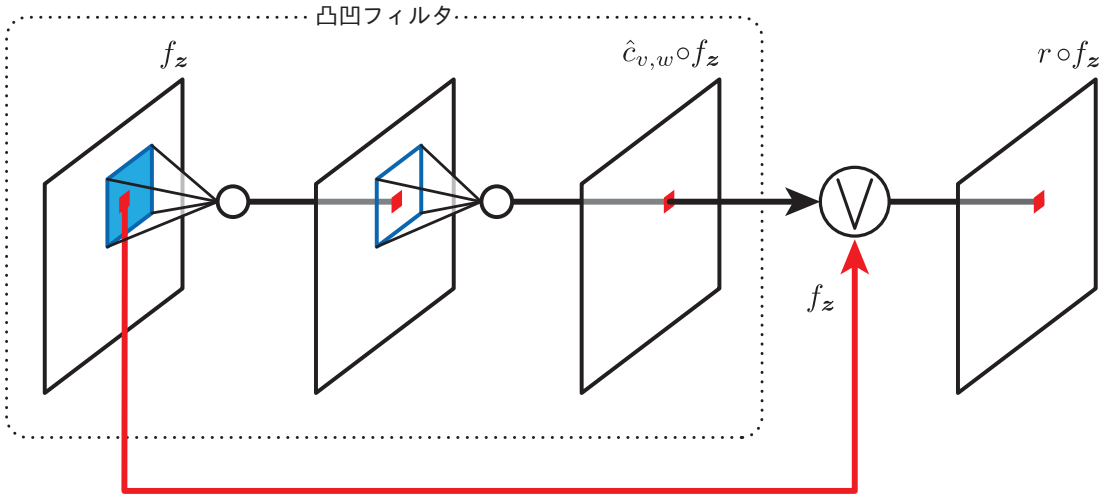


図 4.2　最大値学習フィルタ

4.2　手書き文字抽出ネットワークの学習

本節では，手書き文字を抽出するための Maxout フィルタネットワークのパラメータである荷重パラメータとバイアスの学習方法について述べる．学習に用いる原文書は，2017 年度の画像電子学会学会誌（PDF 版）に掲載された論文を抜き出し，300dpi の解像度のレーザープリンタにより印刷したものである．以降の節では，印刷した文書から，ネットワークの学習データのサンプルとなる，手書きの注釈が記入された文書の画像と手書き文字のみの画像の作成手順について説明した後，本研究で使用したネットワークパラメータの学習の種類及び諸元について述べる．

4.2.1　学習データの作成

学習データの以下のように作成した．書き癖の影響を想定し，14 名に注釈記入を依頼し，ボールペンを使用する 7 名とシャープペンシルを使用する 7 名の 2 グループに分け，印刷した論文に対する手書きの注釈を記入してもらった．手書きの注釈は，1 行あたり任意に選択した 1 単語にアンダーバーを加え，さらに，その文字を下に書き加えることで生成した．

学習に必要となるのは，学習データとなる，原文書に手書きの注釈が記入された文書画

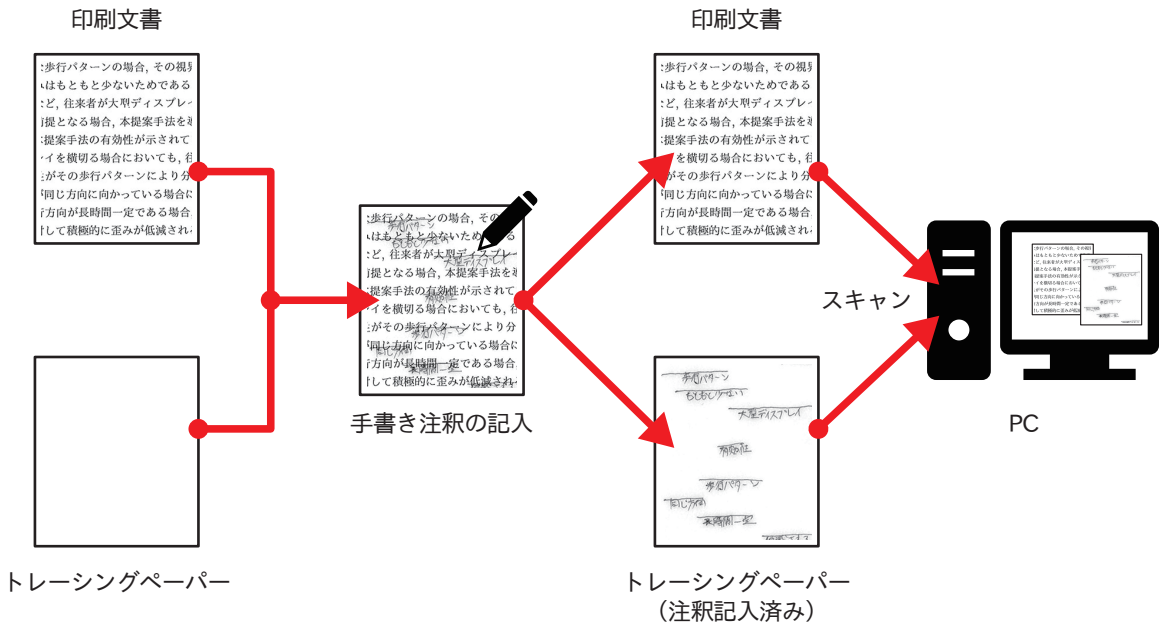


図 4.3　学習データの作成過程

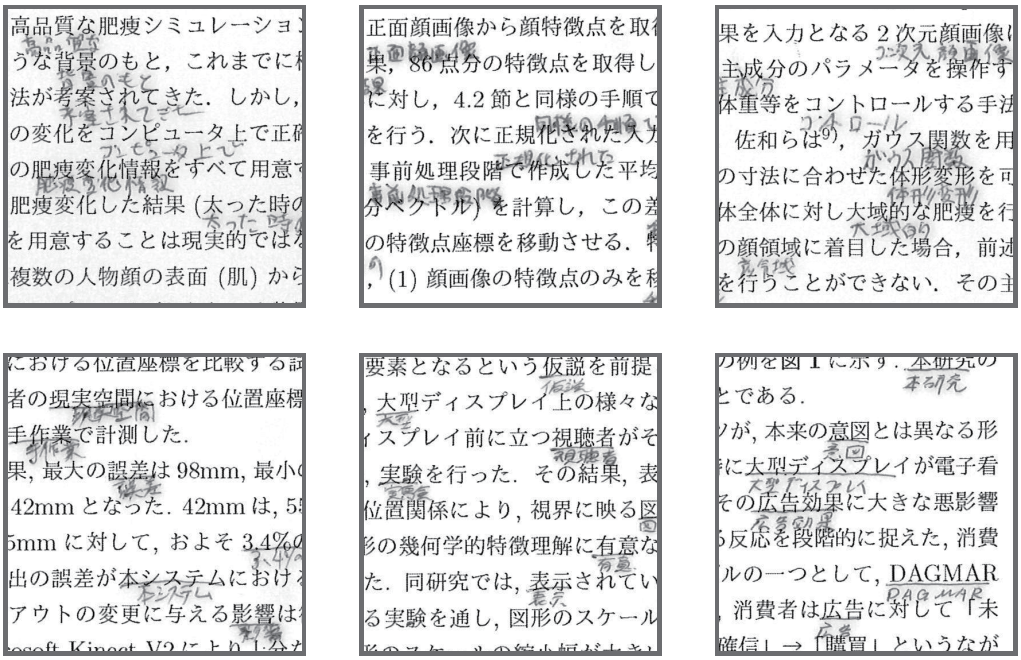
像，それに対する，学習時の教師データとなる理想的な抽出結果として使用する注釈のみが記入された画像の 2 種類であるが，原文書に直接注釈を記入すると，注釈文字のみの文書を作成する際，原文書に記入した注釈と全く同じ注釈を別の紙にもう一度記入してもらう必要が生じてしまう．これは非常に困難な作業であると考えられる．

そこで，一度の注釈記入で済むようにするため，図 4.3 に示すような流れで学習データを作成した．先ず，原文書と同じサイズのトレーシングペーパーを準備し，トレーシングペーパを原文書の上に重ね合わせ，トレーシングペーパー上に注釈を記入してもらい，記入された注釈文字を教師データとして使用することとした．14 名の記入者に，それぞれ 5 ページを担当してもらい，計 70 ページ分のサンプルを作成した．

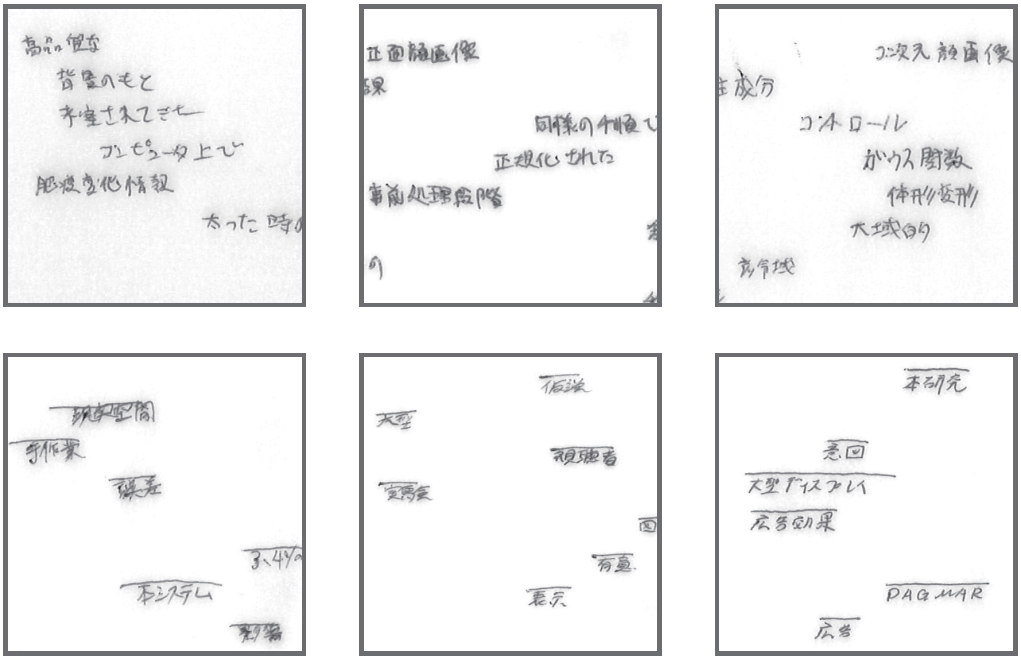
原文書と手書き注釈が記入されたトレーシングペーパーを別々にスキャンして，2480 × 3700 画素，8 ビットのグレースケール画像としてコンピュータに読み込み，それぞれ原文書画像，理想手書き文字画像とした．原文書画像と理想手書き文字画像を，各画素単位にビットごとの論理積演算を適用して合成することで，手書きの注釈が記入された文書の画像を作成した．この画像の各画素値 (0〜255) を入力データとし，これに対応する理想手書き画像の各画素値 (0〜255) を教師データとする．作成した学習データの例を図 4.4 に示す．手書き文字の周囲に若干，影のような部分が存在するが，(a)，(b) に共通であり，印刷文字のみ除去するための学習には影響しないと考えられる．何れもス

キャンした文書画像全体から，ランダムな座標において 512×512 画素の領域を切り抜いたものであり，学習の際には，同様の方法で切り抜いた画像を使用する．

作成した学習データは，14 名のサンプルを，シャープペンシルとボールペンによる記述を行った記入者を 2 名づつ含む G1 セットと G2 セット，3 名づつ含む G3 セットの 3 グループに分け，手書き文字を抽出するテスト用セットとして使用する．更に，この 3 グループから 2 グループずつ選んで，三つの複合セット，G1+G2 セット，G2+G3 セット，G3+G1 セットを作成し，これらを学習用とした．学習用データとテスト用データの組み合わせを複数用意することで，学習結果のデータセットへの依存性を確認する．



(a) 手書きの注釈が記入された文書の画像



(b) 注釈文字のみの画像

図 4.4　学習データの例（512×512 画素）

4.2.2 パラメータの学習

Maxout フィルタネットワークのパラメータである荷重パラメータとバイアスの学習において，本研究ではミニバッチ学習を用いた．ミニバッチ学習は，確率的勾配降下法において 1 回のパラメータ更新において使用する学習サンプルの数を 1 個から複数個に変更し，複数個の学習サンプルから得られた勾配の平均からパラメータを更新する学習法である．勾配降下のステップサイズ，つまり式 (2.20) の ε は，AdaGrad 法 [29] を用いて，t 回目の更新では，

$$\varepsilon_t = \frac{\eta}{\sqrt{\sum_{i=1}^{t} \nabla E_i^2}} \tag{4.9}$$

のように，パラメータ更新回数にしたがって適応的に決定する方法を採用した．ここで，∇E_i は，i 回目の更新における勾配，つまり，損失関数の荷重パラメータによる偏微分値である．1 回のパラメータの更新のために，手書きの注釈が記入された文書画像 $g_{\boldsymbol{z}}^{(i)}$ を入力したときの凸凹フィルタ出力画像と理想手書き文字画像 $f_{\boldsymbol{z}}^{(i)}$ の正規化した二乗誤差

$$E = \sum_i \left(\frac{\sum_{\boldsymbol{z}} \left(\hat{c}_{v,w} \circ g_{\boldsymbol{z}}^{(i)} - f_{\boldsymbol{z}}^{(i)} \right)^2}{\sum_{\boldsymbol{z}} \left(255 - f_{\boldsymbol{z}}^{(i)} \right)^2} \right) \tag{4.10}$$

を損失関数とし，その勾配を用い，全体の二乗誤差が最小となるようネットワークパラメータを更新した．式 (4.10) では，手書き文字が黒で記入されることから，濃淡を反転させた画像の二乗和で正規化を行っている．ここで，i はミニバッチに含まれる学習サンプルに付けられた通し番号であり，1 回のパラメータ更新において，一つのサンプルを使用する確率的勾配降下法の場合は $i = 1$ のみ考慮する．

目的関数 E には，局所的な極小解が存在しており，学習で得られる荷重パラメータとバイアスの解はその初期値に依存する．本研究では，輝度の窪みを除去することができるクロージングフィルタを実現する荷重パラメータを初期値とした [20]．そのため，ウィンドウ内の画素数 $L \times L$ に対し，荷重パラメータとバイアスの組を $K = L \times L$ とし，ウィンドウ内の座標 (m, n)（ただし，$1 \leq m, n \leq L$）における荷重パラメータを，凸フィル

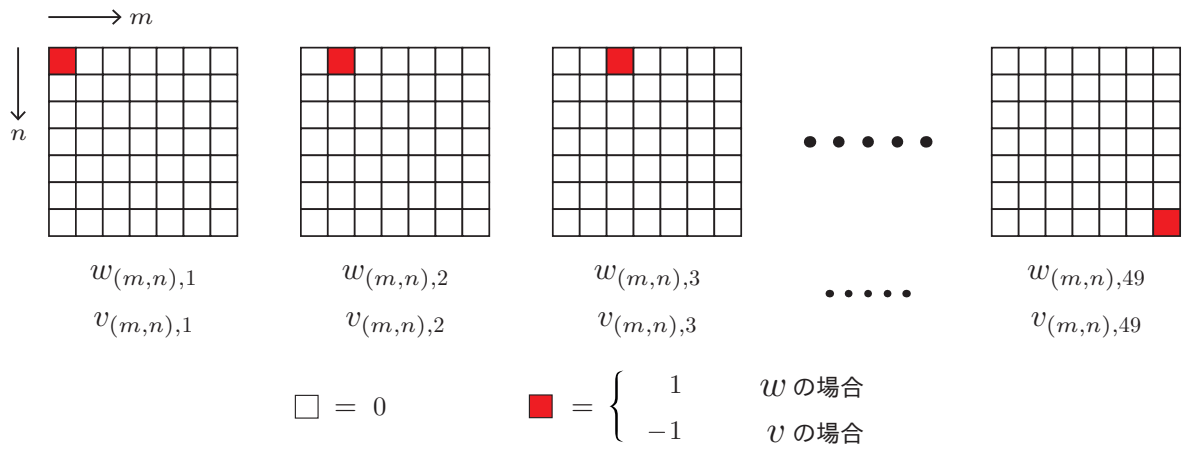


図 4.5　荷重パラメータの初期値設定

タでは，

$$w_{(m,n),k} = \begin{cases} 1 & \text{for} \quad k = (m-1)L + n \\ 0 & \text{otherwise} \end{cases} \tag{4.11}$$

凹フィルタでは，

$$v_{(m,n),k} = \begin{cases} -1 & \text{for} \quad k = (m-1)L + n \\ 0 & \text{otherwise} \end{cases} \tag{4.12}$$

と設定した．これにより，$L \times L$ の内，k によって異なる位置のみで，1（凸フィルタ）または，-1（凹フィルタ）となり，その他は全て 0 となる k 組の荷重パラメータの初期値が与えられる．これを，$L = 7$ として図示したのが，図 4.5 である．一方，バイアスの初期値はすべて 0 とした．

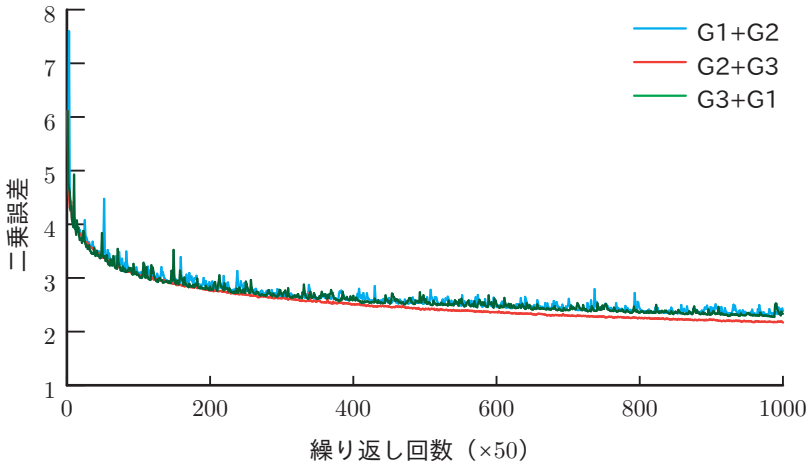
本研究におけるパラメータ学習では，凸フィルタ，凹フィルタともに 7×7 画素のフィルタウィンドウを設定したので，両フィルタともに，荷重パラメータとバイアスの組 $K = 49$ である．したがって，学習すべきパラメータの総数は，凸フィルタ，凹フィルタそれぞれに荷重パラメータは $L \times L \times K = 2,401$ 個，バイアスは $K = 49$ 個であるので，総パラメータ数は $4,900$ 個である．

以上の条件において，学習用として作成した複合セット，G1+G2 セット，G2+G3 セット，G3+G1 セットそれぞれに対し，確率的勾配降下法を用いて凸凹フィルタ単体構成のネットワークパラメータの学習を，また，ミニバッチ学習を用いて凸凹フィルタ単体構成

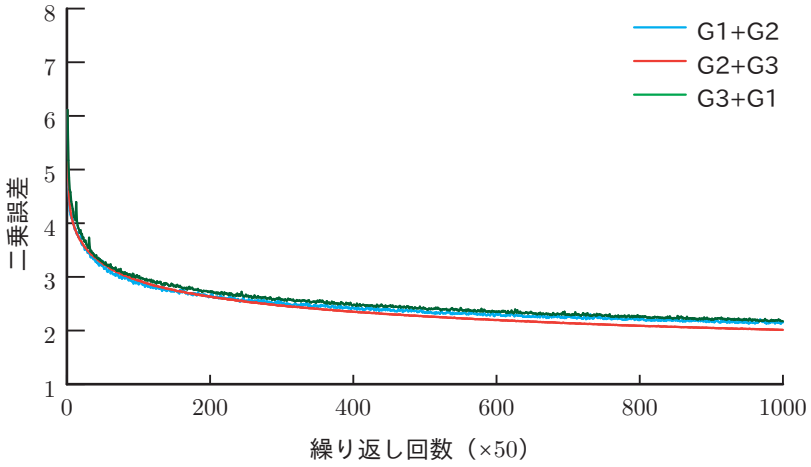
のネットワークパラメータの学習と，4.1.2 節で提案した最大値学習によるネットワークパラメータ学習を実行した．学習の終了を判定するパラメータ更新回数の上限を，それ以上の繰り返しても学習の向上が見られなくなる程度を考慮して，50,000 回に設定した．ミニバッチ学習に関して，ミニバッチに含まれる学習サンプル数（ミニバッチサイズ）は，記入者によって作成された 5 ページ分のサンプル 1 ページあたり 2 個所を想定して 10 とし，1 回のパラメータ更新の際に用いられるミニバッチは，同一記入者のサンプルの中から，図 4.4 に示した例のように，512×512 領域を 10 箇所切り出した画像によって構成される．

3 種類の学習方法を用いたパラメータ学習の反復回数に対する，テスト用理想手書き文字画像と凸凹フィルタの出力の二乗誤差の関係を図 4.6 の学習曲線に示す．二乗誤差は 50 回の反復毎に，あらかじめ 14 名の記入者のサンプルから一つずつ抽出しておいた 14 サンプルに凸凹フィルタを適用して求めた値である．50,000 回 の反復終了時点で，確率的勾配降下法の G1+G2 セットと G3+G1 セットにばらつきがあるが，これ以上反復しても有意な向上は見られなかった．

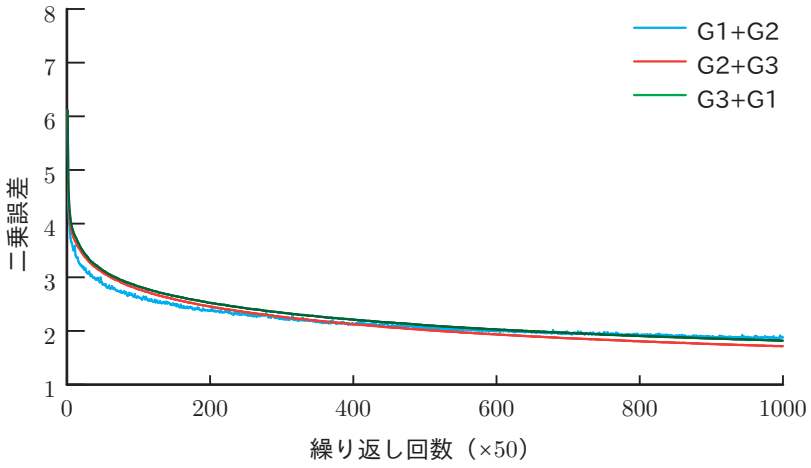
それぞれの学習曲線を詳細に観察してみると，G1 セットを含む学習セットでは，G1 セットを含まない学習セットよりも激しいばらつきが見られる傾向にあるため，G1 セットに特異なサンプルが含まれているものと推測されるが，図の (b) と (c) では，ばらつきが抑えられているばかりでなく，(a) に比べて学習終了時の誤差が小さくなっているので，ミニバッチ学習が有効に働いていることがわかる．また，(c) に示される，本研究で提案した構成のネットワークの学習が最も効果が高くなり，提案した構成が有効であることが確認できる．



(a) 確率的勾配降下法



(b) ミニバッチ学習



(b) ミニバッチ + 最大値学習

図 4.6　学習曲線

4.3 手書き文字の抽出と評価

本節では，Maxout フィルタネットワークによる手書き文字の抽出結果を示す．3 種類の学習セットに対して，それぞれ 3 種類の学習方法で学習した，計 9 種類のネットワークを用いて，テスト用の G1 セット，G2 セット，G3 セットの手書きの注釈が記入された文書画像を入力画像とし，手書きの注釈の抽出テストを実行した．評価には，SNR(Signal-to-Noise Ratio) を用いた．SNR は抽出結果の画像（$\hat{c}_{v,w} \circ g_{\boldsymbol{z}}$）とそれに対応する理想手書き文字画像（$f_{\boldsymbol{z}}$）から，両者の誤差を求め，これと理想手書き画像を反転した画像との比より，

$$\mathrm{SNR} = -10 \log_{10} \left(\frac{\sum_{\boldsymbol{z}} \left(\hat{c}_{v,w} \circ g_{\boldsymbol{z}} - f_{\boldsymbol{z}} \right)^2}{\sum_{\boldsymbol{z}} \left(255 - f_{\boldsymbol{z}} \right)^2} \right) \tag{4.13}$$

と計算した．これは，文字情報は黒で記述されており，最大輝度からの差分で情報を表すと考え，この定義では，信号のエネルギーを，最大輝度からの差分で計算している．SNR の平均値，最大値，最小値を**表** 4.1〜**表** 4.3 にまとめた．

学習結果に示したとおり，G2+G3 セットの学習効果は，G1+G2 セット及び G3+G1 セットより高い．その傾向がテストサンプルの SNR にも若干現れていることは，各表の G2+G3 カラムにおけるほとんどの値が G1+G2 及び G3+G1 カラムの値より大きいことからわかるが，テストセットが学習セットに含まれるかどうかにかかわらず，概ね同程度の SNR になっている．この結果より，記述者 4 名から 6 名で，記述者にほぼ依存せず，印刷文字を除去できるフィルタが学習できることがわかる．

次に，学習セットに含まれないテストセットに対する結果（**表** 4.1 の G2+G3 カラム，**表** 4.2 の G3+G1 カラム，**表** 4.3 の G1+G2 カラム）に着目し，学習方法の違いによる手書き文字抽出性能を SNR の平均値で比較すると，確率的勾配降下法，ミニバッチ学習より，提案法であるミニバッチ + 最大値学習の方が大きくなっていることがわかる．SNR の増加に関しては，確率的勾配降下法から提案法では平均で 1.2 dB の増加，ミニバッチ学習から提案法では平均で 0.77 dB の増加となっており，提案法は効果的であると考える．

表 4.1　G1 セットのフィルタ結果画像の SNR (dB)

学習方法 \ 学習セット		G1+G2	G2+G3	G3+G1
確率的勾配降下法	平均	7.91	8.02	7.96
	最大	9.52	9.51	9.54
	最小	4.36	4.19	4.32
ミニバッチ学習	平均	8.51	8.45	8.37
	最大	10.08	9.98	9.95
	最小	4.77	4.41	4.69
ミニバッチ + 最大値学習	平均	9.34	9.38	9.16
	最大	11.09	11.21	11.03
	最小	5.19	4.70	4.73

表 4.2　G2 セットのフィルタ結果画像の SNR (dB)

学習方法 \ 学習セット		G1+G2	G2+G3	G3+G1
確率的勾配降下法	平均	8.23	8.38	8.26
	最大	12.65	12.80	12.73
	最小	5.07	5.22	5.06
ミニバッチ学習	平均	8.74	8.77	8.61
	最大	13.17	13.27	13.13
	最小	5.52	5.51	5.38
ミニバッチ＋最大値学習	平均	9.47	9.61	9.29
	最大	13.93	14.10	13.58
	最小	6.12	6.25	5.86

表 4.3　G3 セットのフィルタ結果画像の SNR (dB)

学習方法 \ 学習セット		G1+G2	G2+G3	G3+G1
確率的勾配降下法	平均	8.27	8.43	8.33
	最大	12.54	12.59	12.62
	最小	1.70	1.77	1.75
ミニバッチ学習	平均	8.77	8.83	8.69
	最大	13.11	13.06	13.10
	最小	1.99	1.92	1.91
ミニバッチ＋最大値学習	平均	9.49	9.60	9.39
	最大	14.09	14.10	13.94
	最小	2.31	2.01	2.29

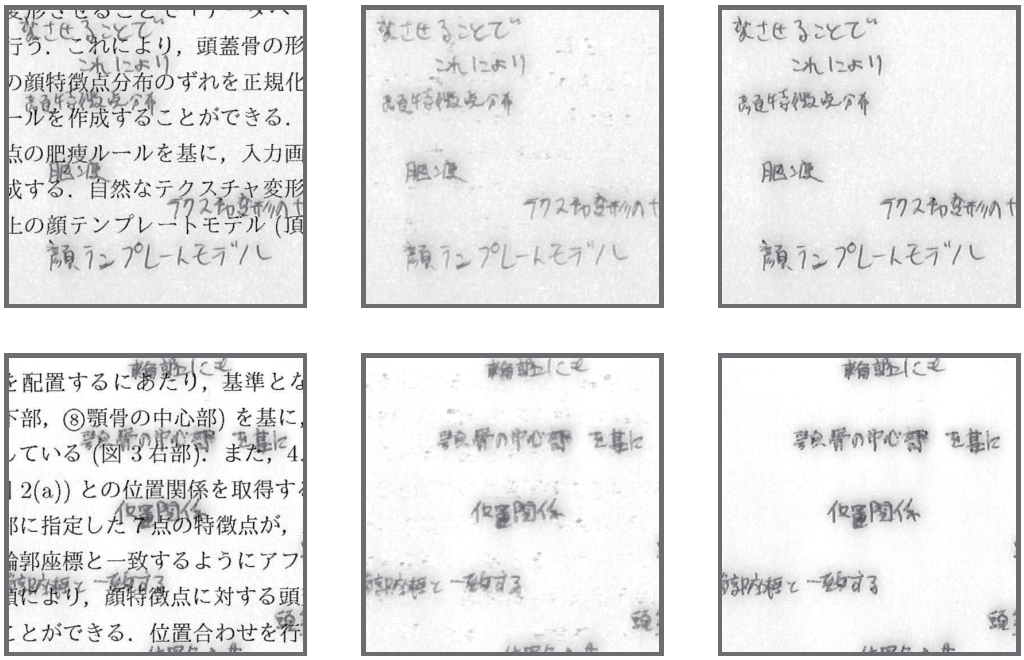
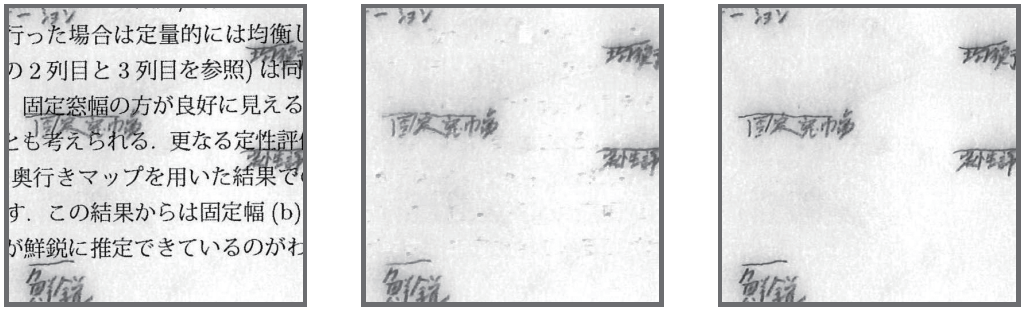


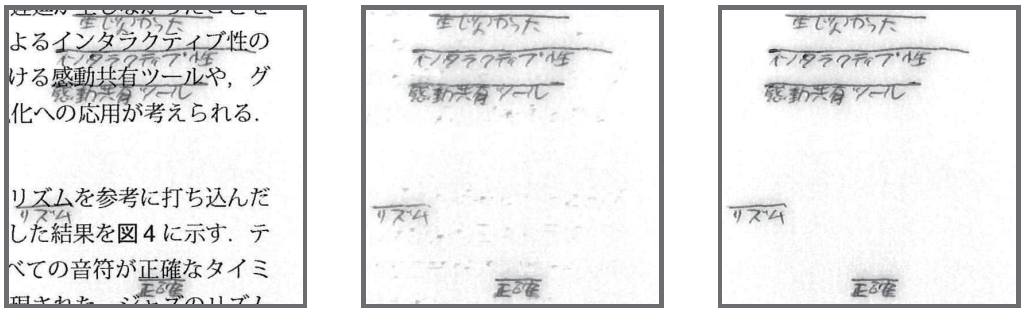
図 4.7　処理結果の例
（各行左から入力画像，抽出された手書き文字，理想手書き文字）

提案法による処理結果に関し，個々の画像について確認する．SNR 最大となったのは，表 4.3 に示されている，G1+G2 セットを用いて学習し，G3 セットを用いてテストした結果の一つ（SNR:14.09 dB）であった．これが最大となったのは，平均値が同程度であることを考慮すれば，テストセットの差によるものであるといえる． その画像を図 4.8(a) に示す．左から，入力画像，抽出された手書き文字，そして理想手書き文字である．活字部分を完全には除去し切れずに薄く残ってしまった個所や，手書き文字が擦れて薄くなってしまった個所が若干見受けられるが，理想手書き文字と比較しても，目視した限りでは，判別可能な程度に収まっていることがわかる．薄く残っている活字の一部は，主に複雑な漢字の一部や，それほど複雑ではない場合でも，「数」や「奥」などに含まれている「米」のように複数画が集中しているところで発生している．また，(b) に示すボールペンで記入したサンプルにおいても，シャープペンシルのサンプルと同様に良好な結果が得られた．

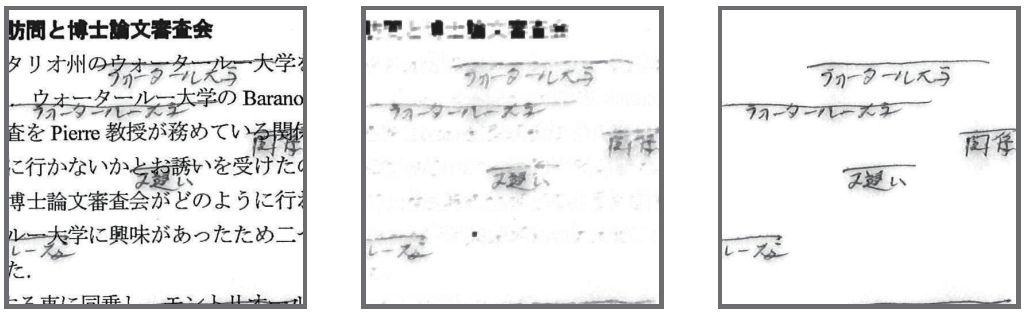
次に，SNR 最小となった結果であるが，これも最大の例と同じく G1+G2 セットを用いて学習し，G3 セットを用いてテストした結果の一つ（SNR:2.31 dB）であった．その画像を図 4.8(d) に示す．活字が残っているのは (a)，(b) と同程度であるが，全体的に手



(a) SNR最大（シャープペンシル, 14.09 dB）の例



(b) SNR最大（ボールペン, 11.15 dB）の例

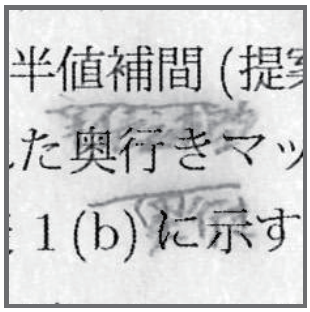


(c) SNR最小（2.31 dB）の例

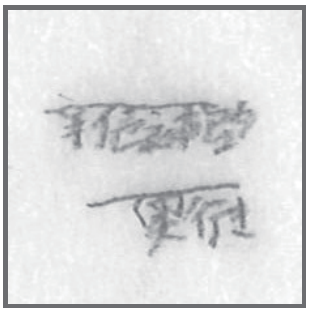
図 4.8　SNR による評価

書き文字が薄くなってしまい，擦れも顕著である．さらに，本文に使用されている明朝体以外のフォント，例えばゴシック体の一部が除去されずに抽出結果に表れている．このサンプルの結果は，同一記入者の他のサンプルと比較しても大きく SNR が低下している結果であり，このサンプルを除いた最小値は，表 4.1，表 4.2 に示された最小値と同程度であった．

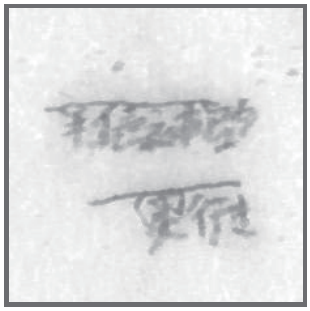
提案法では，文章の大部分に使われるフォントに対して除去能力を得ることができる



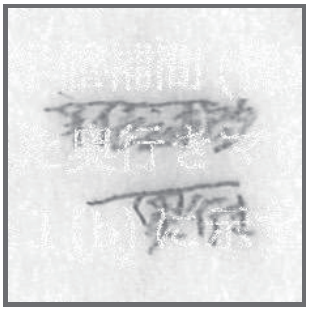
(a) 入力画像



(b) 理想手書き文字



(c) 提案手法



(d) 差分による手法

図 4.9　活字と手書き文字の重なりの処理

が，まれに表れるフォントに対しては除去能力が低い．より多くのフォントが含まれる印刷文書への対応は今後の課題である．

ここで，活字と手書き文字が重なっている場合について検証する．図 4.9(c) の例を見ると，活字と手書き文字が重なっている場合でも，若干の欠けはあるものの，理想手書き文字と比較して遜色の無い抽出が可能であることがわかる．参考文献 [15] では，オリジナルの文書画像との差分から手書き（注釈）部を抽出する方法が検討されている．差分を求めた結果と，提案法の結果を比較すると，重なりの無い部分における手書き文字抽出能力は高いと言えるが，図 4.9(d) に示すとおり，活字領域として除去した部分に重なっている手書き文字は欠けてしまう．これに対して，提案法ではオリジナルの文書画像を必要とせず，文字の重なりにも対応できる．

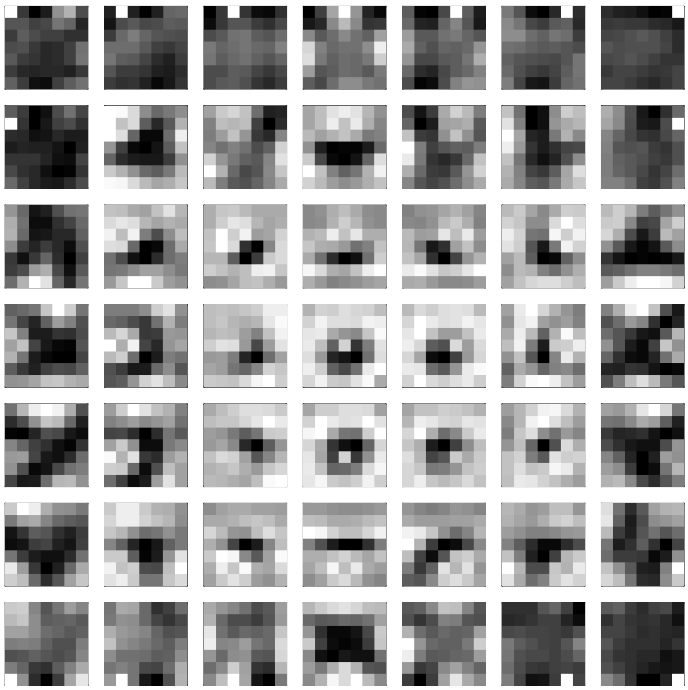


図 4.10　凸フィルタの荷重パラメータ $w_{(m,n),k}$ の分布例

最後に，Maxout フィルタネットワークの荷重パラメータについて考察する．ここでは例として，G1+G2 学習セットを使用して，ミニバッチ＋最大値学習した凸フィルタ及び凹フィルタの荷重パラメータを可視化して図 4.10 に示す．荷重パラメータは，7×7 個を一組として，49 組であるので，7×7 画素からなる 49 枚の画像として可視化した．同図において，左上の画像から右下の画像にかけて，$k = 1, 2, \ldots, 49$ であり，(m, n) は，各画像内での座標を表している．

フィルタの荷重パラメータの初期値は，初期状態でモフォロジカルフィルタのクロージングフィルタを実現するために，図 4.5 に示したとおり，一つの畳み込みで一つの荷重パラメータのみを 1 に設定した．同図の左上の画像の左上の画素のように目立った画素が見受けられる場合があるが，これは $w_{(m,n),k}$ の初期設定において 1 に設定されていた荷重パラメータが，学習終了後にも周辺に比べて大きな値として残った画素である．

図 4.10 では，初期値を 1 に設定した荷重パラメータ以外について，最大値を白，最小値を黒として表示している．図に示した画像の中には，印刷文字のパーツ，枝分かれや交差などの特徴を表現していると考えられる黒い部分を持つものが数多く存在していることがわかる．このことから，ネットワークパラメータは活字の形状を分割して学習しており，

荷重パラメータの負値で重み付けをすることで，活字部分を無視することによって手書き文字の抽出を実現していると考えられる．

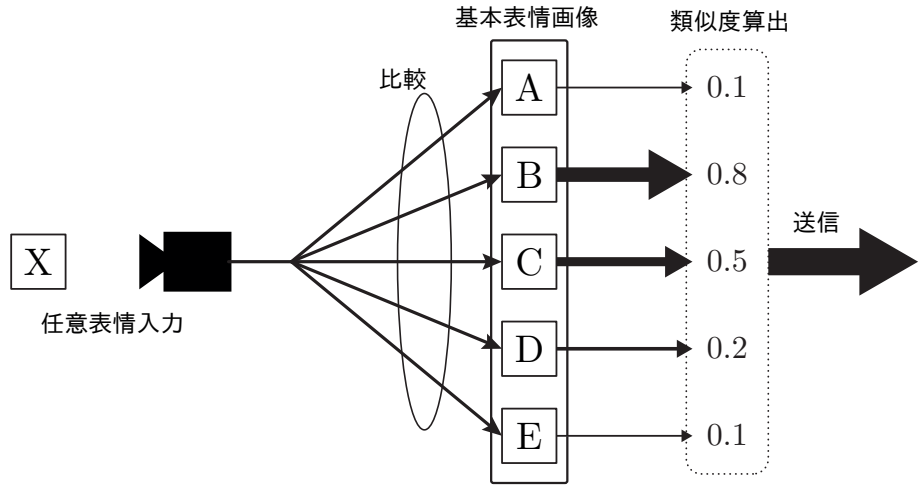
第 5 章　顔表情知的符号化による極低ビットレート伝送

本章では，1.3 節で示した，知的符号化による極低ビットレート伝送について述べる．テレビ電話やテレビ会議システムなどの通信の際に，話者の表情を分析することによって低ビットレート化を実現する知的符号化を提案する．映像の圧縮技術が進歩した現在でも，映像情報の伝達は，コストのかかる処理であり，圧縮率と画質のトレードオフの関係が存在している．しかし，テレビ電話などにおいては，話者の表情がわかることが重要である．そこで，カメラで撮影された送信側の話者の表情を分析することによって得られた少数の数値を送信し，受信側では，受信された数値に従って，送信話者の表情を合成することができれば，低ビットレート通信が可能になる．

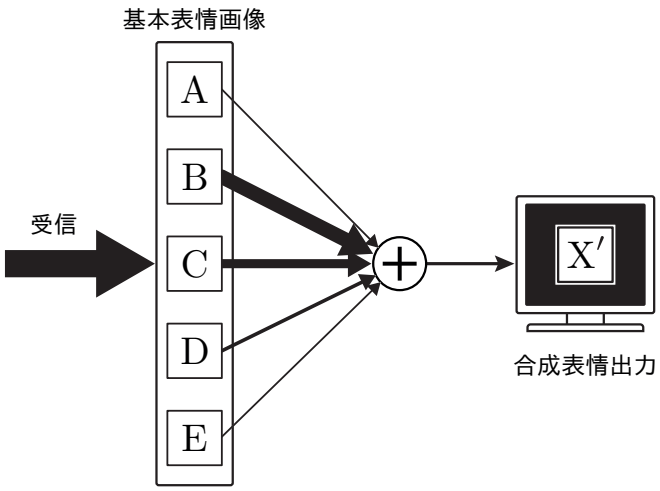
本研究では，ニューラルネットワークを用いた表情の分析とモーフィングを用いた表情の合成を基礎とした，顔画像の知的符号化手法を提案する．この手法の特徴は，送信話者の特徴的な表情を基本表情として選び，送信受信双方で共有しておくことである．送信側の表情分析では，基本表情を階層型ニューラルネットワークに学習させておく．このニューラルネットワークによって，カメラで撮影された刻々と変化する話者の表情を分析し，基本表情との類似度を計算し，随時その結果を送信する．受信側では，送信話者の表情の分析結果である類似度を基に，モーフィングを用いて基本表情を合成する．

実験によって，受信側で出力される合成表情が，送信側の話者の表情変化に追従して変化することを確認する．また，提案手法による顔画像の送信に必要なデータ構造（ビット数）に関する検討も行い，画像のまま送信する場合に比べて，極超低ビットレートを実現できることを確認する．

本章の構成は次の通りである．5.1 節で，知的符号化の概要を述べる．5.2 節で，送信側で送信話者の表情を分析する方法について，5.3 節で，表情の合成方法についてそれぞれ述べる．5.4 節で処理結果をまとめる．



(a) 表情認識部（送信部）



(b) 表情合成部（受信部）

図 5.1　システム概要

5.1　知的符号化の原理

ここで提案する手法は，図 5.1 に示すとおり，表情認識部と表情合成部で構成されており，前者は送信部の一部に，また，後者は受信部の一部になっている．両方に共通の知識として，送信話者の特徴的な表情 (以下「基本表情」，図中 A〜E で示す) の画像を記憶している．

表情認識部は，カメラから入力された話者の任意表情 X を認識することによって，この表情が，それぞれの基本表情とどの程度似ているかを類似度として出力する．図の場合では基本表情が 5 種類なので，5 個の類似度が得られており，表情 B にもっとも似ており，次いで表情 C に似ているという結果を示している．類似度は，ひとつの基本表情対してひとつの数値であるので，このまま送信しても，伝送系の負荷にはならない．また，著しく低い類似度は，合成される表情には影響しないので，送信しなくてもよいと考えられる．そこで，例えば，図のように上位二つのみ送信することにすれば，送信情報は更に少なくて済む．

表情合成部では，表情認識部から送られた類似度を用い，認識部と同じ基本表情からモーフィングによって，認識部に入力された任意表情に近い表情 X' を合成して，画面上に表示する．この任意表情入力から合成表情出力までの処理を，次々と繰り返し，時々刻々と変化する表情の動画として送信しても，1 フレームあたり伝送される情報はごく少数の数値であるので，極低ビットレートでの人物顔表情画像の通信が可能になる．

5.2　表情認識部

表情認識部では，ニューラルネットワークによって表情の認識を行う．顔画像の認識では，直交した基底顔画像を求める手法 [30][31] やポテンシャルネットと KL 展開を用いた手法 [32] 等が提案されており，高い精度でこれらを認識できることが知られている．しかし，前者の処理では，3 次元モデルを変形し，認識すべき入力顔画像に整合させる必要があるが，人間の顔の造作を表す特徴点の抽出の困難性を考慮すると，実時間処理には向いていないと考えられる．同様に，後者もポテンシャルネットを認識すべき入力顔画像に合わせて変形させる処理が必要となるため高速処理は望めない．そこで，提案するシステムの認識部には，3 層パーセプトロンのバックプロパゲーションアルゴリズムによる学習を行ったニューラルネットワークを用いた．ニューラルネットワークは，学習には膨大な時間がかかるが認識は高速に行えるので，実時間処理に適している．

5.2.1　基本表情の学習

学習に用いる基本表情画像は，表情の特徴が現れているものを選ぶ．図 5.2 に基本表情の例を示す．実際に伝送されるのは頭部全体の画像であるが，学習及び認識における

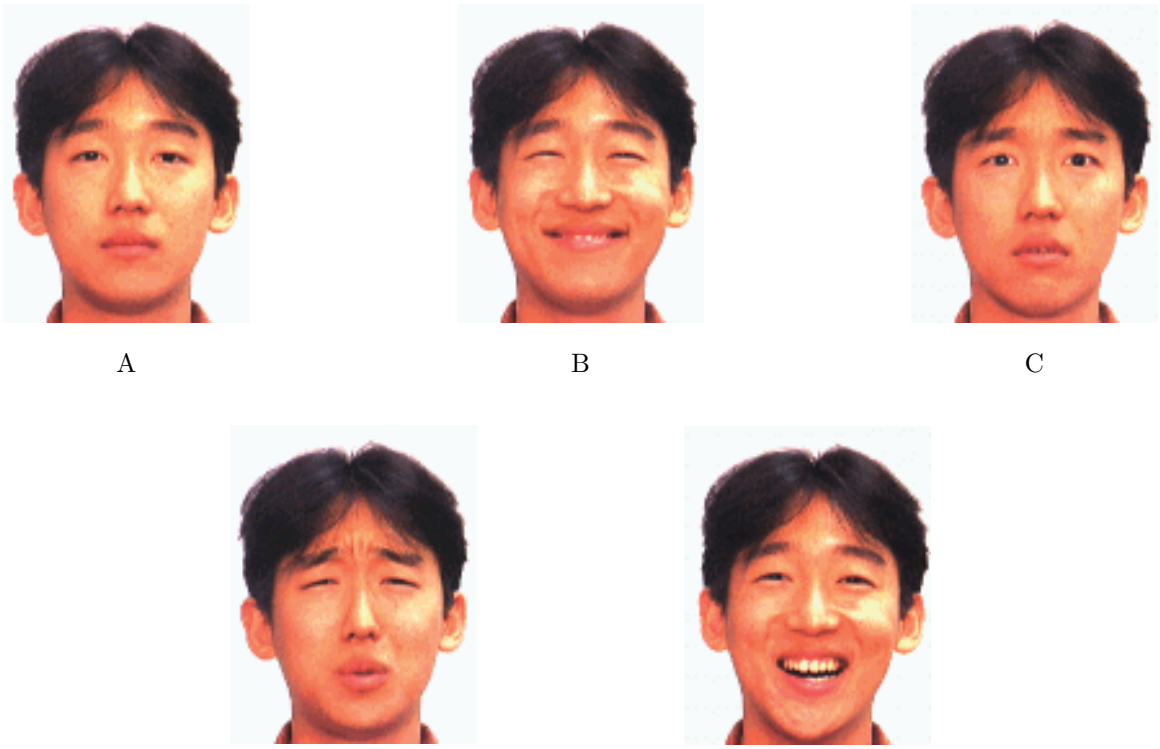


図 5.2　基本表情例

ニューラルネットワークへの入力の際には，表情のよく現れている部分を使用すればよいので，図 5.3 に示すように，

1. 両黒目の中心間の距離を X 画素とする
2. $1.2X$ 画素の正方形を，鼻の頂点を中心に設定する
3. 正方形内を顔領域とする

という条件を用いて，図 5.4(a) のように顔領域を切り出し，128 × 128 画素に規格化した画像を使用する．尚，本研究においては，顔領域の切り出し処理は正確に実行されているという条件が必要であるため，黒目の中心及び鼻の頂点の位置検出は手動で行っている．

入力ユニットに与える顔画像の特徴ベクトルは，大別すると，2 種類考えられる．ひとつは，顔の構造に関する知識，つまり，造作 (目・鼻・口等) の特徴点を抽出して，それらの位置関係をベクトルとして用いるものであり，もうひとつは，顔画像画素の濃淡パターンを用いるものである．

前者の手法は，顔画像から各造作の特徴点を抽出することが困難であることが指摘され

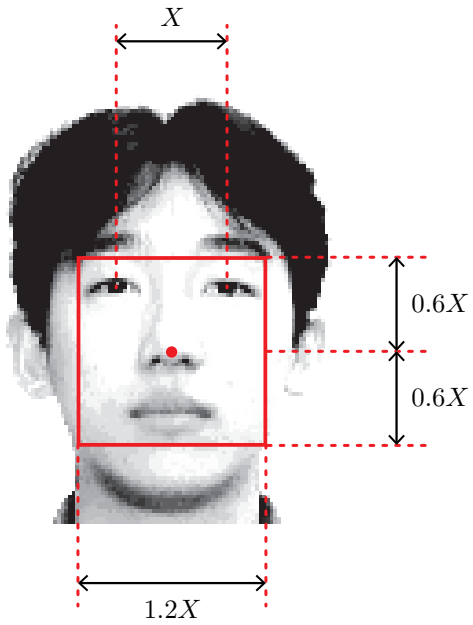


図 5.3　顔領域の切り出し方

(a) 切り出した顔領域　　(b) モザイク化

図 5.4　ニューラルネットワーク入力データの作成

ており [33]，後に実時間処理を実現しようとする際，フレームごとにこれら特徴点を抽出しなければならないことによる処理効率の低下が発生することが予想される．また，後者の手法は，特徴ベクトルの次元数が大きく冗長になると共に，撮影条件の影響が反映されるという問題が発生する可能性がある．しかし，ニューラルネットワークを用いた表情認識においては，入力顔画像をモザイク処理して，それぞれのモザイクブロックの濃淡パターンを特徴ベクトルとすることで，撮影条件の影響の回避，及び次元数を少なくする手法が提案されている [34]．

表 5.1　教師信号

出力層ユニット	基本表情				
	A	B	C	D	E
ユニット A	0.9	0.1	0.1	0.1	0.1
ユニット B	0.1	0.9	0.1	0.1	0.1
ユニット C	0.1	0.1	0.9	0.1	0.1
ユニット D	0.1	0.1	0.1	0.9	0.1
ユニット E	0.1	0.1	0.1	0.1	0.9

そこで，本手法では，入力に基本表情のモザイクブロックの濃淡値を 0.0〜1.0 で表したものを用いた．モザイクのブロック数は，少なすぎると必要な情報が失われ過ぎ，多すぎるとモザイク化の意味が無くなるが，本手法においては，小杉の例 [34] に従い 12×12 を採用した．図 5.4(a) の切り出した顔画像からモザイク画像を生成した例を図 5.4(b) に示す．

また，出力層は，学習する基本表情ひとつに対して 1 ユニットを割り当てる．従って，ニューラルネットワークの構成は，入力ユニット数 144 個，出力ユニット数 5 個とした．中間層ユニット数は，予備実験の結果から 5 個とした．学習は，バックプロパゲーションアルゴリズムを用い，基本表情を入力したときに，それに対応する出力ユニットが 0.9 を出力し，それ以外が 0.1 を出力するように行った (表 5.1)．学習の終了条件は，各出力層ユニットの出力値と対応する教師信号値との誤差の絶対値がすべて 10^{-6} 未満となるまでとした．

5.2.2　類似度の算出

基本表情を学習したニューラルネットワークを持つ表情認識部に，各フレームの任意表情を入力すると，この任意表情に類似した基本表情に対応する出力層ユニットの出力値が大きな値として出力される．入力される表情画像は，学習時と同様に，切り出し・モザイク処理を施した画像であるが，これらの表情画像の照明条件は，基本表情画像が撮影された時と異なっているため，両画像の明度差は異なっている場合がほとんどである．この明度差は，モザイク処理で吸収しにくいため，認識に影響してしまう．そこで，両者の明度の平均を近づけるため，入力画像から切り出された顔領域画像の座標 (x, y) の画素の明度

A

B

C

D

E

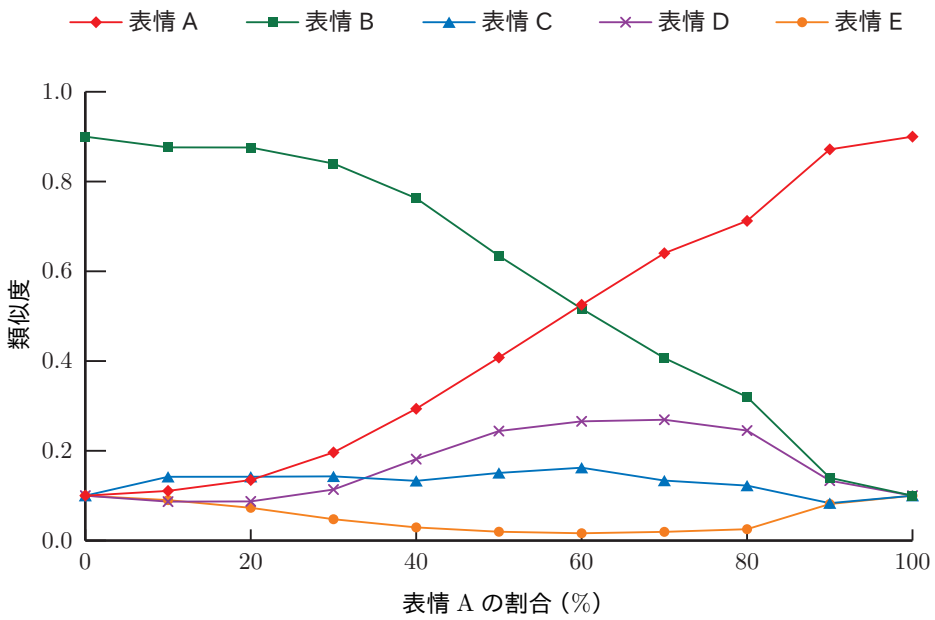
図 5.5　基本表情その 2

値 $I(x,y)$ から，両者の明度平均の差 V を引く処理

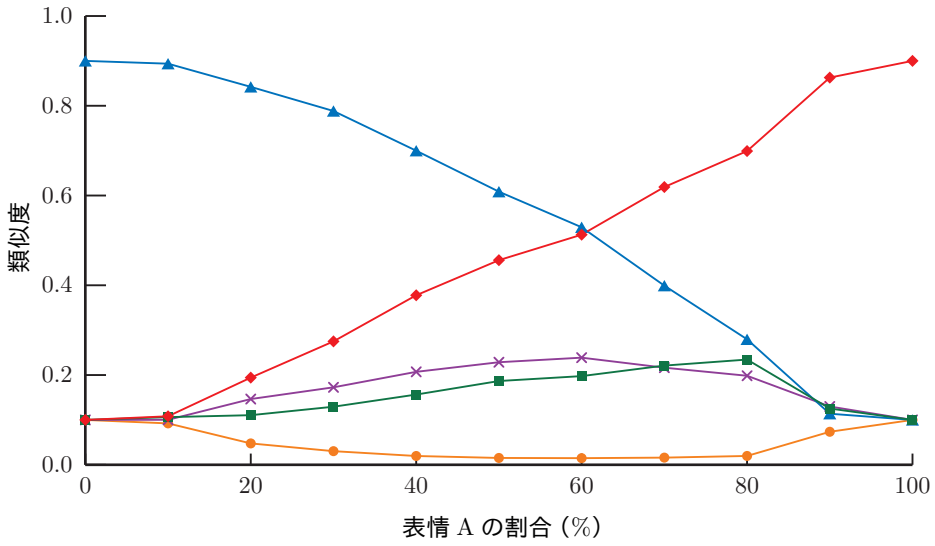
$$I'(x,y) = I(x,y) - V \tag{5.1}$$

を用いて，明度補正を行い補正明度値 $I'(x,y)$ を得た．このような前処理を経て得られたニューラルネットワーク入力値を用い出力を得る．このときの出力値を，任意表情と基本表情それぞれとの類似度として用いる．

次に，ニューラルネットワークの出力値を類似度と定義することの妥当性を検証するため，中割表情の認識実験を行った．中割表情は，基本表情 5 つの内二つを選ぶ組み合わせ 10 通りについて，それぞれの割合を 10% 刻みで変化させモーフィングする事によって生成し，切り出し・モザイク処理を施した後，ニューラルネットワークに入力した．基本表情として，図 5.5 に示す物を用いた場合について，表情 A との組み合わせ 4 つの，出力結果のグラフを図 5.6 および，図 5.7 に示す．いずれのグラフも，表情 A の割合が増えるに従い，その類似度は増加しており，それに伴い，他方の表情 (B〜E) の類似度は減少している．また，組み合わせた二つ以外の表情の類似度は，低い値を保っている．他の組み合



(a) A と B の組み合わせ

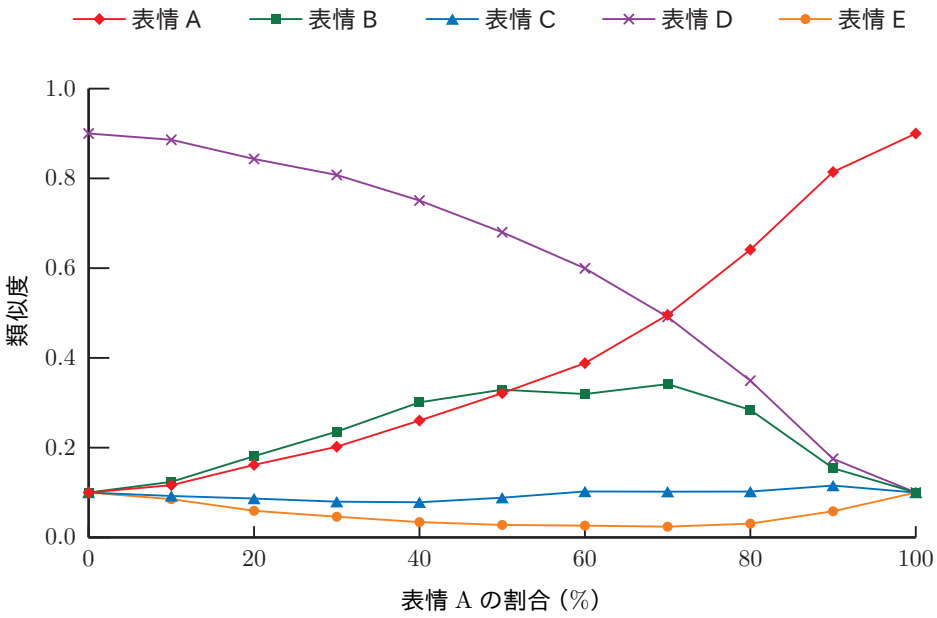


(b) A と C の組み合わせ

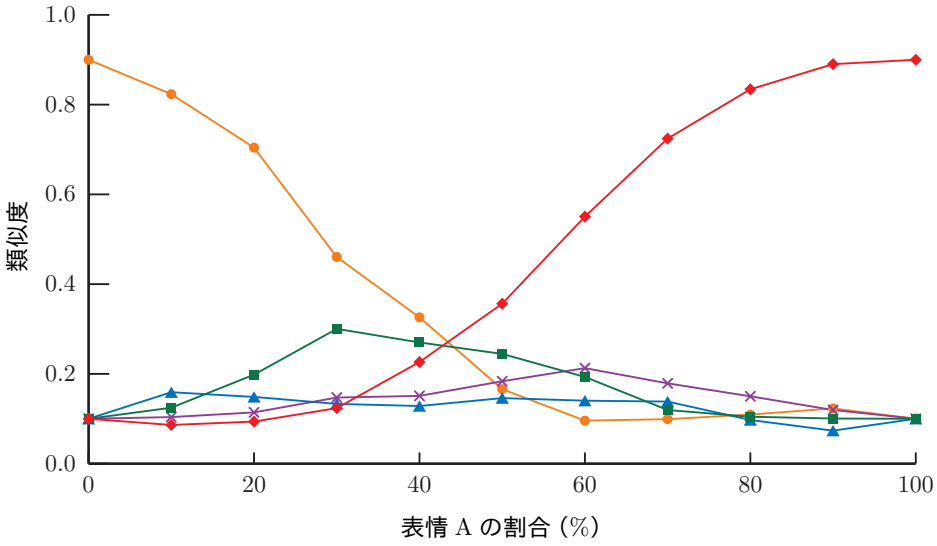
図 5.6　中割表情認識実験の結果（A 対 B,C）

わせに関しても同様の傾向を示しており，従って，表情 A から他表情 (B〜E) への変化は，概ね認識されており，この傾向に従っていれば，ニューラルネットワークの出力値を各基本表上との類似度と定義することは妥当であると考えられる．

しかし，図 5.7(b) のように，2 表情以外の表情の類似度が高くなっている場合があっ



(a) A と D の組み合わせ



(b) A と E の組み合わせ

図 5.7　中割表情認識実験の結果（A 対 D,E）

た．(b) において，表情 A の割合 30 %での入力画像は，図 5.8(a) であった．表情 A と表情 E の中割画像であるにもかかわらず，表情 B が 2 番目になっている．そこで，この結果を用いて，改めて表情 A と表情 B をモーフィングした結果が，図 5.8(b) であるが，(a) と (b) の表情は酷似していることがわかる．これは，基本表情が，AB,AE いずれの組

(a) 入力画像

(b) 再モーフィング

図 5.8　入力・再モーフィング画像の比較

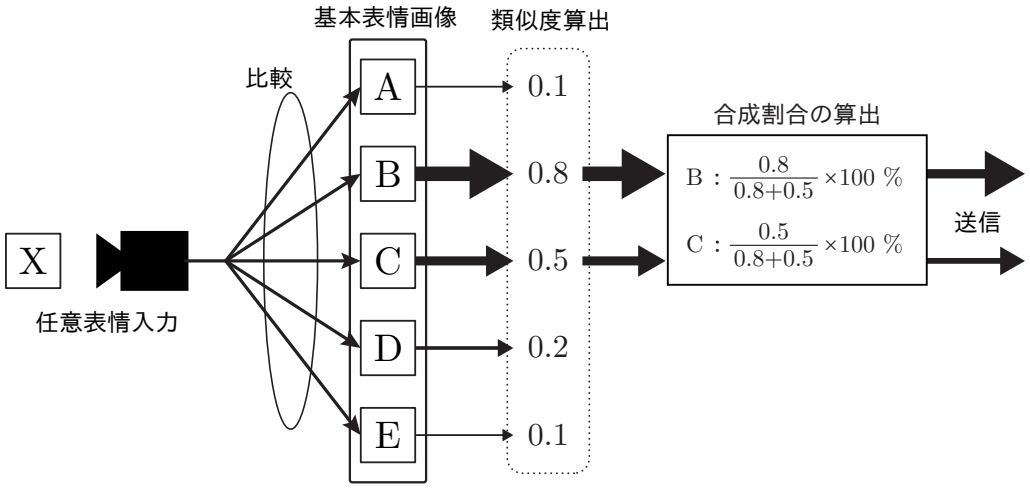
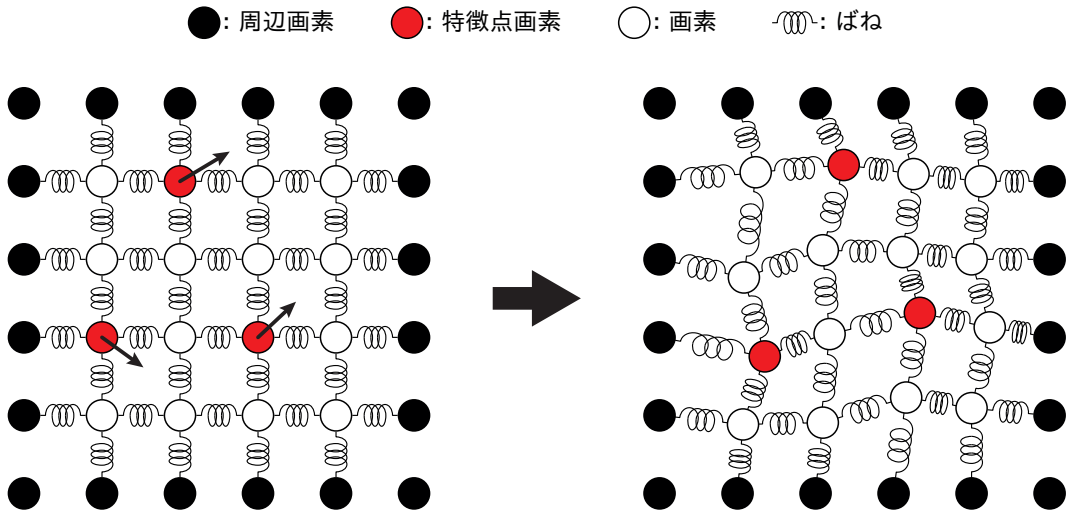


図 5.9　合成する 2 表情の割合

み合わせからでも類似の表情が生成できるような，5 表情であったためであると考えられる．実際，どちらの組み合わせが選ばれても同じ表情になるので，問題にはならないと考えられる．なお，想定される実システムでは，類似度の上位二つから，図 5.9 のように求められる合成割合と，それに対応する基本表情の識別情報を送信する．



(a) 初期状態　　(b) 変形したメッシュ

図 5.10　スプリングメッシュ法

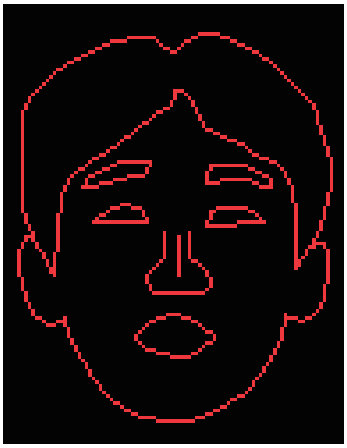
5.3　表情合成部

表情合成部では，受信した情報をもとに選択された二つの基本表情と，その合成割合を用いてモーフィングを行うことによって表情を合成する．ここで必要となる基本表情画像は，後述するモーフィングのための情報を付加して，あらかじめ受信部に送信済みであるとする．モーフィングにおける画像の変形処理であるワーピングには，スプリングメッシュによる手法 [35] を用いた．

これは，図 5.10(a) に示すように，隣接する画素同士をバネで連結した画像モデルとして表す．図中に灰色で示す特徴点画素を強制的に移動させると，周囲の画素がバネを通して力を受けて移動する事によって，画像の変形が生じる．ただし，周辺画素 (図中の黒丸) は，固定されており，動かないもとのする．変形の様子を図 5.10(b) に示す．

ここで，モーフィングに用いる両画像の対応点となる特徴点画素をワープ先の点に移動することにより，ワーピングを実現することができる．特徴点には，目・口の周囲等の表情の特徴が現れる画素を選べばよい．前述のように，顔画像から目・口といった造作の特徴点を抽出するのは困難であるが，本手法では，基本表情にあらかじめ設定しておくだけでよく，フレーム毎に設定する必要はないので，手動で設定してもよく，実時間処理の障害とはならない．

(a) 特徴点入力画像



(b) 特徴点のみの画像

図 5.11　特徴点の入力

特徴点の設定は，マウスなどのポインティングデバイスで，画像上で表情を表す部分をポイントすることにより，連続する線分として指定し，線分上の画素座標を記憶する事で行った．特徴点を入力した画像と，入力された特徴点画素のみを示した画像を図 5.11 に示す．この位置にある画素が，図 5.10 の中で灰色の丸で示した画素に相当する．次に，二つの基本表情 (A:平常顔，B:笑い顔) を用い合成割合を 20 %刻みでモーフィングを行い，中割顔を生成した例を図 5.12 に示す．平常顔から，笑い顔への変化が再現されていることが分かる．

5.4　知的符号化の結果

本手法を用いた実験結果について述べる．基本表情を学習してある人物の会話を撮影し，ビデオキャプチャボードより計算機に入力し，これを任意入力表情とした．入力された各フレーム毎の画像から顔領域を切り出したうえで，モザイク処理を施し，ニューラルネットワークに認識させ，出力値からモーフィングに用いる基本表情の選択・合成の割合の算出を行った．そして，この基本表情とその合成割合をもとに，モーフィングによって表情を合成し，フレーム毎の画像として出力させた．

また，各フレームにおいてどの基本表情がどの程度の割合で選択されているのかを確認するため，各フレームごとの類似度の変化を図 5.13(a) に示す．この結果では，認識誤差

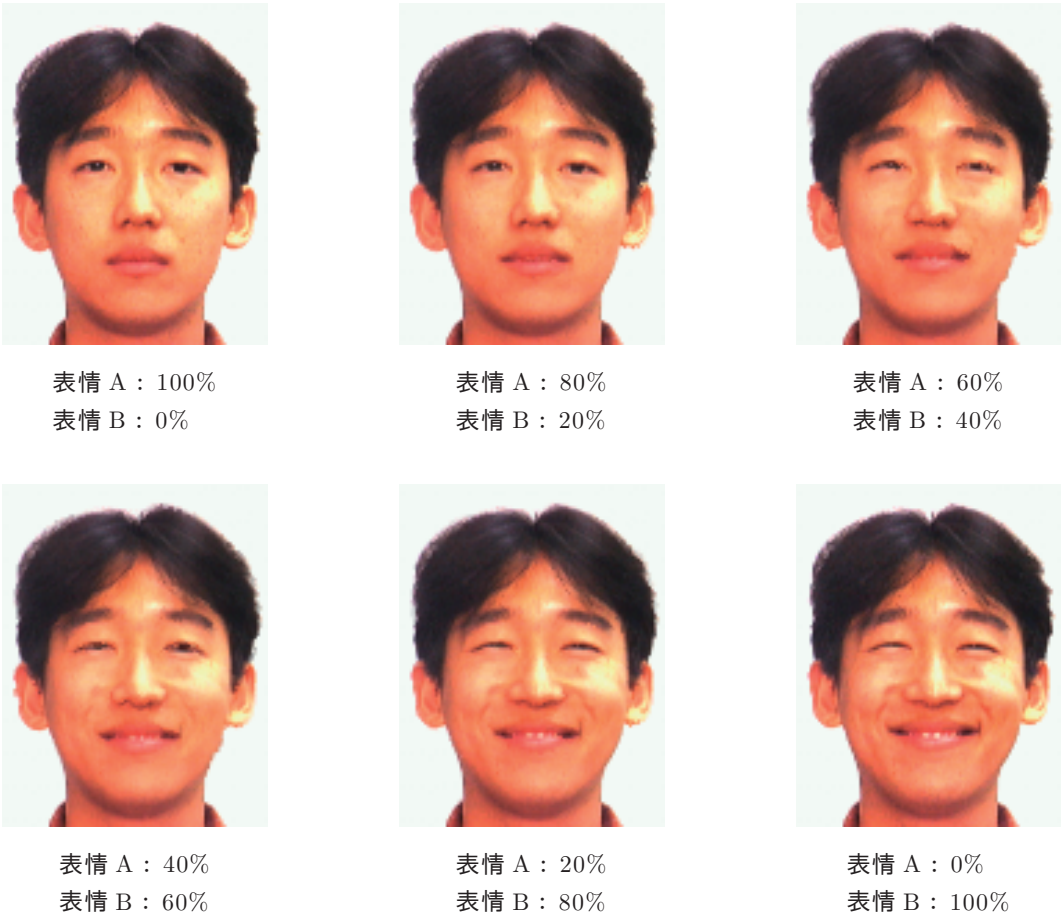


図 5.12　モーフィング結果の例

により，短いフレーム間で類似度が頻繁に変化したため，不自然さが目立ってしまった．そこで，類似度の平滑化を行った．これは，第 n フレームにおいて，実際に出力する平滑化された類似度を $S'(n)$，平滑化前の類似度を $S(n)$ としたとき，次式で示される現フレームと過去 9 フレーム，合計 10 フレームの平均値を現フレームの類似度とするものである．フレーム遅れの発生を考慮し，現フレーム以前の類似度のみを用いた．

$$S'(n) = \frac{1}{10}\sum_{i=0}^{9} S(n-i) \tag{5.2}$$

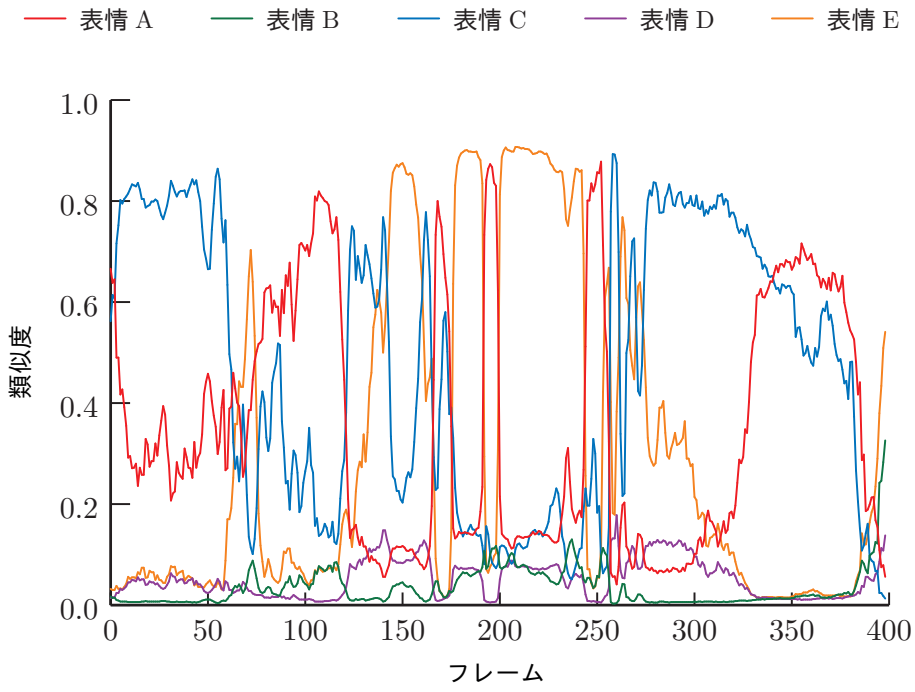
ここで，10 フレームという値は，平滑化効果の現れる最小フレーム数を実験より求めた．このときの，類似度の変化のグラフを図 5.13(b) に示す．入力表情の変化に伴い，選択される基本表情とその割合が滑らかに変化している事が分かる．

合成結果の評価は，入力表情の動画と合成表情画像から動画を生成したものを比較す

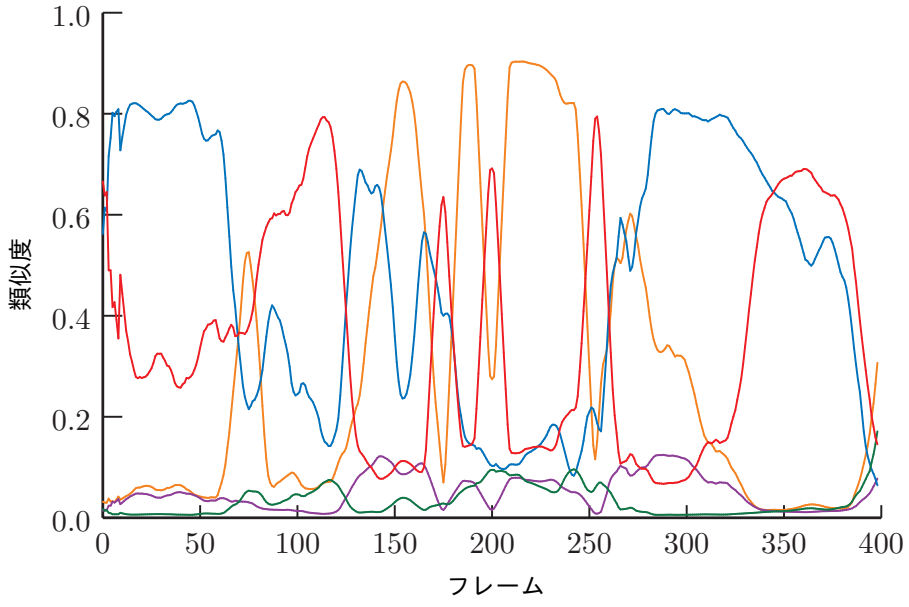
ることによって行った．比較のため，入力表情画像，合成表情画像の例を図 5.14 に示す．このように，フレーム単位の画像として見ると，多少不自然さが目立つ場合もあるが，動画として再生した場合には，ほとんど目立たなくなる．また，現段階では，主に表情の変化を検討しており，会話を考慮していないため，口の動きは不自然になっている．

次に，伝送に必要なビットレートについて検討する．本手法で，フレーム毎に伝送される情報は，選択された基本表情の識別コードとそれに対応する類似度のみである．これらを表現するのに必要なビット数は，以下のようになる．基本表情選択フラグは，基本表情五つそれぞれに対して 1 ビットを割り当て，選択された場合に 1 として，5 ビットで表現する．類似度は，1 基本表情につき，0.0〜1.0 を 64 段階で表現すれば十分な合成画像を得ることができるため，6 ビット必要である．従って，6 ビット× 2 基本表情＝ 12 ビットとなる．従って，1 フレームあたり 17 ビットで表現できる．これにより，1 秒当たり 30 フレームを送信するとすると，510bit/s の伝送となる．

本手法と類似の研究に関しては，分析と合成方式を提案した研究が多く，適切な比較対象が少ないが，ビットレートに言及している金子らの手法 [36] は，頭部の 3D ワイヤーフレームモデルの頂点座標を用いて符号化しており，毎秒 15 フレームで，2.88kbit/s となっている．これを，提案手法で想定している毎秒 30 フレームに換算すると，5.76kbit/s である．また，原島によって提唱されている符号化の世代区分と，それが目標としているビットレートによると，符号化に用いる知識が，本手法と同じ画像の認識と生成に必要な知識であるような場合では，10^2〜10^3 となっており，本手法の 510bit/s は，正面顔に限定されるが，この目標に概ね合致しているといえる．



(a) ニューラルネット出力類似度



(b) 平滑化された類似度

図 5.13　類似度の変化

(a) 入力表情画像

(b) 合成表情画像

図 5.14　入力表情と合成表情の比較

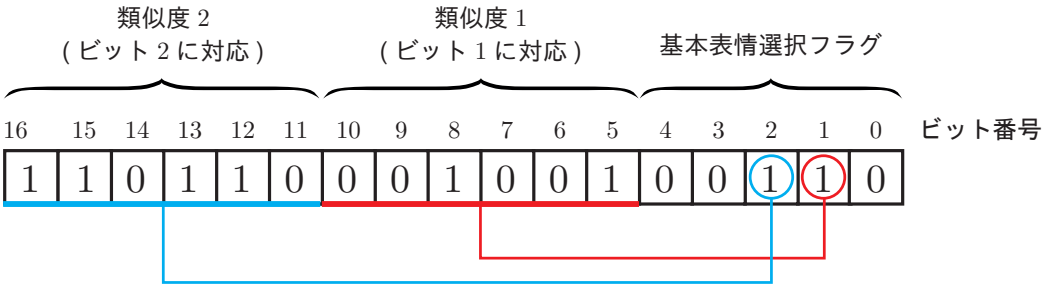


図 5.15　フレームのデータ構造

第 6 章　結論

本論文では，文書画像と顔画像を対象に，形態を指向した非線形画像処理について検討した．

文書画像に関しては，文書を保存する際の省スペース化，情報の再利用及び検索の効率化の観点から進められている紙媒体の文書の電子化において，文字認識を行う際に問題となる，地紋をもつ見出し画像から自動的に地紋を除去する手法および印刷文書に書き込まれた注釈など有益な情報を持つと考えられる手書き文字を抽出する方法を提案した．

地紋の除去では，まず，主要新聞社数社の新聞に掲載されている見出し 1000 個程度を調査し，「網線網点地紋」，「グラデーション地紋」，「片側地紋」という特徴的な 3 種類に分類できることを示し，それぞれの地紋の性質を調べ，有効な除去手法を検討した．そしてこれらを，適切に組み合わせてることで，地紋の種類の判別は，「網線網点地紋」「グラデーション地紋」であるか「片側地紋」であるかの二者択一にすることができた．しかも，もし判別を誤った場合でも，高い確率で除去可能となった．

実験の結果，310 個の見出しのうち，251 個 (81%) から，完全に地紋を除去することが可能となり，人間の目視による判断で “良好” と評価できるものは，295 個 (95.4%) であった．また，市販の OCR ソフトを用いた文字認識の結果，目視によって “良好” と評価された除去結果のサンプルは，本研究の環境において同ソフトに対する地紋無しの見出し文字認識率と同程度となり，逆に “不良” と評価された除去結果のサンプルでも，高い認識率が得られた．今後，各除去ユニットの性能を向上させ評価 (C) に相当するものをなくすことが課題となった．

手書き文字の抽出では，モフォロジカルフィルタの基本処理であるダイレーションとエロージョンを Maxout 関数で置き換え，それぞれ凸フィルタと凹フィルタに拡張し，これらを用いて，画像中に存在する輝度の窪みを除去することができるクロージングフィルタを凸凹フィルタに拡張して構成した．本研究では，凸凹フィルタ単体に加え，フィルタの出力を入力と直接比較し大きい方を出力する「最大値学習」というネットワーク構成を提案した．

また，Maxout フィルタネットワークのパラメータである荷重パラメータおよびバイア

スを学習するにあたり，ミニバッチ学習を採用したが，1 回のネットワークパラメータの更新に使用するミニバッチに含まれる学習サンプルを，同一記入者のサンプルから選んで構成する方法を提案した．凸凹フィルタ単体を確率的勾配降下法で学習したネットワーク，凸凹フィルタ単体をミニバッチ学習したネットワーク，および，提案手法である最大値学習のフィルタをミニバッチ学習したネットワークについて，手書き文字抽出性能をテストした．

抽出結果の画像と理想的な抽出画像より SNR を計算して各ネットワークを比較し，提案手法の有効性を確認することができた．また，画像の目視確認においても，十分判読できる程度に抽出されており，特に，従来の画像の局所特徴量を用いた領域分割型の手法では困難であった，印刷文書に手書き文字が重なって記入されている場合でも抽出が可能であることも確認できた．

SNR が低かったサンプルの画像を確認してみると，印刷文字が完全に除去されない，または，手書き文字が擦れてしまうなどの想定していた不具合とは別に，ゴシック体のように太い文字の一部が除去されずに残ってしまうという不具合が発生した．ネットワークパラメータの学習に論文を使用したために，太い文字を含むサンプルが少なかったこともあり，文書の大半を占める明朝体の部分に較べると完全に除去されず薄く残る箇所が見受けられた．今後，より幅広い文書への対応を検討する必要がある．

提案手法では，Maxout 関数でモフォロジカルフィルタのダイレーションとエロージョンを置き換え，クロージングフィルタを凸凹フィルタに拡張したフィルタネットワークを用いたが，ニューラルネットワークをさらに深層化するように [37]，本研究で用いたフィルタネットワークも多層化することで性能の向上も期待できる．更なる手書き文字の抽出精度の向上を目指し，深層化を検討することが今後の課題である．

人物表情通信に関しては，人間の表情を認識することによる顔画像符号化を提案し，これを用いた顔表情伝送システムの可能性について検討した．送信部での認識に関しては，基本表情として選んだ 5 表情を学習したニューラルネットを用いた．ニューラルネット入力層ユニットへの任意表情入力に対する出力層ユニット出力値を，各基本表情に対する類似度として扱うことで，表情を数個の数値で表現することができた．また，受信部では，各基本表情に対する類似度から計算される割合で，２つの基本表情のモーフィングを行うことによって，送信側のニューラルネットに入力された任意表情に近い表情を合成することができた．

提案手法において，送信部から受信部へ伝送される情報は，1 フレームにつき類似度上

位 2 つの基本表情識別コードとその類似度値のみであり，17 ビットで表現可能となった．従って，毎秒 30 フレームの動画像として伝送する場合には，510 ビット/sec になった．従って，提案した表情の知的符号化方式は，正面顔の表情画像の伝送に限られるが，極低ビットレートで高品質な動画像伝送が可能であることを示した．しかし，実システムを構築するためには，本システムで扱う画像である，会話シーン内の顔領域 (頭部全体/顔領域のみ) の自動切り出し [38][39]，基本表情へのモーフィング用特徴点設定のための，顔造作特徴点の自動抽出 [40][41]，話者の会話に連動した口唇の動きの付加，話者頭部の動きの付加などに関する検討が必要である．これら必要な技術を実装し，実システムを構成して，ネットワーク上での通信実験を行うことが今後の課題である．

以上述べた三つの処理は，いずれも，画像を構成する画素をデータ列（信号）のように扱い演算によって出力を得るようなものではなく，入力された画像に写っているそれぞれの被写体に関する形態に関する知識をコンピュータが理解して，処理する形式のものである．地紋除去では，人間によって分類された地紋の特徴を基に人間が知識を見いだし，必要な手順をアルゴリズムとして記述することで実現された．手書き文字の抽出においては，除去すべき活字と残すべき手書き文字の特徴に関する知識を，また，知的符号化においては，人間の顔表情の特徴に関する知識をニューラルネットワークを用いることによって，言わば，コンピュータがアルゴリズム自体を学習することで実現された．

これらの処理は，アルゴリズム構築過程の違いはあるが，人間が視覚によって得る情報を脳で処理して，状況に応じて適切な何らかの行動を起こす過程を代替となるものであるので，人間社会を便利で快適にするシステムに当てはめてみると，もともと人間が行ってきた作業をコンピュータ（機械）によって置き換え，人間の負担を減らすことに寄与する技術であるといえる．

このような目的のために，序論において述べたとおり，これらの技術は工業，医療，交通，その他の分野において既に応用が進んでおり，今後も応用範囲が広がっていくことが予想されるとともに，いずれの分野においても，特に，人間の生命に関わるシステムへの応用に関しては，更なる処理性能の向上が望まれるようになる．したがって，本研究の主題である，形態を指向する画像処理の研究は，応用範囲および性能向上の両面において，今後一層重要となると考えられる．

謝辞

本研究を遂行し，学位論文をまとめるにあたり，元千葉工業大学工学部 小林幸雄教授，千葉工業大学 工学部 中静真教授に多大なる御指導，御鞭撻を賜りました．小林教授には，大学院修士課程在学中より御指導賜り，修了後も学術論文の執筆および学会発表に多大なる御協力を賜りました．中静教授には，小林教授ご退職の折りに指導教官を引き継いでいただき，本論文をまとめるために多大なるご助言を賜りました．両教授に心より御礼申し上げます．

本論文をまとめるにあたり，ご多忙にも関わらず副査をお務めいただき，貴重なご意見を賜りました，千葉工業大学 工学部 久保田稔教授，菅原真司教授，枚田明彦教授，先進工学部 宮田高道教授に深く感謝いたします．

本論文を構成する学術論文を執筆するにあたり，多大なるご協力をいただいた，千葉工業大学卒業生 御園生靖史氏，久保田哲也氏，地紋付き新聞見だし，注釈付き印刷文書，会話中の動画等実験データの収集等にご協力いただいた，千葉工業大学 小林研究室，中静研究室の卒業生諸氏に深く感謝いたします．

参考文献

[1] ディジタル画像処理編集委員会（編），ディジタル画像処理，公益財団法人画像情報教育振興協会，2004.

[2] R. Szeliski, コンピュータビジョン アルゴリズムと応用，共立出版，2013.

[3] 黄瀬浩一，“文書画像理解の目指すもの，”電子情報通信学会技術研究報告. PRMU, パターン認識・メディア理解，vol.97，no.559，pp.55–62，Feb. 1998.

[4] 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.69，no.8，pp.p1187–1196，Aug. 1986.

[5] 中村　納，“欧文テキスト画像における文字領域の抽出アルゴリズム，”電子通信学会論文誌 D，vol.66，no.4，pp.p437–444，April 1983.

[6] 角谷　浩，木村正行，下平　博，奥村　学，“矩形レイアウトモデルに基づく文書画像の領域識別，”信学技報，pp.PRU93–82，1993.

[7] 林　俊成，高井峰生，成田誠之助，“画像処理による飾り文字の復元，”信学技報，pp.PRU94–12，1994.

[8] 酒匂　裕，松島　整，江尻正員，“自己組織多重解像度フィルタによるテクスチャ識別，”電子情報通信学会論文誌 D-2 情報・システム，vol.73，no.4，pp.p562–573，April 1990.

[9] 岡本正行，林　博仁，“膨張／収縮を用いた地模様のある見出しからの文字抽出，”信学技報，vol.90，pp.PRU90–151，1990.

[10] S. Liang and M. Ahmadi, “A Morphological Approach to Text String Extraction from Regular Periodic Overlapping Text/Background Images,” CVGIP:Graphical Models and Image Processing, vol.56, no.5, pp.402–413, 1994.

[11] 萩田紀博，“背景に雑音を含む 2 値画像認識法，”信学技報，pp.PRU93–133，1994.

[12] M. SAWAKI and N. HAGITA, “Text-line Extraction and Character Recognition of Japanese Newspaper Headlines with Graphical Designs,” Proceedings of 13th

International Conference on Pattern Recognition (ICPR), pp.C73–78, 1996.

[13] 糸井清晃，久保田哲也，小林幸雄，“見出し文字列の地紋除去，” 電子情報通信学会論文誌，J82 − D − II，no.4，pp.763–770，April 1999.

[14] K. Zagoris, I. Pratikakis, A. Antonacopoulos, B. Gatos, and N. Papamarkos, “Distinction Between Handwritten and Machine-printed Text Based on the Bag of Visual Words Model,” Pattern Recogn., vol.47, no.3, pp.1051–1062, March 2014.

[15] T. Nakai, K. Kise, and M. Iwamura, “A Method of Annotation Extraction from Paper Documents Using Alignment Based on Local Arrangements of Feature Points,” Proc. 9th International Conference on Document Analysis and Recognition (ICDAR2007), vol.1, pp.23–27, Sept. 2007.

[16] 小山純平，加藤雅弘，廣瀬　明，“人間の視覚機構に着想を得た手書き文字と活字文字の判別，” 日本神経回路学会誌 = The Brain & neural networks，vol.15，no.3，pp.165–173，Sept. 2008.

[17] U.M. Butt, S. Ahmed, F. Shafait, C. Nansen, A.S. Mian, and M.I. Malik, “Automatic Signature Segmentation Using Hyper-Spectral Imaging,” ICFHR, pp.19–24, IEEE Computer Society, 2016.

[18] P. Maragos and R.W. Scafer, “Morphological Filters-part I: Their Set-theoretical Analysis and Relations to Linear Shift-invariant Filters,” IEEE Trans. Acoustic, Speech and Signal Processing, vol.ASSP-35, pp.1153–1169, 1987.

[19] P. Maragos, “Chapter 3.3: Morphological Filtering For Image Enhancement And Feature Detection,” The Image and Video Processing Handbook, ed. by A.C.Bovik, pp.135–156, Elsevier Academic Press, 2005.

[20] M. Nakashizuka, K. Kei-ichiro, T. Ishikawa, and K. Itoi, “Convex Filter Networks Based on Morphological Filters and their Application to Image Noise and Mask Removal,” IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, vol.100, no.11, pp.2238–2247, 2017.

[21] I. Goodfellow, D. Warde-Farley, M. Mirza, A. Courville, and Y. Bengio, “Maxout Networks,” Proceedings of the 30th International Conference on Machine Learning, eds. by S. Dasgupta and D. McAllester, vol.28, pp.1319–1327, Proceedings of Machine Learning Research, PMLR, Atlanta, Georgia, USA, 17–19 Jun 2013.

[22] 糸井清晃，中静　真，“Maxout フィルタネットワークによる印刷文書上の手書き文字の抽出，”画像電子学会誌，vol.48，no.1，pp.153–160，1 月 2019 年.

[23] 原島　博，“次世代画像通信　　知的画像符号化と知的通信，”テレビジョン学会誌，vol.42，no.6，pp.519–525，1988.

[24] 糸井清晃，御園生靖史，小林幸雄，“ニューラルネットとモーフィングを用いた顔表情の知的符号化，”電気学会論文誌 C（電子・情報・システム部門誌），pp.1165–1171，平成 12 年.

[25] 岡谷貴之，深層学習，機械学習プロフェッショナルシリーズ，(株) 講談社，2014.

[26] 大津展之，“判別および最小 2 乗基準に基づく自動しきい値選定法，”電子情報通信学会論文誌 D，vol.J63-D，pp.349–356，1980.

[27] K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition,” CVPR, pp.770–778, IEEE Computer Society, 2016.

[28] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, “Beyond a Gaussian Denoiser: Residual Learning of Deep CNN for Image Denoising,” IEEE Trans. Image Processing, vol.26, no.7, pp.3142–3155, 2017.

[29] J. Duchi, E. Hazan, and Y. Singer, “Adaptive Subgradient Methods for Online Learning and Stochastic Optimization,” J. Mach. Learn. Res., vol.12, pp.2121–2159, July 2011.

[30] M. Turk and A. Pentland, “Eigenfaces for Recognition,” J. Cognitive Neuroscience, vol.3, no.1, pp.71–86, Jan. 1991.

[31] 岡崎昌石，原島　博，武部　幹，“主成分分析による顔画像の基底生成と記述，”情報処理学会研究報告グラフィクスと CAD（CG），vol.1990，no.66，pp.43–50，Aug. 1990.

[32] 木村　聡，谷内田正彦，“表情認識とその程度推定，”情報処理学会研究報告コンピュータビジョンとイメージメディア（CVIM），vol.1997，no.10，pp.99–106，Jan. 1997.

[33] 赤松　茂，“コンピュータによる顔の認識 - サーベイ -，”電子情報通信学会論文誌. A, 基礎・境界，vol.80，no.8，pp.1215–1230，Aug. 1997.

[34] 小杉　信，“モザイクとニューラルネットを用いた顔画像の認識，”電子情報通信学会論文誌. D-II, 情報・システム, II-情報処理，vol.76，no.6，pp.1132–1139，June 1993.

[35] 黒田敏行，糸井清晃，小林幸雄，“モーフィングを用いた中割り画像の生成，”電子情報通信学会ソサイエティ大会講演論文集，vol.1996，p.397，Sept. 1996.

[36] 金子正秀，羽鳥好律，小池　淳，“形状変化の検出と 3 次元形状モデルに基づく顔動画像の符号化，”電子情報通信学会論文誌，vol.J71-B，no.12，pp.1554–1563，12 月 1988 年.

[37] W. Sun, F. Su, and L. Wang, “Improving Deep Neural Networks with Multi-layer Maxout Networks and a Novel Initialization Method,” Neurocomputing, vol.278, pp.34–40, 2018.

[38] 小杉　信，“個人識別のための多重ピラミッドを用いたシーン中の顔の探索・位置決め，”電子情報通信学会論文誌. D-II, 情報・システム, II-情報処理，vol.77，no.4，pp.672–681，April 1994.

[39] Y. DAI, “Extraction of Facial Images from Complex Background Using Color Information and SGLD Matrices,” Proc. Intl. Workshop on Automatic Face-and Gesture Recognition, pp.238–242, 1995.

[40] 福井和広，山口　修，“形状抽出とパターン照合の組合せによる顔特徴点抽出，”電子情報通信学会論文誌. D-2, 情報・システム 2-情報処理，vol.80，no.8，pp.2170–2177，Aug. 1997.

[41] 横山太郎，八木康史，谷内田正彦，呉　海元，“顔の軸対称性を考慮した顔輪郭の自動抽出，”電子情報通信学会論文誌. D-2, 情報・システム 2-情報処理，vol.80，no.8，pp.2178–2185，Aug. 1997.

本論文に関する原著論文

論文

- 糸井清晃，中静真，“Maxout フィルタネットワークによる印刷文書上の手書き文字の抽出，”画像電子学会誌，vol.48，no.1，pp.153–160，1 月 2019．… [第 4 章]
- 糸井清晃，御園生靖史，小林幸雄，“ニューラルネットとモーフィングを用いた顔表情の知的符号化，”電気学会論文誌 C（電子・情報・システム部門誌），pp.1165–1171，平成 12 年．… [第 5 章]
- 糸井清晃，久保田哲也，小林幸雄，“見出し文字列の地紋除去，”電子情報通信学会論文誌，J82–D–II，no.4，pp.763–770，April 1999．… [第 3 章]

国際会議

- Kiyoaki Itoi and Makoto Nakashizuka, “An Extraction Method of Handwritten Characters on Printed Documents by Maxout Filter Networks,” 2018 International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS2018), pp.504–P.509, Nov.27–30, 2018
- Kiyoaki Itoi, Masanao Sasaki and Hiroaki Nakabayashi, “A Study on Antenna Arrangement Optimization of Massive MIMO,” 2017 International Symposium on Antennas and Propagation (ISAP), Phuket, pp.1-2, 2017.
- Kiyoaki Itoi, Yasushi Misono, and Yukio Kobayashi, “Intelligent Coding of Facial Expression Using Neural Network and Morphing,” 2001 International Symposium on Intelligent Multimedia, Video and Speech Processing May 2-4 2001 Hong Kong, pp.352–355, May 2001.

国内会議

- 中静 真, 小林慧一郎, 石川徹, 糸井清晃, “Maxout 関数によるモフォロジカルフィルタの拡張,” 電子情報通信学会 第 31 回 信号処理シンポジウム, 2016 年 11 月.
- 糸井清晃, 御園生靖史, 小林幸雄, “ニューラルネットとモーフィングを用いた顔画像の知的符号化,” 平成 11 年電気学会電子・情報・システム部門大会, 1999 年 8 月.
- 遠藤康男，糸井清晃，小林幸雄, “顔画像認識のための顔領域の自動切り出し,” 1996 年電子情報通信学会ソサイエティ大会, 1996 年 9 月.
- 糸井清晃，新井浩志，小林幸雄, “文書画像の領域の分割に関する一検討,” 情報処理学会第 51 回全国大会, 1995 年 9 月.
- 糸井清晃, 川越敏史, 新井浩志, 小林幸雄, “網点・網線除去に関する一検討,” 1994 年電子情報通信学会秋期大会講演論文集, 1994 年 9 月.