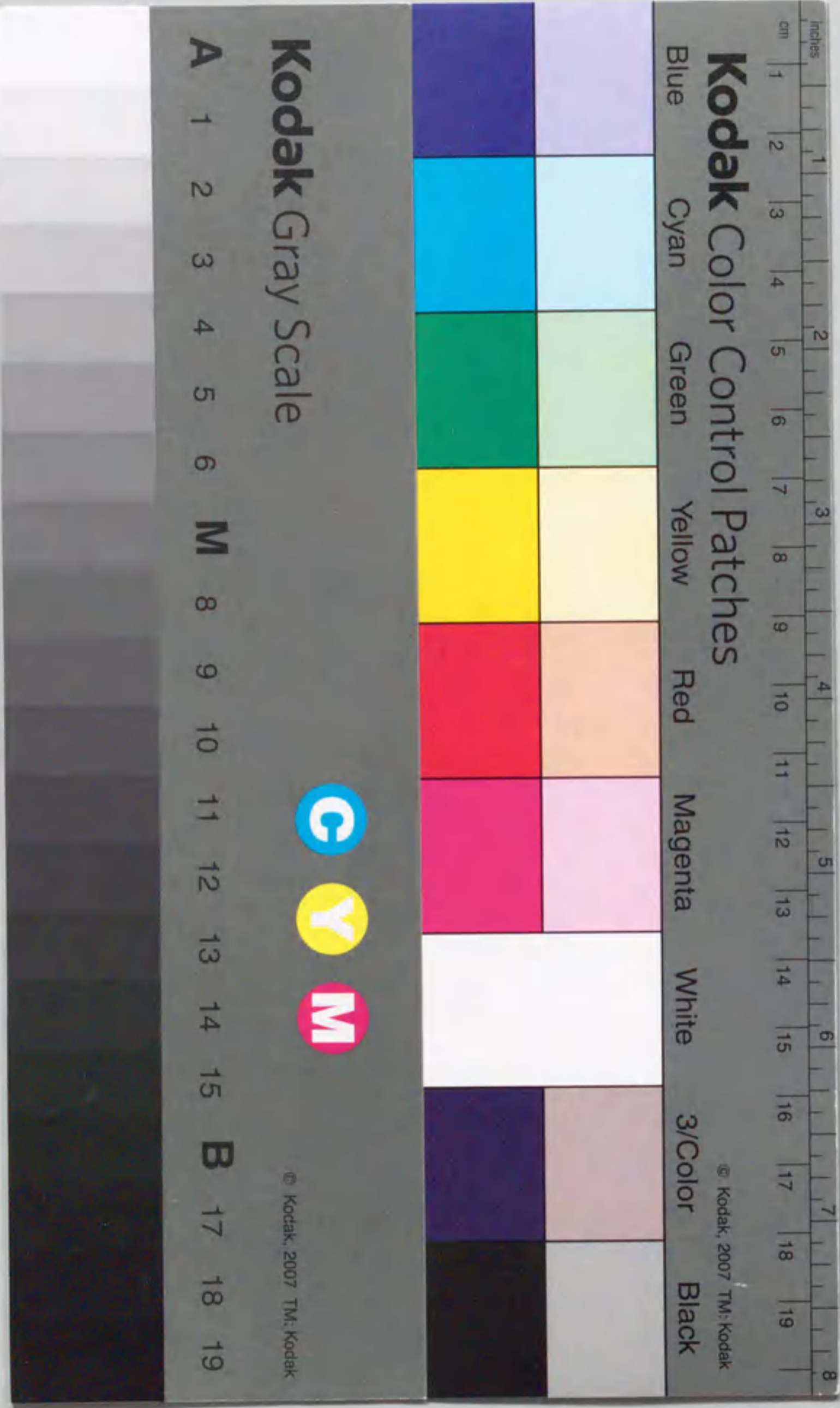


報告番号 乙第...4324 号

ニューラルネットワークの学習機能と パターン認識への応用に関する研究

木 泰 治



正 誤 表

誤	正	
1 頁 2 0 行目 2 1 行目	Rosenb latt	Rosenblatt
1 9 頁 1 8 行目	$f'(x_i)$	$f'(x_i)$
2 0 頁 6 行目	おて	おいて
2 0 頁 1 0 行目	g_{p_j} は	g_{p_j} は
6 2 頁 2 0 行目 2 1 行目	ComN et	ComNet
6 4 頁 5 行目 6 行目	Reason-in g	Reasoning
8 0 頁 1 0 行目	$\alpha_1=2$ 、 $\alpha_2=4$	$\alpha_1=2$ 、 $\alpha_2=4$

ニューラルネットワークの学習機能と パターン認識への応用に関する研究

目次

1. ニューラルネットワークの学習機能に関する研究 1

1.1 ニューラルネットワークの学習機能 1

1.2 ニューラルネットワークの学習機能の応用 1

1.3 ニューラルネットワークの学習機能の応用の応用 1

1.4 ニューラルネットワークの学習機能の応用の応用の応用 1

1.5 ニューラルネットワークの学習機能の応用の応用の応用の応用 1

2. パターン認識へのニューラルネットワークの応用 2

2.1 パターン認識へのニューラルネットワークの応用 2

2.2 パターン認識へのニューラルネットワークの応用の応用 2

2.3 パターン認識へのニューラルネットワークの応用の応用の応用 2

2.4 パターン認識へのニューラルネットワークの応用の応用の応用の応用 2

2.5 パターン認識へのニューラルネットワークの応用の応用の応用の応用の応用 2

✍ 木 泰 治

目次

1. 序論	1
1.1 はじめに	1
1.2 背景	3
1.2.1 ニューラルネットワークの学習	3
1.2.2 パターン処理と記号処理との統合	4
1.3 従来の研究	6
1.3.1 ニューラルネットワークのパターン認識への応用	6
1.3.2 パターンと記号との統合化処理によるパターン認識	8
1.4 本研究の目的	11
1.4.1 パターン認識に適したニューラルネットワーク	11
1.4.2 パターンと記号との統合化処理によるパターン認識	12
1.5 本論文の構成	14
2. パターン認識に適したニューラルネットワークの学習	16
2.1 はじめに	16
2.2 誤差適応型評価関数による学習の高速化	18
2.2.1 BP法とその問題点	18
2.2.2 誤差適応型評価関数とそれを用いた学習法	19
2.2.3 学習進捗の実験	21
2.3 誤差適応型評価関数による演算精度の低減	28
2.3.1 演算精度低減方法	28
2.3.2 精度低減の実験	32
2.4 まとめ	37
3. 量子化ニューロンモデルを用いた	
ニューラルネットワークとその高速学習、追加学習	38
3.1 はじめに	38

3.2 量子化ニューロンモデルを用いたニューラルネットワーク	40
3.2.1 量子化ニューロンモデル	40
3.2.2 量子化ニューロンモデルを用いた機能別階層ネットワーク	41
3.2.3 高速学習と追加学習性の実験	43
3.3 適応増殖量子化ニューロンを用いたニューラルネットワーク	47
3.3.1 適応増殖量子化ニューロン	47
3.3.2 適応増殖量子化ニューロンを用いた機能別階層ネットワーク	50
3.3.3 初期学習の高速性の実験	52
3.3.4 追加学習の実験	56
3.4 まとめ	61
4. ニューラルネットワークによる手書き文字認識	62
4.1 はじめに	62
4.2 ファジイ推論に基づく構造化ニューラルネットワーク による手書き数字認識	64
4.2.1 構造化ニューラルネットワークの構成	64
4.2.2 手書き数字認識実験	69
4.3 ニューラルネットワークを用いた柔軟な文字切り出し	74
4.3.1 手書き文字切り出しニューラルネットワークの構成	74
4.3.2 文字切り出し実験	79
4.4 まとめ	86
5. パターンと記号との統合化処理によるパターン認識	87
5.1 はじめに	87
5.2 パターンと記号の統合化処理(CoPS)による文字認識システム	89
5.2.1 文字認識システムの全体構成	89
5.2.2 前処理部	91
5.2.3 認識部	93
5.2.4 後処理部	95
5.2.5 文字認識全体の制御	98
5.3 文字認識実験	100
5.3.1 パターンと記号の統合化処理の効果	100
5.3.2 低品質文書の認識	102
5.4 まとめ	103

6. 結論	104
6.1 本研究のまとめ	104
6.2 今後の研究展望	107

謝辞	108
----------	-----

参考文献	109
------------	-----

第1章

序論

1.1 はじめに

文字、画像、音声認識などのパターン認識は、人工知能の一分野として古くから研究され、種々の原理や手法が提案されているが、限られた領域でしか実用化されていない。しかし、コンピュータ利用のマンマシンインターフェースとして、パターン認識の必要性は高く、実用的な性能を求める研究が進められている。パターン認識がより高い性能を実現し、マンマシンインターフェースにおいて広く使われるためには、従来とは異なる新しいアプローチが必要である。

従来の手法は、パターンに関する知識をルール化して認識を行うために、入力情報の変動、例えばノイズや変形に対応することが容易ではない。文字認識であれば、活字文字のフォントパターンの字形変動や手書きの文字に対応できず、認識性能を向上させることが容易ではない。そこで、脳機能に学び、これを解決していこうとする研究が進められている。その一つであるニューラルネットワークは学習性があり、柔軟な認識判断ができる可能性があるものとして、モデルや学習法のような研究が行われている。

ニューラルネットワークの研究は、初期のコンピュータ研究における、脳の仕組みを学びそれを実現しようという人工知能の研究の一環として、1940～1960年にかけて開始された[1]。McCullochとPittsは、ニューロンの単純なモデルである形式ニューロンモデルをつくり、その論理演算系としての万能性を示した。次に、Wienerのサイバネティクス、Hebbの学習モデルなど、現在に至るまで大きな影響を及ぼす基本原理が提案され、Rosenblattの単純パーセプトロンにより、学習する人工システムの具体的な設計指針が与えられた。このように、ニューラルネットワークの研究はコンピュータ研究とともに始められたが、1969年にMinskyとPapertが単純パーセプトロンの能力の限界を明かにしたことで、学習機械に関する研究は停滞した。一方、ノイマン型コンピュータの研究は、半導体技術の著しい進歩による、電子デバイスの高集積化、高速化及び低価格化に支えられて格段に進

歩した。学習機械も研究が進み、1980年代中ごろ、LSI化可能な相互結合型ニューラルネットワークモデルであるホップフィールド型ニューラルネットワークを、Hopfieldが提案し、連想メモリや組合せ問題の解決に応用することによりその有効性を示した。また、Rumelhartらが多層パーセプトロンを学習させるアルゴリズムであるバックプロパゲーション学習法を提案し、単純パーセプトロンの能力の限界が超えられることを示した。そして、テキスト音声合成や潜水艦のソナー音の認識などへ適用し、有効性が確認された。これ以後、ニューラルネットワークの研究が再び勢力的に進められている。中でも、ニューラルネットワークの特徴を生かした、パターン認識への応用に関する研究は盛んである。

パターン情報の中でも文字は、そのパターンレベル（例えば形状）と記号レベル（例えば文法）に関して、規則が存在する。それ故、処理の妥当性の評価を明確にできる。そこで本論文では、パターン認識の題材として文字を取り上げ、人間が行っているような学習性のあるパターン認識を模倣し、従来の機械では容易ではなかった柔軟なパターン認識を実現することを目指す。

1.2 背景

人が文字を学習する過程をパターン認識の立場から考えてみると、文字形状の構造のルールを先に学習するのではなく、パターン形状を覚えて、その中から形状のルールを把握している。その学習過程では、学習の不完全な文字や、間違える文字を繰り返し練習している。

次に、人が文章を読む際には、単語や文法の知識とともに、読む対象である文章の内容に関する知識も使いながら読むので、汚れていて判読できない文字、いままで見たこともない文字などがあっても、人はその前後の関係から文字を推定して読むことができる。そして、知らなかった文字は学習して覚える。

そこで本論文では、人間のこのような機構・動作を模倣すべく、ニューラルネットワークの持つ学習機能のパターン認識への応用とニューラルネットワークによるパターン処理と、知識による記号処理を融合化したパターン認識を取り上げた。以下に、それぞれの背景を述べる。

1.2.1 ニューラルネットワークの学習

ニューラルネットワークは、文字、音声認識などのパターン認識の分野に適用され、その性能を確認されている。中でも、多層パーセプトロン型の階層型ニューラルネットワークにバックプロパゲーション学習法（以下BP法）[2]を用いたものや、学習ベクトル量子化（以下LVQ法）[41]を用いたものは、手軽にシミュレーションができ、多くの適用実験が行われている。しかし、実問題への適用に際して、以下の問題がある。

- ・新しいデータに対する学習を活かした適応性の確保（学習可能性）、及び未学習データに対する適切な判断（汎化能力）
- ・認識すべきカテゴリーの増加に伴う、ネットワーク規模の拡大
- ・ハードウェアによる実現

このような問題に対し、従来はパターン認識という観点からの議論が十分ではなかった。

例えば、BP法では、入力に対するネットワークの出力信号と教師信号との二乗誤差総和を評価関数とし、それを最小化するように学習を行う。しかし、パターン認識を行うためには入力パターンに対して、正解を出すべきニューロンの出力値が他のニューロンのそれと差がつけば十分である。それにもかかわらず、従来のBP法は二乗誤差総和が減少していなければ学習を進めるために、パターン認識にとって必要以上の学習を行ってしまう

ことになっている。さらに、BP法は学習時に、結合の重みの変更をすべての結合の重みを対象にして行うために、すでに学習済みのパターンに対する判断に与える影響が大きく出ることとなる。これらは、認識すべきカテゴリーが増加するに従い大きな問題となってくる。

また、BP法は積和としきい値処理の繰り返しであるので、並列処理に適したアルゴリズムであり、その高い学習能力[3]を活かすためには、専用ハードウェアによる実現が望まれる。アナログ回路による実現においては、ノイズや非線形特性のために高い演算精度の実現が困難である。また、ディジタル回路による実現においては、乗算器や、結合の重みを蓄積するメモリなど、ハードウェアの規模が大きくなることが問題となる。学習時の演算に必要な精度を低減することは、アナログ回路による学習を容易にする上でも、ディジタル回路のハードウェア規模を低減する上でも重要である。

ニューラルネットワークをパターン認識の代表例である文字認識に適用する場合、従来の文字認識よりも優れた成果を挙げるためには、いかにして文字パターンの変形に対応するかが重要である。特に、手書き文字などでは筆記者の字癖や筆記具等による字形変動が大きく、同一文字に対しても多種多様なパターンが存在する。その変動を吸収するには、ニューラルネットワークの規模を大きくすればよいが、一般に規模が大きくなるにつれて、学習が困難になり、汎化能力が低下するなどの問題が生じる。さらに、一文字一文字の認識を行うのに際し、手書きなどの字形変動が大きい文字列を文字パターン毎に分離する(切り出し)には、対象パターンに対する処理の適応性を十分に確保する必要がある。

このように、ニューラルネットワークをパターン認識へ適用するには、学習の収束がうまく行かない(時間がかかる)こと、追加学習が困難なこと、過学習になること、演算精度の低減、汎化能力の確保、パターン変動への対応等が、解決すべき大きな問題点である。

1.2.2 パターン処理と記号処理との統合

コンピュータと半導体技術の進歩により、パソコン、ワードプロセッサ、複写機、ファクシミリなどの情報処理機器が、オフィスはもとより、家庭へも普及している。中でも、ワードプロセッサ、デスクトップパブリッシングなどのための文書生成用コンピュータ支援環境の発展は著しい。それと共に、コンピュータ自身による文書情報の入力、管理技術への要求が高まってきている。これに応えるものの一つに、文書認識が注目されている[4]。日本語文書入力用の光学式文字読み取り装置(OCR)の認識精度は年々向上し

てきており、印刷文書に関しては文字品質の良い文書であれば、認識率99%以上の装置が開発されている[5]。しかし、品質の悪い印字や手書き文字の場合には、必ずしも実用的な認識率のレベルに達しているとは限らない。

文字認識の処理は、以下の処理から構成される[5]。

- ・画像入力 : 紙の上に書かれた文字をスキャナにより画像として入力する
- ・文字切り出し : 1文字毎に文字パターンを切り出し、雑音除去や大きさの正規化などのパターン整形を行う
- ・特徴抽出 : 文字パターンから文字固有の特徴を抽出する
- ・文字識別 : 抽出された特徴と予め登録されている文字の特徴とを照合して、最も類似したものを識別結果とする
- ・後処理 : 語彙・文法などの知識を用いて識別結果の妥当性を調べ、必要ならば修正を加え結果として出力する

従来の文字認識の研究では、特徴抽出と文字識別に力点を置き、認識率を向上させようという取り組みが行われてきた。しかし、認識対象がOCR帳票から一般文書に拡大するに従い、自由書式や手書き文字を処理するために、適応的な文字切り出しや後処理が重要になっている。また、日本語情報を扱う場合には、文字種が多く、類似文字も多いために、認識対象が低品質印字や手書き文字であれば、1文字毎の認識処理だけであったり、上述の各処理が個別に動作する処理となる。

人間が文書を読むときは、脳の中の各処理が相互に判断結果をやり取りしながら認識を行っていると考えられている。それ故、文字の大小や変形、かすれ、脱字があっても正しく認識することができる。そこで、このような各処理の結果を有機的に結合して判断を行う認識処理が重要となり、これをどのようにコンピュータ上で実現するかが大きな課題である。

1.3 従来の研究

1.3.1 ニューラルネットワークのパターン認識への応用

ニューラルネットワークのパターン認識への応用に関して様々な研究が行われているが、本節では、文字パターンを認識することに関するもので、高速学習、ハードウェア化のための演算精度の低減、追加学習、パターン変動対応の4つの視点から従来の研究をまとめる。

(1) 高速学習

BP法で学習させた階層型ニューラルネットワークを用いて文字認識をすることで、高い性能を得られることが報告されているが[6][7][8][9]、認識すべきカテゴリの増加に伴うネットワークの大規模化、新たな入力データに対する認識能力を即座に獲得するための学習性を活かした適応性を実現して行くためには、学習の高速化が重要である。

階層型ニューラルネットワークの学習の高速化手法としては以下に分類して示すものが提案されている。

- ・学習データの与え方を工夫したもの[10][11]
- ・学習パラメータを学習の過程で動的に制御したもの[10][12][13][14][15]
- ・非線形最適化手法を導入したもの[16][17]
- ・評価関数を改良したもの[18][19][20]

しかし、これらの手法は誤差収束の高速化を目的としたものであり、パターン認識における認識率性能を向上させるための学習の高速化には必ずしもつながるとは限らない。

(2) ハードウェア化のための演算精度の低減

ニューラルネットワークを実用化するには、ハードウェア化は検討しなければならない課題である。演算精度のとれないアナログ回路、演算精度はとれるがそれによりハードウェア規模が大きくなるデジタル回路の両者に対し、演算精度を低減しても所望の機能を獲得する学習法が必要である。

従来のBP法に必要な演算精度を、実験的もしくは解析的に検討したものには、次のよ

うなものがある。16ビットの固定小数点演算で、11フォントの英字(26カテゴリ)及び音声合成、パリティ問題、潜水艦ソナー音認識の学習が可能であることが示されている[21][22]。また、分類の学習問題には、12~14ビットの演算精度で十分であることが統計的分析で示されている[23]。

従来のBP法にとらわれずに、結合の重みの変更量を3値($\pm\eta, 0$)とし、40文字の識別が可能であることも実験で示されている[24]。

すべての変数に対する演算精度が限定されている学習において、階層型ニューラルネットワークの学習に必要な演算精度を低減する取り組みには、次のようなものがある。正負の符号のみを逆伝搬させ、中間層に教師を与える学習法を提案し、7ビットの演算精度で単一フォント英数字認識の学習が可能であることを示したもの[25]、最小精度以下の変数の確率的切り上げによる学習法で、6ビットパリティ、ソナー、螺旋の学習が7ビット精度で可能なこと[26]を示したものがある。

しかし、いずれの研究もパターン認識の実用的課題の解決についてはさらに検討を加える必要がある。

(3) 追加学習

ニューラルネットワークの学習能力は、新しい認識機能を再構築するために、極めて有効である。このことにより、使用者の好みや、使用される環境に適応して、適切な動作を行うことができる柔軟なシステムが実現可能になる。このような適応性を実現するには、学習済みのデータへの影響が少なく、しかも高速な追加学習機能が要求される。

BP法[2][40]は、非常に高い学習能力を示すものの、ネットワークの動作を決める結合の重みが分散表現されているため、入力パターンを記憶するための追加パターンの学習過程で、ネットワーク全体の結合の重みの変更が行われ、学習済みのパターンの認識性能に影響がでやすいという問題点がある。

学習ベクトル量子化[41]は、入力パターンに対する最近傍ニューロンの選択とそれに関する局所的な結合の重みの変更を行うので、追加学習時に学習済みのパターンによって獲得された結合の重みに影響はでにくい。しかし、最近傍ニューロンの選択時に、入力パターンベクトルとすべての参照ベクトルとの距離計算に長い時間を要するという問題点がある。

追加学習性を保持するモデルとして注目されているリストラクティッド・クーロン・エネルギーネットワーク(以下RCEネットワーク)[42]は、追加パターンと学習済みパターンとの距離に応じて自動的に中間層ニューロンを生成できるが、ベクトル間の距離計算が必要である。

このように、従来のニューラルネットワークモデルでは、追加学習時に学習済みのデー

タと追加データとを同時に再学習する必要性や、学習時間が長いことから、状況に応じた適応的な学習が困難であった。

(4) パターン変動の吸収

従来の文字認識の限界を打ち破るための視点は、いかにして文字パターンの変動に対応するかである。ニューラルネットワークが有している汎化性はこれに応えるものである。この汎化性を学習により如何に獲得するかが重要である。

脳機能の生理的知見を基に、合成的手法で機能を明らかにする取り組みとして、代表パターンを教えるだけでその変形パターンの認識が可能となるネオコグニトロン [50] [51] というニューラルネットワークモデルがある。このモデルはパターン変形への対応は優れているが、構成が大規模で複雑であり、工業的実現の観点からは課題が多い。

階層型ニューラルネットワークを手書き文字認識へ適用したもの [7] [9] [52] [53] があり、従来の文字認識手法に劣らない性能は得られるが、汎化性、すなわち未学習パターンの認識能力に関して、十分な検討は行われていない。

文字パターンの認識に不可欠な処理として、文字パターンの分離がある。これは一般に、文字切り出し処理と呼ばれるものであるが、文字パターンの変動に対処できる柔軟な処理が重要である。特に、手書き文字は文字パターンの変動が大きく、これに十分に対処できる手法はあまりない。手書き文字列に着目した場合、ラベリングによって黒画素連結成分を矩形化し、一定の文字大きさや、方形度の条件を用いて、矩形の統合、分割を繰り返す処理 [67] が提案されているが、処理手法はまだ確立されていない。

1.3.2 パターンと記号との統合化処理によるパターン認識

文字認識の研究の歴史は、パターン認識の研究の中で最も古いのが、実用化という観点では、極限られた領域でしか進展が見られていない [5]。この原因として、次の様なことが挙げられる。まず、文字認識処理は、文字切り出し、特徴抽出、文字識別、後処理の各過程から構成されているが、この中で後処理の部分が最も研究の歴史が浅く、その成果がまだ十分活用できる段階でない。次に、従来の研究は、各処理過程毎に分解して進められており、各過程を有機的に結合した処理として研究されていない。それ故、全体を通して最も性能の低い処理過程がネックとなる。しかし、言語知識の積極的利用や、学習による判断ルールの構築可能なニューラルネットワークの利用が進められてきており、各処理の性能を向上させる素地はある。本節では、各処理過程について、従来の研究を整理する。

(1) 文字切り出し

まず、文字列の切り出しであるが、この代表的な手法は、黒画素を縦横方向に投影し、ヒストグラムの谷を検出するものである。画像が傾いて入力されることに対処するために、投影する方向や対象とする領域の大きさを工夫する必要がある [71] [8]。

次に、文字列から個々の文字を切り出す処理を、以下に示す文字パターンに関する知識を用いて行う。

- ・文字外形は正方形である（日本語の場合）
- ・文字は直線上に並んでいる
- ・連続する文字間の空白は、文字列間の空白より狭い

これらを使い、黒画素の外接矩形を正方形になるように統合する方法 [71]、外接矩形の重心の位置をハフ変換し、文字列の方向を求めて文字を切り出す方法 [72]、文字列の垂直方向に投影を取る方法 [73] や文字行や列方向に膨張させる方法 [74] などがある。

(2) 文字識別

大別して3つの方法に分けることができる。パターンマッチング法、構造解析法、ならびにそれらの中間的手法である [5]。

パターンマッチング法は、パターンの全体的な観測により入力パターンと参照パターンとの照合を行い、両者の類似性を評価して識別するものである。パターンの観測方法や類似性の尺度により多くの手法がある。これに用いられる代表的な特徴量は、濃度特徴と輪郭線から抽出した線素方向の密度特徴とである。

構造解析法は、パターンの局所的な観測と、これに基づく判定を重ねて部分から全体を判断するものである。文字を記述する特徴の配置、つながり、ならびに組み合わせにより多くの手法がある。

両者の中間的な手法は、特徴抽出を精緻化したパターンマッチング法と、文字の変形を能率良く吸収する記号レベルのマッチングを行う構造解析法である。前者には、ベクトル相関 [75]、位相構造化法 [76]、近傍特徴類似度 [77] などがある。後者には、輪郭構造マッチング [78]、弛緩整合法 [79] などがある。

(3) 後処理

語彙や文法などの一般的な知識を用いて、識別部から出力されてくる候補文字集合の中

から、正しい文字列を選び出す後処理では、以下の問題がある。

- ・候補文字集合の中に正解の文字が含まれていない候補外文字の処理
- ・文法などの言語知識により、正解文字列を決定する言語処理の精度

文書画像のつぶれやかすれにより、候補外文字が発生するのに対しては、識別部が特定の文字をどの文字に誤るかを類似文字テーブルとして予め記憶しておき、認識時に候補文字集合の文字を調べて、類似文字テーブルから正解の可能性のある文字を取り出し候補文字集合を補完する候補文字補完 [82] が提案されている。文字間の接続関係を統計的に調べ、統計的な接続関係の確率によって作成した文字接続表を用いて、候補外文字を推定する方法 [83] も研究されている。

文字切り出し間違いに対しては、切り出し位置として可能性のあるすべての切り出し文字の組み合わせの中から正解を選び出す手法 [84] が提案されている。

単語辞書に未登録の単語が含まれる場合も、候補外文字と同様の処理が必要となり、これに対し未登録単語テンプレートを作成して、未登録後の単語範囲を推定する手法 [85] が提案されている。

後処理では、一般の自然言語処理に用いられていると同様の言語処理を行うと良い性能を得ることができるが [86]、計算時間が膨大になり実用的でない。

そこで、構文解析などは行わずに、文節レベルでの言語処理の結果と識別の距離情報を総合的に判断して最適な候補を決定する手法 [87] [88] [89] が多い。最適な候補を選択する手法としては、グラフ探索手法に基づき、前の文節から順に接続できる文節を探索していき、句読点を含む文節まで探索できたら、探索された文節を出力文字列として出力する手法 [90] [91] [92] が提案されている。

最適な候補の評価基準としては、識別結果の距離情報、単語の長さ、文字単位での接続情報、単語間の接続規則、単語出現頻度などを用いる研究 [85] が行われている。

1.4 本研究の目的

人間が行っているような学習性のあるパターン認識を模倣し、従来の機械では困難であった、柔軟なパターン認識を実現することが本研究の目的である。パターン認識の題材として文字を取り上げ、パターン認識に適したニューラルネットワークの学習とその構成に関する検討を行なった後、パターン情報に基づくボトムアップのパターン処理と、適応的知識に基づくトップダウンの記号処理とを融合する手法に関する検討を行い、学習性のある柔軟なパターン認識を実現する。前節で述べた従来の研究に対して、本研究では以下の順序で論を進める。

1.4.1 パターン認識に適したニューラルネットワーク

パターン認識を柔軟に行うために必要なニューラルネットワークの要件として、次に示す事項を明らかにする。パターン認識を目的とするニューラルネットワークの学習について、高速に行える学習法と、そのハードウェアによる実現を前提とした演算精度について明らかにする。柔軟性を増すための現場学習を可能にする追加学習と、高速学習の両者の機能を兼ね備えたニューラルネットワークモデルを提案し、その有効性を示す。さらに柔軟性の要件である教師データとして未提示のパターン（未学習パターン）に対応できるニューラルネットワークの構成を明らかにし、その有効性を示す。

(1) パターン認識に適したニューラルネットワークの学習

階層型ニューラルネットワークの代表的学習法であるBP法では、入力に対するネットワークの出力信号と教師信号との二乗誤差総和を評価関数とし、それを最小化するように学習を行う。しかし、パターン認識においては、入力パターンに対する出力ニューロンの出力値が、他の出力ニューロンの出力値と明確な差を持って最大となれば、そのネットワークは望ましい動作をするようになったこと、つまり学習ができたことになる。そこで、二乗誤差総和を最小化させるだけでなく、各出力ニューロンと各入力パターンに対する出力信号の教師信号に対する誤差を、各ニューロン毎、各入力パターン毎に評価し、誤差の大きいニューロン及びパターンに対してはより積極的に誤差を減少させ、しかもその評価関数を学習の進み具合に応じて変化させることにより、平均的に誤差が減少するような評価関数を提案し、パターン認識に適した階層型ニューラルネットワークの学習の高速化に有効であることを示す。

さらに、本研究で提案する学習法は、パターン認識などの教師信号が、離散値を取る学

習問題の特質に着目したものであるが、演算精度を低減できる特徴について論理的考察を加え、BP学習のハードウェア実現における提案手法の有用性を明らかにする。

(2) 量子化ニューロンモデルを用いたニューラルネットワーク

新しい認識ルールを再構築し、状況に適応して動作させるためには、高速な学習性と、学習済みのデータへの影響が少なく、しかも高速な追加学習性が要求される。しかしながら、従来のネットワークモデルでは、追加学習時に学習済みのデータと追加データをまとめて再学習する必要性があり、学習に長い時間を要していた。そこで、多段階のしきい値処理を行う量子化ニューロンと、量子化ニューロンを用いて特徴量に応じた階層を構成し、それらの判断を学習統合するネットワークからなるニューラルネットワークモデルを提案し、高速学習性、追加学習性に優れていることを明らかにする。

(3) ニューラルネットワークによる手書き文字認識

従来よりも優れた文字認識を実現するためには、いかにして文字パターンの変動を吸収するかが重要である。手書きの文字の場合、筆記者の字癖や筆記具などによる字形の変動が大きく、同一カテゴリでも多種多様なパターンが存在する。それ故、文字の識別境界は非常に入り組んだ複雑なものとなっている。従って、これを単一のニューラルネットワークで学習させようとしても、その複雑な境界を獲得できず認識性能が低下する。そこで、まず特徴空間を複数の識別容易な部分空間に大分類し、各部分空間毎に独立に入力パターンの認識を行い、それらの結果を統合判断することで複雑な識別境界を獲得する構造のニューラルネットワークモデルを提案し、その認識性能を明らかにする。

さらに、認識に先立つ文字の切り出しに際し、手書き文字の大きな外形変動に対応できる、文字外形に関する知識を埋め込んだニューラルネットワークモデルを提案し、その有効性を明らかにする。

1.4.2 パターンと記号との統合化処理によるパターン認識

一般に、文書認識システムは、文字の切り出しを行う前処理部と、1文字分のパターン情報を記号に変換する認識部と、言語情報による修正処理を行う後処理部から構成される。また、人が文書を読むときには、イメージ情報から文字領域を選択的に注視し、特徴データを抽出する。そして、それを文字として認識し、文字列のつながりから単語や意味、概念に結びつける処理を階層的に行っている。さらに、再度イメージ情報に戻って再認する

動作を繰り返すことで、ノイズのある文書イメージに対しても非常に安定した高度な認識処理を行っている。そこで、パターン情報を直接に処理する部分である前処理部、認識部を、パターンの変動に対して対応が柔軟であるニューラルネットワークによるパターン処理で構成する。単語や文法などの言語情報を論理的に処理する後処理部を、知識に基づく記号処理で構成し、それらが階層的に分散、協調して動作するパターン認識法を提案する。そして、その認識精度を明らかにするとともに、対象文書に対する知識の一つであるキーワード情報も得られることを示す。

1.5 本論文の構成

本論文は、以下の各章から構成される。

第2章では、ニューラルネットワークをパターン認識に適用するにあたり、学習の高速化と演算精度を低減する学習法を提案する。パターン認識を正しく行うには、入力パターンに対応する出力ニューロンの出力値を、他のニューロンの出力値に対して差をつけることができれば良いことに着目する。そして、学習に際して出力ニューロンのニューロン毎の誤差、及び入力パターン毎の誤差総和に応じて重み付けする評価関数を用いた学習法を実現する。これをマルチフォント英数字記号認識に適用し、学習時間の短縮と、演算精度の低減が達成できることを示す。

第3章では、未学習データの判断を追加的に学習することで、システムの適応性を向上させるニューラルネットワークモデルを提案する。追加的な学習は基本的に高速でなければならないし、すでに学習済みのパターン認識の性能に影響を与えるものであってはならない。そのために、予めの判断をする多段階のしきい値処理機能を有する量子化ニューロンを提案する。さらに、これを判断情報に応じて階層的に接続し、学習を行う最終層で統合化するニューラルネットワークを提案し、高速学習性と追加学習性を評価する。これをマルチフォント英数字記号認識に適用し、良好な学習能力が得られることを示す。

第4章では、入力パターン変動が大きく、同一カテゴリーでも多種多様なパターンが存在する手書き文字を柔軟に認識するニューラルネットワークモデルを提案する。このモデルは、パターン変動を吸収し、学習能力と汎化能力を確保するために、複数の小規模なニューラルネットワークを機能的に組み合わせたファジィ推論アーキテクチャに基づく構造化ニューラルネットワークであり、不確実な判断結果を的確にリジェクトすることが可能となっている。手書き郵便番号の認識実験評価を行い、その認識精度が高いことを示す。

さらに、認識に先立つ文字の切り出しに際し、手書き文字の大きな外形変動に対応できるニューラルネットワークモデルを提案する。このモデルは、文字外形に関する知識を埋め込んだネットワークとノイズなどを除去するネットワークとの相互作用により、文字を柔軟に切り出すものであり、従来のものより優れた性能が得られることを示す。

第5章では、人のように機械が文書を読めることを目指して、ニューラルネットワークによるパターン処理と知識による記号処理とが協調動作する文字認識法を提案する。前処理部と認識部をニューラルネットワークによるパターン処理で行い、後処理部を言語知識を用いた記号処理で行う。さらに、各処理部間で相互に情報を交換することにより総合的に認識判断することを可能にする。本システムを用いた実験結果より、統合化処理なしの場合の60%の文字を統合化処理により正確に認識できるようになることを示す。さらに、コピーを重ねた原稿やFAX原稿などの低品質の文書においても良好な認識性能を得るこ

とができることを示し、その有効性を実証する。また、処理時間に関しても、従来手法に比べて高速に処理を行うことが可能になることも明らかにする。

第6章では、本研究で得られた研究の成果についてまとめ、今後の課題について述べる。

第2章

パターン認識に適したニューラルネットワークの学習

2.1 はじめに

ニューラルネットワークは、文字、音声認識などのパターン認識の分野に適用されその性能を確認されている。中でも、多層パーセプトロン型の階層型ニューラルネットワークにバックプロパゲーション学習法（以下BP法）を用いたものは、手軽にシミュレーションができ、多くの適用実験が行われている。文字認識では高い認識性能が得られることが報告されている [6] [7] [8] [9]。しかし、実課題への適用を考えると、認識すべきカテゴリーの増加に伴うネットワークの大規模化に対処し、新たな入力データに対する認識判断機能を即座に構築するための学習性を活かすためには、学習の高速化が重要で、このための手法が多く提案されている [10] [11] [12] [13] [14] [15] [16] [17] [18] [19] [20]。しかし、これらの手法は誤差収束の高速化を目的としたものであり、パターン認識における認識率性能を向上させるための学習の高速化には必ずしもつながるとは限らない。

ニューラルネットワークを実用化するには、ハードウェア化は検討すべき課題の一つである。演算精度の十分でないアナログ回路と、演算精度は高いがハードウェアの規模が大きくなるデジタル回路の限界を突破するには、演算精度を低減しても機能は変わらない学習法が必要である。それに関して、従来のBP法に必要な演算精度を、実験的もしくは解析的に検討した結果が報告されている [21] [22] [23]。その他、従来のBP法にとらわれずに、演算精度を削減する取り組みも行われている [24] [25] [26] [28]。しかし、いずれの研究もパターン認識における実用的な課題について、さらに検討を加える必要がある。

階層型ニューラルネットワークの代表的学習法であるBP法では、入力に対するネット

ワークの出力信号と教師信号との二乗誤差総和を評価関数とし、それを最小化するように学習を行う。しかし、パターン認識においては入力パターンに対する答である出力ニューロンの出力値が、答でない他の出力ニューロンの出力値と明らかな差が付き、最大となれば、そのネットワークは望ましい動作をするようになったこと、つまり学習ができたことになる。それにもかかわらず、従来のBP法では、二乗誤差総和が減少していなければ学習を進めるので、パターン認識にとって必要以上の学習を行ってしまうことになっている。

本章では、二乗誤差総和を最小化させるだけでなく、出力信号と教師信号に対する誤差を、出力ニューロン毎、各入力パターン毎に評価し、誤差の大きいニューロン及びパターンに対してはより積極的に誤差を減少させる手法について述べる。しかも、その評価関数を学習の進み具合に応じて変化させることにより、平均的に誤差が減少するような評価関数を提案し、階層型ニューラルネットワークの学習の高速化に有効であることを示す。この学習法は、パターン認識などの教師信号が離散値を取る学習問題の特質に着目したものであるが、演算精度を低減できることについて論理的考察を加え、BP学習のハードウェア実現における提案手法の有用性を明らかにする。

以下、2.2節で学習の高速化を実現するパターン認識に適した誤差適応型評価関数を提案し [29] [30] [31] [32]、2.3節で誤差適応型評価関数による演算精度の削減の効果を示す [33] [34] [35] [36] [37] [38] [39]。最後に、2.4節でまとめを行う。

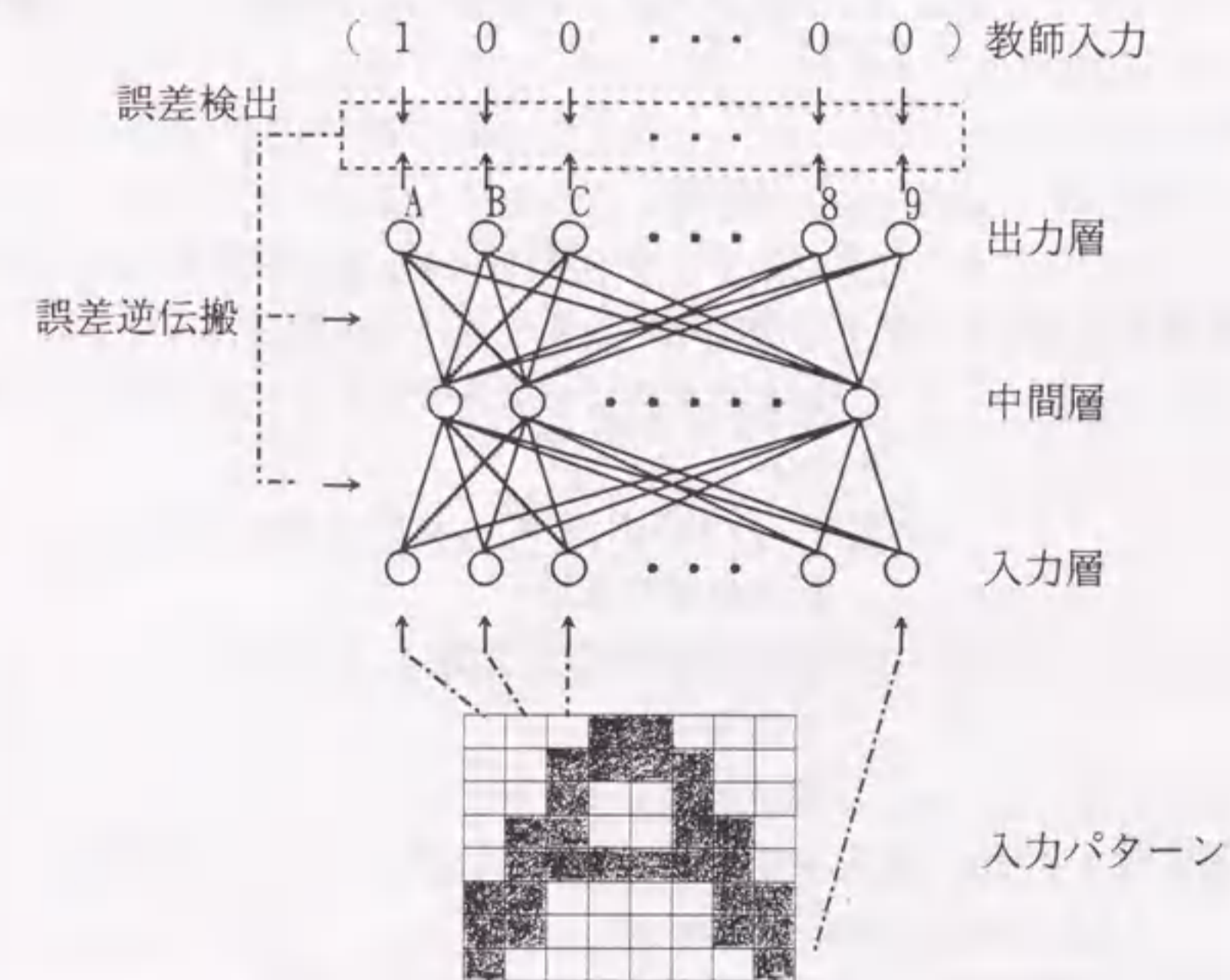


図2.1 階層型ニューラルネットワークの構成

2.2 誤差適応型評価関数による学習の高速化

本節では、パターン認識を行うニューラルネットワークの学習において、従来のBP法の問題点を明らかにし、パターン認識などの教師信号が離散値を取る学習問題の特質に着目した誤差適応型評価関数による学習法を提案し、それが学習の高速化に寄与することを示す。

2.2.1 BP法とその問題点

BP法[2][40]では、ネットワークに対して入力データを与えたときに、出力層ニューロンから出力される出力値 y_j と、教師として与えられる望ましい出力値 t_j の二乗誤差総和 E の値で現在の結合状態(結合の重みの大きさ)を評価し、この二乗誤差総和 E を減少させるように、結合の重みを変更していく。すなわち、二乗誤差総和 E は式(2.1)となる。

$$E = \frac{1}{2} \sum_p \sum_j (t_{pj} - y_{pj})^2 \quad (2.1)$$
$$y_j = f(x_j)$$
$$x_j = \sum_{i=1}^N w_{ij} y_i$$

ここで、

$f(\cdot)$: しきい値関数

y_i : 前段のニューロン層のニューロンの出力

とする。結合の重み w_{ij} は、乱数で初期化され、式(2.2)に従い変更する。

$$\Delta w_{ij} = -\varepsilon \frac{\partial E}{\partial w_{ij}} + \alpha \Delta w'_{ij} \quad (2.2)$$

ここで、

Δw_{ij} : 重み係数変更量

$\Delta w'_{ij}$: 前回の重み係数変更量

ε : 学習率

α : 加速係数

とし、しきい値関数 f は微分可能な連続関数である式(2.3)のシグモイド関数を使用する。

$$f(x_j) = 1 / \{1 + \exp(-x_j)\} \quad (2.3)$$

関数 f は微分可能な連続関数であるので誤差の逆伝搬が可能になる。

具体的には、結合の重みの計算式は次のようになる。

$$\Delta w_{ji} = \varepsilon \delta_j y_i + \alpha \Delta w'_{ji} \quad (2.4)$$

ニューロン j が出力層ニューロンの場合は、 δ_j は式(2.5)となる。

$$\delta_j = (t_j - y_j) f'(x_j) \quad (2.5)$$

中間層(隠れ層)ニューロンの場合は、 δ_j は式(2.6)となる。

$$\delta_j = \sum_k \delta_k w_{kj} f'(x_j) \quad (2.6)$$

である。 k はニューロン j が接続している出力側の階層のニューロンを示す。

このように従来のBP法では、教師信号と出力信号との二乗誤差総和を最小化することが目的であるので、誤差の大きいニューロンが残っていても、結合の重みを変更することによりそれ以外の(誤差が十分小さくなっている)ニューロンで誤差が減少し、二乗誤差総和全体として誤差が減少すれば構わずに結合の重みを変更し続ける。

パターン認識を学習するニューラルネットワークでは、出力層の一つのニューロンが最大値(=1)を出力し、出力層の他のニューロンは最小値(=0)を出力するように教師信号が与えられる。これをBP法で学習させた場合、例えば、どの入力パターンに対しても、すべての出力層のニューロンが最小値近傍の値を出力すれば、教師信号と出力信号との二乗誤差総和は比較的少なく抑えることができる。このような状態に陥ると、それ以後の結合の重みの変更が必ずしも最大値を出力すべきニューロンの誤差を減少させることには結びつかなくなる。それ故、学習が認識率向上につながらない。しかも、(2.5)、(2.6)式から解るように、結合の重み変更量は誤差の大きさに比例して決定されるが、 $f'(x_j)$ の値は $|x_j| \gg 1$ ならば零に近づくので、誤差が大きくても結果的に結合の重み変更量は小さくなり、結合の重みの変更が抑制され、この状態を抜け出すのに時間が掛かり、学習時間が長くなる。また、学習の初期から、出力層ニューロンの教師信号値を、学習の初期における入力パターンに対する出力値に対して差の大きい0あるいは1という値に設定している。このことが、認識すべきカテゴリー及び学習すべきパターン数が増加すれば、学習の負荷となり、誤差が大きいまま取り残されるニューロンが発生し易くなる。

2.2.2 誤差適応型評価関数とそれを用いた学習法

従来のBP法の問題点を考慮すれば、学習が認識率向上に寄与するためには次のような誤差評価を行うのが望ましい。

各ニューロン、各入力パターンに対する誤差を個々に評価し、誤差の大きいニューロン及びパターンに対してはより積極的に誤差を減少させることにより、誤差の減少を直接認識率向上に反映させる。しかも、誤差の評価基準を学習の進み具合に応じて変化させることにより、各ニューロン、各入力パターンのそれぞれの誤差が平均的に減少するようにさせる。このような動作を保証するために、新たな評価関数を導入する。さらに、学習が十

分進展した段階では各ニューロンの誤差は平均的になり、誤差曲面は二次超曲面に近いと考えられる。それ故この段階で、二乗誤差の最小化を積極的に進める非線形最適化手法を導入すればその効果が発揮される。

(1) 誤差適応型評価関数

学習を認識率向上に効率的に寄与させるために、教師信号と出力信号との差が大きい場合には、その誤差をより大きく評価する。さらに、出力層のすべてのニューロンにおいて教師信号と出力信号との差が小さい入力パターンに対しては結合の重みの変更をスキップする評価関数を導入する(式(2.7))。

$$E = \frac{1}{2} \sum_p G_p \sum_j g_{pj} (t_{pj} - y_{pj})^2 \quad (2.7)$$

ここで、 g_{pj} は教師信号と出力信号との大きな差を重み付けして大きく評価するための項であり、

$$\begin{aligned} |t_{pj} - y_{pj}| \geq 0.5 & \quad g_{pj} = m \\ |t_{pj} - y_{pj}| < T_g & \quad g_{pj} = 0 \\ \text{それ以外} & \quad g_{pj} = 1 \end{aligned} \quad (2.8)$$

とする。 G_p は教師信号と出力信号との差が小さな入力パターンをスキップするための項であり、

$$\begin{aligned} |t_{pj} - y_{pj}| \leq T_g & \quad G_p = 0 \\ \text{それ以外} & \quad G_p = 1 \end{aligned} \quad (2.9)$$

とする。但し、 T_g 、 T_g はしきい値として与える定数である。

この重み付けの項 g_{pj} により、誤差の大きいニューロンの結合の重みはより積極的に変更され、誤差がしきい値 T_g 以下のニューロンの結合の重みの変更は行なわれない。そして、重み付け関数 G_p により、すでに認識できているパターンについては結合の重みを変更せず、そうでないパターンのみ結合の重みを変更する。このことにより学習が認識率向上に寄与する効率的な学習が可能となる。

さらに、結合の重みの変更の判定基準であるしきい値 T_g 、 T_g を学習が進むにつれ小さくして行くことにより、学習の前半は大まかに学習し、徐々に学習の精度を上げて行くことが可能になり、誤差を平均的に減少させることができる。この方法では、局所極小値に陥りにくいため、学習の進捗が速くなると期待される。

(2) 非線形最適化手法(直線探索法、共役勾配法)との結合

誤差適応型評価関数による学習で大まかに学習した後は、各ニューロンの誤差は平均化

するので、二乗誤差総和の超曲面は二次超曲面に近いと考えられる。それ故、この二次超曲面上での誤差の最小化を積極的に進めることにより、学習を速やかに進めることができる。そこで、BP法を高速化するために、結合の重み変更方向、学習率の設定に着目した非線形最適化手法[17]を導入する。

結合の重み変更方向の最適化には、大局的にみてより望ましい方向へ進めるために、前回までに進んでいない方向へ学習を進めるという観点で、連続するn回の結合の重み変更方向が互いに線形独立であるように結合の重み変更方向を決定する共役勾配法を用いる。最適な学習率の動的設定では、結合の重み変更方向に対して異なる学習率に関する誤差を求め、最小の誤差を与える学習率で結合の重み変更量を決定する直線探索法を用いる。この手法により、学習の後期において収束がすばやく進むことが期待できる。

2.2.3 学習進捗の実験

先に示した誤差適応型評価関数を用いた学習法とその非線形最適化手法との結合による学習法とをマルチフォント英数字認識(23フォント、17フォント当り62カテゴリ)に適用し、その有効性について実験評価した。図2.2に実験に用いた文字パターンの例を示す。

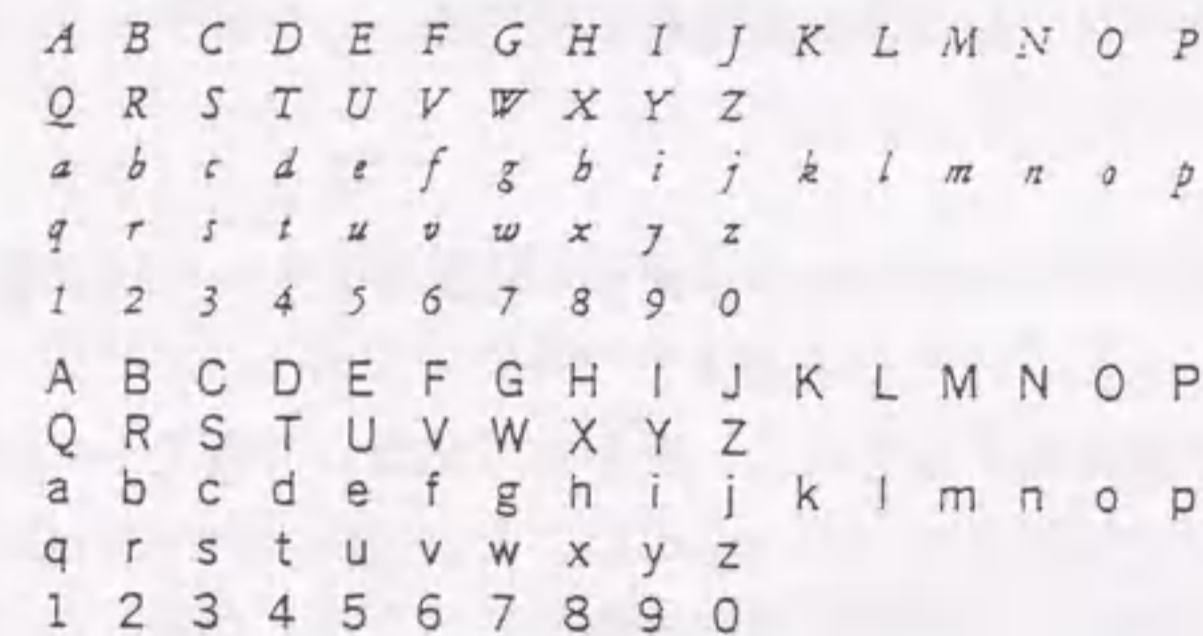


図2.2 マルチフォント英数字のパターン例

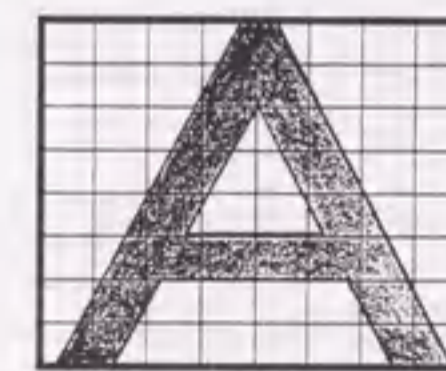


図2.3 メッシュ特徴

検討条件としての入力データと階層型ニューラルネットワークの構成は次のようにした。64×63画素からなる文字データに対して、文字部を切り出した後、8×8次元のメッシュ特徴を求め、それを入力層に与えた。メッシュ特徴は、文字の外接矩形を図2.3に示すように8×8の領域に分割し、その各領域の黒画素密度を要素とする64次元のデータである。階層型ニューラルネットワークのニューロン数は、入力層：64、中間層：42、出力層：62とした。学習のためのパラメータは、学習率 $\epsilon=0.05$ 、加速係数 $\alpha=0.95$ とした。

検討評価は以下の4項目とした。

- ・誤差の大きいニューロンの誤差を大きく評価する重み付け関数 g の効果
- ・誤差の小さい入力パターンの結合の重みの変更をスキップする重み付け関数 G の効果
- ・誤差評価の場合を制御するしきい値 T_g 、 T_c の動的制御の効果
- ・学習の後期に非線形最適化手法を導入した効果

なお、結合の重みの初期値は $[-0.3, +0.3]$ の一様乱数とし、初期値による影響を避けるために、5種類の初期値による認識率の平均値を求め比較した。認識率の変化は、学習時間に対する学習データ(137バイト)の認識率の変化と、未学習データ(107バイト)の認識率の変化である。

(1) 誤差の大きいニューロンの誤差を大きく評価する重み付け関数 g の効果

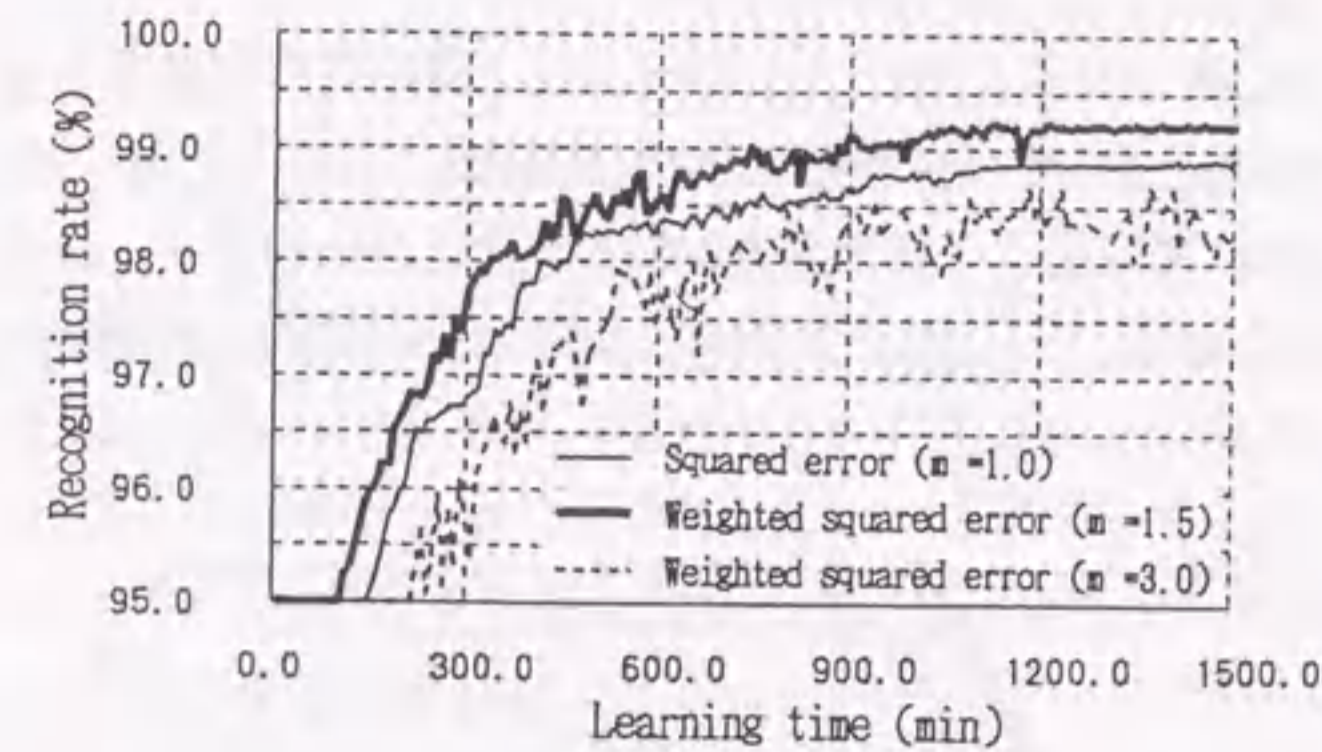
重み付け関数 g の効果を見るために、重み付け関数 G の値を1に固定して(すなわち、 $T_c=0.0$)、重み付け関数 g のパラメータ m 、 T_g を変化させたときの学習データの認識率の変化について二乗誤差評価関数を用いた従来のBP法と比較した。

(a) 重み付け評価量 m による学習時の認識率変化

$T_g=0.0$ に固定し、 m の値に対する学習時の認識率の時間変化を図2.4に示す。 $m=1.0$ の場合が従来の二乗誤差評価関数を用いたBP法に相当する。

$m=1.5$ に設定した場合、認識率99.0%に達するまでの時間は、二乗誤差評価関数の場合に対し約1/2に短縮されている。

重み付け関数を導入することにより、誤差の大きいニューロンに対してはより積極的に誤差を減少させるように、重み変更に関わる係数 δ の値が設定される。それ故、大きい誤差を出力しているニューロンの数が、従来の二乗誤差評価関数よりもより早く減少し、各ニューロンの誤差がそれぞれ同じように減少する。ただし、 m の値を大きすぎると結合の重み変更量が必要以上に大きくなるため収束が困難になる。



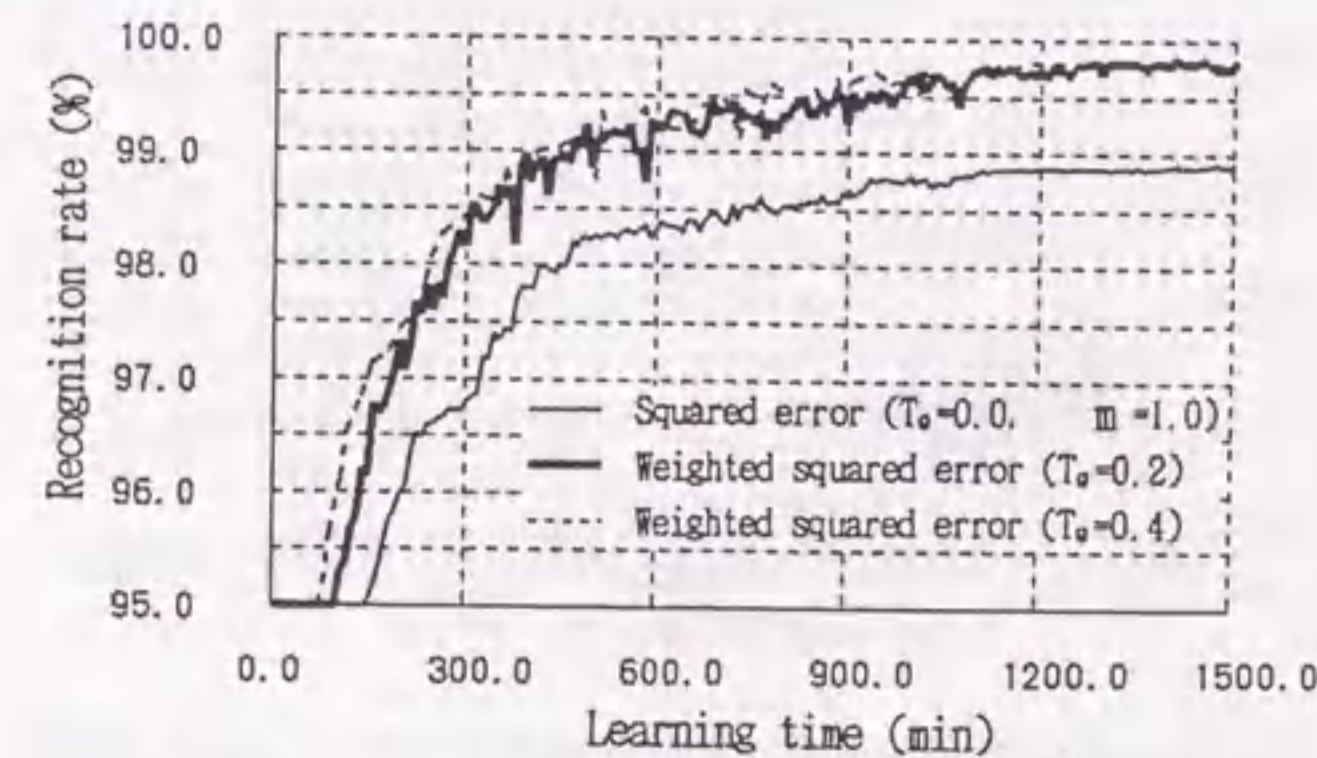
時間はSequent社の並列計算機S 8 1のCPU Timeによる。

図2.4 重み付け評価量 m による学習データの認識率変化 ($T_g=0.0$, $T_c=0.0$)

(b) 出力ニューロン毎の誤差評価のしきい値 T_g による学習時の認識率変化

$m=1.5$ とし、 T_g の値に対する学習時の認識率の時間変化を図2.5に示す。 $T_g=0.2$ の場合と、図2.4 ($m=1.5$, $T_g=0.0$)の変化と比較すると学習時間が短縮されており、認識率99.0%に達するまでの時間は、二乗誤差評価関数の場合に対し約1/4に短縮されている。それ故、誤差の値が0.4以下のニューロンについては、誤差の値を零と評価する方が学習の高速化に有効であるといえる。

認識率の向上に寄与しない重み変更量が抑制され、その分誤差の大きいニューロンの重み変更に振り分けられたため、誤差の大きいまま取り残されるニューロンの数が減少したためと考えられる。



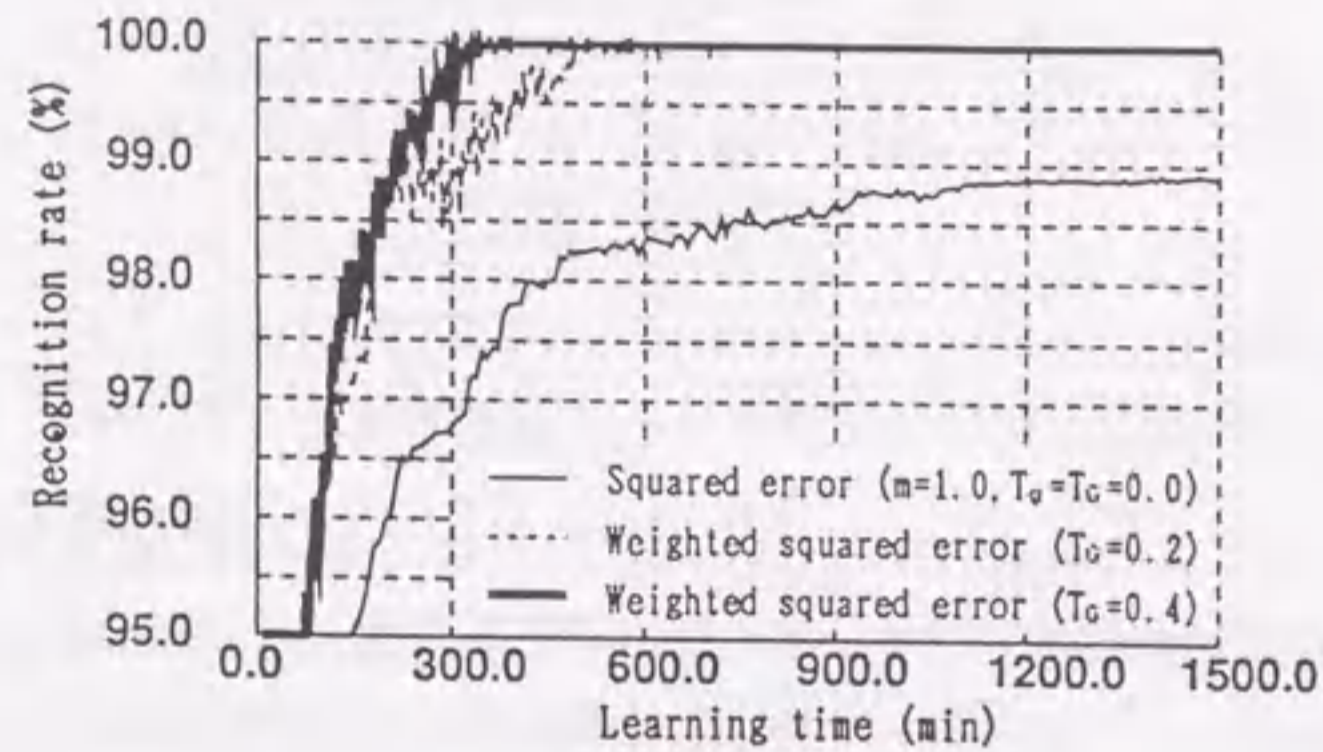
時間はSequent社の並列計算機S 8 1のCPU Timeによる。

図2.5 出力ニューロン毎の誤差評価のしきい値 T_g による学習データの認識率変化 ($m=1.5$, $T_c=0.0$)

(2) 誤差の小さい入力パターンの結合の重み変更をスキップする重み付け関数Gの効果

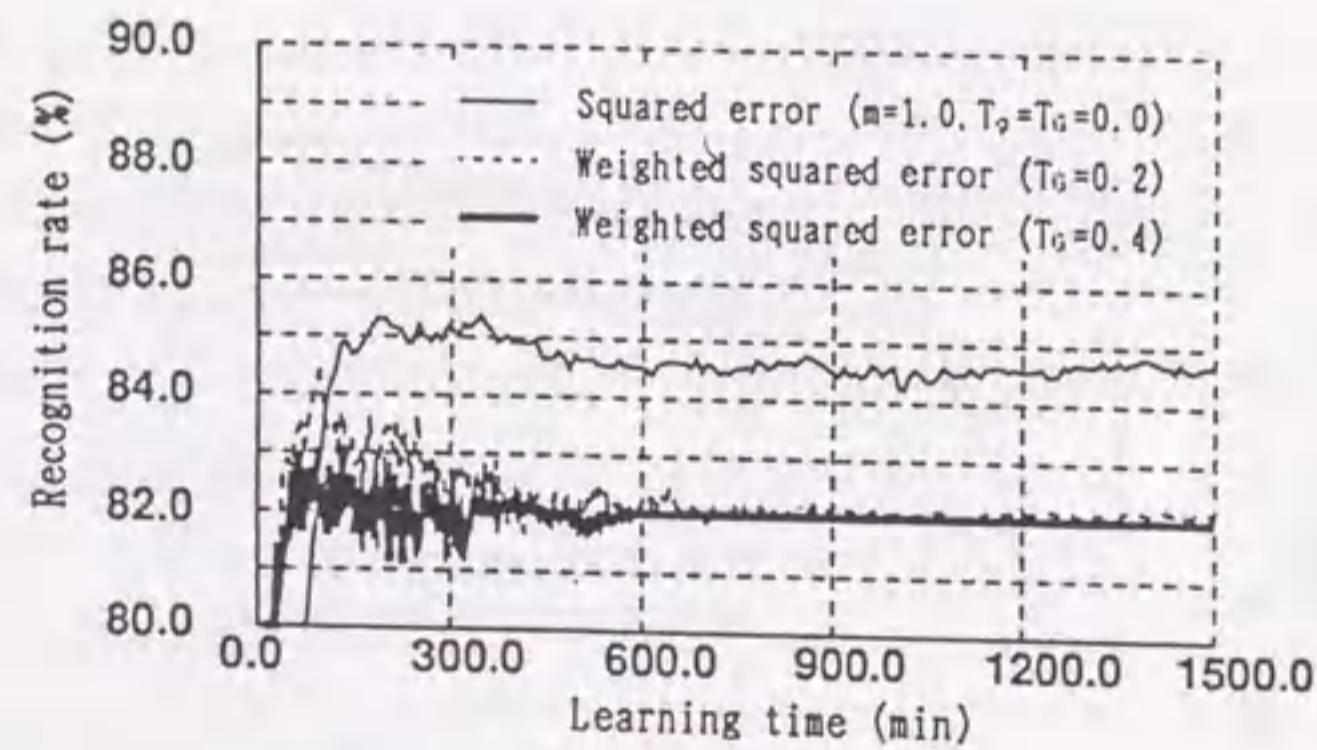
重み付け関数Gの効果を見るために、入力パターン毎の誤差評価のしきい値 T_c による学習及び未学習データの認識率変化を従来のBP法と比較した。

$m=1.5$ 、 $T_g=0.2$ に固定し、 T_c の値に対する学習及び未学習データの認識率の時間変化を図2.6-(a)、(b)に示す。ここで、 $T_c=0.0$ の場合が従来の二乗誤差評価関数によるBP法に相当する。



時間はSequent社の並列計算機S81のCPU Timeによる。

図2.6-(a) 入力パターン毎の誤差評価のしきい値 T_c による学習データの認識率変化 ($m=1.5$, $T_g=0.2$)



時間はSequent社の並列計算機S81のCPU Timeによる。

図2.6-(b) 入力パターン毎の誤差評価のしきい値 T_c による未学習データの認識率変化 ($m=1.5$, $T_g=0.2$)

ある入力パターンに対するすべての出力ニューロンの誤差が0.4以下になればパターン認識としては十分であり、そのパターンについてはそれ以上重みを変更する必要がないことが図2.6-(a)より分かる。

$T_c=0.4$ の場合、認識率99.0%に達するまでの時間は、二乗誤差評価関数の場合に対し約1/7に短縮され、パターン認識の学習の高速化という点でさらに改善されている。また、誤差適応型評価関数の場合は100%の認識率を達成しているが、従来の二乗誤差評価関数では99.0%で飽和しており、重み付け関数の導入が性能向上に有効であることが分かる。

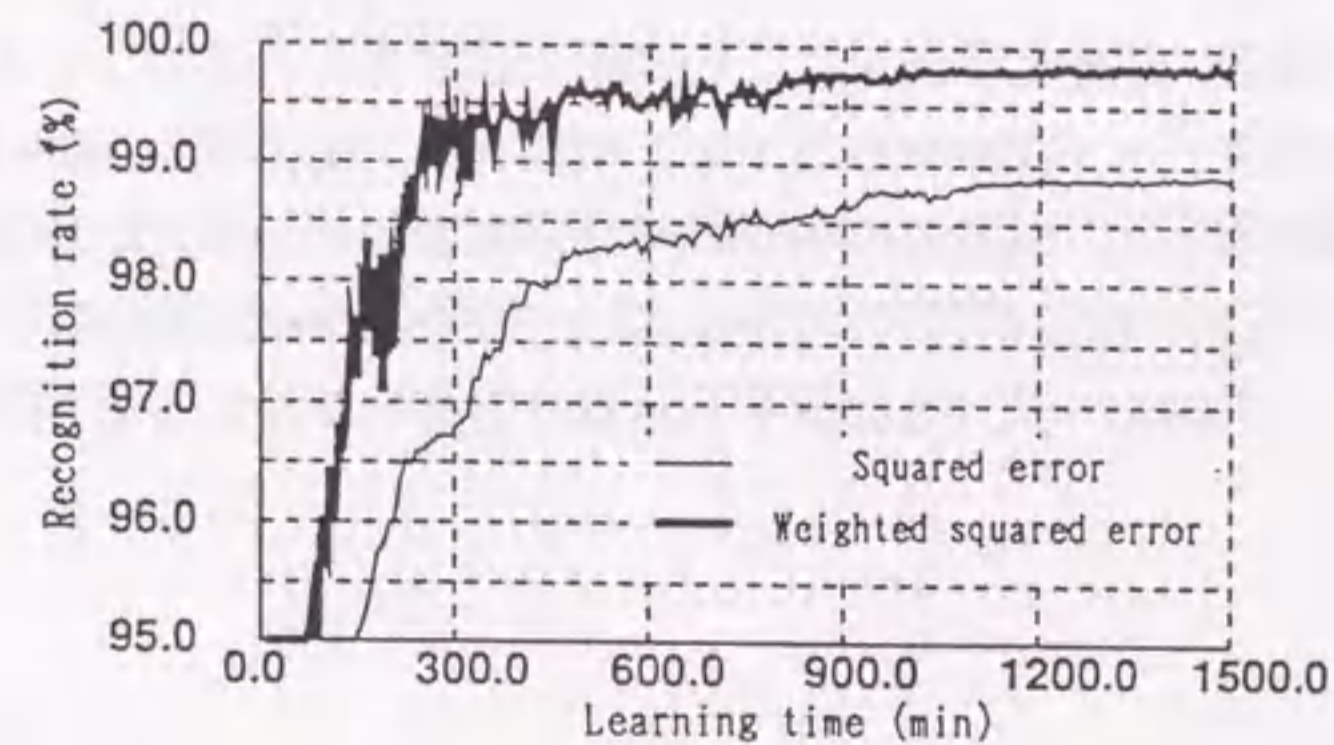
しかし、未学習データの認識率が従来のものに比べて低下しているのが図2.6-(b)より分かる。これは $T_g=0.2$ 、 T_c の値を学習の過程において固定しているために、出力ニューロンの発火状態が不安定なまま学習が終了したことが一因として考えられる。

(3) 誤差評価の場合分けを制御するしきい値 T_g 、 T_c の制御の効果

しきい値 T_g と T_c の制御による学習及び未学習データの認識率変化を従来のBP法と比較した。

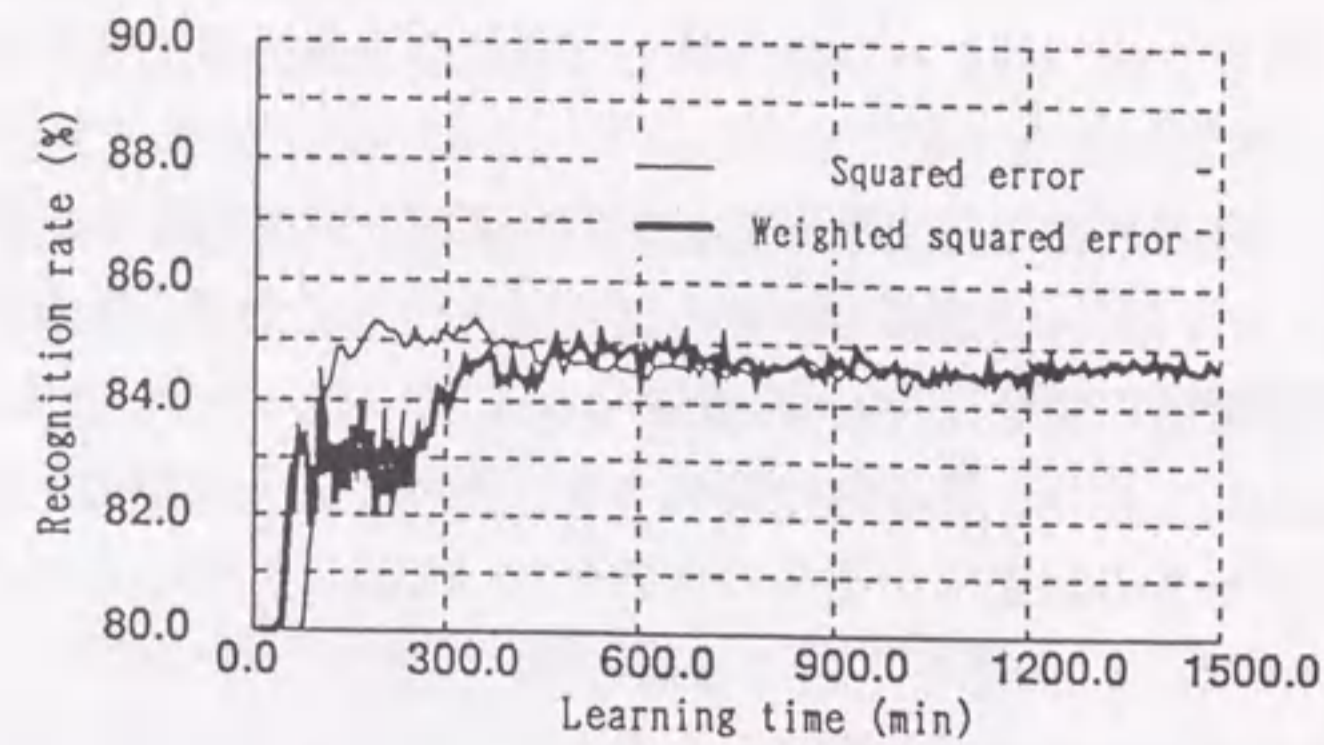
しきい値 T_g と T_c を初期値として0.4に設定し、学習が進むに従い0.1刻みで0.0まで変化させたときの学習及び未学習データの認識率の変化を図2.7-(a)、(b)に示す。 T_g 、 T_c の値の変更時期は、すべての学習パターンの中で誤差の絶対値が0.5を超える出力ニューロンの数が20個以下になった時とした。

学習データの認識率は若干劣化するものの、学習の高速性はほとんど変わらず、未学習データの認識率は大幅に改善された。



時間はSequent社の並列計算機S81のCPU Timeによる。

図2.7-(a) 誤差適応型評価関数による学習データの認識率変化



時間はSequent社の並列計算機S 8 1のCPU Timeによる。

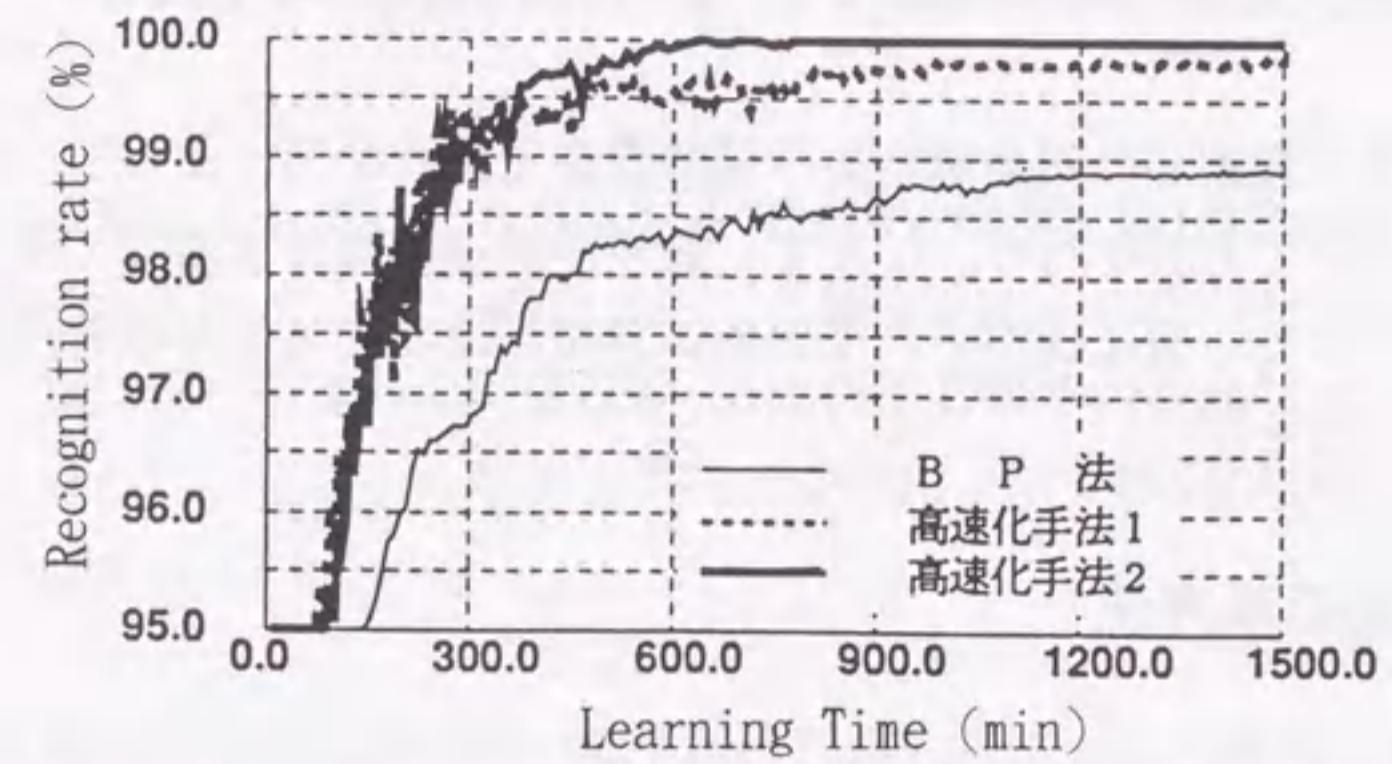
図2.7-(b) 誤差適応型評価関数による未学習データの認識率変化

(4) 学習の後期に非線形最適化手法を導入した効果

学習の後期において、収束がすばやく進み、望ましい出力関数の近似解を得ることができ共役勾配法、直線探索法を従来のBP法と比較した。(3)で示した誤差適応型評価関数のみを導入した高速化手法1と、学習の後半に非線形最適化手法を適用した高速化手法2との比較である。手法1、2とも、 $m=1.5$ 、 $T_s=T_G=0.4\sim 0.0$ とし、手法2では $T_s=T_G=0.0$ となった時点で共役勾配法と直線探索法に切り替えた。

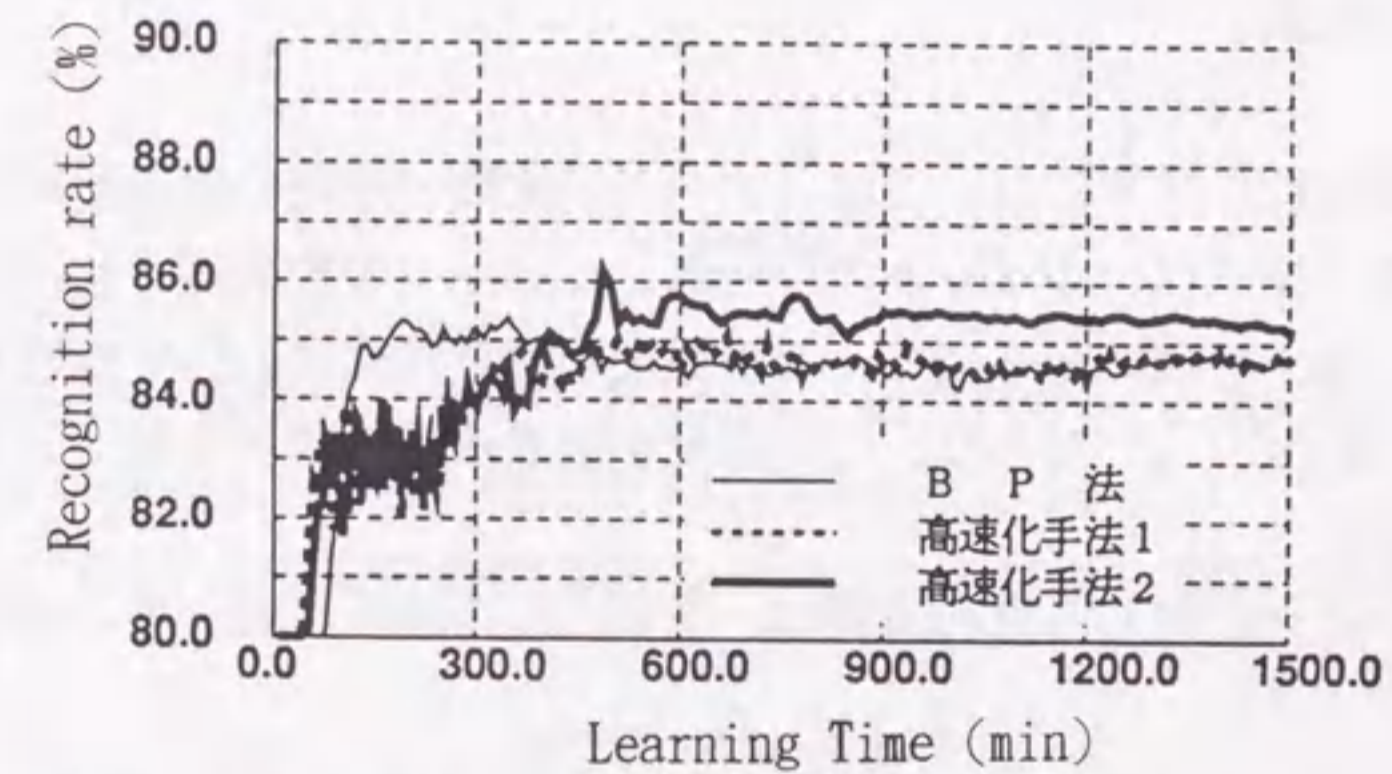
学習時間に対する学習データの認識率を図2.8-(a)に示し、未学習データの認識率の変化図2.8-(b)に示す。参考のために、従来のBP法の結果も示した。認識率が99.0%になるまでの時間を比較すると、高速化手法1、2は従来のBP法に対し約1/5に短縮された。高速化手法2は未学習データ認識率も向上し、学習データ認識率も100%を達成している。

学習の前半で誤差適応型評価関数の誤差評価により各ニューロンの誤差が同じ様になるように学習が進んでいるので、学習の後期において誤差曲面は二乗誤差曲面に近くなっていると考えられ、二乗誤差の減少を促進する非線形最適化手法により学習が進んだと考えられる。



時間はSequent社の並列計算機S 8 1のCPU Timeによる。

図2.8-(a) 誤差適応型評価関数と学習の後期に非線形最適化手法を導入した場合の学習データの認識率変化



時間はSequent社の並列計算機S 8 1のCPU Timeによる。

図2.8-(b) 誤差適応型評価関数と学習の後期に非線形最適化手法を導入した場合の未学習データの認識率変化

2.3 誤差適応型評価関数による演算精度の低減

本節では、前節で提案した誤差適応型評価関数を用いた学習によって、演算精度が低減できる仕組みについて考察を行う。加えて、固定小数点演算によるマルチフォント英数字認識の学習実験により、既に提案されている演算精度低減の可能な学習法よりも本提案手法が有効であることを示す。

2.3.1 演算精度低減方法

演算精度が低いことにより生じる大きな問題点は、丸め誤差による演算誤差の増大と、誤差量が十分小さいにも関わらず誤差評価を行うことによる不必要な演算である。そこで、ここでは丸め誤差の低減、不必要な重み変更の抑制という二つの観点で考察を加え、演算精度低減のための誤差適応型評価関数の特徴を明らかにする。

(1) 丸め誤差の低減

演算精度が限定された学習においては、すべての変数は固定小数点形式で表現される。 n ビット固定小数点形式は、1ビットの符号部、 q ビットの整数部 ($q \leq n-1$)、及び s ビットの小数部 ($s = n-1-q$) からなる。この固定小数点形式において表現し得る値は、最大が $2^q - 2^{-s}$ 、最小が -2^q であり、この固定小数点形式で表現されている値の最小精度は、 2^{-s} である。

演算精度が低いときに生じる丸め誤差は、次の二つの場合において、特に重大な誤差となる。

- ・オーバーフロー $2^q - 2^{-s}$ よりも大きな値は $2^q - 2^{-s}$ に丸められ、 -2^q よりも小さい値は -2^q に丸められる
- ・アンダーフロー 絶対値が 2^{-s-1} よりも小さな値は零に丸められる

しかし、ニューラルネットワークの学習においては、これらの丸め誤差の内オーバーフローはアンダーフローほど重大な誤差ではない。それは、オーバーフローが最も生じ易いのはニューロンの入力と結合の重みとの積和演算であるが、シグモイド関数の飽和特性によって、積和の大きな値を精度よく計算する必要がないからである。それ故、演算精度の低い場合の学習においてはアンダーフローの防止が重要である。

従来のBP法においては、出力ニューロンの逆伝搬信号は前掲した式(2.5)に従って求められる。

$$\delta_j = (t_j - y_j) f'(x_j) \quad (2.5)$$

この式を見ると分かるように、従来のBP法では出力信号 y_j が教師信号 t_j と大きく異なっても、微係数 $f'(x_j)$ が小さければ δ_j は零に丸められ、有効な δ_j の逆伝搬ができなくなる。BP法における重み変更量は δ_j に比例するので、 δ_j が0に丸められると有効な重み変更ができない。このようなアンダーフローによって学習が非効率な状態に陥ることは避けなければならない。

提案する誤差適応型評価関数を用いた学習法においては、出力信号 y_j が教師信号 t_j と大きく異なっているときには、逆伝搬信号は従来のBP法における値よりも大きな値とすることによってアンダーフローを低減できる。具体的には、

$$\delta_j = \{g_j(t_j - y_j)\} f'(x_j) \quad (2.10)$$

とする。式(2.10)を式(2.5)と比較すれば分かるように、提案学習法では、従来のBP法における逆伝搬信号を g_j で重み付けした逆伝搬信号を用いる。 g_j を重み付け係数と呼び、

$$g_j = \begin{cases} m (> 1) & \text{if } |y_j - t_j| \geq T_g \\ 1 & \text{otherwise} \end{cases} \quad (2.11)$$

によって値を設定する。

図2.7に、重み付け係数 g_j の設定の例を示す。出力信号 y_j と教師信号 t_j との差が大きいときには、その差を $m (> 1)$ 倍に大きく評価する。これによって、シグモイド関数の微係数が小さくても、逆伝搬信号 δ_j がアンダーフローによって零に丸められてしまうことを防ぐ。

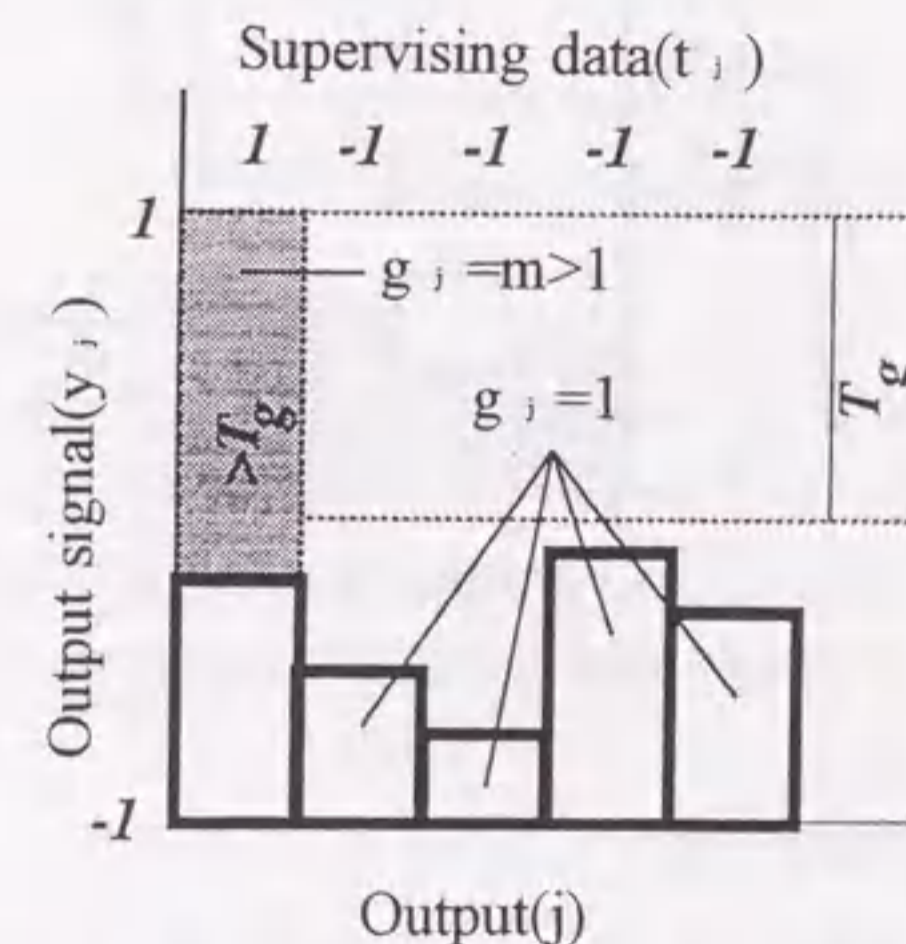


図2.7 重み付け係数 g_j の設定例

(2) 不必要な重み変更の抑制

学習データのパターン数が多いときには一つの入力パターン毎に重みを変更する逐次修正方式 [27] が多く用いられる。この場合、重み変更量は他の入力パターンに関係なく決定されるために、重み変更によって他の入力パターンに関しては誤認識を引き起こす可能性がある。特に、その入力パターンの認識結果が正しいときに重みを変更することは、他の入力パターンの誤認識によって認識率の低下を招く可能性が高い。

提案する学習法においては、すべての出力信号と教師信号との差がある値よりも小さい入力パターンに関しては重み変更を行わない。具体的には重み変更量 Δw_{ji} を式 (2.4) で示したものに係数 G を掛けた

$$\Delta w_{ji} = G (\varepsilon \delta_j y_j + \alpha \Delta w'_{ji}) \quad (2.12)$$

とする。 G をスキップ係数と呼び、

$$G = \begin{cases} 0 & \text{if } |y_j - t_j| \leq T_G, j = 1 \sim N \\ 1 & \text{otherwise} \end{cases} \quad (2.13)$$

によって値を設定する。

図2.8に、スキップ係数 G の設定の例を示す。出力信号 y_j と教師信号 t_j との差がすべての出力層ニューロンにおいて小さい ($|y_j - t_j| \leq T_G$) 時には、重み変更量を零とし重み変更を行わない。このように、出力誤差の小さい入力パターンについては重みを変更しないことによって、不必要な重み変更による認識率の低下を防ぐ。

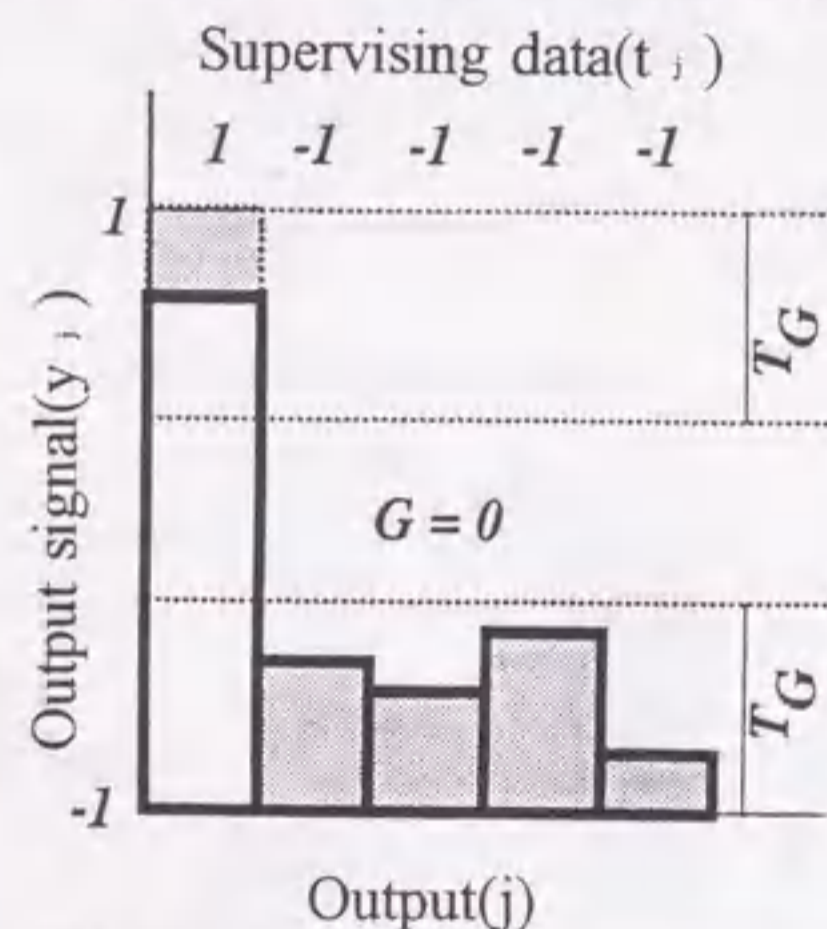


図2.8 スキップ係数 G の設定例

(3) 誤差適応型評価関数の定式化

(1)、(2) の考察から、2.2.2項の(1)で定義した誤差適応型評価関数を再度定式

化すると以下ようになる。

重み付け処理とスキップ処理に基づく提案の学習法は、式 (2.14) に示す評価関数を最小化している。

$$E = G \sum_j \{g_j (y_j - t_j)^2\} \quad (2.14)$$

この評価関数では、 $(y_j - t_j)$ の値をその誤差の重要度に応じて、

$$\begin{cases} m (y_j - t_j) & \text{if } |y_j - t_j| \geq T_g \\ 0 & \text{if } |y_j - t_j| \leq T_g \\ (y_j - t_j) & \text{otherwise} \end{cases} \quad \text{for all } j \quad (2.15)$$

と評価する。

この提案学習法が有効に働くことを示すための予備実験を行った。

(a) アンダーフロー低減の重要性

表2.1は、低い演算精度の固定小数点演算で従来のBP法を行った場合に、最高の認識率を与える学習率の値を示す。実験に用いたデータは次に示すマルチフォント英数字認識の学習実験に用いたものと同じである。従来のBP法においては、重み変更量を逆伝搬信号 δ_j に学習率 ε を掛け合わせて求めるので、学習率を大きく設定することによってアンダーフローを低減している。従来のBP法においても、演算精度が低くなるとアンダーフローを低減することが重要になることが分かる。

表2.1 演算精度が限定された学習における学習率の最適値

演算精度	学習率
3.2 bit	0.005
1.6 bit	0.010
1.2 bit	0.020
1.0 bit	0.050

(b) 学習に用いられる変数の値の分布

誤差適応型評価関数を最小化する提案学習法において、アンダーフローが低減できているかどうかについて調べるために、学習に用いられるすべての変数の絶対値の分布を調べた。マルチフォント英数字認識の学習を倍精度 (64ビット浮動小数点演算の精度) 演算で行ったときの変数値の分布を図2.9に示す。この実験では、 $T_g=1$ 、 $m=2$ 、 $T_c=\sqrt{0.5}$ 、学習率及び慣性項の係数はどちらの学習でも $\varepsilon=0.005$ 、 $\alpha=0.9$ とした。誤差適応型評価関数を用いた学習では、二乗誤差評価関数を用いる従来の学習法に比べて、学習に用いられるすべての変数値において、絶対値の小さな値が学習に用いられる頻度が低く、アンダーフローの起こる頻度が低い。従って、提案学習法によれば、演算精度の低い学習におい

ても高い認識率が得られるものと期待できる。

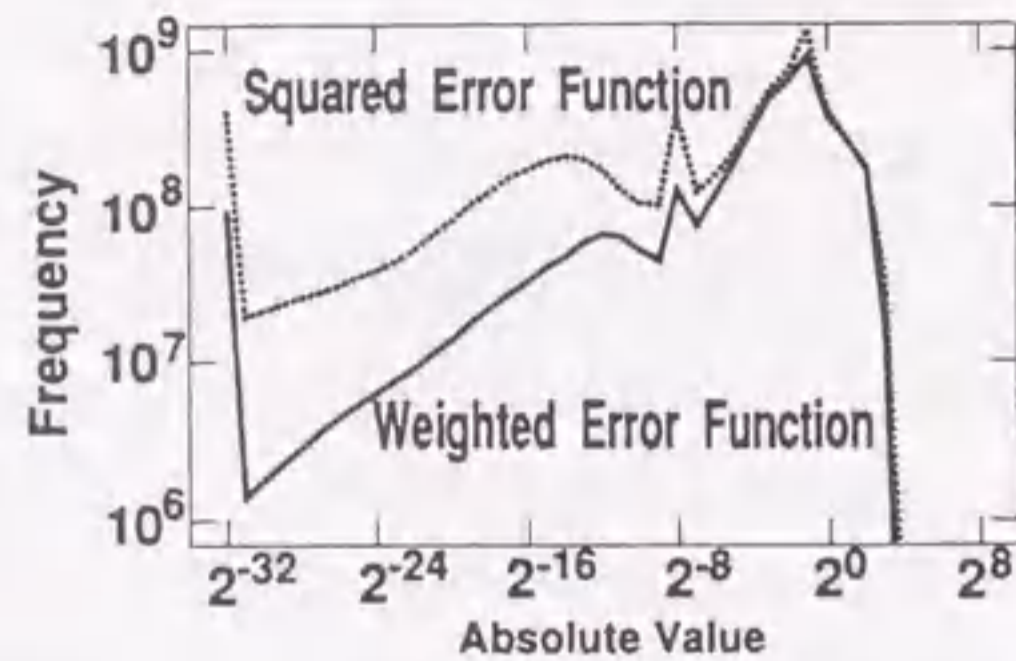


図2.9 学習に用いられる変数の絶対値の分布

2.3.2 精度低減の実験

演算精度が限定された学習における提案学習法の性能評価実験を、13フォント英数字認識及び単一フォント英数字認識の学習について行った。限定精度の学習実験においては、すべての演算（乗算、加算などを含む）を一定の固定小数点演算で実行し、すべての変数（重み、逆伝搬信号、入出力信号、その他の中間的演算結果を含む）を一定の精度の固定小数点で表現した。

(1) 13フォント英数字認識の学習実験

実用的学習問題における提案学習法の性能を評価するために、マルチフォントの英数字認識（62カテゴリ）の学習実験を行った。

重みの変更は、1文字入力される毎に行った（誤差適応型評価関数を用いた学習ではスキップされる可能性もある）。13フォント（806文字）に対する重み変更を1回の学習と定義する。1回の学習が終了する毎に、学習に用いた文字に対する学習認識率、及び学習に用いなかった別の10フォントに対する未学習認識率を評価する。これは2.2.3項と同じものである。

学習率 ϵ は、誤差適応型評価関数を用いた学習においても表2.1の値を用いた。また、どちらの学習においても慣性項の係数は $\alpha = 0.9$ とした。

パターン認識などの学習問題においては、教師信号は2値とすることが多く、出力信号と教師信号との誤差が二つの教師信号との差の半分よりも大きいと誤認識を生ずる。重み付けのしきい値 T_c は、この二つの教師信号の差の半分とするのが適当である。本実験では

値 -1 と 1 を教師信号としたので、 $T_c = 1$ とした。重み付けの定数 m は $2 \sim 4$ の値に設定するのが適当で [29]、本実験では 2 とした。学習を効率よく進めるためには、スキップのしきい値 T_c は二つの教師信号の差の半分より小さくする必要がある。学習が進むにつれて T_c を小さくすると、学習の最終曲面で出力信号を教師信号に十分近づけることができ、学習文字と少し異なる未学習文字に対して望ましい出力を得ることができる。本実験では、 T_c の初期値を $\sqrt{0.5} = 0.7$ に設定し、学習文字の認識率が 90% を超える毎に T_c を半分にした。

限定精度の学習における丸め誤差を小さくするためには、 n ビット固定小数点表現における整数部の幅 ($q \leq n-1$) 及び小数部の幅 ($s = n-1-q$) として、最適な値を選択する必要がある。図2.9の学習に用いられた値の分布において、固定小数点表現によるオーバーフロー（絶対値が 2^q 以上）やアンダーフロー（絶対値が 2^{-s-1} 以下）ができるだけ少なくなるように、 q 及び s を表2.2に示すように選択した。固定小数点表現の最大値 ($= 2^3$) は、Bakerらの実験 [21] における値と一致している。

表2.2 n bit固定小数点表現における整数部 (q bit) 及び小数部 (s bit) のビット割当

n	q	s
32	3	28
16	3	12
12	3	8
10	3	6

(a) 演算精度の低減

図2.10は限定精度の学習において、学習が進むにつれて認識率が上昇していく様子を示している。これらは重みの初期値を5通りに変えた実験結果の平均値を示している。この実験では中間層ニューロン数は56に固定している。図2.10に示されるように、誤差適応型評価関数を用いることによって、学習の速さも学習後の認識率も大幅に改善されている。10ビットの演算精度で従来の二乗誤差評価関数を用いた学習を行うと、200回学習後の認識率は、学習文字に対して 66.7% 、未学習文字に対して 54.5% であり、誤差適応型評価関数を用いることにより $25 \sim 30\%$ 程度改善されている。

(b) 中間層ニューロン数と認識率の関係

学習によって得られる認識率は、演算の精度だけでなく中間層のニューロン数にも依存する [27] [23]。そこで中間層のニューロン数を $84, 56, 42$ と変化させ、同時に演算の精度も変化させたときの、200回学習後の認識率を調べた。実験結果を簡明に示すためにニューラルネットワークの重みを記憶するためのメモリ量と200回学習後の認識率との関係を

図2.11に示す。重みを記憶するためのメモリ量Mは、入力層が64ニューロン、出力層が、62ニューロンであることから、しきい値処理のための重みも合わせると、

$$M = n \{65H + 62(H+1)\} \text{ ビット} \quad (2.16)$$

によって与えられる。但し、nは学習に用いられた固定小数点表現のビット幅であり、Hは中間層のニューロン数である。

誤差適応型評価関数を用いることによって、中間層ニューロン数及び変数のビット幅を低減したことによる認識率の劣化は、従来の学習法よりも少なくなっている。特に、重みを記憶するメモリ量を少なくした場合には、誤差適応型評価関数を用いることによる認識率の改善は大きい。

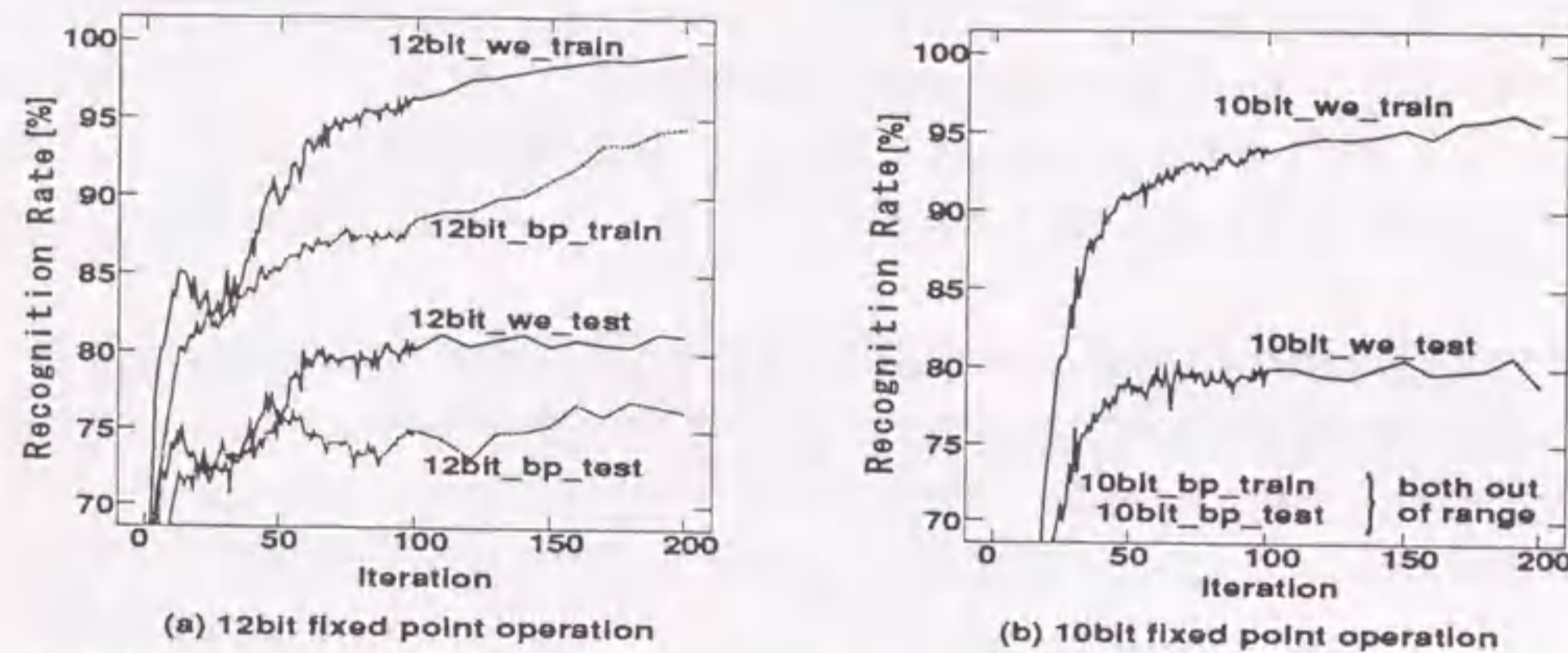


図2.10 固定小数点演算によるマルチフォント英数字認識(62カテゴリ)の学習における認識率(-weは誤差適応型評価関数を用いた学習を表し、-bpは従来の2乗和評価関数を用いた学習を表す、-trainは学習に用いた13フォントの文字に対する認識率を表し、-testは学習には用いなかった別の10フォントの文字に対する認識率を表す)

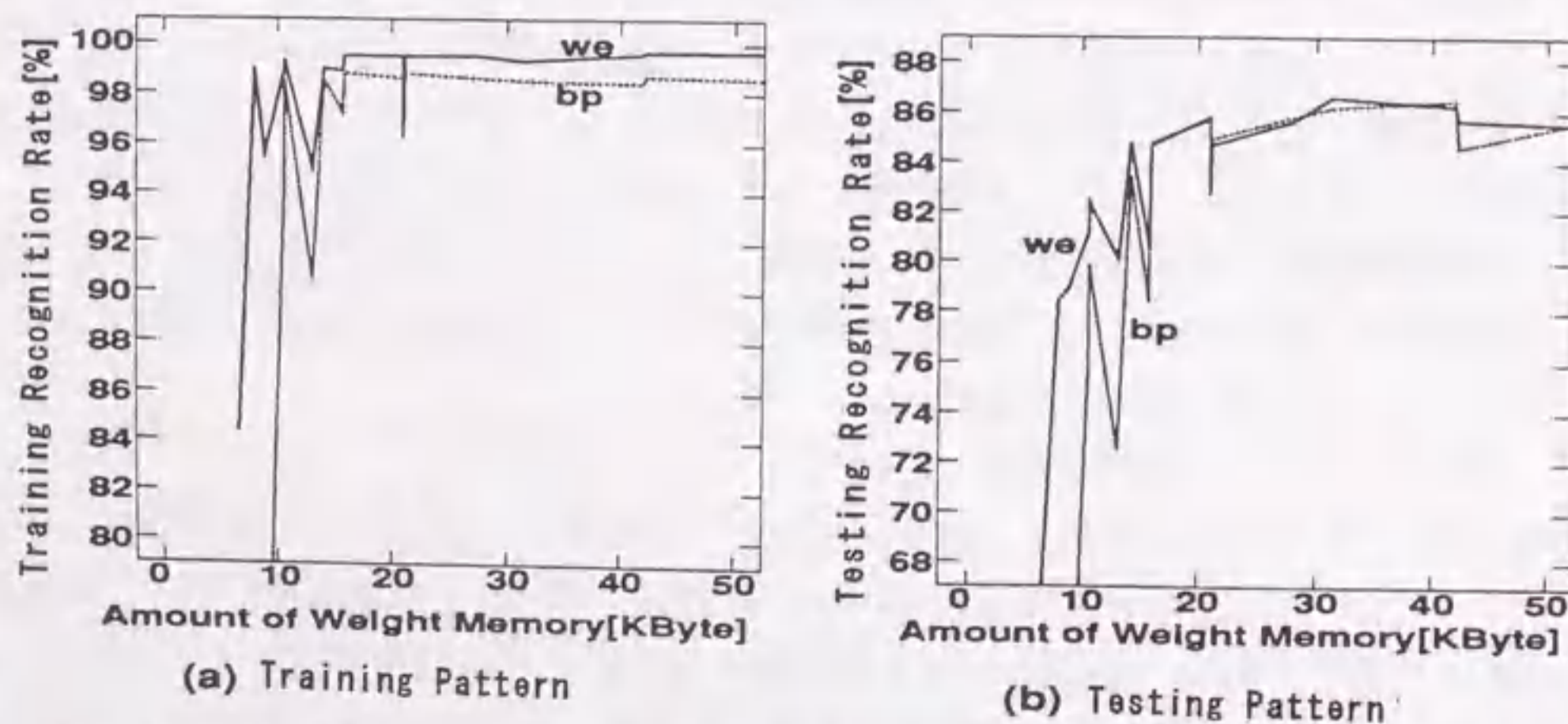


図2.11 重みを記憶するためのメモリ量と200回学習後の認識率との関係(-weは誤差適応型評価関数を用いた学習を表し、-bpは従来の2乗和評価関数を用いた学習を表す)

(1) 単一フォント英数字認識の学習実験

学習問題の難易による影響を調べるために、固定小数点演算による単一フォント英数字認識の学習実験を行った。

学習に用いたデータは、マルチフォント英数字認識の学習実験に用いたデータの第一フォントであり、ニューラルネットワークの構成は、入力層64ニューロン、中間層56ニューロン、出力層62ニューロンである。表2.3に8ビット及び7ビット精度の学習における整数部/小数部のビット割当、及び学習率を示す。その他の実験条件は13フォントの学習実験と同じである。

表2.3 単一フォント英数字認識の学習実験における固定小数点表現のビット幅(n)、整数部のビット幅(q)、小数部のビット幅(s)及び学習率

n	q	s	ϵ
8	2	5	0.1
7	2	4	0.1

(a) 演算精度の低減

図2.12に、固定小数点演算の学習によって認識率が上昇していく様子を示す。これらは、いずれも重みの初期値を5種類に変えて行った実験の平均値である。7ビット精度で500回学習を行った後の認識率は、誤差適応型評価関数を用いた学習法では、5回の試行の内3回は100%、1回は98.4%、1回は96.8%であった。誤差適応型評価関数を用いることで、7ビット固定小数点の精度で学習しても、ほぼ十分な性能が得られることが分かる。一方、従来の二乗誤差評価関数を用いるBP法においては、8ビット精度で学習しても、学習回数が200を超える辺りから、学習を進めるにつれて認識率が下がり、500回学習を行った後の認識率は平均75.8%となった。これは、二乗誤差評価関数を用いる学習においては、ある文字の認識結果が正しいときにも重みを変更するために、他の文字に対する認識結果が誤ってしまうことによる。この傾向は、マルチフォントの英数字認識の学習実験においても、従来の学習法に関してみられることを確かめている。

(b) 13フォントの学習実験との比較

いずれの学習法でも、マルチフォント英数字認識と単一フォント英数字認識とでは、必要な演算精度が大きく異なる。単一フォントでは一つの教師信号に対する入力パターンが一つしかないため、入力信号に対する分離の境界は比較的単純でよいのに対し、マルチフォントでは一つの教師信号に対する入力パターンが複数あるので、入力信号に対する分離の境界は複雑になるためと考えられる。

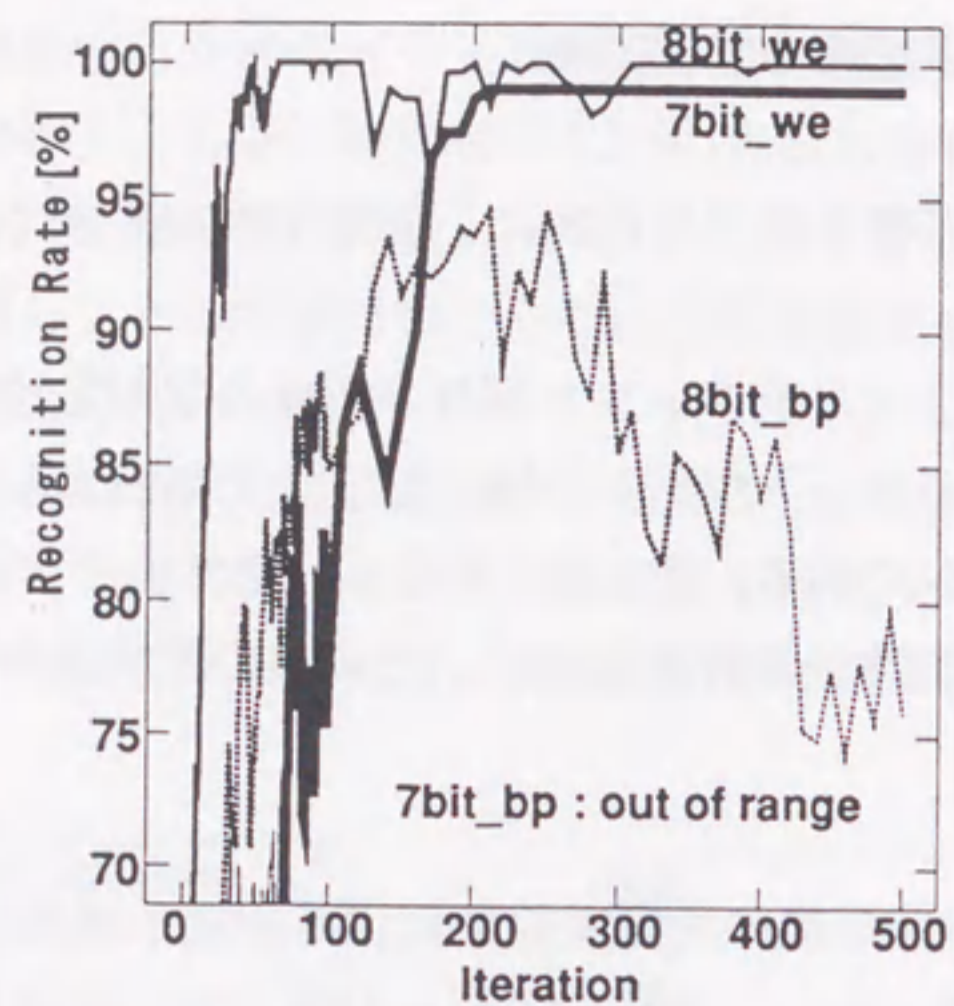


図2.12 固定小数点演算による単一フォント英数字認識の学習実験 (-we は誤差適応型評価関数を用いた学習を表し、-bp は従来の2乗和評価関数を用いた学習を表す)

2.4 まとめ

本章ではパターン認識に適したニューラルネットワークの学習法として、誤差の大きさに応じて重み付けをする関数を導入した誤差適応型評価関数を提案し、マルチフォント英数字認識に適用してその有効性を検討した。その結果、従来の二乗誤差評価関数による学習に比べて、学習時間が約1/5以下に短縮され、未学習認識率も向上できることを確認した。さらに、この提案学習法は、逆伝搬信号を重み付けするために、アンダーフローを低減することができ、出力誤差が小さいときは重み変更をスキップするために、不要な重み変更による認識率の低下を防ぐことができるので、精度が限定された状態においても学習が有効に行われ、低い演算精度でも高い性能を維持することができることを示した。

第3章

量子化ニューロンモデルを用いた ニューラルネットワークと その高速学習と追加学習

3.1 はじめに

使用者の好みや使用される環境に適応して、動作させることができる柔軟な機器・システムが求められている。ニューラルネットワークは、その学習性から状況に応じた判断ができると期待されている。しかし、このような適応性は、単に学習が高速であるだけでは、実現は困難である。既に学習で獲得した動作を保証しつつ、新しい学習データに対する動作を獲得する必要がある。つまり、学習済みのデータへの影響が少なく、しかも高速な追加学習機能が要求される。

パターン認識に適したニューラルネットワークは、以下の3種類に分類できる。BP法による学習を行う階層型ニューラルネットワーク、学習ベクトル量子化(LVQ)ネットワークとリストラクティッド・クローン・エネルギー(RCE)ネットワークである。ここで、BP法[2][40]は、非常に高い学習性を示すものの、ネットワークの動作を決める結合の重みが分散表現されている。言い換えれば、追加パターンの学習の過程で、結合の重みの変更がネットワーク全体で行われるために、学習済みのパターンの認識性能に影響が出易い問題がある。学習ベクトル量子化[41]とRCEネットワーク[42]は、いずれも入力パターンに対する最近傍ニューロンの選択、あるいは追加パターンに応じた中間層ニューロンの生成と、それに関する局所的な結合の重み変更を行う学習である。従って、追加学習を行っても学習済みのパターンに関する結合の重みに影響は出にくいと考えられる。しかし、最近傍ニューロンの選択時あるいは中間層ニューロンの追加時に、入力パターンベクトルとすべての参照ベクトルとの距離計算に長い時間を要するという問題がある。このようなことから、従来のニューラルネットワークモデルでは、追加学習時に学

習済みのデータと追加データとをまとめて再学習する必要性や、学習に長い時間を要することから、状況に応じた適応的な学習が必ずしも容易ではなかった。

本章では、認識のための入力特徴量を量子化する多段階のしきい値処理機能を有する量子化ニューロンを提案する。そして、これを入力特徴量の種類に応じて階層的に接続し、学習を行う最終層で統合化するニューラルネットワークを構築し、高速学習性、追加学習性を評価する。これをマルチフォント英数記号認識に適用し、学習済みの認識への影響が、従来のものに比べて少なく、高速で良好な追加学習性が得られることを示す。さらに、ネットワークを構築するに際し、入力特徴量空間の性質を予め調べることなく、学習データから自動的にネットワーク構造を決定していく、適応増殖量子化ニューロンへの拡張を行う。この拡張によっても、その追加学習性は損なわれないことも示す。

以下、3.2節で量子化ニューロンを用いたニューラルネットワークを提案し、その高速学習性、追加学習性を明らかにし[43][44][45][46]、3.3節で適応増殖量子化ニューロンを用いたニューラルネットワークを提案し、高速学習性と追加学習性が損なわれないことを示す[47][48][49]。そして、3.3節でまとめを行う。

3.2 量子化ニューロンモデルを用いた ニューラルネットワーク

生体の神経回路網の生理的研究の進歩により、ネットワークの構造や機能の新たな知見が得られつつある。BP法で用いられるような単純ニューロンモデル以外の機能を持つニューロンや、特定の機能を有するネットワークの存在も認められつつある。

そこで本節では、良好な追加学習機能を実現するための、ニューロン構造とネットワーク構造を示す。単純ニューロンではなく、多段階のしきい値処理を有し、認識のための入力特徴量を量子化する量子化ニューロンを提案する。そして、これを階層的に接続し、学習による動作の獲得を行う機能別階層ネットワーク (MFLN: Multi Functional Layered Network) を構成し、その学習性能、追加学習性を明らかにする。

3.2.1 量子化ニューロンモデル

現在、広く実用化研究されているニューラルネットワークのニューロンは、多入力出力のニューロンで、積和演算としきい値処理を行うものである。これに対し、入力情報に対する既知の知識を活用するために、多段階のしきい値処理を行う量子化ニューロンを提案する。量子化ニューロンの構造と動作は以下に示すものである。図3.1に量子化ニューロンの基本構造を示す。量子化ニューロンは、量子化信号入力端子Rから入力された入力データxをNレベルで量子化し、この量子化値に応じて、出力端子に連結した次層の量子化ニューロンqを選択する。この次層のニューロンの選択に際し、複数のニューロンを重み付けて選択する。つまり、量子化値に対応する一つのニューロンだけを選択するのではなく、他の量子化レベルである可能性を考慮するように、予め設定した特性に基づいて、次層の各ニューロンiとの結合係数 τ_i を、選択ニューロンqを中心として分布するように決定する。選択信号入力端子Sから入力された選択信号tを、この τ_i で重み付けすることで、式(3.2)に示すように出力 y_i を求め、次層の量子化ニューロンの選択信号入力端子Sへの入力とする。

$$\tau_i = g(x, q) \quad (3.1)$$

$$y_i = t \times \tau_i \quad (3.2)$$

$g(x, q)$ はニューロンの発火の分布を示す関数で、ガウス関数などを用いれば良いが、計算を容易にするために、これを二次関数で表現した式(3.3)を用いる。

$$\tau_i = 1 - \beta (i - x)^2 \quad (3.3)$$

x: 入力データ

以上の量子化ニューロンの動作をまとめると、入力データを離散値に量子化して位置情

報に変換し、各位置ごとの結合の重みを適応的に設定することにより、多重にしきい値処理を行うことになる。図3.2は三つの出力端子を持つ量子化ニューロンを例に、この動作の概要を示したものである。入力データが中程度の場合、中を担当する端子の結合係数が大きく設定され、大及び小を担当する出力端子の結合係数は小さく設定される。入力データが大、小の場合にも同様に、大~小を担当する結合係数が、適応的に設定される。

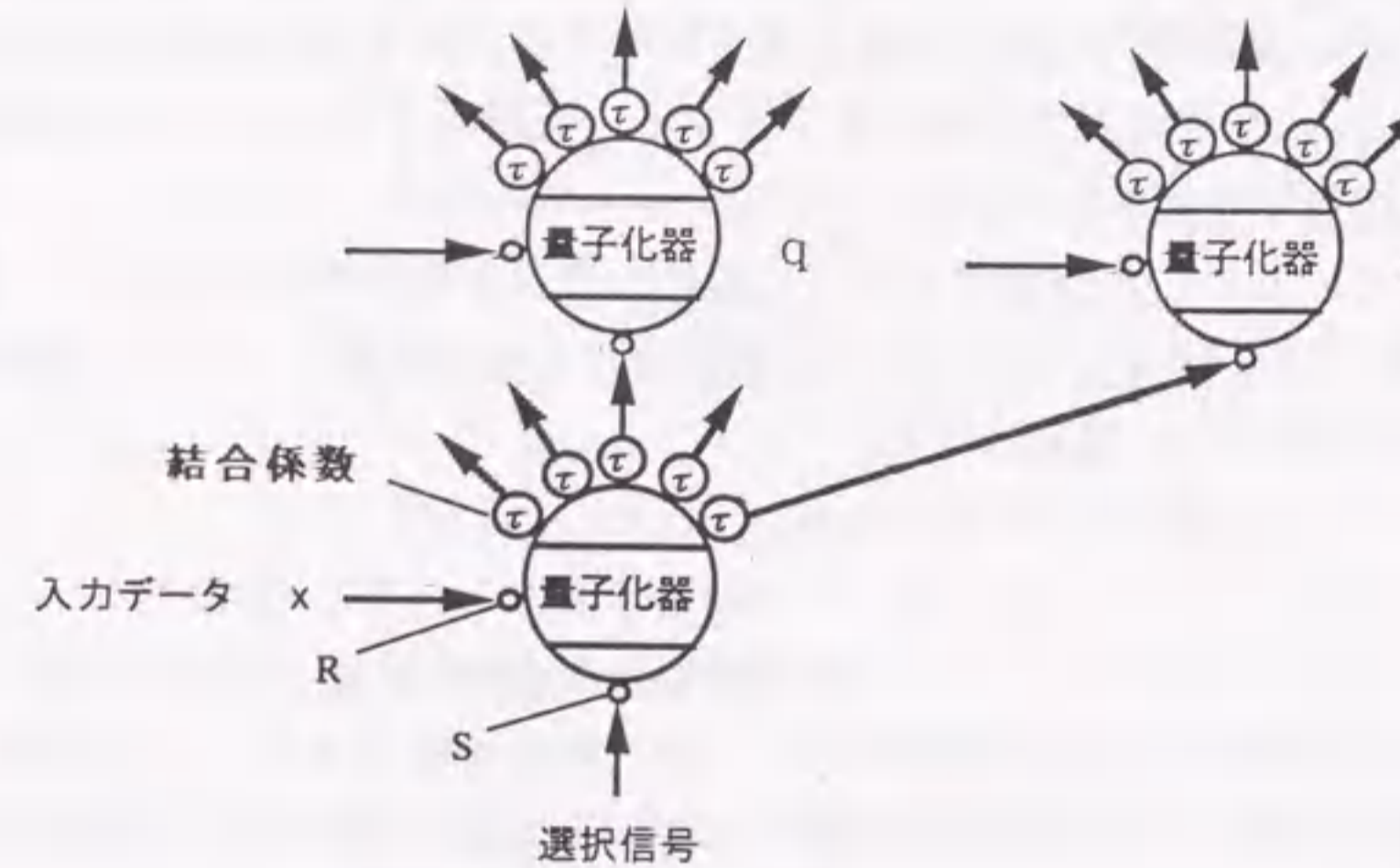


図3.1 量子化ニューロンの基本構造

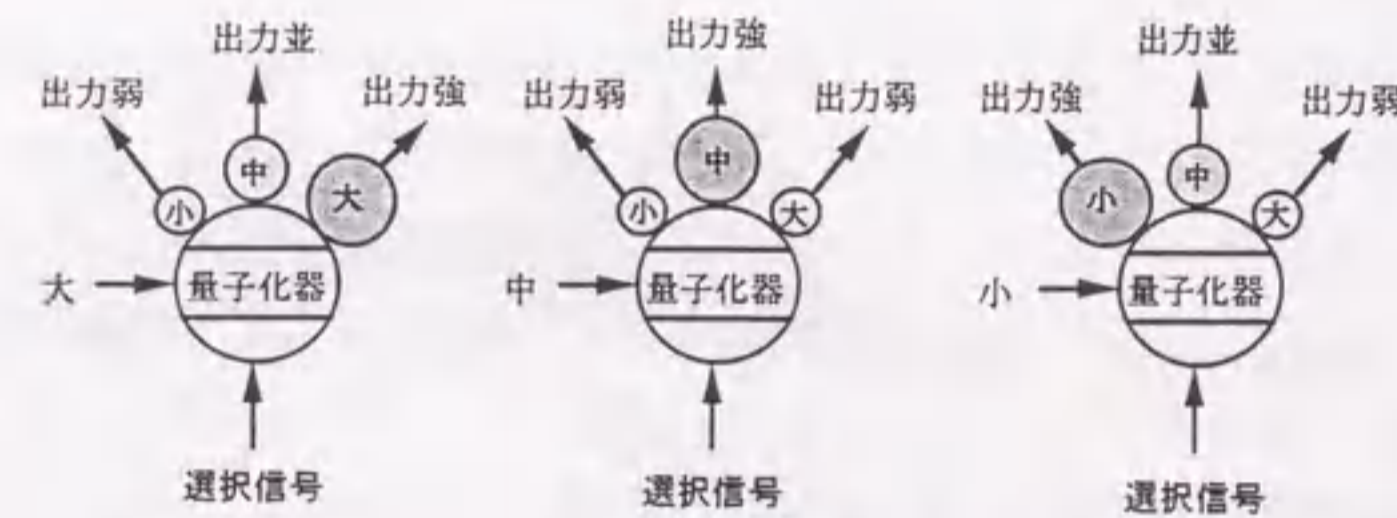


図3.2 量子化ニューロンの動作

3.2.2 量子化ニューロンモデルを用いた 機能別階層ネットワーク (MFLN)

入力データの次元数に対応して、前節で提案した量子化ニューロンを並べ、各入力データを量子化する量子化ニューロン層を構成し、入力データの種類に応じてこのニューロン

層を階層的に結合させる。この各層の量子化結果を最終的に統合する層を、従来の多入力一出力のニューロンで構成する。この最終層の結合の重みは学習により形成させる。

具体的な構成を、マルチフォント英数字認識に用いる場合について図3.3に示す。量子化すべき入力データに応じて、第1～3層の量子化ニューロン層を設け、それらを階層状に結合する。このように、各層が予め判断を行うようにし、最終層には通常のニューロンを結合する。第1層は文字画像をブロックに分割した時の各ブロックの濃度情報、第2層は縦線成分を表現する各ブロック間の濃度情報の横方向微分値、第3層は横線成分を表現する各ブロック間の濃度情報の縦方向微分値を判断する。第4層は教師入力層で、最終層との結合係数 τ_{ijkhp} を教師入力に基づき学習する。各層のニューロンの結合係数に基づく選択信号の分布の様子を図3.4に示す。

学習は、間違っているところを修正していく教師付きHebb学習で行なう。最終層は、式(3.4)に示すように、前層ニューロンからの選択信号と結合係数 τ_{ijkhp} との積和を求め、しきい値関数 f に基づいて y_p を出力する。

$$y_p = f(\sum T \cdot \tau_{ij} \cdot \tau_{ijk} \cdot \tau_{ijkh} \cdot \tau_{ijkhp}) \quad (3.4)$$

但し、 i, j, k, h : 1～3層の各層におけるニューロン番号

p : 最終層のニューロン番号

以上のように、量子化により予め判断を行い、その結果を統合するところの層のみの学習を行うこと、及び間違っているところを修正していく教師付きHebb学習を行うという二つの特徴を持つことから、学習で変更すべき結合数が少なく、学習が高速になるということと、既に学習済みの状態に与える影響を少なくできるので、追加学習が容易になるということが期待される。

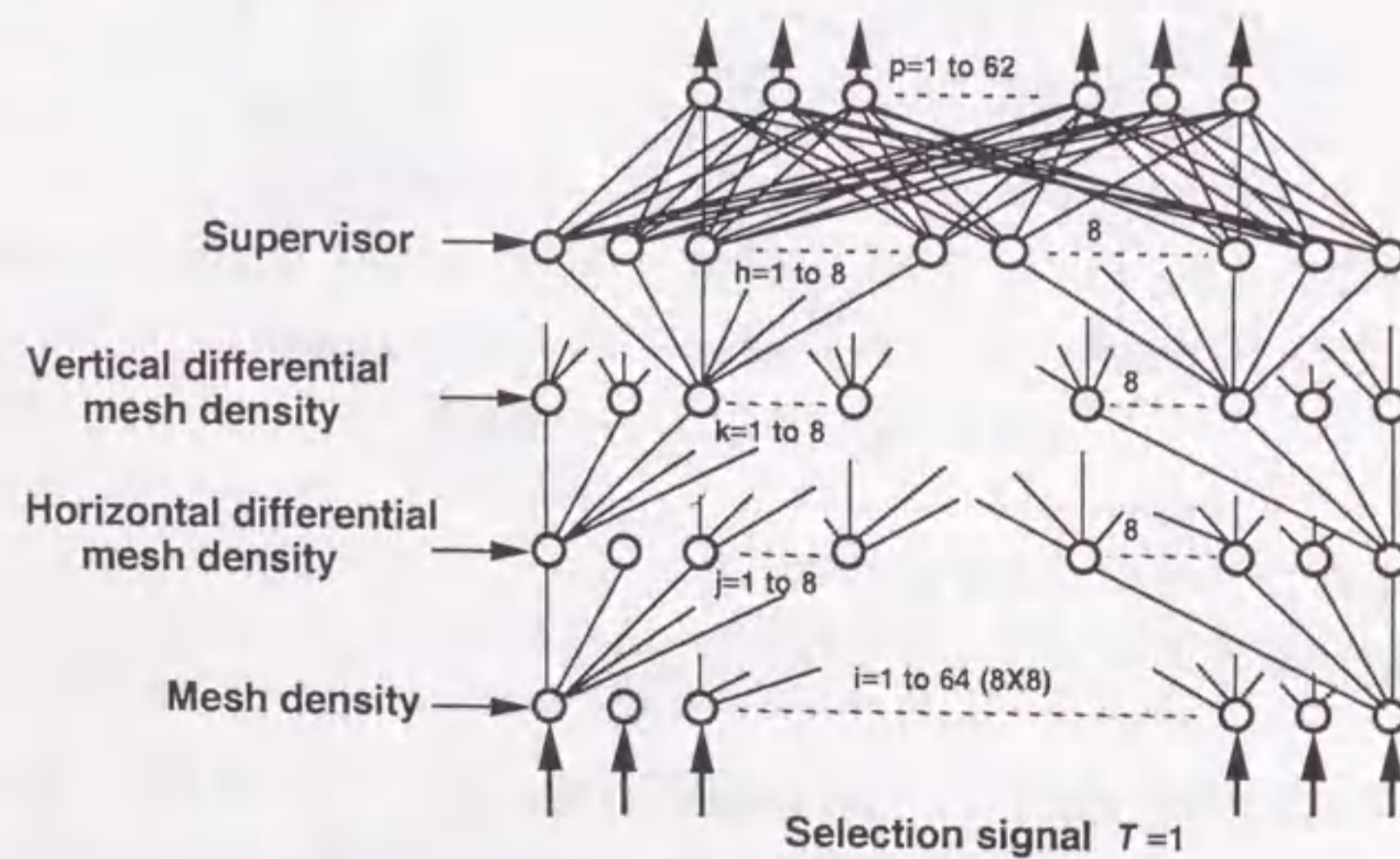


図3.3 機能別階層ネットワークの構造

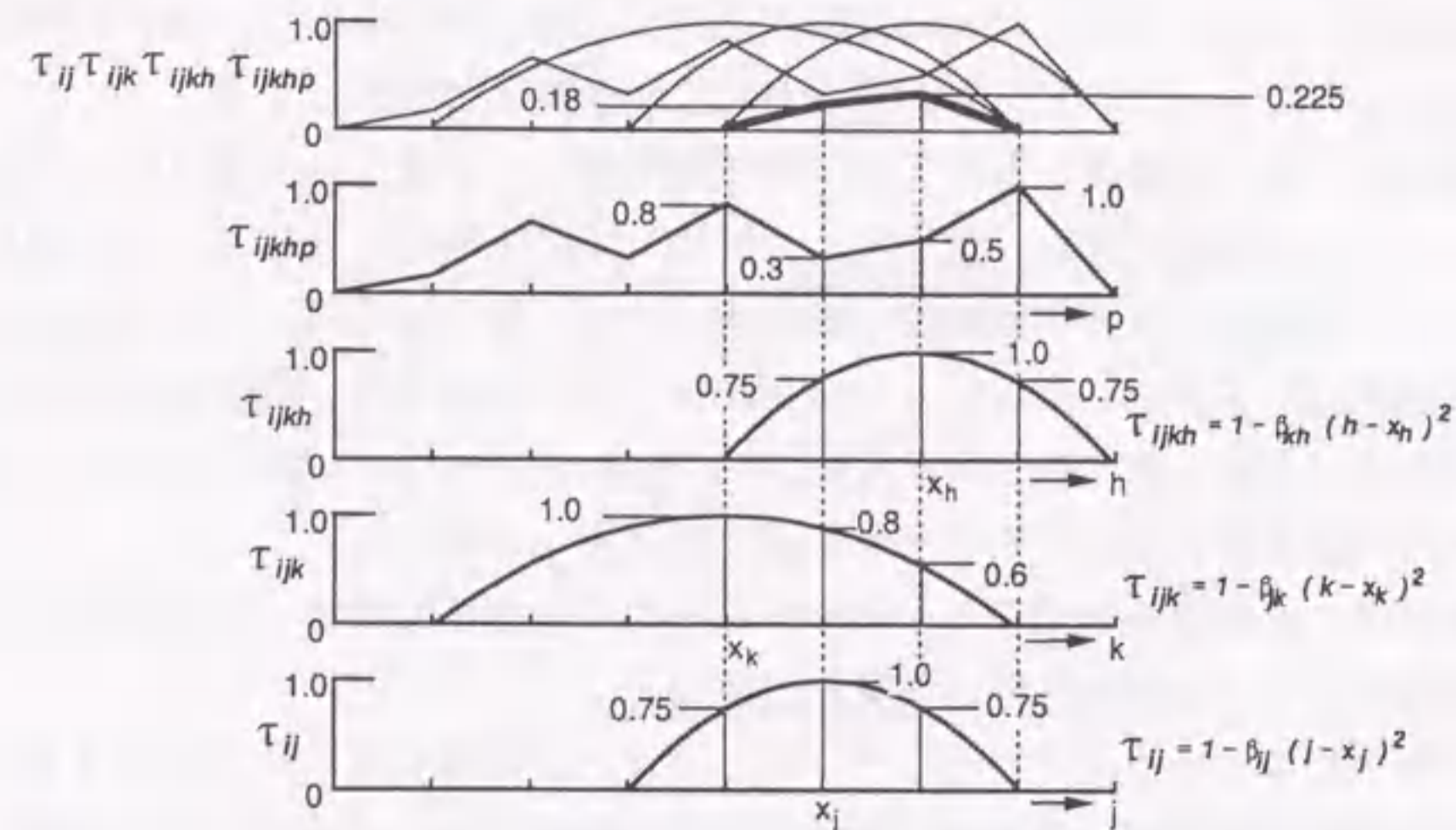


図3.4 各層のニューロンの結合の様子

3.2.3 高速学習と追加学習性の実験

前節で提案した量子化ニューロンを用いた機能別階層ネットワーク(MFLN)を、マルチフォント英数字認識(237フォント、17フォント当り62カテゴリ)に適用し、その高速学習と追加学習性の有効性について、以下の観点から従来のニューラルネットワークモデルと比較実験した。

パターン認識の学習問題においては、予め与えられた学習データが識別すべきパターン空間の一部しか表現しない場合が少なくない。例えばマルチフォント文字認識では、予め学習できるフォントには限りがある。それゆえ、そのシステムを使用していく中で、認識性能を確保するには、新しいフォントを再度学習する必要が生ずることが十分考えられる。このとき、既に学習したデータも追加データとまとめて学習し直す必要があるシステムは望ましくない。また、システムへの組み込みを考えた場合、現場学習、すなわちシステムを使用しながら学習が行える機能が必要となり、そのためには学習の高速性も要求される。

そこで、追加学習性の評価項目として以下の2点を用いることとした。

- ・追加パターンのみ学習したときに、学習済みのパターンに対する性能を如何に保持できるか
- ・追加パターンのみ学習したときに、追加パターンを如何に速く学習できるか

つまり、既に学習済みの状態から、未学習データを与えて、このデータを100%認識できるまで学習を行う。その学習に要した時間がどれほど短い、そして、以前の学習データを認識させたときに、どれほど正しく答えられるかを評価の尺度とする。

検討条件は、64×63画素からなる文字データに対して、文字部を切り出した後、8×8=64次元のメッシュ特徴を求め、量子化ニューロンの第1層にはこのメッシュ濃度を、第2層にはメッシュ濃度の横方向微分値(縦線成分)を、第3層にはメッシュ濃度の縦方向微分値(横線成分)を入力した。ニューロン数は、第1層目を64、最終層を62とした。各層での量子化レベルを、第1層目はN1=10、第2層目はN2=5、第3層目はN3=5とした。この量子化レベルの設定は、予め入力データの統計的性質より求めた。

検討項目は、認識性能が望ましい値になるまでの初期学習時間、追加学習を行なった場合の初期学習データの認識性能と学習時間である。

比較対照したニューラルネットワークは、パターン認識に適したニューラルネットワークの代表的なものである、BP法による学習を行う階層型ニューラルネットワーク(BPNN)、学習ベクトル量子化(LVQ)ネットワーク及びRCEネットワークである。学習ベクトル量子化ネットワークとRCEネットワークは、最近傍ニューロンの選択時あるいは中間層ニューロンの追加時に、入力パターンベクトルとすべての参照ベクトルとの距離計算に時間を要する傾向がある。比較対照したニューラルネットワークの構造は、それぞれ最適化を行った結果次のようにした。BPNNは、中間層ニューロンを42個、入力層ニューロンを64個、出力層ニューロンは62個に設定した。LVQは、代表ベクトルを表現するニューロン数(隠れ層ニューロンの数)を248個で、その学習係数を0.1と設定し、出力層ニューロン数は62個と設定した。RCEは初期学習の隠れ層ニューロンを382個、出力層ニューロン数は当然62個と設定した。LVQ、RCE共に、その隠れ層ニューロンの入力ベクトルの次元数は64次元である。実験結果を次に示す。

(1) 初期学習時の学習速度

初期学習データ(137フォント)の学習回数に対する、認識率の変化が図3.5である(但し、ネットワークの最適化を行わずとも、学習速度が十分速いことを確認するために、この場合のみ、量子化ニューロンを用いた機能別階層ネットワーク(MFLN)の各層の量子化レベルはN1=8、N2=8、N3=8とした)。○は学習データの認識率変化、□は未学習データの認識率の変化を示す。学習速度の比較結果は表3.1である。量子化ニューロンを用いた機能別階層ネットワーク(MFLN)はBP学習の階層型ニューラルネットワークに対し40倍以上速く学習が終了する。

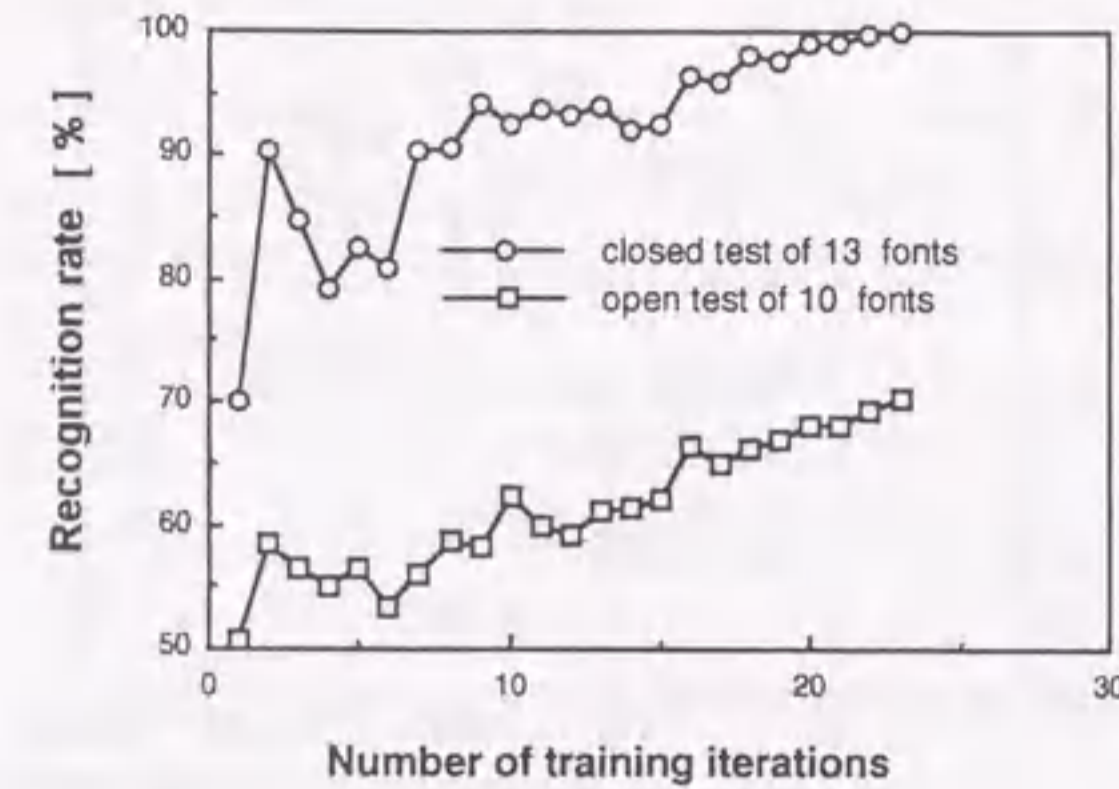


図3.5 初期学習データ(137フォント)の学習回数に対する認識率の変化

表3.1 学習速度比較

(時間は学習データの認識率が100%になるまでの時間)

モデル	速度	繰り返し回数	時間(分)
MFLN		23	4.5
BPNN		200	192
		250分後でも100%未達	
RCE		5	17
LVQ		20	170

(2) 追加学習特性

初期学習したネットワークに対して、追加学習(1、3、5、7フォントの追加)を行った場合の初期学習フォントの認識率を縦軸に、追加フォントの学習時間を横軸に取り、追加フォント数毎の追加学習性能マップを示したものが図3.6である。この図の左上部に位置するほど、追加学習時間が短く、初期学習フォントの認識性能への影響が少ないので、追加学習性が優れていることになる。量子化ニューロンを用いた機能別階層ネットワークMFLNは総合的に追加学習性が他のモデルより良いことが分り、その構成の有効性が示されている。

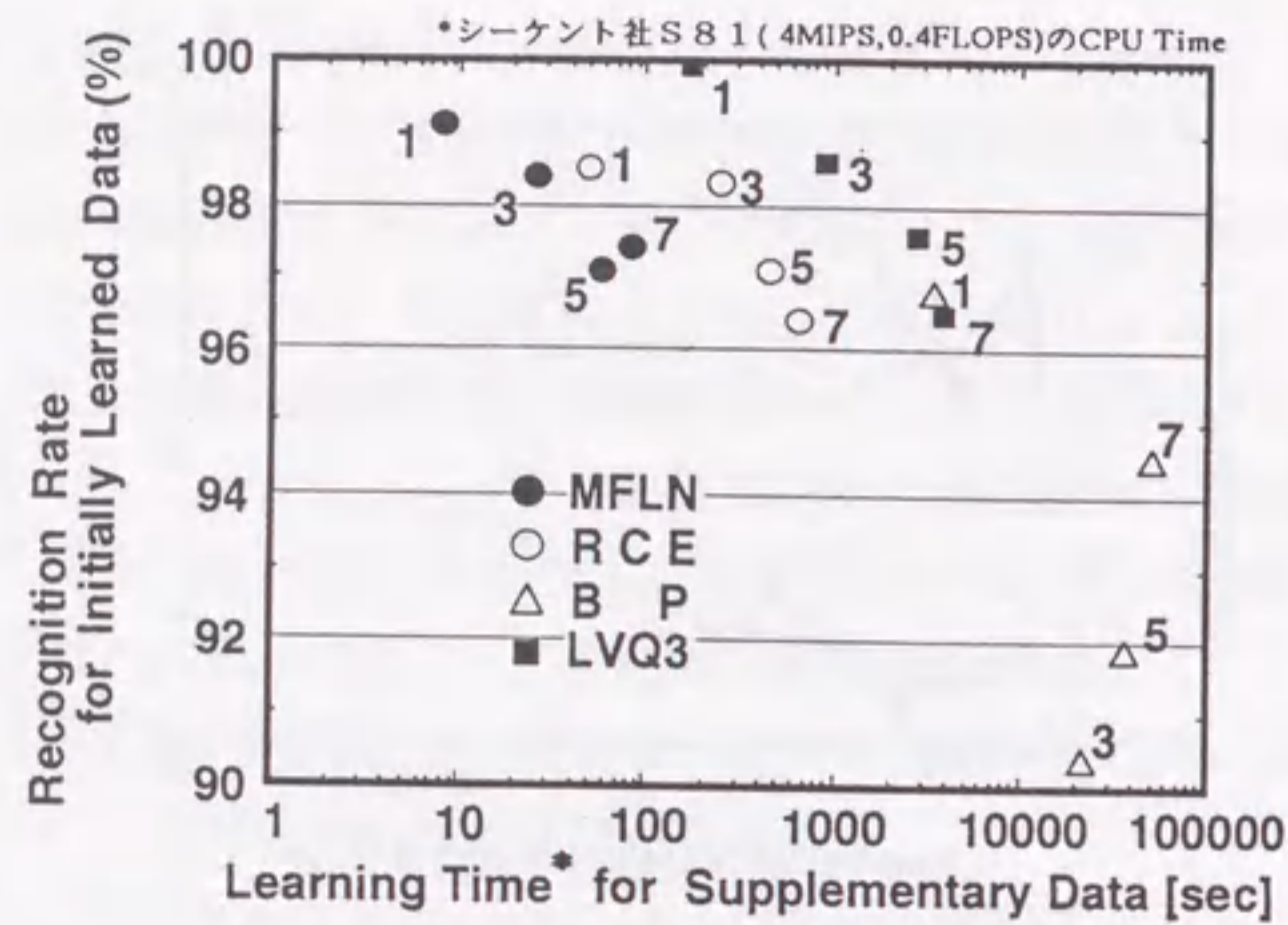


図3.6 追加学習性の比較

3.3 適応増殖量子化ニューロンを用いたニューラルネットワーク

前節で、量子化ニューロンを用いた、入力データの量子化と学習による機能獲得を持つ機能別階層ネットワークを提案した。そして、これをマルチフォント文字認識に適用し、高速学習性及び良好な追加学習性を実証した。この量子化ニューロンを用いた機能別階層ネットワークの認識性能は、ネットワークの構造に依存する。特に、量子化ニューロンの量子化レベルが量子化すべき入力データの空間分布をどの程度反映しているかが、その認識性能を左右する。このネットワーク構造の有効な決定方法が確立されていないために、入力データの大きな分布状態を調べ、ネットワーク構造を決定するという試行錯誤的な方法を用いざるを得なかった。

本節では、これを解決するために、入力データに応じて適応的に増殖し、ネットワークの自己形成が可能な適応増殖量子化ニューロン (ASQA: Adaptive Segmentation of Quantizer neuron Architecture) を用いたニューラルネットワークを提案し、その高速学習と追加学習性について明らかにする。

3.3.1 適応増殖量子化ニューロン

(1) 適応増殖量子化ニューロンの構造

適応増殖量子化ニューロンは、前節で説明した量子化ニューロンをベクトル次元での量子化を行うように拡張したものである。各次元毎に上限値 X_{Uim} と下限値 X_{Lim} (i はユニット番号、 m は次元を表す) で量子化範囲が設定されたユニット U_i によって量子化器を構成する。各ユニット U_i は、量子化範囲内に入力された量子化入力データの各次元毎の平均 A_{im} 、分散 σ_{im} 、学習回数 n_{im} 、及び内部状態 E_{im} を有しており、それぞれのベクトル表現を \vec{A}_i 、 $\vec{\sigma}_i$ 、 \vec{n}_i 、 \vec{E}_i とする。

入力データ \vec{x} が各ユニットに入力され、式(3.5)に示すように、 \vec{x} がユニット U_q の量子化範囲に入っておればユニット U_q が反応し、ユニット U_q に連結した結合係数 τ_q を 1 に設定する。量子化範囲がユニット U_q の量子化範囲に隣あうすべてのユニット U_i に連結した結合係数 τ_i を、ユニット U_i との距離 $|x_m - A_{im}|$ に応じて、式(3.6)に示すように、係数 β を用いて設定する。入力データ \vec{x} とユニット U_i との距離が近いほど結合係数 τ_i が大きく設定される訳である。

$$\tau_q = 1 \quad \text{when } X_{Lqm} \leq x_m \leq X_{Uqm} \quad \text{for all } m \quad (3.5)$$

$$\tau_i = \beta \cdot \left(1 - \frac{|x_m - A_{im}|}{|A_{qm} - A_{im}|} \right) \cdot \tau_q \quad (3.6)$$

when $\vec{X}_{Li} \cdot \vec{X}_{Uq} = 1$ or $\vec{X}_{Ui} \cdot \vec{X}_{Lq} = 1$

$$\vec{X} = \{x_1, x_2, \dots, x_M\}$$

$$\vec{A}_i = \{A_{i1}, A_{i2}, \dots, A_{iM}\}$$

ここで、 \vec{X}_{Li} はユニット U_i の量子化範囲の下限のベクトル表現、 \vec{X}_{Ui} は量子化範囲の上限のベクトル表現である。

図3.7に2次元ベクトル量子化ニューロンの結合係数設定の様子を示す。 τ_8 が1に設定され、また隣あうユニット U_2 、 U_7 、 U_9 、 U_{12} の結合係数 τ_2 、 τ_7 、 τ_9 、 τ_{12} が式(3.6)に基づいて設定される。

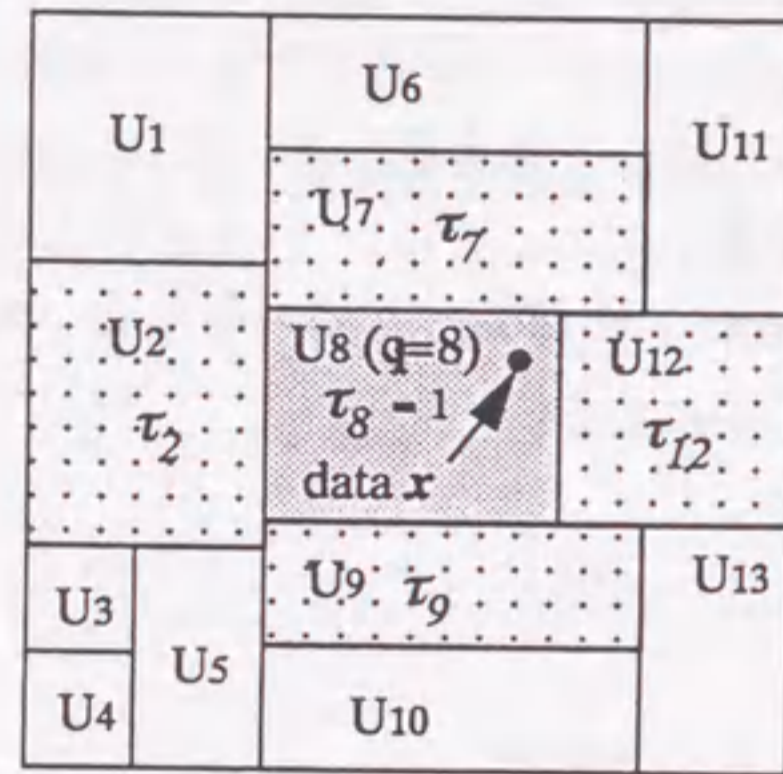


図3.7 ベクトル量子化ニューロンの結合係数の設定

(2) 適応増殖量子化ニューロンの増殖

適応増殖量子化ニューロンの増殖は、量子化ユニットの分裂と、連結されている他の適応増殖量子化ニューロンの複製によって行われる。初期状態の適応増殖量子化ニューロンの量子化ユニット数は1に設定してある。適応増殖量子化ニューロンに何らかの学習指示信号が与えられると、入力データ \vec{x} に反応した量子化ユニット U_i の一つの次元 m が選択され、その次元における過去の平均 A_{imo} 、分散 σ_{imo} 、学習回数 n_{imo} 、及び入力データ x_m から、量子化入力端子からの全入力データの平均 A_{im} 、分散 σ_{im} 、学習回数 n_{im} 、及び内部状態 E_{im} が近似式(3.7)~式(3.10)に基づいて計算される。

$$A_{im} \doteq \frac{n_{imo} \cdot A_{imo} + x_m}{n_{imo} + 1} \quad (3.7)$$

$$\sigma_{im} \doteq \frac{n_{imo} \cdot \sigma_{imo} + |x_m - A_{im}|}{n_{imo} + 1} \quad (3.8)$$

$$n_{im} = n_{imo} + 1 \quad (3.9)$$

$$E_{im} = n_{im} \cdot \sigma_{im} \quad (3.10)$$

この時、内部状態が式(3.11)で示す分裂条件を満たした場合に、量子化ユニットの分裂及び適応増殖量子化ニューロンの複製を行い、適応増殖量子化ニューロンを増殖させる。

$$E_{im} > \alpha \quad (\alpha : \text{定数}) \quad (3.11)$$

(a) 量子化ユニットの分裂

量子化ユニット U_i における次元 m の内部状態を表す値 E_{im} が、しきい値 α を超えた場合、図3.8に示すように、量子化ユニット U_i を U_i' と U_{i+1} の二つに分裂させる。この時、量子化範囲を平均 A_{im} を境界として二つに分割し、分裂後の量子化ユニット U_i' と U_{i+1} の量子化範囲を式(3.12)~式(3.15)を用いて求める。

$$X_{Li m'} = X_{Lim} \quad (3.12)$$

$$X_{Ui m'} = A_{im} \quad (3.13)$$

$$X_{Li+1 m} = A_{im} \quad (3.14)$$

$$X_{Ui+1 m} = X_{Uim} \quad (3.15)$$

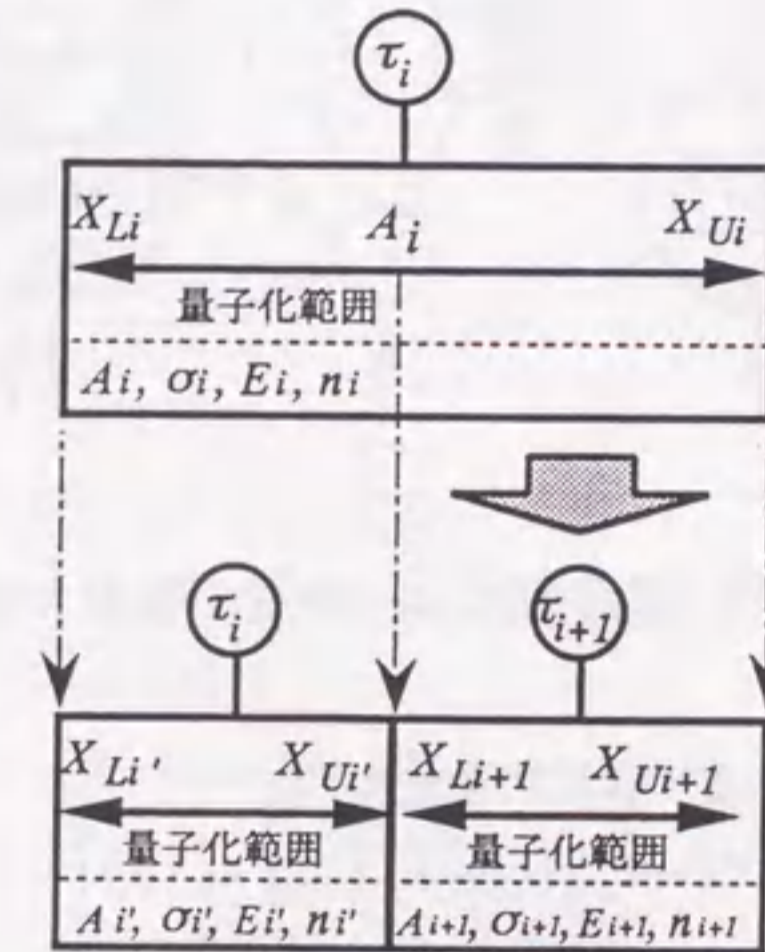


図3.8 量子化ユニットの分裂

さらに平均 A 、分散 σ 、学習回数 n を式(3.16)~式(3.19)を用いて求め、内部エネルギー E を式(3.10)を用いて計算する。量子化ユニット U_{i+1} の結合係数 τ_{i+1} は、式(3.20)に示すように量子化ユニット U_i の結合係数 τ_i をそのまま用いる。

$$A_{im'} = \frac{X_{Li m'} + X_{Ui m'}}{2} \quad (3.16)$$

$$A_{i+1 m} = \frac{X_{Li+1 m} + X_{Ui+1 m}}{2} \quad (3.17)$$

$$\sigma_{im'} = \sigma_{i+1m} = 1 \quad (3.18)$$

$$n_{im} = n_{i+1m} = 1 \quad (3.19)$$

$$\tau_{i+1} = \tau_i \quad (3.20)$$

(b) 量子化ニューロンの複製と増殖

量子化ユニットの分裂を行わせた後、図3.9に示すように、当該ニューロンQN₁に連結したニューロンQN₂の複製QN₂'を作成し、量子化ユニットU_{i+1}の結合係数τ_{i+1}に連結する。以上の処理により、入力データに応じて適応的に量子化ニューロンを増殖させ、ネットワークの自己形成を行わせることができる。

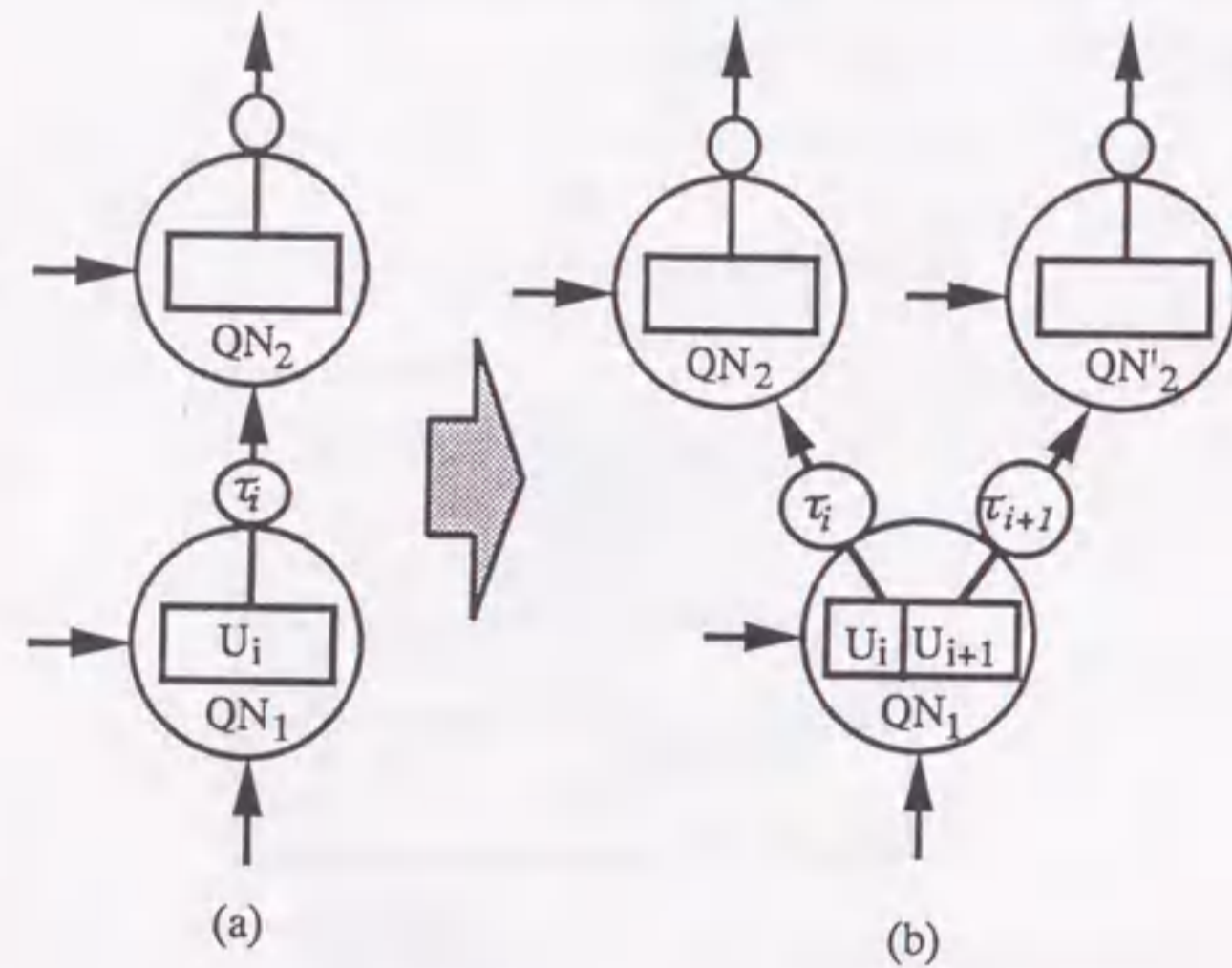


図3.9 量子化ニューロンの複製と増殖

3.3.2 適応増殖量子化ニューロンを用いた 機能別階層ネットワーク (MFLN-II)

適応増殖量子化ニューロンを用いれば、ネットワークを次に示すように自動的に構成することが可能となる。図3.10は、2次元の入力を二つの領域(A, B)に分離する機能別階層ネットワークの構成例とネットワークの自己生成の様子を示したものである。

入力層を1個の2次元ベクトルの適応増殖量子化ニューロンによって構成し、出力層に2個の通常のニューロンを配置する。出力層のニューロンpは式(3.21)に示すように、前層のニューロンから送られた信号t・τ_{ij}と学習によって形成された結合の重みτ_{ijp}との積和を取り、しきい値関数f()に基づいて発火出力y_pを出力する。但し、分裂により増殖した中間層のニューロンの現在の数をmaxとおく。

$$y_p = f\left(\sum_{i=1}^N \max_{j=1} \tau_{ij} \cdot t \cdot \tau_{ijp}\right) \quad (3.21)$$

ネットワークの学習(結合の重み変更)は、最終層のみで行う。最終層の出力結果が誤っている場合に、選択された量子化ユニットと教師信号により決定される最終層の発火すべきニューロンとの結合の重みτ_{ijp}をt・τ_{iq}だけ強化する。係数βが零でない場合は、選択された量子化ユニットに隣接するユニットと最終層ニューロン間の結合の重みについても同様に強化する。そして、選択された量子化ユニットqの平均、分散、学習回数及び内部状態を近似式(3.7)~(3.10)に基づいて更新する。ここで内部状態E_iが式(3.11)に示す分裂条件を満たせば、適応増殖量子化ニューロンの量子化器の量子化ユニットを分裂させ、中間層のユニットを増殖させる。

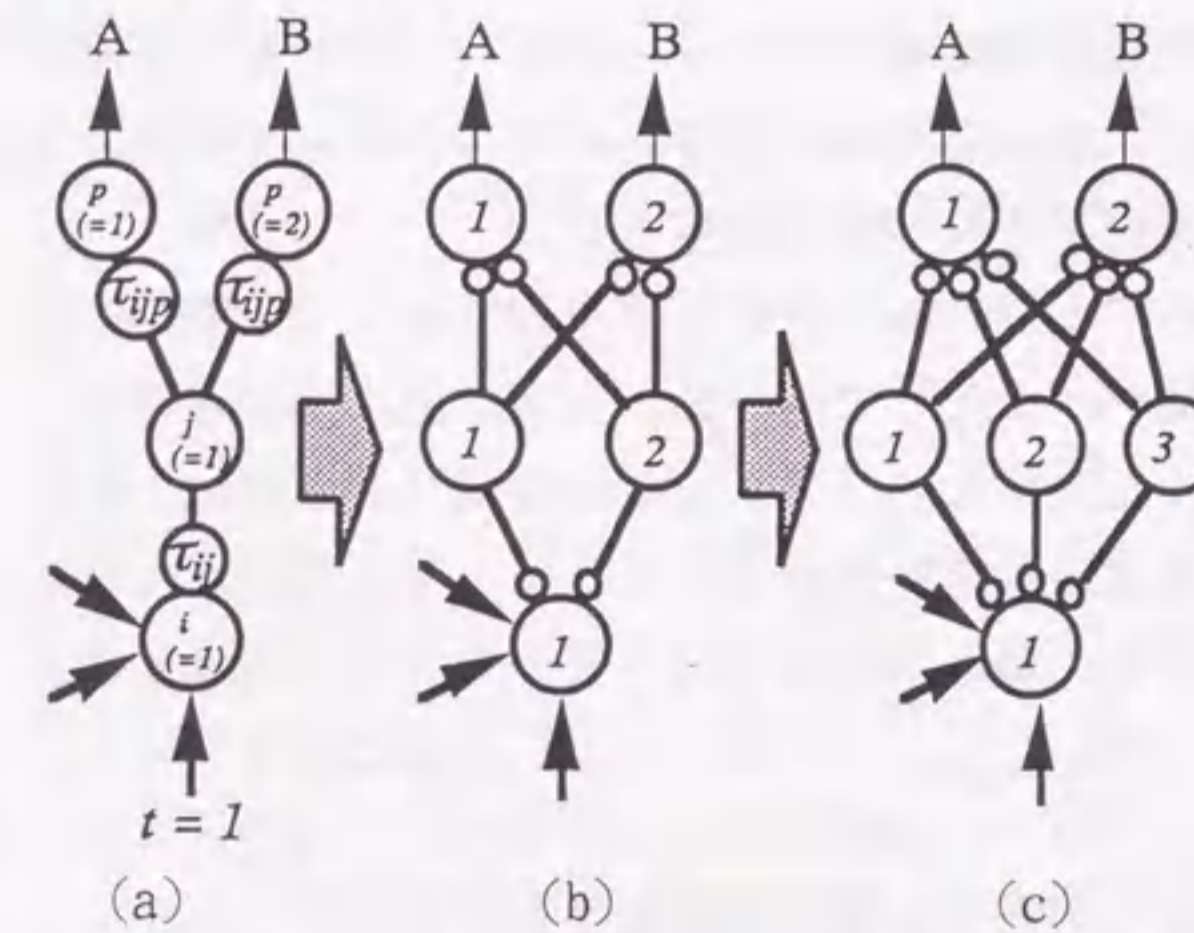


図3.10 適応増殖量子化ニューロンによる機能別階層ネットワークの構成例とネットワークの自己形成

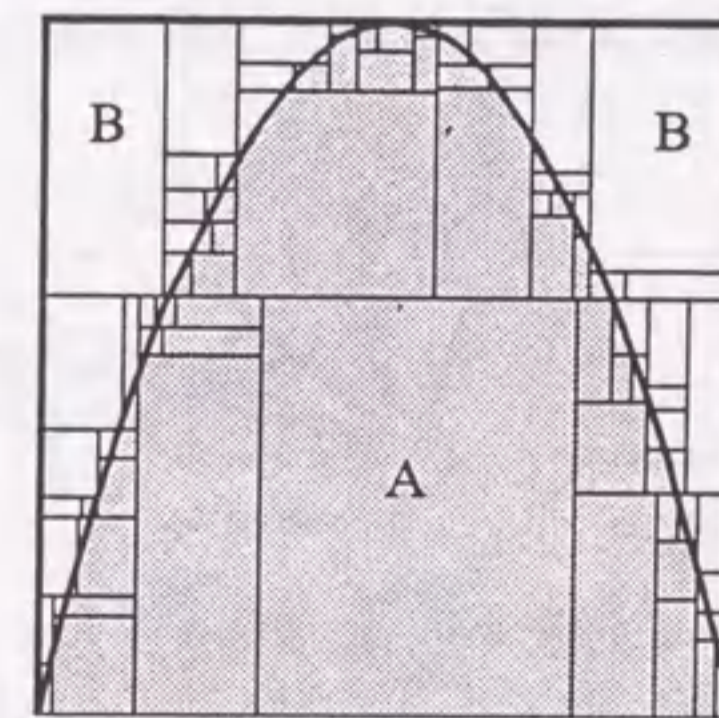


図3.11 量子化ユニットの分裂の様子

このようにして、教師信号と発火出力y_pが一致するまで学習を行う。パターン認識に用

いる場合には、出力が最大であるニューロンの番号を検出し、それを認識結果とするので、教師信号で示されるニューロンが最大発火出力となれば学習ができたことになる。

初期状態では、図3.10(a)に示すように中間層のユニット数は1であるが、データの入力に適應して中間層のユニットが図3.10(b)(c)に示すように増殖し、ネットワークの自己生成が行われる。図3.11は、2次元領域の二次曲線による分離をネットワークに学習させた場合の、量子化ユニットの分裂の様子を量子化範囲の分割として表したものである。図に示すように、非線形領域ほど細かく分割され、量子化ユニットが分裂していく。

3.3.3 初期学習の高速性の実験

これまで説明した適應増殖量子化ニューロン (ASQA) を用いた機能別階層ネットワーク (MFLN-II) を、実用的課題であるマルチフォント英数字認識に適用し、学習認識実験を行いその高速学習性の評価を行った。

英数字62字種からなる23フォントの文字中13フォントを学習に用い、残り10フォントを未学習認識率の評価データとして用いた。文字認識に用いる特徴データとして、2値イメージとして取り込んだ文字の8×8のメッシュ特徴 (濃度値)、メッシュ濃度の横方向微分値、及びメッシュ濃度の縦方向微分値を8ビットで正規化した3次元ベクトルを用いた。この3次元ベクトルを、適應増殖量子化ニューロンの入力とした。この条件は3.2節の量子化ニューロンを用いたニューラルネットワークと同じものである。

実験に用いたネットワーク構成を図3.12に示す。8×8のメッシュ数に対応させて、64個の3次元ベクトルを入力とする適應増殖量子化ニューロンによって入力層を構成し、出力層に62個の通常のニューロンを配置し、中間層に教師信号を入力するようにした。中間層のニューロンは、初期状態では64個であるが、学習によって適應増殖量子化ニューロンの量子化ユニットが分裂し、設定した最大個数 J_{max} まで増加して行く。

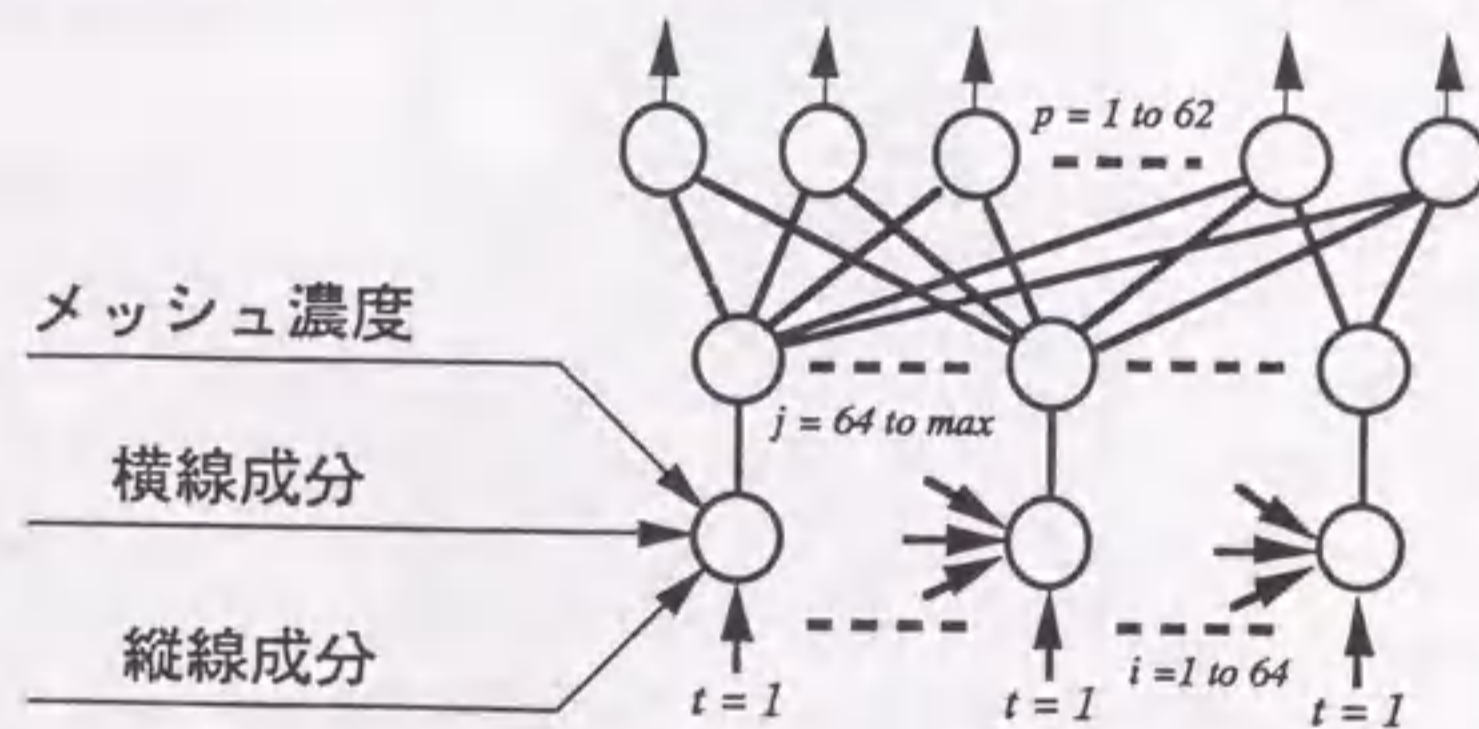


図3.12 英数字認識ネットワークの構成

ネットワーク形成のパラメータである分裂条件と中間層ニューロン数、学習速度、未学習フォント認識率、及び係数 β と未学習フォント認識率の関係を調べ、その結果に基づき他のネットワークとの比較を行った。以下、順にその結果と考察を示す。

(1) 分裂条件と中間層ニューロン数

ニューロンの分裂変化を調べるので、ニューロンの選択信号は一ヶ所にしか出力されないように、選択信号の分布を決める係数 β を零に設定した。各メッシュ毎の中間層ニューロン数の最大値 J_{max} を変化させて、学習フォント認識率が100%に達した時点での中間層ニューロン総数と分裂条件との関係を調べた。図3.13がその結果である。

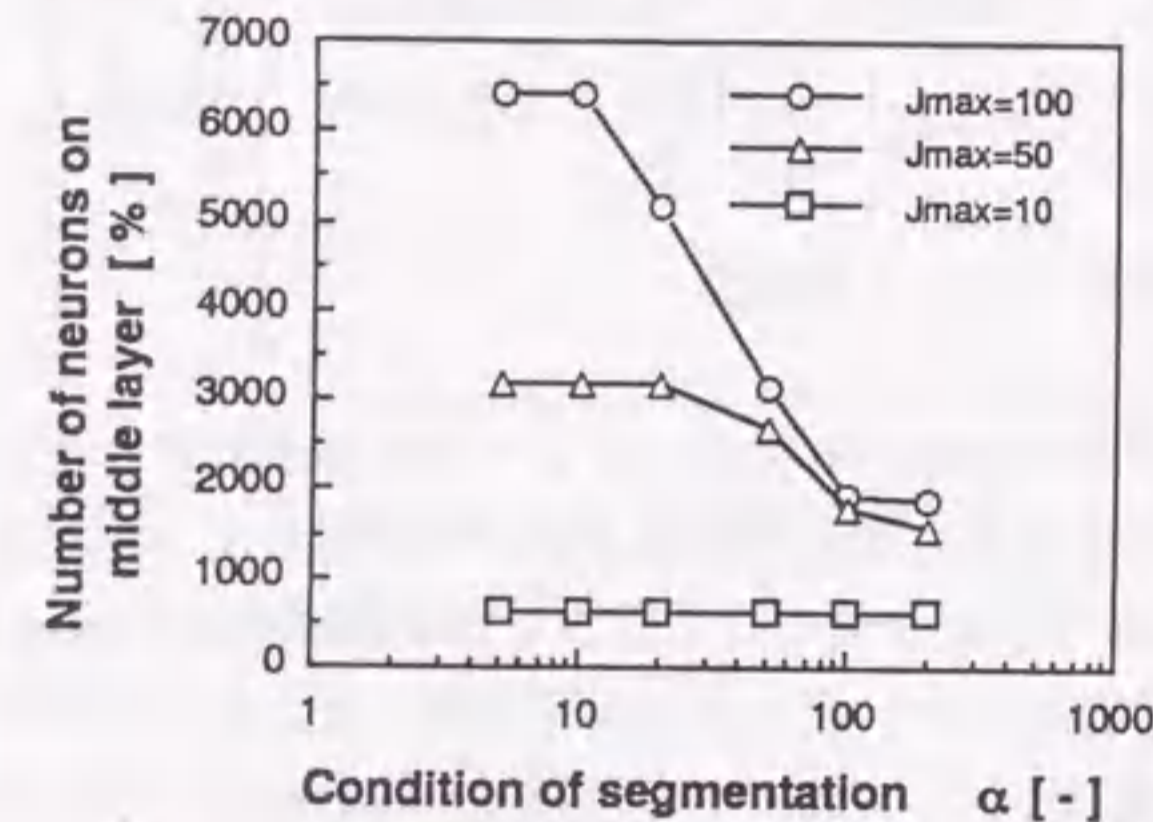


図3.13 分裂条件と中間層ニューロン数

分裂のしきい値 α を大きくし分裂条件を厳しくすると、中間層総ニューロン数が減少しネットワーク規模が小さくなる。いずれの J_{max} 値においても α を100程度に設定すれば、ネットワーク規模の十分な縮小が可能である。

(2) 分裂条件と学習速度

ニューロンの分裂の条件を調べるため、ニューロンの選択信号は一ヶ所にしか出力されないように、選択信号の分布を決める係数 β を零に設定した。各メッシュ毎の中間層ニューロン数の最大値 J_{max} を変化させて、学習フォント認識率が100%に達するまでに要する時間と分裂のしきい値 α との関係を調べた。図3.14がその結果である。

分裂のしきい値 α を高くすれば、最終層の結合の重みの変更で学習を行うことになるので、学習に時間がかかる。このことは、 α が100を超えると、学習に要する時間が急激に増

加することから分かる。

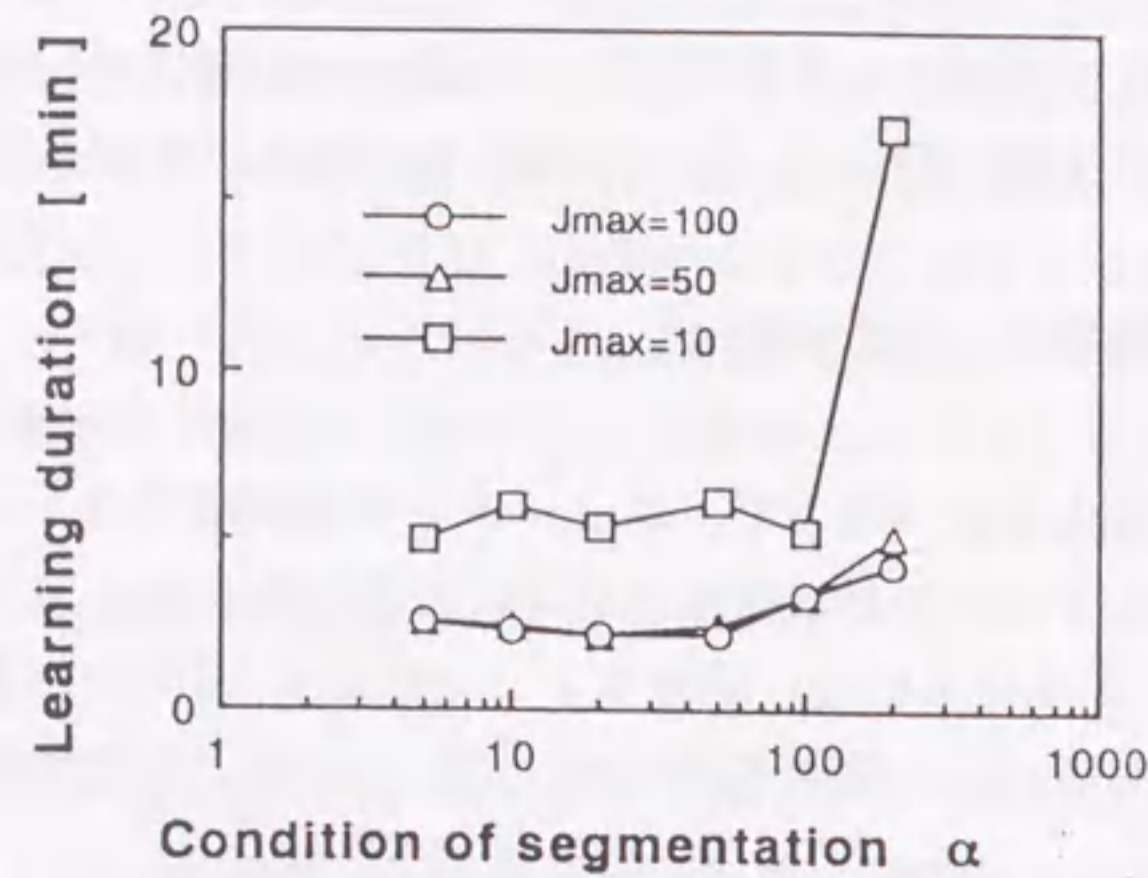


図3.14 分裂条件と学習に要する時間

(3) 分裂条件と未学習フォント認識率

ニューロンの分裂の条件を調べるため、ニューロンの選択信号は一ヶ所にしか出力されないように、選択信号の分布を決める係数 β を零に設定した。各メッシュ毎の中間層ニューロン数の最大値 J_{max} を変化させて、学習フォント認識率が100%に達した時点での未学習フォント認識率と分裂のしきい値 α との関係を調べた。図3.15がその結果である。

未学習フォント認識率に関しては、 α の最適値が存在することを示しており、 α を概ね10~100程度に設定することが望ましいことが分る。

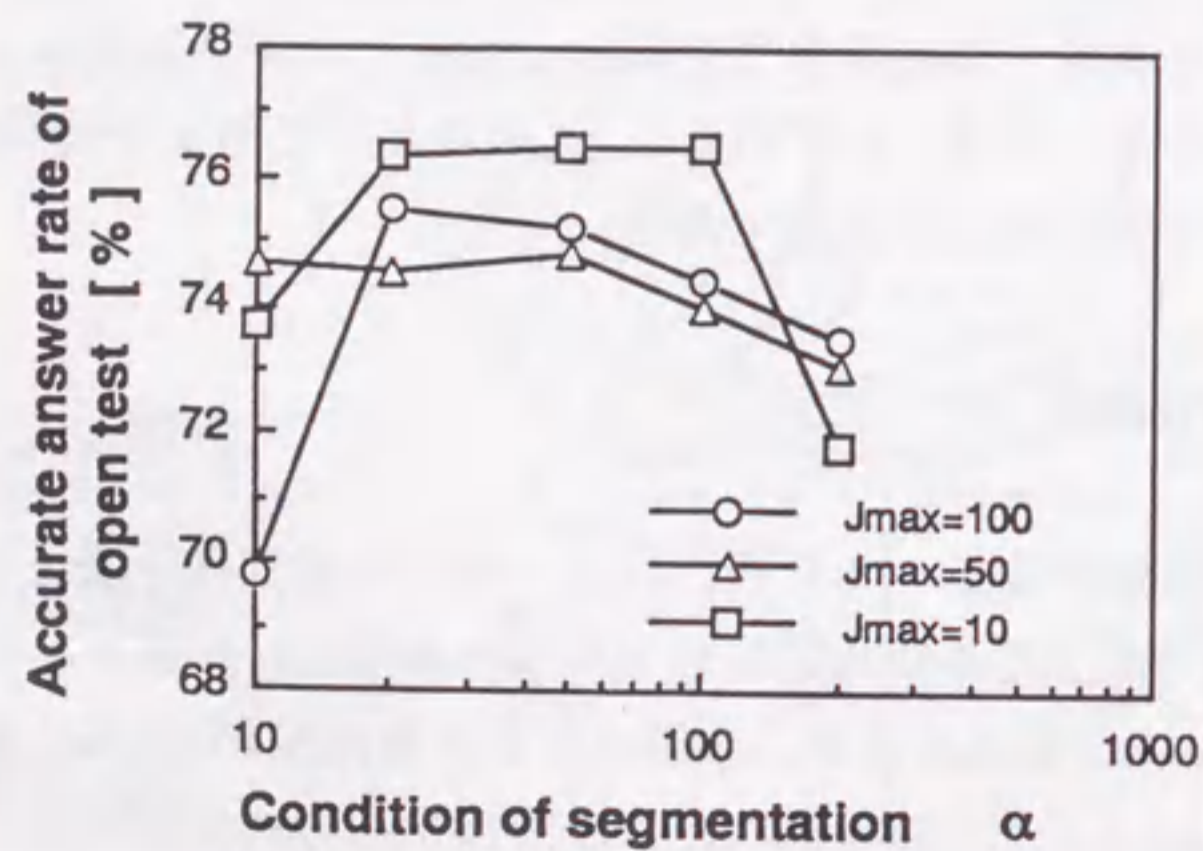


図3.15 分裂条件と未学習フォント認識率

(4) 結合係数 β と未学習フォント認識率

これまでの結果に基づき、各メッシュ毎の中間ニューロン総数の最大値 J_{max} を100に設定し、分裂のしきい値 α が20と100の場合について、係数 β と未学習フォント認識率との関係を調べた。図3.16がその結果である。

係数 β を適切に設定することにより未学習フォント認識率を向上させることができる。概ね $\beta = 1$ に設定すれば比較的良好な未学習認識率を得ることができる。

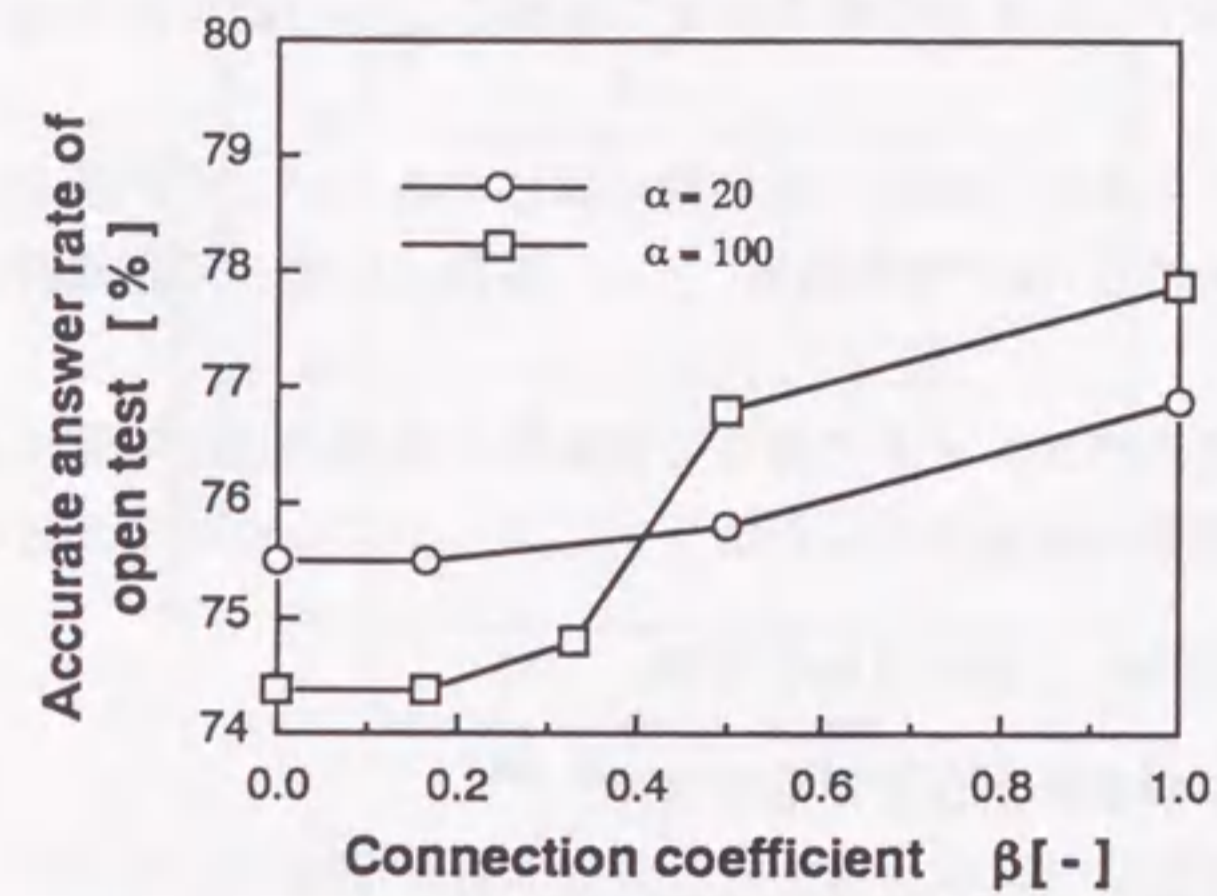


図3.16 結合係数 β と未学習フォント認識率

(5) 他のネットワークモデルとの比較

MFLN-IIの学習速度を他のネットワークモデルと比較した。他のネットワークモデルは、BP学習の階層型ニューラルネットワーク(BPNN)とRCEネットワークとした。MFLN-IIは、中間層ニューロン数の最大値 J_{max} を100、分裂のしきい値 α を20、係数 β を0.16に設定した。表3.2がその結果である。

表3.2 学習速度比較

モデル	学習速度[分]	
	学習フォント認識率 99%時点	学習フォント認識率 100%時点
MFLN-II	8	11
BPNN	200	250分でも100%未達
RCE	20	24

MFLN-IIの学習速度はBPNNの約25倍以上となり、高速学習が可能であることが示されており、適応増殖量子化ニューロンの高速学習の有効性が確認できる。

3.3.4 追加学習の実験

ここではマルチフォント英数字認識に適用し、初期学習フォントに対する、追加学習後の認識性能と、追加フォントの学習に要する時間の二つの観点から追加学習性の検討を行った。

前節の初期学習の高速性の実験と基本的に同じであるが、中間層ニューロン数の最大値 J_{max} を100、内部状態 \vec{E}_i は学習回数 \vec{n}_i とし、係数 β に関して認識時は零に固定し、学習時のみ変化させた。

ネットワーク形成のパラメータである分裂条件と係数の追加学習に与える影響を調べるために、以下の3種類の実験を行った後、他のネットワークモデルとの比較を行った。

- ・初期学習時の係数 β による追加学習性
- ・初期学習時の分裂条件による追加学習性
- ・追加学習時の分裂条件による追加学習性

以下にその結果と考察を示す。

(1) 初期学習時の係数 β による追加学習性

初期学習時の係数 β が追加学習性能に与える影響を調べるために、初期学習時に係数 β を0に設定し、選択されたユニット q の結合係数 τ_q のみを初期学習に用いた場合と、 $\beta = 0.15$ に設定し選択されたユニット q に隣接するユニットの結合係数も初期学習に用いた場合の追加学習性について検討した。

図3.17は、分裂のしきい値 α を20、係数 β を0及び0.15に設定して、初期学習したネットワークに対して、追加学習(1, 3, 5, 7フォント)を行った結果である初期学習フォントの認識率を縦軸に、追加学習フォントの学習時間を横軸に取り、追加フォント数毎の追加学習性能を表したものである。図の左上部に位置する点ほど、追加学習性が優れていることを表す。初期学習フォントの認識率及び追加フォントの学習時間は、追加学習時の分裂条件を6通り ($\alpha = 5, 10, 20, 30, 40, 50$) に変化させた場合の平均値とした。

図に示すように、初期学習時に係数 β を0.15に設定し、選択されたユニット q に隣接す

るユニットの結合係数も初期学習時に利用した方が、初期学習フォントの認識率の劣化が少なく、学習時間も速くなることが分かる。これは、選択されたユニット q に関する最終層の結合の重みだけでなく、隣接するユニットに接続される最終層の結合の重みも変更されるので、初期学習時に形成されるカテゴリ間の識別境界が滑らかになり、追加学習時に極端な結合の重み変更の必要がなかったためと考えられる。

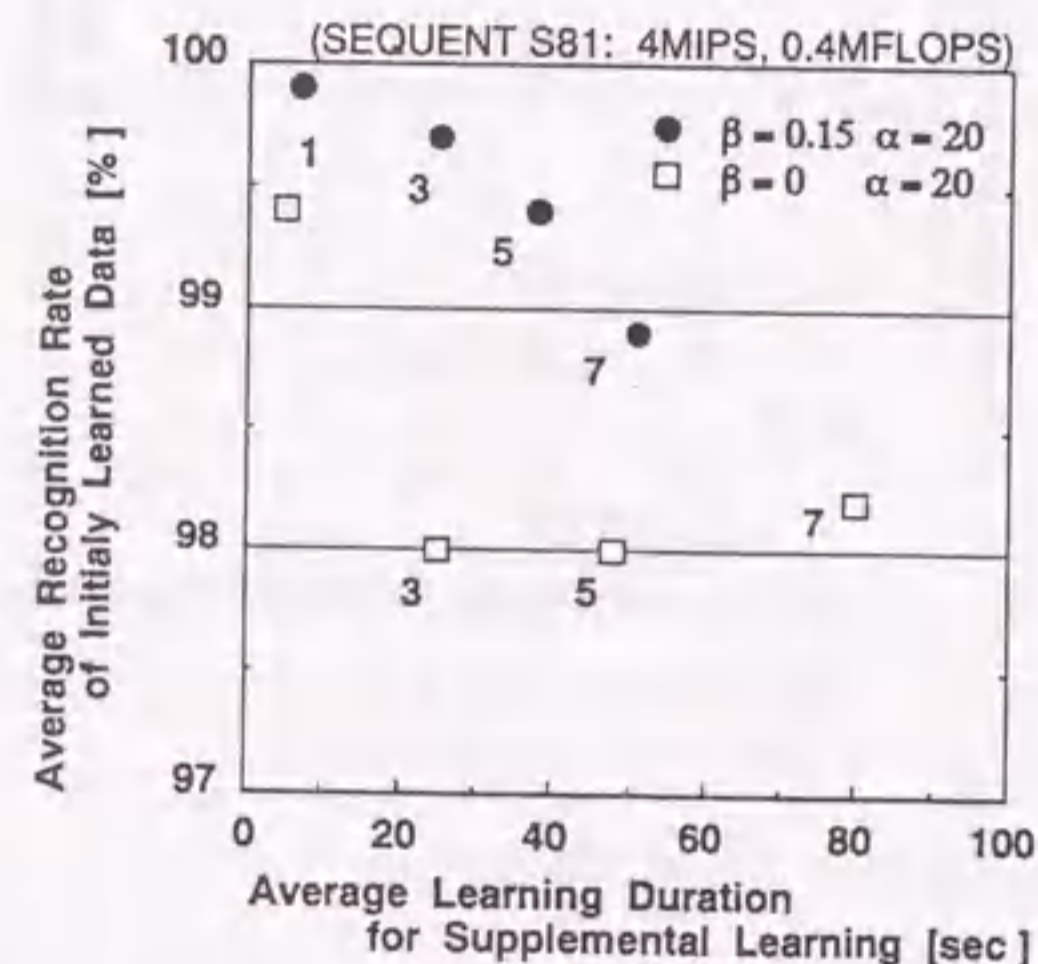


図3.17 初期学習時の結合係数 β による追加学習性

(2) 初期学習時の分裂条件による追加学習性

初期学習時の分裂のしきい値 α が追加学習性能に与える影響を調べるために、初期学習時に分裂のしきい値 α を小さく設定し、ユニットの分裂を中心に初期学習した場合と、分裂のしきい値 α を大きめに設定し、ユニットの分裂を抑制することにより最終層での重み変更を中心に初期学習した場合との追加学習性について検討した。

図3.18は、係数 β を0.15、分裂にしきい値 α を20及び100に設定して初期学習したネットワークに対して、追加学習(1, 3, 5, 7フォント)を行った結果を示す。初期学習フォントの認識率を縦軸に、追加フォントの学習時間を横軸に取り、追加フォント数毎の追加学習性能を表したものである。図の左上部に位置するほど追加学習性能に優れている。初期学習フォントの認識率及び追加フォントの学習時間は、追加学習時の分裂のしきい値を6通り ($\alpha = 5, 10, 20, 30, 40, 50$) に変化させた場合の平均値とした。

初期学習の分裂のしきい値を変化させても、追加フォントの学習時間に大差はないが、初期学習フォントに対する認識性能は、初期学習時に分裂のしきい値 α を小さめに設定した方が良好な認識率を示している。これは、初期学習においては、最終層での結合の重み変更よりも中間層でのユニットの分裂を中心に学習し、ある程度ネットワーク規模を大き

く取っておいた方が追加学習に有利であることを示している。

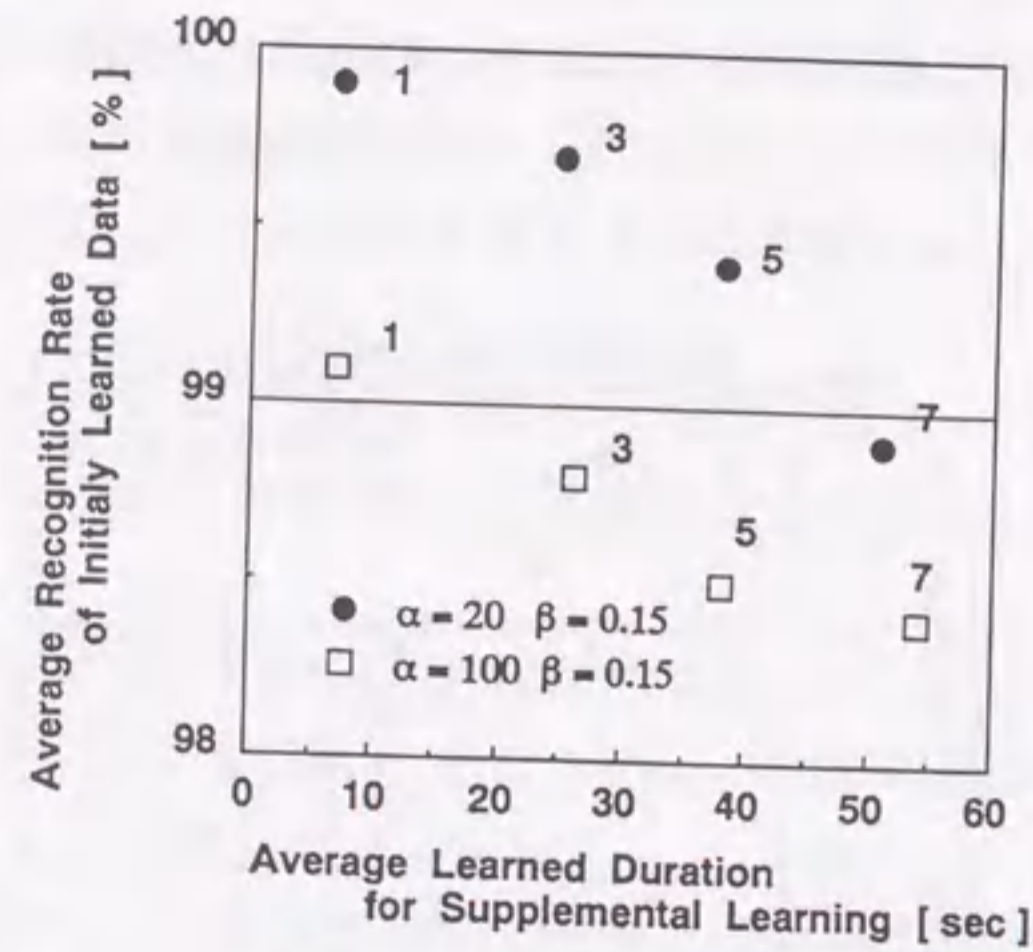


図3.18 初期学習時の分裂条件 α による追加学習性

(3) 追加学習時の分裂条件による追加学習性

これまでの実験で明らかにされた初期学習法に基づいて学習したネットワークに対して、分裂のしきい値 α を小さく設定しユニットの分裂を中心に追加学習した場合と、分裂のしきい値 α を大きめに設定しユニットの分裂を抑制することにより最終層での結合の重み変更を中心に追加学習した場合の、追加学習性に与える影響について検討した。

図3.19に、追加フォント数が1の場合の追加学習時の分裂のしきい値 α と中間層総ニューロン数との関係を示す。 α を大きくし分裂条件を厳しくすると、中間層ニューロン数は増加せず、最終層での結合の重み変更のみで追加学習が行われている。逆に、 α を小さくすると、学習が進むにつれて中間層ニューロンが増加し、ユニットの分裂を中心に学習が行われていることが分かる。

図3.20は、分裂のしきい値 α を20、係数 β を0.15として初期学習を行ったネットワークに対して、係数 β を零、分裂のしきい値 α を5及び20に設定して追加学習を行った結果を示す。初期学習フォントの認識率を縦軸に、追加フォントの学習時間を横軸に取り、追加フォント数毎の追加学習性能を表した。分裂のしきい値 α を小さく設定した方が、初期学習フォントに対する認識率の劣化が少ないことが分かる。

初期学習同様、追加学習時においても、分裂中心型で学習した方が良好な追加学習性を有しており、本モデルの特徴であるネットワークの結合を適応的に変化させられる点が追加学習に有効に作用していると考えられる。

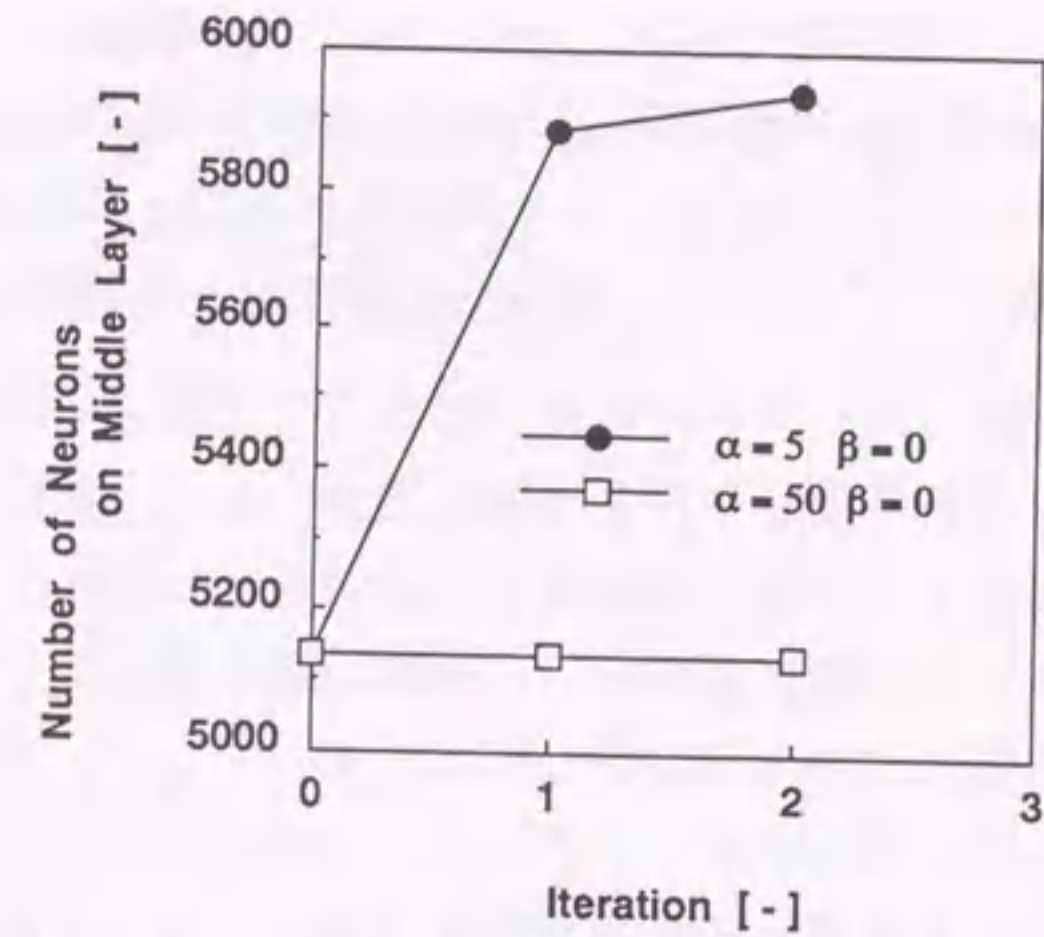


図3.19 追加学習時の分裂条件と中間層ニューロン数

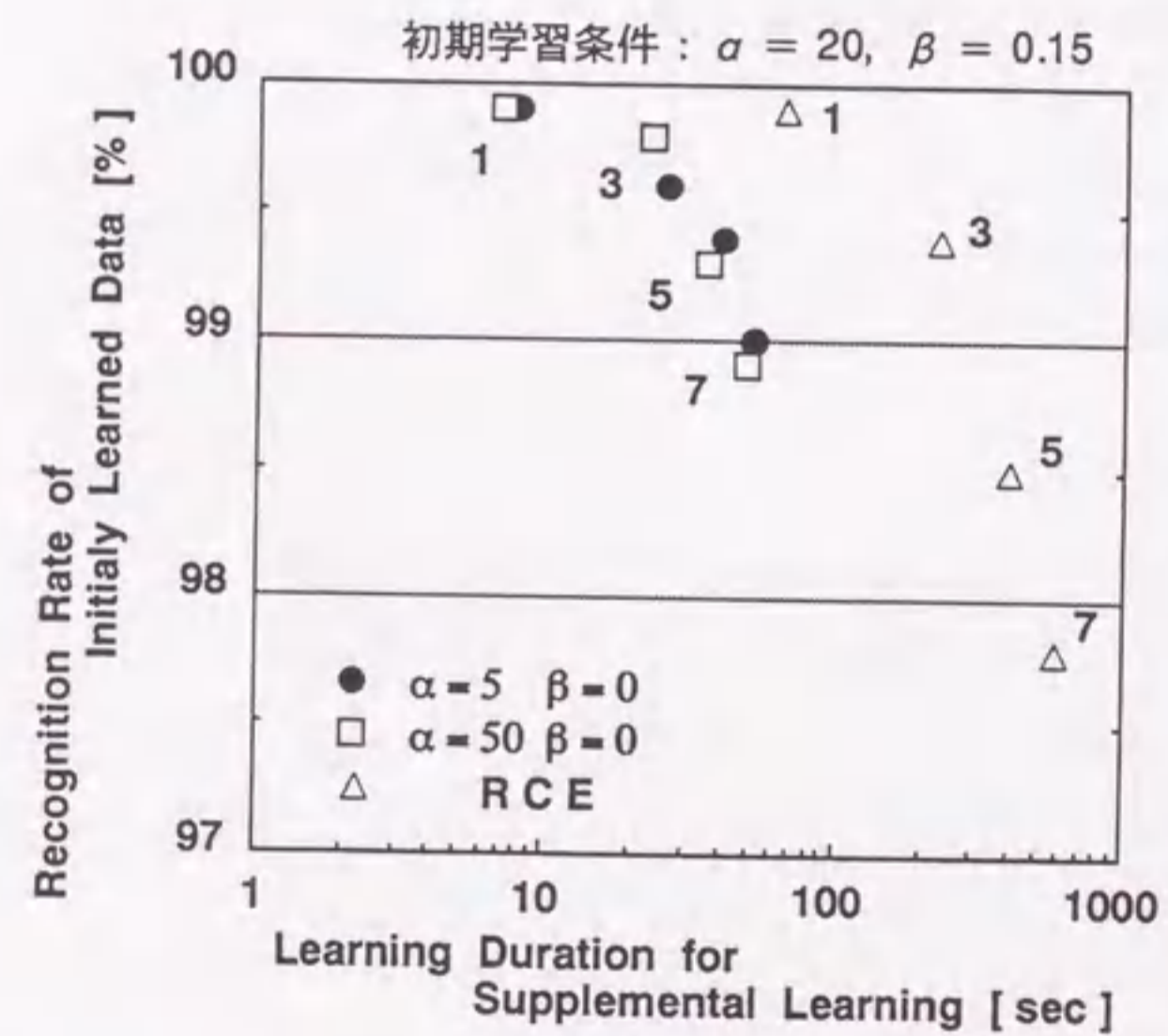


図3.20 追加学習時の分裂条件 α による追加学習性

(4) 他のネットワークモデルとの追加学習速度比較

図3.20には、これまで高速な追加学習の点で注目されていたRCEネットワークの追加学習性能も示している。RCEネットワークの実験条件は、前節のニューロンを用いたニューラルネットワークに対するものと同じである。

初期フォントに対する認識率は、追加フォント数が少ない場合には、両モデルとも同等

の性能を示しているが、追加フォント数が増えると、本モデル ($\alpha = 5$) の方が認識率の劣化が少なく、7フォント追加時においても99.0%と高い認識率を示している。

追加学習速度は、本モデルの方が圧倒的に速く、約10倍の高速学習が可能である。これは、RCEネットワークが入力ベクトルと参照ベクトルとの距離計算が必要なのに対し、本モデルはベクトルの各次元のスカラー量の比較演算のみで学習が行えるためであると考えられる。

3.4 まとめ

本章では、既に学習で獲得した認識性能を確保しつつ、新しい学習データに対する認識性能を獲得するために、学習済みの状態に対する影響が少なく、しかも高速な追加学習機能を実現するニューロンモデルと、それをを用いたニューラルネットワークモデルを提案し、マルチフォント英数字認識に適用して有効性を明らかにした。提案したモデルは、入力特徴量の量子化を行う多段階のしきい値処理機能を有する量子化ニューロンモデルと、入力特徴量に応じて階層的に接続し、学習を行う最終層で統合化する機能別階層ネットワークである。マルチフォント英数字認識に適用し、その高速学習性と追加学習性が優れていることを、他の代表的なネットワークモデルと比較検討する中で明らかにした。さらに、このモデルを、入力データに応じて適応的に増殖し、ネットワークの自動構築が可能な適応増殖量子化ニューロンモデルと、それをを用いた機能別階層ネットワークに拡張した。このにおいても、その高速学習性と追加学習性は保たれており、これまで高速追加学習の点で注目されていたRCEネットワークに比べても、約10倍の高速化を実現した。

第4章

ニューラルネットワークによる 手書き文字認識

4.1 はじめに

近年、OA化の進展に伴って、種々の印刷物や手書きの伝票等を、コンピュータに自動入力したいという社会的ニーズが高まり、文字認識の技術が広く利用されるようになってきた。しかし、手書き文字など対象が一般化されたものについては、まだまだ解決されるべき技術的課題が多く、十分な認識性能が得られていない。この技術的課題の中で特に、手書き文字の大きな字形変動に対応できる判断機構と、画像イメージの中に存在する文字を特定し、その判断機構に一文字一文字を送り込むための文字の切り出しが重要である。

ニューラルネットワークは従来のパターン認識手法よりも有効であると期待され、盛んに研究されているが、手書き文字の認識に対しては、その実用性は明確になっていない。手書き文字の認識性能を高くするには、既知でないパターン変動に対しても認識判断ができなければならない。これにはニューラルネットワークの特徴の一つである、汎化能力を活用することが期待される。また、代表パターンを教えるだけで、その変形パターンの認識ができるネオコグニトロン [50] [51] というニューラルネットワークモデルがある。このモデルのパターン変形への対応には優れたものがあるが、その研究アプローチの性格上、大規模かつ構成が複雑であり、工業的実用性には課題が多い。実用的観点でみた場合には、階層型ニューラルネットワークを手書き文字認識へ適用したもの [6] [7] [9] [52] [53] や自己組織化写像をベースにしたニューラルネットワークモデルの ComNet [54] を手書き文字認識に適用した研究があり、従来の文字認識手法と劣らない性能は得られるが、汎化性、つまり未学習パターンに対しても判断を的確にできることに関してはまだ不十分である。手書き文字では筆記者の字癖や筆記具等による字形変動が大きく、同一カテゴリでも多種多様なパターンが存在する。それゆえ、その変動の大きさをうまく

吸収する工夫が必要になる。ニューラルネットワークを用いてその変動に対処する場合、ニューラルネットワークの規模を大きくすることがあげられるが、一般に、ニューラルネットワークの規模が大きくなるにつれて、学習が困難になり、なおかつ、学習データへの近似能力が過剰になるために、汎化能力が低下する等の問題が生じてくる。

また、手書き文字の認識性能を向上させるには、文字切り出し処理においても文字パターンの変動に対処できる柔軟な処理が重要である。手書きの文字は文字外形の変動は非常に大きく、これに対処できる十分な手法はあまりない。手書き文字列に着目した場合、ラベリングによって黒画素連結成分を矩形化し、一定の文字の大きさや、正方形度の条件を用いて、矩形の統合、分割を繰り返す処理 [67] が唯一であり、処理手法の確立はまだまだである。

本章では、以上のような問題点を解決するために、先ず特徴空間を複数の識別容易な部分空間に大分類し、各部分空間毎に独立に入力パターンの認識を行い、それらの結果を統合判断することで複雑な識別境界を獲得させる構造のニューラルネットワークモデルを提案し、手書き文字認識においてその認識性能を明らかにする。更に、認識に先立つ文字の切り出しに際し、手書き文字の大きな外形変動に対応できる、文字外形に関する知識を埋め込んだニューラルネットワークモデルを提案し、その有効性を明らかにする。

以下、4.2節で手書き文字の変動を吸収し認識判断を実行できる、学習性と汎化性を確保したファジィ推論アーキテクチャに基づく構造化ニューラルネットワークを提案し、手書き郵便番号認識の実験を行い、その性能を明らかにする [55] [56] [57]。4.3節で手書き文字の大きな外形変動に対応できる、文字外形に関する知識を埋め込んだネットワークと、ノイズなどを除去するネットワークとの相互作用により、文字を柔軟に切り出すニューラルネットワークモデルを提案し、従来のものより優れた性能を得られることを示す [68] [69] [70]。そして、4.4節でまとめを行う。

4.2 ファジィ推論アーキテクチャに基づく 構造化ニューラルネットワークによる手書き数字認識

手書き文字の大きな字形変動に対しても柔軟に対応するために、複数の小規模なニューラルネットワークを機能的に組み合わせたファジィ推論アーキテクチャに基づく構造化ニューラルネットワーク (NARA: Neural networks designed on Approximate Reasoning Architecture) [58] を手書き文字認識ができるように拡張する。

以下本節では、手書き文字認識に対応できるNARAの構成について明らかにし、これを手書き数字認識に適用し、その認識性能を従来のニューラルネットワークモデルと比較検討する。

4.2.1 構造化ニューラルネットワーク (NARA) の構成

数字はカテゴリ数は少ないが、構造が簡単で自由度が大きいため、手書きの場合、筆記者の字癖や筆記具等による字形の変形が大きく、同一カテゴリでも多種多様なパターンが存在する。図4.1はこの状態を文字の特徴空間上で模擬的に表したものであり、文字のカテゴリの識別境界は非常に入り組んだ複雑なものとなっている。従って、これを単一のNNで学習しようとしても、その複雑な境界を獲得できず認識性能が低下する。

そこで、図4.1の太線が示すように、まず特徴空間を複数の識別容易な部分空間 (図4.1ではA, B, C) に大分類し、各部分空間毎に独立に入力パターンの認識を行い、その後これらの結果をうまく統合するならば、複雑な識別境界も獲得できると考えられる。

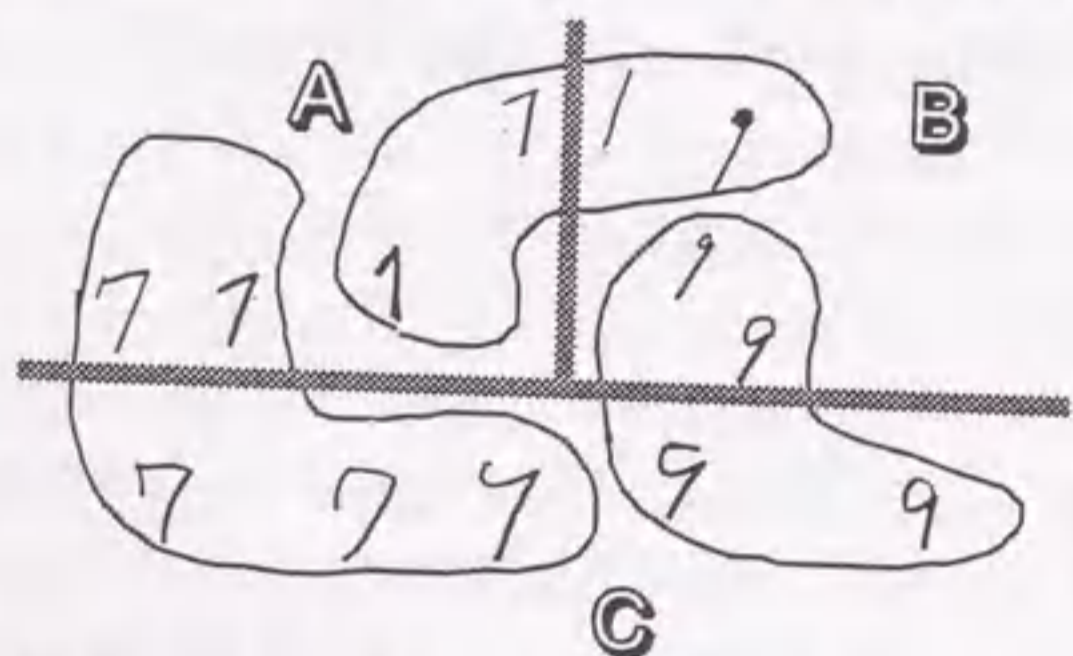


図4.1 特徴空間における文字の分布の例

以上の考察に基づき、文字の特徴空間の大分類を行う大分類部と、各部分空間において認識を行う複数のニューラルネットワークから成る細分類部と、すべての細分類部の出力と大分類部の出力をファジィ推論に基づいて統合し最終的な認識を行う統合部によって構

成するものが構造化ニューラルネットワーク (NARA) である。以下各部の構成及び動作について明らかにする。

(1) 全体構成と認識方法

図4.2にNARAの構成図を示す。図4.2において、FVQ (Fuzzy Vector Quantizer) が大分類部であり、 $NN_1 \sim NN_N$ が細分類部である。ファジィ推論に対応させてNARAを説明する。大分類部は、ファジィ推論の前件部 (IF部) に相当し、入力空間をファジィクラスタリング [59] する。細分類部は、後件部 (THEN部) に相当し、分割された部分空間毎に入力データが各カテゴリに類似している度合を求める。統合部は、菅野らの簡略化ファジィ推論 [60] を後件部がベクトル値を出力する非線形関数で表されるように拡張したファジィ推論を用いて、大分類部と細分類部の出力から最終的な認識結果を求めている。

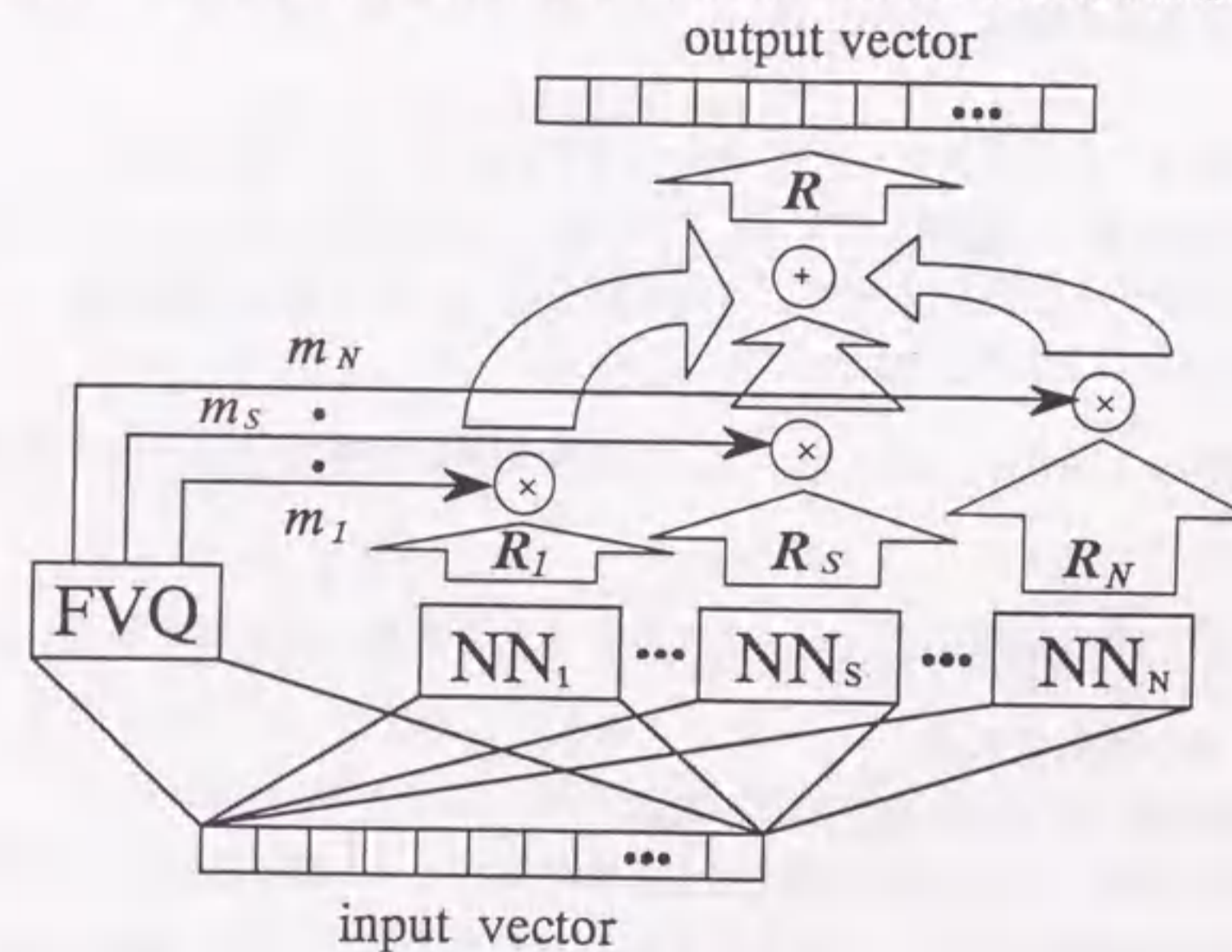


図4.2 NARAの構成

NARAにおける入力データの認識アルゴリズムを順を追って説明する。

[1] 入力データの特徴ベクトル \vec{X} を大分類部に入力し、 \vec{X} が各グループ C_i ($i=1, \dots, N$; N はグループ数) に属する度合 (\vec{X} の各グループへの帰属度) を求める。これを大分類部のメンバーシップベクトル \vec{M} とする。

$$\vec{M} = (m_1, m_2, \dots, m_N)$$

[2] 入力データの特徴ベクトル \vec{X} を細分類部のすべての NN_s ($s=1, 2, \dots, N$) に入力し、各 NN_s の出力層の値を求める。これを、認識ベクトル \vec{R}_s ($s=1, \dots,$

N) とする。

$$\vec{R}_s = (r_{s1}, r_{s2}, \dots, r_{sp})$$

p : 出力ユニット数

- [3] 各細分類部から出力された \vec{R}_s を M の各要素で重みづけする。こうして各グループ毎に得られた認識ベクトル $m_s \cdot \vec{R}_s$ を同一カテゴリを表す要素毎に加算し (同一カテゴリのうち上位要素を加算)、最終認識ベクトル \vec{R} を得る。

$$\vec{R} = \sum_{s=1}^N m_s \cdot \vec{R}_s \quad (4.1)$$

- [4] \vec{R} の要素の中で最も値の大きいもの r_{max} に対応するカテゴリを認識結果とする。但し、次に大きいカテゴリの値 r_{sec} との比 (r_{sec}/r_{max}) が、予め設定されたしきい値より大きい場合はリジェクトを行う。

(2) 大分類部の構成

大分類部は、クラスタリング手法の一つであるベクトル量子化器 (VQ: Vector Quantization) を拡張して構成している。VQ は、入力空間上にデータの分布に応じて参照用ベクトルを配置することによって、入力空間をいくつかの部分空間 (クラスタ) に分割するものであるが、以下の2点について拡張した。

- ・参照用ベクトルの生成において、各クラスタに含まれるカテゴリ数の均一化を図っている。
- ・入力データの分類にファジィクラスタリングを用いて認識する。

以下に、その詳細を示す。

(a) 参照用ベクトル生成アルゴリズム

入力部分空間 (クラスタ) を代表する参照用ベクトル \vec{W}_i の生成アルゴリズムは、ベクトル量子化器の設計法として、クラスタ内のカテゴリ数がしきい値以上の場合のみ、通常用いられているリンデ・ブゾー・グレイの LBG アルゴリズム [62] を用いて、クラスタの分 行うように改良したものである。このような条件を組み込むことによって、各クラスタに含まれるカテゴリ数の上限が制限され、各クラスタに含まれるカテゴリ数を均一化することができる。それゆえ、

細分類部における出力ユニット数を均一化でき、細分類部の学習を容易に行える。

具体的には、学習ベクトル \vec{X}_k ($k=1, \dots, L$: L は学習データ数) を用いた次のようなアルゴリズムである。ここで、学習に必要なパラメータは

ε : クラスタ内分散の変動のしきい値

δ : クラスタ内分割のパラメータ ($0 < \delta < 1$)

Θ : 一つのクラスタ内に認める最大カテゴリ数

である。また、参照用ベクトルの集合を

$$A_N = \{\vec{W}_1, \vec{W}_2, \dots, \vec{W}_N\} \quad \text{但し、} N \text{ は参照用ベクトル数}$$

これらの参照用ベクトルによって分割される学習ベクトルの部分集合を

$$P(A_N) = \{S_1, S_2, \dots, S_N\}$$

とする。また、各 S_i に含まれる学習ベクトルのデータ数を g_i 、同じくそのカテゴリ数を c_i とする。以下、 t は計算のループ変数であり、 $D(t)$ は以下で定義するクラスタ内平均距離の平均で、DD は十分に大きい値に設定することを意味する。

[1] 初期化

$$N=1, \varepsilon > 0, D(-1)=DD, t=0$$

$$\vec{W}_1(0) = \sum_{i=1}^L \vec{X}_i / L, A_1(0) = \{\vec{W}_1(0)\} \quad (4.2)$$

[2] すべての X_k を $A_N(t)$ によって、N 個の部分集合

$$P(A_N(t)) = \{S_1(t), S_2(t), \dots, S_N(t)\}$$

に分類する。すなわち、もし $\|\vec{X}_k - \vec{W}_c(t)\| < \|\vec{X}_k - \vec{W}_i(t)\|$ ならば

$$\vec{X}_k \in S_c(t) \quad (i \neq c, i=1, \dots, N)$$

とする。

[3] すべてのクラスタ内平均距離の平均を求める。

$$D(t) = \sum_{i=1}^N \sum_{\vec{X}_k \in S_i(t)} \|\vec{X}_k - \vec{W}_i(t)\| / L \quad (4.3)$$

[4] もし $(D(t-1) - D(t)) / D(t) < \varepsilon$ なら [6] へ、それ以外なら [5] へ。

[5] $P(A_N(t))$ を用いて、参照用ベクトル

$$A_N(t+1) = \{\vec{W}_1(t+1), \vec{W}_2(t+1), \dots, \vec{W}_N(t+1)\}$$

を再設定する。すなわち、

$$\vec{W}_i(t+1) = \sum_{\vec{X}_k \in S_i(t)} \vec{X}_k / g_i(t) \quad (4.4)$$

を求め、 $t = t + 1$ として [2] へ。

[6] クラスタ分割処理を行う。

各 $\vec{W}_i(t)$ に対応するクラスタ $S_i(t)$ において、もし $c_i > \Theta$ ならば、参照用ベクトル $\vec{W}_i(t)$ を、[6-1] ~ [6-3] に示すようにクラスタ内分散の大きい軸に沿って 2 つの参照用ベクトル $\vec{W}_i(t)$ 、 $\vec{W}_{N+1}(t)$ に作り直す。但し、すべての c_i が $c_i < \Theta$ を満たすならば終了する。

[6-1] クラスタ $S_i(t)$ の第 j 次元方向の標準偏差 σ_{ij} ($j=1, \dots, n$) を求める。

[6-2] $\sigma_i = (\sigma_{i1}, \sigma_{i2}, \dots, \sigma_{in})$ の最大値 σ_{imax} を求める。

[6-3] $\vec{W}_i(t)$ を $\sigma_{i_{max}}$ 軸上に沿って、 $\delta\sigma_{i_{max}}$ を加えたものと、 $\delta\sigma_{i_{max}}$ を減じたものをそれぞれ $\vec{W}_i(t)$ 、 $\vec{W}_{N+1}(t)$ とする。

$N=N+1$ 、 $D(-1)=\infty$ 、 $t=0$ として、[2]へ。

(b) 認識動作

未知の入力データを大分類する際に、入力データが各クラスにどのくらいの割合で所属するか(帰属度) m_s を求めるファジィクラスタリングを行う。ここで、入力ベクトルの各クラスへの帰属度は、 $f (>1)$ をファジィ性を調節するパラメータとして設けた次式を用いて求める。

$$m_s = 1 / \left(\sum_{i=1}^N d^2_{si} / d^2_{i1} \right)^{1/(f-1)}, \quad (4.5)$$

但し、 $d^2_{si} = \|\vec{X} - \vec{W}_i\|^2 = \sum_{j=1}^n (x_j - w_{ij})^2$

ここで、 $\vec{X} = (x_1, x_2, \dots, x_n)$: 入力ベクトル
 $\vec{W}_i = (w_{i1}, w_{i2}, \dots, w_{in})$: 参照用ベクトル
 n : 入力ベクトルの次元数

なお、帰属度 m_s は

$$m_s \in [0, 1], \quad \sum_{s=1}^N m_s = 1$$

を満たす。そして、パラメータ f を1に近づけるほど、ハードクラスタリングに近づき、 f が1より大きくなるほどファジィなクラスタリングを行うことができる。つまり、NARAでは f をある程度大きくすることによって、未知入力に対する判断を行えるようにしている [61]。

このようにして求めた帰属度を大分類部におけるメンバーシップベクトルとして用いる。

(3) 細分類部の構成

細分類部の各ニューラルネットワークは、学習ベクトル量子化(LVQ)によって構成する。

大分類部における参照用ベクトルの生成後、各学習データに対して、どの参照用ベクトルが最近傍ベクトルであるかを求めることによって、すべての学習データは、参照用ベクトル数と同数のグループに分割される。各学習ベクトル量子化ニューラルネットワークは、分割されたグループ内に含まれる学習データのカテゴリを識別するために、学習アルゴリズムのLVQ3 [41]を用いて学習を行う。なお、各LVQの参照用ベクトルは、そのグ

ループに属する1カテゴリ当りの学習データ数に応じて、1カテゴリあたり複数個用意する。また、LVQの参照用ベクトルの初期値は、各カテゴリ毎に学習データをベクトル量子化することによって設定した。

学習ベクトル \vec{X} とすべての参照用ベクトルとの距離を計算し、式(4.5)と同様にファジィクラスタリングを行って、認識ベクトルを求める。

(4) リジェクト処理

NARAでは、大分類部、及び各細分類部から出力されるメンバーシップ値は、基本的には、入力ベクトルと各参照用ベクトルの距離をその各距離の総和で正規化したものであり、各メンバーシップ値は入力パターンと各参照用ベクトルとの間の類似度とみなすことができる。それ故、これらの乗算結果である最終出力は、入力ベクトルが各カテゴリを代表している参照用ベクトル群にどの程度類似しているかを表す類似度である。

一般にリジェクトすべき文字は、ある一つのカテゴリに特定することが不可能な文字である。このようなリジェクトすべき文字の特徴ベクトルは、この文字に類似した二つのカテゴリをもつ参照用ベクトル間の中央(つまり、カテゴリの境界領域)に位置することが多くなる。この場合、類似度である最終出力はどちらのカテゴリに対してもほぼ同じ値が得られることが容易に予想される。

そこで、リジェクト処理は、最大出力値と2番目に大きい出力値がある程度以上類似している場合はリジェクトするという以下の方法で行なう。

$$\text{リジェクト if } r_2 / r_1 > T_h \quad (4.6)$$

但し、 r_1 : 第1候補の出力値
 r_2 : 第2候補の出力値
 T_h : 予め設定したしきい値

つまり、第一候補の出力値と第二候補の出力値との比が予め設定されたしきい値以上に類似している場合、第一候補の出力結果をリジェクトする。

4.2.2 手書き数字認識実験

NARAの手書き文字に対する耐変動性を検証するために、手書き数字を対象とした認識実験を行った。

NARAへの入力データとしては、文字のデータをそのまま与えるのではなく、位置変動の吸収と入力次元の削減を目的として、各画素を2値化した後、文字部分の切り出し、大きさの正規化、及び雑音除去など必要な前処理を行う。そして、96次元の輪郭方向密度

特徴量と背景密度特徴量 [63] を求めて与えた。

輪郭方向密度特徴は、局所領域内のストロークの方向及び複雑性を示す特徴量である。具体的には、図4.3に示すように正規化後の文字イメージを4×4の粗いメッシュ領域に分割し、各メッシュ領域に含まれる文字輪郭の方向性が図4.3に示す4方向のいずれに属するかを求め、各メッシュ領域毎に各方向の密度を求めた64次元の特徴量である。

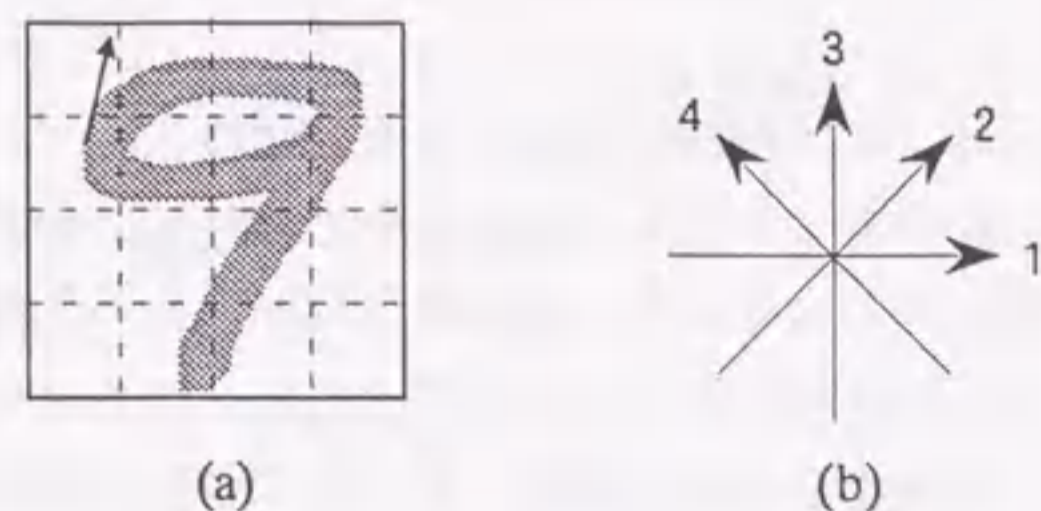


図4.3 輪郭方向密度特徴

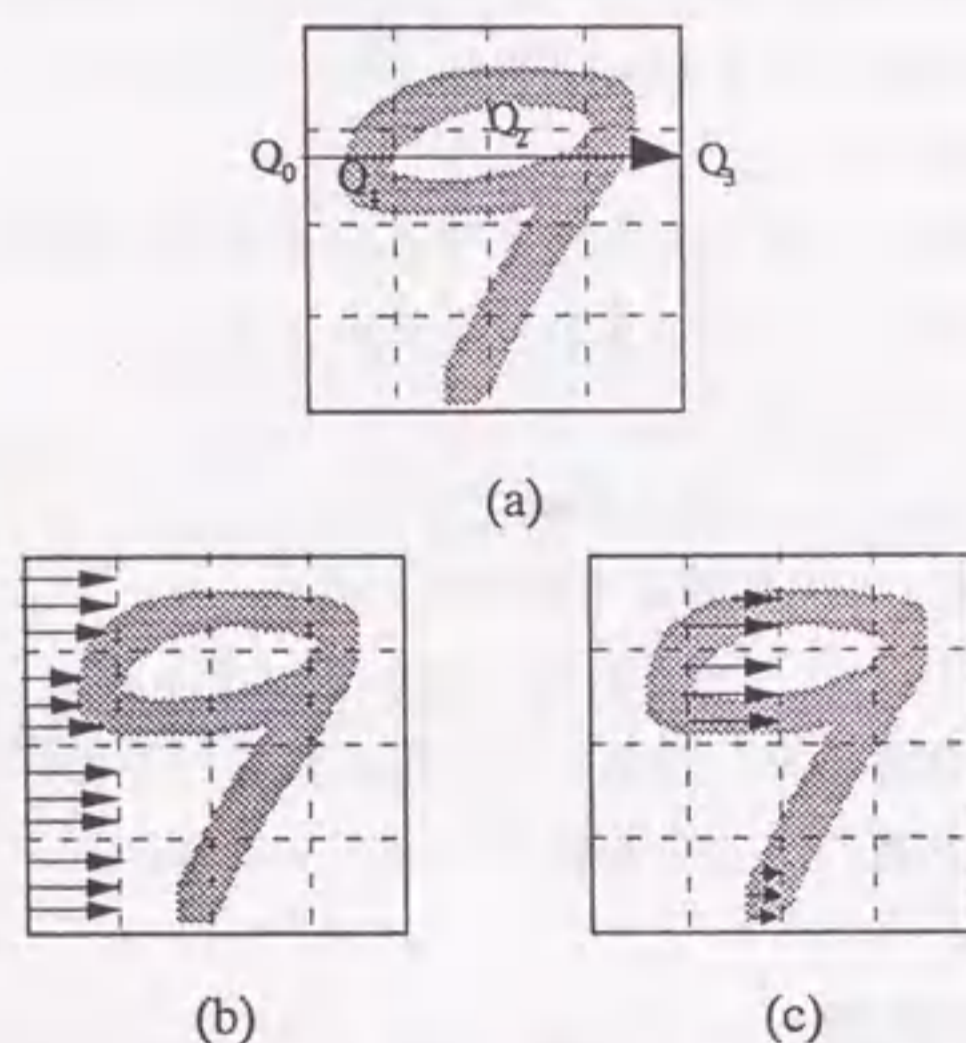


図4.4 背景密度特徴

背景密度は、文字背景の複雑さを示す特徴量である。図4.4(a)に示すように文字切りだし矩形の各辺から対辺へ走査し、文字の背景部から文字部が変わる点 Q_1 を検出し、点 Q_1 と点 Q_0 間のデータ列をレベル1のデータ列と定義する。正規化後の文字イメージを4×4の粗いメッシュ領域に分割し、矩形の各辺毎に、その辺に最も近いメッシュ領域におけるレベル1のデータ列の密度(図4.4(b))、及びその辺に2番目に近いサブ領域におけるレ

ベル2のデータ列の密度(図4.4(c))からなる32次元の特徴量である。

なお、輪郭方向密度、背景密度とも領域分割は、文字の変形を吸収するために重心分割を用いた。このようにして得られた二つの特徴量を組み合わせ96次元の特徴量とする。

手書き数字認識に対するNARAの有効性を確認するために、ニューラルネットワークの構造化による効果を見るために、従来のニューラルネットワークモデルとの認識性能を比較した。更に、NARAの性能を向上させるために、構造化の最適化の検討を行った。

(1) 構造化による効果

構造化の効果を従来のニューラルネットワークと比較することにより検証する。比較する従来のニューラルネットワークとしては、BP法による学習を行う階層型ニューラルネットワーク(BPNN)と学習ベクトル量子化(LVQ3)を用いた。以下、NARA及びBPNN、LVQ3の各ニューラルネットワークモデルの構成を示す。各モデルの総ニューロン数は異なるが、従来のニューラルネットワークモデルは学習時に未提示のデータに対する認識性能が最高になるようにパラメータのチューニングを行ったものである。このため、BPNNはニューロン数が他のモデルに比較して少なくなっている。

表4.1 ニューラルネットワークモデルの構成

モデル	ニューロン数	
	NARA	大分類部、細分類部の入力層のニューロン数 大分類部のグループ数 各細分類部の出力層のニューロン数
BPNN	入力層のニューロン数 中間層のニューロン数 出力層のニューロン数	96 48 10
LVQ3	入力層のニューロン数 出力層のニューロン数	96 1280

認識対象とする手書き数字は、主に工業技術院電子総合研究所作成の自由手書き文字データベースETL1を用いて、1字種あたり1500文字を用意した。実験に際しては、そのうち半分(7500文字)を学習データとして用い、残りを評価データとして用いた。

結果を表4.2に示す。ここで、認識精度Sは、郵政省郵政研究所主催の文字認識コンテストで用いられた認識性能を総合評価する評価基準で、誤り率(E)及びリジェクト率(R)を用い

$$S = R + 10 \times E \quad (4.7)$$

で表されるものである。Sが小さいものほど優れた認識性能をもつ。従って、リジェクトの性能が認識精度を左右するポイントであり、効率よく誤りデータをリジェクトし誤り率を減少させる能力が非常に重要になる。

表4.2より、NARAは従来のニューラルネットワークモデルに比べ、優れた認識性能を示すことがわかる。これは、単一のニューラルネットワークモデルでは十分に獲得できない複雑なカテゴリ間の識別境界でも、構造化を行うことによって各細分類部において学習が容易になり、最終的に識別境界を獲得できるようになったことを示している。

表4.2 手書き数字の認識結果

	NARA	BPNN	LVQ3
認識精度 S	6.38	7.22	6.92
正読率 [%]	95.41	94.80	95.24
誤り率 [%]	0.20	0.28	0.24
リジェクト率 [%]	4.38	4.92	4.52

(2) 構造の最適化

構造化を行うことによって識別境界の獲得が容易になり、学習データに対しては大分類のグループ数を多くすればするほど認識性能は向上する。しかし、大分類部のグループ数を多くするほど細分類部では学習が容易になるが、過学習になり易く、未学習データに対しては認識性能が劣化する可能性がある。更に、グループ数を多くすることによって大分類部における誤分類を多く生じるようにもなり、これも認識性能の低下を招くことになる。そこで、大分類部のグループ数の認識性能に与える影響を明らかにする。

大分類部のグループ数が2、4、8、13、16のNARAを構成し、その認識精度として式(4.7)を求めた。但し、細分類部も出力層のニューロンの数が、各カテゴリあたり最大32個の参照用ベクトルが配置されているものと、最大128個の参照用ベクトルを配置されているものの2通りを、各大分類部のグループ数毎に用意し学習を行い、これら5×2通りについて評価を行った。

認識対象とする手書き数字は、主に工業技術院電子総合研究所作成の自由手書き文字データベースETL1と郵政省郵政研究所作成のサンプルデータを合わせて、27,486文字を用意した。このうち25,270文字(1字種当たり2,527文字)を学習データとして用い、残り

2,216文字を評価データとした。この評価データはすべて郵政省のサンプルデータである。

図4.5に実験結果を示す。ここで、 T_{max} は各カテゴリ当たり設定する参照用ベクトルの最大個数である。この図より、細分類部の最大参照ベクトル数に関わらず、大分類部のグループ数が認識精度に影響を大きく与えることがわかる。NARAにおいて認識性能を向上させるためには、データに応じて最適な大分類部のグループ数を設定することが非常に重要であることがわかる。手書き数字に関してはグループ数8が最適値である。この時、 $T_{max}=32$ において、認識精度は8.57(正読率94.7%、誤り率0.36%、リジェクト率4.96%)を達成しており、非常に高い認識精度を実現している。同じ文字特徴量を用いた、代表ベクトルとの比較による辞書方式の認識手法では、認識精度は倍以上の値となることを確認している。

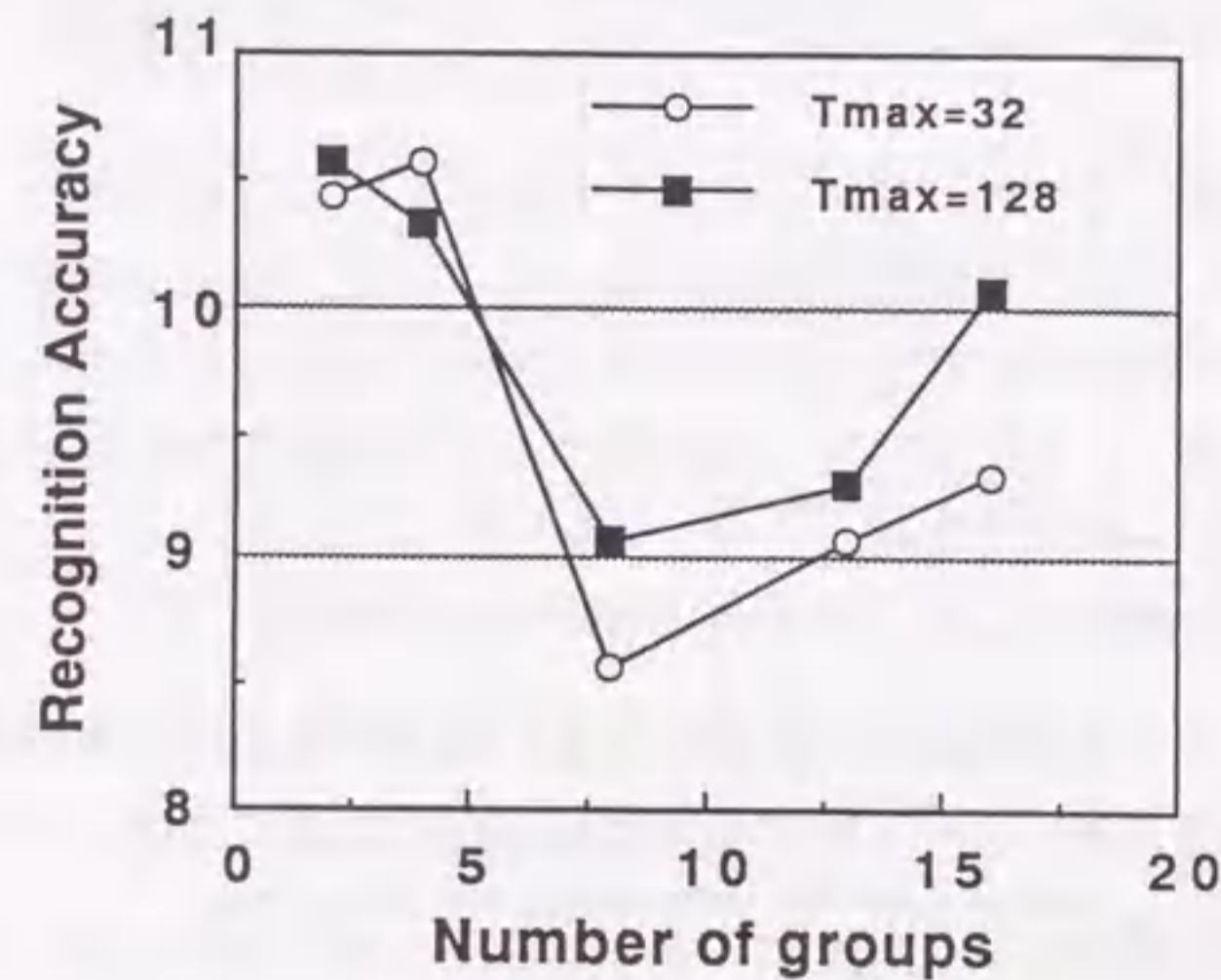


図4.5 大分類部のグループ数と認識精度の関係

4.3 ニューラルネットワークを用いた柔軟な文字切り出し

人間が文字列から文字を抽出する場合には、2通りの過程があると思われる。一つは全体を大ざっぱに見たパターン配置から文字位置を推定する場合と、もう一つは文字と思われるパターンを一旦認識し、意味情報まで考えてこれは文字であると判断する場合である。

本節では、この点を注目し、手書き文字の大きな外形変動によって生じる結合、分離文字が存在しても柔軟に対応できる、文字外形に関する知識を埋め込んだネットワークとノイズなどを除去するネットワークとの相互作用により、文字を柔軟に切り出すニューラルネットワークモデルを提案し、従来のものより優れた性能が得られることを示す。

4.3.1 手書き文字切り出しニューラルネットワークの構成

文字切り出し処理には当然、高い切り出し性能が求められる。また、文字認識システムにおける前処理としては、単純な図形的特徴をもちいて、簡単な処理で高速に行う必要がある。そこで、文字列の黒画素を文字列と垂直方向に射影したヒストグラム成分に適当な雑音除去を施した後、しきい値処理した結果から文字の切り出し位置を文字端から順に逐次的に推定するのが一般的である。

このような手法の課題として、以下のことがあげられる。

- ・文字の大きさや文字間の長さの変動が大きい場合には、処理条件を一定にすると、切り出し誤りが生じ易く、一ヶ所でも切り出し誤りが生じると、それは全体に影響
- ・認識、後処理の結果をフィードバックすることで更に精度を向上させることが可能であるが、その場合には、できるだけ単純なパラメータ制御によって複数の切り出し候補を選び出せることが必要

以上の課題に対しては、エネルギー関数の最小化という単純な枠組みを用いた並列演算によって解を求める、ホップフィールド型ニューラルネットワーク [64] の特徴を生かすことが考えられる。

この特徴を生かした例として、初期視覚の処理 [66] がある。これは、両眼視によって得られるノイズを含む奥行きデータから、物体のエッジを保存しながら曲面を再構成する処理である。画素間の不連続性を取り扱う変数としての線過程と、画素値を表す変数としての強弱過程とを相互に作用させながら、画像の一般的性質を表す制約条件を満たすようにエネルギー関数を最小化することによって処理を進め、画像のエッジ部分の推定と、ノイズ成分を除去した曲面の再構成が可能になる。

ここで、この画素間の不連続性を射影ヒストグラム値の不連続性に置き換え、文字の切

り出しとはこの不連続点を見つけることであるととらえると、初期視覚の処理と同じ枠組みで文字の切り出し点を見いだすことが可能になる。

(1) 文字切り出しニューラルネットワークの構造

射影ヒストグラムを用いた文字切り出しアルゴリズムは、一般に図4.6に示すように構成されている。

- [1] 文字の黒画素の列方向のヒストグラムを求め、平滑化を行う。
- [2] ヒストグラムの谷の部分に切り出し点があるという一般的な知識により、求めた射影ヒストグラムを適当な値でしきい値処理する。
- [3] 文字の高さと横幅、文字数に関する一般的な知識を基にしきい値処理した結果を統合、分割し、一文字の位置を推定する。

この時、統合分割に際して、処理は逐次的に行われるために、誤りの連鎖とバックトラックの必要が生じる。これらの処理をネットワークの相互作用を用いて並列に進めることで、精度と処理効率を上げたのが以下に示す文字切り出しニューラルネットワークである。

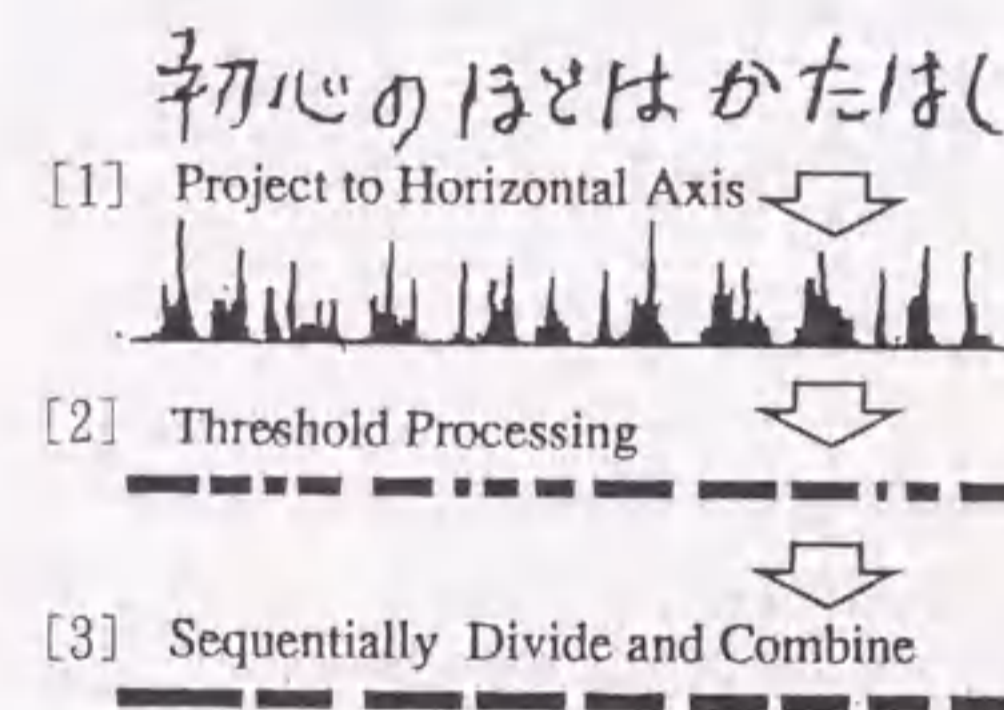


図4.6 射影ヒストグラムを用いた従来の文字切り出し手法

文字切り出しニューラルネットワークを図4.7に示す。文字列中の画素を文字列の垂直方向に射影して得られる射影成分を扱う射影値ネットワークと、切り出し位置を扱う切り出しネットワークとを階層的に結合した2層構造のネットワークである。射影値ネットワーク、切り出しネットワークの各々は、ホップフィールド型ニューラルネットワークで構成する。射影値ネットワークには、ノイズ除去のための知識を制約条件として表したエネルギー関数を設定する。切り出しネットワークには、切り出しのための知識を制約条件として表したエネルギー関数を設定する。

第1層目の射影値ネットワークでは、射影ヒストグラム分布を”ぼかす”ことでかすれ

やノイズによる影響を軽減する。

第2層目の切り出しネットワークは、そのぼかされた射影ヒストグラム分布から切り出し位置を推定する。このニューラルネットワークのエネルギー関数が最小になったときに、発火しているニューロンの位置を切り出し位置とみなす。

ここで重要なことは、文字切り出しを柔軟に行うための制約条件とエネルギー関数をどのように設定するかである。

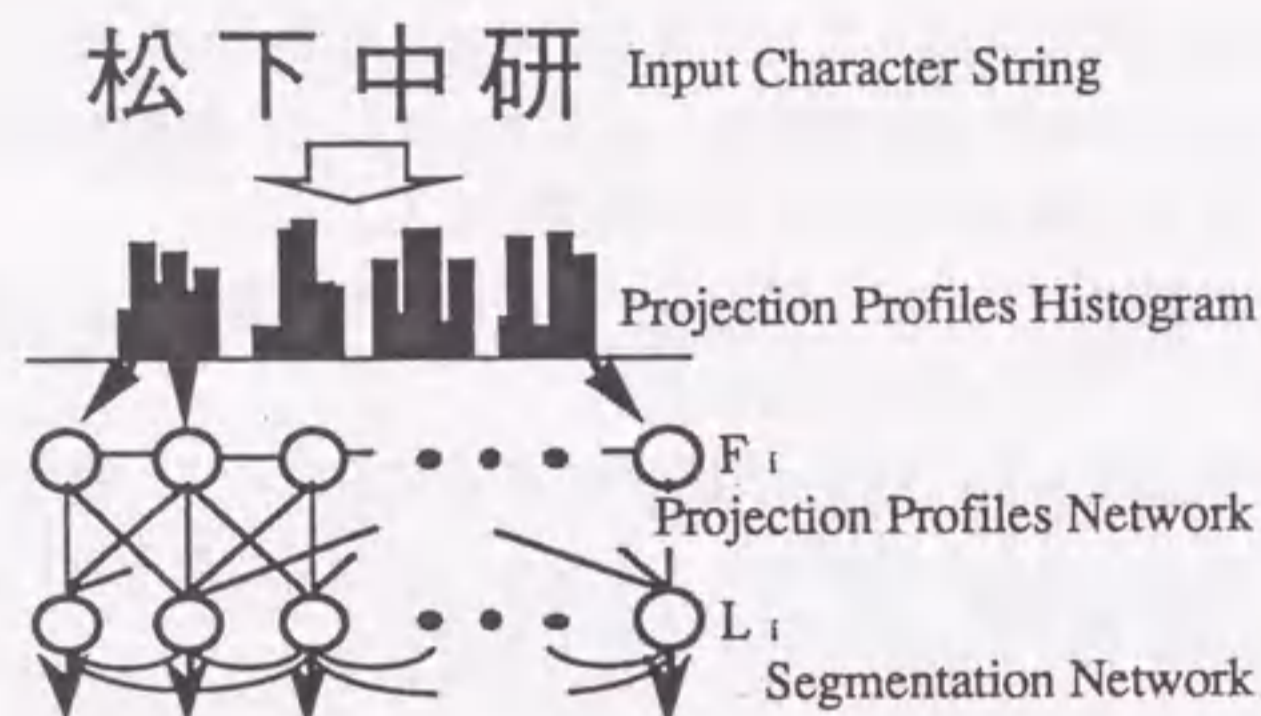


図4.7 文字切り出しニューラルネットワークの構造

(2) 制約条件とエネルギー関数

ノイズを除去する制約条件を表すエネルギー関数と、文字切り出しに使われる知識の制約条件を表すエネルギー関数は次のように設定する。ここで、射影値ネットワークと切り出しネットワークにおいて、文字列端から i 番目のニューロンの出力値を各々、 F_i 、 L_i とする。

(a) 射影値ネットワーク

雑音を除去するために、射影ヒストグラム値は滑らかに変化すべきであることから、式(4.8)のエネルギー関数を設定した。

$$E_p = \sum_i (F_{i+1} - F_i)^2 \quad (4.8)$$

式(4.8)からも明らかなように、このエネルギー関数を最小化することは近傍の射影値差分を小さくすることになる。したがって、射影ヒストグラム分布の平滑化が達成される。

(b) 切り出しネットワーク

文字切り出しの知識として、第1に、切り出し位置は、射影ヒストグラム分布の谷であり、かつ射影値が小さいことが挙げられる。このことから、式(4.9)のエネルギー関数を設

定した。

$$E_a = \sum_i \left\{ (1 - L_i) \sum_{j=0}^{\alpha_1} (A - F_{i-j} - F_{i+j}) \times \sum_{k=1}^{\alpha_2} (F_{i-k} + F_{i+k} - 2k \cdot F_i) \right\} \quad (4.9)$$

但し、 A は射影値の最大値によって決定される値である。 α_1 、 α_2 は、図4.7に示すように射影値を考慮する範囲であり、標準文字幅によって決定される。

近傍の射影値 F_i が小さいところで $(A - F_{i-j} - F_{i+j})$ の値が大きくなり、射影ヒストグラム分布が谷であるところで $(F_{i-k} + F_{i+k} - 2k \cdot F_i)$ の値が大きくなる。それゆえ、エネルギー項 E_a を小さくするには、 L_i が1に近くなる必要がある。

このエネルギー項によって、近傍の射影値が小さく、かつ、射影ヒストグラム分布が谷であるニューロンに対してより多くの活性化入力を与えられることになる。

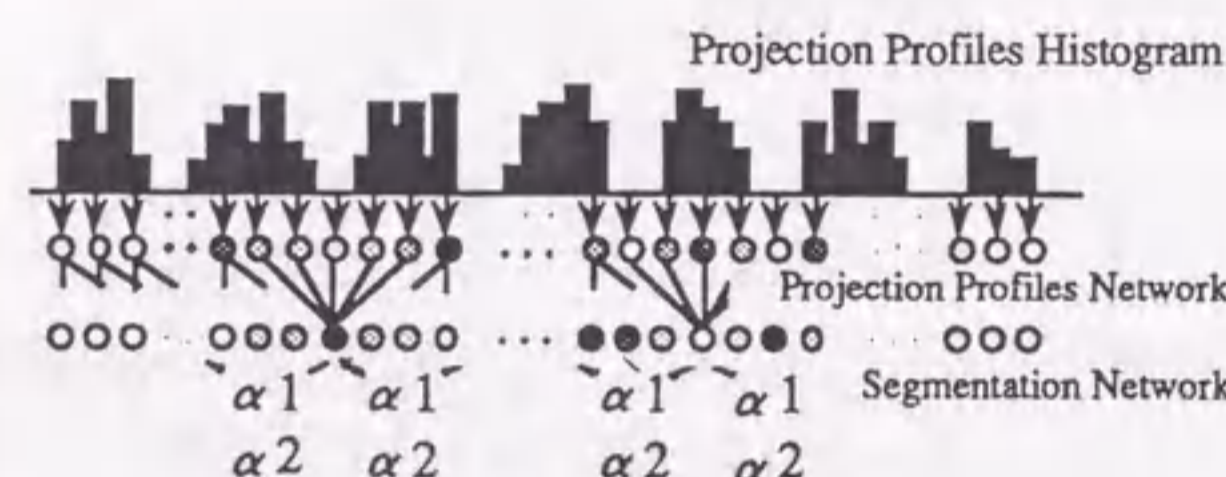


図4.8 エネルギー項 E_a による結合

知識の第2には、切り出し位置はある一定の幅で分布することが挙げられる。このことから、式(4.10)のエネルギー関数を設定した。

$$E_l = \sum_i L_i \sum_{j=1}^{\beta} C(j) (L_{i-j} + L_{i+j}) \quad (4.10)$$

但し、 $C(j)$ は正の値を取り、 j に関して単調減少する関数である。

切り出し位置に相当するニューロン L_i が1に近い大きな値を取るとき、エネルギー項 E_l を小さくするには、その近傍ニューロン L_{i-j} 、 L_{i+j} の値は0に近くなる必要がある。つまり、近傍ニューロンの発火を抑制することで、ネットワークは、発火するニューロンが一定の幅で分布するように作用する。パラメータ β は標準文字幅から決定する。以降この C を抑制関数と呼ぶことにする。このパラメータ β と抑制関数 C の意味は重要であり、図4.9に示すように各ニューロン間での抑制結合の範囲とその抑制強度を表している。 β の値が小さいときは切り出し位置の幅は狭く、また大きいときは切り出し位置の幅も広くな

る可能性が大きくなる。

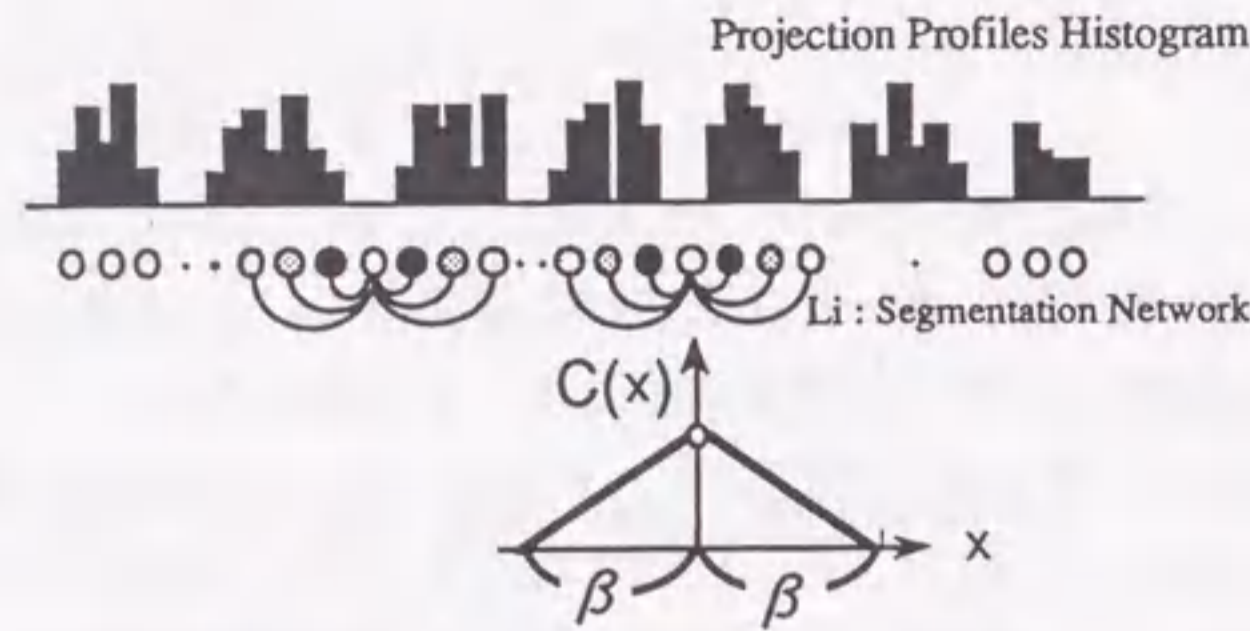


図4.9 エネルギー項 E_1 による結合

第3には、切り出し位置の数は、予め決められた値を持つことが挙げられる。このことから、式(4.11)のエネルギー関数を設定した。

$$E_n = \left(\sum_i L_i - N \right)^2, \quad N = f(\beta) \quad (4.11)$$

但し、 N は切り出されるべき文字の数であり、標準文字幅を決める β を変数とする関数でもある。

エネルギー項 E_n を最小化するには、ネットワークは、切り出し領域における切り出し位置の数 N に、発火するニューロンの数を一致させるように作用しなければならない。

以上の他に、ニューロン自身の動作に関する制約を与えるエネルギー項を設定する。すなわち、ニューロンの出力値は0か1であることから、式(4.12)のエネルギー関数を設定した。

$$E_v = \sum_i L_i (1 - L_i) \quad (4.12)$$

また、ニューロンへの入力総和が発散しないことから、式(4.13)のエネルギー関数を設定した。

$$E_g = \sum_i \int_0^{L_i} g^{-1}(L_i) dL_i \quad (4.13)$$

$$L_i = g(m_i) = 1 / (1 + \exp(-2\lambda m_i))$$

但し、 m_i はニューロンの入力総和、 λ は定数である。

以上の各エネルギー項が同等の影響をもって文字切り出しのエネルギー最小化を実行すると仮定すれば、切り出しネットワークが持つエネルギー関数 E_s を式(4.14)に示すように、上述した各項を重み付きで加えた式で定義する。

$$E_s = E_a + C_1 \cdot E_1 + C_n \cdot E_n + C_v \cdot E_v + C_g \cdot E_g \quad (4.14)$$

4.3.2 文字切り出し実験

文字切り出しの性能を評価するために自由手書きの文書の切り出しを行った。実験データは異なる筆記者によって書かれた図4.10に示すような自由手書きの文書7サンプルを用意した。これらの手書き文書を200dpiで読み込み、射影ヒストグラムによって正しく切り出された文字列を評価データとして用いた。文字数は1サンプル当たり平均で15文字/列、合計791文字である。以下に実験方法と結果及び考察を示す。

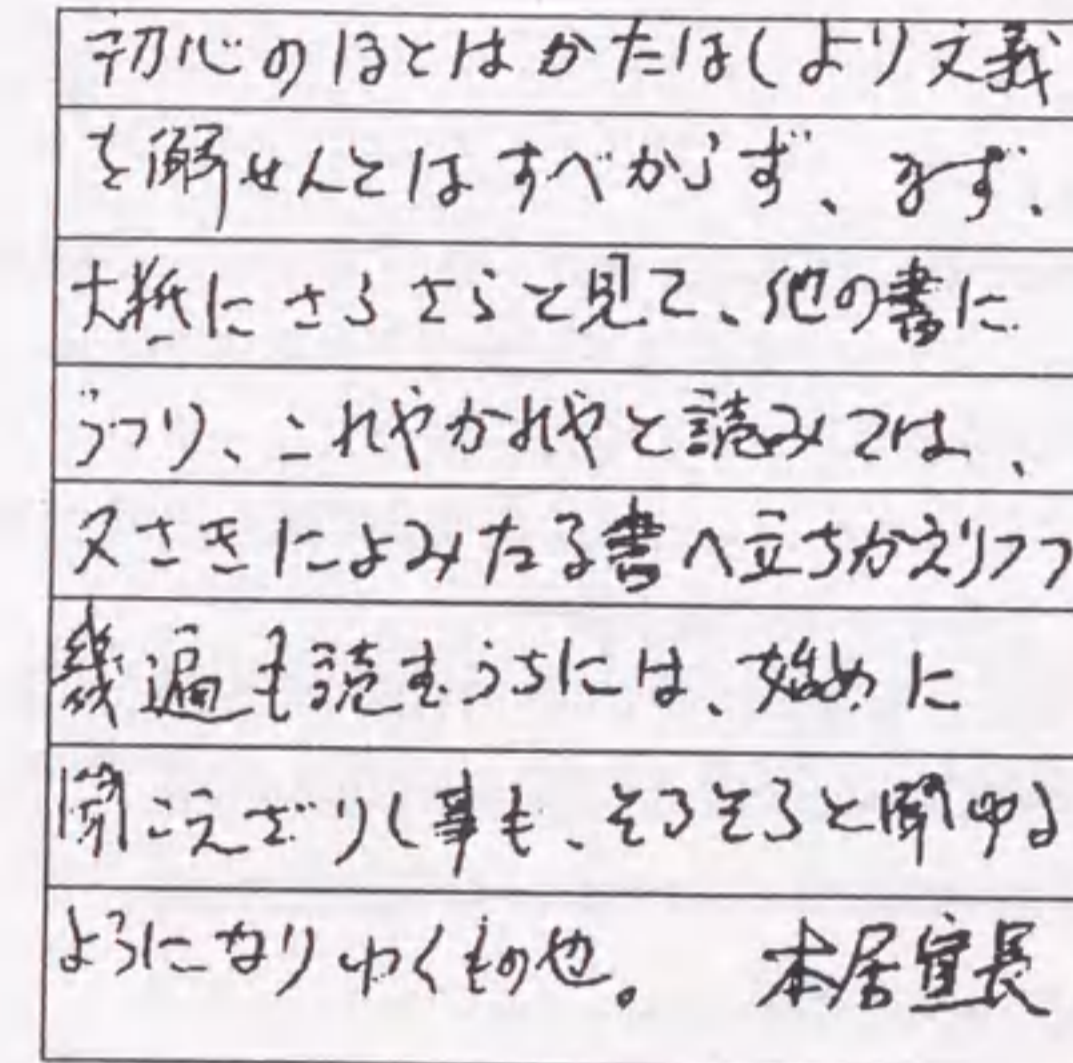


図4.10 用いた手書き文書の例

(1) 実験条件と方法

処理は、[1]標準文字幅の推定、[2]処理パラメータの決定、[3]射影値ネットワークの状態遷移、[4]切り出しネットワークの状態遷移、[5]切り出し位置の決定の順で行った。それぞれについて説明する。

[1] 標準文字幅の推定

まず入力される文字列の高さと幅を求める。文字列の高さは、文字列方向に射影したヒストグラムにおいてしきい値以上のヒストグラム値を持つ区間の長さとした。また文字列幅は、文字列と垂直方向に射影したヒストグラムにおいて、しきい値以上のヒストグラム値を持つ左右端座標間の長さとした。標準文字幅は、文字列高さ

に一定値Wを掛けた値とした。

[2] 処理パラメータの決定

次に、この標準文字幅を用いて各パラメータの値を決定する。計算量の削減のために、射影値ネットワーク、切り出しネットワークにおいて文字列高さの10分の1の幅につき1個のニューロンを配置することとし、それに基づき以下のようにパラメータを決定した。

・ E_s におけるパラメータ

A: 射影値ネットワークのニューロン1個が受け持つ範囲内の射影値総和の最大値の4倍

$\alpha_1 = 2, \alpha_2 = 4$

・ E_i におけるパラメータ

β : 標準文字幅に対応するニューロン数

抑制関数 $C(x) = (x = \beta$ のとき 0, $x = 1$ のとき 900)

・ E_n におけるパラメータ

N: 文字列幅に対応するニューロン数を β で割った値

・ E_s における各エネルギー項の重み係数

各エネルギー項がとる値の次数が同じ程度になるように実験的に決めた一定の値とする。本実験では以下のように値を設定した。

$C_1 = 1 \times 10^3, C_n = 5 \times 10^4, C_v = 10.0, C_g = 5.0$

[3] 射影値ネットワークの状態遷移

文字列の射影ヒストグラムにおいて、各ニューロンが受け持つ範囲における射影値の総和をそのニューロンに与える初期値とした。次に、式(4.15)で表されるホップフィールド型ニューラルネットワークの状態遷移式に従って一定回数状態遷移を繰り返し、射影値ネットワークのニューロンの出力値を決定する。

$$\Delta F_i / \Delta t = -\partial E_p / \partial F_i = E'_{pi} \quad (4.15)$$

$$F_i(t+1) = F_i(t) + \varepsilon_p \cdot E'_{pi}$$

但し、 $F_i(t)$ は時刻 t での i 番目のニューロンの出力値である。また ε_p は十分に小さな値である。

[4] 切り出しネットワークの状態遷移

ニューロンの初期値は、文字列両端に対応するニューロンには1、その他のニューロンには0.5を与えた。射影値ネットワークのニューロンの出力値によって決まる E_s に基づいて、式(4.16)で表される切り出しネットワークの状態遷移を十分な回数を行う。但し、文字列両端に対応するニューロンは常に1の出力を出すものとした。

$$\Delta m_i / \Delta t = -\partial E_s / \partial L_i = E'_{ki} \quad (4.16)$$

$$m_i(t+1) = m_i(t) + \varepsilon_s \cdot E'_{ki}$$

但し、 $m_i(t)$ は時刻 t での i 番目のニューロンの入力総和値である。また ε_s は十分に小さな値である。

[5] 切り出し位置の決定

切り出しネットワークの状態遷移の終了後、切り出しネットワークのニューロンの出力値に対してしきい値処理を行う。しきい値より大きい出力値を持つニューロンが受け持つ範囲内において、射影値が最小値をとる座標を切り出し位置と決定した。

(2) 実験結果と考察

図4.11に状態遷移途中の切り出しネットワークのニューロンの出力値を示す。図4.11からわかるように、エネルギー関数を評価することによって、まず局所的に切り出し位置の可能性が高いところのニューロンの出力値が大きくなり、徐々に近傍のニューロンの出力値が修正され、最終的に正しい結果が得られている。

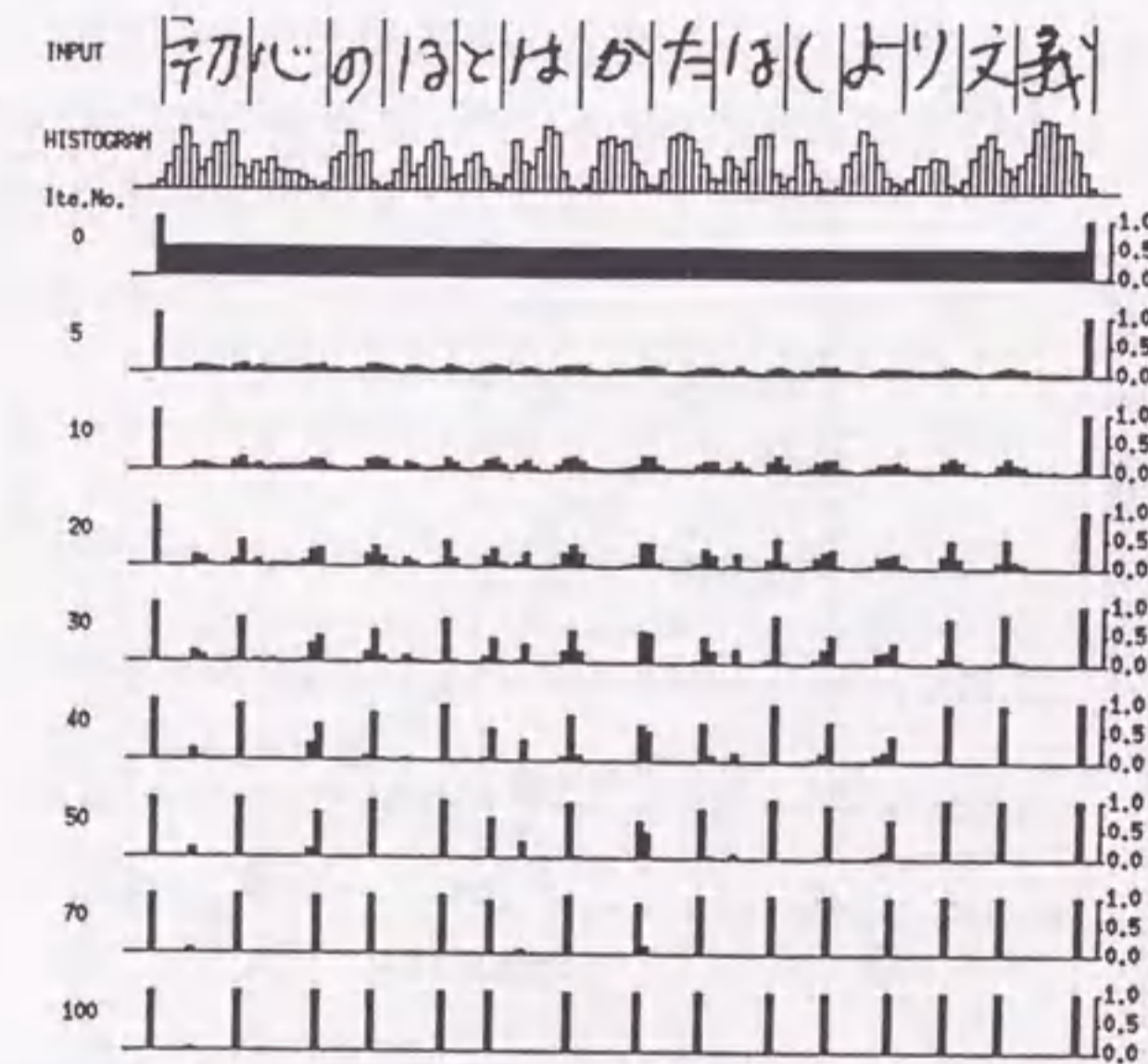


図4.11 切り出しネットワークのニューロン出力

切り出し性能を定量的に評価するために、認識情報を用いない手書き文字の切り出しの従来手法との比較を行った。この従来手法 [67] は、ラベリングによって黒画素連結成分を矩形化し、一定の文字大きさ(外接矩形面積)や正方形度の条件を用いて矩形の統合、

分割を繰り返すものである。結果を表4.2に示す。

表4.2 手書き文字列に対する正切り出し率

文字切り出しニューラルネットワーク	82.8%
従来手法	75.9%

注：7#7°ル×113文字/#7°ル=791文字

但し、標準文字幅は文字高さに0.7(=W)を掛けた値とした。従来手法と比較して約7%の精度向上が図れており、文字切り出しを行うニューラルネットワークが手書き文字列のような変動の大きい対象に対して有効であることが確認できる。

以上のように、手書き文字列のような変動の大きい対象に対して有効である結果が得られたが、その要因について、ネットワークの状態遷移、及び従来手法に対する優位性の点から考察を加え、本手法の処理限界について以下に示す。

(a) ネットワークの状態遷移

図4.12は、図4.11における状態遷移時の切り出しネットワークのエネルギー関数 E_s 及び、各エネルギー項の値の変化を示したグラフである。但し、縦軸は E_s の値に対するものであり、他の各エネルギー項の値は、それぞれの最大値で正規化してある。

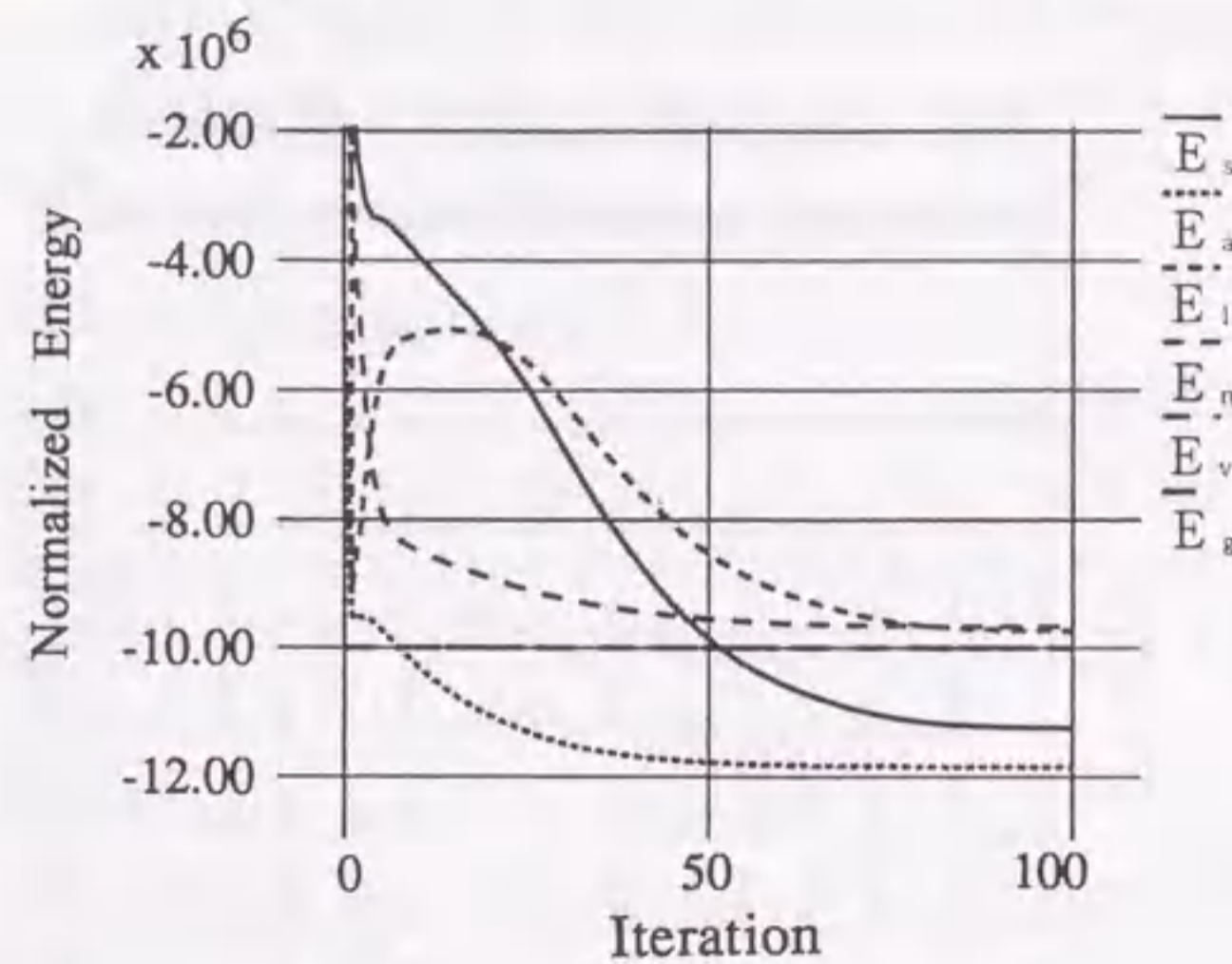


図4.12 切り出しネットワークにおけるエネルギー関数の値の変化

状態遷移直後に各エネルギー項の値が大きく減少しているのがわかる。この時のニューロンの出力値は、図4.12に示したようにすべての範囲において減少している。これは切り出しネットワークのニューロンの初期値を一律0.5としたために、各ニューロンにエネルギー項 E_l 、 E_n 、 E_v に起因する抑制が強く作用したためであると考えられる。その後、エネ

ルギー項 E_a 、 E_n に起因する活性化入力によって徐々にニューロンの出力値が大きくなって行くにつれて一端 E_l の値が増加し、次に近傍のニューロンの出力値が修正されるとともに、再び E_l の値が減少している。

このことから、主にニューロンへの活性化を図る項である E_a 、 E_n と、抑制を行う E_l との作用によりネットワークの状態遷移がうまく行われ、近傍ニューロンの相互作用によって全体的に最適な切り出し位置が確定されていくことがわかる。

(b) 従来手法に対する優位性

従来手法では、各処理が逐次的に適用されて行くために、誤りが生じるとそれが伝搬する。それ故、正しく切り出されている位置の前後の関係を加味することができない。

この点において本手法では、エネルギー関数の最小化という枠組みによってネットワークが持つエネルギーを相対的に減少させる方向に処理が進むために、基準パラメータの値が少し外れても引き込みが生じ、誤りの発生を少なくできる。しかも、切り出しのためのルール、すなわち、制約条件を並列的に適用しながら全体の状態遷移が行われるために、正しく切り出されている位置の前後の関係をも考慮した文字切り出しを行うことができ、誤りの伝搬が起こらない。

また本手法では、切り出しのための知識を微分可能な関数として表現している。このことは、筆記者の個人性を有する基準パラメータを教師なし学習によって最適化でき、各個人に適応した文字切り出しを行えることを示している。

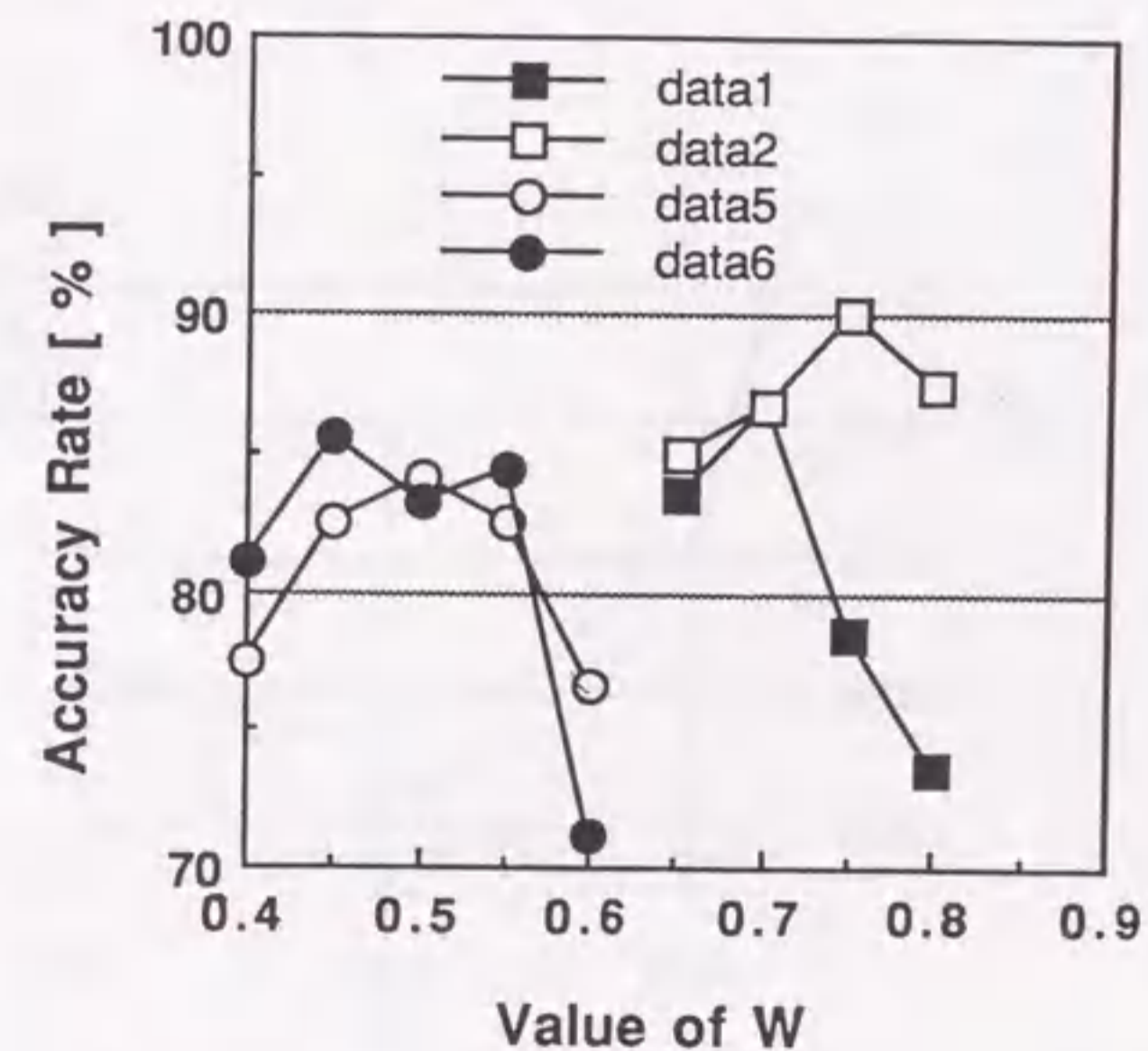


図4.13 パラメータWを変化させたときの正しく切り出された割合

本手法における標準文字幅を求める際のパラメータ W （抑制関数のパラメータ β に比例）は、筆記者が書く標準の文字の縦横比を表しており、 W の値が小さいほど縦長の文字である。従って、このパラメータ W には個人によって異なる最適値があると考えられる。図4.13にパラメータ W の値を変化させたときの各サンプル（data1, data2, data5, data6）の正切り出し率を示す。ピーク値が、縦長の文字を書く人と正方形に近い文字を書く人とに顕著に分かれているのが確認できる。

図4.14は、正しい切り出し位置を与えた場合に各パラメータ W の値に対して持つエネルギー関数 E_s の値を示したものである。 W の最適値と E_s の最小値をとる W の値とは完全に一致はしないものの、 W と E_s との間には相関がみられる。このことから W すなわち式(4.10)のパラメータ β を適当な初期値から E_s の値が減少する方向へ変化させることによって最適値を探索することが可能であると考えられる。しかも、この β の値による E_s の変化は文字切り出し処理における E_s の変化の方向と同じ方向であるので、この最適化過程は文字切り出し処理と同時に行える。

これまでも個人性を考慮した文字切り出し手法が提案されている[76]。この手法は、予め設定したパラメータを用いて切り出し位置の結果を求め、解として与えた切り出し位置との差を求めた後、パラメータの変更を行い切り出し位置を求める必要があった。しかし本手法によれば、エネルギー値の減少方向にパラメータ値を変化させることで、教師を用いることなく文字切り出し処理を行いながら、パラメータ値の最適化も同時に行えるのである。

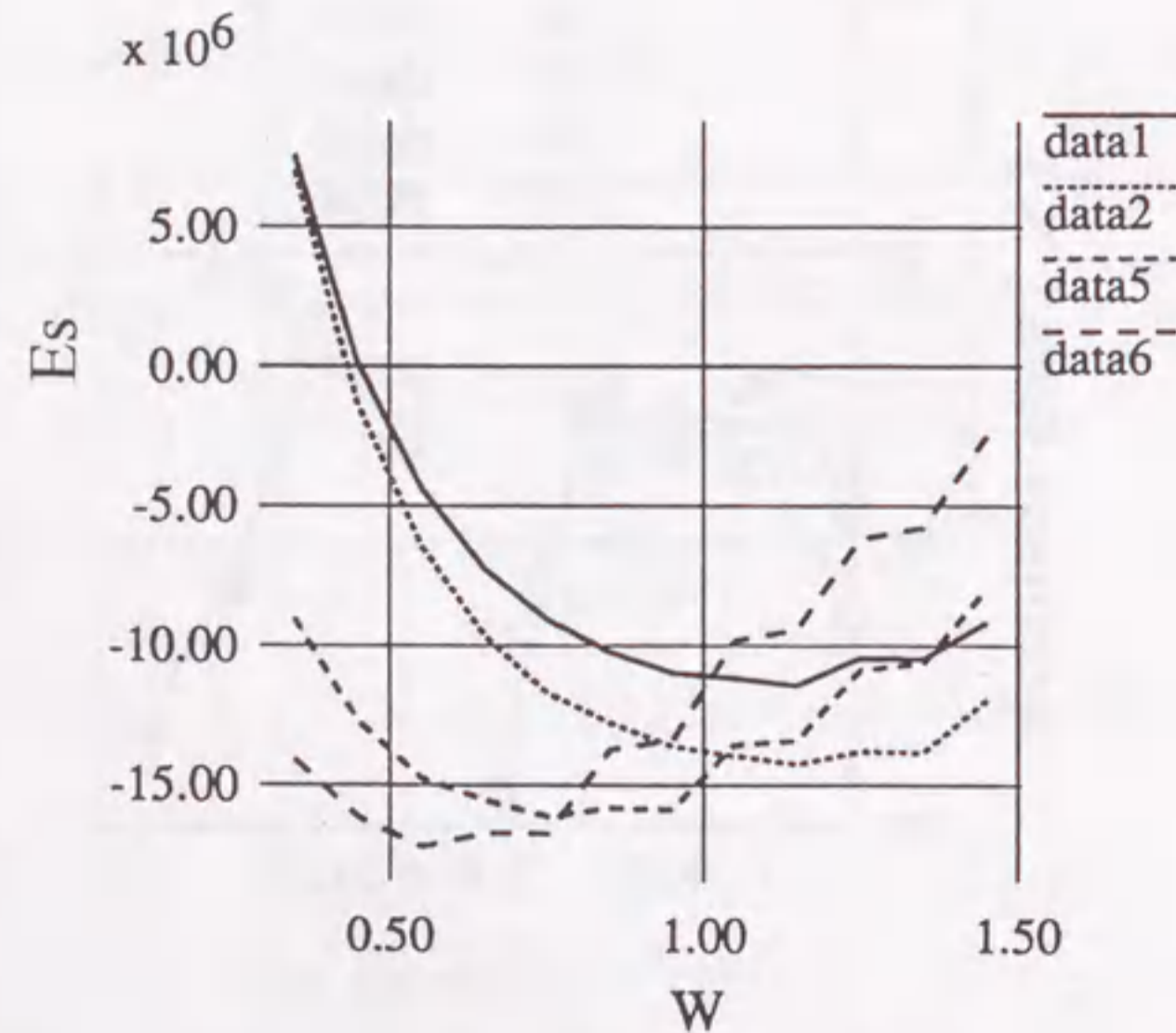


図4.14 正しい切り出し位置を与えたときのパラメータ W とエネルギー E_s の関係

(c) 本手法の処理限界

図4.15に誤った例を示す。これは、文字幅の変動が著しい場合で、文字列に垂直方向にしか分離できない射影ヒストグラムを用いたことによる限界を示している。

このような場合に対処するためには、切り出した後の文字を認識、後処理した結果の情報を加味したエネルギー関数、あるいは、2次元的に分離が可能なラベリング処理による結果の情報を加味したエネルギー関数の設定が必要であろう。これらは今後の課題である。

文字の大きさが近傍と著しく異なる 射影ヒストグラム分布を用いることによる限界

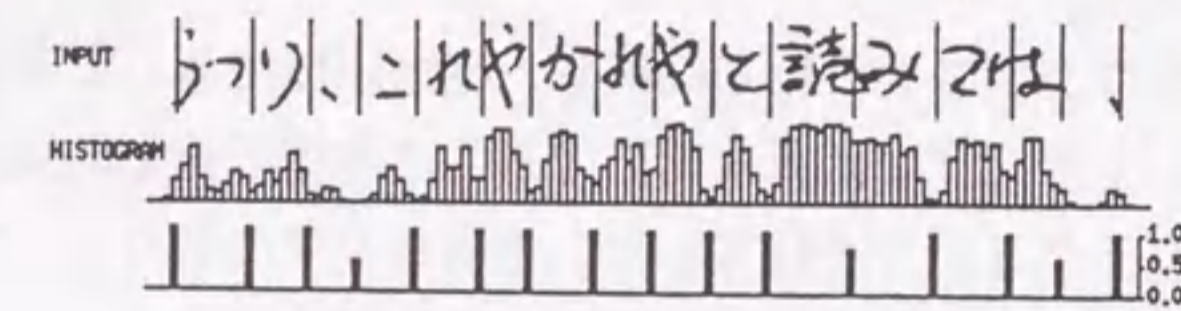


図4.15 切り出し誤りの例

4.4 まとめ

本章では、筆記者の字癖や筆記具等による字形変動が大きく、同一カテゴリでも多種多様なパターンが存在する手書き文字の認識性能を向上させるために、ファジィ推論アーキテクチャに基づく構造化ニューラルネットワークを提案した。これは、特徴空間を複数の識別容易な部分空間に大分類し、各部分空間毎に独立に入力パターンの認識を行い、それらの結果を統合判断することで複雑な識別境界を獲得する構造を有するものである。手書き数字認識に適用してその有効性を検討した結果、構造化によって認識性能が改善されることを確認し、正読率94.7%、誤り率0.36%、リジェクト率4.96%という高い認識精度を実現することができた。

更に、認識に先立つ文字の切り出しに際し、従来困難であった手書き文字の外形変動によって生じる結合、分離文字に柔軟に対応できる文字切り出しに関する知識を埋め込んだニューラルネットワークモデルを提案した。手書き文字の切り出しに適用した結果、従来よりも高い切り出し性能を得ることができた。この手法は、ノイズ除去の知識を制約条件として持つ文字列画素の射影値成分を扱う射影値ネットワークと、切り出しのための知識を制約条件として持つ切り出しネットワークとを各々ホップフィールドネットワークで構成し、それらを階層的に結合した2層構造のネットワークである。それ故、並列演算による処理が可能となり、文字パターンの変形が存在してもその近傍条件を考慮した切り出し位置を推定することができる。更に切り出し条件は、少ないパラメータの設定だけで済むので、種々の文字パターンの混在があっても対応が可能である。

第5章

パターンと記号との統合化処理による パターン認識

5.1 はじめに

文字認識の研究の歴史はパターン認識の研究の中で最も長く続いている。しかし、文字認識の実用化は極く限られた領域でしか実現されていない[5]。文字認識処理は、文字切り出し、特徴抽出、文字識別、後処理から構成されているが、従来の文字認識の研究では、特徴抽出と文字識別の過程に力点を置き、認識率を向上させようという取り組みが行われていた。この部分にニューラルネットワークを導入するのもその一つである。しかし、認識対象がOCR帳票から一般文書に拡大するに従い、自由書式や手書き文字を処理するために、適応的な文字切り出しや後処理が重要になっている。後処理の部分が注目されつつあるが、研究の歴史が浅く、その成果がまだ十分利用できる段階でない現状である。また、今までの研究は、それぞれの処理に対して個別的に進められており、各処理を有機的に結合して研究することは行われていない。それ故、総合性能は最も性能が低い処理過程で制限されてしまうこととなっている。特に、日本語情報を扱う場合には、文字種が多く、類似文字も多い。それ故、日本語の文字認識では、認識対象が低品質文字や手書き文字のとき、文字認識処理は1文字ごとであったり、上述の各処理が結合されていても個別に動作する処理であるので、認識率向上に限界がある。しかし、近年コンピュータ能力の向上により、言語知識の積極的利用や学習による判断ルールを構築できるニューラルネットワークの利用が可能となってきており、各処理の性能を向上させる素地は整いつつあるとともに、各処理の有機的結合が必要不可欠となっている。

人が文書を読むときは、脳の中の各処理が相互に判断結果をやり取りしながら認識を行っている。具体的には、イメージ情報から文字領域を選択的に注視し、特徴データを抽出し、文字として認識し、文字列のつながりから単語や意味、概念に結びつける処理を階層

的に行っている。そして、再度イメージ情報に戻って再認する動作を繰り返すことで文書を読んでいる。それ故、文字の大小や変形、かすれ、脱字があっても正しく認識することができる。そこで、このような各処理の結果を有機的に結合して判断を行う認識処理が重要となり、これをどのようにコンピュータ上で実現するかが大きな課題である。

文字認識処理は、文書画像から1文字ごとの文字領域を切り出す前処理部、文字領域のパターン情報から特徴量を抽出し認識辞書とのマッチングにより認識を行う認識部、語彙や文法などの知識を用いて認識結果の妥当性を調べこれを修正する後処理部からなる。従来、低品質な文書に対応するために、文字認識に上位の知識である言語情報を利用することにより認識精度の向上を図ってきた[82][85]。しかし、認識部の出力だけを用いて後処理部で文字訂正処理を行った場合、前処理における切り出し誤りなどのパターン処理の誤りを訂正することは困難である。そこで、前処理部での文字切り出しにおいて複数の文字切り出し候補を出力し、これらの候補の中から最適な認識結果を得る手法が提案されている[85][94][95][96][97]。しかし、これらの手法はすべての候補について認識処理、言語処理を行う必要があるため、計算量が非常に多くなる。特に、低品質の文書に対しては候補の組み合わせの数が膨大となる。

本章では、人のように機械が文書を読めるを目指して、前処理部と認識部をニューラルネットワークを用いたパターン処理で、後処理部を言語知識を用いた記号処理で構成し、それぞれの処理過程間で相互の情報を交換することにより、総合的に判断を行うパターンと記号の統合化処理法C o P S (Cooperation of Pattern and Symbolic processing) を提案する。C o P Sでは、認識部や後処理部でパターン処理の誤り位置を検出し、その情報を前処理部や認識部にフィードバックする。従って、誤り部分のみ再処理するために計算量は少なく済む。また、誤り部分がなくなるまで各処理過程間で相互に情報を交換するために、各処理過程の総合的な判断に基づいて最終的な認識結果を出力できる。

以下、5.2節でパターンと記号の統合化処理による文字認識システムの構成を示し、各処理部の処理方法について述べる[106]。5.3節では認識実験結果を示しその有効性を明らかにし、5.4節でまとめを行う。

5.2 パターンと記号の統合化処理 (C o P S) による文字認識システム

脳では、パターン処理と記号処理がうまく融合されて知的活動を行っていることが知られている。イメージ情報から文字領域を選択的に注視し、特徴データを抽出、それを文字として認識し、文字列のつながりから単語や意味、概念に結びつける処理を階層的に行っている。そして、再度イメージ情報に戻って再認する動作を繰り返すことで、ノイズのある文書イメージに対しても非常に安定した高度な認識処理を行っている。

提案する文字認識システムは、このような人の行っている処理を積極的に模倣し、パターン処理と記号処理を融合させること(C o P S)により、人間が行っている柔軟な処理を実現する。C o P Sではニューラルネットワークによるパターン処理(より低次元処理)と言語知識による記号処理(より高次元処理)とを、階層的に分散、協調して動作させる。図5.1にC o P Sの概念図を示す。各処理間で情報や知識の交換を行うことにより、最終的な認識精度を向上させることができる。

そこで本節では、C o P Sによる文字認識システムの構成を示し、各処理過程について述べる。

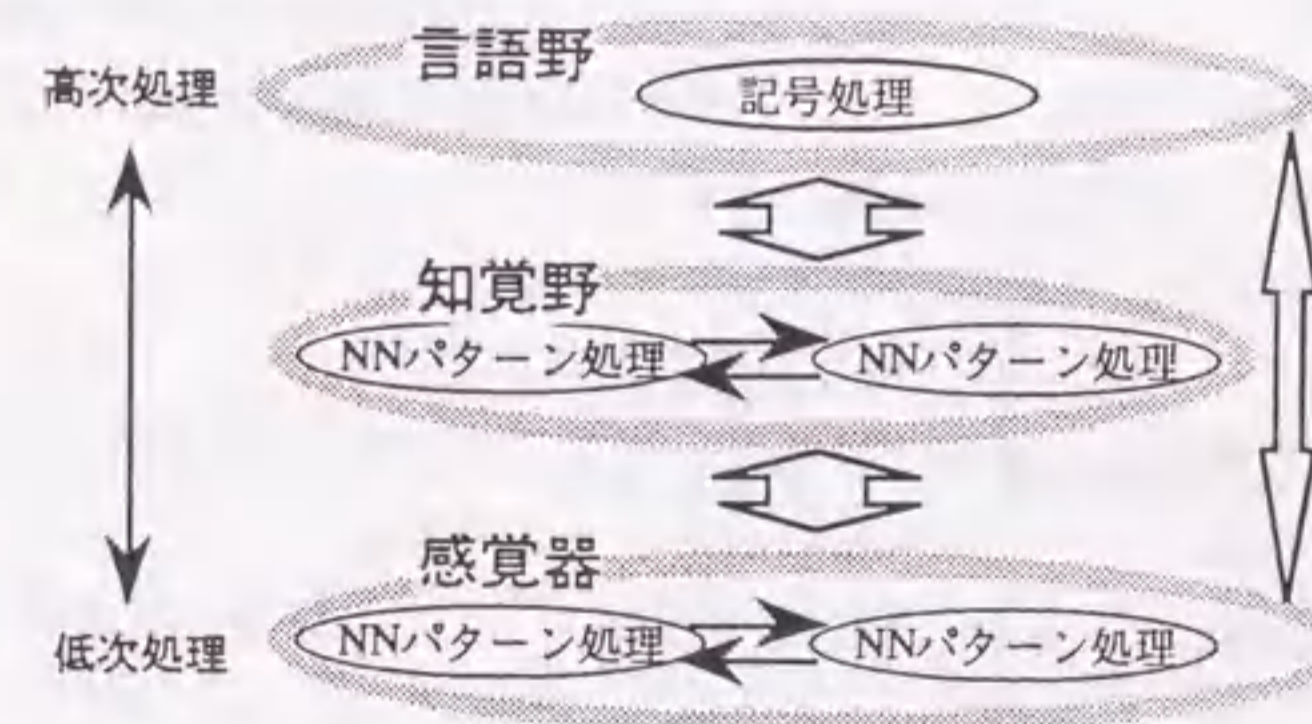


図5.1 パターンと記号の統合化処理の概念図

5.2.1 文字認識システムの全体構成

文字認識処理は、一般に文書画像から1文字ごとの文字領域を切り出す前処理部、文字領域のパターン情報から特徴量を抽出し認識辞書とのマッチングにより認識を行う認識部、語彙や文法などの知識を用いて認識結果の妥当性を調べてこれを修正する後処理部から構成される。前処理部、認識部は、パターン情報を直接的に処理する部分であるので、パタ

ーンの変動に対して対応が容易なニューラルネットワークを用いる処理が効果的である。一方、後処理部は、言語情報である単語や文法の知識を扱うので論理的な記号処理が効果的である。また、各処理がバランスして協調動作を行うためには、誤りを出力した処理に対してその判断基準を容易に変更し判断精度を上げていく仕組みが必要である。このような考えに基づく構成が図5.2のシステムである。

前処理部は、ホップフィールド型ニューラルネットワークを用いて切り出し処理を行う。前処理部では、まず、文字を全角であると仮定し、一次切り出し処理を行い、認識部と後処理部へ、その結果の情報を送る。そして、認識部、後処理部からのリジェクト情報を受けて再切り出し処理を行う。このとき、後処理部で得られる文字種の情報も用いて、切り出し制御を行う。

認識部は、構造化ニューラルネットワークを用いた認識処理を行う。認識部では、まず、全文字種を対象とした認識を行う。そして、文字の矩形、位置情報を用いた認識後処理を行い、リジェクト情報を前処理部へフィードバックする。また、認識結果の情報を後処理部に送る。さらに、認識部は、後処理部からの誤認識フィードバック情報と文字種の推定情報に基づいて、再認識処理を行う。

後処理部は、文法、単語に関する知識に基づき認識誤りの修正処理を行う。このとき、認識結果情報から得る単語の信頼度により、誤認識単語のリジェクトを行い、その情報を前処理部、認識部へフィードバックする。また、文字種の推定結果も併せて前処理部、認識部へフィードバックする。リジェクトがなくなるか、認識結果の向上がないと判断されるまで、フィードバックを繰り返す。

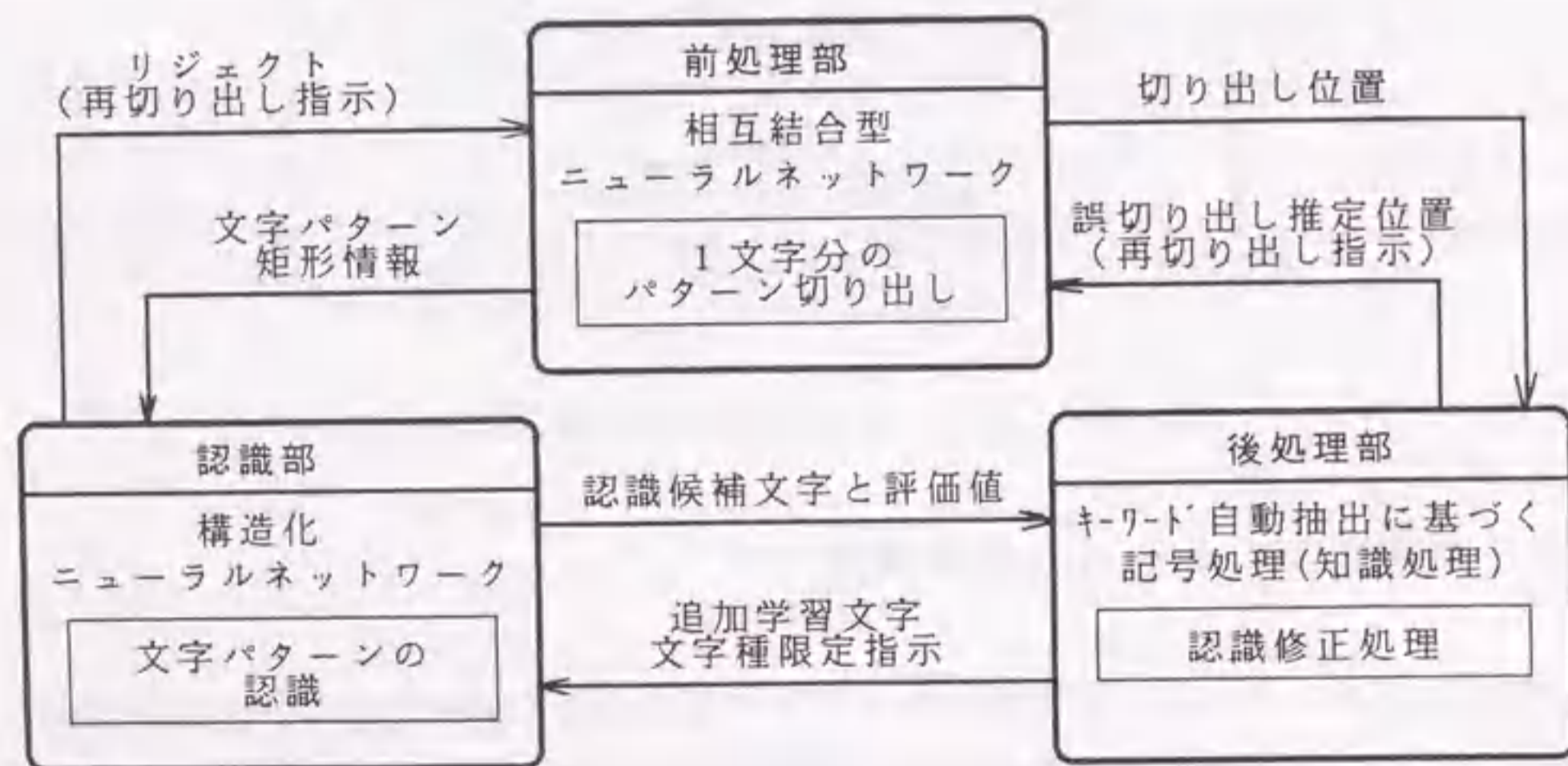


図5.2 パターンと記号の統合化処理による文字認識システムの構成

本システムでは、前処理部で切り出し条件のパラメータ（文字ピッチ）、認識部で認識候補文字の類似度、後処理部で文節の言語的正確度という情報により、各処理部間の制御を行っている。このような情報を用いることにより、各処理間の相互の情報交換が可能となった。

5.2.2 前処理部

前処理部は、スキャナから入力した文書画像より1文字として認識すべき領域を切り出す処理を行う。本システムでは、認識部、後処理部からの切り出し誤りの検出情報がある場合に、その情報に基づき切り出し処理のパラメータを切り換えて再切り出し処理を行う。そのために、切り出しのパラメータを速やかに調整する必要がある。そこで、本前処理部では、制御パラメータを少なくでき、その値の設定も容易に行える、4.3節で示したホップフィールド型のニューラルネットワークを処理に用いている[70]。これにより、全体情報を反映した局所的パラメータ変更が可能となり、全角、半角文字混在文書や手書き文書に対しても正確な切り出しが可能となる。

(1) 前処理部の構成

文字切り出しは、文字の黒画素濃度の射影ヒストグラムを基に処理を行う。文字切り出し処理を行うニューラルネットワークは第4章の図4.7に示すように、文字列方向の座標位置に対応するニューロンで構成し、文字列画素の射影成分 $F(i)$ の平滑化を行う射影値ネットワークと、切り出し位置 $L(i)$ の推定を行う切り出しネットワークからなる。射影値ネットワークでは雑音除去、切り出しネットワークでは切り出しのための知識をエネルギー関数として持つ。そしてこのニューラルネットワークがエネルギー関数を最小化したときに、しきい値以上の値を持つ $L(i)$ の位置を切り出し位置とする。このエネルギー関数のパラメータ設定により文字切り出し条件の変更が行える。

切り出しネットワークのエネルギー関数を再度以下に示す。

(a) 射影値が小さく、射影分布が下に凸な部分に切り出し位置がある可能性が強いことから、次式を与える。

$$E_a = \sum_i \left\{ (1 - L_i) \sum_{j=0}^{\alpha_1} (A - F_{i-j} - F_{i+j}) \times \sum_{k=1}^{\alpha_2} (F_{i-k} + F_{i+k} - 2k \cdot F_i) \right\} \quad (5.1)$$

但し、 A は射影値の最大値によって決定される値、 j, k は $0 \leq j \leq \alpha_1, 1 \leq k \leq \alpha_2$ 、

α_1, α_2 は射影値を考慮する範囲であり、文字列高さから求められる標準文字幅によって決定される。

(b) 切り出し位置はある一定の幅で分布しやすいことから、次式を与える。

$$E_1 = \sum_i L_i \sum_{j=1}^{\beta} C(j) (L_{i-j} + L_{i+j}) \quad (5.2)$$

但し、 $C(j)$ は j に関する単調減少関数で、ニューロンの発火を抑制する範囲とその強度を表したもので、図4.9にその例を示した。 j は、 $1 \leq j \leq \beta$ で、 β の値によって文字幅の条件を切り換えることができる。 β の値が小さいと切り出し位置の幅は狭く、また大きいと切り出し位置の幅も広がる可能性が大きくなる。

(c) 文字の数は文字列の高さで決まる値から推定される値に近いことから、次式を与える。

$$E_n = (\sum_i L_i - N)^2 \quad (5.3)$$

但し、 N は文字の推定数である。

(d) ニューロンの出力値に対する制約条件として、次式を与える。

$$E_v = \sum_i L_i (1 - L_i) \quad (5.4)$$

(e) ニューロンへの入力に対する制約条件とその制御として、次式を与える。

$$E_g = \sum_i \int_0^{L_i} g^{-1}(L_i) dL_i \quad (5.5)$$

$$L_i = g(m_i) = 1 / (1 + \exp(-2\lambda m_i))$$

但し、 m_i はニューロンの入力総和、 λ は定数である。

以上の各エネルギー項がを重み積和したものが、切り出しに際して最小化すべきエネルギー関数 E_s となる。

$$E_s = E_a + C_1 \cdot E_1 + C_n \cdot E_n + C_v \cdot E_v + C_g \cdot E_g \quad (5.6)$$

(2) 切り出し処理とそのパラメータ制御

切り出し処理は、一次切り出し処理とフィードバック情報に基づくパラメータ制御による再切り出し処理からなる。

(a) 一次切り出し処理

初めに、文字列中の文字をすべて全角文字と仮定して切り出し処理を行う。 $L(i)$ の初期値は、射影成分の分布からみて、切り出し位置の可能性が高い位置と可能性が低い位置を求め、その位置に基づき決定する。そして、ニューラルネットワークにより一次切り出し位置を求め、その結果の情報を認識部に出力する。

(b) フィードバック情報に基づくパラメータ制御

切り出し処理で、半角文字を全角文字と推定した場合には、切り出し誤りが生じる。切り出し誤りが起こっているか否かは認識部、後処理部で判断され、前処理部にその情報がフィードバックされる。前処理部は、切り出し誤りのある文字領域情報を受け取り、その領域に対して再切り出し処理を行う。

再切り出し処理では、エネルギー関数 E_s の式(5.6)における切り出し位置の分布を決めるエネルギー E_1 の式(5.2)のパラメータ β の値を複数設定し、切り出し処理を行う。もし、複数の処理の切り出し結果が異なる場合は、それらの切り出し位置を認識部に送り、認識処理の結果、評価値が最も高くなった切り出し位置を再切り出し位置として用いる。

一方、切り出し誤りが起こっている領域の文字種がアルファベットであると後処理部で判断された場合は、以下の再切り出し処理を行う。不定ピッチ英文字では、文字ピッチに関する性質よりも射影成分のヒストグラムを基に切り出し位置を求める方が望ましい。よって、正切り出し位置のほとんどは射影成分の空白部分に相当する下に凸の部分である。従って、上述した定ピッチ切り出しに関するエネルギー E_1 の式(5.2)のコストを下げるために C_1 の値を小さくし、切り出し位置の特定に寄与するエネルギー E_s の式(5.1)のコストを相対的に上げ、切り出し処理を行い、再切り出し位置を求める。

このようなフィードバック情報によるパラメータの値の制御を行うことにより、切り出し誤りの部分をなくし、正確な切り出し位置を求めることが可能となる。しかも、制御すべきパラメータは β, C_1 程度である。更に、エネルギー関数の E_s の最適化により値を決定するので、その値設定は大まかであっても、切り出し位置の推定に与える影響は少ない。

5.2.3 認識部

認識部は、前処理部から送られてきた1文字分のパターン情報から特徴抽出を行い、各文字を認識する。本システムでは、文字認識結果と前処理部から送られた文字の矩形情報から、認識文字のリジェクト処理を行い、リジェクトした文字に対しては前処理部へ誤切り出しの情報をフィードバックする。また、後処理部からフィードバックされる文字種情報に従って、ニューラルネットワークの結合係数を切り替える。この操作は、認識のための認識辞書の切り替えを意味している。これにより、文字種に最適な認識を行うことができるため認識率が向上する。本システムでは、認識判断基準を容易に変更できる学習性を考慮して、認識部を構造化ニューラルネットワーク [98] [101]を用いて実現している。このニューラルネットワークは、4.2節で手書き数字認識に用いたものを漢字をも含めた大カテゴリーの認識を可能にするように拡張したものである。

(1) 認識部の構成とその制御

図5.3に本認識部の構成を示す。NARAは、大分類部、再分類部、統合部の三つの部分から構成される。大分類部では、入力データの特徴空間をファジィクラスタリングし、細分類部では、分割した各特徴空間毎に入力データと細分類部から最終的な認識を行い、候補文字とその類似度を出力する。さらに、本システムの認識部は、認識精度を上げるために、文字種の限定制御とリジェクト処理を行っている。

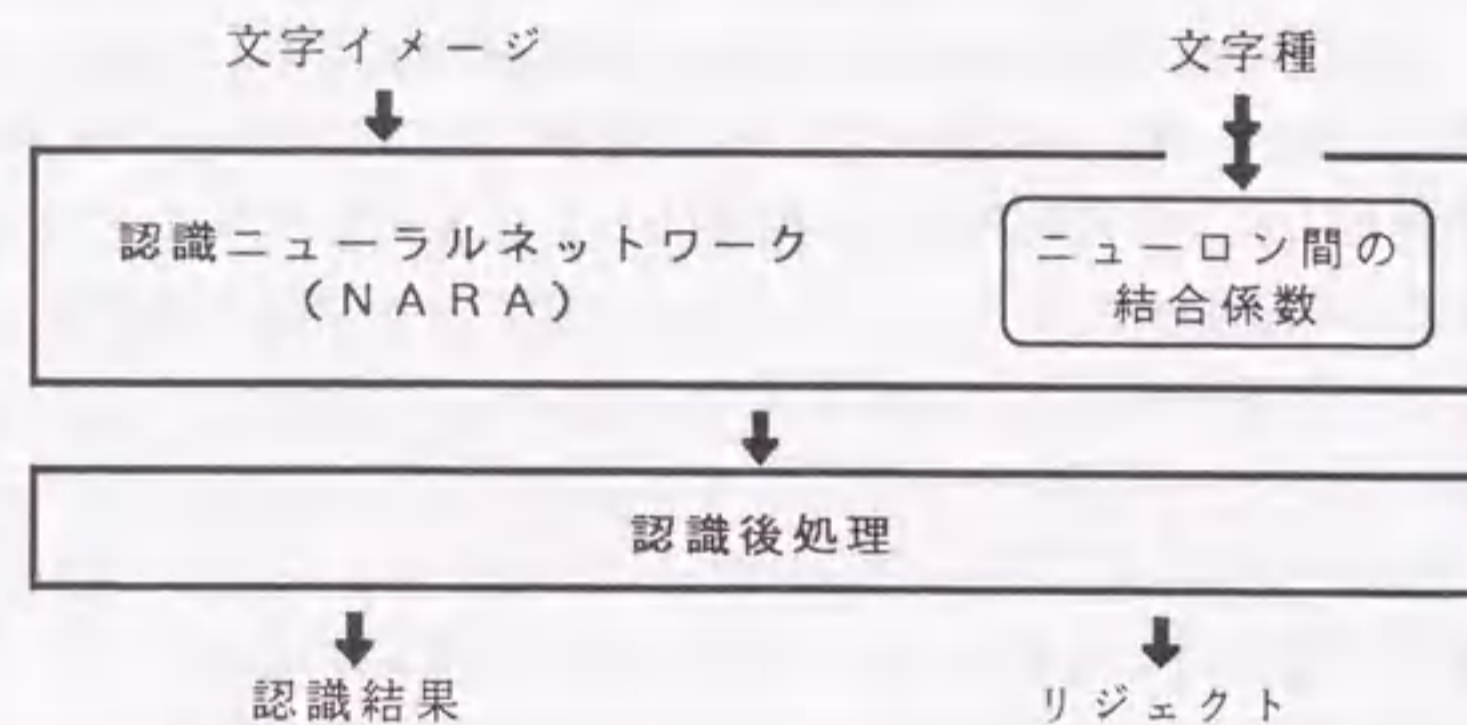


図5.3 認識部の構成

(a) 文字種の限定による認識

限定された文字種を対象とする認識では、すべての文字を対象とする認識に比べ認識率が向上する。特に、漢字、アルファベット、数字のように文字の構造が異なるものについては、それぞれの文字種に応じた認識パラメータを用いて認識を行った方が認識精度が上がる。そこで、本認識部では、「数字」、「アルファベット」、「片仮名」の文字種において、後処理部からの誤認識フィードバック情報と文字種推定情報があった文字に対して、通常的全文字種を対象としたニューラルネットワークの結合係数を、限定文字種を対象とするニューラルネットワークの結合係数に切り替えて認識処理を行う。これにより、各文字種に最適なニューラルネットワークの結合係数を用いるので認識精度が向上する。

表5.1に全文字種を対象とした結合係数と限定文字種に対する結合係数を用いた認識処理の実験結果を示す。文字種を限定することにより3.0%の認識精度向上が得られた。

(b) 認識後処理部

特徴量抽出時の正規化処理によって、例えば、縦長、横長などの形状を特徴とする文字（I、1、一、一、句読点など）は正規化され、特徴量だけでは正確な認識は行うことができない。そこで、切り出し部から送られる文字の矩形及び位置情報も用いて、正確な認識を行う。

このとき、文字の矩形、位置情報に一致する認識候補文字がないときは、切り出し誤りの可能性が大きいので、その文字をリジェクトし、前処理部に誤切り出しのフィードバック情報を返す。この処理により、文字単位で切り出し誤りと分かる部分についての判断を行うことができる。

表5.1 文字種の限定による認識

文字種	全文字種 認識率(%)	限定文字種 認識率(%)
数字	92.5	99.5
アルファベット	96.9	99.2
片仮名	93.8	96.7

5.2.4 後処理部

後処理部は、認識部から送られてきた候補文字から、語彙や文法などの言語知識及び文書内容の知識を用いて文字の修正を行う[104]。しかし、前処理部における切り出し誤りのようなパターン情報の誤りについては修正を行うことは困難である。そこで、推定した誤切り出し位置や修正不可能な誤認識文字を、前処理部及び認識部にフィードバックし、再切り出し、再認識の指示を出す。そして、再切り出し、再認識の結果をもとに再度文字の修正を行い、最終的にすべての文字が正しいと推定されるまでフィードバックを繰り返す。本システムでは、切り出し誤りを推定するために、単語の言語的正確さを評価する単語の不信頼度を定義し、その値を用いて各処理過程間の制御を行う。

(1) 誤訂正文字の推定

切り出し誤りや認識誤りがあると、後処理部では誤訂正が発生する。一般に、後処理部でのリジェクトの対象となる誤訂正の主な要因には次のものがある。

- ・文書画像からの文字の切り出し誤り
- ・候補外文字（候補外文字とは認識部の出力において候補の文字の中に正解が含まれない文字を表す。）
- ・後処理で用いる単語辞書の未登録語

これらの文字や単語が存在する場合には、その文字を含む文節において正解の単語を検索することができないので、隣接する文字を含めて誤訂正し認識率が低下する。そこで、この誤訂正の文字を推定し、前処理部及び認識部に誤り情報をフィードバックする。

誤訂正を推定するために、訂正を行った単語に対してその単語の確からしさを信頼度として求め、信頼度の低い単語を誤訂正文字と推定する。

単語の信頼度は、訂正された単語が正解であったかどうかという実際の統計データに基づいて定義した。

(a) 訂正単語の正解率

科学分野の書籍及び一般の書籍から約20件のデータ(約13,000文字)を使用して後処理を行い、後処理において訂正された単語の正解率を分析した。なお、これらのデータの認識部での第1候補認識率は77.5%~97.5%で、平均は89.1%である。表5.2に訂正された単語の正解率を示す。

表5.2 訂正単語の正解率

単語の文字数	訂正文字数	文字種類	正解率(%)
3	1~3	漢字, かな	91.5
2	2	漢字+漢字	50.9
		かな+かな	9.1
		漢字+かな	0.0
1	1	漢字+漢字	89.9
		かな+かな	58.0
		漢字+かな	50.6
1	1	漢字	14.9
		かな	28.7

訂正された単語は、以下の要素によって分類されている。

- ・ 単語を構成する文字の数
- ・ 訂正された文字(認識部で第1候補文字でなかった文字を正解と推定した文字)の数
- ・ 単語を構成する文字の種類

但し、単語を構成する文字の数が3文字の場合は、訂正文字数及び文字種類による区別はしていない。表中の正解率は、単語が訂正された結果正解になった割合を示しており、正解率が低い項目ほど誤訂正が多く生じている。

(b) 単語の不信頼度

単語wの不信頼度 C_w は、単語種類別の正解率と文字評価値及び当該単語の前後の単語の信頼度から決定されるとの考えの下に、次のように定義する。まず、各単語のペナルティ P_w を式(5.4)と定義する。

$$P_w = \begin{cases} \{100 - (\text{表5.2における単語種類別の正解率})\} \\ \times (\text{第1候補文字と訂正文字との文字評価値の差}) \\ \text{単語 } w \neq 1 \text{ 文字文節} \\ \\ \{100 - (\text{表5.2における単語種類別の正解率})\} \\ \times (\text{第1候補文字と訂正文字との文字評価値の差}) \\ + c \\ \text{単語 } w = 1 \text{ 文字文節} \end{cases} \quad (5.4)$$

但し、類似度は認識部のニューラルネットワークの出力をその最大値で正規化したものである。ここで、単語が1文字の文節を構成していれば、誤訂正の確率が高いので P_w に c (c は定数)を加えている。さらに、正しい単語を検索できない場合には、前後に隣接する単語も誤訂正する確率が高いことを考慮して、単語の不信頼度を式(5.5)によって計算する。

$$C_w = (P_w + P_{wf} + P_{wb}) \times 0.5 \quad (5.5)$$

P_{wf} : 前の単語のペナルティ
 P_{wb} : 後の単語のペナルティ

単語の不信頼度 C_w があるしきい値を超える場合に、その単語に含まれる文字を誤訂正文字とする。

(c) リジェクト処理実験

文書データ(約13,000文字)に対して、誤訂正文字の推定処理を行った。その結果、リジェクト率3%の時に、誤認識した文字の30%をリジェクトできた。リジェクトした例を表5.3に示す。但し、訂正文字列における"/"は単語の区切りを示す。

表5.3で、Aは文字の切り出し誤りの例、Bは候補外文字の例("磁"が候補外文字)、Cは単語辞書の未登録語の例である。これらの例では不信頼度が100以上の単語を誤訂正文字と推定している。このように、単語の不信頼度を導入して単語のリジェクトを行うこと

により、切り出し誤り、候補外文字、未登録語などによる誤訂正箇所を推測することが可能となった。この誤訂正箇所の情報を前処理部、認識部にフィードバックすることにより、再切り出し処理、再認識処理が可能になり、認識精度を向上させることができる。

表5.3 訂正単語のリジェクト例

	正解文字列	第1候補文字	訂正文字列 (単語不信頼度)	リジェクト文字
A	1000字	n m字	刈/皿/字 (127/170/127)	刈皿字
B	正確な	正磁な	5/枚/だ (140/119/104)	も枚だ
C	ユーラシヤ 大陸に	ユーラシヤ 大陸に	二/-/ラン (140/148/137) /7/人/陸に (143/112/37/0)	ニ-ラン 早大陸に

(2) 文字種の推定

認識部の認識誤り文字に対して、その文字の文字種を推定し、推定結果を認識部にフィードバックする。誤訂正文字の推定処理においてリジェクトされた文字を対象に、以下のように文字種の推定を行う。後処理部では言語処理として、形態素解析を行っている。この形態素解析の結果得られた各単語の品詞及び文字の種類などの情報から文字種に関する知識を用いて各文字の文字種の推定を行う。文字種としては、数字、アルファベット、片仮名の推定を行っており、文字種に関する知識として、例えば、「数字と助数詞（個、本など）に挟まれた文字は数字」、「アルファベットに挟まれた文字はアルファベット」などの知識を用いている。文字種が推定できた文字に対して、文字位置と文字種の情報を認識部にフィードバックする。

5.2.5 文字認識全体の制御

文字認識は、以下の手順で行う。

- [1] 認識対象画像から、前処理部において一次切り出し処理を行う。
- [2] 認識部において全文字種を対象とした1文字ごとの認識を行い、認識後処理でリジ

ェクトした文字の情報を前処理部にフィードバックする。

- [3] 認識部からのフィードバック情報に基づき、前処理部でパラメータ制御を行い、再切り出しをする。
- [4] 認識部において、認識後処理でのリジェクトがなくなるまで、[2]と[3]を繰り返す。
- [5] 後処理部において、文字修正を行い、文字種推定の情報を前処理部と認識部にフィードバックする。
- [6] 後処理部からのフィードバック情報に基づき、パラメータ制御を行い再切り出しをする。
- [7] 後処理部からのフィードバック情報に基づき、認識部で文字種を限定した認識を行い、認識後処理でリジェクトされた文字の情報を前処理部にフィードバックする。
- [8] 認識部からのフィードバック情報に基づき、前処理部でパラメータ制御を行い再切り出し処理をする。
- [9] 認識部において、認識後処理でのリジェクトがなくなるまで、[7]と[8]を繰り返す。
- [10] 後処理部において、文字修正を行い、リジェクト情報と文字種推定の情報を前処理部と認識部にフィードバックする。
- [11] 後処理において、リジェクトがなくなるか、あるいは単語の不信頼度が低下しなくなるまで、[6]から[10]を繰り返す。

後処理部におけるリジェクトがなくなったとき、あるいは単語の不信頼度が低下しなくなったときの後処理部の出力を最終的な文字認識結果として出力する。

5.3.2 低品質文書の認識

さらに、低品質な文書における性能を調べるための実験を行った。実験は、コピーを複数回重ねた文書原稿及びFAXで送られてきた文書原稿の低品質文書に対して文字認識を行った。実験条件は上記の実験と同様である。実験の結果を表5.5、図5.5に示す。

表5.5 低品質文書に対する認識結果

統合化処理	オリジナル原稿	コピー2回	FAX 1	コピー5回	FAX 2
あり	98.4%	96.0%	95.4%	94.6%	93.1%
なし	95.7%	93.1%	92.4%	90.5%	87.0%

実験結果から、コピーやFAXにより文書の品質が低くなるにつれて文字認識率も低下するが、統合化処理がないときに比べて以下する割合は小さい。このことは、文書の品質が低下することにより認識誤りが生じていても、総合的な認識判断を行うことにより、認識誤りをなくすことができることを示している。従って、本システムが低品質な文書に対して有効であることが実証できた。

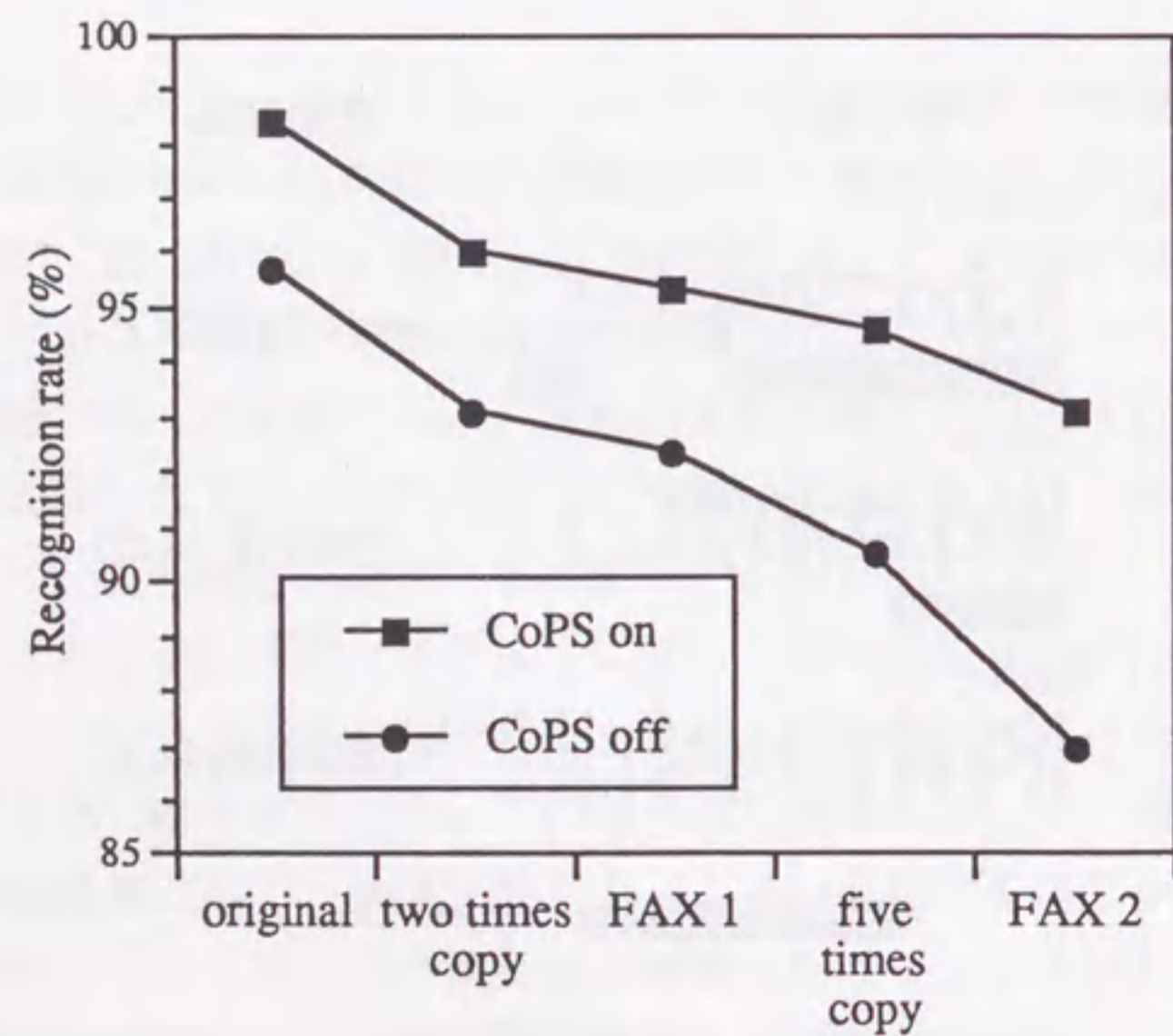


図5.5 低品質文書に対する認識結果

5.4 まとめ

文書の中に文字の大小や変形、かすれ、脱字があっても人間は正しく認識できる。本章では、この機能をコンピュータ上で実現するために、パターンと記号の統合化処理による文字認識を提案した。文字認識システムの前処理部、認識部を、ニューラルネットワークによるパターン処理で実現し、また、後処理部を言語知識を用いた記号処理で実現しただけでなく、各処理過程間で相互に情報を交換することにより総合的に認識判断することが可能になった。本システムを用いた実験結果より、統合化処理なしの場合の誤認識例の60%を、統合化処理により正確に認識できるようになった。さらに、コピーを重ねた原稿やFAX原稿などの低品質の文書においても良好な認識性能を得ることができ、その有効性を実証することができた。また、処理時間に関しても、従来手法に比べて高速に処理を行うことができるようになった。

第6章

結論

6.1 本研究のまとめ

本論文は、人間の行う学習性のあるパターン認識を模倣し、従来の手法では容易ではなかった柔軟なパターン認識を実現するために、以下の観点に関する研究成果をまとめたものである。

- ・柔軟なパターン認識に適したニューラルネットワークの学習とニューラルネットワークの構成及びその工学的実現
- ・パターン情報に基づくボトムアップのパターン処理と適応的知識に基づくトップダウンの記号処理の融合によるパターン認識

本研究では、パターン認識の対象として文字を取り上げ、人間が行っているような柔軟なパターン認識の実現を目指した。

パターン認識に適したニューラルネットワークに対する要件として、次の内容を明らかにした。パターン認識に適したニューラルネットワークの学習について、高速に行える学習法を提案し、さらにそのハードウェア実現を前提にした演算精度について明らかにした。また、柔軟性を増すための現場学習を可能にする追加学習と高速学習を備え持つニューラルネットワークモデルを提案し、学習済みの認識性能に与える影響が少なく、学習速度も高速であることを示した。そして、未学習パターンへの対応ができるニューラルネットワークとして、手書き数字の認識精度を向上させた構造化ニューラルネットワークの提案と、手書き文字の切り出しを柔軟に行う文字切り出しニューラルネットワークを提案し、その有効性を示した。

パターン処理と記号処理の融合によるパターン認識では、次の手法を提案した。パターン情報を直接に処理する部分である前処理部、認識部を、パターンの変動に対して対応が柔軟であるニューラルネットワークによるパターン処理で構成する。単語や文法などの言

語情報を論理的に処理する後処理部を、知識に基づく記号処理で構成する。そして、それらの各処理過程が、階層的に分散、協調して動作するパターン認識である。その結果、文字認識の認識精度が向上することを示した。

本論文では、以上の研究成果を以下の5章に分けて述べた。

第1章では、研究の背景、従来の研究、研究の目的について述べた。本研究では、従来の手法では容易でなかった柔軟なパターン認識を目指した。研究の対象として、文字を認識することを題材に、パターン認識に適したニューラルネットワークの学習法及び構成法と、パターン処理と記号処理が融合したパターン認識を目標とすることを述べた。

第2章では、ニューラルネットワークをパターン認識に適用するにあたり、学習の高速化と演算精度を低減する学習法を提案した。パターン認識が正しく行われるには、入力パターンに対応する出力ニューロンの出力値が他のニューロンの出力値に対して差をつけることができれば良いことに着目した。そして、学習に際して出力ニューロンのニューロンごとの誤差及び、入力パターンごとの誤差総和に応じて適応的に重み付けする評価関数を用いた学習法を実現した。これをマルチフォント英数字記号認識に適用し、学習時間の短縮と演算精度の低減を達成できることを示した。

第3章では、未学習データの判断を追加的に学習することで、システムの適応性を向上させるニューラルネットワークモデルを提案した。追加的な学習は基本的に高速でなければならないし、すでに学習済みのパターン認識の性能に影響を与えるものであってはならない。そのために、入力データの予めの多段階のしきい値処理を行う機能を有する量子化ニューロンを提案した。そして、これを判断のための入力データに応じて階層的に接続し、学習を行う最終層で統合化するニューラルネットワークを提案し、高速学習性と追加学習性を検討した。これをマルチフォント英数字記号認識に適用し、学習の高速性と、学習済みの認識への影響が従来のものに対し少ない追加学習性が得られることを示した。

第4章では、パターンの変動が大きく、同一カテゴリーでも多種多様なパターンが存在する手書き文字を柔軟に認識するニューラルネットワークモデルを提案した。このモデルは、パターンの変動を吸収し、学習性と汎化能力を確保するために、複数の小規模なニューラルネットワークを機能的に組み合わせたファジィ推論アーキテクチャに基づく構造化ニューラルネットワークである。その結果、不確実な判断結果を的確にリジェクトすることが可能となった。手書き郵便番号の認識実験を行い、その認識精度の高いことを示した。

さらに、認識に先立つ文字の切り出しに際し、手書き文字パターンの変動に対応できる、ニューラルネットワークモデルを提案した。このモデルは、文字外形に関する知識を埋め込んだネットワークとノイズなどを除去するネットワークとの相互作用により、文字を柔軟に切り出すものであり、従来のものより優れた性能が得られることを示した。

第5章では、人のように機械が文書を読むを目指して、ニューラルネットワークによるパターン処理と知識による記号処理とが協調動作する文字認識システムを提案した。文字

認識システムの前処理部、認識部をニューラルネットワークによるパターン処理で、後処理部を言語知識を用いた記号処理で実現し、各処理間で相互に情報を交換することにより総合的に認識判断することが可能になった。本システムを用いた実験結果より、統合化処理なしの場合の60%を統合化処理により正確に認識できるようになった。さらに、コピーを重ねた原稿やFAX原稿などの低品質の文書においても良好な認識性能を得ることができ、その有効性を実証した。また、処理時間に関しても、従来手法に比べて高速に処理を行うことも明らかにした。本システムによれば、各処理間の交換情報として、後処理部で訂正した文字を認識部にフィードバックし、誤認識した文字をニューラルネットワークで追加学習し、認識文書に適応した文字認識を行えるように認識部を最適化することも実現可能となる。

6.2 今後の研究展望

人と同じような柔軟なパターン認識を高速に実行する機械を実現するためには、以下の要素技術が今後必要である。

- ・ニューラルネットワークによるパターン認識を高速に実行できるハードウェアの実現
- ・対象物に対する知識・ルールを自動的に獲得し、連想推論ができる情報処理

次に、本論文で述べたそれぞれの研究に対する今後の課題について述べる。

本論文で述べた誤差適応型評価関数による学習は、学習の高速化並びに低い演算精度でも高い性能を維持することができるが、ハードウェア化に際しては適用課題を明確にした上での実現を考えるべきである。本論文で述べた量子化ニューロンモデルと、それを拡張した適応増殖量子化ニューロンモデルを用いたニューラルネットワークモデルは、その高速学習性と追加学習性は優れており、それをハードウェア化すれば従来にない高速な処理が可能になる。そのために、このモデルのアルゴリズムを反映するアーキテクチャを明確にする必要がある。そして、実用課題へ適用できるようにするために、さらに多くのカテゴリーを識別できる大規模なネットワークへの拡張を行う必要がある。このとき、追加学習性を確保できる構造を明確にしなければならない。本論文で述べた構造化ニューラルネットワークは、手書き数字認識においてその有効性を確認できた。先に述べた量子化ニューロンを用いたニューラルネットワークモデルをその要素とすることで、高速学習性と追加学習性を持った大規模ニューラルネットワークに拡張でき、そのハードウェア化が可能になる。認識部はこのように高性能化が図られるのであるが、前処理部もそれに適合するように高速処理が必要である。そのためには、本論文で述べた、結合、分離文字に柔軟に対応できる文字切り出しに関する知識を埋め込んだニューラルネットワークモデルを発展させる必要がある。特徴抽出も可能な構成を持ったモデルに展開し、アーキテクチャを明らかにし、ハードウェア化を実現する必要がある。

本論文では、ニューラルネットワークによるパターン処理と知識による記号処理とが有機的に結合して判断を行うパターン認識は、認識精度を向上できることを示した。しかし、文書の認識が自動化されるためには、文書の構造を反映した処理を導入する必要がある。そのためには、文書画像や言語情報などに関する知識・ルールを獲得する手法を明らかにしなければならない。本論文で述べたパターンと記号との統合化処理は、文書認識以外でも通用する手法であり、対象を文書以外の情報へも拡張することも大切である。

謝辞

本研究を行うにあたり、数多くの方々にお世話になりました。お世話になった方々を以下に列記させていただき、これらの方々に心より深く感謝申し上げます。

本論文をまとめるにあたり、名古屋大学 内川嘉樹教授には励ましと懇切なるご指導をいただきました。

名古屋大学 大熊繁教授、鳥脇純一郎教授からは、多くのご指摘とご指導をいただきました。

松下電器産業(株)研究本部 新田恒治取締役本部長には励ましと本研究を遂行する機会ならびにご指導をいただきました。

松下電器産業(株)電化研究所 藤原義博所長、情報通信研究所 山下貞彦所長には、本研究を遂行する機会と励ましをいただきました。

松下電器産業(株)研究本部 研究企画部 山添祥則部長には、本論文作成のための暖かいご配慮と激励をいただきました。

松下電器産業(株)研究本部 研究企画部 照井康明参事、松島宏司参事には励ましとご配慮をいただきました。

松下電器産業(株)中央研究所 丸野進氏、萱嶋一弘氏、高木英行博士、丹羽寿男博士、山本浩司氏、小島良弘氏、前川英嗣氏、今川太郎氏、伊藤哲氏、オーディオ・ビデオ研究所 阪上茂生氏、松下通信工業(株)AVC研究所 香田敏行氏には、研究の遂行に際し多大なお世話と数多くの討論をいただきました。

その他、松下電器産業(株)中央研究所をはじめとする関係部門の方々には大変お世話になりました。心からお礼申し上げます。

最後に、学会・研究会などを通じて数多くの先輩諸氏の方々に暖かいご助言をいただいたことを感謝申し上げます。

参考文献

- [1] 合原一幸著：“ニューラルコンピューター脳と神経に学ぶ”，東京電機大学，(1988年)
- [2] D. E. Rumelhart, G. E. Hinton and R. J. Williams：“Learning representation by back-propagation errors”，Nature, 323-9, pp. 533-536, (1986年)
- [3] 船橋賢一：“ニューラルネットワークのCapabilityについて”，電子情報通信学会メディカル・バイオエンジニアリング研究会技術報告, MBE88-52, (1988年)
- [4] 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年)
- [5] 坂井邦夫：“文字・文書の認識と理解”，電子情報通信学会誌 Vol. 71, No. 11, pp. 1182-1191, (1988年)
- [6] 山田啓嗣, 上博行, 溝口正規, 天満努：“ニューラルネットを用いた文字認識”，電子情報通信学会パターン認識・理解研究会技術報告, PRU88-58, pp. 79-86, (1988年)
- [7] 森吉弘：“PDPモデルによる手書き漢字認識”，電子情報通信学会論文誌D-II, Vol. J73-D-II, No. 8, pp. 1268-1274, (1990年)
- [8] 今井勝次, 郷原一寿, 内川嘉樹：“フィードバック結合をもつ3層BPモデルを用いた印刷横書き文字列の認識”，電子情報通信学会論文誌D-II, Vol. J74-D-II, No. 11, pp. 1556-1564, (1991年)
- [9] 今井勝次, 郷原一寿, 内川嘉樹：“3層BPモデルを用いたノイズのある原画像に対する手書き文字認識”，電子情報通信学会論文誌D-II, Vol. J75-D-II, No. 2, pp. 314-323, (1992年)
- [10] 城和貴, 森吉弘，“ニューラルネットの学習における新しいタイプの教師”，電子情報通信学会パターン認識・理解研究会技術報告, PRU88-94, pp. 23-30, (1988年)
- [11] 城和貴, 森吉弘，“ニューラルネットの学習における新しいタイプの教師”，電子情報通信学会パターン認識・理解研究会技術報告, PRU88-94, pp. 23-30, (1988年)

- [12] 益岡竜介：“バックプロパゲーションにおける高速学習アルゴリズム”，電子情報通信学会メディカル・バイオエンジニアリング研究会技術報告，MBE88-176，pp.121-126，（1989年）
- [13] 中村雅己，鹿野清宏：“英文テキストリーダからのニューラルネットによる単語予測モデルの検討”，電子情報通信学会音声処理研究会技術報告，SP88-26，pp.39-46，（1988年）
- [14] P. Haffner, H. Sawai, A. Waibel and K. Shikano：“Fast Back-Propagation Learning Methods for Large Phonemic Neural Networks”，日本音響学会講演論文集，1-6-14，（1989年）
- [15] R. A. Jacobs：“Increased Rates of Convergence Through Learning Rate Adaptation”，Neural Networks, Vol. 1, pp. 295-307，（1988年）
- [16] T. P. Vogl, J. K. Mangis, A. K. Rigler and D. L. Alkon：“Accelerating the Convergence of the Back-Propagation method”，Biol. Cybern., 59, pp. 257-266，（1988年）
- [17] 高木英行，阪上茂生，戸川隼人：“非線形最適化手法を用いたニューラルネットワーク学習アルゴリズムの高速化”，電子情報通信学会平成1年総合全国大会，SD-1-12，（1989年）
- [18] 高木英行，阪上茂生，戸川隼人：“ニューラルネットにおける非線形最適化手法の効果”，電子情報通信学会論文誌D-II, Vol. J74-D-II, No. 4, pp. 528-535，（1991年）
- [19] 根岸成郎，高橋治久，富田悦次：“Kullback誤差関数による誤差伝播と性能評価”，電子情報通信学会MEとバイオサイバネティクス研究会技術報告，MBE88-175，pp.115-120，（1989年）
- [20] S. J. Hanson and D. J. Burr：“Minkowski-r Back-Propagation：Learning in Connectionist Models with NonEuclidian Error Signals”，Neural Inf. Process. Syst., pp. 348-357，（1987年）
- [21] Baker T. and Hammerstrom D.：“Characterization of Artificial Neural Network Algorithms”，Proc. Int. Symposium on Circuits and Syst., 1, pp. 78-81, IEEE, （1989年）
- [22] Holt J. L. and Baker T. E.：“Back Propagation Simulations using Limited Precision Calculations”，Proc. Int. Joint Conf. on Neural Networks, Seattle, II, pp. 121-126, IEEE, （1991年）
- [23] Holt J. L. and Hwang Jenq-Neng：“Finite Precision Error Analysis of Neural Network Electronic Hardware Implementations”，Proc. Int. Joint Conf. on Neural Networks, Seattle, I, pp. 519-525, IEEE, （1991年）
- [24] Shoemaker P. A., Carlin M. J. and Shimabukuro R. L.：“Back Propagation Learning with Ternary Quantization of Weight Updates”，Neural Networks, 4,

- pp. 231-241, （1991年）
- [25] 片山浩，中山謙二：“階層型神経回路網における低ビット学習の一方法”，電子情報通信学会ニューロコンピューティング研究会技術報告，NC91-11，（1991年）
- [26] Hoehfeld M. and Fahlman S. E.：“Learning with Limited Numerical Precision Using the Cascade-Correlation Algorithm”，CMU technical report, CMU-CS-91-130, Carnegie Mellon Univ., （1991年）
- [27] 麻生英樹：“誤差逆伝播学習の数理的性質”，電子情報通信学会パターン認識・理解研究会技術報告，PRU89-14，（1989年）
- [28] 中山謙二，猪股悟，竹内由幸：“階層型神経回路網のデジタル構成におけるビット数低減方法”，電子情報通信学会論文誌D-II, Vol. J73-D-II, No. 8, pp. 1336-1345, （1990年）
- [29] 香田敏行，阪上茂生，高木英行，木泰治：“ニューラルネットワークによる英数字認識—計算量の削減及び学習の高速化に関する検討—”，電子情報通信学会パターン認識・理解研究会技術報告，PRU88-151，pp. 33-40，（1989年）
- [30] 香田敏行，阪上茂生，高木英行，木泰治：“ニューラルネットによる手書き英数字認識—計算量の削減および学習の高速化に関する検討—”，電子情報通信学会パターン認識・理解研究会，PRU88-151，（1989年）
- [31] 香田敏行，阪上茂生，山本浩司，高木英行，木泰治：“誤差適応型評価関数によるバックプロパゲーション学習法の高速化”，電子情報通信学会秋季全国大会D-208, p. 6-209, （1989年）
- [32] 香田敏行，阪上茂生，山本浩司，木泰治：“パターン認識に適したニューラルネットワークの高速学習法”，電子情報通信学会秋季全国大会，D-18，p. 6-18, （1990年）
- [33] 阪上茂生，香田敏行，山本浩司，木泰治：“精度が限定されたニューラルネットワークによる文字認識”，電子情報通信学会秋季全国大会D-38, p. 6-38, （1990年）
- [34] 阪上茂生，香田敏行，山本浩司，木泰治：“精度が限定されたニューラルネットワークによる文字認識”，電子情報通信学会ニューロコンピューティング研究会，NC90-33, pp. 21-28, （1990年）
- [35] 阪上茂生，香田敏行，山本浩司，木泰治：“精度が限定されたニューラルネットワークによる文字認識”，電子情報通信学会春季全国大会，D-52，p. 6-52, （1991年）
- [36] 阪上茂生，香田敏行，山本浩司，丸野進，木泰治：“バックプロパゲーション学習に必要な演算精度の削減”，電子情報通信学会ニューロコンピューティング研究会，NC91-32, pp. 17-24，（1991年）
- [37] S. Sakaue, T. Kohda, H. Yamamoto, S. Maruno and Y. Shimeki：“Reduction of Necessary Precision for the Learning of Pattern Recognition”，Inter-

- national Joint Conference on Neural Networks, IJCNN' 91 Singapore, Vol. 2, pp. 1795-1800, (1991年)
- [38] 阪上茂生, 香田敏行, 山本浩司, 丸野進, 木泰治: "パターン認識を学習する誤差逆伝搬法に必要な演算精度の削減", 電子情報通信学会論文誌D-II, Vol. J76-D-II, No. 3, pp. 672-679, (1993年)
- [39] S. Sakaue, T. Kohda, H. Yamamoto, S. Maruno, and Y. Shimeki: "Reduction of Precision for Back-Propagation Applied to Pattern Recognition", IEEE Transaction on Neural Networks, Vol. 4, No. 2, pp. 270-275(1993年)
- [40] D. E. Rumelhart, G. E. Hinton and R. J. Williams, "Learning Internal Representations by Error Propagation", Parallel Distributed Processing, Vol. 1, pp. 318-362, M. I. T. Press, (1986年)
- [41] E. McDermott, "LVQ3 for Phoneme Recognition", Proc. 1990. Conf. ASJ, 2-p-16, pp. 151-152, (1990年)
- [42] D. Reilly, L. Cooper and C. Elbaum: "A Neural Model for Category Learning", Biol. Cybern., Vol. 45, pp. 35-41, (1992年)
- [43] 丸野進, 木泰治: "機能別階層ネットワークを用いた文字認識システム", 平成2年度第18回画像電子学会全国大会 No. 18 pp. 77-80, (1990年)
- [44] 香田敏行, 丸野進, 小島良宏, 阪上茂生, 山本浩司, 木泰治: "量子化ニューロンを用いた機能別階層ネットワークの追加学習", 電子情報通信学会春季全国大会, D-65, p. 6-65, (1991年)
- [45] 丸野進, 香田敏行, 小島良宏, 阪上茂生, 山本浩司, 木泰治: "量子化ニューロンを用いた機能別階層ネットワークによる文字認識", 電子情報通信学会春季全国大会, D-66, p. 6-66, (1991年)
- [46] S. Maruno, T. Kohda, Y. Kojima, S. Sakaue, H. Yamamoto, and Y. Shimeki: "Multi-Functional Layered Network with Quantizer Neurons", International Joint Conference on Neural Networks, IJCNN' 91 Singapore, Vol. 1, pp. 541-546, (1991年)
- [47] 丸野進, 香田敏行, 小島良宏, 阪上茂生, 山本浩司, 木泰治: "適応増殖量子化ニューロンを用いた文字認識", 電子情報通信学会ニューロコンピューティング研究会, NC91-51, pp. 23-28, (1991年)
- [48] 香田敏行, 丸野進, 小島良宏, 阪上茂生, 山本浩司, 木泰治: "適応増殖量子化ニューロンによるニューラルネットの追加学習性", 電子情報通信学会ニューロコンピューティング研究会, NC91-52, pp. 29-35, (1991年)
- [49] S. Maruno, T. Imagawa, T. Kohda, Y. Kojima, H. Yamamoto and Y. Shimeki: "Adaptive Segmentation of Quantizer Neuron Architecture (ASQA)", International Joint Conference on Neural Networks, IJCNN' 93-NAGOYA, (1993)
- [50] Fukushima, K.: "Neocognitoron: A hierarchical neural network capable of visual pattern recognition", Neural Networks, 1(2), pp. 119-130, (1988年)
- [51] 今川太郎, 福島邦彦: "選択的注意機構による英字筆記体連結文字列認識", 電子情報通信学会論文誌D-II, Vol. J74-D-II, No. 12, pp. 1768-1775, (1991年)
- [52] D. J. Burr: "A neural network digit recognizer", Proc. IEEE SMC, pp. 1621-1625, (1986年)
- [53] J. S. Denker, et al: "Neural network recognizer for hand-written ZIP code digits", Proc. IEEE NIPS, pp. 323-331, (1988年)
- [54] 諏訪喜久, 川尻博光, 岩田彰, 鈴木宣夫: "ComNet-IIによる手書き数字の認識", 電子情報通信学会1991年秋季大会講演論文集, D-19, p. 6-19, (1991年)
- [55] 小島良宏, 山本浩司, 香田敏行, 阪上茂生, 丸野進, 木泰治, 川上和隆, 水谷幹男: "構造化ニューラルネットワーク(NARA)による手書き数字認識", 電子情報通信学会秋季全国大会, D-40, p. 6-40, (1992年)
- [56] 小島良宏, 山本浩司, 香田敏行, 阪上茂生, 丸野進, 木泰治, 川上和隆, 水谷幹男: "構造化ニューラルネットワーク(NARA)による手書き数字認識", 電子情報通信学会「手書き文字認識技術の過去・現在・未来」に関するシンポジウム, pp. 31-27, (1993年)
- [57] Y. Kojima, H. Yamamoto, T. Kohda, S. Sakaue, S. Maruno, Y. Shimeki, K. Kawakami and M. Mizutani: "Recognition of Handwritten Numeric Characters Using Neural Networks Designed on Approximate Reasoning Architecture", International Joint Conference on Neural Networks, IJCNN' 93-NAGOYA, (1993年).
- [58] 高木英行, 香田敏行, 小島良宏: "ファジィ推論アーキテクチャに基づくニューラルネット", 日本ファジィ学会誌, Vol. 3, No. 1, pp. 133-141, (1991年)
- [59] 珠, 浅居, 川 共著: "ファジィシステム入門", オーム社, (1987年)
- [60] 菅野道夫著: "ファジィ制御", 日刊工業新聞社, (1988年)
- [61] 小島良宏, 香田敏行, 高木英行, 木泰治: "ニューラルネットワークの汎化性に関する一検討", 電子情報通信学会1990年春季大会講演論文集, D-5, p. 6-5, (1990年)
- [62] Y. Linde, A. Buzo and R. M. Gray: "An algorithm for vector quantizer design", IEEE Trans. Commun., COM-28, 1, pp. 84-95, (1980年)
- [63] 脇康, 藤原啓恵, 竹之内磨理子, 横江享, 清水正博: "文書清書システム(3)-文字認識アルゴリズムとそのハードウェア-", 電子情報通信学会昭和61年総合全国大会, 1510, (1986年)
- [64] J. J. Hopfield and D. W. Tank: "Computing with Neural Circuits: A Model",

- Science, Vol. 233, Aug. 8, p. 625, (1986年)
- [65] C. Koch, J. Marroquin and A. Yuille: "Analog Neural networks in early vision", Proc. Natl. Acad. Sci. U.S.A., Vol. 83, pp. 4263-4267, (1986年)
- [66] S. Geman and D. Geman: "Stochastic Relaxation, Gibbs Distribution and the Bayesian Restoration of Images", IEEE Trans., Vol. PAMI-6, pp. 721-741, (1984年)
- [67] 馬場口登, 塚本正敏, 相原恒博: "手書き日本文字列からの文字切り出しの基礎的考察" 電子情報通信学会論文誌D, Vol. J68-D, No. 12, pp. 2123-2131, (1985年)
- [68] 山本浩司, 阪上茂生, 丸野進, 木泰治: "神経回路網を用いた文字切り出し手法の検討", 電子情報通信学会秋季全国大会, D-71, p. 6-71, (1992年)
- [69] H. Yamamoto, S. Sakaue, S. Maruno and Y. Shimaki: "Segmentation of Handwritten Japanese Character Strings with Hopfield type Neural Networks", International Joint Conference on Neural Networks, IJCNN'93-NAGOYA, (1993年)
- [70] 山本浩司, 阪上茂生, 丸野進, 木泰治: "ホップフィールド型神経回路網を用いた文字切り出し手法", 電子情報通信学会論文誌D-II, Vol. J78-D-II, No. 2, pp. 347-377, (1995年)
- [71] 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", 電子情報通信学会論文誌D, Vol. J69-D, No. 8, pp. 1187-1196, (1986年)
- [72] 豊田順一, 野口要治, 西村康: "日本語印刷文書における文字の切り出し-新聞読み取りへの応用-", 情報処理学会論文誌, Vol. 24, No. 4 pp. 481-487, (1983年)
- [73] Fletcher L. A. and Katsuri R.: "A Robust Algorithm for Text String Separation from Mixed Text/Graphics Images", IEEE Trans. on PAMI, Vol. 10, No. 6, pp. 910-918, (1988年)
- [74] 秋山照雄, 内藤誠一郎, 増田功: "非接触文字優先切り出しによる印刷物からの文字切り出し法", 電子情報通信学会論文誌D, Vol. J67-D, No. 10, pp. 1194-1201, (1984年)
- [75] 中村納, 氏家誠, 岡本教佳, 南敏: "ミックストモード通信のための文字領域の抽出アルゴリズム", 電子情報通信学会論文誌D, Vol. J67-D, No. 11, pp. 1227-1284, (1984年)
- [76] 村瀬洋: "走り書き文字列認識に於ける文字切り出しの個人適応化", 電子情報通信学会論文誌D-II, Vol. J72-D-II, No. 1, pp. 132-139, (1989年)
- [77] 安田道夫, 藤沢浩道: "文字認識のための相関法の一改良", 電子情報通信学会論文誌D, Vol. J62-D, No. 3, pp. 217-224, (1979年)
- [78] 小森和昭, 川谷隆彦, 石井健一郎, 飯田行恭: "特徴集積による手書き仮名文字の認識", 電子情報通信学会論文誌D, Vol. J63-D, No. 11, pp. 962-969, (1980年)

- [79] 入江文平, 矢口克己: "2×2近傍特徴による手書き漢字認識", 電子情報通信学会昭和63年春季全国大会, D-453, (1988年)
- [80] 黒沢由明, 麻田治男: "自由手書き文字認識", 東芝レビュー, Vol. 41, No. 12, pp. 1012-1015, (1986年)
- [81] 山本和彦: "弛緩整合法による手書き教育漢字認識", 電子情報通信学会論文誌D, Vol. J65-D, No. 9, pp. 1167-1174, (1982年)
- [82] 杉村利明: "候補文字補完と言語処理による漢字認識の誤り訂正処理法", 電子情報通信学会論文誌D-II, Vol. J72-D-II, No. 7, pp. 993-1000, (1989年)
- [83] 杉村利明, 斎藤珠喜: "文字接続情報を用いた読み取り不能文字の判定処理-文字認識への応用", 電子情報通信学会論文誌D, Vol. J68-D, No. 1, pp. 64-71, (1985年)
- [84] 北村正, 仲林清, 大光明直孝, 中村修: "単語知識を利用した手書き文字列処理方式", NTT R&D, Vol. 39, No. 3, pp. 429-436, (1990年)
- [85] 高尾哲康, 西野文人: "日本語リーダ後処理の実現と評価", 情報処理学会論文誌, Vol. 30, No. 11, pp. 1391-1401, (1989年)
- [86] Kise K., Shiraishi T., Takamatsu S. and Kusaka H.: "Improvement of Text Image Recognition based on Linguistic Constraints", Proc. of MVA'92, pp. 511-514, (1992年)
- [87] 吉村賢治, 竹内美津乃, 津田建蔵, 首藤公昭: "コスト最小法を用いた日本語文の形態素解析", 情報処理学会自然言語研究会資料, 60-1, (1987年)
- [88] 五十嵐治一, 松田透: "形態素解析を利用したOCR誤読文字の自動修正", Richoh Technical Report, No. 15, APRIL, (1986年)
- [89] 木谷強: "手書き文字の文字認識結果に対する後処理方式", 情報処理学会自然言語研究会資料, 86-1, (1991年)
- [90] 小黒雅己, 中村修, 北村正: "手書き単語組織別のための最良優先探索方式", 電子情報通信学会人工知能と知識処理研究会技術報告, AI90-62, (1990年)
- [91] 小黒雅己, 中村修, 北村正: "手書き複合語文字列識別のための最適単語組探索方式", 電子情報通信学会論文誌D-II, Vol. J75-D-II, No. 1, pp. 96-102, (1992年)
- [92] 紺野章子, 本郷保夫: "日本語OCRの後処理に関する一手法", 電子情報通信学会パターン認識・理解研究会技術報告, PRU92-21, (1992年)
- [93] 阿曾弘具, 大町真一郎, 木村正行, 勝山裕: "高速構成度知的認識システムSEIUN", 電子情報通信学会論文誌D-II, Vol. J72-D-II, No. 7, pp. 993-1000, (1989年)
- [94] 村瀬洋, 新谷幹夫, 若原徹, 小高和己: "言語情報を利用した手書き数字列からの文字切り出しと認識", 電子情報通信学会論文誌D, Vol. J69-D, No. 9, pp. 1292-1301, (1986年)
- [95] 澤田清, 吉川隆敏, 堀井洋, 亀井勇: "認識情報・言語情報を統合したオンライン

- 文字切り出し”, 情報処理学会第43回全国大会, 2D-3, pp. 2-23-2-24, (1991年)
- [96] 早川恵子, 吉田敏之: ”フリーピッチ文字認識に於ける文字切り出し方法の検討”, 情報処理学会第43回全国大会, 2D-12, pp. 2-41-2-42, (1991年)
- [97] 小林弥生, 津雲淳: ”文字連接情報を利用した手書き文字列認識”, 電子情報通信学会自然言語理解研究会技術報告, NLC91-32, (1991年)
- [98] 小島良宏, 香田敏行, 丸野進, 阪上茂生, 山本浩司, 木泰治: ”構造化ニューラルネットワーク(NARA)によるマルチフォント漢字認識”, 電子情報通信学会ニューロコンピューティング研究会, NC91-31, pp. 9-16, (1991年)
- [99] 丸野進, 小島良宏, 山本浩司, 丹羽寿男, 萱嶋一弘, 木泰治: ”パターンと記号の統合化処理CoPSによる文書認識(1)”, 電子情報通信学会秋季全国大会, D-313, p. 6-315, (1992年)
- [100] 山本浩司, 小島良宏, 丹羽寿男, 丸野進, 木泰治: ”CoPSによる文書認識(2)神経回路網による文字切り出し処理-フィードバック情報に基づくパラメータ制御-”, 電子情報通信学会秋季全国大会, D-314, p. 6-316, (1992年)
- [101] 小島良宏, 山本浩司, 丹羽寿男, 香田敏行, 丸野進, 木泰治: ”CoPSによる文書認識(3)-複数特徴量を統合した構造化ニューラルネットワークによる漢字認識-”, 電子情報通信学会秋季全国大会, D-315, p. 6-317, (1992年)
- [102] 丹羽寿男, 山本浩司, 小島良宏, 萱嶋一弘, 木泰治: ”CoPSによる文書認識-後処理における誤訂正単語のリジェクター-”, 電子情報通信学会秋季全国大会, D-316, p. 6-318, (1992年)
- [103] 丹羽寿男, 伊藤哲, 小島良宏, 山本浩司, 萱嶋一弘, 木泰治: ”CoPSによる文書認識-後処理におけるキーワード情報を用いた候補外文字救出-”, 電子情報通信学会秋季全国大会, D-317, p. 6-319, (1992年)
- [104] 丹羽寿男, 萱嶋一弘, 木泰治: ”キーワード抽出と最適文節選択による文字認識後処理”, 電子情報通信学会論文誌D-II, Vol. J76-D-II, No. 5, pp. 940-948, (1993年)
- [105] 丹羽寿男, 萱嶋一弘, 木泰治: ”文字認識誤りモデルを用いた文字認識の誤り分析評価”, 電子情報通信学会論文誌D-II, Vol. J76-D-II, No. 9, pp. 1843-1850, (1993年)
- [106] 丹羽寿男, 山本浩司, 小島良宏, 木泰治, 萱嶋一弘, 丸野進, ”パターンと記号の統合化処理による文字認識”, 電子情報通信学会論文誌D-II, Vol. J78-D-II, No. 2, pp. 263-271, (1995年)

