マイクロコンピュータと　61－1
ワークステーション
（1990．5．23）

ニューラルネットワークアクセラレータ NEURO-TURBO

柳橋喜久治*　松田　聡*　吉田征夫*　岩田　彰**　鈴村宣夫**
*(株) マイテック　**名古屋工業大学電気情報工学科

２４ビット浮動小数点方式の汎用ＤＳＰとして開発されたMB８６２２０を４個用いたニューラルネットアクセラレータNEURO-TURBOについて述べる。NEURO-TURBOは、ＤＳＰ４個をデュアルポートメモリ（ＤＰＭ）を介してリング状に結合したＭＩＭＤ型並列処理プロセッサである。バックプロパゲーションの学習時の演算速度で２ＭＣＰＳ、また、前向きの演算速度で１１ＭＣＰＳを得た。また、NEURO-TURBOに、我々が先に提案した大規模４層ニューラルネット (CombNET) をインプリメントした。その結果、印刷漢字認識の処理時間が、学習時に数時間、認識時には100文字/秒と高速化され、実用的な処理速度を得ることができた。

A NEURAL NETWORK ACCELERATOR NEURO-TURBO

Kikuji YANAGIHASHI*, Satoshi MATSUDA*, Masao YOSHIDA*
Akira IWATA and Nobuo SUZUMURA****
*** Mitec Corp., 3-32-1, Takada, Toshima-ku, Tokyo, 171, JAPAN**
****Dept. of Electrical and Computer Eng., Nagoya Institute of Technology**
Gokiso-cho, Showa-ku, Nagoya, 466, JAPAN

A Neural Network Accelerator named Neuro Turbo was developed using four general purpose 24 bits floating point Digital Signal Processor(DSP) MB86220. Neuro Turbo is a MIMD type parallel processor having ring coupled 4 DSPs and 4 Dual Port Memories(DPM). The performance was evaluated by constructing a neural network to recognize the 26 type fonts of the alphabet set. The processing speeds of 2 MCPS for learning procedure and 11 MCPS for forward pass were achieved. This paper also describes the implementation of CombNET on NEURO-TURBO. A network to recognize 2965 printed Chinese characters has been constructed. The recognizing rate of 99.5% has been achieved even for test data sets.

1　まえがき

近年、ニューラルネットを音声認識、画像処理、ロボット制御などさまざまな分野に適用した研究が多く報告されているが、これらの分野に実際にニューラルネットを適用する場合、実用的処理速度を得るにはかなり高速な処理能力を有するプロセッサが必要とされる。ニューラルネットにおける計算処理は元来並列処理であるため、計算速度を上げるためには並列処理アーキテクチャを採用することが有効である。究極的には、ニューロンごとに専用プロセッサを用意し、それらを適当なネットワークで連結してニューラルネットを実現する方式（ニューロコンピュータ）が見込まれるが、現状技術ではその方式はコストパーフォーマンス面から考えると得策ではない。

そこで、積和演算を高速に行うことのできる信号処理プロセッサ（ＤＳＰ）を複数個用いたニューラルネットアクセラレータがいくつか発表されている。それらのシステムでは、ニュラルネットを複数個のプロセッサに分割して配置し、プロセッサ間を結ぶバスによって互いのプロセッサが通信しながら並列的にニューラルネットの計算し実行する。近年、大きなアドレス空間を持ちまた浮動小数点演算のできる第３世代ＤＳＰが開発されるに至り、大規模なニューラルネットをＤＳＰを用いたシステムで構築することができるようになった。汎用ＤＳＰを用いたニューラルネットアクセラレータとしては、ＴＩのＯｄｙｓｓｅｙ[1]、ＩＢＭのＮＥＰ[2]などが発表されている。前者は共通バスに複数のＤＳＰを配置した構成で２ＭＣＰＳの速度を得ている。後者は複数のＤＳＰをリングに結合したシステムで４９０ＫＣＰＳの速度を得ている。また、Ｗａｒｐ[3]では専用の浮動小数点プロセッサ１０個をリニアアレイ状に配置したシステムで１７ＭＣＰＳを実現している。

我々は、汎用ＤＳＰとして最近開発されたＭＢ８６２２０を４個用いたニューラルネットアクセラレータ、NEURO-TURBOを構築した[4][5]。表１にＭＢ８６２２０の諸元を示す。ＭＢ８６２２０は２４ビット浮動小数点方式ＤＳＰ（ただし、内部演算は３０ビット浮動小数点方式）であるが、ニューラルネットの計算ではニューロン毎に特性関数によって０から１までの間に正規化されることと、実際応用の際、入力データとしては高々１６ビット精度であることを考慮すると、２４ビットの浮動小数点演算による演算精度で十分であると考える。ＭＢ６８２２０は富士通のＣＭＯＳ１．２μｍルールのＶＬＳＩとして構成され安価に供給される。NEURO-TURBOは、このＤＳＰ４個を独自のリング結合したＭＩＭＤ型並列処理プロセッサである。

2　ＤＳＰのリング結合アーキテクチャ

ニューラルネットは積和演算をベースとする単純な計算の集合で、並列的に構成される。このニューラルネットの並列性を生かした高速な演算処理装置が要求される。この目的のために従来から複数のコンピュータによる並列分散処理が考えられてきたがコンピュータ間のシステムバスのバッファ及びスイッチング回路等のハードウェアの増大とバスアービトレーション等の制御の複雑さが問題となっている。

ここでは最もハードウェアが簡単化される並列分散処理トロボジを選択し、その上でニューラルネットの実行速度を落とさない最適な計算アルゴリズムを組み合わせて実行速度の向上を図った。図１にこの構成の概念的なブロックダイヤグラムを示す。図に示されるように４個のＤＳＰはお互いにデュアルポートメモリ（ＤＰＭ）を介してリング状に結合され、その間にバッファやスイッチング回路をなくし、ハードウェアの簡単化を図っている。１つのＤＳＰの外部データメモリとして、隣接する２つのＤＰＭと１つのワーキングメモリ（ＷＭ）があり、それらはＤＳＰのシステムバスで結合されている。

１つのＤＰＭは隣接する２つのＤＳＰからランダムにデータをアクセスすることができ、ＤＳＰ間のデータ転送用として用いられる。ＷＭは、それが接続されているＤＳＰのワーキングメモリとして結合の重みやデータ及び中間結果をストアするのに用いられる。ニューラルネットは連続した単純な積和演算を繰り返し実行することが多く、計算アルゴリズムの工夫により、演算を分割しデータをパイプライン状にリングに沿って淀みなく転送し、並列処理することによって高速実行が可能となる。この時、一方のＤＰＭのデータをＤＳＰに入力し処理後、他方のＤＰＭに出力することによって、データのメモリアクセスが実質的にデータ転送となり、データ転送時間を最小にして高速並行処理を実現している。

このリング結合アーキテクチャを用いた並列処理プロセッサ NEURO-TURBO の構成を図２に示す。図３は NEURO-

表１．ＭＢ８６２２０の諸元

DATA FORMAT	24 bits floating point 18 bits mantissa, 6 bits exponent
MULTIPLIER	18E6 x 18E6 → 24E6
ALU	18E6 + 18E6 → 24E6 24E6 + 24E6 → 24E6
MACHINE CYCLE	150nsec per floating point arithmetic instruction 75nsec per other instruction
PROGRAM MEMORY	2KW (internal) 64KW (external)
DATA MEMORY	256W x 2 banks (internal) 64KW x 4 banks (external)

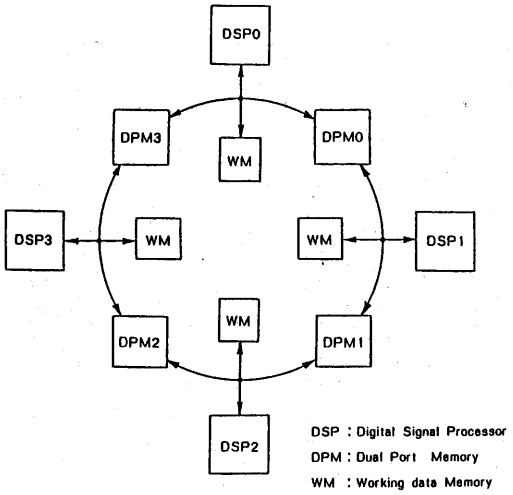


図1．デュアルポートメモリ(DPM)を介したリング結合アーキテクチャ

TURBO外観である。ホストコンピュータとして、NECのパーソナルコンピュータＰＣ－９８０１を用い、その拡張スロットに本装置をプラグインする。ホストコンピュータと本装置のインターフェースとして別のＤＰＭを用いて、データ転送を行う。ホストコンピュータからは、各ＤＳＰのプログラムメモリ（ＰＭ）に、実行前に予めプログラムをロードすることができ、問題毎にプログラムを入れ換えて使用することが出来る。WMはアドレスを２ビット拡張し、メモリ空間を６４ｋｗ～２５６ｋｗとしている。さらに、このシステムのＤＳＰは内部システムクロック１２ＭＨｚで動作し、外部データメモリとのアクセスは１ウェイトを必要する。各ＤＳＰは積和演算を２クロックサイクルで実行する。４つのＤＳＰを用いたこのシステムの最大スループットはほぼ２４ＭＯＰＳとなる。

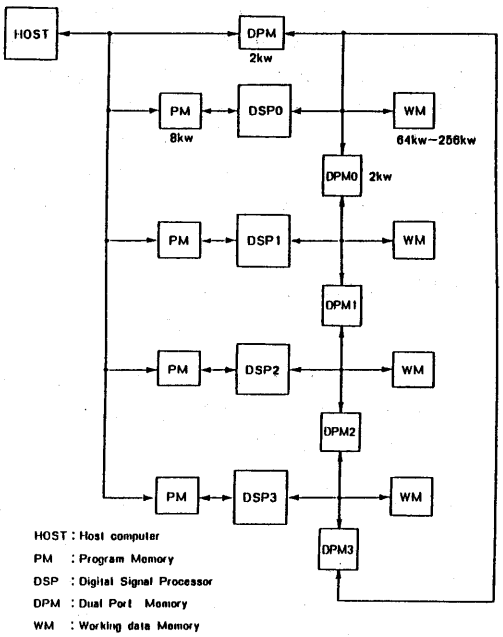


図2．NEURO-TURBOの構成図

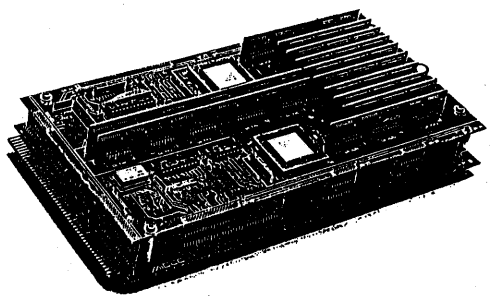


図3．NEURO-TURBOの外観

3　性能評価

NEURO-TURBOの性能評価は、バックプロパゲーションアルゴリズムを用いて、アルファベット２６文字のタイプフォントの認識のための３層（１隠れ層）ニューラルネットワークの作成によって行った。入力パターンは、イメージスキャナによって読み込まれたタイプフォントイメージを７×５の小領域に分割し、小領域ごとの濃度値を０から１の値に正規化することによって得た。ネットワークの構成は、入力層ユニット３５個、中間層ユニット２５個、出力層ユニット２６個、バイアスユニット２個であり、ユニット間の結合数は１５７６結合である。NEURO-TURBOを用いて、このネットワークの一回の学習に要した時間は７８０μｓｅｃであった。この速度は、ほぼ2ＭＣＰＳに相当する。また、前向きの演算速度は、このネットワークで１４０μｓｅｃであり、これは、ほぼ１１ＭＣＰＳに相当する。ちなみに、同じネットワークのシミュレーションをＳＵＮ４／２６０ワークステーションを用いて行ったところ、学習時１００ｋＣＰＳ、前向き演算で３７０ＫＣＰＳであった。すなわち、NEURO-TURBOはＳＵＮ４／２６０の約２０倍の高速演算能力を持っている。この速度は専用の浮動小数点プロセッサ１０個をリニアアレイ状に配置したＷａｒｐには及ばないものの、本システムと同様に汎用ＤＳＰを用いたニューラルネットアクセラレータＯｄｙｓｓｅｙやＮＥＰより優っている。NEURO-TURBOで

用いたＤＳＰ、ＭＢ８６２２０は６ＭＯＰＳの積和演算速度を持っており、これを４個使用した本アクセラレータは最大２４ＭＯＰＳの積和演算速度を持っている。この点を考慮すると、３層ニューラルネットワークのバックプロパゲーションアルゴリズムの実行において上記の高速演算性能を得たことは、デュアル・ポート・メモリを介してリング状に結合された４個のＤＳＰが効率よく動作していることを示している。

4　大規模４層ニューラルネット（ＣｏｍｂＮＥＴ）

これまでに行なわれたニューラルネットワークに関する研究は、分類カテゴリー数の少ない比較的小規模なニューラルネットを取り扱っており、実用的で大規模なニューラルネットにこれらの成果をそのまま拡張できるかは疑問である。例えばJIS第一水準の漢字を認識するような大規模なニューラルネットを単純なバックプロパゲーションアルゴリズムによって構築することは実際には多くの困難を伴う。そのような大規模なネットワークの学習では、ローカルミニマムに陥る可能性も高く、たとえ収束するにせよ膨大な計算量を費すことになる。

大規模なニューラルネットを実用的な計算時間で構築するには、ひとつにはニューラルネットのシミュレーションを高速に実行できるプロセッサを用いることであるが、ネットワークを学習を行い易い小規模なネットワークに分割し、それらの統合として大規模なネットワークを構築することも必要となる。そこで、我々は、従来のネットワークモデルでは困難であった多数のカテゴリを分類するための大規模ニューラルネットワーク(CombNET)を提案している[6][7]。

CombNETは、図４に示すように、第１層にベクトル量子化型ニューラルネットを配置し、後段の第２、３、４層には小グループ内のデータを分類する小規模な階層型ニューラルネットを多数並列に配置したネットワークである。第１層を櫛の幹、第２、３、４層を櫛の葉とみなすと、ちょうど櫛のような構造をもっているので、櫛型ネットワークという意味からＣｏｍｂＮＥＴと名付けた。そして、第１層を**Stem Network**、第２、３、４層を**Branch Network**と呼んでいる。

CombNETの学習は次の手順で行う。まず、Stem Networkの学習をKohonenの自己組織化アルゴリズム[8]により行う。Stem Networkの学習後、入力データの各カテゴリがStem Networkのどのニューロンと最適整合になるかを調べ、各ニューロンの分担すべきカテゴリを求め、全カテゴリをStem Networkのニューロン数と同数のグループに分割する。次に、その分割されたグループごとにその中にあるカテゴリを識別するためのBranch Networkの学習をバックプロパゲーション法[9]を用いて行う。

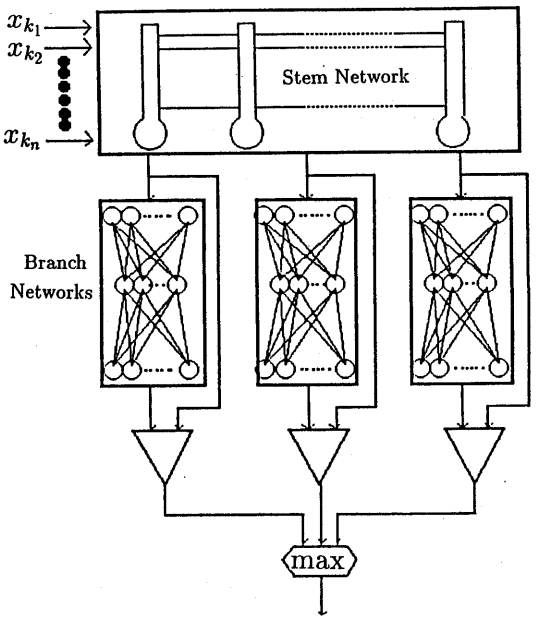


図４．CombNETの構成

識別は次のようにして行う。まず、Stem Networkに入力データを通し、適合度の高いものから３ないし５番目までのニューロンを選ぶ。次に、選ばれたニューロンが担当するカテゴリグループを分割する後段のBranch Networkに入力データを入力し、出力層の中でもっとも高い発火レベルになったものの出力値を調べる。そして、

(適合度)$^{\alpha}$(出力値)$^{\beta}$

の値がもっとも高くなったものを識別結果としてを選ぶ。こうして、後段の複数の階層型ネットワークの出力と競合させることで識別能力を高めることを目指した。

このネットワークの利点は学習の容易さにある。バックプロパゲーション法による学習は小規模なネットワークについては収束も容易であるが、大規模なネットワークになるとローカルミニマムに陥る可能性も高く、たとえ収束するにせよ膨大な計算量を費すことになる。本方式は、Stem Networkによって大分類をしてからバックプロパゲーションによる学習を行なうため、Branch Networkの規模は小さくなり、したがって、学習は容易になる。また、Branch Networkは相互には結合のない独立したネットワークであるから、大規模なネットワークであっても全体の結合数を少なく抑えることができる。

5　ＣｏｍｂＮＥＴによるJIS第１水準印刷漢字の識別

ＣｏｍｂＮＥＴのパターン識別能力を検討するために、ＪＩＳ第１水準の印刷漢字２９６５字種を識別するニューラルネットの構築を試みた。

学習データは次のように作成した。まず、ＪＩＳ第１水準漢字を印刷したものを解像度が４００ｄｏｔ／ｉｎｃｈのイメージスキャナで読み込む。ここでは、原稿はＡ４縦置き、横書きとした。そして、イメージスキャナーで読み込んだ画像から、

（１）横（行）方向に画素値を積分した周辺分布から、しきい値処理により行を切り出し、

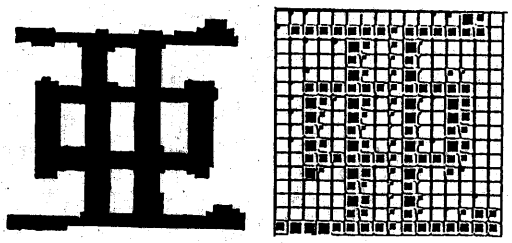
（２）縦方向に画素値を積分した周辺分布から文字を切り出した。

１文字の大きさは４ｍｍ程度であり、これをイメージスキャナで読み取り切り出すと縦横がおよそ６４ドット×６４ドット程度の２値画像が得られる（図５（ａ））。これを１６×１６の小領域にまとめて、濃度正規化したパターンを作成した（図５（ｂ））。今回、これを５セット作成し、そのうちの４セットの平均パターンを学習データにした。また、４セットのうちの１セット（学習パターン）と、残り１セット（未学習パターン）を用いてネットワークのパターン識別能力の評価を行なった。

Stem Networkのニューロン数は１４４個（１２×１２）とした。参照ベクトルの初期値は一様乱数で与え、学習回数は１字種当たり１００回の学習を行なった。その結果得られた各ニューロンの参照ベクトルを図６に示す。この結果は非常に興味深いものが得られた。図６には、参照ベクトルの値を１６×１６の２次元マトリックス状のパターンで示したので、各参照ベクトルがどのようなテンプレートを形成しているかが明らかとなっている。各参照ベクトルには、漢字の「へん」、「つくり」、「かまえ」が形成されている。自己組織化によって図６に示すような参照ベクトルが自律的に形成されたことは注目に値する。

次に、後段のBranch Networkについて、各ニューロンに割り当てられた漢字のグループごとに、バックプロパゲーション法を用いて学習を行なった。各グループの字種数は最大でも９１個であり、比較的小規模なニューラルネットとなるため、すべてのグループについて容易に学習が収束した。

このように学習を行なったネットワークの識別能力を検討した。さまざまなα、βの値を用いて識別を行なったところ、α/βが5程度のところで未学習パターンについても９９．５％の正当率を得られることがわかった。適合度や出力値はともに０～１の間の値であるから、$\alpha/\beta=5$というのはStem



(a)　　(b)

図５．文字パターンの作成
(a) 切り出された文字イメージ
(b) 濃度正規化された文字パターン

Networkの適合度の結果よりも、Branch Networkの出力値をより強く考慮して判断していることを表している。これはＢＰ学習による階層型ニューラルネットの２つの性質、１つは学習パターンについては特定の素子が強い出力を出し、未知パターンについてはどの素子も弱い出力を出す性質、もう１つは汎化能力に優れパターン変動に強いという性質をうまく利用したものになっている。

実際にこのことを基本の『基』という文字を例に上げて説明する。図７は、この文字をStem Networkに通した結果を示している。白抜き四角内黒い部分は入力パターンに対する各ニューロンの適合度を示しており、その下は各ニューロンの参照ベクトルを表わしている。そして、四角枠は、適合度の高いニューロンを３つ示している。

次に、この３つのニューロンに接続されたBranch Networkにこの文字を入力する。その様子を図８に示す。一番左の部分は、Stem Networkで最も整合のとれたニューロンとして選んだものにつながっているBranch Networkである。しかし、この中にはこの基本の『基』という文字が学習されていないので、Branch Networkの出力値はどれも小さくなっている。そのため、Stem NetworkとBranch Networkを結合した出力値(CombNETの棒グラフ)もそれほど大きくなっていない。一方、真中のNetworkにおいては、基本の『基』という文字が学習されているため、Branch Networkの出力値の中に非常に値の大きなものがある。このため、結合した出力値は他のものと比べて大きくなり正しく認識されたことになる。

これは第１層における誤りを階層型ネットワークで回復していることになり、全体として汎化能力に優れたネットワークとなっている。このように実用的な規模で実用的な識別能力を持ったネットワークを構築することができた。

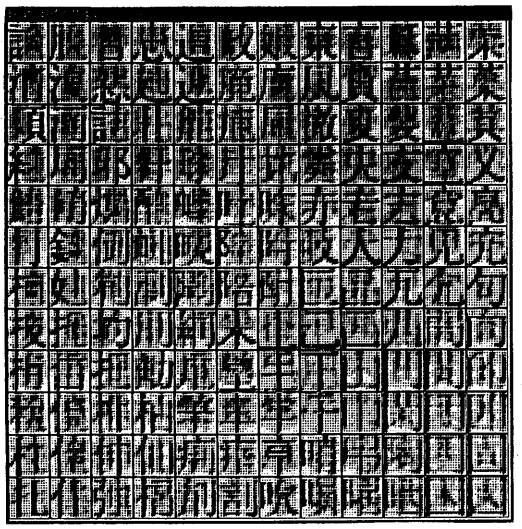


図6．学習後の Stem Network の参照ベクトル

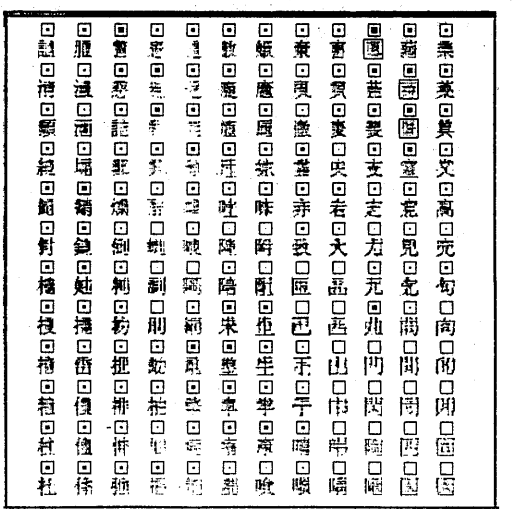


図7.「基」を Stem Network に通した結果

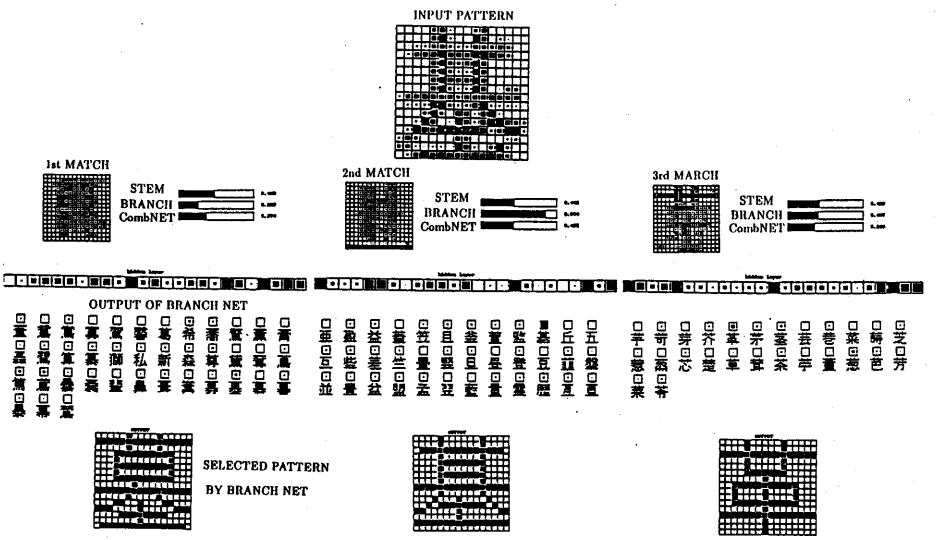


図8.「基」を Stem Network で選んだ Branch Network に通 した結果

6 NEURO TURBOへのCombNETのインプリメント

次に NEURO-TURBO に CombNET をインプリメントした。まず。Stem Network を NEURO-TURBO 上にインプリメントする。ここでは、12×12 の二次元格子状に配置した多数のニューロンを 4 つのグループに分割し、各 DSP は担当するグループのニューロンのみを計算することで並列化を図っている。

次に、Branch Network の前向き計算は、シストリックアレイの考え方と同様にしてリング結合された 4 つの DSP に分割して計算を行わせることができる。そこで、ニューロン間の結合の重みを分割して各ＤＳＰに割り当てる

今回の実験に用いたシステムは、図９に示すように、ホストコンピュータとしてＰＣ－９８０１を用い、印刷漢字の画像データを読み込むためのイメージスキャナと、イメージスキャナから取り込まれた画像データなどを保存しておくためのＲＡＭボードと、ネットワークの計算を行うための NEURO-TURBO から構成されている。

まず、学習時において、NEURO-TURBO 上に Stem Network を構築し、学習データを用いて学習を行う。次に、学習のされた参照ベクトルを用いてカテゴリの分割を行った後、各グループ毎に Branch Network を NEURO-TURBO 上に構築し学習を行う。そしてこの学習によって得られた参照ベクトルの重み及びニューロン間の結合の重みを保存しておく。

また、認識時においては、ネットワークの学習時に生成された Stem Network の参照ベクトル、及び、Branch Network の結合重みを NEURO-TURBO の WM にまず送り込み、NEURO-TURBO 上に Stem Network と 144 個の Branch Network を構築する。そして、新たにイメージスキャナより取り込まれた画像から入力データを作成し、NEURO-TURBO に送り込みネットワークに通して一気に認識を行う。

本システムによる識別率は 先に述べたワークステーション上でのシミュレーションと同様の結果が得られた。また、処理速度については、前段の Stem Network の学習時間は、２９６５文字で１００回の学習を行うのに４７分、後段の Branch Network の学習時間は、1000 回ずつの学習を行なうのにおおよそ 3 時間を要した。認識速度については、Stem Network において一文字当たり 8.1 ミリ秒であり、Branch Network においては 0.4 ミリ秒であった。

この結果から、Stem Network により適合度の高いものを 5 番目まで選び、それを Branch Network 通す場合を考えると、一文字当たり約 10 ミリ秒の時間を要することになる。すなわち、100 文字毎秒の文字識別速度が得られたことになる。

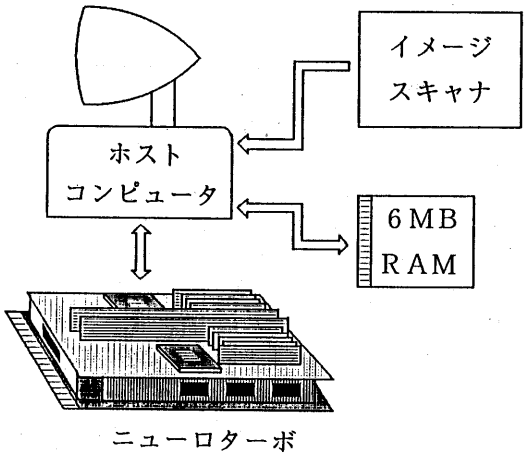


図９．文字認識システム

表２．処理時間

(単位:分)

	Stem Network	Branch Network
学習時間 (NEURO TURBO)	47	約 180
学習時間 (SUN4-260)	1,110	約 1,800
認識時間 (NEURO TURBO)	8.1 ms (123 文字/秒)	0.4 ms (2300 文字/秒)

表３．ネットワーク規模

	Stem Network	Branch Network
ニューロン数	144	43,429
結合数	36,864	228,146

この CombNET は表 3 のようにニューロン数は合計で約 43,500 個、ニューロン間の結合数は約 26 万コネクションである。このような大規模なネットワークにおいてもこのような高速処理能力を持つ専用のハードウェアを用いることにより実用的な速度を得られることが確認できた。

7 まとめ

本論文では、汎用ＤＳＰとして最近開発されたＭＢ８６２２０を４個用いたニューラルネットアクセラレータ、NEURO-TURBO について述べた。NEURO-TURBO は、ＤＳＰ４個をデュアルポートメモリを介してリング状に結合したＭＩＭＤ型並列処理プロセッサである。NEURO-TURBO はバックプロパゲーションの学習時の演算速度で２ＭＣＰＳ、また、前向きの演算速度で１１ＭＣＰＳを得た。この速度は、ＳＵＮ４／２６０ワークステーションの約２０倍であった。 Neuro-Turbo のリング結合アーキテクチャは、プロセッサの数を４以上に増加することができる。その場合、ニューラルネットの演算速度は、プロセッサにほぼ比例して速くなる。

つぎに、NEURO-TURBO に、我々が先に提案した大規模４層ニューラルネット (CombNET) をインプリメントした。その結果、これまでエンジニアワークステーション上で学習時に２～３日、認識時に数文字/秒かかっていた印刷漢字認識の処理時間が、学習時に数時間、認識時には、100 文字/秒と高速化され、実用的な処理速度を得ることができた。

References

[1] Penz, P.A. and Wiggins, R. : Digital Signal Processor Accelerators for Neural Network Simu- lations, Neural Networks for Computing, AIP Conf. Proc. 151, pp.345-355.

[2] Cruz, C.A., Hanson, W.A. and Tam, J.Y.: Neural Network Emulation Hardware Design Considerations, ICNN87, pp.III427-434, 1997.

[3] Pomerleau, D.A., Gusciora, G.L., Touretzky, D.S. and Kung, H.T. : Neural Network Simulation at Warp Speed : How We Got 17 Million ConnectionPer Second, ICNN88 pp. II-143-150, 1988.

[4] 佐藤、岩田、鈴村、松田、吉田：「汎用浮動小数点ＤＳＰによるニューラルネットアクセラレータ」、電子情報通信学会技術研究報告、 ME とバイオサイバネティックス MBE88-134, pp.83-88, 1989

[5] Iwata,A., Yoshida,Y., Matsuda,S., Sato,Y. and Suzumura,N. : An Artificial Neural Network Accelerator Using General Purpose 24 Bits Floating Point Digital Signal Processors, Proc. of Int. Joint Conf. on Neural Networks, Washington D.C., Vol.2, pp171-175, 1989

[6] 岩田、當麻、松尾、鈴村：「大規模４層ニューラルネットの構築手法」、電子情報通信学会技術研究報告、ニューロコンピューティング,NC89-7, pp.37-42, 1989

[7] 當麻、岩田、堀田、松尾、鈴村：「大規模４層ニューラルネット（ＣｏｍｂＮＥＴ）による印刷漢字認識」、電子情報通信学会技術研究報告、ニューロコンピューティング,NC89-39, pp.39-44, 1989

[8] Kohonen,T. : Self-Organization and Associative Memory, Springer-Verlag, 1984 and 1988.

[9] Rummelhart, D.E., Hinton, G.E. and Williams, R.J. : Learning Representations by Back-propa- gating Errors, Nature, 323-9, pp533-536, 1986.