

IV-37

ニューラルネットワークモデルによる運転者支援システムの開発(その2)

北海道大学大学院 学生員 野口智之、中井幸治
北海道大学 正員 中辻 隆、加来照俊

1. はじめに

前回、筆者らはニューラルネットワークモデルによる運転者支援システムの構築を目標として逆伝搬(バックプロパゲーション)法を用い、画像認識の準備段階として、まず手書き文字認識におけるこのモデルの適用性について検討を行った。その結果をもとに本研究では、まず、今回新たに用いたニューラルネットワークのモデルであるKohonen feature mapについて簡単に紹介する。次に、入力データの前処理として用いたWalsh変換について概説する。さらに、数字パターン認識を例にして、逆伝搬法では前処理をした場合としない場合の認識率の比較、およびKohonen feature mapが運転者支援システムにおいて将来的に有効であるか検討する。

2. The Kohonen Feature Map

図1に示すネットワークを例に、Kohonen feature map(以下 KFM と記す) のアルゴリズムについて説明する。KFM は、入力層と競合層から成る2層のネットワークで構成され、教師信号なしで自己組織化を行うモデルである。

$$E = [e_1, e_2, e_3, \dots, e_n]$$

$$U_i = [u_{i1}, u_{i2}, u_{i3}, \dots, u_{in}]$$

e_n は入力パターンベクトルであり、0.0~1.0の大きさの範囲で与えられ、そしてこの入力ベクトルから競合層の i 番目のneuronに結合するウェイト u_{in} は、大きさ 0.5 (1 ± 0.1) の範囲でランダムに配分する。ここで、ベクトル E と U_i の距離は、

$$\|E - U_i\|$$

で表され、競合層内でこの距離が最小となる neuron が、入力に対応する出力となる。すなわち、winner neuron c は次式により決まる。

$$\|E - U_c\| = \min_i \{\|E - U_i\|\} \quad (1)$$

次に、winner neuron 及び周辺の 競合neuron から成る競合面を定め、その面内に存在するすべての neuron の結合ウェイトを調整する。ウェイトの調整量は次式による。

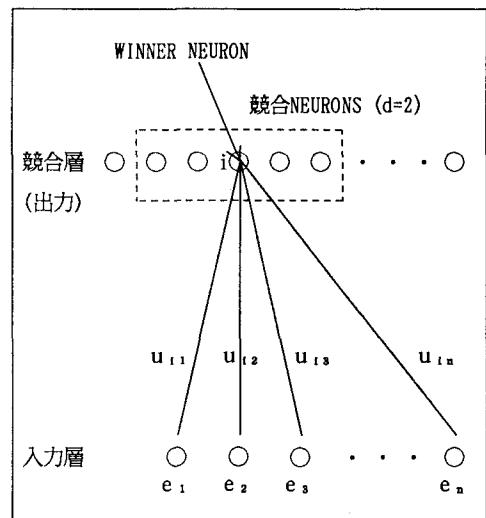


図. 1 THE KOHONEN FEATURE MAP

$$\delta u_{ij} = \alpha (e_j - u_{ij}) \quad (2)$$

$$u_{ij}(\text{NEW}) = u_{ij}(\text{OLD}) + \delta u_{ij} \quad (3)$$

すなわち、KFM の学習とは、ウェイトを入力パターンに近づけるように調整し、距離 $\|E - U_i\|$ を最小にすることに他ならない。そのため、ある入力パターンに反応したwinner neuronは、以後類似したパターンが入力されると、再びwinnerに成り易くなる。

また、 α はウェイトの調整割合を決める変数であり、学習の繰り返しが進むにつれ、次式に従って減少する。

$$\alpha_t = \alpha_0 (1 - t / T) \quad (4) \quad \begin{aligned} \alpha_0 &: 0.2 \sim 0.5 \\ t &: \text{現在の繰り返し回数} \\ T &: \text{規定の繰り返し回数} \end{aligned}$$

winner neuron周辺の競合面の幅 d についても同様に、

$$d = d_0 (1 - t / T) \quad (5) \quad d_0: \text{競合層の幅の } 1/3 \sim 1/2$$

最終的に競合面はwinner neuronのみになり、入力に対応する出力のマッピングを完了する。

3. 入力データの预处理について

本研究では、文字データをそのままネットワークに入力するのではなく、Walsh変換で前処理したものを入力データとして用いた。この変換は、文字データのような0か1の信号しか持たないようなパルス波に対して有効である。また、平行移動などの位置ずれをおこしている文字にもこの変換は威力を発揮する。さらに変換の結果、入力層のニューロン数を減少させることができた。Walsh変換は、変換すべきデータ数が N 個あるとすると、図 2 に示すような、 N 個の基本モードを持ち、その直交性により次式が得られる。

$$x_m = \sum_{k=0}^{N-1} W_k w_k(m) \quad (6)$$

$$W_k = 1 / N \left(\sum_{m=0}^{N-1} x_m w_k(m) \right) \quad (7)$$

ただし、 x_m : 地点 m での基本モードの大きさ、 $w_k(m)$: k 番目の基本モード関数、 W_k : k 番目の基本モード関数の振幅を規定する係数、である。Walsh変換とは式(6)から式(7)への流れのことをいう。

変換前の文字データを示したのが図3の左図であ

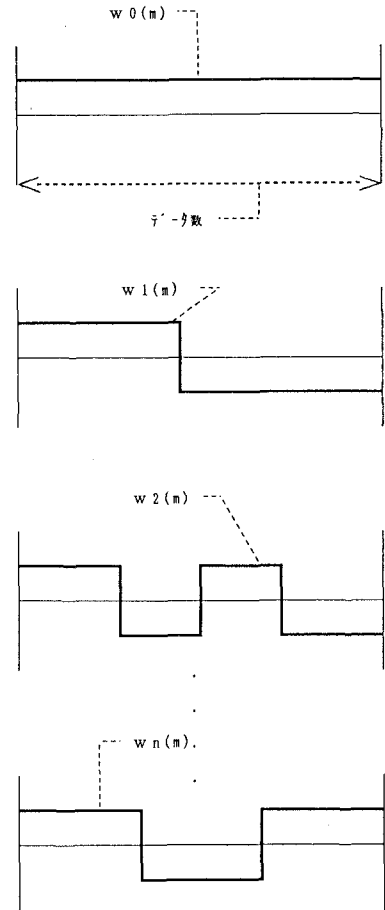


図. 2 基本モードの例

り、これをWalsh変換した後の状態を同図に示す。

なお、2 番目の図は各行ごとに、3 番目の図は各列ごとにWalsh変換をした状態である。本研究ではさらに、各行ごとに変換した場合には各列ごとに総和をとり、各列ごとに変換した場合には各行ごとに総和をとったものを状態量としてネットワークへの入力データとして用いる。ただし、各行列ごとの総和はすべて 0 から 1 の範囲におさまるように正規化しておくものとする。以上に述べた手法を使って、0 から 9 までの文字データを変換した結果を図 4 に示す。両図ともに 1 番目の状態が 0 となっているが、これは各文字データごとの状態量に特徴を持たせるために強制的に 0 にした。

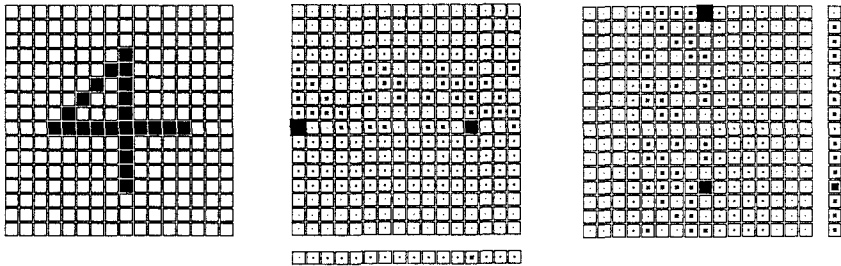


図. 3 データの変換後の状態

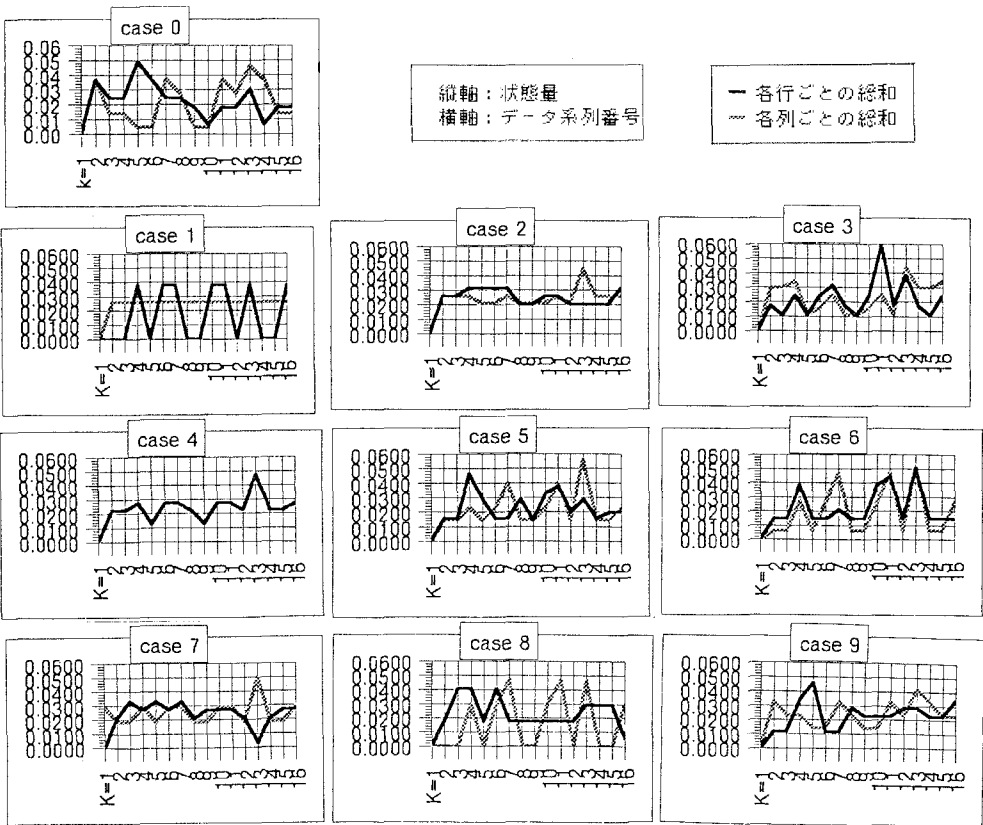


図. 4 各データの変換後の状態

4. 適用例

ここでは、Backpropagation（以下 BPN と記す）と KFM の2つのニューラルネットワークモデルを取りあげ、手書き数字「0」「1」…「8」「9」の10文字を識別対象としてパターン認識のシミュレーションを行った。BPNは、教師信号を持ち、強力な学習・自己組織化能力を有する階層型ニューラルネットの代表的なモデルであり、KFM は教師信号なしの階層型モデルである。両モデルの評価に用いた未学習データは比較的標準と思われる文字であり、1文字につき10パターンずつ用意した。なお、前処理については、BPNに限り前述のWalsh変換を施したデータと無変換のデータを用い、その効果の比較検討を行った。

4. 1 Backpropagation

シミュレーションに用いたネットワークは、出力層、中間層、入力層の3層で構成した。出力層及び中間層にはそれぞれ10個のニューロンを与えた。入力層のニューロン数については、生データを入力する場合は $16 \times 16 = 256$ 個、Walsh変換を施したデータを入力する場合は $15 \times 2 = 30$ 個となる。学習に用いたデータは、10文字 $\times$ 10パターンで計100パターンあり、相似や位置ズレの型を含む変化に富んだ構成である。学習過程においては、どちらの場合についても入出力の誤差が充分小さくなるまで収束を繰り返した。両データによる認識率の比較を図5に示す。

変換なしでは80%であったが、Walsh変換を施した場合には、認識率は58%に低下した。これは、用いた学習データには相似・位置ズレのパターンが含まれており、Walsh変換がこれらのデータの特徴を表現しきれていないことによる。本来、前処理とは1つの方法だけで施すのではなく、画面に占める画像の割合いや、線分の角度など様々な情報を付加して行うものである。その上で、Walsh変換が平行移動に強いことを考慮に入れ、前処理を工夫すれば変形にも対処できるようになると思われる。

4. 2 The Kohonen feature map

ネットワークは図1に示すように中間層を持たず、入力層及び出力層の2層で構成されている。学習データは1文字につき1パターンだけを入力

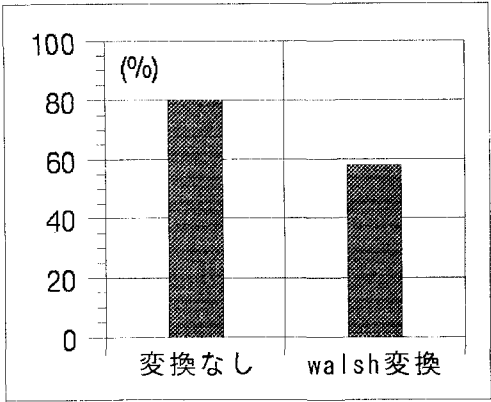


図. 5 入力データによる影響

表. 1 KOHONEN FEATURE MAP の学習パラメータ

	出力層ニューロン数	
	10	20
入力層ニューロン数	256	256
α	0.3	0.2
d	5	7
学習繰返し回数	60,000	60,000
学習時間*	68時間	120時間

* UNIX ワークステーションを使用

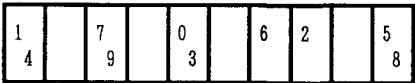


図. 6 文字の分離パターン（ニューロン数10）

している。まず出力層のニューロン数を、識別文字数と対応させて10個とした場合についての分類結果を図6に、併せて学習パラメータをニューロン数を20個とした場合と比較して表1に示す。図6を見ると2つの文字が1つのニューロンに対応しており完全な分離は行われていない。これは、KFMの出力層はBPNとは異なり教師信号なしであること、また競合という操作があることが、文字とニューロンの一対一の対応の妨げとなっていることによる。その結果、出力層のニューロン数は10個では不足であることがわかった。

次に、出力層のニューロン数を20個とした場合の入出力の一覧を表2に示した。学習終了後のニューロンと文字の対応関係は表の上部に図示している。表中の数字は、1文字につき10パターンの未学習データを入力した場合、その入力パターンがどのニューロンに反応したかを回数で表している。学習過程において、”2”と”3”の分離が不完全ではあるが、それでも”6”を除いた未学習データは各文字に対応するニューロンの近傍に出力されている。今後の対策として、出力層のニューロン数をさらに増加させるか、層を線ではなく面として与え、出力層を各文字に対応する面で区切れるようにする方法を考えている。

また、どのニューロンにどの文字が対応するのか、分離に最低限必要となるニューロン数はいくつなのかといった問題は、ニューロン間の結合ウェイトの初期状態や、学習パターンの順序などに左右されるので、これらに制約を与えながら試行錯誤的に求めていくことになる。

表. 2 KOHONEN FEATURE MAP による文字の分離パターン（ニューロン数20の例）

		6	8	2 3	0	9	7	1	4	5
入 力 文 字 10 パ タ ー ン	0				9	1				
	1							3	4	3
	2			3	4	3				
	3			6	2		2			
	4							2	1	
	5			1					1	6
	6	1		5	1					8
	7				1		2	3	4	2
	8		1	1	6	1	1			
	9			1		1	6	2		

5. まとめ

今回は、BPN と KFM による画像の識別、及びWalsh変換による前処理についてのシミュレーションを行いながら、ニューラルネットワークモデルの運転支援システムへの適用性を検討してきた。

KFM は、入力パターンベクトルが2次元程度の問題についてはよい報告が聞かれるが、実際の車載カメラから送られてくる画像は、今回取り上げた256次元の文字データよりも多次元で複雑であるので、KFMを適用するには単体では限界があり、他のモデルと組み合わせるなどの工夫が必要である。

Walsh変換は、データによってはその特徴を表現しきれないという欠点を現したものの、平行移動に強いという長所を活かした入力データの表現ができれば改善は行えるであろう。そうなれば、もう1つの長所であ

る変換によるデータ圧縮はニューラルネットワークのニューロン数を大幅に削減することができ、このことは学習速度の向上やリアルタイム処理に利用できる。解決すべき課題を多く抱えているが、ニューラルネットワークの柔軟性や長所を積極的に取り入れて、システムの構築をめざしていきたい。

参考文献

- 1) Philip D. Wasserman : 「 NEURAL COMPUTING 」 VAN NOSTRAND REINHOLD, 1989
- 2) Judith Dayhoff : 「 Neural Network Architectures 」 VAN NOSTRAND REINHOLD, 1990