

多層競合ネットによる不変パターン認識に関する研究

A Study of Invariant Pattern Recognition by Multi-Layered Competitive Nets

九州工業大学大学院 工学研究科

博士後期課程 設計生産工学専攻



西田 健

平成14年1月

九州工業大学附属図書館



0010471381

概要

現在までに提案されている2次元座標変換に対する不変パターン認識を行うニューラルネットのアーキテクチャは基本的にアフィン変換に含まれる座標変換に対する不変認識を扱っており、パターン全体に施される非線形座標変換に対する不変認識を行うニューラルネットはこれまでに提案されていない。そこで本論文では線形および非線形変換に不変なパターン認識を行う多層競合ネットワークを提案する。また非線形座標変換の具体例として射影変換を用い、提案するネットが射影変換に対する高い不変認識能力を獲得することをシミュレーションにより示す。さらに本論文では、このネットの学習アルゴリズムとして用いるベクトル量子化 (VQ: Vector Quantization) アルゴリズムを新たに提案する。従来から提案されている数多くのVQアルゴリズムは、収束が遅い、任意の確率分布をもつ入力に対しては局所解に陥る、時間的に変化する入力ベクトルの分布に適応できないなどの問題がある。提案アルゴリズムは、入力ベクトルに対する荷重ベクトルの歪を逐次的に観察し歪の大きな荷重ベクトルを再初期化するという手法と、局所最適化を行う勾配法を統合したものである。本論文では、入力分布が時間変化する2次元ベクトルや、画像などの高次元ベクトルを用いたシミュレーションにより、提案アルゴリズムが上述の問題点を解消し従来のVQアルゴリズムよりも高いVQ性能を持つことを示す。

目次

第1章 序論	1
1.1 背景と目的	1
1.2 論文構成	3
第2章 座標変換に不変なパターン認識を行う多層競合ネット	5
2.1 はじめに	5
2.2 Jacobian を用いた座標変換に対して不変なパターン認識	5
2.2.1 一般の座標変換に対する不変認識	6
2.2.2 射影変換とその Jacobian	9
2.3 ネットワークの構造	11
2.4 エネルギー関数と学習	13
2.5 シミュレーション	15
2.5.1 使用するパターンセット	15
2.5.2 射影変換に対する不変認識	15
2.5.3 回転変換に対する TC 汎化能力	18
2.6 まとめ	21
第3章 再初期化法を用いる適応ベクトル量子化アルゴリズム	22
3.1 はじめに	22
3.2 勾配法と局所解問題	23
3.3 再初期化法を用いた適応 VQ	25
3.3.1 等歪条件	25

3.3.2	再初期化法	25
3.3.3	各パラメタの決定法	27
3.4	2次元ベクトルへの適用	29
3.4.1	一定時間区間ごとに変化する入力分布	29
3.4.2	連続的に変化する入力分布	30
3.4.3	異なるパラメタ値に対する性能	31
3.5	高次元ベクトルへの適用	34
3.5.1	画像データのコーディング	34
3.5.2	回轉變換パターンの学習	37
3.6	まとめ	38
第4章	結論と今後の課題	40
4.1	結論	40
4.2	今後の課題	41
	謝辞	42
	参考文献	43

第1章 序論

1.1 背景と目的

現在までに提案されている2次元平面パターンの様々な変形に対して不変な認識をニューラルネットを用いて実現しようとする代表的な研究として文献 [1]-[5] などがある。これらの研究は、2次元画像理解のための包括的研究 [6] や実用的文字認識を目指す大規模ニューラルネットの研究 [7] と異なり、より基本的な不変認識の原理やアーキテクチャをその学習法とともに検討することに特徴がある。

これまでに提案されている不変認識を行うアーキテクチャとその機能には次のようなものがある。福島の新オコグニトロン [1] は視覚系を模倣した多層神経回路モデルを用いて位置ずれや局所的な変形に対して不変に認識することができる。Widrowらの多層ネット [2] は、ADALINEと呼ばれる素子からなるインバリエンスネットと多層パーセプトロンからなり、平行移動、回転、拡大縮小や投影（水平方向のみの拡大縮小）に対する不変性を得るためのものである。GilesとMaxwell[3]やPerantonisとLisboa[4]のHONN (High-Order Neural Network) は、画素間の高次多項式で実現できる不変性を用いて平行移動、回転および拡大縮小に不変な認識を行う手法である。KonenらのDLA (Dynamic Link Architecture)[5]は多層ネットの層間の結合を動的に自己組織化することにより平行移動、回転、鏡面对称および歪みに対して不変認識を行う。一方、Kurogiはこれらのネットとは異なるアーキテクチャをもつネットとして、アフィン変換に不変な認識を行う多層競合ネット [8] を提案している。このネットは構造的にはWTA (Winner-Take-All) 機能をもつ層を用いる点で新オコグニトロンに類似しており、クラスタと呼ばれるセルの集合ごとに同値なパターンを記憶させて不変性を実現するという点でslabと呼ばれるセルの集合を用いるWidrowらの多層ネットに類似している。また不変認識を行う上述のすべてのニューラルネットは、その特徴的な学習機能として次のような汎化能力をもつ：ま

いくつかの例を用いて不変性を学習させ、それを未知のパターンの不変認識に適用させることができる。さらに多層競合ネットは、ある変換のクラス（正確にはアフィン変換で構成した群）に関する不変性と識別能力を汎化学習する能力をもつことが示されている [8]。より具体的には、この能力は TC 汎化（Generalization of Training Classes）能力と呼ばれ、つぎのように定義される：いくつかの訓練パターンをあるクラスに属するすべての変換により変形し学習させると、このクラスに属する変換のみに対する不変認識を未知のテストパターンに対して行うことができる。しかし上述のネットは入力パターンの部分的な歪に対する頑健性を持つものもあるが、いずれも基本的にはアフィン変換に含まれる座標変換に対する不変認識を扱っており、パターン全体に施される非線形座標変換に対する不変認識を行うニューラルネットのアーキテクチャはこれまでに提案されてない。そこで本論文では、上述の TC 汎化能力をもつ従来の多層競合ネットが適用可能な変換のクラスを一般の座標変換群にまで拡張し、線形および非線形な座標変換群内の変換に不変な認識を行うことのできるネットを提案する。

ところで、非線形変換に対する不変認識の研究として Kanatani[9] はカメラの回転変換（射影変換集合の部分集合の要素）に対する不変性について 3次元シーンの画像復元の観点から研究を行っている。しかしそこで扱っているのは 3次元シーンを構成する要素（点、線、テンソル、対称多項式、オプティカルフローなど）に対する不変性とその性質を用いた画像復元であり、本稿で扱うような画像パターン全体を一度に処理して不変認識を行う手法ではない。なお、本研究で不変性のために座標変換群を考えるのは、座標変換前後でパターンが属するクラスが同一でなければパターン分類が正しく定義できないからであるが、この問題設定は Kanatani[9] が指摘するように「幾何学的概念は座標変換群に関する不変性の性質の抽象化である」ことに関係しており、幾何学的パターンを扱う観点から一般性があると考えられる。

さて、上述した TC 汎化能力を獲得するために、ネットは数種類の訓練パターンをあるクラスに属するすべての変換により変形し学習させる必要があるが、一般にこの変換パターンの数はネットの記憶容量よりも非常に多く、全てを記憶することはできない。そこで提案ネットは訓練パターンの学習にベクトル量子化 (VQ: Vector Quantization) アルゴリズムを用いる。すなわち、VQ アルゴリズムにより各セルの荷重ベクトルは入力ベクトルとの誤差を最小化するように更新され、ネットは有限の記憶容量に対して非常に多く存在する訓練パターンを効率良く

学習する。しかし、現在までに数多く提案されている VQ アルゴリズムには、入力信号の確率分布によっては局所解におちいる、収束が遅い、設定すべきパラメタが多いなどの問題があり [27]、提案ネットが座標変換に対して高い不変認識能力を獲得するためにはこれらの問題を解決する必要がある。そこで本論文では、提案ネットの学習アルゴリズムとして上述の問題を解決するための VQ アルゴリズムを新たに提案する。

1.2 論文構成

前節で述べたように、本論文では 2 次元平面パターンの線形および非線形座標変換に不変な認識を行う多層競合ネットを提案し、さらにこのネットの学習アルゴリズムとして用いる VQ アルゴリズムを新たに提案する。ただし後者の VQ アルゴリズムは競合ネットの学習アルゴリズムとして用いられるだけでなく、画像圧縮や情報伝送における情報量の圧縮など、より広範囲な分野での応用が可能であるため、本論文では前者のネットに関する提案とは独立して述べることにする。

本論文の構成を以下に示す。まず第 2 章で一般の座標変換に対する不変性を獲得するための原理と、それを実現するための多層競合ネットの構成について述べる。また、非線形変換の具体例として射影変換施した文字パターンを用い、このネットの認識シミュレーションを行う。次に第 3 章において、再初期化法を用いる適応ベクトル量子化アルゴリズムの原理について述べ、従来アルゴリズムとの性能比較を行う。そして最後に第 4 章で結論を述べる。本論文の構成を Fig.1.1 に示す。図中の破線は、その破線で指された節がもう一方の章と関連することを示している。すなわち、2.4 項と 2.5 項は第 3 章に述べるアルゴリズムを用い、3.5 項は第 2 章で述べるネットを用いてシミュレーションを行う。

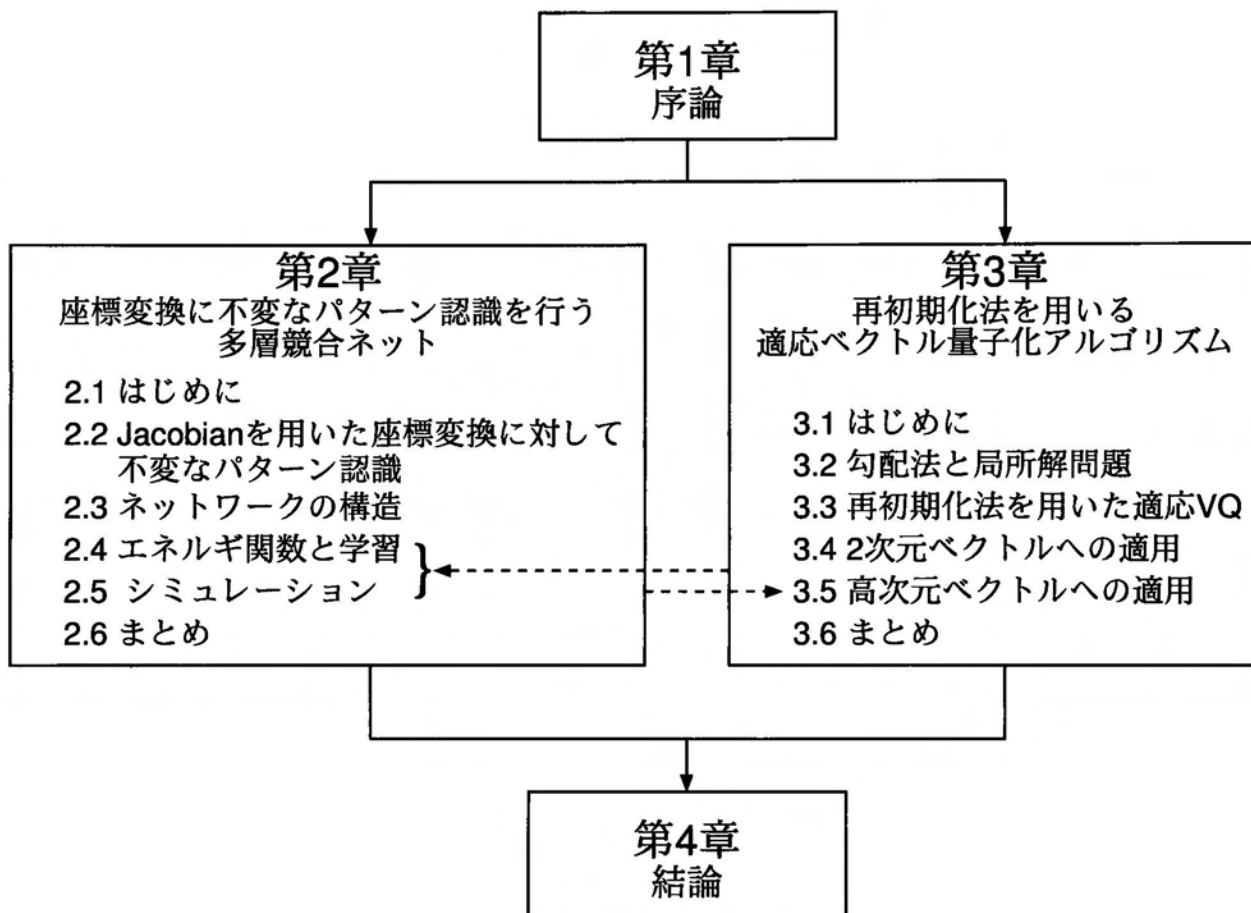


Fig. 1.1: Relations between the chapters.

第2章 座標変換に不変なパターン認識を行う多層競合ネット

2.1 はじめに

本章では、まず 2.2 で一般の座標変換に対する不変パターン認識の方法を示し、その後非線形座標変換の例として射影変換について述べる。さらに、2.3 でネットの構造および 2.4 でその学習について述べる。最後に構成したネットの不変認識と TC 汎化能力についてのシミュレーション結果を 2.5 に示す。

2.2 Jacobian を用いた座標変換に対して不変なパターン認識

2次元空間 $x = (x, y)^T \in \mathbb{R}^2$ 上でパターン $p_i(x)$ 、および座標変換 $g(x) = (g_1(x), g_2(x))^T$ により変換したパターン $p_i(g(x))$ を考える。ここで $i = 1, 2, \dots$ は区別すべきパターンの番号、 $(x, y)^T$ は行ベクトル (x, y) の転置すなわち列ベクトルを表す。またここで考えるパターンは文字などの濃度や輝度のパターンであり、 $p_i(g(x))$ は非負かつ有限であり、あるパターン $p_i(g(x))$ とその定数倍のパターン $ap_i(g(x))$ ($a > 0$) は区別しない。さらに $p_i(g(x)) \neq 0$ となるすべての点 x は変換前後で、ある与えられた正方領域 $D = \{(x, y)^T \mid |x| \leq \Delta_D, |y| \leq \Delta_D\}$ の中にあるとする。

以下に、従来の線形変換に対する不変認識の手法 [8] を拡張し、任意の座標変換群 G 内の変換 g に対して不変認識を行うための手法を示す。ここで、線形変換を $g(x) = Ax$ 、アフィン変換を $g(x) = Ax + b$ ($A \in \mathbb{R}^{2 \times 2}, b \in \mathbb{R}^2$) として区別し解析する場合があるが本論文ではこれらの用語を区別せずに用いる。なお、以下では恒等変換を $e(x) = x$ とし、簡単のため $p_i(x)$ や $g(x)$ などを p_i や g などとも書く。

2.2.1 一般の座標変換に対する不変認識

まず座標変換 $g(\mathbf{x}) \in G$ の Jacobian

$$|\mathbf{g}_x| = \left| \frac{\partial \mathbf{g}}{\partial \mathbf{x}} \right| = \frac{\partial g_1}{\partial x} \frac{\partial g_2}{\partial y} - \frac{\partial g_1}{\partial y} \frac{\partial g_2}{\partial x} \quad (2.2.1)$$

を訓練パターン $p_i(\mathbf{g}(\mathbf{x}))$ に乗じたパターン

$$p_i \mathbf{g}(\mathbf{x}) = p_i(\mathbf{g}(\mathbf{x})) |\mathbf{g}_x| \quad (2.2.2)$$

をすべての $\mathbf{g} \in G$ について作成し、それらを正規化したパターンの集合 $P_{iG} = \{p_i \mathbf{g} / \|p_i \mathbf{g}\| \mid \mathbf{g} \in G\}$ を記憶する。ここで $\|p(\mathbf{x})\| = (\int p(\mathbf{x})^2 d\mathbf{x})^{1/2}$ はパターン $p(\mathbf{x})$ のノルムを表す。また訓練パターン $p_i(\mathbf{x})$ の番号 i の集合を $I = \{1, 2, \dots\}$ とし、その要素数 $|I|$ は有限であるとする。さらに、パターンの非零値が存在する正方領域 D 内で $|\mathbf{g}_x|$ は非零、有限かつ連続であるとする。ここで座標変換 $z = \mathbf{g}(\mathbf{x})$ が面積 S_x の微小領域 D_x を面積 S_z の微小領域 D_z に写すとする

$$S_z = \int_{D_z} dz = \int_{D_x} |\mathbf{g}_x| d\mathbf{x} \simeq |\mathbf{g}_x| S_x \quad (2.2.3)$$

より

$$|\mathbf{g}_x| \simeq S_z / S_x \quad (2.2.4)$$

となることを注意しておく。これにより式 (2.2.2) の記憶パターン $p_i \mathbf{g}(\mathbf{x})$ は座標変換により拡大 (縮小) された部分のパターンの値を小さく (大きく) するように変換パターン $p_i(\mathbf{g}(\mathbf{x}))$ を修正していることが分る。

次に、テストパターン $p_j(\mathbf{x})$ を $a (> 0)$ 倍し、座標変換 $\mathbf{h}(\mathbf{x}) \in G$ を施した $ap_j(\mathbf{h}(\mathbf{x}))$ と $p_i \mathbf{g} / \|p_i \mathbf{g}\| \in P_{iG}$ との内積の最大値

$$s_{j(a, \mathbf{h})i} = \max_{\mathbf{g} \in G} \left\{ \int ap_j(\mathbf{h}(\mathbf{x})) \frac{p_i(\mathbf{g}(\mathbf{x})) |\mathbf{g}_x|}{\|p_i(\mathbf{g}(\mathbf{x})) |\mathbf{g}_x|\|} d\mathbf{x} \right\} \quad (2.2.5)$$

を計算する。ここで $j \in J$ であり、 J はテストパターン $p_j(\mathbf{x})$ の番号 j の集合である。すると $\mathbf{h}(\mathbf{x}) \in G$ であるから G の中には $z = \mathbf{h}(\mathbf{x})$ の逆変換が存在するので、 $\mathbf{g}(\mathbf{x}) = \mathbf{g}(\mathbf{h}^{-1}(z)) = \mathbf{f}(z)$

および $|\mathbf{f}_z| = |\partial \mathbf{g} / \partial \mathbf{x}| \cdot |\partial \mathbf{x} / \partial \mathbf{z}| = |\mathbf{g}_x| \cdot |\mathbf{h}_x|^{-1}$ を用いると,

$$\begin{aligned}
s_{j(a, \mathbf{h})_i} &= \max_{\mathbf{g} \in G} \left\{ \int a p_j(\mathbf{h}(\mathbf{x})) \frac{p_i(\mathbf{g}(\mathbf{x})) |\mathbf{g}_x|}{\|p_i(\mathbf{g}(\mathbf{x})) \mathbf{g}_x\|} d\mathbf{x} \right\} \\
&= \max_{\mathbf{f} \in G} \left\{ \int a p_j(\mathbf{z}) \frac{p_i(\mathbf{f}(\mathbf{z})) |\mathbf{f}_z| |\mathbf{h}_x^{-1}|^{-1}}{\|p_i(\mathbf{f}(\mathbf{z})) \mathbf{f}_z\| |\mathbf{h}_x^{-1}|^{-1}} |\mathbf{h}_x^{-1}| dz \right\} \\
&= \max_{\mathbf{f} \in G} \left\{ \int a p_j(\mathbf{z}) \frac{p_i(\mathbf{f}(\mathbf{z})) |\mathbf{f}_z| |\mathbf{h}_x|}{\left(\int \left(p_i(\mathbf{f}(\mathbf{z})) |\mathbf{f}_z| |\mathbf{h}_x| \right)^2 |\mathbf{h}_x^{-1}| dz \right)^{1/2}} |\mathbf{h}_x^{-1}| dz \right\} \\
&= \max_{\mathbf{f} \in G} \left\{ \int a p_j(\mathbf{z}) \frac{\|p_i(\mathbf{f}(\mathbf{z})) \mathbf{f}_z\|}{\left(\int \left(p_i(\mathbf{f}(\mathbf{z})) |\mathbf{f}_z| |\mathbf{h}_x| \right)^2 |\mathbf{h}_x^{-1}| dz \right)^{1/2}} \cdot \frac{p_i(\mathbf{f}(\mathbf{z})) |\mathbf{f}_z|}{\|p_i(\mathbf{f}(\mathbf{z})) \mathbf{f}_z\|} dz \right\} \\
&= \max_{\mathbf{f} \in G} \left\{ a \left(\frac{\|p_i(\mathbf{f}(\mathbf{z})) \mathbf{f}_z\|^2}{\int \left(p_i(\mathbf{f}(\mathbf{z})) |\mathbf{f}_z| \right)^2 |\mathbf{h}_x| dz} \right)^{1/2} \int p_j(\mathbf{z}) \frac{p_i(\mathbf{f}(\mathbf{z})) |\mathbf{f}_z|}{\|p_i(\mathbf{f}(\mathbf{z})) \mathbf{f}_z\|} dz \right\} \\
&= \max_{\mathbf{f} \in G} \left\{ a \left(\int \left(\frac{p_i \mathbf{f}(\mathbf{z})}{\|p_i \mathbf{f}(\mathbf{z})\|} \right)^2 |\mathbf{h}_x| dz \right)^{-1/2} \int p_j(\mathbf{z}) \frac{p_i \mathbf{f}(\mathbf{z})}{\|p_i \mathbf{f}(\mathbf{z})\|} dz \right\} \\
&= \max_{\mathbf{f} \in G} \left\{ \frac{a}{\langle |\mathbf{h}_x| \rangle_i^{1/2}} \int p_j(\mathbf{z}) \frac{p_i \mathbf{f}(\mathbf{z})}{\|p_i \mathbf{f}(\mathbf{z})\|} dz \right\} \tag{2.2.6}
\end{aligned}$$

となる。ここで

$$\langle |\mathbf{h}_x| \rangle_i = \int \left(\frac{p_i \mathbf{f}(\mathbf{z})}{\|p_i \mathbf{f}(\mathbf{z})\|} \right)^2 |\mathbf{h}_x| dz \tag{2.2.7}$$

である。以下、この値がいくつかの仮定のもとで i に対してほぼ一定となることを示す。まず、式 (2.2.6) の右辺の最大値を与える変換 $\mathbf{f} \in G$ を \mathbf{f}^* とし、パターン $p_{i\mathbf{f}^*}(\mathbf{z})$ の非零領域を $D_{i\mathbf{f}^*} = \{\mathbf{z} | p_{i\mathbf{f}^*}(\mathbf{z}) \neq 0\}$ 、パターン $p_j(\mathbf{z})$ の非零領域を $D_j = \{\mathbf{z} | p_j(\mathbf{z}) \neq 0\}$ とする。さらに、通常の文字認識において仮定される平行移動や拡大縮小などを含む合成変換が G に含まれるとすると、式 (2.2.6) の最大値を求める操作により D_j と $D_{i\mathbf{f}^*}$ はかなり大きな共通部分をもつことになる。このとき変換 $\mathbf{h}(\mathbf{x})$ が線形 $\mathbf{z} = \mathbf{h}(\mathbf{x}) = A\mathbf{x} + \mathbf{b}$ ならば、 $\langle |\mathbf{h}_x| \rangle_i = |A|$ は i に無関係な定数になる。変換 $\mathbf{h}(\mathbf{x})$ が非線形ならば、 $\int (p_i \mathbf{f}(\mathbf{z}) / \|p_i \mathbf{f}(\mathbf{z})\|)^2 dz = 1$ であるから、式 (2.2.7) より $\langle |\mathbf{h}_x| \rangle_i$ は $D_{i\mathbf{f}^*}$ における $|\mathbf{h}_x|$ の最小値と最大値の中間値をとる。よって $D_{i\mathbf{f}^*}$ と D_j が大きな共通部分をもちかつ変換 $\mathbf{h} \in G$ が連続であるという条件のもとでは、 $\langle |\mathbf{h}_x| \rangle_i$ は D_j の中心付近における $|\mathbf{h}_x|$ の値に近い値をもち i に対するばらつきはかなり小さくなると考えられる。

そこで i に無関係な定数 $\alpha_{\mathbf{h}}$ を用いて $\langle \|\mathbf{h}\mathbf{x}\|_i \rangle^{1/2} \simeq \alpha_{\mathbf{h}}$ とおくと、式 (2.2.6) は

$$\begin{aligned} s_{j(a,\mathbf{h})i} &\simeq \left(\frac{a}{\alpha_{\mathbf{h}}} \right) \max_{\mathbf{f} \in G} \left\{ \int p_j(z) \frac{p_i \mathbf{f}(z)}{\|p_i \mathbf{f}(z)\|} dz \right\} \\ &= \left(\frac{a}{\alpha_{\mathbf{h}}} \right) s_{ji} \end{aligned} \quad (2.2.8)$$

となる。ここで s_{ji} は変換前のテストパターン $p_j(z)$ と $p_i \mathbf{g} / \|p_i \mathbf{g}\| \in P_{iG}$ との内積の最大値、すなわち $\mathbf{h}(x)$ として恒等変換 $e(x) = x$ を用いたときの $s_{j(1,e)i}$ である。すると $s_{j(a,\mathbf{h})i}$ を $i \in I$ について並べたベクトル

$$\mathbf{s}_{j(a,\mathbf{h})} = (s_{j(a,\mathbf{h})1}, s_{j(a,\mathbf{h})2}, \dots, s_{j(a,\mathbf{h})|I|}) \quad (2.2.9)$$

を正規化したものは、

$$\frac{\mathbf{s}_{j(a,\mathbf{h})}}{\|\mathbf{s}_{j(a,\mathbf{h})}\|} \simeq \frac{(s_{j1}, s_{j2}, \dots, s_{j|I|})}{\|(s_{j1}, s_{j2}, \dots, s_{j|I|})\|} = \frac{\mathbf{s}_j}{\|\mathbf{s}_j\|} \quad (2.2.10)$$

となり、 a および \mathbf{h} に関して不変な値をもつ。

さて、このベクトルを用いて不変認識を行うには不変性だけでなく異なるパターンを識別する能力も必要である。このための前提条件として、あるパターン番号 $k \in I \cup J$ をもつパターンが G に属する座標変換およびパターン値の定数倍により他のパターン番号をもつパターンになることはないとする。これは $k, l \in I \cup J$ に対して

$$\max_{\mathbf{g}, \mathbf{h} \in G} \left\{ \int \frac{p_k(\mathbf{g}(\mathbf{x}))}{\|p_k(\mathbf{g}(\mathbf{x}))\|} \frac{p_l(\mathbf{h}(\mathbf{x}))}{\|p_l(\mathbf{h}(\mathbf{x}))\|} d\mathbf{x} \right\} \leq 1 \quad (2.2.11)$$

の等号が成立するのは $k = l$ のときに限ることを意味する。すると、以下のように、テストパターンに非常に良く似たパターンを記憶することにより識別能力が得られる。すなわち、極端な場合、 $I = J$ 、すなわちテストパターンをすべて記憶すれば、式 (2.2.6) の積分の最大値は式 (2.2.11) の条件から、 $i = j$ および $\mathbf{f}(z) = e(z) = z$ のとき得られ、その値は $s_{j(a,e)j} = a \|p_j(z)\|$ になる。従ってテストパターン $a p_j(\mathbf{h}(\mathbf{x}))$ に対する式 (2.2.10) のベクトルはその第 j 要素のみが最大値をとるので、このパターンを識別できることになる。

以上は訓練パターンに内在する変換群 G のみに対する不変認識、すなわち TC 汎化による不変認識を行うための本手法の基本原理であり、第 j パターンを認識するには $\mathbf{s}_j / \|\mathbf{s}_j\|$ だけを

記憶しておけばよいことが分かる。これにより各パターンが正しく不変認識されるかどうかは式(2.2.8)における不変性のための近似と式(2.2.11)左辺のパターン間の類似度との関係に依存する。

一方、実際の文字認識などでは式(2.2.8)により十分な不変性が得られないような変形文字や異なる書体文字を用いる場合もある。そのような場合にはそれらのパターン $ap_k(\mathbf{h}(\mathbf{x}))$ に対する $s_{k(a, \mathbf{h})} / \|s_{k(a, \mathbf{h})}\|$ も第 j パターンとして記憶することが必要となる。すなわち第 j パターンとして認識すべきパターンでありながら $s_j / \|s_j\|$ を用いて認識できない場合は、それらに対する $s_{k(a, \mathbf{h})} / \|s_{k(a, \mathbf{h})}\|$ も記憶する(追加学習する)ことにより所望の不変認識が可能になると考えられる。

2.2.2 射影変換とその Jacobian

射影変換 $g(\mathbf{x})$ は一般に次の式で表される非線形座標変換である [12]。

$$g(\mathbf{x}) = \frac{A\mathbf{x} + \mathbf{b}}{\mathbf{p}^T \mathbf{x} + 1} \quad (2.2.12)$$

ここで、

$$A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}, \mathbf{b} = \begin{pmatrix} e \\ f \end{pmatrix}, \mathbf{p} = \begin{pmatrix} p \\ q \end{pmatrix} \quad (2.2.13)$$

であり、 $a, b, c, d, e, f, p, q \in \mathbb{R}$ である。なお $p = q = 0$ とすると射影変換は線形変換となることが分る。また射影変換されたパターン $p_i(g(\mathbf{x})) = p_i(\mathbf{z})$ は射影変換により変換前の点 \mathbf{z} が変換後に $\mathbf{x} = (\mathbf{z}\mathbf{p}^T - A)^{-1}(-\mathbf{z} + \mathbf{b})$ に写る。例えば、式(2.2.13)の各パラメタが $a = 1, d = 1, p = 0.03$ でそのほかが零のとき、Fig.2.1(a) のパターンは同図(b) のパターンに射影される。 $p_i(\mathbf{z}) = p_i(g(\mathbf{x}))$ により点 $\mathbf{z} = (x', y')^T$ は点 $\mathbf{x} = g^{-1}(\mathbf{z}) = (x', y')^T / (1 - 0.03x')$ に写る。なおこの例および後述するシミュレーションでは Fig.2.1(a) と同程度の大きさのパターンが正方領域 $D = \{(x, y)^T \mid |x| \leq 30, |y| \leq 30\}$ 内に存在するように $p = 0.03$ 等のパラメタの上限値が決定されたことを注意しておく。

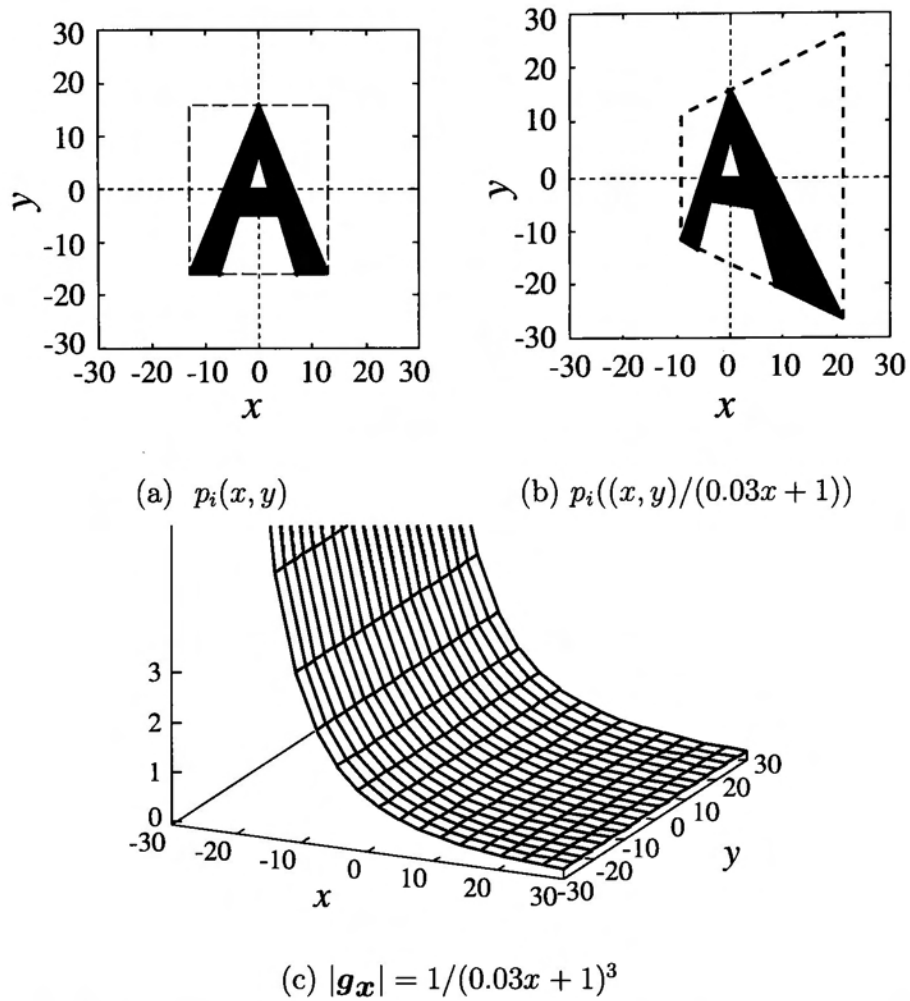


Fig. 2.1: Example of the relation between a projective transformation and the Jacobian. The pattern values at white and dark areas of (a) and (b) are zero and non-zero constant, respectively.

次に、式(2.2.12)で与えられる射影変換 $g(\mathbf{x})$ の Jacobian は、

$$|g_{\mathbf{x}}| = \frac{|M|}{(\mathbf{p}^T \mathbf{x} + 1)^3} \quad (2.2.14)$$

となる。ここで M は射影変換行列

$$M = \begin{pmatrix} A & \mathbf{b} \\ \mathbf{p}^T & 1 \end{pmatrix} \quad (2.2.15)$$

である。従って、例えば前述の例で与えられる射影変換に対する Jacobian は Fig.2.1(c) のようになる。

2.3 ネットワークの構造

ここで用いるネットはアフィン不変認識を行う多層競合ネット [8, 10, 11] と同じく Fig.2.3 に示す3層構造をもつ (ただし Fig.2.3 は認識時の様子を表し、学習時の信号はこれとは異なることに注意)。第2層と第3層は複数のクラスタから成り、各クラスタは複数の競合セル (competitive cell) と1つの統合セル (integration cell) をもつ。

まず前節で考えた連続空間 \mathbb{R}^2 のかわりに $N \times N$ 画素の離散空間を考える。すなわち実際のネットは有限の容量しか持たないため、第1層においては \mathbb{R}^2 空間における入力パターンを N^2 次元ベクトルに離散化して出力する。ここで訓練パターン $p_i(g(\mathbf{x}))$ の学習時には式(2.2.2)を離散化した

$$\mathbf{p}_i^{(2)} = (p_i g(\mathbf{x}_1), p_i g(\mathbf{x}_2), \dots, p_i g(\mathbf{x}_{N^2}))^T \quad (2.3.1)$$

を出力し、テストパターン $a p_j(\mathbf{h}(\mathbf{x}))$ の入力時にはこのパターンをそのまま離散化した

$$\mathbf{p}_{j(a, \mathbf{h})}^{(2)} = a (p_j(\mathbf{h}(\mathbf{x}_1)), p_j(\mathbf{h}(\mathbf{x}_2)), \dots, p_j(\mathbf{h}(\mathbf{x}_{N^2})))^T \quad (2.3.2)$$

を出力するとする。

次に第2層の第 i ($i \in I$) クラスタは、そのクラスタ内の競合セルの荷重ベクトルの集合 $W_i^{(2)} = \{\mathbf{w}_{ij}^{(2)} \mid j \in K^{(2)}\}$ が第 i 番目の訓練パターン $p_i(g(\mathbf{x}))$ に対する第1層の出力ベクトルを正規化した集合 $P_{iG}^{(2)} = \{\mathbf{p}_i g / \|\mathbf{p}_i g\| \mid g \in G\}$ にほぼ等しくなるように学習する (学習の詳細は2.4を

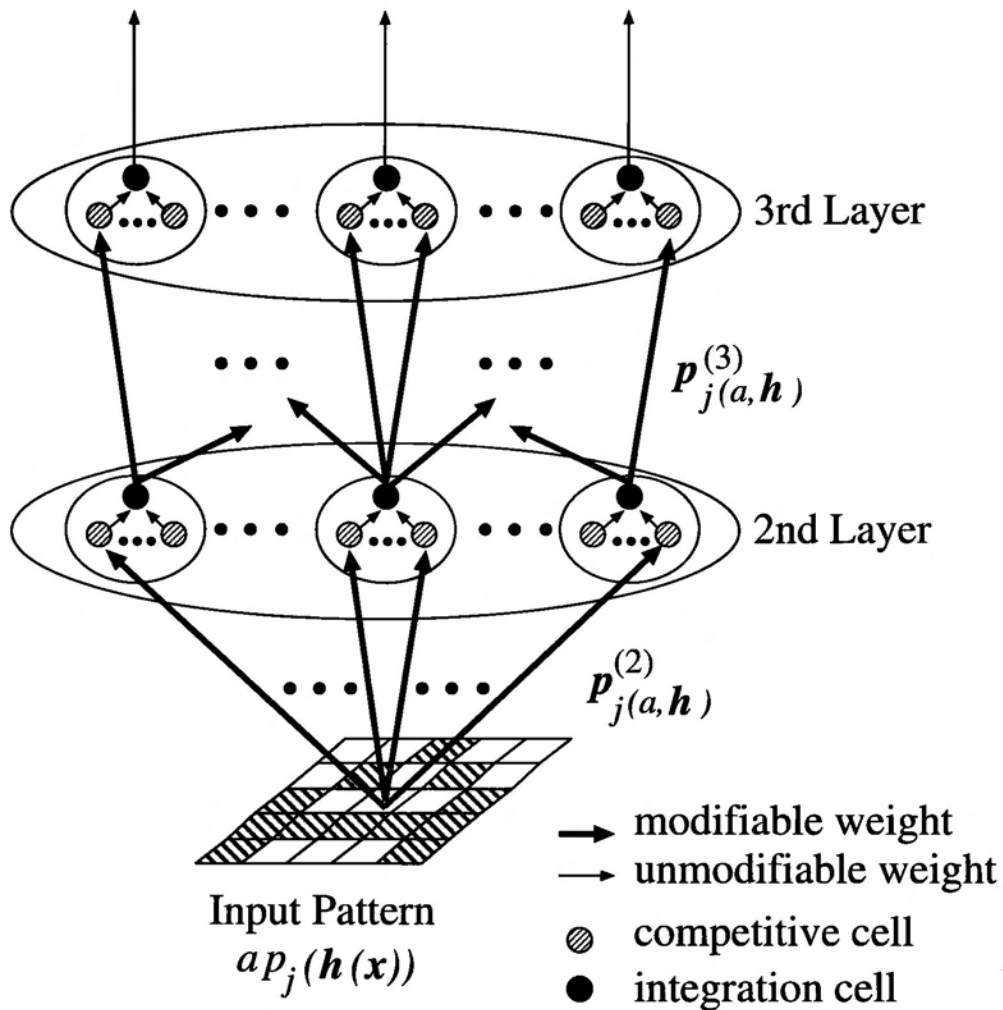


Fig. 2.2: Three layered competitive net, where $ap_j(\mathbf{h}(x))$, $p_{j(a,h)}^{(2)}$ and $p_{j(a,h)}^{(3)}$ represent the signals appearing through the recognition process.

参照のこと)。ここで $K^{(2)} = \{1, 2, \dots, |K^{(2)}|\}$ は第2層の各クラスタ内の競合セルの番号の集合である。すると学習後のネットにテストパターン $ap_j(\mathbf{h}(\mathbf{x}))$ が入力されると、各競合セルは荷重ベクトル $\mathbf{w}_{ij}^{(2)} \in W_i^{(2)}$ と入力ベクトル $\mathbf{p}_{j(a, \mathbf{h})}^{(2)}$ との内積を計算し、統合セルがそれらの内積の最大値を出力する。この計算は式(2.2.5)を離散化したものと見る事ができるので、結局、第2層は式(2.2.9)にほとんど等しいベクトル

$$\mathbf{p}_{j(a, \mathbf{h})}^{(3)} \simeq \mathbf{s}_{j(a, \mathbf{h})} \quad (2.3.3)$$

を出力する。

第3層では、その第 $j (\in J)$ クラスタ内の競合セルの荷重ベクトル $\mathbf{w}_{jk}^{(3)} (k \in K^{(3)})$ のひとつが、変換していない第 j テストパターン $p_j(\mathbf{x})$ に対する第3層への入力ベクトル $\mathbf{p}_j^{(3)} \equiv \mathbf{p}_{j(1, \mathbf{e})}^{(3)}$ を正規化した $\mathbf{p}_j^{(3)} / \|\mathbf{p}_j^{(3)}\|$ を記憶する。ここで $K^{(3)}$ は第3層の各クラスタ内の競合セルの番号の集合である。すると式(2.2.10)より

$$\frac{\mathbf{p}_{j(a, \mathbf{h})}^{(3)}}{\|\mathbf{p}_{j(a, \mathbf{h})}^{(3)}\|} \simeq \frac{\mathbf{s}_{j(a, \mathbf{h})}}{\|\mathbf{s}_{j(a, \mathbf{h})}\|} \simeq \frac{\mathbf{s}_j}{\|\mathbf{s}_j\|} \simeq \frac{\mathbf{p}_j^{(3)}}{\|\mathbf{p}_j^{(3)}\|} \quad (2.3.4)$$

が成り立つので、学習後のネットに第 j テストパターンの変換パターン $ap_j(\mathbf{h}(\mathbf{x}))$ が入力されると、すべてのクラスタ内の正規化された荷重ベクトルのなかで第3層への入力パターン $\mathbf{p}_{j(a, \mathbf{h})}^{(3)}$ との内積が最大になるのは $\mathbf{p}_j^{(3)} / \|\mathbf{p}_j^{(3)}\|$ を記憶している第 j クラスタ内の荷重ベクトルになる。よって内積の最大値をもつクラスタの番号をネットが認識するパターン番号であると解釈すると上の過程により訓練した座標変換に対する不変認識ができることになる。また前述したように実際の文字認識などで異なる書体の文字 $ap_k(\mathbf{h}(\mathbf{x}))$ を第 j パターンとして認識させるには、第3層の第 j クラスタ内に $\mathbf{p}_{k(a, \mathbf{h})}^{(3)} / \|\mathbf{p}_{k(a, \mathbf{h})}^{(3)}\|$ を記憶させればよい。

2.4 エネルギー関数と学習

ここでは第 L 層の第 i クラスタでの学習を考えるとし、層番号 L およびクラスタ番号 i を省略して説明する。ここで $L = 2$ または $L = 3$ であり $L = 2$ のとき $i \in I$, $L = 3$ のとき $i \in J$ である。さらに記憶すべきベクトルの集合を $P_G = \{\mathbf{p}_g \mid \|\mathbf{p}_g\| = 1, g \in G\}$ と表す。ここで G は記憶すべきベクトルの添字集合である。これは2.3の議論から座標変換以外のパターンも含

む必要があるためここでは座標変換に限らないことにする。さて各クラスタ内の荷重ベクトル \mathbf{w}_k ($k = 1, 2, \dots, |K|$) の個数 $|K|$ は記憶すべきベクトルの集合 P_G の要素数 $|P_G| = |G|$ より一般に小さく、すべての変換パターンを記憶することはできない。そこでエネルギー関数

$$E = \sum_{k \in K} \sum_{g \in G_k} \frac{1}{2} \|\mathbf{p}_g - \mathbf{w}_k\|^2 f(g) \quad (2.4.1)$$

を最小化することを考える。ここで G_k は集合 G を \mathbf{w}_k に関して次のように分割したもの、すなわち \mathbf{p}_g に最も近い荷重ベクトルが \mathbf{w}_k となる $g \in G$ の集合：

$$G_k = \left\{ g \mid \mathbf{w}_k = \underset{\mathbf{w}_m}{\operatorname{argmin}} \|\mathbf{p}_g - \mathbf{w}_m\|, g \in G, m \in K \right\} \quad (2.4.2)$$

である。さらに $f(g)$ は g が現れる確率である。ここで、もしセル数が十分大きくて $|K| \geq |G|$ であり、すべての $g \in G$ に対し $\mathbf{w}_m = \mathbf{p}_g$ となる \mathbf{w}_m ($m \in K$) が存在すればエネルギー E を零にできることを注意しておく。(2.4.1) 式より荷重変化に対するエネルギーの勾配は

$$\frac{\partial E}{\partial \mathbf{w}_k} = -(\mathbf{p}_g - \mathbf{w}_k) f(g) \quad (2.4.3)$$

となる。したがって勾配法または確率近似法（例えば文献 [13] 参照）により、

$$\frac{dE}{dt} = -\frac{\partial E}{\partial \mathbf{w}_k} \quad (2.4.4)$$

の解すなわち E の極小値に到達するための次の手順が与えられる。すなわち時刻 $t = 1, 2, \dots$ において、学習のための入力 $\mathbf{p}_g(t)$ が与えられたとき、それに最も近い荷重 $\mathbf{w}_k(t)$ を選び

$$\mathbf{w}_k(t+1) = \mathbf{w}_k(t) + \alpha(t)(\mathbf{p}_g(t) - \mathbf{w}_k(t)) \quad (2.4.5)$$

により変更すればよい。ここで $\alpha(t)$ は小さな定数、あるいは時刻 t における入力信号の確率密度の変化にともなって調整される変数とする。また多層競合ネットの学習では記憶すべき $\mathbf{p}_g(t)$ が正規化されていることから $\mathbf{w}_k(t+1)$ を正規化する必要があることに注意しなければならない。

さて、以上の勾配法に基づいて得られるアルゴリズムでは局所解に陥る恐れがあるため、本研究ではネットの学習アルゴリズムとして CRL アルゴリズム [27] を用いる。CRL は勾配法における局所解問題の解決や収束速度向上のために我々が提案するベクトル量子化の手法である。すなわち、CRL は入力信号の分布に対する荷重ベクトルの歪を逐次的に計算し、それを勾配法と非利用セルの再配置という 2 種類の操作の切り替えと学習率 $\alpha(t)$ の値の決定に用いるアルゴリズムである。これらの詳細は 3 章において述べる。

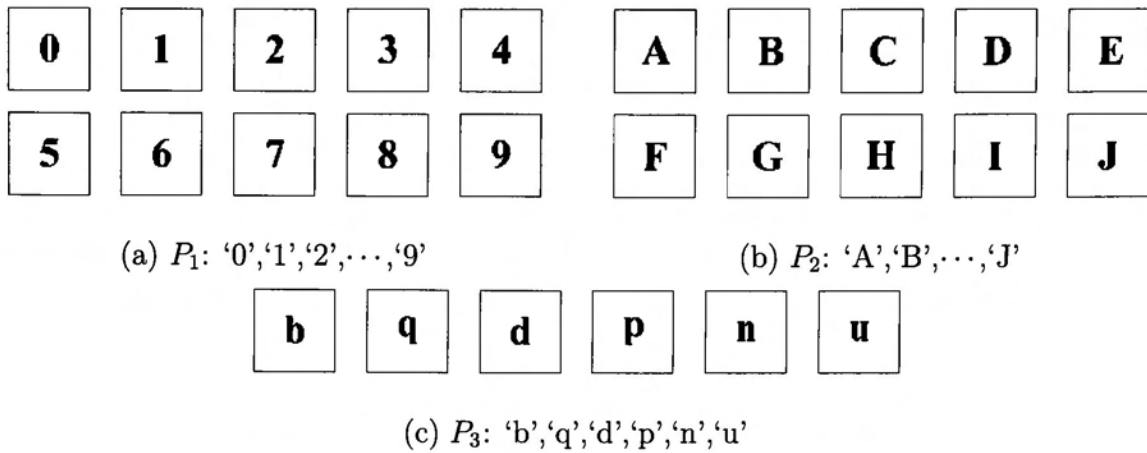


Fig. 2.3: Pattern sets.

2.5 シミュレーション

2.5.1 使用するパターンセット

Microsoft Windows 98に付属のフォント (Times New Roman) を用いて Fig.2.3 に示すパターンセットを構成した。 P_1 は '0' から '9' の数字パターンセット、 P_2 は 'A' から 'J' のパターンセットである。 P_3 は回転すると等しい 'b' と 'q'、 'd' と 'p' および 'n' と 'u' のパターンセットである。これらのパターンは、各フォントの画像を 60×60 画素の背景の中央にはめ込み 2 値画像 (0 または 1) としたものである。

2.5.2 射影変換に対する不変認識

(1) 第 2 層出力の不変性

まず射影変換パターンに対する第 2 層出力の不変性を得るため、射影変換行列

$$M = \begin{pmatrix} 1 + \Delta_a & \Delta_b & \Delta_e \\ \Delta_c & 1 + \Delta_d & \Delta_f \\ \Delta_p & \Delta_q & 1 \end{pmatrix} \quad (2.5.1)$$

とその逆変換行列を

$$\begin{aligned} |\Delta_a| \leq 0.1, \quad |\Delta_b| \leq 0.1, \quad |\Delta_e| \leq 2, \\ |\Delta_c| \leq 0.1, \quad |\Delta_d| \leq 0.1, \quad |\Delta_f| \leq 2, \\ |\Delta_p| \leq 0.03, \quad |\Delta_q| \leq 0.03, \end{aligned} \tag{2.5.2}$$

の範囲でランダムに生成して P_1 の各パターンについてそれぞれ 70000 通りの変換パターンを作成し、第 2 層の対応するクラスタにそれぞれ学習させた。第 2 層の各クラスタの競合セルの数はそれぞれ 20 とした。

この後、テストパターンとして式 (2.5.1) の変換行列 M の Δ_p と Δ_q の値をそれぞれ -0.025 から 0.025 まで 0.005 間隔で変化させ、これら以外のパラメタの変化分は 0 に固定して P_2 の各パターンの射影変換パターンを生成した。ここで、射影変換行列のパラメタについて訓練パターンをテストパターンのそれよりも少し広い範囲の値を用いて生成したのは、学習時に第 2 層において形成する訓練パターンに対する射影変換群を、テストパターンの射影変換群に対して余裕を持って形成することでより高い汎化能力を獲得するためである。 P_2 の 'A' から生成されたテストパターンの一部を Fig.2.5.2(a)–(d) に示し、これらを含む変換パターンに対する第 2 層の出力結果を (e) に示す。この結果は第 2 層で学習された P_1 の変換パターンに施された変換を未学習のテストパターン 'A' に施してネットに入力すると、第 2 層の出力はほぼ一定となるが、完全な不変性は得られていないことを示す。この不完全な不変性は、式 (2.2.8) の α_h の i に対するばらつきや、実際のネットは連続なパターンや変換群を記憶できないために生ずる離散化誤差などに起因すると考えられる。テストパターン 'A' 以外のパターンに対しても同様に不完全な不変性が得られるが、これらの不変認識への影響は次節で検討する。

(2) 不変認識

変換を施していない P_2 の各パターンを、前節 2.5.2 で訓練した第 2 層をもつネットの第 3 層の対応するクラスタにそれぞれただ一度ずつ学習させた。次に、 P_2 の各パターンに対して前節のテストパターンの生成方法と同様に Δ_p , Δ_q を変化させて生成した射影変換パターンを用いて認識テストを行った。全テストパターンに対する認識率は 92.4% となり、より詳細な認識率として Fig.5 の (a) と (b) が得られた。(a) はパターンによって認識率が異なることを示し、(b)

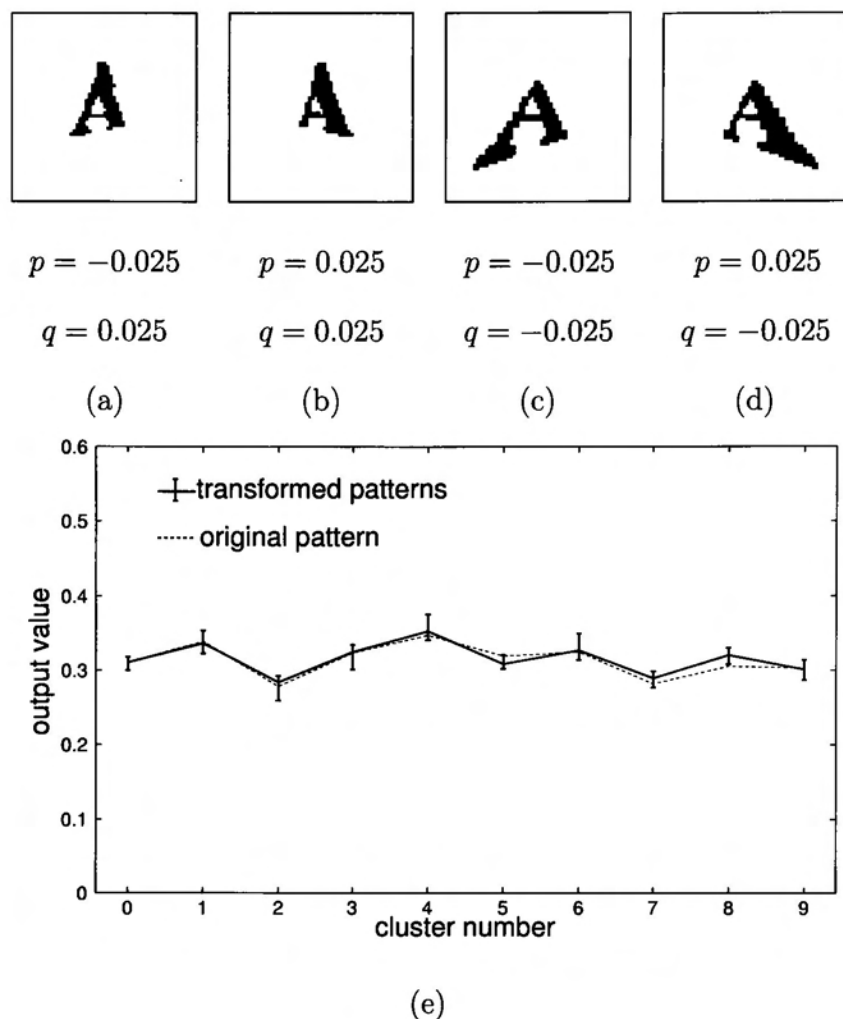


Fig. 2.4: (a)–(d) Transformed patterns and the parameters of projective transformations applied to ‘A’ in P_2 , and (e) invariance.

は Δ_p と Δ_q の絶対値が大きいとき認識率が低いことを示す。これら結果はそれぞれ次のふたつの誤認識の原因と対策を示唆する。ひとつの原因は訓練パターンとテストパターンの関係に起因している。すなわち 2.2.1 の議論によると、各テストパターンに非常に類似した訓練パターンを記憶すると、そのテストパターンに対する識別能力が得られるので、逆に任意に選んだ訓練パターンを用いると各テストパターンの認識率が Fig.5(a) のようにそれぞれ異なることが理解できる。このことを考慮し、第 2 層への訓練パターンセットとして P_1 の代わりにテストパターンセット P_2 を用いて上と同様の実験を行った結果、すべてのパターンが認識できた。この結果はテストパターンと訓練パターンを等しくすることにより得られたものであり、不変認識にお

けるネットの高い学習能力を示している。

もうひとつの原因は前節で示した第2層出力の不完全な不変性である。特に射影変換行列の Δ_p と Δ_q の絶対値が大きくなると式 (2.2.8) の近似誤差が大きくなるので Fig.5(b) の結果が理解できる。これらの誤認識は、2.2.1 の議論から、大きな Δ_p と Δ_q に対する第2層の出力も第3層で学習することにより除去できると考えられる。そこで第3層に $(\Delta_p, \Delta_q) = (\pm 0.015, \pm 0.015)$ を用いた4通りの組合せに対する P_2 の変換パターンを追加学習させ認識テストを行った結果、Fig.5(c) が得られ全体の認識率は 97.0% に増加した。これはネットの高い汎化能力を示すと言える。

2.5.3 回転変換に対する TC 汎化能力

提案するネットは TC 汎化能力、すなわち訓練パターンに内在する変換群のみに対する不変性を獲得する能力をもつ。ここでは回転を含まない変換群と回転を含む変換群を用いた P_1 の変換パターンを学習させ、回転して位置を調整すると等しくなるパターンを含む P_3 を用いてテストすることによりこの能力を検討する。

まず回転を含まない変換群として式 (2.5.1) の変換行列 M の各パラメタを

$$\begin{aligned} |\Delta_a| \leq 0.1, \quad |\Delta_b| \leq 0.1, \quad |\Delta_e| \leq 2, \\ |\Delta_c| \leq 0.1, \quad |\Delta_d| \leq 0.1, \quad |\Delta_f| \leq 2, \\ |\Delta_p| \leq 0.01, \quad |\Delta_q| \leq 0.01, \end{aligned} \tag{2.5.3}$$

の範囲でランダムに生成した。実際にはこの範囲の変換行列 M により構成される射影変換群は -5.74° から $+5.74^\circ$ までの回転変換を含むが、この範囲は離散化や入力パターン間の回転中心のずれなどの影響を除くためのものであり、この節では上式の範囲により生成される変換群を非回転変換群と呼ぶことにする。これに対し回転を含む変換群は上式の範囲の M と回転角 θ が $0^\circ \leq \theta < 360^\circ$ の範囲のランダムな値をとる回転変換 M_θ を合成した

$$MM_\theta = M \begin{pmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{2.5.4}$$

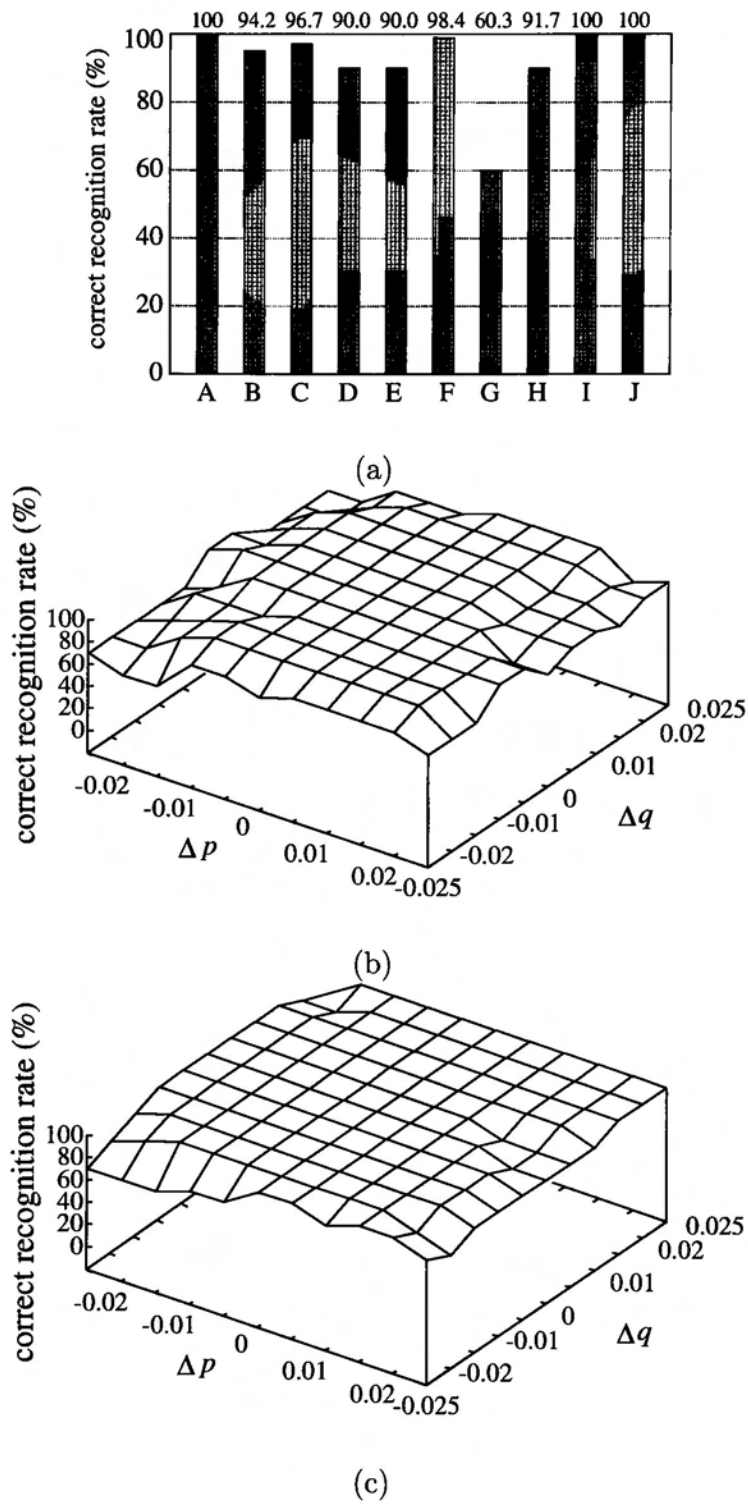


Fig. 2.5: Correct recognition rates for (a) the transformed patterns, (b) the parameters Δ_p and Δ_q , and (c) those after the training for $\Delta_p = \pm 0.015$ and $\Delta_q = \pm 0.015$.

Tab. 2.1: Results of the recognition tests for rotations.

	非回転パターン認識テスト	回転パターン認識テスト
非回転パターン学習ネット	100.0%	47.6%
回転パターン学習ネット	50.0%	100.0%

により生成した。

第2層には、上記の M と MM_θ をそれぞれ用いて 2.5.2 と同様の方法でパターンセット P_1 の各パターンを学習させた。以下では学習後のこれらのネットをそれぞれ非回転パターン学習ネットおよび回転パターン学習ネットと呼ぶ。次にこれらのネットの第3層に P_3 の各パターンをそれぞれ別のクラスタに1回ずつ学習させた場合と、 P_3 の 'b', 'd', 'n' だけをそれぞれ別のクラスタに1回ずつ学習させた場合について認識テストを行った。前者は P_3 の各パターンの変換パターンをネットに入力したとき第3層の対応するクラスタのセルが最大値を出力した場合を正しい認識とし、後者は 'b' と 'q', 'd' と 'p', 'n' と 'u' の変換パターンをネットに入力したときそれぞれ 'b', 'd', 'n' に対応する第3層のクラスタのセルが最大値を出力した場合を正しい認識とする。以下ではこれらをそれぞれ非回転パターン認識テストおよび回転パターン認識テストと呼ぶ。なお非回転パターン認識テストは離散化などに起因するわずかな回転変動に対してもうまくいくことを確かめるため、 P_3 の各パターンについて -5° から 5° までの範囲で 1° 間隔で生成した回転を含む変換パターンを用いて行った。また回転パターン認識テストは -180° から 180° までの範囲で 1° 間隔で生成した回転を含む変換パターンを用いて行った。Table 1 は以上の認識テストの結果を示す。この表により、学習した変換群のみに対して高い不変認識能力、すなわち TC 汎化能力が得られていることが分る。

2.6 まとめ

本章では、多層競合ネットの訓練パターンに不変認性を獲得しようとする座標変換の Jacobian を乗じて学習することにより、線形および非線形座標変換に対する不変認識能力を獲得することができることを示した。さらに非線形座標変換の具体例として射影変換を用い、提案手法を用いて学習を行った多層競合ネットが射影変換に対する不変認識能力および TC 汎化能力を獲得することをシミュレーションにより示した。

第3章 再初期化法を用いる適応ベクトル量子化アルゴリズム

3.1 はじめに

VQとは、連続信号もしくは大量の離散ベクトル信号である標本データを有限個の代表ベクトルを用いて量子化することで情報伝送における情報量や保存データ量の圧縮を行おうとするものであり、画像データや音声データなどの大量の信号のデジタルコーディングに有用であることが示されている [14]。しかし従来の多くのVQの手法には、局所解に陥る、収束が遅い、または入力信号の確率分布が一定であることを前提としているため、実際の多くの応用における時間変化する入力信号の確率分布に対して直接これらの手法を用いることは困難であるといった問題があった。したがって、近年、これらの問題を解決することを目的とした非定常入力信号に適応するためのVQアルゴリズム(適応VQアルゴリズム)が多く提案されている。現在までに提案されている適応VQの手法は、バッチ処理による手法とオンライン処理による手法に大きく分類される [14, 15]。前者の手法は、一定時間区間ごとに一定の入力信号をバッファに蓄え、これに対する最適なVQを順次行うものであり、入力信号の変化への適応は各一定時間ごとに行われる。代表的なものとしてLBG(Linde-Buzo-Gray)法 [16]を拡張したGershoらの手法 [17]やCSL(Competitive and Selective Learning)[18]などがある。これに対し後者の手法はGrossberg[19]やKohonen[20]による競合学習を発展させたものであり、各時刻の入力信号に対して逐次的に適応しVQを行う手法である。この手法は全ての入力データをバッファに蓄えておく必要がないため、バッチ処理手法では一度に扱うことができないような多くの入力信号のVQに用いることができ、またバッチ処理手法の代替として用いることもできる。現在までに提案されているオンライン処理型の適応VQアルゴリズムにはOPTM(Optimal Adaptive K-Means Algorithm)[21]やDOCL-II(Diversity Oriented Competitive Learning)[22]などがある。OPTM

は時間変化する歪変数を導入し、これを用いて勾配法の学習率を逐次的に計算することで適応 VQ を可能にしている [21]. しかし OPTM による VQ は入力信号の変化によっては局所解に陥ることが芳澤ら [22] によって指摘されており、この問題を解決するための手法として DOCL-II が提案された. DOCL-II は発火回数に基づくエントロピーを導入し、代表ベクトルの数が十分多い場合の最適解条件 (漸近的最適解条件)[18, 24] の近似条件である等発火回数条件を満たす解に近付ける工夫によって適応 VQ を達成しようとするものである. しかし DOCL-II は決定すべきパラメタの数が多く、入力信号の時間変化に対する適応が遅いという問題があり、さらに等発火回数条件による VQ は漸近的最適性を保証しないことが Ueda ら [18] によって示されている. そこで、本論文では漸近的最適解のための条件を与える等歪原理 [18] を理論的背景とする再初期化法と、局所最適化を行う勾配法を統合したオンライン処理型の適応 VQ の手法を提案する. 提案手法は入力信号の逐次学習が可能であるニューラルネットワークを用いて実現され、従来の手法よりもアルゴリズムが比較的簡潔であると同時に、迅速に入力信号の変化に適応することができることに特徴がある.

以下では 3.2 で時間変化する入力信号の確率分布に適応するための勾配法について述べる. 3.3 では提案手法の理論的背景となる等歪条件を説明した後、競合学習に組み込む再初期化法について述べる. 3.4 では 2次元ベクトルに対する VQ について、入力が一定時間区で切り換わる場合と連続的に変化する場合について提案手法と DOCL-II, OPTM の性能比較を行う. 3.5 では画像データのコーディングと高次元ベクトル VQ における提案手法の有効についての検証を行う.

3.2 勾配法と局所解問題

ニューラルネットワークを用いた VQ を行う場合には、標本データを入力ベクトルに、代表ベクトルを各セルの持つ荷重ベクトルにそれぞれ対応付けし、入力ベクトルに最近隣の荷重ベクトルのみを勾配法によって更新するという競合学習法を用いるのが一般的である. ここでは $\boldsymbol{x} \in \mathbb{R}^k$ を確率密度関数 $P(\boldsymbol{x})$ を持つ k 次元の入力ベクトルとし、荷重ベクトル $\boldsymbol{w}_i \in \mathbb{R}^k$ ($i \in I = \{1, 2, \dots, N\}$) を持つ競合セルからなるニューラルネットワークを考える. このニューラル

ネットワークの学習においては入力ベクトル \mathbf{x} と荷重ベクトル \mathbf{w}_i の歪

$$D = \frac{1}{k} \sum_{i=1}^N \int_{V_i} \|\mathbf{x} - \mathbf{w}_i\|^2 P(\mathbf{x}) d\mathbf{x} = \sum_{i=1}^N D_i \quad (3.2.1)$$

を最小化することを目的とする。ここで D_i は \mathbf{w}_i に対するボロノイ領域 V_i における歪 (部分歪) を表し、 V_i は以下の式で与えられる。

$$V_i = \{\mathbf{x} \mid \|\mathbf{x} - \mathbf{w}_i\| \leq \|\mathbf{x} - \mathbf{w}_j\| \text{ for each } j \neq i\} \quad (3.2.2)$$

この歪 D の最小化を行うための勾配法による荷重ベクトルの更新法は、確率近似法 [13] より

$$\mathbf{w}_c(t+1) := \mathbf{w}_c(t) + \alpha(t)(\mathbf{x}(t) - \mathbf{w}_c(t)) \quad (3.2.3)$$

で与えられる。ただし $\mathbf{w}_c(t)$ は離散時刻 t における入力 $\mathbf{x}(t)$ に最近隣の荷重ベクトルであり “:=” は代入を意味する。また $\alpha(t)$ は刻み幅であり学習率とも呼ぶ。しかし勾配法のみによる荷重ベクトルの配置は確率密度関数で重み付けされた各ボロノイ領域の重心

$$\mathbf{w}_i = \frac{\int_{V_i} \mathbf{x} P(\mathbf{x}) d\mathbf{x}}{\int_{V_i} P(\mathbf{x}) d\mathbf{x}} \quad (3.2.4)$$

に収束するため、大域的最適解ではなく局所最適解にしかない [23]。すなわち、この手法のみによる荷重ベクトルの最終配置は初期配置に大きく依存し、例えば一度も競合に勝てないセルが存在するような状況が生ずる場合があることが知られている。またこの式を荷重ベクトルの更新に用いる多くの VQ アルゴリズムでは収束のために学習率 $\alpha(t)$ を $t \rightarrow \infty$ で $\alpha(t) \rightarrow 0$ とするが、これでは本研究で考えるような時間変化する入力信号の確率分布に対しては適用できない。すなわち、長い時間が経過し $\alpha(t)$ が 0 に非常に近い値となると $\mathbf{w}_c(t)$ はほとんど更新されなくなり、新たな確率分布を持った入力には対応できなくなる。そこで本研究では勾配法の局所解問題を解決するための再初期化法を導入し、また時変入力分布に適応するため学習率を学習の収束の度合によって逐次的に決定する (学習率の計算法の詳細は 3.3.3(1) で述べる)。

3.3 再初期化法を用いた適応VQ

3.3.1 等歪条件

最適な荷重ベクトル配置の条件は式(3.2.1)で定義された歪を用いて以下のように与えられる[18, 24]; 入力ベクトル \mathbf{x} の確率密度関数 $P(\mathbf{x})$ は連続であり, 荷重ベクトル \mathbf{w}_i ($i \in I$) の数 N は十分大きく, 全ての $\mathbf{x} \in V_i$ と全ての $i \in I$ について近似 $P(\mathbf{x}) \simeq P(\mathbf{w}_i)$ が成り立つとする. このとき D の下限もしくは最小値は $i \in I$ において全ての D_i が互いに等しくなった時に得られる. 以上の最適VQに関する条件を等歪条件という.

3.3.2 再初期化法

まず \mathbf{w}_i を持つセルは各離散時刻 t における入力ベクトルに対する歪を加算した歪変数 $d_i(t)$ を

$$d_i(t+1) := \begin{cases} \eta d_i(t) + \|\mathbf{x}(t) - \mathbf{w}_i(t)\|^2 & \text{if } i = c \\ \eta d_i(t) & \text{otherwise} \end{cases} \quad (3.3.1)$$

として蓄えることとする. ここで η ($0 < \eta < 1$) は忘却率であり, $\mathbf{w}_c(t)$ は $\mathbf{x}(t)$ の最近隣荷重ベクトルである. また $d_i(t)$ ($i \in I$) の初期値は非常に小さな値とする. 式(3.3.1)より $\mathbf{w}_i(t)$ ($i \in I$) の平衡状態における $d_i(t)$ は,

$$\begin{aligned} d_i(t) &= \frac{1}{1-\eta} \int_{V_i} \|\mathbf{x}(t) - \mathbf{w}_i(t)\|^2 P(\mathbf{x}) d\mathbf{x} \\ &= \frac{kD_i}{1-\eta} \end{aligned} \quad (3.3.2)$$

に収束するため, 等歪は歪変数 $d_i(t)$ が全ての $i \in I$ で等しい時に達成されるといえる. そこで, 等歪に対する全体的な歪変数の状態を評価する指標として歪変数 $d_i(t)$ のエントロピー $H(t)$ を導入する. $H(t)$ は以下のように計算される.

$$H(t) = - \sum_{i=1}^N \frac{d_i(t)}{\sum_{j=1}^N d_j(t)} \ln \left(\frac{d_i(t)}{\sum_{j=1}^N d_j(t)} \right) \quad (3.3.3)$$

これを正規化して

$$I(t) = \frac{H(t)}{\ln(N)} \quad (3.3.4)$$

とし $I(t)$ を学習収束度と呼ぶことにする。この値は $0 \leq I(t) \leq 1$ であり、全ての歪変数 $d_i(t)$ が等しい時のみ 1 となり、歪変数 $d_i(t)$ の値のばらつきが大きいほどこの値は小さくなる。そこで学習収束度の閾値 I_{th} を 1 よりも小さなある定数で定め、時刻 t において

$$I(t) < I_{th} \quad (3.3.5)$$

を満たす場合は、全体的な荷重ベクトルの配置は等歪の状態から遠いと判断することにする。

さて式 (3.3.5) とは対照的に、以下のような条件によっても各セルの歪変数の等歪からの偏りを逐次的かつ容易に判断することができる。すなわち、競合学習では各時刻で入力ベクトルに最近隣である荷重ベクトルを持つセルを探索するので、そのセルの歪変数が

$$d_c(t) > d_{th} d_{mean}(t) \quad (3.3.6)$$

を満たすとすると、この歪変数が等歪を満たさない要因となっていると判断することにする。ここで $d_{mean}(t) = \sum_{i=0}^N d_i(t)/N$ であり、 d_{th} は 1 より大きな定数である。以下では d_{th} を歪変数の閾値と呼ぶ。

最後に、これら 2 種類の等歪に対する歪変数の判断基準を以下のように統合する。すなわち式 (3.3.5) と式 (3.3.6) を同時に満たす場合には全てのセルの中で最小の歪変数 $d_l(t)$ を持つ第 l 番目のセルの持つデータと競合に勝つセルの歪変数 $d_c(t)$ を以下のように更新する。

$$\begin{aligned} \mathbf{w}_l(t+1) &:= \mathbf{x}(t) \\ d_i(t+1) &:= \begin{cases} \eta d_{mean}(t) & \text{if } i = c \text{ or } i = l \\ \eta d_i(t) & \text{otherwise} \end{cases} \end{aligned} \quad (3.3.7)$$

これを第 l 番目のセルを再初期化するという。すなわちこの処理は、歪変数が小さいセルのデータを初期化し、大きな歪変数を持つセルの近傍に再配置することを意味する。また、この処理によって歪変数が平均化されることで歪変数のエントロピー $H(t)$ が上昇し、 $I(t)$ も上昇するこ

となる。なお、この処理を行った直後の歪変数は正しい値を持たないが、時間が十分経過すると式 (3.3.1) より式 (3.3.2) が成立するようになり、正しい値を持つようになると考えられる。

以上をまとめると以下のアルゴリズムとなる; 各時刻 t における入力 $\mathbf{x}(t)$ に対して、式 (3.3.5) と式 (3.3.6) を満たす場合は式 (3.3.7) により第 l 番目のセルを再初期化する。その他の場合は式 (3.2.3) により $\mathbf{w}_c(t)$ を修正し、式 (3.3.1) により $d_i(t)$ を更新する。以下、このアルゴリズムを CRL (Competitive and Re-initialization Learning) と呼ぶことにする。

3.3.3 各パラメタの決定法

(1) 学習率

前述したように、式 (3.2.3) の学習率 $\alpha(t)$ を単に学習回数や学習時刻に対して減少させるようなアルゴリズムでは入力分布の変化に対応できない。理想的には、学習率はベクトル配置が収束するにしたがって減少し、もし入力分布に変化が生じ、入力信号に対する荷重ベクトルの歪のばらつきが大きくなった場合には増加するべきである。したがって、学習収束度を用いて学習率を以下のように決定することにする。

$$\alpha(t) = \frac{\ln(N) - H(t)}{\ln(N)} = 1 - I(t) \quad (3.3.8)$$

これは文献 [21] において提案された手法と同様のものであるが $H(t)$ の導出法は異なる。この式によって逐次的に求められる学習率は学習の進行状態によって自動的に調節される。

(2) 学習収束度の閾値と歪変数の閾値

学習収束度の閾値 I_{th} と歪変数の閾値 d_{th} の設定のために、まず各時刻における歪変数 $d_c(t)$ と学習収束度 $I(t)$ の値の変動の性質について述べる。

まず歪変数 $d_i(t)$ ($i \in I$) の値は、初期時刻もしくは入力分布が変化してしばらくは荷重ベクトル $\mathbf{w}_i(t)$ の更新に応じて変動し、時間経過とともに式 (3.2.4) を満たす解に収束していく。ここで各 $d_i(t)$ の収束値は入力分布と荷重ベクトルの数に依存して決まり、一般にはそれぞれ異なる値に収束する。例えば、入力分布に対して等歪を満たす荷重配置が存在せず、 $d_i(t)$ を等しい

値に収束させることが原理的にできない場合があり (3.4.1 参照), このような場合には最適な荷重配置でも $d_i(t)$ の収束値はばらつく. したがって, 各時刻において競合に勝つ荷重ベクトルの歪変数 $d_c(t)$ の値は確率的に入力されるベクトルに対応して $d_i(t)$ の中から選択されるため, 各 $d_i(t)$ の収束後も変動する. これに対し $I(t)$ はすべての $d_i(t)$ のばらつきを計る尺度であるので, $d_i(t)$ が収束すると $I(t)$ も収束する. より詳しく言うと, 初期時刻や入力分布が変化してしばらくは $d_i(t)$ がかなりばらついているのでかなり低い値になるが, $d_i(t)$ の収束に対応して時間とともにあまり変動しなくなり, ある程度高い値になっていく.

さて以上の $I(t)$ と $d_i(t)$ の変動の性質を考慮して, 入力信号が一定時間区間ごとに変化する場合と連続的に変化する場合の I_{th} と d_{th} の値の設定について考える. まず前者の場合には, 同一の入力信号の学習を一定時間行うことができるため, より高い最適化が可能であり, 各処理の後半には再初期化はなるべく起こらない方がよい. そこで, $d_c(t)$ は学習収束後もかなり変動するが $I(t)$ はあまり変動しなくなることに注意して, 式 (3.3.5) と式 (3.3.6) の閾値の値を具体的に 3.4.1 で用いる値 $d_{th} = 1.4$, $I_{th} = 0.985$ とすることを考える. すると式 (3.3.5) の $I(t) < I_{th} = 0.985$ という条件は, $d_i(t)$ が一様にばらついているとすると $d_i(t)/d_{mean}$ の値のばらつく範囲がおよそ 1 ± 0.53 以上であることを示しており, この条件が満たされる時には式 (3.3.6) の $d_c(t)/d_{mean} \geq d_{th} = 1.4$ という条件も高い頻度で満たされることになる. したがって, これらの閾値を用いて式 (3.3.5) と式 (3.3.6) を同時に満たすときにのみ再初期化を行うことにすると, 初期時刻付近あるいは入力分布が変化してしばらくは式 (3.3.5) と式 (3.3.6) が同時に満たされて高い最適性を達成するように再初期化が起こる. しかし時間が十分経過すると $I(t)$ が $I_{th} = 0.985$ よりも高い値に収束していくことにより再初期化が起こらなくなり, 勾配法だけで最適化が行われる. なお d_{th} を小さくしすぎると再初期化すべきでない荷重も再初期化し, I_{th} を小さくしすぎると入力分布の変化が検知できなくなることに注意しなければならない [25]. 次に入力信号が連続的に変化する場合には, 高い最適性や少ない再初期化よりも適応速度の方が重視される. したがって収束の遅い $I(t)$ を用いる式 (3.3.5) の条件は用いず (あるいは $I_{th} = 1.000$ とし), 式 (3.3.6) の d_{th} による条件だけを用いると勾配法と再初期化により適応の速さが保たれる.

(3) 忘却率

忘却率 η は過去の入力ベクトルを徐々に忘却し、入力分布の変化に適応するために 1 に近い値で与えられるパラメタである。各荷重ベクトルが更新されるのに要する平均時間は荷重ベクトルの数 N によって異なるので、忘却率を

$$\eta = \exp\left(\frac{-1}{N\tau}\right) \quad (3.3.9)$$

で与えることにすると、 η は $N\tau$ 時刻後に元の値を e^{-1} 倍に減衰させる効果を持ち、 τ でこの忘却率を決定できるので直観的に理解しやすい。この τ は、入力分布の変化する場合などに過去の入力分布の忘却を行う必要があるならば小さくし、逆に入力分布があまり変化しないような場合には大きくするといった調節が必要である。

3.4 2次元ベクトルへの適用

3.4.1 一定時間区間ごとに変化する入力分布

まず一定時間区間ごとに変化する入力分布に対する CRL の性能の検討を行う。ここでは比較のために文献 [22] と同様のデータセットを用い (Fig. 3.1(a) 参照), これらは以下のようにして生成した。まずガウス分布 $N(0, 0.0025)$ に従う 2000 のベクトル $\mathbf{x}_{0j} = (x_{0j}, y_{0j})^T$ ($j = 1, 2, \dots, 2000$) を生成し、その集合を X_0 とする。次にこれらを $\mathbf{x}_{ij} = \mathbf{m}_i + \mathbf{x}_{0j}$, $\mathbf{m}_i = (a_i, b_i)$ として平行移動し生成される入力ベクトルの集合をそれぞれ X_1, \dots, X_4 とした。ただし $(a_i, b_i) = (0, \pm 0.5), (\pm 0.5, 0)$ であり、以下ではこれらのベクトル集合をクラスタと呼ぶ。これらのクラスタの組合せを 300000 回の学習回数ごとに Fig.3.1(a) のように切り換えて入力し、CRL, DOCL-II, OPTM の 3 種類のアルゴリズムを用いて VQ を行った。荷重ベクトル数は $N = 24$ とし、学習に必要なパラメタの値は OPTM, DOCL-II においてはそれぞれの文献に挙げてあるものを用い、CRL においては $I_{th} = 0.985, d_{th} = 1.4, \tau = 300$ とした。各荷重ベクトルの各要素の初期値は -1 から 1 の間のランダムな値とした。Fig.3.1(b)~(d) にそれぞれ 300000 回, 600000 回, 900000 回の学習が終了後の各アルゴリズムによる荷重ベクトルの配置を示し、Fig.3.2 に入力データセットと荷重ベクトルの自乗誤差の平均 (MSE: Mean Squared Error) の推移を示す。

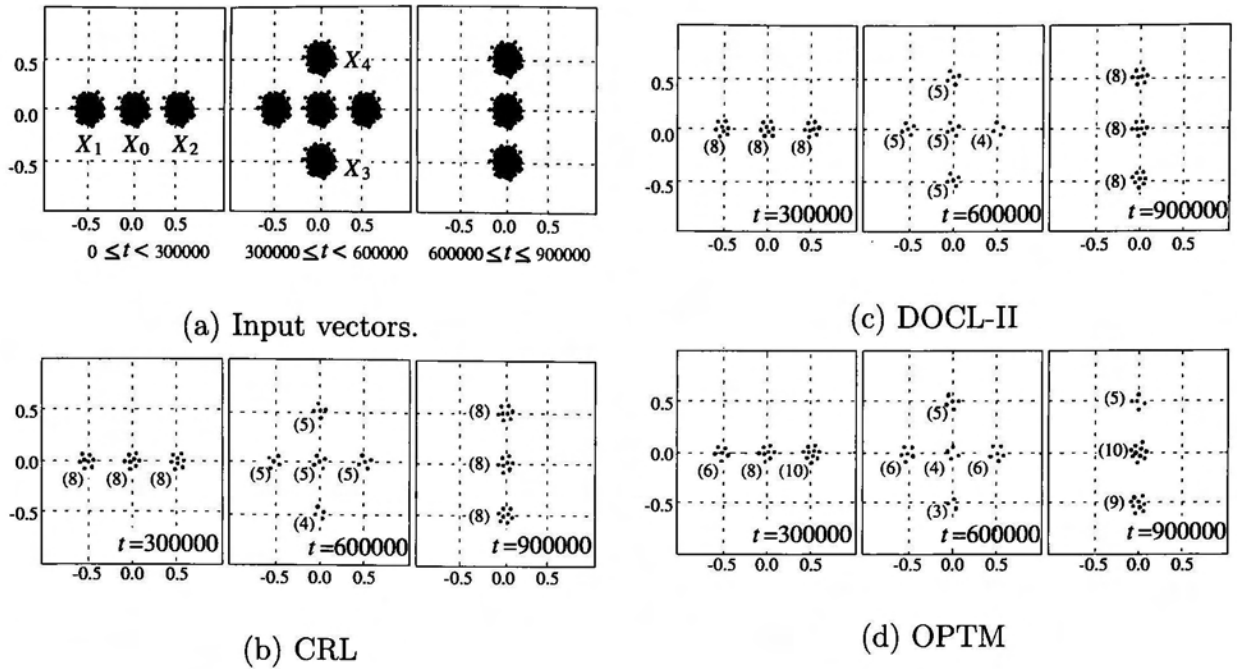


Fig. 3.1: Result of VQ for abruptly changing input distributions: (a) input vectors, the final weight vectors obtained by (b)CRL, (c)DOCL-II, (d)OPTM.

ただし $MSE = kD/N$ であり, Fig.3.1(b)~(d)における各クラスタ付近の数字は配置されたベクトルの数を表している. まず Fig.3.1(d)より, OPTMでは同図(b)のCRLや(c)のDOCL-IIとは異なり, 各クラスタに対する荷重ベクトル配置に偏りが生じ局所解に陥っていることがわかる. また, 各クラスタに対する荷重ベクトル配置がうまくいっているDOCL-IIにおいても, Fig.3.2では最も低いMSEに収束するまでに多くの学習回数を必要としていることがわかる. これらの結果より, CRLは局所解に陥ることなく, 変化する入力分布に最も速く適応していることがわかる.

3.4.2 連続的に変化する入力分布

次に連続的に変化する入力分布に対するCRLの性能について検討する. 連続的に変化する入力分布としてFig.3.3(a)に示すようなデータセットを用いた. すなわち, 入力信号は初期時刻に中心がそれぞれ $(-0.5, 0.5)$, $(-0.5, 0.0)$, $(-0.5, -0.5)$ である3つのクラスタからなり, これらは水平方向の移動によって50000回の学習の後にそれぞれのクラスタ中心が $(0.5, 0.5)$, $(0.5, 0.0)$, $(0.5, -0.5)$

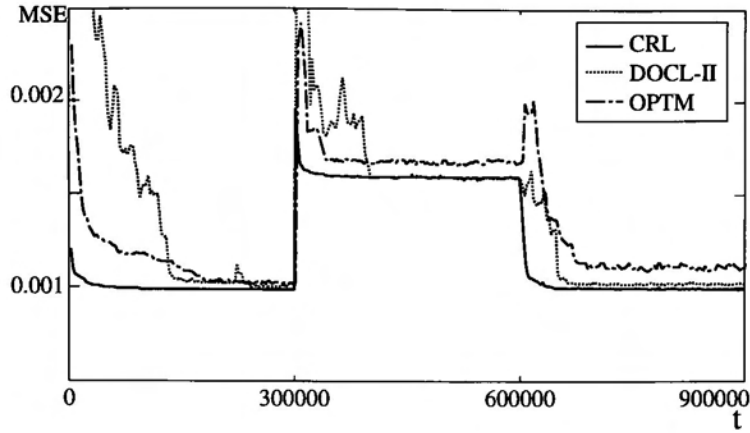


Fig. 3.2: The mean squared error (MSE) of the three VQ algorithms.

に移り、さらに次の50000回の学習の後に水平方向の移動によって再び元に戻る。この入力信号に対してCRL, DOCL-II, OPTMを用いてVQを行った。このとき3.3.3(2),(3)の議論より、連続的に変化する入力分布に対する適応を考慮してCRLのパラメタを $I_{th} = 1.000, d_{th} = 1.4, \tau = 50$ とし、 I_{th} による再初期化の発動条件を用いず、また τ を小さめに定めた。他のアルゴリズムのパラメタは3.4.1と同様とした。100000回の学習後の荷重ベクトルの配置をFig.3.3(b)~(d)に示し、MSEの推移を同図(e)に示す。まずFig.3.3(b)~(d)よりDOCL-IIやOPTMでは荷重ベクトルが最終時刻には全く入力の無い領域に留まっているが、CRLの荷重ベクトルの配置は入力分布の変化に適応できていることがわかる。次に同図(e)の結果よりCRLによるMSEの推移が3つのアルゴリズムの中で最良のものとなっていることがわかる。それに対しDOCL-IIでは入力信号の変化に適応が追い付かず大きなMSEの値で推移しており、またOPTMにおいてもMSEが低い値に推移するまでには多くの学習回数が必要であることがわかる。

3.4.3 異なるパラメタ値に対する性能

ここではCRLのパラメタ I_{th}, d_{th}, τ によって、その性能がどのように変化するかを検討する。まず、3.4.1の一定時間区間で入力信号が切り換わる場合についてそれぞれ I_{th} と d_{th} について値を変化させシミュレーションを行った時のMSEの推移をFig.3.4に示す。ただし同図(a)のシミュレーションでは $d_{th} = 1.4$, (b)では $I_{th} = 0.985$ とし、他の条件は3.4.1と同様と

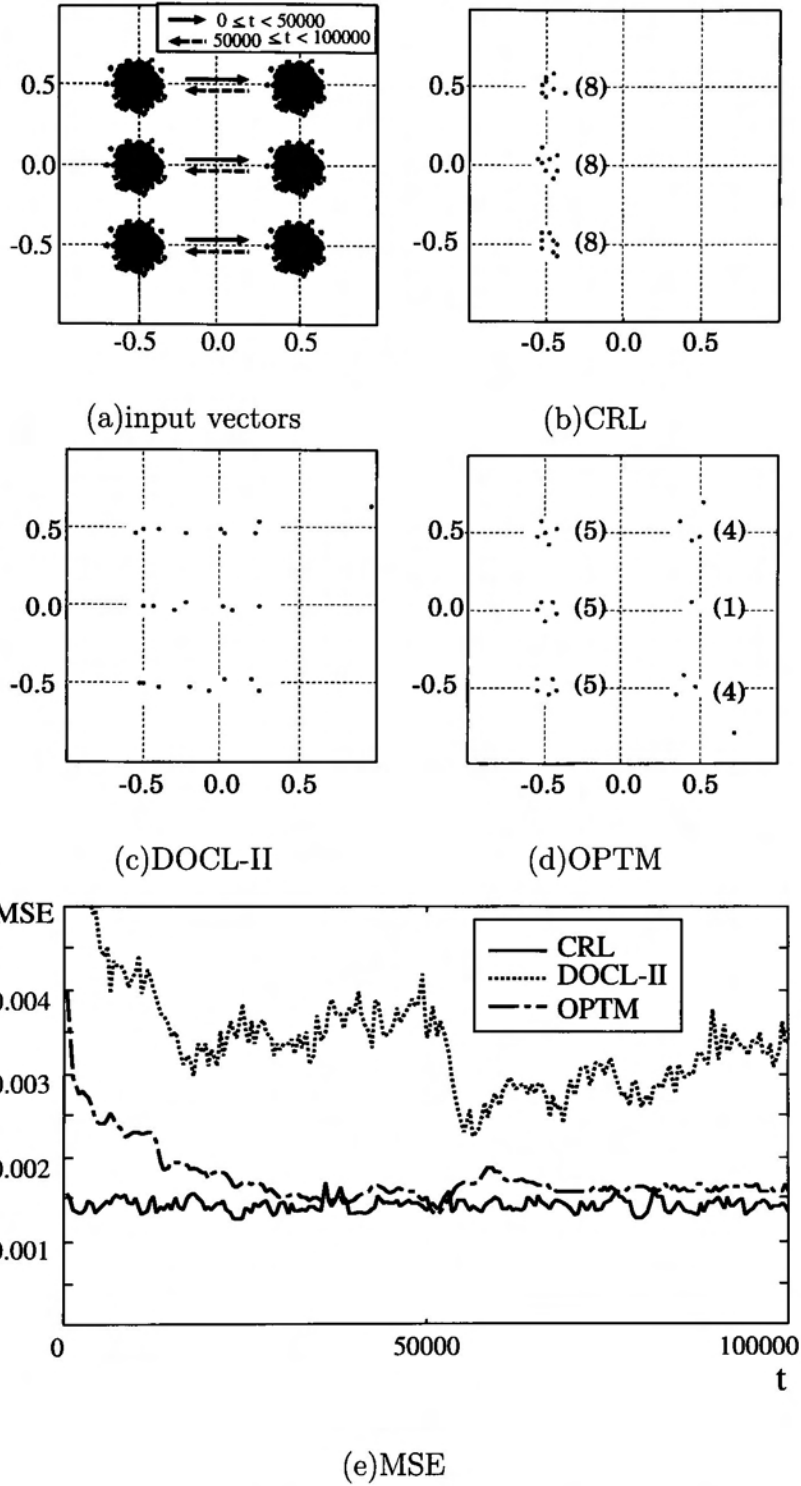


Fig. 3.3: Result of VQ for constantly changing input distributions: (a)input vectors, the final weight vectors obtained by (b)CRL, (c)DOCL-II, (d)OPTM, and (e)MSE of the three VQ algorithms.

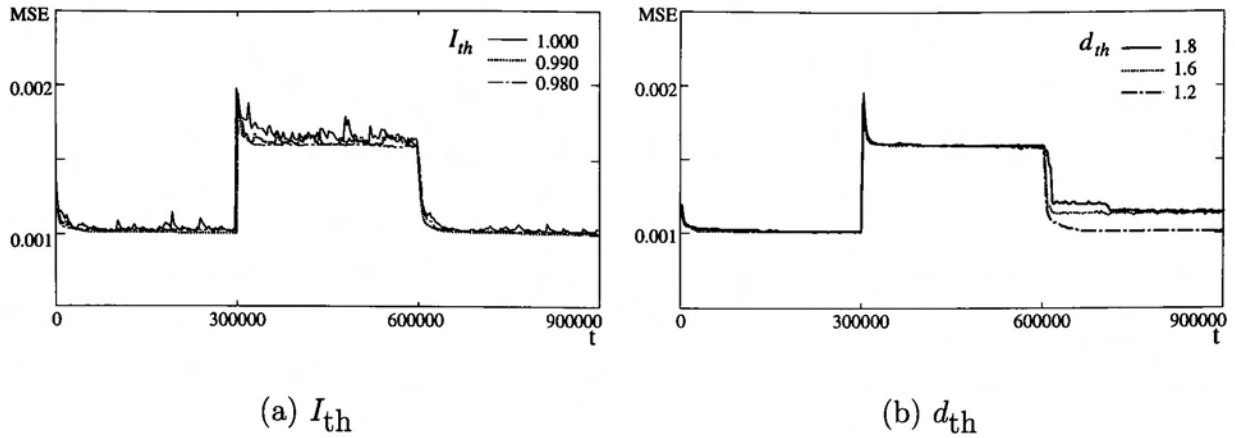


Fig. 3.4: Performances for different parameter values for abruptly changing input distributions: MSE for the several parameters of (a) I_{th} and (b) d_{th} .

した。Fig.3.4(a)より、 $I_{th} = 0.980$ とした場合には Fig.3.2 の $I_{th} = 0.985$ の場合とほぼ同様に MSE が推移したが、 $I_{th} = 0.990, 1.000$ とした場合には MSE が振動的に推移していることがわかる。この MSE の振動は過度の再初期化が生じたことを示している。しかし Fig.3.2 の結果と比較して、最も大きく振動した $I_{th} = 1.000$ においても OPTM と同等以下の MSE を維持していることがわかる。また Fig.3.4(b) より、 $d_{th} = 1.2$ の場合は Fig.3.2 の $d_{th} = 1.4$ とした場合とほぼ同様に MSE が推移したが、 $d_{th} = 1.6, 1.8$ の場合には学習回数 600000 回以後に MSE が高い値に留まり局所解に陥っていることがわかる。これは d_{th} の値が大きすぎたため、各クラスに対する荷重ベクトルの配置の偏りによって生ずる各セルの歪変数のばらつきが許容され、局所解に陥ったのだと考えられる。このような場合には d_{th} をさらに低く設定する必要があるといえる。

次に 3.4.2 の連続して変化する入力分布に対して、それぞれ d_{th} と τ を変化させてシミュレーションを行った時の MSE の推移を Fig. 3.5 に示す。ただし同図 (a) のシミュレーションでは $\tau = 50$ 、(b) では $d_{th} = 1.4$ とし、また I_{th} の値はいずれの場合も 1.000 とした。この結果より、いずれの MSE の推移も振動的ではあるが CRL による MSE は Fig. 3.3(e) の DOCL-II よりも小さく、また OPTM と同等程度の値で推移していることがわかる。以上より、CRL のパラメタ値の設定にはある程度の許容範囲があり、望ましい値と多少異なっても DOCL-II や OPTM と同等以上の性能が望めることがわかる。

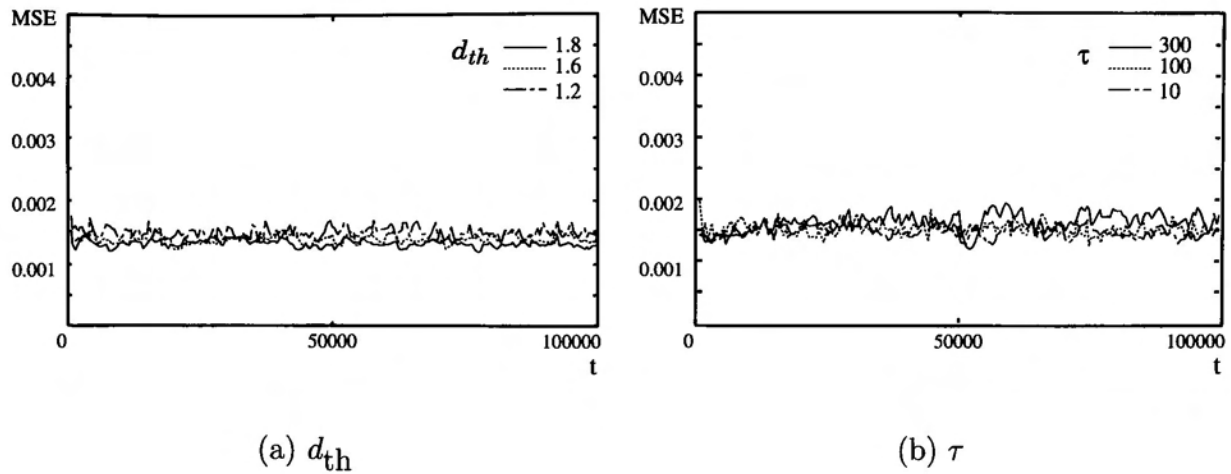


Fig. 3.5: Performances for different parameter values constantly changing input distributions: MSE for the several parameters of (a) d_{th} and (b) τ .

3.5 高次元ベクトルへの適用

3.5.1 画像データのコーディング

ここでは画像データのコーディングにおける CRL の性能の検討を行った。比較のために、文献 [21] と同様の “Lena” 画像 (Fig.3.6(a) 参照) を用いた。これは 512×512 画素からなるグレースケールの画像であり、1 画素は 8bit (256 階調) で表現される。この画像に対する VQ は文献 [21] と同様の手順により、まず画像を 4×4 画素の方形ブロックに分割し、それぞれを 16 次元の一つの入力ベクトルとした。さらに図形の中心を原点として画像を 256×256 画素の 4 象限に分割し、それぞれ左下、右上、左上、右下の画像の順にバッチ的な VQ を行っていくものとした。それぞれの象限の VQ においては、入力ベクトルをランダムに選択し 80000 回の学習を行った。この回数は各象限に含まれる 4096 のベクトルを約 20 回学習することに相当する。32 個の荷重ベクトルを用意し、初期の荷重ベクトルはランダムに生成した。またここでも CRL, DOCL-II, OPTM についての性能比較を行った。CRL のパラメタは $I_{th} = 0.985$, $d_{th} = 1.4$, $\tau = 300$ とし、他はそれぞれの文献の高次元 VQ を行うためのパラメタの値を用いた。これらの結果を Fig.3.6 に示す。また同図 (b)~(d) には、静止画像の品質測度として用いられるピーク信号対雑音比

(PSN 比) を示した。ただし PSN 比は以下のように計算される。

$$\text{PSN} = 10 \log_{10} \left(\frac{256^2}{\text{MSE}} \right) \quad (3.5.1)$$

これらの結果より CRL は他のアルゴリズムよりも高い品質で画像のコーディングを達成しており、同図 (e) に示した MSE の遷移より、各象限の画像の VQ においても最も速く低い MSE への推移が達成されていることがわかる。さらに他の画像に対する VQ においても、CRL のパラメタを調節することで高い VQ が達成されることも示されている [28]。



(a)Original.

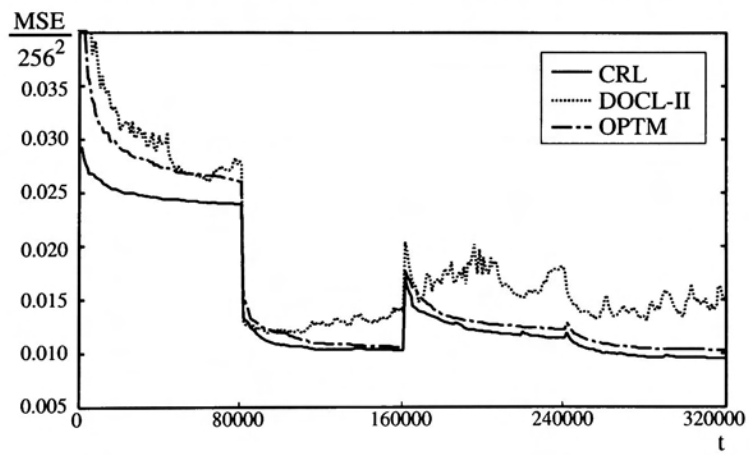
(b)CRL PSN=18.60



(c)DOCL-II PSN=17.29



(d)OPTM PSN=18.30



(e)MSE

Fig. 3.6: Result of VQ for the “Lena” image: (a)original image, coded images by (b)CRL, (c)DOCL-II, (d)OPTM, and (e)MSE of the three VQ algorithms.

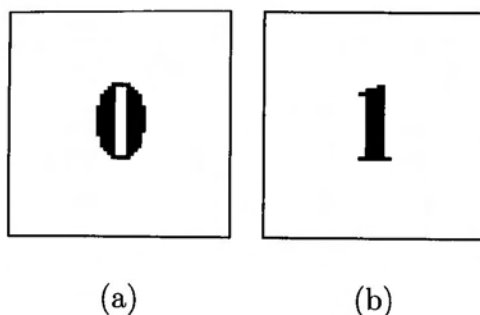


Fig. 3.7: (a)pattern '0' and (b)pattern '1'.

3.5.2 回転変換パターンの学習

ここでは多層競合ニューラルネット [26] を用いて回転変換されたパターン集合の学習を行い、高次元 VQ における CRL の性能の検討を行った。このネットへの入力ベクトルは Fig.3.7 に示す 2 値 (0,1) パターンを回転したものを用いた。これらのパターンは 60×60 画素の背景の中央に Times New Roman Bold フォントの数字 '0', '1' をはめ込み回転したものであり、3.5.1 の入力ベクトルよりもさらに高次元である 3600 次元のベクトルとして 1 パターンが扱われる。以下のシミュレーションでは 36 個の荷重ベクトルを用いた。このネットの訓練は、Fig.3.7 のパターンに 0° から 360° までの間でランダムな回転変換を施すと同時に正規化し、各時刻に 1 パターンずつ入力することで以下のように行った。まず訓練パターンを '0' として訓練を 10000 回繰り返した後、訓練パターンを '1' へと変化させ、再び 10000 回の訓練を行った。ここでも比較のために OPTM, DOCL-II について同様のテストを行った。各アルゴリズムのパラメタは 3.5.1 と同様の値を用いた。

まず MSE の推移を Fig.3.8 に示す。これはテストパターンとして 3° ごとに 120 通りの回転変換パターンを生成し、各時刻における荷重ベクトルとテストパターンとの MSE を求めたものである。この結果より CRL による VQ が最も迅速に変化した入力分布に適応しており、DOCL-II や OPTM による VQ には収束のために、より多くの学習回数が必要であることがわかる。次に 10000 回、20000 回の学習が終わった時点での荷重ベクトルをそれぞれ Fig.3.9(a)~(f) に示す。

まず Fig.3.9(a),(b) より CRL による VQ では全ての荷重ベクトルにおいて入力信号の変化に適応できていることがわかる。それに対し DOCL-II は同図 (c) においては学習が収束しているが、入力信号が切り換わった後の図 (d) では学習が収束しておらず、10000 回の学習回数では

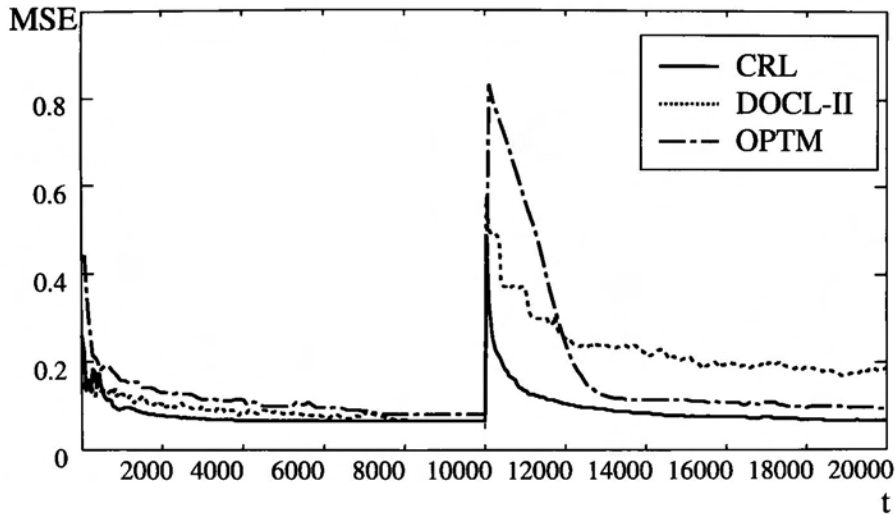
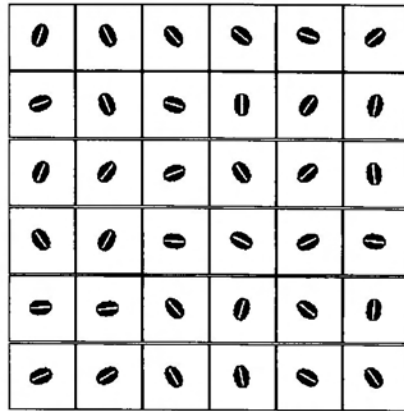


Fig. 3.8: The normalized MSE of the three VQ algorithms.

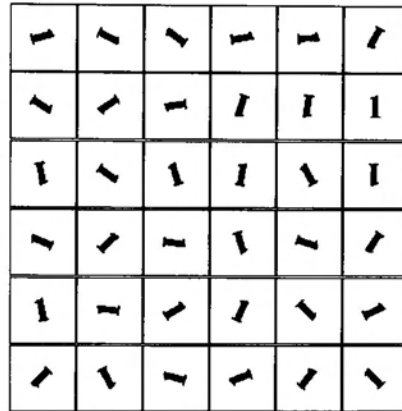
入力信号の変化に適応しきれていない。また OPTM は学習にうまく利用されていない荷重ベクトルが存在し、入力信号が変化した後も適応できない荷重ベクトルが多く存在することが同図 (e),(f) よりわかる。以上より、非常に高い次元における VQ に対しても CRL は有効であることがわかる。

3.6 まとめ

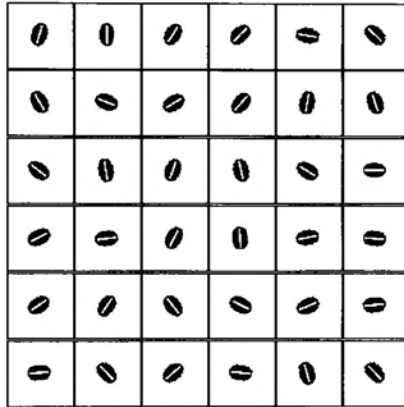
本章では入力信号の確率分布の変化に適応可能な VQ アルゴリズム、CRL を提案し、従来の適応 VQ アルゴリズムよりも高い性能を持つことを示した。CRL による適応 VQ は、適応が迅速である、設定するパラメタが少ないなどの特徴がある。また CRL による VQ は局所解に陥らず、高次元ベクトルに対する応用が十分に行えることをシミュレーションにより確認した。ただし、3つのパラメタ（忘却率、学習収束度の閾値、歪変数の閾値）の最適な値を決定する方法の検討は今後の課題として残されている。



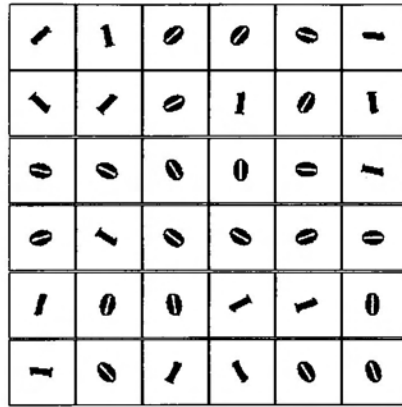
(a)CRL, $t=10000$



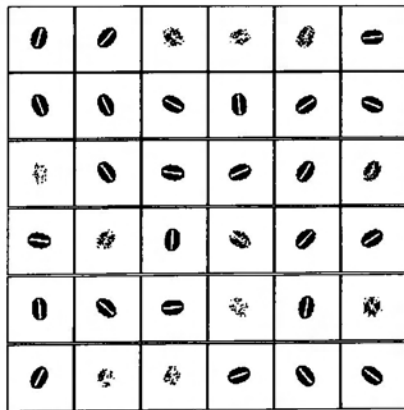
(b)CRL, $t=20000$



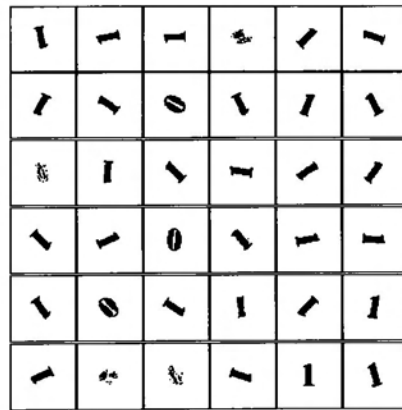
(c)DOCL-II, $t=10000$



(d)DOCL-II, $t=20000$



(e)OPTM, $t=10000$



(f)OPTM, $t=20000$

Fig. 3.9: Weight vectors after learning.

第4章 結論と今後の課題

4.1 結論

本論文では、非線形変換に対する不変パターン認識を行う多層競合ネットを提案し、また提案ネットの学習アルゴリズムとして用いる適応ベクトル量子化アルゴリズム、CRLを新たに提案した。

第2章では、訓練パターンに不変認性を獲得しようとする座標変換のJacobianを乗ずることにより、線形および非線形座標変換に対する不変性が獲得できることを示した。また、ここで示した原理を実現するための多層競合ネットの構成とその学習法を示した。その後、非線形座標変換の具体例として射影変換を用い、提案手法を用いて学習を行った多層競合ネットが射影変換に対する不変認識能力を獲得することをシミュレーションを行い検証した。その結果、ネットは射影変換に対する高い学習能力および不変認識能力を獲得することが示された。

第3章では、提案ネットの学習アルゴリズムとして用いる入力信号の確率分布の変化に適応可能なVQアルゴリズム、CRLを提案した。CRLは漸近的最適解条件を与える等歪原理を理論的背景とする再初期化法と、局所最適化を行う勾配法を統合したオンライン処理型の適応VQアルゴリズムである。CRLによる適応VQは他のアルゴリズムのそれと比較して、適応が迅速である、設定するパラメタが少ないなどの特徴がある。またCRLは従来の適応VQアルゴリズムよりも高いVQ性能を持ち、さらに局所解に陥らず、高次元ベクトルに対する応用が十分に行えることをシミュレーションにより確認した。

4.2 今後の課題

第2章で構成したネットは異なる書体の文字が混合される場合の文字認識や手書き文字認識 [11] などへも応用できることが2.2.1の議論から理解できる。このとき、より高い認識能力や汎化能力を得るには使用する訓練パターンの数と種類、訓練する変換の種類などをうまく選ぶ必要があるが、これらについての考察や実験は今後の課題である。さらに、射影変換以外の非線形座標変換に対する不変認識、例えば、ピンのラベル、ボールの表面に描かれた文字パターン、衣服などの柔らかく変形しやすい物体の表面の文字パターンなどの不変認識への応用も考えられる。このとき記憶すべき訓練パターンを生成するための Jacobian $|g_{\mathbf{x}}|$ は考慮する座標領域 D 内の主な点における変換前後の微小面積の比を式(2.2.4)により計算すればよいと考えられるが、この手法の検討と実際の場面への応用実験も今後の課題である。

また、第3章で提案した CRL アルゴリズムには3つのパラメタ（忘却率、学習収束度の閾値、歪変数の閾値）があることを述べた。しかしこれらの最適な値を決定する方法の検討は今後の課題として残されている。

謝辞

本論文の作成にあたり多くの御教授を頂いた九州工業大学工学部機械知能工学科松岡清利教授，石川聖二教授，同学部電気工学科 永松正博教授に深く感謝致します。また，本論文作成にあたり御指導下さった九州工業大学工学部機械知能工学科黒木秀一教授に深く感謝致します。さらに，海外での学術講演会への参加のための奨学金を提供して下さった社団法人明専会，日頃より御協力頂いた機械知能工学科制御工学教室の教職員の皆様ならびに，同教室黒木研究室の皆様感謝致します。

参考文献

- [1] K.Fukushima: Neocognitron: a new algorithm for pattern recognition tolerant of deformations and shifts in position, *Pattern Recognition*, Vol.15, No.6, pp.455-469, 1982.
- [2] B.Widrow, R.G.Winter and R.A.Baxter: Layered neural nets for pattern recognition, *IEEE Trans. Acoust., Speech and Signal Process.*, Vol.36, No.7, pp.1109-1118, 1988.
- [3] C.L.Giles and T.Maxwell: Learning, invariance, and generalization in high-order neural networks, *Applied Optics*, Vol.26, No.23, pp.4792-4978, 1987.
- [4] S.J.Perantonis and P.J.G.Lisboa: Translation, Rotation, and Scale Invariant Pattern Recognition by High-Order Neural Networks and Moment Classifiers, *IEEE Trans. Neural Networks*, Vol.3, No.2, pp.241-251, 1992.
- [5] W.K.Konen, T.Maurer and C.Malsburg: A fast dynamic link matching algorithm for invariant pattern recognition, *Neural Networks*, Vol.7, No.6-7, pp.1019-1030, 1994.
- [6] D.Marr: *Vision — A Computational Investigation into the Human Representation and Processing of Visual Information*, W.H.Freeman and Company, 1982.
- [7] K.Hotta, A.Iwata, H.Matsuo and N.Suzumura: A large scale neural network CombNET-II, *IEICE Trans.*, Vol.J75-D-II, No.3, pp.545-553, 1992.
- [8] S.Kurogi: Competitive neural network for affine invariant pattern recognition, *Proc.of IJCNN'93*, pp.181-184, 1993.
- [9] K.Kanatani: *Group-Theoretical Methods in Image Understanding*, Springer-Verlag, 1990.

- [10] S.Ren and S.Kurogi: Learning Algorithms using Firing Numbers of Weight Vectors for WTA Networks in Rotation Invariant Pattern Classification, IEICE Trans. on FECCS, Vol.e81-A, No.1, 1998.
- [11] S.Kurogi, T.Nishida and S.Ren: Recognition of Hand-Written Characters by A Multi-Layered Competitive Net, Proc.of ICONIP'98, Vol.2, pp.1086-1089, 1998.
- [12] 山口 富士夫 : コンピュータディスプレイによる図形処理工学, 日刊工業新聞社, 1990.
- [13] K.I.Diamantaras and S.Y.Kung: Principal Component Neural Networks, John Wiley & Sons, Inc., 1996.
- [14] R.M.Gray: Vector quantization, IEEE ASSP Magazine, pp.4-29, April, 1984.
- [15] J.E.Fowler: Adaptive vector quantization for the coding of nonstationary sources, SPANN Laboratory technical report TR-95-05, The Ohio State University, April, 1995.
- [16] Y.Linde, A.Buzo and R.M.Gray: An algorithm for vector quantizer design, IEEE Trans. on Communications, Vol.com-28, No.1, pp.84-95, January, 1980.
- [17] A.Gersho and M.Yano: Adaptive vector quantization by progressive codevector replacement, Proc. of the International Conference on Acoustics, Speech, and Signal Processing, pp.4.6.1-4.6.4, 1985.
- [18] N.Ueda and R.Nakano: A new competitive learning approach based on an equidistortion principle for designing optimal vector quantizers, Neural Networks, Vol.7, No.8, pp.1211-1227, 1994.
- [19] S.Grossberg: Adaptive pattern classification and universal recoding: I. Parallel development and coding of neural feature detectors, Biological Cybernetics, Vol.23, pp.121-134, 1976.
- [20] T.Kohonen, Self-organization and associative memory, Springer-Verlag, New York, 1984.

- [21] C.Chinrungrueng and C.H.Séquin: Optimal adaptive K-means algorithm with dynamic adjustment of learning rate, IEEE Trans. on Neural Networks, Vol.J82-D-II, No.4, pp.807–817, 1999.
- [22] 芳澤 伸一, 道木 慎二, 大熊 繁: 変化する入力データに対する学習の定常特性と過渡特性を改善する新しいベクトル量子化アルゴリズム, 信学論 (D-II), Vol.J82-D-II, No.4, pp.807–817, 1999.
- [23] M. Iri, K. Murota and T. Ohya: A fast Voronoi-diagram algorithm with applications to geographical optimization problems, Lecture Notes in Controls and Information Science, 59, System Modelling and Optimization, eds. P.Thoft-Christensen, pp.273–288, Springer-Verlag, 1984.
- [24] A.Gersho: Asymptotically optimal block quantization, IEEE Transactions on Information Theory, Vol.28, No.2, pp.157–166, 1979.
- [25] S.Kurogi and T.Nishida: Competitive learning using gradient and reinitialization methods for adaptive vector quantization, Proc. of NNSP2000, pp.346–355, 2000.
- [26] 西田 健, 黒木 秀一: 多層競合ネットによる座標変換に不変なパターン認識, 日本神経回路学会誌, Vol.7, No.4, pp.106–114, 2000.
- [27] 西田 健, 黒木 秀一: 再初期化法を用いた適応ベクトル量子化, 信学論, Vol.J84-D-II, No.7, pp.1503–1511, 2001.
- [28] T.Nishida, S.Kurogi and T.Saeki: An Analysis of Competitive and Reinitialization Learning for Adaptive Vector Quantization, Proc. of IJCNN2001, pp.978–983, 2001.