

5 R-2

最適化 feature map と強制学習との結合

黒澤 泰 松山 泰男

茨城 大学

1. はじめに

feature map (特徴地図) とは、多次元のサンプルデータをより低い次元のデータに縮約した結果として現れるパターンである。これは、教師無しの学習 (自己組織化) を行うことにより生成することができる。競合学習は、この特徴地図を学習によって生成するアルゴリズムとして知られており広く利用されている^[1]。

一方、学習法を分類すると、もう一つのクラスとして教師有りの学習がある。この場合、サンプルデータの個々に対してラベルやコンセプトのような教師信号を与えておく必要があり、その手間はかなりのものとなる。そこで自己組織化と教師有り学習との結合を用いて、学習システム全体の高度化と効率化を図ろうという考え方が生じる。この報告では特に学習を用いた情報処理の高度化の方に重点を置いている。

ところが、自己組織化の途中あるいは結果に対して、強制学習を適用するという考えに立つてみると、通常の自己組織化手法や特徴地図では、ヒューマンインターフェースの面において外部知性との結合に難点があることがわかる。これに対して、多重降下競合学習^[2]により得られる特徴地図は最適化の過程を含んでおり、外部知性との結合を行い易い程度に原情報を抽出できる。

本稿では、上に述べた自己組織化と強制学習の結合、そして多重降下競合学習による特徴地図の利用について、画像処理、特に表情の変形を対象としている。

2. 多重降下競合学習による feature map の生成

今回使用した多重降下競合学習は、メッシュパターンを最適化していくフェーズと標準ピクセルパターンを最適化していくフェーズを繰り返すことによって行われている。ここで標準ピクセルパターンの最適化フェーズに注目すれば、これはベクトル量子化に次のような拡張を施したものと考えることができる。

- コードブック (標準ピクセルパターン) の形が三角形である。
- 要素領域 (コードブックと対応づけられる画像上の領域) の形が変形可能である。(基本形は直角二等辺三角形)
- 要素領域の境界線上のピクセルが、2つ以上の要素領域に共有されることがある。

[三角形コードブック]

我々は従来、通常の正方形コードブックとそれに対応する凸四角形のメッシュパターンを使用してきたが、メッシュパターンの可変性の向上と sqrt を使用しない座標変換が可能な三角形に注目し、コードブックとメッシュパターンに三角形のものを使用することにした。

コードワードは直角二等辺三角形の形態となっており、後述の座標変換式とピクセル値補間式によって、メッシュパターンの領域内の点に対応付けられている。この対応は、従来のベクトル量子化に用いられてきたピクセルの1対1対応の関係ではなく、コードワードやメッシュパターンの領域を一枚のプレートとしてそのプ

レートどうしの対応関係と考えている。つまり、コードワード内に含まれているピクセル数とメッシュパターンの領域内に含まれているピクセル数は、一致する必要がある。

メッシュパターンの領域プレートに対し最も類似したコードワードを探す際には、コードワードプレートを領域プレートと同形に変形して、領域内の全ピクセルに対応するコードワード内の仮想ピクセル値を求める必要がある。またコードワードを更新する際にはこれとは逆に、領域プレートをコードワードプレートと同形に変形して、コードワード内の全ピクセルに対応するメッシュパターンの領域内の仮想ピクセル値を求める必要がある。

この、プレートの変形とは、座標系の変換と考えることができ、まず後者の場合、次式によって変換が行われる。

$$\begin{aligned}x' &= cx \times x + bx \times y \\y' &= cy \times x + by \times y\end{aligned}\quad (1)$$

ここで x, y は、コードワード内のピクセルの座標で、 x', y' が対応する領域内の仮想ピクセル座標である。

さらに、この仮想ピクセルのピクセル値を求める際、メッシュパターンの領域内のピクセルだけでなく原画像全体のピクセルの利用を許せば、次の補間式により仮想ピクセル値を求めることができる。

$$\begin{aligned}P &= (1-x) \times (1-y) \times P_0 + (1-x) \times y \times P_1 \\&+ x \times y \times P_2 + (1-x) \times y \times P_3\end{aligned}\quad (2)$$

また、メッシュパターンの領域内のピクセルに対応するコードワード内の仮想ピクセルの座標を求めるには、(1) 式を x, y について解いた式を使用する。

$$\begin{aligned}x &= (bx \times y' - by \times x') / (bx \times cy - by \times cx) \\y &= (cy \times x' - cx \times y') / (bx \times cy - by \times cx)\end{aligned}\quad (3)$$

この仮想ピクセルのピクセル値を求める際に、仮想ピクセルの周りに4個の実ピクセルが存在する場合は前の補間式(2)がそのまま使用できる。仮想ピクセルの周りに3個の実ピクセルしか存在しない場合は、次の補間式を使うことにする。

$$P = (1-x-y) \times P_0 + y \times P_1 + x \times P_2 \quad (4)$$

[学習アルゴリズム]

STEP 1

サンプルデータ、初期メッシュパターン、初期標準ピクセルパターン、終了条件が与えられる。

STEP 2

メッシュ上の任意の頂点を選び、その位置を歪が小さくなる様な位置に移動する。

この時、その周りの領域に関して最も誤差の小さい標準ピクセルパターンが決定される。

STEP 3

上で選ばれた標準ピクセルパターンについて、対応する領域内のピクセルパターンに近づける。

STEP 4

終了判定を行い、条件が満たされていない場合は STEP 2 へ。

なお、STEP 2 に対する実際のインプリメントでは、頂点の移動先を候補の中からランダムに選び、その点を選んだ場合に誤差が減少するかどうか計算して、実際に移動するかどうか決定している。

頂点を移動させる際、頂点の周りの領域が三角形を保てないような移動や、メッシュパターンが裏返るような移動は許さないものとしている。

3. 強制情報の注入

上記の学習アルゴリズムによってある程度学習が進んだ時点での feature map に対して制約情報を課してみる。具体的には、人間の直感により現在のメッシュパターンにおいて適している部分（原画像の特徴がよく現れている部分）を指定し、この部分は後で再開する学習においては移動を禁止する。なおこの時、feature map を人間が理解しやすい形態で対話的にアクセスできるような、ユーザーインターフェースを備えたシステムが必要となる。

原画像がある程度大きなものになると、メッシュパターンはかなり細かいものが生成され、このままでは制限情報を加える際に指定が難しくなる。そこで、このメッシュパターンの中から特徴が強く現れていると思われる部分、つまり原画像において濃度の差が大

きい境界と重なっているエッジを抜き出し、これを参考にして制限情報を注入する。

ここでこの代表エッジを決定する際に、学習の過程で生成されるもう一つの feature map である標準ピクセルパターンを利用する。メッシュの全エッジに対して、それを両側からはさむ2つの領域に対応する標準ピクセルパターンを比較し、削除するかどうか決定する。

4. まとめ

多重降下競合学習により得られる2種類の特徴地図を利用して、画像の変形を試みた。実例としては、最適なピクセルグループ化により得られる特徴地図をベースにして、表情の変形を与えた。これは、自己組織化と強制学習の結合という観点から、意義深いと考えられる。

[参考文献]

- [1] Kohonen, T. Self-Organization and Associative Memory, Springer (1984).
- [2] Matsuyama, Y. Multiple descent cost competitive learning, Proc. IJCNN-San-Diego, II, pp.299-306 (1990)



図1 特徴地図のエッジの絞り取り



図2 特徴地図への情報注入を用いた表情の変形