原著論文

高階 SOM による形状表現マップの自己組織化 －トポロジー拘束のない形状空間法－[†]

大谷 誠*・古川 徹生*

本研究の目的は、形状集合を取り扱う形状空間法を理論的に確立し、さらに形状のトポロジーに関する事前知識がない場合でも形状空間が推定可能なアルゴリズムを開発することである。この目的を実現するため、まず形状集合の確率的生成モデルを多様体を用いて構築した。次にトポロジー制約のない自己組織化マップ TFSOM を開発し、これを高階 SOM に組み込んだ TFSOM×SOM を作った。さらに形状の非均一な伸縮にも対応できるようにアルゴリズムを拡張した。提案手法を手書き数字識別に応用したところ、10 人分の手書き数字を学習させるだけで 1000 人分の手書き数字を 95% の精度で識別できた。本手法は線画に特化したものではないため、フォールのデザイン分析などさまざまな形状に応用が可能である。本提案手法は形状集合を扱う汎用的な手法を提供する。

キーワード：自己組織化マップ、高階 SOM、ニューラルガス、形状空間法、形状表現

1. はじめに

同じクラスに属する物体形状の集合は、しばしばそのクラスに固有の形状空間を作る。たとえば「さまざまなコーヒーカップの形状集合」というクラスは、コーヒーカップに固有の形状空間を作る。この形状空間は、各点がそれぞれ異なるコーヒーカップの形を表現しており、さらにモーフィングによる形状の連続変化や形状間距離の定義できる位相空間である。このような形状空間を扱う理論は形状空間論と呼ばれ、Kendall によって構築された [1, 2, 3, 4, 5]。また形状空間論に基づいて形状解析や形状情報処理、形状識別等を行う方法を本稿では形状空間法と呼ぶ。

一般に一つの形状は一つの多様体として表現できる。したがって形状空間は多様体の集合によって構成される高次の多様体であり、ファイバー束のような積多様体やホモトピーのような連続変化する写像として表現できる。そのため形状空間法には多様体学習や部分空間法が要素技術として取り入れられる。代表例がカーネル PCA であり、また Kohonen の自己組織化マップ (Self-Organizing Map: SOM) もしばしば用いられる [6, 7]。形状空間法を実現するにはこうした多様体学習を階層的に組み合わせる必要があり、その手法のひとつとして高階 SOM が挙げられる [8, 9]。高階 SOM

はデータ分布をファイバー束として表現する能力があり、形状空間法に適したアルゴリズムとなっている。Yakushiji らは高階 SOM による形状空間法を実現し、その有効性を示した [10, 11, 12]。

さて積多様体を用いて形状空間を表現するためには、通常、その形状クラスのトポロジーがあらかじめわかっている必要がある。たとえば輪郭線形状の場合は、対象クラスに関わらず閉曲線とわかっているので容易である。しかしながら、もしトポロジーが事前にわからない場合はより難しい課題となる。その典型的な場合が手書き文字をはじめとする線画形状である。たとえばアルファベットの手書き文字の場合、同一クラスの文字（たとえば 'A'）はどれも同じトポロジーを持つが、しかし異なる文字種（たとえば 'A' と 'B'）では必ずしもトポロジーは一致しない。この点が輪郭線形状の場合と決定的に異なる。

本研究の目的は、高階 SOM を拡張することでトポロジーに拘束されない形状空間法を開発することである。すなわち何かの形状クラスが与えられたとき、そのクラス固有の形状空間推定と、そのクラス固有のトポロジー推定を同時に行う手法の開発である。特に手書き文字などの線画によって構成された形状集合に対して応用可能な形状空間法のアルゴリズム開発を行うことを目的とする。

本研究ではトポロジー制約のない形状空間法を実現するため、次の 5 点に焦点を当てて研究を行った。第一は、トポロジー制約をなくした SOM、すなわち Topology Free SOM (TFSOM) を開発することである。

[†] Self-Organization of Shape Space by the Higher-Rank SOM – Topology-Free Shape Space Method –
Makoto OTANI and Tetsuo FURUKAWA

* 九州工業大学
Kyushu Institute of Technology

これにより1個の形状をトポロジー制約なしで表現することが可能になる。第二は、TFSOMを高階SOMに組み込んだTFSOM×SOMを開発することである。これにより基本となるアルゴリズムができる。第三は、TFSOM×SOMを形状空間法に応用する際に問題となるいくつかの課題を解決することである。第四は、開発したTFSOM×SOMによる形状空間法を手書き文字識別に応用することである。そして第五に、形状空間法という概念自体を本研究の立場で再構築することである。

本論文の構成は以下のとおりである。第2節では研究背景となる形状空間法や関連研究について説明し、形状空間法の再構築を試みる。また本論文で取り組む問題点について述べる。第3節では提案手法であるTFSOMおよびTFSOM×SOMのアルゴリズムについて述べる。第4節では人工データを用いて提案手法の評価を行い、さらに手書き文字識別への応用結果を報告する。第5節は討論である。

2. 研究背景

2.1 形状記述法と形状表現法

物体の形状情報は視覚画像を理解する上で重要な手掛かりである。しかしながら形状そのものを直接数値化することは容易ではなく、なんらかの方法で数ベクトル化して取り扱う必要がある。形状を取り扱う方法は大きく分けて形状記述法(shape description)と形状表現法(shape representation)に分けられる[13, 14]。広く用いられるのは形状記述法であり、形状の持つなんらかの特徴に着目して特徴ベクトル化するのが形状記述法である(e.g., [15, 16, 17, 18])。形状記述法は簡便であり、良い特徴量が見つければ高い精度で形状認識を行うことが可能である。また特徴量をうまく選べば、拡大・縮小や回転などの変換に対してもロバストにできる。ただし何が良い特徴量かはタスクに依存す

るため、あるタスクにとって良い特徴量が他のタスクにとっても良いとは限らない。したがって形状記述法は一般にタスクとセットで考える必要がある。また特徴ベクトル化の際に形状自体の情報が失われるため、特徴ベクトルから元の形状を復元することは一般にできない。また2つの形状の中間的な特徴ベクトルは、必ずしも中間的な形状に対応するとは限らない。

一方、形状表現法は形状そのものをモデル化する方法である。もっとも簡便な方法は、輪郭線(2次元形状の場合)や形状表面(3次元形状の場合)を点の並びとして扱う方法である(図1)。このような点をランドマーク点と呼ぶ。1個の形状を N 個のランドマーク点で表現する場合、 $\mathbf{r} = (\mathbf{y}_1, \dots, \mathbf{y}_N)$, $\mathbf{r} \in \mathcal{R} = \mathbb{R}^{D \times N}$ が形状全体を表現するベクトルとなる。またこれらのランドマーク点をスプライン曲線で結べばより詳細な表現が可能になる[19, 20]。ここで $\mathbf{y}_n \in \mathbb{R}^D$ は D 次元ユークリッド空間における第 n 番ランドマーク点の位置ベクトルであり、 $\mathcal{R} = \mathbb{R}^{D \times N}$ は形状表現空間(shape representation space)である。また輪郭形状のような2次元形状ならば $D=2$ 、物体の表面形状の場合は $D=3$ となるが、色や線分方向などの局所特徴量を加えた高次元空間を考えることもできる。

より一般的には、形状という概念は次のように扱える。ひとつの形状はユークリッド空間 $\mathbb{R}^D$ 内の多様体 Q として表される。本稿では Q を輪郭多様体(outline manifold)と呼ぶこととする。一方、 Q を抽象化した位相同形な空間(固有輪郭空間: intrinsic outline space)を $\mathcal{P}$ とする。2次元図形の輪郭形状ならば $\mathcal{P}$ は単位円であり、3次元曲面ならば単位正方形を考えればよい。すると $\mathcal{P}$ から Q への写像 $f: \mathcal{P} \rightarrow Q$, $\xi \mapsto \mathbf{x}(\xi)$ が定義できる。このとき関数 $\mathbf{x}(\xi)$ が形状に該当する。また形状表現ベクトル $\mathbf{r}$ は、 $\mathbf{x}(\xi)$ の関数空間における位置ベクトルに相当する。したがってランドマーク点を使わなくても、 $\mathbf{x}(\xi)$ をフーリエ展開等の基底関

図1 ランドマーク点を用いた形状表現法。第 n ランドマーク点の座標を $\mathbf{y}_n$ とすれば、ランドマーク点列 $\mathbf{r} = (\mathbf{y}_1, \dots, \mathbf{y}_N)$ が形状全体を表す。

数展開で表現することも可能である[21]。なお ξ は一般に直接観測することができないので、潜在変数として取り扱う。

形状表現法の長所は情報の欠落がないことや、モーフィングなどの形状操作が容易なこと、特定のタスクに依存しない汎用的な形状解析や形状処理が可能であること等が挙げられる。また自然画像の中から物体形状を抽出する際にも形状表現法が用いられる。しかし形状表現法は一般に計算コストがかかる。形状記述法と形状表現法はどちらが優れているというものではなく、目的に応じて相補的に使い分けたり、組み合わせたりするものである。

2.2 形状空間法

形状空間法は着目するクラスの形状集合、すなわち形状のさまざまなバリエーションを扱うことを目的とする。形状空間法は形状表現法に基づくタスク非依存の手法であり、タスクに応じてパターン識別器等と組み合わせて利用する。したがって特徴ベクトルを用いる形状記述法と比較・競合する性質のものではない。また形状記述法と組み合わせて使うことも可能である。

形状空間という概念はKendallによって提案され、形状空間論として体系化された[1, 2, 3]。またYakushijiらは形状空間法を確率的生成モデルとして表現し、EMアルゴリズムによる潜在変数推定問題に帰着した[10, 11]。両者は本質的に同じものであるが、ここではYakushijiらの生成モデル的アプローチを元にしつつ、

形状空間法をより掘り下げた説明を試みる(図2)。したがって以下に述べる形状空間法の定式化も本研究の一部である。

今、魚という形状クラスについて考えよう。魚の形状クラスの固有形状空間(intrinsic shape space)を S とする。 S はあらゆる魚形状が詰まった空間で、任意の $\sigma \in S$ はひとつの魚形状に対応する。形状空間 S は単に魚形状の集合であるだけでなく、形状の連続変化や形状間の距離が定義できる距離空間である。このような S は直接知ることはできないので、 S は潜在空間である。

一方、 $\sigma \in S$ に対応する形状を、形状表現法でベクトル化したものを $\mathbf{r}(\sigma)$ とすれば、埋込み写像 $g: S \rightarrow \mathcal{M} \subset \mathcal{R}$, $\sigma \mapsto \mathbf{r}$ を定義することができる。すなわち形状表現空間 $\mathcal{R}$ における固有形状空間 S の像は位相同形な多様体 $\mathcal{M}$ となる。本稿では $\mathcal{M}$ を形状多様体(shape manifold)と呼ぶ。

さらに一個の形状はユークリッド空間 $\mathbb{R}^D$ 内の輪郭多様体 Q として表される。したがって σ に対応する形状は写像 $\mathbf{x}(\xi|\sigma)$ となる。このような写像はホモトピーと呼ばれる。また輪郭点 $\mathbf{x}$ の集合は形状多様体 $\mathcal{M}$ と輪郭多様体 Q の積多様体、すなわちファイバー束 $\mathcal{F}$ となる。そして形状空間は、固有形状空間 S 、固有輪郭空間 $\mathcal{P}$ 、ファイバー束 $\mathcal{F}$ の組 $(S, \mathcal{P}, \mathcal{F})$ として定義される。

ここで重要な点は、同一の ξ に対応する輪郭点 $\mathbf{x}(\xi|\sigma)$ は、各々の形状の中で同一の役割を担うという

図2 形状空間の概念図

ことである。たとえば魚のヒレの先端に相当する点は、たとえ形状が異なってもすべて同じ ξ に対応する。すなわち「ヒレの先端」というランドマーク点に対応する。第 n ランドマーク点 y_n の輪郭座標を ξ_n とすれば、 $y_n(\sigma) \equiv x(\xi_n|\sigma)$ はランドマーク点がモーフィングにより描く軌跡になる。これはファイバー束における 1 本のファイバーに相当する。そこで本稿ではこれをランドマークファイバー (landmark fiber) と呼ぶ。

形状空間法とは、学習用の形状集合 $\{S_1, \dots, S_l\}$ が与えられたときに、これらの形状集合が属する形状空間 $(S, \mathcal{P}, \mathcal{F})$ を推定することがタスクである。そのためには以下の 3 つの推定を同時に行う必要がある。(1) 各形状の輪郭点に対応する固有輪郭空間の位置ベクトル ξ を推定すること、(2) 固有形状空間における各形状の位置ベクトル $\{\sigma_i\}$ を推定すること、(3) ホモトピー $x(\xi|\sigma)$ を推定すること。これは潜在変数 ξ, σ と位相保存写像 $x(\xi|\sigma)$ の同時推定問題に帰着でき、EM アルゴリズムの枠組みで解くことができる。

形状空間法は形状そのものを扱うのであって、画像を扱うわけではない。図 3 は本手法で推定した手書き数字「8」の形状空間 (形状マップ) と、画像を直接 SOM に学習させた結果 (イメージマップ) を比較したものである。形状空間法ではランドマーク点の連続的移動で形状から形状への変化を表現する。一方、画像を直接 SOM で扱ったものはピクセルイメージの重ね合わせ、すなわちクロスディゾルブでイメージからイメージへ

の変化を表現する。したがって両者はまったく異なるものである。

2.3 SOMを用いた形状表現法

形状表現法の中で、SOMを代表例とする位相保存写像を用いる方法がある [22, 23, 24]。たとえば輪郭線形状を表現する場合、輪郭点をデータ点とみなして環状のトポロジーを持つ SOM で学習させれば、各ユニットがランドマーク点を表現することになる。また Neural Gas (NG) のようなベクトル量子化法や Topology Representing Network (TRN)、Evolving SOM (ESOM) などのグラフ学習型のアルゴリズムを使うことでトポロジーに依存しない形状表現が可能になる [25, 26, 27]。

SOM はマップ空間と呼ばれる潜在空間から高次元のデータ空間への写像を学習するアルゴリズムである。また同時に、観測データに対応する潜在変数を並行して推定する。形状表現法 (図 1) に置き換えれば、輪郭点集合 $\{x_j\}$ というデータ集合から、写像 $x(\xi)$ および潜在変数 $\{\xi_n\}$ を推定することに相当する。またマップ空間は固有輪郭空間 $\mathcal{P}$ に該当する。また SOM のユニットはランドマーク点 $\{y_n\}$ に対応する。

SOM を用いた輪郭線形状表現のアルゴリズムを簡潔に述べると以下ようになる。まず輪郭線をドットに分解し、各ドットの位置ベクトルをデータ点とする。2 次元形状を J 個のドットに分解した場合は、 X

図 3 (a) 本提案法による形状空間の推定結果 (形状マップ)、(b) ピクセルイメージを SOM で学習したイメージマップ。形状マップは形状集合のモーフィングを表現する。一方イメージマップは画像集合のクロスディゾルブを表現する。

$=\{\mathbf{x}_1, \dots, \mathbf{x}_J\}$, $\mathbf{x}_j \in \mathbb{R}^D$ が学習用データ点となる。一方 SOM にはランドマーク点と同数のユニットを用意する。各ユニットにはデータ空間とマップ空間の位置ベクトル $\mathbf{y}^n, \xi^n$ が割り当てられる*1。このうち ξ^n は固定であり、 $\mathbf{y}^n$ は学習により更新する。マップ空間が輪郭空間 $\mathcal{P}$ に対応し、データ空間がユークリッド空間 $\mathbb{R}^D$ に対応する。したがって SOM のマップは $\mathcal{P}$ と同じトポロジーにする必要がある。また参照ベクトルを連結した $\mathbf{r} = \bigoplus_n \mathbf{y}^n$ が形状表現ベクトルとなる。

学習アルゴリズム自体は通常の SOM と同じである。SOM は EM アルゴリズムの一種なので、E-step と M-step の繰り返しで計算する。

■E-step (SOM)

E-step では潜在変数 ξ^n の事後分布を推定する。

$$E_j^n = \frac{1}{2} \|\mathbf{x}_j - \mathbf{y}^n\|^2 \quad (1)$$

$$n_j^* = \arg \min_n E_j^n \quad (2)$$

$$\beta_j^n = \exp \left[-\frac{1}{2\lambda^2} d^2(n_j^*, n) \right] \quad (3)$$

ここで $d(n_1, n_2) = \|\xi^{n_1} - \xi^{n_2}\|$ であり、2 つのユニット間のマップ上での距離を表す。なお ξ^n が第 n ユニットのマップ上での位置である。SOM ではトポロジーが既知である必要があるが、それは $d(n_1, n_2)$ を事前に決めることにほかならない。

■M-step (SOM)

M-step では誤差の期待値を最小にする写像を求める。すなわち形状表現ベクトル $\mathbf{r}$ を最尤推定する。

$$B^n = \sum_j \beta_j^n \quad (4)$$

$$\mathbf{y}^n := (1 - \varepsilon) \mathbf{y}^n + \varepsilon \frac{1}{B^n} \sum_{j=1}^J \beta_j^n \mathbf{x}_j \quad (5)$$

ε は 1 以下の正定数であり、小さいほど学習が安定化するが、その代わり学習が遅くなる。通常の SOM の場合は $\varepsilon = 1$ で問題ない。

一方 SOM と異なり NG にはトポロジーの制約がない。SOM が弾性固体とするならば NG は文字通り気体のようにふるまう。SOM と NG の違いは E-step のみである。

■E-step (NG)

$$\beta_j^n = \exp \left[-\frac{1}{2\lambda^2} d^2(n_j^*, n) \right] \quad (6)$$

*1 本稿では、データベクトルに関する添字を下付き、SOM に関する添字を上付きで表した。

$$d(n_j^*, n) = \text{RANK}_{n'=1, \dots, N}(E_j^n; E_j^{n'}) \quad (7)$$

ここで $\text{RANK}_n(E_j^n; E_j^{n'})$ は $\{E_j^1, \dots, E_j^N\}$ を昇順にソートした時の E_j^n の順位を与える関数 (Excel の RANK 関数) である。具体的に書けば

$$\text{RANK}_{n'=1, \dots, N}(E_j^n; E_j^{n'}) := \sum_{n'=1}^N \theta(E_j^n - E_j^{n'}) \quad (8)$$

である。なお $\theta(\cdot)$ は単位ステップ関数である。また NG の M-step は SOM と同じである。

NG の場合、ユニット間の近傍関係は直接与えられず、必ずデータ点を中心とした近傍順位で決まる。そのため形状のトポロジーに関する事前知識は必要としない。なお NG の場合は M-step の ε を 0.1 ないし 0.5 程度のやや小さめにとる。

ひとつの形状を表現するには SOM や NG を単体で用いればよい。しかし形状空間法の場合は形状のバリエーションを表現することが目的であるので、複数の形状を学習する必要がある。そのためには階層的な多様体の学習が必要となる。そこで SOM を階層化した高階 SOM を用いる必要が生じる。

2.4 高階 SOM による形状空間法

今、 I 個の学習用形状 $\{S_1, \dots, S_I\}$ が与えられたとする。このとき I 個の SOM を用いれば、それぞれの形状表現ベクトル $\{\mathbf{r}_1, \dots, \mathbf{r}_I\}$ が得られる。これらは輪郭多様体 $\{Q_i\}$ に対応する。そこでもうひとつ上位の SOM を用意し、下位 SOM の学習結果 $\{\mathbf{r}_i\}$ をデータベクトルとして学習させれば、原理的には形状空間が推定できる。すなわち上位の SOM は形状多様体 $\mathcal{M}$ を推定する。これが高階 SOM の基本的な考え方である。高階 SOM では下位の SOM を 1st SOM、上位の SOM を 2nd SOM と呼ぶ。また 2 次の高階 SOM は SOM^2 と表記されるが、これは 1st SOM と 2nd SOM の積空間 $\text{SOM} \times \text{SOM}$ を意味する。

1st SOM 集合の学習結果を 2nd SOM に学習させるという素朴なアルゴリズムは、一見するとうまく動きそうに見えるが、それだけでは期待通りに動かない。その理由は、各 1st SOM がそれぞれ勝手にランドマーク点を決めてしまうため、2nd SOM がそれらをつなぎ合わせてもランドマークファイバーが正しく形成されないからである。そこで SOM^2 では、2nd SOM が推定したランドマークファイバーの情報を 1st SOM にフィードバックし、ランドマーク点の対応関係が正しく取れるようにする。これは 2nd SOM の学習結果を用いて 1st SOM の事前分布を与えることに相当する。

SOM^2 を用いた形状空間法のアプローチは以下の

ようになる。第 i 形状から抽出した輪郭ドットの座標データを $X_i = \{\mathbf{x}_{i1}, \dots, \mathbf{x}_{ij}\}$ とする。(ドット数は形状ごとに異なってもかまわないが、話を単純にするため、ここではドット数が等しいと仮定する)。一方、SOM² は I 個の1st SOMと1個の2nd SOMにより構成される。1st SOMの参照ベクトルを $\mathbf{y}_i^n \in \mathbb{R}^D$ とすれば、 $\mathbf{r}_i = \bigoplus_n \mathbf{y}_i^n$ が S_i の形状表現ベクトルとなる。また2nd SOMの参照ベクトルを $\mathbf{u}^m = (\mathbf{z}^{m1}, \dots, \mathbf{z}^{mN}) \in \mathbb{R}^{D \times N}$ とする。SOM² のアルゴリズムは以下ようになる[8, 9]。

■E-step(1st SOM)

1st SOMのE-stepでは各輪郭点 $\mathbf{x}_{ij}$ に対して、対応する固有形状空間の位置 ξ_{ij} を推定する。

$$E_{ij}^n = \frac{1}{2} \|\mathbf{x}_{ij} - \mathbf{y}_i^n\|^2 \quad (9)$$

$$n_{ij}^* = \arg \min_n E_{ij}^n \quad (10)$$

$$\beta_{ij}^n = \exp \left[-\frac{1}{2\lambda^2} d^2(n_{ij}^*, n) \right] \quad (11)$$

■M-step(1st SOM)

次に各形状表現ベクトル $\{\mathbf{r}_i\}$ を推定する。これは各形状の輪郭多様体 $\{Q_i\}$ を求めることにほかならない。

$$B_i^n = \sum_{j=1}^J \beta_{ij}^n \quad (12)$$

$$\mathbf{y}_i^n := (1 - \varepsilon) \mathbf{y}_i^n + \varepsilon \frac{1}{B_i^n} \sum_{j=1}^J \beta_{ij}^n \mathbf{x}_{ij} \quad (13)$$

$$\mathbf{r}_i = \bigoplus_{n=1}^N \mathbf{y}_i^n \quad (14)$$

■E-step(2nd SOM)

2nd SOMのE-stepでは、各形状の固有形状空間における位置 $\{\sigma_i\}$ を推定する。

$$E_i^m = \frac{1}{2} \|\mathbf{r}_i - \mathbf{u}^m\|^2 \quad (15)$$

$$m_i^* = \arg \min_m E_i^m \quad (16)$$

$$\alpha_i^m = \exp \left[-\frac{1}{2\lambda^2} d^2(m_i^*, m) \right] \quad (17)$$

■M-step(2nd SOM)

2nd SOMのM-stepでは形状多様体 $\mathcal{M}$ を推定する。

$$A^m = \sum_i \alpha_i^m \quad (18)$$

$$\mathbf{u}^m := (1 - \varepsilon) \mathbf{u}^m + \varepsilon \frac{1}{A^m} \sum_{i=1}^I \alpha_i^m \mathbf{r}_i \quad (19)$$

■Copy Back(2nd to 1st SOM)

最後に2nd SOMの学習結果を1st SOMにフィードバックする。

$$\mathbf{r}_i := \mathbf{u}^{m_i^*} \quad (20)$$

以上の5ステップを繰り返して計算することで次第にランドマークファイバーが揃い、形状空間が推定される[10, 11, 12]。最終的に得られる3次テンソル $\mathbf{U} = (\mathbf{u}^m) = (\mathbf{z}^{mn})$ が形状多様体 $\mathcal{M}$ および形状ファイバー束 $\mathcal{F}$ を表現する。

2.5 本研究における論点

以上をふまえ、固有輪郭空間 $\mathcal{P}$ のトポロジーが未知の場合の形状空間法を開発することが本研究の目的である。本研究の位置づけは、できるかぎり汎用的な形状空間アルゴリズムを開発することである。したがってタスクに依存した事前知識や作りこみ的なアルゴリズムは極力用いないこととする。

本研究での主な論点は三点ある。第一点は、どのようにしてトポロジーに依存しない形状空間表現を実現するかという点である。第二点は、(形状空間法に共通の問題であるが)どのようにしてランドマークファイバーを正しく推定するかという点である。すなわち異なる形状の間に正しい対応関係を見つけるかという問題である。第三点は、本研究に即した形状空間法の理論的な再構築であり、本節で述べた内容である。以下、第一と第二の論点について詳しく述べる。

第一の論点、トポロジー非依存な形状表現法として、SOMの代わりにNGやその発展形であるTRNを使う方法がすでにある。しかしながらこれらは単一形状の表現法にすぎない。形状空間法として用いるにはSOM² のような階層的アルゴリズムが必要となる。しかし1st SOMをNGに置き換えたNG×SOMにはひとつ大きな欠陥がある。NGはデータ分布のトポロジーにまったく拘束を受けないため、データ分布のトポロジーという概念自体が消失する。一方で手書き文字の‘A’という形状クラスは、どの形状も同じトポロジーを持っており、トポロジーという概念自体が消失したわけではない。すなわちトポロジーが既知という制約は回避したいが、トポロジーという概念自体は残しておきたい。NGの持つこの問題点は今まで指摘されることがなかった。なぜならば、この問題はNG×SOMのように階層化によって初めて生じる問題であり、NGを単独で用いる限りは問題なかったためである。この問題を解決することが本研究の第一の論点である。

第二の論点は、どのようにして異なる形状間に共通のランドマークファイバーを見つけるかという問題で

ある。この論点はさらに二つの課題に分けられる。ひとつめの課題は非均一な伸縮に対する対応である。形状がモーフィングする場合、ある領域は縮み別の領域が伸びることがしばしば生じる。それにより形状の各領域を構成するドット数が非均一に増減する。すなわち縮んだ領域は見かけ上のデータ点数が減り、伸びた領域は見かけ上のデータ点数が増える。こうした非均一な伸縮にロバストなアルゴリズムを作らなくてはならない。ふたつめの課題は局所的な特徴量の利用である。輪郭線の折れ曲がりなど、局所的な特徴は形状間の対応関係を発見する良い手掛かりである。そこで局所特徴量をデータベクトルに織り込むことで、より正確なランドマークファイバーの推定が可能になると期待できる。ただし本研究は特定の目的に特化しないアルゴリズム開発を目的とするため、できるだけ一般的な形でアルゴリズムに埋め込みたい。以上の二つの課題を取り上げるのが第二の論点である。

以上が本研究で取り上げる主な論点である。

3. TFSOM × SOM のアルゴリズム

3.1 Topology Free SOM: TFSOM

形状空間法におけるSOM²の問題点は、形状のトポロジー、すなわち固有輪郭空間 $\mathcal{P}$ のトポロジーの事前知識を必要とすることであった。その事前知識を利用して1st SOMのマップ上でのユニット配置を統一化することができた。一方NG×SOMの問題点は、形状トポロジーの制約をまったく受けないため、各1st NGがそれぞれ自由にふるまってしまうことであった。

われわれがほしいのは次のような条件を満たすベクトル量子化アルゴリズムである。(1)ひとつの形状 S_i におけるランドマーク点 $\mathbf{y}_i^n$ と $\mathbf{y}_i^{n'}$ が互いに近傍であるならば、他の形状 $S_{i'}$ におけるランドマーク点 $\mathbf{y}_{i'}^n$ と $\mathbf{y}_{i'}^{n'}$ も互いに近傍となつてほしい。(2)しかし一方で、 $\mathbf{y}_i^n$ と $\mathbf{y}_i^{n'}$ が近傍かどうかは事前に決めず、データを学習することで推定したい。SOM²は(1)の条件を満たすが(2)は満たさない。一方でNG×SOMは(2)を満たすものの(1)を満たさない。

われわれが提案するのは、SOM、NGのE-stepを次のように修正したアルゴリズムである。

■E-step(TFSOM)

$$n_j^* = \arg \min_n \|\mathbf{x}_j - \mathbf{y}^n\|^2 \quad (21)$$

$$\beta_j^n = \exp \left[-\frac{1}{2\lambda^2} d^2(n_j^*, n) \right] \quad (22)$$

$$d(n_1, n_2) = \text{RANK}(E^{n_1, n_2}; E^{n_1, n'}) \quad (23)$$

なお E^{n_1, n_2} は

$$E^{n_1, n_2} = \frac{1}{2} \|\mathbf{y}^{n_1} - \mathbf{y}^{n_2}\|^2 \quad (24)$$

である。ただし $d(n_1, n_2) \neq d(n_2, n_1)$ なので、正確には擬距離である。なおM-stepはSOM、NGと同様に共通である。このアルゴリズムを本稿ではTFSOMと呼ぶ。

TFSOMのアルゴリズムを一言で言えば、「参照ベクトルを更新するたびに、ユニットを近い順にソートして、近傍関係を見直すSOM」である。マップ空間(=輪郭空間 $\mathcal{P}$)のトポロジーがわからないので、もはや各ユニットの位置ベクトル ξ^n は直接求めることはできない。しかし $\mathcal{P}$ 内における各ユニット間の距離 $d(n_1, n_2)$ が推定できればSOMのアルゴリズムは実行可能になる。そこで学習の各ステップごとに $d(n_1, n_2)$ を推定しなおすのがTFSOMのアルゴリズムである。

このようなTFSOMはSOMとNG双方の性質を併せ持つ。ユニット同士の近傍関係がデータと独立に決まる点はSOMと同じである。また各データについて勝者を決定し、勝者の近傍ユニットへ学習量を分配する点もSOMと同じである。SOMと唯一異なる点は、ユニット間距離 $d(n_1, n_2)$ が学習途中で逐次的に修正されることである。一方、ユニット間の関係が学習を通じて更新される点はNGと共通する。また距離が順位関数で与えられる点もNGと同じである。TFSOMがNGと唯一異なるのは、NGがデータ点を中心にして各ユニットへの順位を付けるのに対し、TFSOMでは勝者ユニットを中心にして各ユニットへの順位を付ける点だけである。通常データとその勝者は位置が近い。ため、NGとTFSOMの学習結果はほとんど変わらない。このようなTFSOMのアルゴリズムは上で述べた2つの条件(1)(2)のどちらも満たす。

TFSOMは単独でベクトル量子化アルゴリズムとして利用できる。しかしその結果はほとんどNGと変わらない。TFSOMは計算速度がやや速く、一方でNGは量子化誤差がわずかに少ないという違いがあるものの、実質的なレベルでの差異はほとんどない。

3.2 TFSOM×SOM

TFSOM×SOMの概念図を図4に示す。単独で用いた場合はTFSOMとNGの差異がないが、高階化したTFSOM×SOMはSOM²と同様にファイバーの概念が存在するため、形状空間法に適したアルゴリズムとなる。SOM²からTFSOM×SOMへの変更点は一点、1st SOMのE-step(11)で現れるユニット間距離 $d(n_1, n_2)$ を次式で置き換えるだけである。

図4 TFSOM×SOMのアーキテクチャ. 1st TFSOMは各形状表現を求め、2nd SOMは形状マップを自己組織化する。

$$d(n_1, n_2) = \frac{1}{M} \sum_{m=1}^M \text{RANK}(E^{mn_1, mn_2}; E^{mn_1, mn'}) \quad (25)$$

$$E^{mn_1, mn_2} = \frac{1}{2} \|\mathbf{z}^{mn_1} - \mathbf{z}^{mn_2}\|^2 \quad (26)$$

(25)はユニット間の順位をファイバーに沿って平均したものであり、いわばファイバー間の平均順位に相当する。このユニット間距離を用いることで、すべての1st TFSOMが共通のユニット間距離に従って学習することになる。

3.3 非均一な伸縮への対応

SOM, TFSOMはベクトル量子化としての性質を持つため、どのユニットもおおむね均等の割合で勝者になろうとする傾向がある(正確にはmagnification factorを考慮する必要がある)。しかし形状空間法においては、各ユニットがランドマーク点に相当するため、形状の伸縮に応じて勝者となる頻度が変わらなければならない。この問題も階層的多様体学習に特有の問題であり、従来の多様体学習では考える必要のない問題だった。

この問題は見かけのデータ数を補正することで対応できる。すなわち縮んだ領域は輪郭線のドット数が減って見えるが、実は縮んだことで隣のドットとくっついてしまったと考える。したがって1個のドットであっても複数のデータ点が存在すると考え、データ点1個あたりの学習量を増やす。逆に伸びた領域はドット数が増えるが、この場合はデータ点あたりの学習量を減らせばよい。この補正を行うためには、まず各ユ

ニットの勝率(勝者になった頻度)を求める。

$$p_i^n = \frac{\sum_j \delta(n_{ij}^*, n)}{J} \quad (27)$$

$$\bar{p}^n = \frac{\sum_i \sum_j \delta(n_{ij}^*, n)}{IJ} \quad (28)$$

ここで $\delta(\cdot, \cdot)$ はクロネッカーデルタである。 p_i^n は第*i*番1st TFSOMにおける第*n*ユニットの平均勝率であり、 $\bar{p}^n$ は全1st TFSOMを通算したときの第*n*ユニットの勝率である($\bar{p}^n$ は第*n*ファイバーの勝率と言っても良い)。そしてこれら2つの勝率比 $p_i^n / \bar{p}^n$ が伸縮比と考える。縮んだところでは学習量を増やせばよいので、伸縮比の逆数を学習量に乗じる。

$$\beta_{ij}^n = \frac{\bar{p}_{ij}^n}{p_i^n} \exp \left[-\frac{1}{2\lambda^2} d(n_{ij}^*, n) \right] \quad (29)$$

これが非均一伸縮に対応したTFSOM×SOMのE-stepになる。この補正もTFSOM×SOMにファイバーという概念が存在することで可能になった。TFSOM×SOMとNG×SOMは単なる性能の違いではなく、TFSOM×SOMは原理的に形状空間法の理に適っているのである。

3.4 局所的形状特徴量の活用

形状空間法では、異なる形状間の対応関係を推定することが重要なタスクになる。そのためには輪郭線を構成するドットの位置ベクトルだけでなく、曲率や色などの局所の特徴量を活用することでより正確に対応関係が得られる。ランドマーク点が表現するのは対応

するドット点の平均値であるため、局所平均が意味を持つ特徴量でなければならない。より正確に言えば、着目するドットを形状輪郭に沿って連続的に動かしたときに、連続的に変化する特徴量でなければならない。そうでないと、形状を多様体として表現できなくなる。一例として色の情報を付加する場合、各ドットの位置ベクトルにRGBベクトルを付加してデータベクトルとすることは好ましくない。なぜなら色は空間に対して不連続に変化するからである。またRGBベクトルの平均値も意味を持たない(青い領域と黄色い領域の中間にあるランドマーク点がシアン色になってしまう)。一方、着目するドットを中心とした局所近傍の色ヒストグラムは意味を持つ。なぜなら色ヒストグラムは連続的に変化するし、また平均の操作も意味を持つからである(青と黄色の境界にあるランドマークは青・黄色が50%のヒストグラムを表現する)。SIFTなどで用いる局所勾配ヒストグラムなどは形状表現に好ましい付加情報である。

3.5 TFSOM×SOMによる形状空間推定

図5はTFSOM×SOMの解くべきタスクをグラフィカルモデルで表したものである。このタスクは潜在変数 $\{\xi_{ij}\}$ 、 $\{\sigma_i\}$ および写像 $\{\mathbf{r}_i\}$ 、 $\mathbf{U}=(\mathbf{u}^m)$ 、ユニット間の距離行列 $\mathbf{D}=(d(n_1, n_2))$ を求める5重推定問題である。なお $\mathbf{D}$ の推定が形状の位相推定に相当する。

今、 $\{S_1, \dots, S_I\}$ を学習用形状集合とする。これらは

図5 形状空間法のグラフィカルモデル。観測データ $\{\mathbf{x}_{ij}\}$ から潜在変数 $\{\xi_{ij}\}$ 、 $\{\sigma_i\}$ 、写像 $\{\mathbf{r}_i\}$ 、 $\mathbf{U}$ および形状の位相(ランドマーク間の距離行列) $\mathbf{D}$ を推定する5重推定問題である。1st TFSOMは $\{\xi_{ij}\}$ と $\{\mathbf{r}_i\}$ の推定を行い、2nd SOMは $\{\sigma_i\}$ と $\mathbf{U}$ の推定を行う。

輪郭線でも良いし、手書き文字やシンボルマークのような線画でも良い。また色付きであってもかまわない。必ずしも2次元形状である必要もなく、3次元的な形状であっても良い。なお事前に大きさや回転等の補正はしておく。アフィン補正を組み込んだ形状空間法はYakushijiらがすでに提案しており、必要ならばそれを利用すればよい[10, 11]。

学習データの準備

最初に行うことは、学習用形状をドットに分解し、TFSOM×SOMに与える学習データ集合を作ることである。すなわち各形状をそれぞれ J 個のドットに分解し、 IJ 個のデータベクトル $\mathbf{x}_{ij}$ を作る。ここで $\mathbf{x}_{ij}$ はドットの位置ベクトルと、ドットを中心とする局所的な特徴量の連結ベクトル $\mathbf{x}_{ij} = (\mathbf{x}_{ij}^{position}, \mathbf{x}_{ij}^{localfeature})$ である。

TFSOM×SOMの初期化

次にTFSOM×SOMの各ユニットを初期化する。本研究ではTFSOM×SOMの学習能力を判断するため、学習が難しくなる乱数初期化を用いた。しかし現実の応用では、学習形状の中からもっとも典型的な形状をプロトタイプとして選び、まずそれを単一のTFSOMで学習する。そしてプロトタイプの学習結果を各1st TFSOMの初期値とする。こうすることでランドマークファイバーがプロトタイプに合わせた状態が初期状態になり、好ましくない局所解に収束する危険性を減らせる。なお本研究の結果では、乱数による初期化でも問題を生じなかったことを補足しておく。こうした初期化に関する扱いは従来のSOMと同様なのでここでは詳述しない。初期化が終了したら学習回数を $t=0$ とする。

TFSOM×SOMの学習

初期化が終わったら、以下の学習ステップを収束するまで繰り返す。

■Step 1：準備

各ループの開始にあたって2nd SOMにおける平均ユニット間距離行列 $\mathbf{D}$ を求める。

$$E^{mn_1, mn_2} = \frac{1}{2} \|\mathbf{z}^{mn_1} - \mathbf{z}^{mn_2}\|^2 \quad (30)$$

$$d(n_1, n_2) = \frac{1}{M} \sum_{m=1}^M \text{RANK}(E^{mn_1, mn_2}; E^{mn_1, mn'}) \quad (31)$$

また1st TFSOMと2nd SOMの近傍半径もあらかじめ計算しておく。

$$\lambda_1(t) = (\lambda_1^{\max} - \lambda_1^{\min}) \exp\left[-\frac{t}{\tau_1}\right] + \lambda_1^{\min} \quad (32)$$

$$\lambda_2(t) = (\lambda_2^{\max} - \lambda_2^{\min}) \exp\left[-\frac{t}{\tau_2}\right] + \lambda_2^{\min} \quad (33)$$

ここで(32)(33)は近傍関数を徐々に縮小するためのものであり、 τ^1, τ^2 は縮小の時定数である。さらに前回のループにおける各ランドマーク点の勝率を求める。

$$P_i^n = \frac{\sum_j \delta(n_{ij}^*, n)}{J} \quad (34)$$

$$\bar{P}^n = \frac{\sum_i \sum_j \delta(n_{ij}^*, n)}{IJ} \quad (35)$$

なお $t = 0$ のときはすべて等確率 $P_i^n = \bar{P}^n = 1/N$ にする。

■Step 2 : E-step (1st TFSOM)

次に1st TFSOMの学習を行う。1st TFSOMの目的は、各形状の形状表現ベクトル $\{\mathbf{r}_i\}$ を推定し、輪郭多様体 $\{Q_i\}$ を得ることである。まずE-stepでは潜在変数 $\{\xi_{ij}\}$ を推定する。

$$E_{ij}^n = \frac{1}{2} \|\mathbf{x}_{ij} - \mathbf{y}_i^n\|^2 \quad (36)$$

$$n_{ij}^* = \arg \min_n E_{ij}^n \quad (37)$$

$$\beta_{ij}^n = \frac{\bar{P}^{n_{ij}}}{P_i^{n_{ij}}} \exp\left[-\frac{1}{2\lambda_1^2(t)} d(n_{ij}^*, n)\right] \quad (38)$$

$$B_i^n = \sum_{j=1}^J \beta_{ij}^n \quad (39)$$

ここで $\xi_{ij}^{n_{ij}^*}$ が ξ_{ij} の最尤推定点である (正確には近傍関数の広がりを持つ事後分布が得られる)。

■Step 3 : M-step (1st TFSOM)

1st TFSOMのM-stepでは形状表現ベクトル $\mathbf{r}_i$ を推定する。

$$\mathbf{y}_i^n := (1 - \varepsilon) \mathbf{y}_i^n + \varepsilon \frac{1}{B_i^n} \sum_j \beta_{ij}^n \mathbf{x}_{ij} \quad (40)$$

$$\mathbf{r}_i = \bigoplus_{n=1}^N \mathbf{y}_i^n \quad (41)$$

■Step 4 : E-step (2nd SOM)

次に2nd SOMの学習を行う。2nd SOMの目的は、形状ベクトル集合から形状多様体 $\mathcal{M}$ を推定することである。まずE-stepでは潜在変数 $\{\sigma_i\}$ を推定する。

$$E_i^m = \frac{1}{2} \|\mathbf{r}_i - \mathbf{u}^m\|^2 \quad (42)$$

$$m_i^* = \arg \min_m E_i^m \quad (43)$$

$$\alpha_i^m = \exp\left[-\frac{1}{2\lambda_2^2(t)} d^2(m_i^*, m)\right] \quad (44)$$

$$A^m = \sum_i \alpha_i^m \quad (45)$$

ここで σm_i^* が σ_i の最尤推定点である (正確には近傍関数の広がりを持つ事後分布が得られる)。

■Step 5 : M-step (2nd SOM)

M-stepでは形状多様体 $\mathcal{M}$ を求める。

$$\mathbf{u}^m := \frac{1}{A_i^m} \sum_{i=1}^I \alpha_i^m \mathbf{r}_i \quad (46)$$

$\mathbf{U} = (\mathbf{u}^m) = (\mathbf{z}^{mn})$ は 3 次テンソルであり、 $(\mathbf{u}^m)$ が形状多様体 $\mathcal{M}$ 、 $(\mathbf{z}^{mn})$ が形状ファイバー束 $\mathcal{F}$ を表す。

■Step 6 : Copy Back (2nd to 1st SOM)

最後に2ndSOMから1stTFSOMへフィードバックし、Step1に戻る。

$$\mathbf{r}_i := \mathbf{u}^{m_i^*} \quad (47)$$

$$t := t + 1 \quad (48)$$

以上がTFSOM×SOMによる形状空間推定のアルゴリズムである。

3.6 新規形状から形状空間への射影推定

形状空間推定が終わった状態で、新規の形状 S_{new} が与えられたとする。このとき知りたいのは次の2点である。(1) S_{new} にもっとも近い形状空間の点はどこか。すなわち S_{new} から形状多様体 $\mathcal{M}$ への射影点 $\tilde{S}_{\text{new}}$ はどこになるのか。(2) S_{new} と形状多様体 $\mathcal{M}$ との距離はどれくらいか。すなわち S_{new} と射影点 $\tilde{S}_{\text{new}}$ との形状表現空間における誤差はどれくらいか。

これらを求めるには以下のようにする。まず新たに1st TFSOM $\mathbf{r}_{\text{new}}$ を用意する。 $\mathbf{r}_{\text{new}}$ の初期化は S の中心 (2nd SOMのマッパ中央) にある形状で初期化する (あるいは量子化誤差 $\sum_{j=1}^J \|\mathbf{x}_j^{\text{new}} - \mathbf{y}_j^m\|^2$ を最小化する $\mathbf{u}^m$ で初期化しても良い)。そしてTFSOM×SOMの学習ステップのうち、Step1, 2, 3, 4, 6を $\mathbf{r}_{\text{new}}$ に関してのみ繰り返す。 $\mathbf{r}_{\text{new}}$ の学習が終了した時点での勝者を m_{new}^* とすれば、 $\mathbf{u}^{m_{\text{new}}^*}$ が射影形状 $\tilde{S}_{\text{new}}$ に該当する。また誤差

$$E_{\text{new}}^* = \frac{1}{2} \|\mathbf{r}_{\text{new}} - \mathbf{u}^{m_{\text{new}}^*}\|^2 \quad (49)$$

が S_{new} と $\mathcal{M}$ の距離の二乗になる。

この結果を利用すれば形状空間を利用したパターン識別が可能になる。形状空間法をパターン識別に応用する場合、大きく2つの異なるケースがある。第一のケースはクラスごとに形状空間が対応する場合であり、第二のケースはクラス全体でひとつの形状空間を共有する場合である。

第一のケースの典型例は手書き文字識別である。手書き数字であれば‘0’から‘9’に対応する10個の形状空間が存在することになるので、TFSOM×SOMも10セット用意する必要がある。今、識別したい形状クラスが K 個存在し、第 k 形状空間を $S^{(k)}$ とする。 K 個の形状空間は学習用形状データを用いてTFSOM×SOMで学習を行っておく。また新規形状の識別に際しては、1st TFSOMも $\{\mathbf{r}_{\text{new}}^{(1)}, \dots, \mathbf{r}_{\text{new}}^{(K)}\}$ の K 個用意する。そして上述の方法で射影形状 $\tilde{S}_{\text{new}}^{(k)}$ および誤差 $E_{\text{new}}^{*(k)}$ を計算し、誤差 $E_{\text{new}}^{*(k)}$ のもっとも小さいクラス

$$k_{\text{new}}^* = \arg \min_k E_{\text{new}}^{*(k)} \quad (50)$$

を識別結果とする。

第二のケースの典型例は手書き文字の筆者識別である。たとえば手書き文字の‘A’から筆者を識別する場合は、‘A’の形状空間がひとつあれば良い。そして各筆者の文字が形状空間のどこに該当するかを学習用データで調べておく。新規の文字‘A’に対しては1st TFSOMを1個用意し、 S の中央で初期化する(平均的な筆跡で初期化する)。そして上述の方法で射影点を求めて該当する筆者を推定する。このように、何を識別したいかによって形状空間の利用法が異なってくる。

なお現実の課題に形状空間法を適用する場合は、計算コストも考える必要がある。形状空間法は計算コストが高いので、第一のケースで識別クラス数が増えたと膨大な計算量が必要になる。一方で第二のケースでは求めるべき形状空間が1個だけなので、そこまで膨大な計算コストにはならない。

上記の方法はTFSOM×SOMのみで完結した識別法であるが、計算コストも下げ、かつ識別性能を上げたいのであれば、サポートベクターマシン(SVM)等の他手法と組み合わせることが有効である。特に学習用の形状データ数が少ない場合は、形状空間法で補間することで学習用データを増やすことができる。今、ある識別タスクに有効な特徴量がわかっているとすると、ふつうはこの特徴量を用いてSVM等を学習させればよいが、学習データ数が少ない場合は必ずしも十分な性能を引き出せるとは限らない。そこでまずTFSOM×SOMで形状間の補間を行い、得られた中間的な形状も特徴量に変換して学習用データとして用いるのであ

る。こうすることで識別時の計算コストを上げることなく形状空間法のメリットを利用できる。ここで大事なことは、タスク依存の特徴量空間で補間するのではなく、形状表現空間で補間を行うという点である。特徴空間での補間は必ずしも中間的な形状に対応しないので、補間した特徴量が学習データに適切かどうかの保証はない。しかし形状表現空間で補間すれば確実に中間的な形状が得られるので、この中間形状から特徴量を得れば良いわけである。

4. 実験結果

4.1 予備実験：NG×SOMとTFSOM×SOMの比較

本提案法であるTFSOM×SOMと、単にSOM²の1st SOMをNGに置き換えただけのNG×SOMを比較するため、図6(a)の学習形状を与えて形状空間推定を行った。与えた形状は2本の線分により構成されており、それぞれ長短が異なる。形状1つあたりのドット数は400点とし、これを100点のランドマーク点で表現させた。

図6(b)(c)はNG×SOMとTFSOM×SOMが推定した形状空間である。この実験では1次元の形状空間を用いており、1次元形状空間に沿って形状が連続的に表現する様子が表されている。また図6(d)(e)はランドマークファイバーを示したもので、形状変化に伴ってランドマーク点がどのように移動するかを示す。TFSOM×SOMの場合は線分が伸縮する形状変化が得られており、一方NG×SOMの場合は線分の両端が移動する形状変化が得られた。したがってTFSOM×SOMの方が好ましい結果が得られた。TFSOM×SOMが適切な形状空間を推定できた理由は、ファイバーという概念を利用して形状トポロジーの統一化および非均一伸縮への対応が可能になったためである。これよりTFSOM×SOMは期待通りの動作を示したと言える。

4.2 局所特徴量の活用

次に手書き数字を用いて形状空間推定を行った。この実験の目的は、局所特徴量を追加することで形状空間推定が改善されるかどうかを調べることである。

今回用いた局所特徴量は局所加重方向ヒストグラムである。着目しているドット点を中心に、近傍の他のドットがどの方向に分布するかを調べることで、着目点を通る線分方向のヒストグラムを求めるものである。具体的には図7に示した角度フィルタを用い、グレーの領域内におけるドットが占める割合を求め、12次元のヒストグラムベクトルとした。この特徴量は空間的連続性を持つため、輪郭多様体の連続性は保持される。

図6 NG×SOMとTFSOM×SOMの形状空間推定の比較. (a) 学習用形状集合 (b) NG×SOMによる形状空間 (c) TFSOM×SOMによる形状空間 (d) NG×SOMの表現するファイバー束 (e) TFSOM×SOMの表現するファイバー束. NG×SOMでは▲印のところでランドマーク点の好ましくない移動が起きている.

図7 局所特徴量を得るための角度フィルタ

図8 (a)が学習用に用いた‘8’の形状である. これらの文字は $[-1, +1]^2$ の正方形をbounding boxとして切り出し, ピクセル点の座標を求めた. したがってピクセル点の位置 $\mathbf{x}_{ij}^{\text{position}}$ は各成分が $[-1, +1]$ の値を取る2次元ベクトルであり, 局所特徴量 $\mathbf{x}_{ij}^{\text{local feature}}$ は各成分が $[0, 1]$ の値を取る12次元ベクトルである. 両者を統合するには適切な重みづけが必要であるが, 今回は次元数で割って

$$\mathbf{x}_{ij} = \frac{1}{2}\mathbf{x}_{ij}^{\text{position}} \oplus \frac{1}{12}\mathbf{x}_{ij}^{\text{local feature}} \quad (51)$$

とした. また抽出したピクセル点は400点, ランドマーク点は100とした.

TFSOM×SOMの目的は, これらの手書き数字を自然なモーフィングとして補間し, 形状空間を推定する

ことである。自然なモーフィングが表現できたかどうかは、ランドマークファイバーにねじれなどが生じたかどうかで確認できる。

図 8 (b) (c) が学習結果の結果得られた形状空間である。またそれぞれの図における枠内のファイバー束を表示したものが図 8 (d) (e) である。どちらのマップも ‘8’ の形状が連続的に変化する形状空間が得られた。位置情報のみの場合 (図 8 (b) (d)) でもおおむね

良好な結果ではあったが、一部でランドマークファイバーのねじれが見られた (図 8 (b) の▲)，一方で局所特微量を用いた結果 (図 8 (c) (e)) はファイバーのねじれが見られなかった。初期値をランダムに変えてファイバーのねじれの発生率を調べたところ、50回の試行に対して局所特微量なしでは9回、局所特微量ありでは2回であった。このことから、局所特微量を利用することでランドマークファイバーがより適切に推定で

図 8 TFSOM × SOM による手書き数字 ‘8’ の形状空間. (a) 学習用形状データ. (b) 局所特微量を使用しない場合の形状マップ. ▲の個所でファイバーの乱れがあった. (c) 局所特微量を使用した際の形状マップ. 図中の枠内のファイバーを示したものが (d) (e) である. 丸印で囲った領域でファイバーの乱れがあった.

きることが示された。

なおねじれの発生率は形状の複雑さや学習用データの個数にも依存する。図8(a)は崩れた形状の数字を含んでおり、敢えて学習が困難な状況で実験を行った。崩れの少ない手書き数字では、局所特徴量の有無にかかわらずファイバーのねじれが発生しなかった。しかし複雑な形状では困難度も上がるため、そのような場合は局所特徴量の有効性が高まると考える。

4.3 手書き数字識別への応用

形状空間法の応用例として、TFSOM×SOMを用いた手書き数字識別を行った。実験では数字0～9に対応する10個のTFSOM×SOMを用いて手書き数字を学習させ、それぞれの数字に対応する形状空間を推定した。その後、学習に用いなかったテスト用手書き数字を用いて識別率を評価した。

実験で用いた手書き数字データはNational Institute

図9 TFSOM×SOMによる手書き数字‘4’の形状空間。学習に用いた文字数を1文字あたり10, 100, 1000個と変えた。赤は学習データに対する勝者である。

図10 手書き数字の識別率。識別率は10回の試行の平均値であり、標準誤差をエラーバーで示した。

of Standards and Technology (NIST) の Handprinted Form sand Characters Database (HFCD) である [28]. 機械学習のベンチマークでは通常、解像度を落とし、事前処理をすませた MNIST データベースがよく用いられる。しかし本実験ではドットの座標をデータとして用いるため、解像度を落とさないオリジナルデータを用いた。事前処理として、MNIST 同様に文字領域を切り出し、サイズを $[-1, +1]^2$ に規格化したうえでドットの座標を求めた。また局所特徴量として上で述べた角度フィルタによる局所加重方向ヒストグラムを付加した。位置と局所特徴量の重みは (51) を用いた。

学習に用いるデータ数は、1 数字クラスあたり 10 個から 1,000 個の間で変化させた (総計 100 ~ 10, 000 文字)。これは 10 人分から 1,000 人分の手書き数字を学習させたことに相当する。TFSOM $\times$ SOM のマップサイズは 8×8 とした。したがって 1 数字クラスあたりの学習文字数が 64 未満の場合は補間、64 以上の場合は量子化となる。学習終了後、学習データとは異なる 1,000 人分の手書き数字 (総計 10,000 文字) で識別率を調べた。各手書き数字に対して、0 ~ 9 に対応する形状空間への射影を取り、射影点との誤差が最小となるクラスを識別結果とした。

また比較対象として、画像を直接ベクトルデータ化した場合の識別率も調べた。まず NIST の手書き文字画像に対して MNIST と同様の前処理を行った後、 30×30 ピクセルまで解像度を落として 900 次元のデータベクトルを作った。識別方法としては最近傍法 (Nearest Neighbor 法: NN 法)、サポートベクターマシン (SVM、ガウスクERNEL を使用)、通常の SOM を用いた最近傍法である (図 3 (b) 参照)。

これらの実験は学習用・テスト用データを入れ替えて 10 回繰り返し、識別率の平均と標準誤差を求めた。

図 9 は TFSOM $\times$ SOM が推定した手書き数字 '4' の学習結果である。学習データ数に関わらず '4' の傾きが連続的に変化するマップが得られた。このことは、学習データ数が少なくても形状空間を効率よく推定できることを示唆する。他の数字についても同様の結果が得られた。

図 10 は識別結果である。TFSOM $\times$ SOM による形状空間法の識別率が抜き出て高かった。特に学習データ数が少ない場合での識別率の高さは顕著であり、各数字クラスに 10 個ずつしか学習データを与えていない場合でも 95% という識別率を示した。これは形状空間法が学習用形状データを自然に補間し、実質的に識別用テンプレートを増やしたことが理由である。

一方、比較対象であるピクセルデータを用いたデータ表現方法では、NN 法、SVM、SOM のどれも似た

結果を示した。すなわち SVM や NN 法という識別器の問題ではなく、データ表現方法が識別率を決めたと考えられる。なおデータ数が少ないところで SOM が NN や SVM より性能が良いのは、データ間を補間したためと考えられる。

ここで着目すべき点は TFSOM $\times$ SOM と SOM の差である。どちらも SOM を用いてデータ分布の多様体を推定し、データ間を補間するという点においては同じである。両者の違いは TFSOM $\times$ SOM が形状表現空間内の多様体を推定するのに対し、SOM はピクセルデータ空間内の多様体を推定する点にある。そもそもデータ間を補間するためには、データ同士の線形内分点の意味を持つような適切なデータ表現が成されていることが前提である。TFSOM $\times$ SOM と SOM の性能の差異はこの点が現れたものであり、本手法の意義を理解する上で重要な点である。

4.4 フォント形状空間

TFSOM $\times$ SOM を用いた形状空間法の特徴として、線画に特化したアルゴリズムを組み込んでいない点がある。すなわち本手法はより広範囲の図形に対応できるもので、ランドマーク点が定義できるような対象であれば線画以外にも応用できる。特に図と地から構成される画像に対しては広く適用が可能である。すなわち図の部分がデータ集合 (ランドマーク集合) に該当し、地の部分がデータ点の不在箇所に相当する。あるいは図と地の境界をデータ点としてもよい。したがって形状を輪郭多様体で表現する方法を包含するより一般的な手法となっている。

本手法が輪郭形状以外にも応用できることを示すため、文字の書体形状の空間推定を行った。図 11 は「永」の 13 種の字体から推定したフォント形状空間である。学習の結果、肉厚の細いフォントから太いフォントへの連続変化や、明朝体・ゴシック体のような標準的な字体からデフォルメされた字体への連続変化が表現できた。

このフォント形状空間に用いたアルゴリズムは、手書き文字識別に用いたアルゴリズムと同一であり、何も変更していない (ただし複雑な形状を表現するため、1 形状あたりのランドマーク数は増やした)。また図 11 (c) はランドマークファイバーを示したもののだが、数字よりも複雑な形状であっても適切にランドマークファイバーが形成された。

(a)

(b)

(c)

図 11 「永」のフォント形状空間. (a) 学習用形状データ. 明朝体, ゴシック体など 13 種の字体を用いた. (b) 推定した形状空間. 明朝体系が左上, ゴシック系が左下, デフォルメされた字体が右側に配置された. 赤で示した箇所が学習データの勝者である. マップ中の枠内のファイバーを示したものが (c) である.

5. 議論

本論文では, 形状空間法の理論的枠組みを再構築し, そしてTFSOM×SOMというアルゴリズムとして実装した. 本節では本研究の意義について検討する.

5.1 形状空間理論としての意義

理論面における本論文の意義は, 形状の集合を取り扱う理論的な枠組みを構築したという点である. 図 2 がその全体像であるが, これは形状の集合を扱うための一般的な枠組みとなっている. したがって形状集合を扱うアルゴリズムはなんらかの形でこの理論体系の下で理解できる. すなわち本論文は, 既存の土俵の上でより性能の良いアルゴリズムを提案するものではなく, より良い研究を展開するための土俵づくりを行ったものである.

本論文の先駆となるのがKendallの形状空間論である[1, 2, 3]. Kendallの理論と比較して本理論が進展した点がいくつかある. まずKendallはリーマン幾何学に基づき形状多様体上の空間を定式化した. 一方われわれは形状の本質を表す固有空間(潜在変数空間)が存在するという立場に立ち, 観測される形状と形状に内在する本質を切り離れた. 第二にKendallは形状集合のつくる空間について議論しているのに対し, われわれは確率的生成モデルに基づいてベイズ推論の枠組みに帰着させた(図 5). これにより工学的な取り扱いが明確化され, また何を推定すべきかという目標や, 推定結果をどう評価するかという基準が明確化された. すなわち数学から工学の問題へ転換することができた. その実装例がTFSOM×SOMである. ここで強調したいのは, TFSOM×SOMはあくまで形状空間法の一実装例に過ぎないことである. SOMを使わず, 図 5 を元にオーソドックスなベイズ推論で形状空間を求めることも可能である.

なお本論文で構築した理論は, Yakushijiの確率的形状生成モデルを発展させたものである[10, 11]. Yakushijiらは観測時に形状が受ける(見かけ上の)変換に対する取り扱いも理論に組み込み, アルゴリズム導出を行った. 観測変換についてはYakushijiらの論文で十分議論されているとして本論文では扱わなかった. しかしKendallの形状理論においても実際の応用に際しても, 観測変換の推定と除去は重要な論点であることは指摘しておく.

5.2 TFSOM×SOMのアルゴリズムとしての意義

本論文では形状空間法のアルゴリズム実装としてTFSOM×SOMを提案した. TFSOM×SOMの第一の

意義は、形状空間法が計算機的に実装可能なもので、理論とアルゴリズムを結び付けられることを示した点である。特に形状表現空間における多様体推定のアルゴリズム実装の基本形を提示するものであると考える。

本論文ではアルゴリズムの基本要素としてSOMを利用したが、計算コストを問題にしなければオーソドックスな階層ベイズ法も利用できる。たとえばSOMの代わりに生成位相マップ(Generative Topographic Mapping: GTM)を用いれば、よりベイズ的アルゴリズムになる[29]。ただし5重推定問題という難易度の高い問題であるため、単純に階層ベイズ法を用いれば解けるかどうかはわからない。なおSOMのベイズ的理解は多くの論文でなされており、SOMをベイズ的手法の一種として用いても問題はない[30, 31]。したがってTFSOM×SOMも近似的に階層ベイズ法の一種として考えることができる。このあたりを理論的に明確化することは今後の課題のひとつである。

TFSOM×SOMの第二の意義は、形状のように捉えどころのないものを操作する良い空間を提供できるという点である。ここで言う「良い空間」とは、形状間の局所線形和を取ったり、距離を測ったりできる空間である。またこの空間はもとの形状の情報を失わないため、演算をして得られた形状を具体的に表示することも容易である。

古典的な手法では、高次元の観測データベクトル(たとえば画像データ)から直接低次元の特徴ベクトルに変換することが多かった。これはメモリや計算コストの制約からも必然的な要請であるが、どんな特徴量を選ぶかはタスクに依存する。また特徴ベクトル同士の演算は一般に形状同士の自然な演算にならないし、特徴ベクトルから形状に逆変換することもできない。したがって形状演算の可能な空間へ可逆的に変換する本手法と、タスクを実行するための(低次元)特徴ベクトル空間へ非可逆的に変換することは分けて考える必要がある。実際、両者は必要に応じて併用するものであり、対立するものではない。本論文で提案したTFSOM×SOMは前者の手法であり、特徴ベクトルを用いる従来法と対立するものではない。実利的な観点で言えば、本手法は「汎用的な“前処理用空間”を提供するもの」であると言えるだろう。

TFSOM×SOMの第三の意義は、より広範囲な形状について扱えるということである。従来の閉多様体やフーリエ変換による形状表現法では連続した物体の輪郭線しか扱えなかったが、本手法では連続していない形状(複数の分かれたパーツから構成された形状)でも扱うことができる。また曲線に限らず、太さを持つフォント形状でも扱うことができる。より正確に言え

ば、ランドマーク点の集合として表現可能なものであれば基本的にはどんなものでも扱える。一般には図・地から構成される図形ならば扱える。また本手法は2次元図形に限らないので、顔の3次元形状から表情の表現空間を得ることもできる。

5.3 応用面での意義

本手法は、形状という数値的に扱いにくい対象を処理する一般的な枠組みと手法を提供するものである。たとえば図11はフォントというデザインを扱った例とも言える。このほかにも車のフォルムや翼の形状分析など、工業デザインへの応用も期待できる[32]。また書家の筆跡分析などアート面への展開も考えられる。

デザイン以外の分野では、生物の形状解析、特に進化的な身体形状変化の解析などの応用も考えられる。

5.4 新たな発展の基盤としての意義

本研究は形状空間の理論にとどまらず、トポロジーというより抽象化された形状概念へ発展する基盤を与える。なぜならばTFSOM×SOMのアルゴリズムでは、形状の位相的構造を距離行列 $\mathbf{D}$ として推定するからである。

本論文のアルゴリズムでは、距離行列 $\mathbf{D} = d(\xi^{n_1}, \xi^{n_2})$ をすべての1st SOMで統一した。すなわち形状のトポロジーは一定であると仮定した。しかし距離行列も σ によって連続的に変化するようにアルゴリズムを修正することは容易である。すなわち(30)(31)を次式のように修正する。まず1st TFSOMのE-step終了後に

$$E_i^{n_1, n_2} = \frac{1}{2} \|\mathbf{y}_i^{n_1} - \mathbf{y}_i^{n_2}\| \quad (52)$$

$$D_i^{n_1, n_2} \equiv d_i(n_1, n_2) = \text{RANK}(E_i^{n_1, n_2}; E_i^{n_1, n'}) \quad (53)$$

として各形状ごとに距離行列 $\mathbf{D}_i$ を求め、続く2nd SOMのM-stepでは

$$\mathbf{D}^m = \frac{1}{A^m} \sum_{i=1}^I d_i^m \mathbf{D}_i \quad (54)$$

として距離行列が滑らかに変化するようにする。そして最後にフィードバック

$$\mathbf{D}_i := \mathbf{D}^{m_i} \quad (55)$$

とする。この修正により形状トポロジーも連続的に変化するような状況でも対応が可能になる。たとえば楷書から行書、草書といったトポロジーの変化を伴う形状変化が表現できるのではないかと考えている。

またこの考えをさらに推し進めれば、距離行列 $\mathbf{D}_i$ 自体がより抽象化した形状の情報を表現したデータベク

トルとみなすことができる。すなわちトポロジカルな意味での形状構造集合をクラスタリングしたり空間推定したりすることが可能になると思われる。したがって本手法は形状構造というより大きな枠組みを扱うための最初の基盤を提供するものであると考えている。

一方で形状という概念をより一般化すれば、力学系のアトラクタ形状や関数の形状なども抽象化された形状とみなすことができ、形状空間論を応用することができる[33]。こうした抽象化された形状の取り扱いには転移学習やメタ学習への発展基盤とも密接なつながりがあるため、知能に関する重要な枠組みを提供するものであると考える。

6. まとめ

本論文で提案した形状空間法は、きわめて高い汎用性を持つことが特長だった。本手法はフォルムデザインやスタイル分析など工学的に扱いが難しかった領域への貢献が期待できるものである。

本論文ではまた、形状空間法の理論的な再構築にも取り組んだ。理論の再構築だけでは新しいアルゴリズムの開発につながらないが、形状のトポロジーという抽象化された概念の工学的取扱いや、転移学習・メタ学習への発展などにつながる研究領域を創る上で重要な役割を果たす。

こうした理論的側面としては、ホモトピーやファイバー束といった高次の多様体学習の側面と、階層ベイズ法による多重推定問題という統計的機械学習の側面があり、これらを融合することに大きな意義があると考えている。したがって形状を取り扱うアルゴリズム開発が目的というよりは、形状空間を典型例とする機械学習の普遍的問題を扱うことが本研究の真の目的であり、今後引き続き取り組む考えである。

謝辞

本研究は科研費基盤C(課題番号23500280)の支援を受けて実施した。

参 考 文 献

- [1] D.G. Kendall, "The diffusion of shape", *Advances in Applied Probability*, 9, pp.428-430, 1977.
- [2] D.G. Kendall, "Shape manifolds, Procrustean metrics, and complex projective spaces", *Bulletin of the London Mathematical Society*, 16, pp.81-121, 1984.
- [3] D.G. Kendall, D. Bardon, T.K. Carne, H. Le, "Shape and shape theory", John Wiley & Sons, 1999.
- [4] I.L. Dryden, "The statistical analysis of shape data", PhD thesis, University of Leeds, 1989.
- [5] I.L. Dryden, K.V. Mardia, "Statistical shape analysis", John Wiley & Sons, 1998.
- [6] Y. Rath, S. Dambreville, A. Tannenbaum, "Statistical shape analysis using kernel PCA", In: *SPIE, Electronic Imaging*, 2006.
- [7] T. Kohonen, "The Self-Organizing maps (3rd ed.)", Berlin: Springer-Verlag, 2001.
- [8] T. Furukawa, "SOM of SOMs", *Neural Networks*, 22, 463-478, 2009.
- [9] T. Furukawa, "SOM of SOMs: An extension of Map to Homotopy", *Lecture Notes in Computer Science*, 4232, 950-957, 2006.
- [10] S. Yakushiji, T. Furukawa, "Shape space estimation by SOM²", *Lecture Notes in Computer Science*, 7063, 618-627, 2011.
- [11] S. Yakushiji, T. Furukawa, "Shape space estimation by higher-rank of SOM", *Neural Computing & Applications*, 22, 1-11, 2012.
- [12] 薬師寺翔, 古川徹生, "高階化 SOM による形状表現マップ", *知能と情報*, 24, 648-659, 2012.
- [13] S. Loncaric, "A Survey of Shape Analysis Techniques", *Pattern Recognition*, Vol.31, 983-1001, 1998.
- [14] D. Zhang, G. Lu, "Review of Shape Representation and Description Techniques", *Pattern Recognition*, 37, 1-19, 2004.
- [15] I.T. Young, J.E. Walker, J.E. Bowie, "An analysis technique for biological shape. I", *Comput. Graphics Image Processing*, Vol.25, 357-370, 1974.
- [16] M. Peura, J. Iivarient, "Efficiency of simple shape descriptors", *Proceedings of the Third International Workshop on Visual Form*, 443-451, 1997.
- [17] J. Hopkins, T. Andersen, "A Fourier descriptor based character recognition engine implemented under the camera open-source document processing framework", *Proceedings of the Society of Photo-Optical Instrumentation Engineers*, 5676, 111-118, 2005.
- [18] J. Ohm, F. Bunjamin, W. Liebsch, B. Makai, K. Muller, A. Smolic, D. Zier, "A set of visual feature descriptors and their combination I a Low-Level Description Scheme", *Signal Processing-Image Communication*, Vol.16, No.1-2, pp.156-179, 2000.
- [19] S. Kumar, Gopinath S, S. Kumar, R. Chhikara, "Use of Splines in Handwritten Character Recognition", *International Journal on Computer Science and Engineering*, Vol.02, No.07, pp.2421-2426, 2010.
- [20] S.K. Pauri, A. Datta, T. Pal, "Shape Approximation of Arc Patterns using Dynamics Neural Networks", *Signal Processing*, Vol.42, pp.221-225, 1995.
- [21] A. Goshtasby, "Description and Discrimination of Planar Shapes Using Shape Matrices", *IEEE Trans Pattern Analysis and Machine Intelligence*, Vol.7, No.6, 7380-743, 1985.
- [22] A. Datta, T. Pal, S.K. Parui, "A Modified Self-Organizing Neural Net for Shape Extraction", *Neuro-computing*, Vol.14, pp.3-14, 1997.
- [23] A. Datta, S.K. Parui, "Skeletons from Dot Patterns: A Neural Network Approach", *Pattern Recognition Letters*, 18, pp.335-342, 1997.
- [24] G.S. Kumar, P.K. Kalra, S.G. Dhande, "Curve and Surface Reconstruction from Points: an Approach based on SOM", *Applied Soft Computing*, Vol.5, pp.55-66,

- 2004.
- [25] T.M. Martinetz, S.G. Berkovich, K.J. Schulte, "Neural Gas" Network for Vector Quantization and its Application to Time-Series Prediction", IEEE TRANSACTIONS OF NEURAL NETWORKS, Vol.4, No.4, pp.558-567, 1993.
 - [26] T. Martinetz, K. Schulten, "Topology Representing Network, Neural Networks", Vol.7, No.3, pp.507-522, 1994.
 - [27] D. Deng, N. Kasabov, "On-line Pattern Analysis by Evolving Self-Organizing Maps", Neurocomputing, 87, pp.87-103, 2003.
 - [28] P.J. Grother, "NIST special database 19", Handprinted forms and characters database, National Institute of Standards and Technology, 1995.
 - [29] C.M. Bishop, M. Svensen, C.K.I. Williams, "The generative topographic mapping", Neural Computation, 10, pp.215-234, 1998.
 - [30] T. Heskes, "Self-organizing maps, vector quantization, and mixture modeling", IEEE Trans. Neural Networks, 12, pp.1299-1305, 2001.
 - [31] J.J. Verbeek, N. Vlassis, B.J.A. Kröse, "Self-organizing mixture models", Neurocomputing, 63, pp.99-123, 2005.
 - [32] A. Oyama, S. Obayashi, K. Nakahashi, N. Hirose, "Aerodynamic Wing Optimization via Evolutionary Algorithms based on structured coding", CFD Journal, 8, pp.570-577, 2000.
 - [33] T. Ohkubo, T. Furukawa, K. Tokunaga, "Requirements for the Learning of Multiple Dynamics", Advances in Self-Organizing Maps, Lecture Notes in Computer Science, 6731, pp.101-110, 2011.
- (2012年9月12日 受付)
(2013年1月24日 採録)
- 【問い合わせ先】
〒808-0196 北九州市若松区ひびきの2-4
九州工業大学大学院生命体工学研究科
古川 徹生
TEL : 093-695-6124
FAX : 093-695-6134
E-mail : furukawa@brain.kyutech.ac.jp

著者紹介

おおたに まこと
大谷 誠 [非会員]

九州工業大学大学院生命体工学研究科にて脳情報工学を学び、2007年に脳情報専攻博士前期課程修了、2011年に後期課程単位取得退学。同年4月より同大学・大学院生命体工学研究科研究員。現在は、自己組織化マップに関する研究に従事。

ふるかわ てつお
古川 徹生 [正会員]

大阪大学基礎工学部にて生物工学を学び、1989年博士前期課程修了。同年、九州工業大学情報工学部助手。1998年博士(工学)。2004年より同大学・大学院生命体工学研究科助教授、2006年より教授。現在は位相保存画像や積多様体の学習理論、およびそれらを用いた知的情報処理の研究に取り組んでいる。電子情報通信学会会員、日本神経回路学会会員。

Self-Organization of Shape Space by the Higher-Rank SOM - Topology-Free Shape Space Method -

by

Makoto OTANI and Tetsuo FURUKAW

Abstract :

The aims of this paper are to establish the shape space theory which deals with a set of shapes, and to develop a shape space estimation method which does not require the prior knowledge about the topology of shapes. To achieve these aims, first we developed a Self-Organizing Map (SOM) that is free from topological restriction, namely, Topology-Free SOM (TFSOM). Then we built TFSOM into the higher-rank SOM, which is called TFSOM×SOM in this paper. TFSOM×SOM was applied to handwritten digit recognition task, and it showed about 95% accuracy in recognizing digits written by 1000 people, after the TFSOM×SOM was trained by the digits of only 10 people. Since the proposal method is not specialized to line-drawing, it can be applied to wider tasks relevant to shapes.

Keywords : self-organizing map, higher-rank SOM, neural gas, shape space method, shape representation

Contact Address : Tetsuo FURUKAW

*Graduate School of Life Science and Systems Engineering, Kyushu Institute of Technology
2-4, Hibikino, Wakamatsuku, Kitakyushu, 808-0196, JAPAN*

TEL : 093-695-6124

FAX : 093-695-6134

E-mail : furukawa@brain.kyutech.ac.jp