

解 説

自己増殖型ニューラルネットワークと教師無し分類学習

久保田直行*・里見 将志*

1. はじめに

近年, インターネットの大規模な普及により, 多くのデータを瞬時に収集することができるようになった. データから価値のある情報を抽出するために, データマイニングや多次元データの可視化に関する研究が盛んに行われており, 最もよく使われる手法の一つに自己組織化マップ (Self-Organizing Map; SOM) がある. SOM は, 多次元空間上の参照ベクトルを用いて, データの分布にあわせて, 低次元空間へのトポロジカルな写像 (topology preserving mapping, あるいは, topographic mapping) を行えることが知られている [1-4]. また, SOM のような手法は, 与えられたデータのみを用いて参照ベクトルの学習を行うため, 教師無し学習 (unsupervised learning) とよばれ, winner-take-all による競合学習 (competitive learning) を行う. さらに, クラスタリングを行うことを目的に用いられる場合, 教師無し分類 (unsupervised classification) の一つとしてみなされる. SOM は, ノード数や近傍構造を決定すれば, 比較的, 容易に様々な問題に適用できる反面, 未知なデータに対し, ノード数や近傍構造の設計を行うことが困難であるため, ノードの追加削除を行う方法なども提案されている [5]. 一方, 教師無し分類を行う代表的な方法として, k-means [6,7] アルゴリズムなどがあるが, クラスタ数の事前設計をどのように行うかが問題となっており, また, データの分布によっては, クラスタリングをうまく行えない場合があるなどの問題が指摘されている. このように教師無し分類を行う場合, 事前に必要なノード数をどのように設計するかが一つの課題となる. このような問題を解決するために, 最小のノード数から分類学習を始め, 入力データの分布にあわせて適宜ノードや結合関係の追加・削除を行う Growing Neural Gas (GNG) などが提案されている [8,9,18].

一般に, ニューラルネットワーク (Neural Networks; NN) の研究分野では, ニューロンを追加・削除する研究が盛んに行われており, 総称として, 自己増殖型 (あるいは, 増殖型) NN (self-growing NN あるいは, growing

NN) としばしばよばれる [10,11]. 基本的には, 大きく分けて, 階層型 NN における中間層のニューロンの追加・削除に関する研究と SOM の問題点に基づく拡張に関する研究などがある. 本稿では, 紙面の都合上, GNG に関する後者の研究に焦点をあて, まず, 研究背景について説明し, つぎに, 教師無し分類への適用例など簡単な数値比較実験結果や適用事例, さらに, 関連研究などを紹介する.

2. 自己増殖型ニューラルネットワーク

2.1 Voronoi 図と Delaunay 三角形分割

教師無し分類を行う目的の一つは, 入力データの総数よりも少ない有限個の情報で入力データ全体の特徴を表現することである. ここで, ユークリッド空間上での入力データの次元を n とし, この特徴を表現する情報として m 個の n 次元参照ベクトル w_i ($i=1,2,\dots,m$) を用いることとし, ある入力データ v が与えられたときに, 参照ベクトルとの距離を

$$d(v, w_i) = \|v - w_i\| \quad (1)$$

と定義すると, 最も近い参照ベクトル k は,

$$k = \operatorname{argmin}_i \{d(v, w_i)\} \quad (2)$$

により求められる. ここで, 未知な確率分布からサンプルされるデータ v の確率密度関数を $p(v)$ とすると, 教師無し分類の目的は,

$$f(W) = \int p(v) d(v, w_k) dv \quad (3)$$

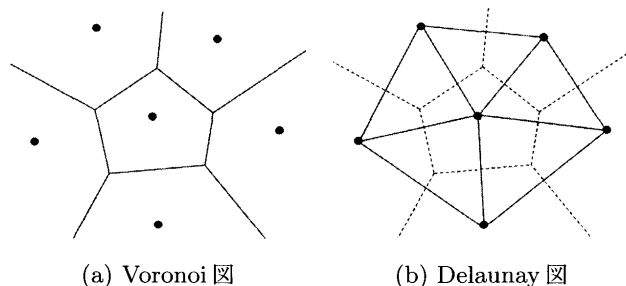
を最小化する参照ベクトルの集合 W を求める問題として帰着される. ただし, 実際は, 必要とされる精度の希求水準や参照ベクトルの個数 m なども考慮される. また, このような操作は, 入力データがサンプルされる領域 V を以下のような部分領域に分割することに等しい.

$$U_i = \{v | v \in V, d(v, w_i) < d(v, w_j), j \neq i\} \quad (4)$$

これは, Voronoi 図とよばれ, 参照ベクトルの点は, Voronoi 点となり, 多角形により囲まれる. ここで, $n+1$ 個の Voronoi 点で構成される三角形は Delaunay 三角形とよばれ, この操作を Delaunay 三角形分割とよぶ [12].

* 首都大学東京

Key Words: neural networks, unsupervised classification, topological mapping, unsupervised learning, Delaunay triangulation.



第1図 Voronoi図とDelaunay図

第1図に2次元平面上でのVoronoi図とDelaunay三角形の例を示す。

2.2 Neural Gas (NG)

Martinetz と Schulten は、結合関係を適宜追加する Neural Gas (NG) を提案した [13,14,18]。SOM と NG の根本的な相違は、SOM がノード間の結合関係を事前に設計しているのに対し、NG では、結合関係は初期化の段階で全くなく、入力データにあわせて適宜結合関係を構築していく点である。さらに、NG では、結合関係の有効性を評価するために年齢の概念を導入している。NG において、各ノードが参照ベクトルにより表現されているものとし、 m 個のノードから構成されるノードの集合を A 、ノード i と j の結合関係を $c_{i,j}$ (0 または 1)、ノード i と j をつなぐエッジの年齢を $a_{i,j}$ とする。ただし、簡略化のため、 $c_{i,j} = c_{j,i}$ および $a_{i,j} = a_{j,i}$ とする。以下、手順を示す。

Step 0 初期化として、ノード集合 A の i 番目の参照ベクトル w_i をランダムに生成し、すべての結合関係を $c_{i,j} = 0$ ($i \neq j$)、エッジの年齢を $a_{i,j} = 0$ ($i \neq j$) とする。

Step 1 入力データ v を $p(v)$ に従ってランダムに取得する。

Step 2 入力データ v に対する各ノードとの距離に基づき、並べ替えを行い、 s_1, s_2, \dots, s_m を得る。

$$d(v, w_{s_1}) < d(v, w_{s_2}) < \dots < d(v, w_{s_m}) \quad (5)$$

Step 3 ここで i 番目のノードの順序を s_j とし、重みの更新に関する適用範囲に関する関数を ζ_{s_j} 、学習係数を η とすると、すべてのノードに対し以下の更新を行う。

$$w_i \leftarrow w_i + \eta \cdot \zeta_{s_j} \cdot (v - w_i), \quad i = 1, 2, \dots, m \quad (6)$$

Step 4 ノード s_1 と s_2 との間にエッジが存在しなければ、新たにエッジを作成する ($c_{s_1, s_2} = 1$)。また、エッジの年齢を 0 にリセットする ($a_{s_1, s_2} = 0$)。

Step 5 ノード s_1 と結合関係のあるすべてのエッジの年齢をインクリメントする。

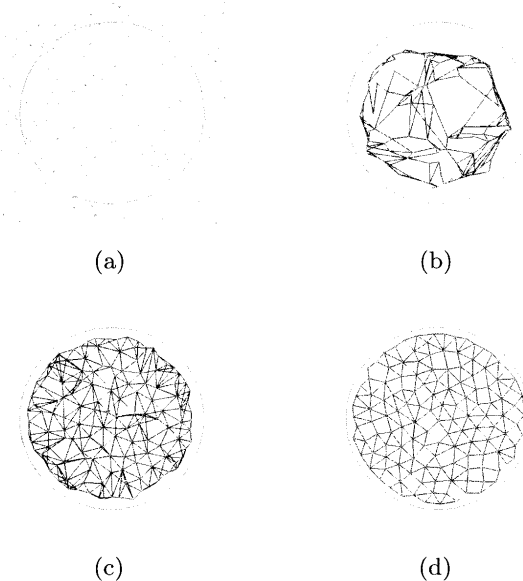
$$a_{s_1, j} \leftarrow a_{s_1, j} + 1 \quad \text{if } c_{s_1, j} = 1 \quad (7)$$

Step 6 事前に設定した閾値 a_{\max} を超える年齢のエッジ

を削除する ($c_{s_1, s_2} = 0$)。

Step 7 Step 1 に戻る

ここで、Step 3 の ζ_{s_j} として、たとえば、 $\zeta_{s_j} = e^{-\frac{k_j}{k_1}}$ ($k_1 < k_2 < \dots < k_m$) が用いられる。SOM の近傍関数と類似した手法が行われ、学習初期では、 k_j の値を小さくし、一つの入力に対し多くのノードが更新されるのに対し、次第に k_j の値を大きくすることにより、近接するノードのみの更新を行うことにより学習が行われる。ただし、SOM では、位相的近傍、すなわち、ノードの結合関係上の近さにあわせて更新されるのに対し、NG では、ユークリッド空間上での近さにあわせて更新される。第2図に NG の学習例を示す。



第2図 NGの学習例

2.3 Growing Cell Structures (GCS)

Fritzke は、サンプルされる入力データにあわせて、最小のノード数から適宜ノードを追加する際、Delaunay 三角形分割 (Delaunay Triangulation) を行う Growing Cell Structure (GCS) を提案した [15-18]。GCS では、トポロジカルな写像を行うだけでなく、入力データの分布に関する確率密度が高い領域では、多くのノードが集まるような特徴をもつ。具体的には、 k 次元単体 (k -dimensional simplex) を用いた教師無し分類であり、Delaunay 三角形分割を行うことができる。また、GCS では、新しいノードを追加する位置を決定するために、積算誤差が導入されている。以下、2次元単体を用いた GCS の手順を述べる。

Step 0 初期化として、2次元単体を構成する三つの参照ベクトル w_1, w_2, w_3 をランダムに生成し、結合関係 $c_{i,j} = 1$ ($i, j = 1, 2, 3, i \neq j$) とする。

Step 1 入力データ v を $p(v)$ に従ってランダムに取得する。

Step 2 入力データ v に対して勝者ノード s を選択する.

$$s = \operatorname{argmin}_{i \in A} \|v - w_i\| \quad (8)$$

Step 3 ノード s について入力データ v との二乗誤差を積算誤差 E_s に加算する.

$$E_s \leftarrow E_s + \|v - w_s\|^2 \quad (9)$$

Step 4 ノード s およびノード s と結合関係があるノードの参照ベクトルを更新する. ただし, η_1 と η_2 を学習係数とする ($\eta_1 > \eta_2$).

$$w_s \leftarrow w_s + \eta_1 \cdot (v - w_s), \quad (10)$$

$$w_j \leftarrow w_j + \eta_2 \cdot (v - w_j), \quad \text{if } c_{s,j} = 1 \quad (11)$$

Step 5 GCS へのデータ入力が回ごとに, つぎの操作を行う.

i. 積算誤差が最大のノード q を選択する.

$$q = \operatorname{argmax}_{i \in A} E_i \quad (12)$$

ii. ノード q と結合関係のあるエッジの中で最も長いエッジを選択し, このエッジに結合するノードを f とすると, このエッジを2分するようにノード r を挿入する.

$$w_r = 0.5 \cdot (w_q + w_f) \quad (13)$$

iii. つぎに, ノード q, f 間のエッジを削除し ($c_{q,f} = 0$), ノード q, r および r, f 間にエッジを追加する ($c_{q,r} = 1, c_{r,f} = 1$). 最後に q と f とともに結合関係のあるノードとノード r 間にエッジを追加する.

$$c_{r,j} = 1, \quad \text{if } c_{q,j} = 1 \text{ and } c_{f,j} = 1 \quad (14)$$

iv. ノード r と結合しているノードの積算誤差を以下の式により更新する. ただし, ノード r に結合しているノードの集合を N_r , 係数を α とする.

$$E_j \leftarrow E_j - \frac{\alpha}{|N_r|} E_j, \quad \text{if } c_{r,j} = 1 \quad (15)$$

vi. 最後に, ノード r の積算誤差をノード r と結合するノードの積算誤差の平均とする.

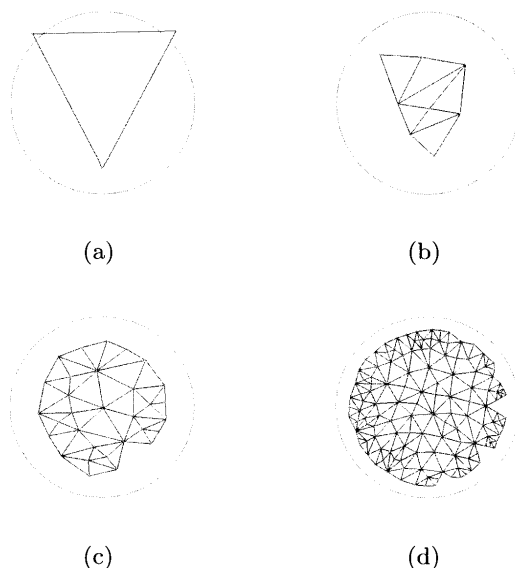
$$E_r = \frac{1}{|N_r|} \sum_{i \in N_r} E_i \quad (16)$$

Step 6 減衰係数を β とし ($1.0 > \beta > 0$), すべてのノードの積算誤差を減少させる.

$$E_i \leftarrow E_i - \beta E_i \quad (\forall i \in A) \quad (17)$$

Step 7 終了条件が満たされない場合は, Step 2 に戻る.

GCS では, Delaunay 三角形分割を行うために, Step 5 iii において, 新しいノードを追加する際, 挿入元の二つのノードに共通して結合関係のあるノードとの結合を



第3図 GCS の学習例

行っている. また, Step 3 において提案初期 [15] においては, 積算誤差ではなく, 選択回数に合わせてノードが追加された. 第3図に GCS の学習例を示す.

2.4 Growing Neural Gas (GNG)

Fritzke は, ノードや結合関係の追加・削除を行うことができる Growing Neural Gas (GNG) を提案した [8,9]. 一般には, GNG は, GCS において必ずしも Delaunay 三角形分割を行わないモデルであるとともに, 年齢の概念など NG といくつかの特徴を共有する. 具体的には, GNG は, サンプルされた入力データに対し, 入力データとノードの参照ベクトルとの距離に基づき, ノードや結合関係の追加・削除を逐次的に行う手法であり, クラスタとして認識可能なかたまりに境界を生成しながら, 教師無し分類が行える手法である. 以下, GNG の手順を述べる.

Step 0 初期化として, 二つのノードの参照ベクトル w_1 と w_2 をランダムに生成し, 結合関係 $c_{1,2} = 1$, エッジの年齢 $a_{1,2} = 0$ とする.

Step 1 入力データ v を $p(v)$ に従ってランダムに取得する.

Step 2 入力データ v に対する勝者ノード s_1 と第2勝者ノード s_2 を選択する.

$$s_1 = \operatorname{argmin}_{i \in A} \|v - w_i\| \quad (18)$$

$$s_2 = \operatorname{argmin}_{i \in A \setminus \{s_1\}} \|v - w_i\| \quad (19)$$

Step 3 ノード s_1 について入力データ v との二乗誤差を積算誤差 E_s に加算する.

$$E_{s_1} \leftarrow E_{s_1} + \|v - w_{s_1}\|^2 \quad (20)$$

Step 4 ノード s_1 およびノード s_2 と結合関係があるノ

ドの参照ベクトルを更新する. ただし, η_1 と η_2 を学習係数とする ($\eta_1 > \eta_2$).

$$w_{s_1} \leftarrow w_{s_1} + \eta_1 \cdot (v - w_{s_1}), \quad (21)$$

$$w_j \leftarrow w_j + \eta_2 \cdot (v - w_j), \quad \text{if } c_{s_1, j} = 1 \quad (22)$$

Step 5 エッジの年齢を 0 にリセットする ($a_{s_1, s_2} = 0$). また, ノード s_1 と s_2 との間にエッジが存在しなければ, 新たにエッジを作成する ($c_{s_1, s_2} = 1$).

Step 6 ノード s_1 と結合関係のあるすべてのエッジの年齢をインクリメントする.

$$a_{s_1, j} \leftarrow a_{s_1, j} + 1, \quad \text{if } c_{s_1, j} = 1 \quad (23)$$

Step 7 事前に設定した閾値 a_{\max} を超える年齢のエッジを削除する ($c_{s_1, s_2} = 0$). その結果, 他のノードと結合関係をもたないノードが現れた場合は, 当該ノードを削除する.

Step 8 GNG へのデータ入力が入回ごとに, つぎの操作を行う.

- i. 積算誤差が最大のノード q を選択する.

$$q = \operatorname{argmax}_{i \in A} E_i \quad (24)$$

- ii. ノード q と結合関係のあるエッジの中で最も長いエッジを選択し, このエッジに結合するノードを f とすると, このエッジを 2 分するようにノード r を挿入する.

$$w_r = 0.5 \cdot (w_q + w_f) \quad (25)$$

- iii. つぎに, ノード q, f 間のエッジを削除し ($c_{q, f} = 0$), ノード q, r および r, f 間にエッジを追加する ($c_{q, r} = 1, c_{r, f} = 1$).

- iv. ノード q, f の積算誤差を以下の式により更新する.

$$E_q \leftarrow E_q - \alpha E_q \quad (26)$$

$$E_f \leftarrow E_f - \alpha E_f \quad (27)$$

- v. 最後に, ノード q, f の積算誤差の平均をノード r の積算誤差とする.

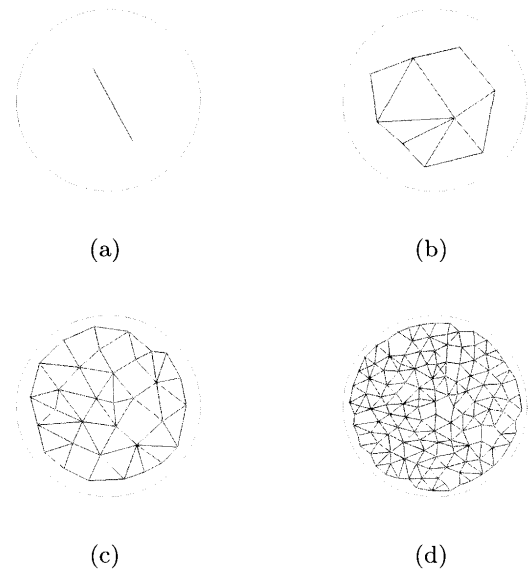
$$E_r = 0.5 \cdot (E_q + E_f) \quad (28)$$

Step 9 すべてのノードの誤差を減らす.

$$E_i \leftarrow E_i - \beta E_i \quad (\forall i \in A) \quad (29)$$

Step 10 終了条件が満たされない場合は, Step 2 に戻る.

第 4 図に GNG の学習例を示す.



第 4 図 GNG の学習例

2.5 Growing Neural Gas with Triangulation (GNG-T)

GNG は, 複数のクラスタに分割可能であるが, GCS はノードや結合関係の削除を行わないため, 複数のかたまりへの分割が行えない. GCS の拡張として, 不要なノードやエッジを削除する手法が提案されており [17,18], Delaunay 図の外周上のエッジやノードの削除は容易であるが, Delaunay 三角形分割を保持しながら Delaunay 図の内部のエッジやノードの削除は困難であることが指摘されている. したがって, GNG の拡張の一例として, ヒューリスティックに三角形分割を行う手法を導入した Growing Neural Gas with Triangulation (GNG-T) を紹介する [19].

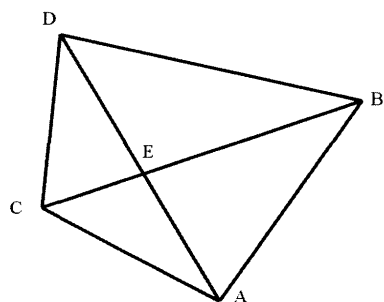
初期化は, GCS と同様に 2 次元単体を構成する三つの参照ベクトルをランダムに生成する. それ以降は, GNG と同じ操作を用い, GNG-T では, GNG のノードやエッジの追加・削除があった際, 当該ノード近辺を集中して三角形分割のための探索を行う. これにより計算コストを軽減できることが期待される. したがって, GNG の Step 8 のつぎに, 対角線をもたない四角形の探索と交差点探索を追加する.

2.5.1 四角形探索

ここでは, 対角線をもたない四角形探索を行い, 対角線を追加することにより三角形分割を行う手法を説明する. 第 5 図のように, 任意に頂点 A を決め, A と結合関係のある二つのノード (図中では, B と C) を任意に選択し, そのノード間に結合関係がない場合, さらに B と C に共通の結合関係があるノード D を探索する. さらに D と A に結合関係がなければ $A-B-D-C$ を四角形の候補とする. ここで, 第 5 図のように, 対角線 AD や BC をもたなくても, 四角形の候補と誤認識してしまう

場合がある．したがって，A,B,D,C 以外のノード E で A,B,D,C すべてのノードから E との間に結合関係がなければ，A-B-D-C を四角形と判断する．

つぎに候補となる三角形分割を行うために，四角形探索で候補として選択された四つのノード A,B,D,C について，対角線 AD, BC の距離を求め，短い方のノード間にエッジを追加する．



第 5 図 四角形探索

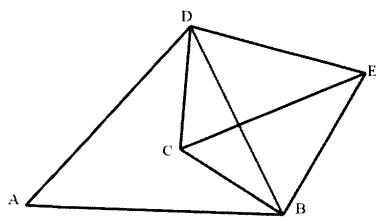
2.5.2 交差点探索

四角形探索を行った後，交差点探索を行う．たとえば，第 6 図に示すように，対角線をもたない四角形 A-B-C-D において，エッジ BD が追加された場合，他のエッジと交差する場合があります（図中では，エッジ CE），長い方のエッジを削除する必要がある．図中において，線分 BD と CE が交差しているかどうかの判断基準として，次式で計算された γ_1 と γ_2 の値を用いる．

$$\gamma_1 = (x_C - x_E)(y_D - y_C) + (y_C - y_E)(x_C - x_D) \quad (30)$$

$$\gamma_2 = (x_C - x_E)(y_B - y_C) + (y_C - y_E)(x_C - x_B) \quad (31)$$

ここで， γ_1 と γ_2 の積が 0 以下であれば交差していると判断し，長い方のエッジを削除する．



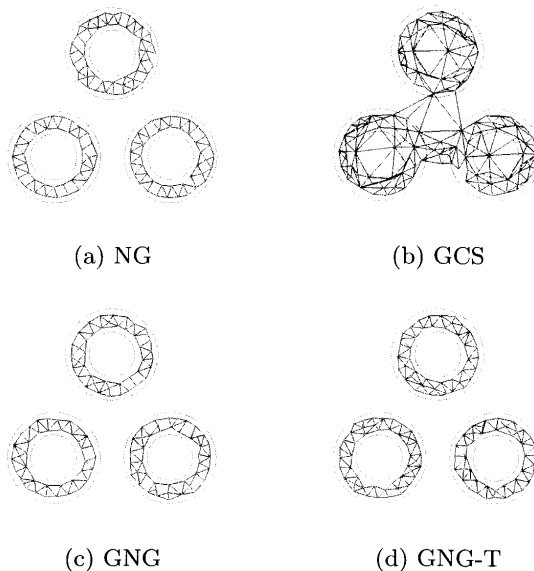
第 6 図 交差点探索

3. 各種手法の比較と適用事例

3.1 シミュレーション結果比較

ここでは，GCS, GNG, GNG-T の教師無し分類のシミュレーション結果比較を行う．比較は色々行われており，詳細については，参考文献を参照されたい [19,20]. 入力データとして，三つのリング状のデータ分布を与え，等確率でサンプルされるものとする．NG, GCS, GNG, GNG-T の 50000 回の学習結果をそれぞれ第 7 図に示す．

第 7 図 (b) において，NG は GNG と類似した結果が得られているが，距離に基づく順序付けを行っているため，計算時間が長くなっている．GCS は三角形分割を行いながら入力データを覆っているが，入力データのない領域にもノードやエッジが生成されている．第 7 図 (c) において，GNG もまた入力データを覆うように構成されているが，三角形分割を完全には行っていない．第 7 図 (d) において，GNG-T は三角形分割が行えていることがわかる．第 1 表に GCS, GNG, GNG-T の比較を示す．GCS は，ノードやエッジの削除がないため，計算時間が短くなるが，その反面，ノード数やエッジ数が多くなっている．GNG-T は GNG と比較して，交差点判定などの操作を導入しているため，計算時間が長くなっている様子がわかる．



第 7 図 各アルゴリズムの学習の比較

第 1 表 シミュレーション比較結果

	計算時間 [ms]	ノード数	エッジ数
NG	11400	135	264
GCS	1200	136	407
GNG	3900	135	261
GNG-T	5400	135	284

3.2 適用事例と拡張例

本稿で紹介した自己増殖型 NN は，様々な問題に適用されている．たとえば，GCS は，3 次元サーフェスモデルを構築するためのメッシュ作成に用いられている [21]. また，有限要素法では，詳細な解析を必要とする領域を密に再分割する方法がとられる．本手法では当該領域のサンプル回数を増やすことにより，より小さな三角形へと分割が行える．他の例として，強化学習において，連続値を知覚入力とする場合や多次元の知覚入力の場合

は、状態空間が爆発することが指摘されている。そこで、GNGを用いることにより、エージェントの状態空間を離散化・削減する手法が提案されている[22,23]。また、ロボットビジョンや画像処理における前処理などにも適用されている[24]。

一方で、教師無し分類において、GNGを非定常なデータ分布に追従させるための拡張手法として、Self-Organizing Incremental Neural Network (SOINN)[25,26]などが提案されている。さらに、半教師あり学習への拡張なども行われている[24,25]。

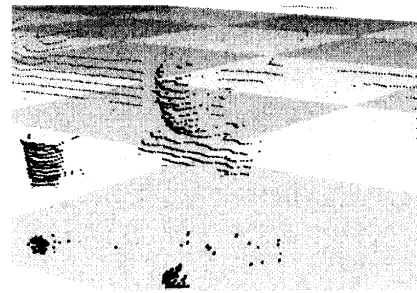
ここでは、GNG-Tを用いた3次元復元への適用事例を紹介する[19,27]。具体的には、チルト機能をもつ測域センサから得られる距離情報とカメラ画像情報を融合するシンセシス処理を行った基礎実験を示す。CCDカメラの解像度は 160×120 ピクセル、測域センサの1回あたりの水平方向のスキャンにおける計測点数は680点である。チルト制御は1秒間に約10度、上方向に4秒間チルトアップさせ、人間の顔データをGNG-Tによって学習させた。実験結果を第8図に示す。計測された距離情報は背景の壁や人間の周りの物体も含まれるため、GCSを用いると、人間だけを切り出すことは困難である。しかしながら、GNG-Tを用いることにより、人間の領域だけ抽出するとともに、サーフェス面を構成するために、三角形分割が行えていることがわかる。しかしながら、実世界のデータを扱う場合、データの偏りやノイズの影響で、データが存在しない四角形を三角形分割したり、データが少し点在する五角形を三角形分割が行えないなどの問題がある。

3.3 関連研究

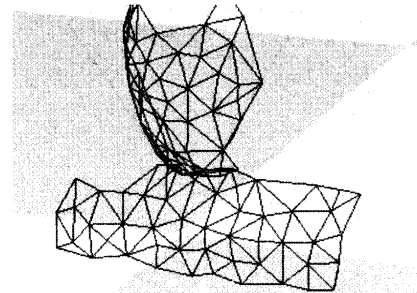
自己増殖型NNについて様々な方法論的比較が行われてきた。たとえば、SOMの他の拡張として、Growing Hierarchical SOMや、parameterless SOMなどが提案されており、様々な比較が行われている[28,29]。ノードの追加・削除が行われる他のNNの例として、適応共鳴理論(Adaptive Resonance Theory; ART)やradial basis function network, Generative Topographic Mapping (GTM)などがある。

ARTは、注意サブシステムと適応サブシステムにより構成され、警戒変数を用いて、入力データと記憶パターンの整合性を判断し、どのカテゴリにも属さない場合は、新たなカテゴリノードが追加される構造をもつ[30,31]。また、バックプロパゲーション法などがエラーに基づく学習であるのに対し、ARTの特徴として、距離に基づく高速学習があげられる。

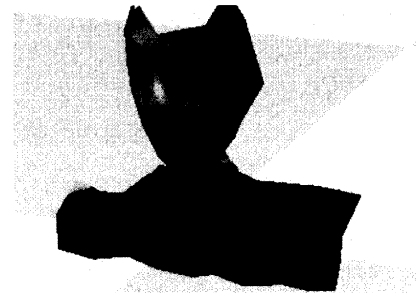
GTMは、Expectation-Maximization (EM) アルゴリズムを用いてconstrained mixture of expertsのパラメータを推定する手法で、SOMのいくつかの問題点を解決している[32-35]。また、ノードの追加削除を容易に行いながら、トポロジカルな写像を行うTopographic



(a) 計測した距離情報



(b) GNG-Tを用いたクラスタリングの結果



(c) カメラ画像と融合した結果

第8図 GNG-Tを用いた3次元復元の一例

Products of Experts (ToPoE) や k-means アルゴリズムを GTM に適用した Harmonic Topographic Mapping (HaToM) など提案されている[36]。

以上のように、自己増殖するNNの研究は多岐にわたり、生物学的視点や統計学的視点など、様々な視点から研究が行われている。

4. おわりに

本稿では、自己増殖型ニューラルネットワークを用いた教師無し分類学習に関する最近の研究動向について紹介した。自己増殖型ニューラルネットワークの研究は、ノード数を自動調整しながら、結合関係をも学習する手法であるが、教師無し分類が行われた後は、データの分布にあわせた結合関係を可視化のために用いることができる。さらに、各ノードを探索の対象点とすると、この結合関係であるエッジをたどり、最適化問題などで局所探索を行う手法へと拡張可能である。このように、自己

増殖型ニューラルネットワークは、トポロジカルな構造を、様々な問題解決への手掛かりの一つとして積極的に用いることができ、応用範囲は多岐にわたる。最後に、自己増殖型ニューラルネットワークに関する基礎理論や実世界の様々な問題への応用に関する研究のさらなる発展を期待する。

(2008年9月1日受付)

参考文献

- [1] T. Kohonen: *Self-Organization and Associative Memory*, Springer-Verlag (1984)
- [2] T. Kohonen: Self-organized formation of topologically correct feature maps; *Biological Cybernetics*, pp. 43–69 (1982)
- [3] T. コホネン(著), 徳高, 堀尾, 大北, 大藪, 藤村(訳): 自己組織化マップ, シュプリンガーフェアラーク東京 (2005)
- [4] T. Kohonen: *Self-Organizing Maps*, Springer (2000)
- [5] 須藤, 湯野, 田向, 関根: 適応型木構造 SOM の提案と画像分類への応用; 電子情報通信学会技術研究報告 ニューロコンピューティング研究会, Vol. 107, No. 413, pp. 109–114 (2008)
- [6] T. Hastie, R. Tibshirani and J. H. Friedman: *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, Springer-Verlag (2001)
- [7] R. O. Duda, P. E. Hart and D. G. Stork: *Pattern Classification* (2nd.Ed.), John Wiley & Sons, Inc. (2001)
- [8] B. Fritzke: A growing neural gas network learns topologies; *Advances in Neural Information Processing Systems* 7, pp. 625–632 (1995)
- [9] B. Fritzke: Growing self-organizing networks – why?; *European Symposium on Artificial Neural Networks*, pp. 61–72 (1996)
- [10] J. A. Anderson and E. Rosenfeld: *Neurocomputing*; The MIT Press, [This book collects historical papers concerning neurocomputing] (1988)
- [11] J.-S. R. Jang, C.-T. Sun and E. Mizutani: *Neuro-Fuzzy and Soft Computing*, Prentice-Hall, Inc. [This book introduces the basics of soft computing such as fuzzy, neural, evolutionary computing, reinforcement learning, and their applications] (1997)
- [12] S. M. Omohundro: The Delaunay triangulation and function learning; *Technical Report Tr-90-001*, International Computer Science Institute (1990)
- [13] T. M. Martinetz and K. J. Schulten: A “neural-gas” network learns topologies; *Artificial Neural Networks* (T. Kohonen, K. Mäkisara, O. Simula, and J. Kangas, eds.), pp. 397–402 (1991)
- [14] T. M. Martinetz and K. J. Schulten: Topology representing networks; *Neural Networks*, Vol. 7, No. 3, pp. 517–522 (1994)
- [15] B. Fritzke: Growing cell structures - a self organizing network in k dimensions; *Artificial Neural Networks*, Vol. 2, No. 2, pp. 1051–1056 (1992)
- [16] B. Fritzke: Growing cell structures - a self-organizing network for unsupervised and supervised learning; Technical Report ICSTR-93-026, International Computer Science Institute (1993)
- [17] B. Fritzke: Unsupervised clustering with growing cell structures; *Proc. Int. Joint Conf. on Neural Networks*, Vol. 2, pp. 531–536 (1991)
- [18] B. Fritzke: Unsupervised ontogenetic networks; *Handbook of Neural Computation*, IOP Publishing and Oxford University Press (1996)
- [19] N. Kubota and M. Satomi: Growing neural gas with triangulation for reconstructing a 3D surface model; *World Automation Congress* (2008) (to appear in)
- [20] D. Heinke and F. H. Hamker: Comparing neural networks: A benchmark on growing neural gas, growing cell structures, and fuzzy ARTMAP; *IEEE Transactions on Neural Networks*, Vol. 9, No. 6, pp. 1279–1291 (1998)
- [21] I. P. Ivriissimtzis, W.-K. Jeong and H.-P. Seidel: *Using Growing Cell Structures for Surface Reconstruction*, IEEE Computer Society (2003)
- [22] H. Handa: State space construction of reinforcement learning agents based upon anticipated sensory changes; *International Joint Conference on Neural Networks* (2004)
- [23] 松本, 畠中, 魚崎: Growing Neural Gas を用いた Q-Learning とそのサッカーエージェントの行動選択への適用; 第 16 回インテリジェントシンポジウム講演論文集, pp. 349–352 (2006)
- [24] N. Kubota and M. Satomi: Visual tracking of a partner robot through interaction with people; *International Conference on Soft Computing and Intelligent Systems and International Symposium on advanced Intelligent Systems* (2008) (to appear in)
- [25] 須藤, 佐藤, 長谷川: 自己増殖型ニューラルネットを用いたノイズのある環境下での追加学習が可能な連想記憶モデル; 日本神経回路学会誌, Vol. 15, No. 2, pp. 98–109 (2008)
- [26] Y. Kamiya, F. Shen and O. Hasegawa: An incremental neural network for online supervised learning and topology learning; *Journal of Advanced Computational Intelligence and Intelligent Informatics* (2006)
- [27] N. Kubota, M. Satomi, K. Taniguchi and Y. Nogawa: Human three-dimensional modeling based on intelligent sensor fusion for a tele-operated mobile robot; *Knowledge-Based Intelligent Information and Engineering Systems*, pp. 98–113 (2007)
- [28] E. Berglund and J. Sitte: The parameterless self-organizing map algorithm; *IEEE Transaction on Neural Networks*, Vol. 17, No. 2, pp. 305–316 (2006)
- [29] A. Forti and G. L. Foresti: Growing hierarchical tree SOM: An unsupervised neural network with dynamic topology; *Neural Networks*, Vol. 19, No. 10, pp. 1568–1580 (2006)

- [30] S. Grossberg: Adaptive pattern classification and universal recoding. I.: parallel development and coding of neural feature detectors; *Biol. Cybern.*, Vol. 23, pp. 121-134 (1976)
- [31] G. A. Carpenter and S. Grossberg: Adaptive resonance theory; *The Handbook of Brain Theory and Neural Networks, Second Edition* (M.A. Arbib (ed.)), MIT Press, pp. 87-90 (2003)
- [32] M. Pena and C. Fyfe: The topographic neural gas; *Intelligent Data Engineering and Automated Learning*, pp. 241-248 (2006)
- [33] C. M. Bishop, M. Svensen and C. K. I. Williams: Gtm: A principled alternative to the self-organizing map; *Advances in Neural Information Processing Systems* (1997)
- [34] A. Vellido, W. El-Deredy and P. J. G. Lisboa: Selective smoothing of the generative topographic mapping; *IEEE Trans. Neural Networks*, Vol. 14, No. 4, pp. 847-852 (2003)
- [35] C. Fyfe: Two topographic maps for data visualization; *Data Mining and Knowledge Discovery*, Vol. 14, No. 2, pp. 207-224 (2007)
- [36] M. Pena and C. Fyfe: Outlier identification with the harmonic topographic mapping; *European Symposium on Artificial Neural Networks*, pp. 289-294 (2006)

著者略歴

久保田直行 (正会員)



1969年8月10日生。1997年名古屋大学大学院工学研究科マイクロシステム工学専攻博士後期課程修了。博士(工学)。1997年大阪工業大学工学部機械工学科助手, 1999年同講師。2000年福井大学工学部知能システム工学科助教授, 2004年東京都立大学大学院工学研究科機械工学専攻助教授を経て, 2005年首都大学東京システムデザイン学部ヒューマンメカトロニクスシステムコース准教授, 現在に至る。2002年から2006年まで科学技術振興機構戦略的創造研究推進事業「相互作用と賢さ」領域研究員, 2006年から2008年まで同事業発展研究研究員兼任。英国王立協会客員教授(ポーツマス大学, 2007年)。計測自動制御学会, 日本知能情報ファジィ学会, 日本ロボット学会, ヒューマンインタフェース学会, 日本機械学会, 電気学会, IEEEなどの会員。

里見将志



2007年東京都立大学工学部精密機械工学科卒業。現在, 首都大学東京大学院システムデザイン研究科ヒューマンメカトロニクスシステム専攻博士前期課程在学。主として, 実環境における三次元計測に関する研究に従事。