

Growing Neural Gas with Different Topologies を用いた 3 次元物体のトラッキング

3D Object Tracking using Growing Neural Gas with Different Topologies

和田 亮雅 (岡山大) 竹田 宗馬 (岡山大)
宮瀬 光梨 (岡山大) 〇正 戸田 雄一郎 (岡山大)
正 松野 隆幸 (岡山大) 正 見浪 護 (岡山大)

Akimasa WADA, Okayama University
Soma TAKEDA, Okayama University
Hikari MIYASE, Okayama University
Yuichiro TODA, Okayama University
Takayuki MATSUNO, Okayama University
Mamoru MINAMI, Okayama University

3D perception system of an autonomous robot is one of the most important technologies for performing tasks in an unknown environment. This paper proposes a real-time 3D object tracking method based on Growing Neural Gas with Different Topologies (GNG-DT). First of all, we explain a GNG-DT for learning the topological structures from the 3D point cloud data with multiple properties such as color and normal vector information. Next, we propose a real-time 3D object tracking method from the topological structure by utilizing the center of gravity in each cluster. Finally, we show some experimental results of the proposed method and discuss the effectiveness of the proposed method.

Key Words: 3D object tracking, Growing Neural Gas, Topological structure learning

1 緒言

近年, 自律移動ロボットへの期待が高まっており, 災害現場やホテルや病院における接客などその用途は様々である. 自律移動ロボットにおいて, 空間の構造を把握しつつその位置に含まれる特徴となる属性を抽出する能力は, 未知環境において自律ロボットがタスクを適応的に遂行していく上で, 必要不可欠な能力である. そこで空間的な構造を学習すると同時に, 形状や物体の位置関係を記述するためのノードとエッジから構成された位相構造を学習する手法として, Kohonen が提案した自己組織化地図に基づく手法が提案されている [1]. その中でも, Fritzke によって提案されている Growing Neural Gas (GNG) は, ノードを追加しつつ, 冗長なノードやエッジを削除するアルゴリズムを含んでおり, 未知なデータ分布に対して, 空間の構造を適切に学習することが可能な手法となっている [2, 3]. しかしながら, GNG を用いたこのような研究には, 未解決問題が存在する. その中の 1 つに, 複数の特徴量から構成される入力ベクトルに対して, それぞれの属性ごとにおける位相構造を同時に学習できないといった課題が存在する. 例えば, カメラから取得された画像を用いて物体の位置情報と色情報を組み合わせた入力ベクトルを GNG により学習させた場合, 位置情報と色情報の両方が含まれた空間を学習してしまうため, 物体の位置を把握するための位置情報に関する位相構造を適切に構築することができない. そこで, 先行研究では関連重要度と呼ばれる重みつき距離を用いた手法である改良型 GNG with Utility (GNG-U II) を提案しており [4, 5], 重みを適切に設計することによって, 位置情報の近傍関係を適切に構築すると同時に色情報の学習を行うことが可能となる. しかしながら, 新たな属性の特徴ベクトルを追加する際には, 重みの再設計が必要となる. また, 最大値が時々刻々と変化するような属性の特徴ベクトルにおいては正規化が困難であり, 空間構造のスケールが変化した際に, 属性ごとのスケールに合わせた重みの設計をその都度しなければならないといった問題点が存在する. さらに学習後に構築された位相構造のエッジ部分を削除することによって異なるクラスターの生成を行うなどの後処理が追加されているため, 複数の属性から構成されている特徴ベクトルを同時に学習し属性の情報に応じたクラスタリングをオンラインで行うことは未だできていない. そこで我々は複数の属性から構成される入力ベクトルに対し

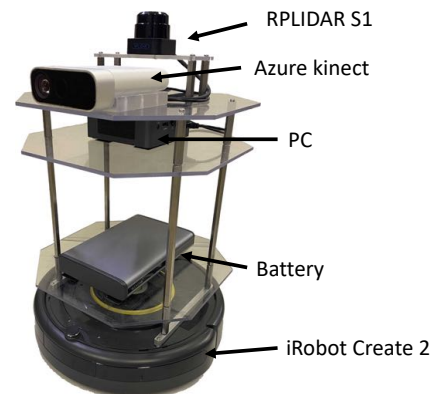


Fig.1 Mobile robot

て, 位置情報の空間構造を学習しつつ異なる属性ごとの位相構造を構築し, 各属性情報に応じたクラスタリングが可能な学習アルゴリズムである Growing Neural Gas with Different Topologies (GNG-DT) を提案してきた [6]. 本稿では, GNG-DT によって取得したクラスターをトラッキングすることを目的とする. 以降の章ではそのトラッキングの手法について述べる.

2 システム構成

本研究に用いたロボットの外観を Fig.1 に, システム構成を Fig.2 に示す. ベースとなる対向二輪型の移動機構は iRobot 社の iRobot Create2 を使用し, 制御 PC はインテル R NUC キット NUC8i5BEH. ロボットは 3 つの階層で作成されており, 1 層目にバッテリー, 2 層目に PC, 3 層目に 2 次元環境地図構築と自己位置推定のための Lidar 及び 3 次元物体をトラッキングするための RGB-D カメラを配置している. LIDAR には, SLAMTEC 社製の RPLIDAR S1 を用いており, 40 m の範囲内で 2 次元, 360 度のスキャンが可能で, 1 秒間に最大 9200 サンプルのレーザ測

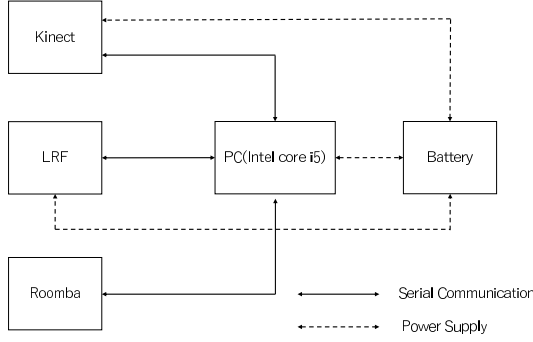


Fig.2 System configuration

距を行う。また、RGB-D カメラには、Microsoft 社が開発している Azure Kinect を用いる。制御方法はファジィ制御を用いており、目的地追従および障害物回避を行う [7]。

3 Growing Neural Gas with Different Topologies

本研究では、GNG-U II における関連重要度を用いた距離尺度の概念を用い位置座標空間の構造を保持しつつ、各属性の位相構造を構築するアルゴリズムである GNG-DT を用いる [6]。そのため、本章では GNG-DT のアルゴリズムに関して説明していく。はじめに、GNG-DT において用いる主な変数の定義を行う。まず、属性の集合を $S = \{ \text{位置情報 (pos)}, \text{色情報 (col)}, \text{輝度情報 (bri)}, \dots \}$ とし、入力ベクトルと参照ベクトルをそれぞれ、 $\mathbf{v} = \{ \mathbf{v}^{pos}, \mathbf{v}^{col}, \mathbf{v}^{bri}, \dots \}$, $\mathbf{h}_i = \{ \mathbf{h}_i^{pos}, \mathbf{h}_i^{col}, \mathbf{h}_i^{bri}, \dots \}$ と定義する。次に、ある属性 o における入力ベクトルと i 番目のノードの参照ベクトルとの距離を d_i^o とし、以下のように定義する。

$$d_i^o = \| \mathbf{v}^o - \mathbf{h}_i^o \| \quad (1)$$

また、ある属性 $o (\in S)$ のエッジの集合を $C^o = \{ c_{1,2}^o, \dots, c_{i,j}^o, \dots \}$ と定義することによって、複数の属性ごとに位相構造を構築していく。

以下では、全体アルゴリズムの詳細な内容に関して説明する。

Step0. 初期化として、二つのノードの参照ベクトル $\mathbf{h}_1$ と $\mathbf{h}_2$ をランダムに生成し、結合関係 $c_{1,2}^o = 1$ ($\in S$)、エッジの年齢 $g_{1,2} = 0$ とする。

Step1. 入力ベクトル $\mathbf{v}$ を計測データからランダムに 1 つ取得する。

Step2. 入力ベクトル $\mathbf{v}$ に対する第 1 勝者ノード s_1 と第 2 勝者ノード s_2 を選択する。

$$\begin{aligned} s_1 &= \arg \min_{i \in A} d_i^{pos} \\ s_2 &= \arg \min_{i \in A \setminus s_1} d_i^{pos} \end{aligned} \quad (2)$$

ここで、 A はノード番号の集合を表す。

Step3. ノード s_1 について入力データ $\mathbf{v}$ との二乗誤差を積算誤差 E_{s_1} に加算する。

$$E_{s_1} \leftarrow E_{s_1} + \{ d_{s_1}^{pos} \}^2 \quad (3)$$

Step 4. ノード s_1 及びノード s_1 と各属性における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 ($\eta_1 > \eta_2$)。

$$\begin{aligned} \mathbf{h}_{s_1} &\leftarrow \mathbf{h}_{s_1} + \eta_1 (\mathbf{v} - \mathbf{h}_{s_1}) \\ \mathbf{h}_j^o &\leftarrow \mathbf{h}_j^o + \eta_2 (\mathbf{v} - \mathbf{h}_j^o) \quad \text{if } c_{s_1,j}^o = 1 \end{aligned} \quad (4)$$

Step 5. エッジの年齢を 0 にリセットし ($g_{s_1,s_2} = 0$)、ノード s_1 と s_2 との間に位置情報のエッジが存在しなければ、新たに位置情報のエッジを作成する ($c_{s_1,s_2}^{pos} = 1$)。また、位置情報以外の属性 $o (\in S \setminus pos)$ に関するエッジの作成は、以下の式によって行われる。

$$\begin{cases} c_{s_1,s_2}^o = 1 & \text{if } \| \mathbf{h}_{s_1}^o - \mathbf{h}_{s_2}^o \| < \tau^o \\ c_{s_1,s_2}^o = 1 & \text{otherwise} \end{cases} \quad (5)$$

ここで、 τ^o は、しきい値を表す。

Step6. ノード s_1 と位置情報の結合関係がある全てのエッジの年齢をインクリメントする。

$$g_{s_1,j} \leftarrow g_{s_1,j} + 1 \quad \text{if } c_{s_1,j}^{pos} = 1 \quad (6)$$

Step7. 事前に設定した閾値 g_{max} を超える年齢の全ての属性 $o (\in S)$ のエッジを削除する ($c_{s_1,s_2}^o = 0$)。その結果、位置情報の結合関係において、他のノードと結合関係を持たないノードは、そのノードを削除する。

Step8. データ入力が入回ごとに、次の操作を行う。

i. 積算誤差が最大のノード u を選択する。

$$u = \arg \max_{i \in A} E_i \quad (7)$$

ii. ノード u と結合関係のあるノードの中で最も積算誤差の大きいノードを f とし、 u と f を結合するエッジを 2 分するようにノード r を挿入する。

$$\mathbf{h}_r = 0.5(\mathbf{h}_u + \mathbf{h}_f) \quad (8)$$

iii. ノード u, f 間の全ての属性 $o (\in S)$ に関するエッジを削除し ($c_{u,f}^o = 0$)、ノード u, r 及び r, f 間に位置情報のエッジを追加する ($c_{u,r}^{pos} = 1, c_{r,f}^{pos} = 1$)。また、位置情報以外の属性 $o (\in S \setminus pos)$ に関して以下の式によりエッジの追加の判断を行う。

$$\begin{cases} c_{i,j}^o = 1 & \text{if } \| \mathbf{h}_i^o - \mathbf{h}_j^o \| < \tau^o \\ c_{i,j}^o = 1 & \text{otherwise} \end{cases} \quad (9)$$

iv. ノード u, f の積算誤差を減衰率 α ($0 \leq \alpha \leq 1$) を用いて以下の式により更新する。

$$\begin{aligned} E_u &\leftarrow E_u - \alpha E_u \\ E_f &\leftarrow E_f - \alpha E_f \end{aligned} \quad (10)$$

v. ノード u, f の積算誤差の平均をノード r の積算誤差とする。

$$E_r = 0.5(E_u + E_f) \quad (11)$$

Step9. 全てのノードの誤差を減衰率 β ($0 \leq \beta \leq 1$) により減衰させる。

$$E_i \leftarrow E_i - \beta E_i \quad (\forall i \in A) \quad (12)$$

Step10. 終了条件が満たされない場合は Step1 に戻る。

GNG-DT では、位相構造の構築時に複数属性の位相構造を逐次的に更新することによって、異なる属性でのデータのクラスタリングがオンライン学習の枠組みの中で可能となっている。

4 対象物のトラッキング手法

本研究における対象物のトラッキングでは、位相構造の重心を利用して対象物のトラッキングを行う。Fig.3 に手法の概略図を示す。時刻 t でのあるクラスタ cl_i ($i=1,2,\dots$) の各属性の重心 $\mathbf{v}_{g_i,t} = \{ \mathbf{v}_{g_i,t}^{pos}, \mathbf{v}_{g_i,t}^{col}, \mathbf{v}_{g_i,t}^{nor} \}$ が次の時刻 $t+1$ のどの位相構造の重心に近いかを計算し、最も近い位相構造を同じラベルにするという手法である。今回は法線情報、位置情報、色情報の 3 つの属性について調べたのでそのアルゴリズムを記載する。

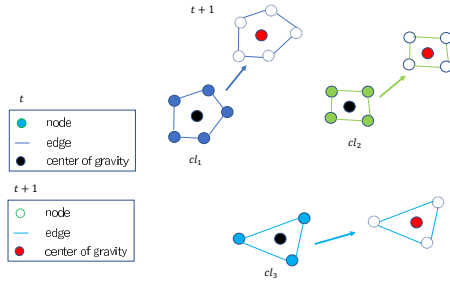


Fig.3 Tracking the center of gravity of the position of the topological structure

Step0. 法線情報について、時刻 t と $t+1$ のクラスタ重心の角度を比較する。ここでは法線を正規化した後、それらの内積を取ることで最も大きい値を取るクラスタ重心を探索し、判断する。

$$\cos \theta_j = \frac{|\mathbf{v}_{g,i}^{nor} \cdot \mathbf{v}_{g,j}^{nor}|}{|\mathbf{v}_{g,i}^{nor} - \mathbf{v}_{g,j}^{nor}|} \quad (13)$$

Step1. 計算された $\cos \theta$ が $\max_{nor}$ 以上であればその時の番号を保存する。異なる場合は step0 に戻る。

$$N_1 = j \quad \text{if} \quad \cos \theta > \max_{nor} \quad (14)$$

$$(15)$$

Step2. 保存したクラスタの番号から最小の 2 乗距離を持つクラスタ重心を選択する。

$$d_j^{col} = \|\mathbf{v}_{g,n_1}^{col} - \mathbf{v}_{g,j}^{col}\|^2 \quad (16)$$

Step3. d_{col} が $\min_{col}$ 未満の時、クラスタ番号 N_1 を保存する。ここで条件を満たさなかった場合は step0 に戻る。step3 における N_1 は時刻 t における番号 i のクラスタ番号に複数の属性において最も近いクラスタとなる。

Step4. 保存したクラスタの番号から最小の 2 乗距離を持つクラスタ重心を選択する。

$$d_{pos} = \min_{j \in A} \|\mathbf{v}_{g,i}^{pos} - \mathbf{v}_{g,j}^{pos}\|^2 \quad (17)$$

Step5. d_{pos} が $\min_{pos}$ 未満の時のクラスタ番号 N_1 を保存する。条件を満たさなかった場合は step0 に戻る。以上のプロセスを経て位置、色、法線について条件を満たしたクラスタ重心を選択できる。

5 実験結果および考察

本実験は静的なデータおよび動的なデータに対して実験を行うことによって、提案手法のトラッキングに関する有効性を検証していく。

5.1 静的なデータに対して

静的なデータの画像を Fig.4(a) に示す。本実験では、トラッキングを行う対象として箱、缶の 2 つの物体を用意している。このデータに対して、GNG-DT を適用し、500 エポックの状態を Fig.4(b) に示す。本実験では、色情報と法線情報による位相構造によりクラスタリングを行っており、法線情報があるため、壁は 2 つにクラスタリングされているが、缶は曲率があり法線ベクトルが一定ではないため一つの対象としてクラスタリングされていない。

この実験において色の変化はクラスタ番号の変化を示すため、色が変わるとトラッキングに失敗している事を意味する。最初は色が大きく変動し、トラッキング出来ていないが、エポック数が上がり、徐々にノードの密度が安定すると、色の変化がかなり抑えられた。それぞれのクラスタにおける、3 つの属性の重心差の変動を次の Fig.5 に、その中でもクラスタ 3 の 1000 エポック



Fig.4 Input data. (a) Original 3D point cloud data. (b) Result of GNG-DT.

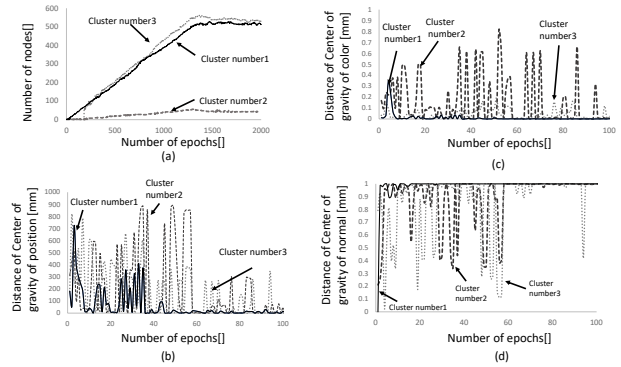


Fig.5 Experimental results of the static data. (a) Number of Nodes, (b) Distance of Center of Gravity (COG) about position, (c) Distance of COG about color, (d) Distance of COG about normal.

までのデータを Fig.6, 7, 8 に示す。これらのクラスタを選択する条件は「1000 エポック数までにクラスタを構成するノードの数が 1 度でもクラスタに含まれるノード数が 20 以上となったクラスタ」とした。つまり Fig.5 は条件を満たした 3 つのクラスタのグラフとなる。Fig.5 では 100 エポックまでのグラフであり、これらを見るとおよそ 250 エポックまでは Fig.6, 7, 8 のようにならなかった。これはノードの数が安定せず、重心位置が大きく変動することが原因と考える。250 エポック以降、それぞれの属性における重心の差は徐々に安定しているが、ノード数の増加が安定し始めるのはエポック数が 1300 以降である。これは今回の 3 つの対象物をトラッキングするのに必要なノード数が必要以上であったからと考える。また、例えば Fig.8 では約 60 エポックから 90 エポックの間で一度安定しているが、そこから一気に値が減少している。これは Fig.5 の (a) でクラスタ番号 3 の急な上昇によるものと考えられる。このことから、ノード数の変動は重心位置に影響することが分かる。

5.2 動的なデータに対して

動的なデータの環境を Fig.9 (a) に、その環境下で入力データを取得した図を Fig.9 (b) に示す。主に 2 つの段ボールと 1 つの箱を用意し、それぞれの対象物および地面に対してクラスタリングを行った。結果は Fig.10 のように二つある段ボールの内、右側の段ボールのクラスタの色は (a) の実験開始 3 秒後、(b) の開始 5

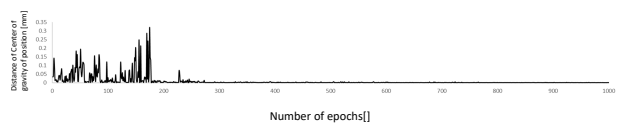


Fig.6 Distance of Center of gravity of position

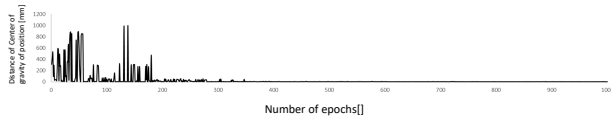


Fig.7 Distance of Center of gravity of color

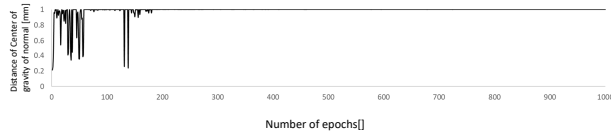


Fig.8 Distance of Center of gravity of normal

秒後で異なっており、トラッキングできなかった。主な原因は動くことによるクラスタを構成するノード数の変動が考えられる。本実験では先ほどとは異なり、1000 エポック数までに同じクラスタ内に 1 度でも 20 以上のノード数を持ったクラスタデータがあまりにも多く存在した。これは一度付けられたクラスタの番号が何度も入れ替わった事が原因と考える。Fig.11, 12 より、処理速度についてはノード数およびクラスタ数の上下があっても 25[msec]に抑えることができた。このように動的環境において重心を使った手法は適用が難しいことからトラッキング手法の改善が必要となると考えられる。

6 結言

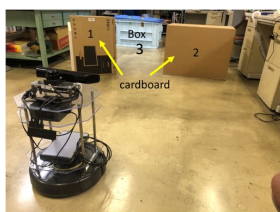
本研究では、対象物のトラッキングを目指した。クラスタの位置、色、法線の重心を用いた手法を用いたが、結果として静的なデータのみ対応できた。動的環境ではノード数の上下があるのに加えて、重心そのものが動くため、トラッキング出来なかった。今後の方針としては重心と重心の比較ではなく、重心とその周りのノードとの比較から最も近いノードを選択することで、重心による影響を最小限に抑えたい。

7 謝辞

本研究は JSPS 科研費 JP20K19894 の助成を受けたものです。

参考文献

- [1] T.Kohonen, "Self-Organizing Maps, Springer," 2000.
- [2] B. Fritzke, "A growing neural gas network learns topologies," Advances in Neural Information Processing Systems 7, pp. 625-632, 1995.
- [3] B. Fritzke, "Growing self-organizing networks - why?," European Symposium on Artificial Neural Networks, pp. 61-72, 1996.
- [4] Y. Toda, W. Chin and N. Kubota, "Unsupervised neural network based topological learning from point clouds

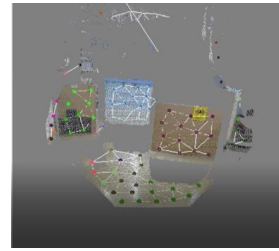


(a)

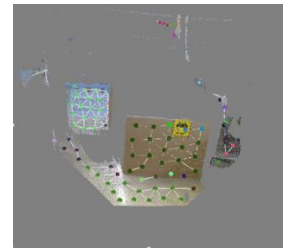


(b)

Fig.9 An example of the experimental data. (a)Experimental environment. (b)3D point cloud data



(a)



(b)

Fig.10 Experimental results for each time. (a)3 [s], (b)5 [s]

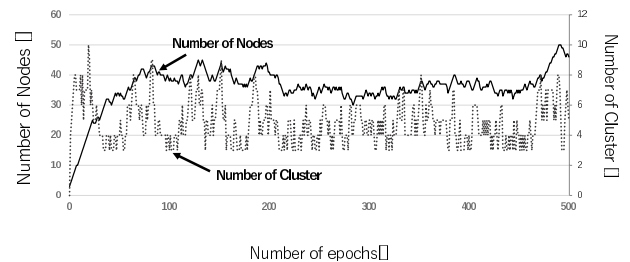


Fig.11 Number of nodes and clusters

for map building," 2017 International Symposium on Micro-NanoMechatronics and Human Science (MHS), Nagoya, Japan, 2017.

- [5] Y. Toda, H. Yu, Z. Ju, N. Takesue, K. Wada, N. Kubota, "Real-time 3D Point Cloud Segmentation using Growing Neural Gas with Utility," The 9th International Conference on Human System Interaction, pp.418-422, 2016.
- [6] 戸田雄一郎, 和田亮雅, 松野隆幸, 見浪護, "複数属性から構成される特徴ベクトルにおける Growing Neural Gas に基づく空間構造の学習", 計測自動制御学会論文集, (published), 2021.
- [7] Y. Toda and N. Kubota. "Self-localization Based on Multi-resolution Map for Remote Control of Multiple Mobile Robots", Industrial Informatics, IEEE Transactions on, Vol.9, No.3, pp. 1772-1781, 2013.

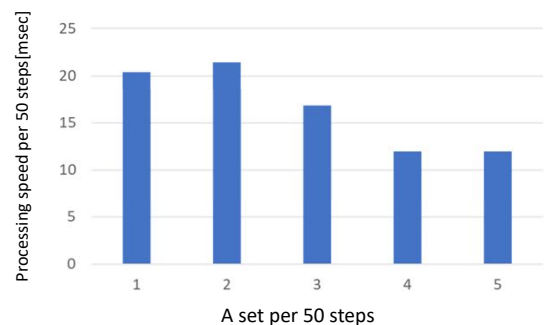


Fig.12 Clustering speed