

Growing Neural Gas を用いた3次元形状認識のための輪郭検出手法の検討

Study of Contour Detection for Growing Neural Gas based 3D Shape Recognition

吉田 優泰¹, ○戸田 雄一郎¹, 松野 隆幸¹

Yudai Furuta¹, ○Yuichiro Toda¹, Takayuki Matsuno¹

¹岡山大学

¹Okayama University

Abstract: This paper presents the algorithm of extracting contour nodes with the use of self-organising networks, which is called Growing Neural Gas (GNG). The system imitates how to get the contour nodes in binary image. And, this algorithm also includes getting contour nodes with angle information for dealing with GNG. Therefore, algorithm consists of their two methods. The proposed method can get the contour nodes exactly compared with advance literature.

1 緒言

近年、様々な分野において自律的に判断し、行動する自律型ロボットの期待が高まっている。このようなロボットにとって、3次元空間に存在する対象を認識することは、移動や把持など様々なタスクを遂行するためには、必要不可欠な能力となる。特に、未知環境においては、未知の対象をロボットに搭載されたセンサにより計測し、カメラ画像や3次元点群からクラスタリングする必要があり、このようなクラスタリング手法として、我々はこれまで、Growing Neural Gas (GNG)に基づく未知空間の知覚手法をこれまでに提案してきた[1, 2, 3]。これらの提案手法を用いることにより、ロボットは未知のデータから対象となるオブジェクトを分割することが可能となるが、クラスタリングされた物体に対して把持などの何かしらの作業を行うためには、物体の輪郭情報の検出などの形状に関する知覚能力が求められる。

そこで本稿では、先行研究におけるGNGに基づくロボットの知覚システムを発展させるため、GNGで構築された位相構造からの輪郭検出を研究目的とする。このような、未知物体に対する輪郭検出は、ロボットが物体の概形を認識するうえで必要となる情報になる。戸田らは輪郭検出に関してGNGにより学習された位相構造から輪郭となるノードの推定手法をこれまでに提案している[3]。しかしこのアルゴリズムでは、検出対象のノードの近傍ノード間のなす角に基づき検出を行っているため、輪郭部分を検出できない事例や誤検出が存在していた。したがってこのアルゴリズムでは、物体の把持における把持点の検出などの正確な輪郭情報が求められるタスクにおいては適用が難しいと考えられる。そこで本稿では、画像処理技術によりこれまでに提案してきた二値化画像における輪郭検出の手法[4]をもとに、GNGに適用可能な新たな輪郭検出アルゴリズムを提案する。さらに、2次元の点群データにおいて、提案

手法と従来手法の比較を行うことで、その有効性を検証していく。

2 従来の輪郭検出アルゴリズム

ここでは、まず戸田ら[3]が提案した従来のアルゴリズムの概要を述べる。

- Step1. 隣接ノードにおける角度情報を取得する。
- Step2. 取得した角度全ての中で、最大値をもつノードを選定する。
- Step3. その角度がしきい値を超えていれば輪郭とみなす。

ここで、図1に上記アルゴリズムのStep1, Step2の適用例を示す。このように、このアルゴリズムでは、あるノードにおけるすべての隣接ノードとの角度を取得し、その最大値をしきい値にかけすることで輪郭であるかどうかを判断している。また図2にこのアルゴリズムに沿って輪郭検出を行った様子を示す。図2において、緑点はデータ点群、青丸は輪郭として検出されたノード、赤丸はそれ以外のノードを示す。図2から、左上の上部の輪郭ノード等、一部輪郭検出ができていないノードがあることを確認できる。したがって先述した通り、正確な輪郭検出を必要とする際は不備が生じることとなる。そこで次に2次元の画像処理技術において用いられている二値化画像での輪郭検出について説明していく。

3 二値化画像における輪郭検出

以下に、図3に示している二値化画像での輪郭検出アルゴリズムを外側輪郭、内側輪郭に分けてそれぞれ示す。

外側輪郭について

- Step1. ラスター走査を行い、始めの黒のマスの探索

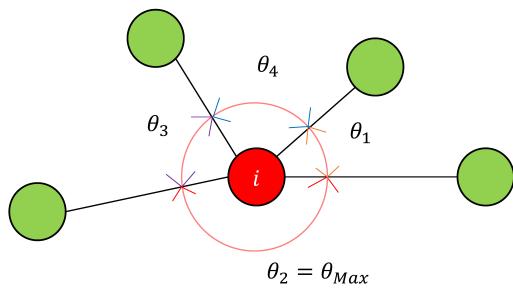


図 1: 輪郭ノードの選出方法

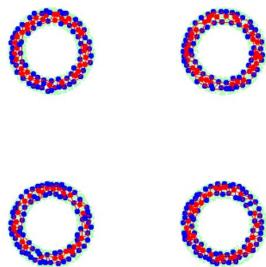


図 2: 従来のアルゴリズムを用いた輪郭検出

- Step2. 黒のマスを見つけたら図4で示す0の方向から反時計回りに探索
 Step3. 輪郭対象を見つけたら次はその点で以下の式を用いてその方向から反時計回りに探索開始.

$$V_{new} = (V_{old} + 6) \bmod 8 \quad (1)$$

なお、 V_{old} は探索しているマスの1つ前のマスから探索しているマスへの方向を、 V_{new} は探索しているマスにおいて次の探索開始方向をそれぞれ示している。

- step4. step3を繰り返し、初期輪郭が再度検出されれば終了

外側輪郭では、まず初めにstep1で、二値化画像の左上から右下にかけて1つずつマスを見していくラスター走査を実行している。ここで一番最初に見つかる黒のマスは、ある物体の輪郭であることは自明であるため、そのマスを初期輪郭とする。図3では二3のマスがこれにあたる。step2では、その点の左下から反時計回りに探索することを実行している。ここで左下から始めるのはラスター走査によって初期輪郭の左のマスはもう探索済みであるためである。この操作により次の輪郭点が決まれば、次はstep3に移る。ここでは見つけた輪郭点で式(1)により次の方向を決め、進む操作をしている。例えば図3での、初期輪郭マス二3の後の輪郭マス三2では、式1を用いると、 $V_{old} = 0$ で $V_{new} = 6$ となるため、

次は輪郭マス三2の左上から反時計回りに探索することになる。

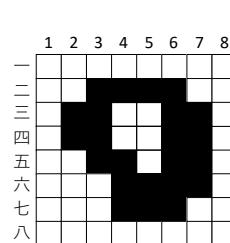


図 3: 二値化画像

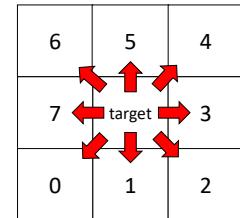


図 4: 探索開始方向の数値化

次に内側輪郭について述べる。

内側輪郭について

- step1. ラスター走査を行い、輪郭判断がされておらず4連結近傍で白色のマスがある黒色のマスを内側輪郭の最初のマスとする。
 step2. 図4で示す2の方向から反時計回りに探索
 step3. 輪郭対象を見つけたら、次はその点で式(1)を用いてその方向から反時計回りに探索開始
 step4. step3を繰り返し、初期輪郭が再度検出されれば終了

内側の輪郭検出でも、黒色のマスを探索していくため反時計回りで探索していく必要がある。例えば図3では内側初期輪郭が三3であり、まず初めは2の方向から反時計回りで探索を行うため、次の輪郭マスが二4となる。式(1)を用いると、次の方向は $V_{old} = 4$ で $V_{new} = 2$ となるため、次は2の方向から反時計回りに探索することになる。このようにして二値化画像では外側および内側の輪郭探索が可能となる。

4 GNGにおける輪郭検出

GNGにおける位相構造は、画像のようにきれいな格子状にはなっていないため、二値化画像とは別のアプローチが必要となる。そこで本稿では、GNGで用いられていた従来のアルゴリズムと二値化画像における輪郭検出の2つを組み合わせた輪郭検出アルゴリズムを提案する。GNGで用いる場合は、隣接ノードにおける角度情報を用いることができるため、二値化画像における方向決め(図4)を角度で表すこととする。ここで図5aに、二値化画像で用いた方向決めの配列とGNGで用いるために角度変換させた配列をそれぞれ示す。また、あるクラスタのノードを図5bに示す座標空間でy方向について並べ替えさせることで擬似的なラスター走

査も可能となる。これにより GNG でも二値化画像の輪郭検出のようにふるまうことができる。以下に GNG における輪郭検出のアルゴリズムを外側と内側に分けてそれぞれ示す。

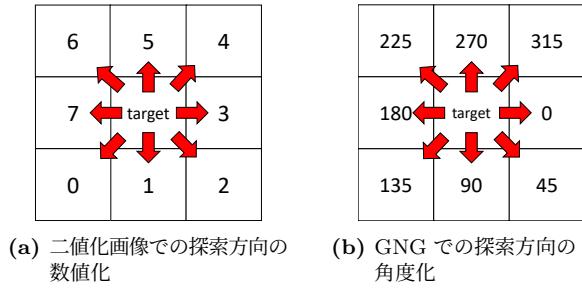


図 5: 探索方向の数値

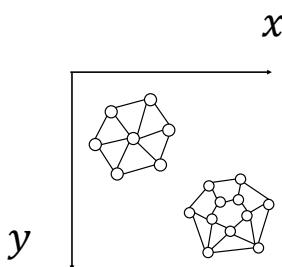


図 6: GNG の座標空間

外側輪郭について

- step1. あるクラスタの中のノード座標情報を y 方向にソートする。
- step2. y_{min} を有しているノード N_s を初めのターゲットノード N_s とし、探索開始方向 V を 180° として、反時計回りに探索を行っていく。
- step3. ターゲットノードに隣接しているノードの角度情報を全て調べ、探索開始方向から反時計回り方向に対して一番近い角度を有しているノードを次のターゲットノードとする。
- step4. 以下の式を用いて次の V を算出する。

$$V_{new} = (V_{old} - 225) \bmod 360 \quad (2)$$

- step5. step3, step4 を繰り返し、再度 N_s が検出されれば終了する。

外側輪郭ではまず、あるクラスタの中のノード座標情報を y 方向にソートし、二値化画像と同じように輪郭検出を行っていく。初期ターゲットノード N_s が再度検出されれば外側輪郭アルゴリズムが終了し、内側輪郭へ移行する。また図 7 に step4 で述べたターゲットノード

の探索の様子について示す。step4 では図 7 のように、ターゲットノードと隣接ノードの角度情報を用いて次のターゲットノードを探している。例えば図 7 では、次のターゲットノードは start direction から反時計回り方向に一番近い 2 となる。次に内側輪郭について述べる。

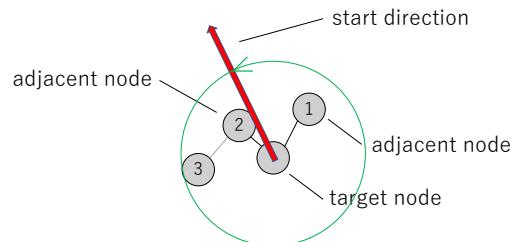


図 7: GNG での輪郭検出

内側輪郭について

- step1. $y_{min} \rightarrow y_{Max}$ ヘラス走査を行っていく。
- step2. すでに輪郭判断がなされているノードであれば 1 へ戻る。まだ輪郭判断がなされていなければ、そのノードを初期ターゲットノード N_s として 3 に進む。
- step3. 以下の式を用いて、ターゲットノードと隣接ノードの角度 θ_{xy} の最大値 θ_{Max} がしきい値 θ_{thv} 以上であれば内側輪郭とみなし、4 に進む。そうでないときは 1 へ戻る。なお、本稿では $\theta_{thv} = 135.0$ とした。

$$T_n = \begin{cases} 1 & \text{if } \theta_{Max} \geq \theta_{thv} \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

$$\theta_{Max} = \max_k \theta_k$$

- step4. 初めは探索開始方向 V を、step3 で発見したターゲットノードと隣接ノード t の角度として探索を開始する。
- step5. ターゲットノードに隣接しているノードの角度情報を全て調べ、 V から反時計回り方向に對して一番近い角度を有しているノードを次のターゲットノードとする。
- step6. 式 (2) を用いて次の探索開始方向を算出する。
- step7. step5, step6 を繰り返し、 N_s が再度検出されれば終了。すべてのノードが輪郭判断されるまで続行する。

GNG における内側輪郭では、初めに輪郭開始点を見つけるため、式 (3) を用いて隣接ノードとの角度における

るしきい値判定を行う。条件に合致するノードが発見されれば、外側輪郭と同様にして探索を行い、次の探索開始方向を式(2)を用いて行う。

5 実験

5.1 シミュレーション実験

ここでは、先述した戸田ら[3]のアルゴリズムと今回の提案手法であるアルゴリズムを2次元データ内で比較することとする。図8に今回使用する2次元データを示す。GNGではこの点群データセットからランダムにデータ抽出を行い入力ベクトルとする。この入力ベクトルを用いてGNGにより位相構造を学習していく。

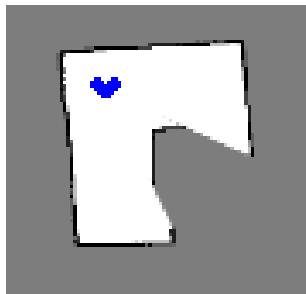


図8: 実験で使用する2次元データ

5.1.1 従来のアルゴリズムを用いた輪郭検出

図9に先行研究のアルゴリズムを用いて輪郭を描画した様子を示す。図9(a)-(c)から中央の輪郭において輪郭が不足していることを読み取ることができる。これはアルゴリズムの性質上、角度のしきい値を超えられなかったり、未知のエラーが起こったりすることによって生じることが考えられる。

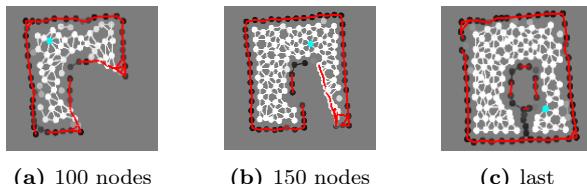


図9: 従来のアルゴリズムを用いた輪郭検出

5.1.2 提案手法を用いた輪郭検出

続いて図10に提案手法を用いた輪郭検出の様子を示す。図10から、外側内側それぞれ1本の線で正確に輪郭を描画できていることがわかる。ただし、輪郭が正しく描画されずエラーが生じることがある。この原因の1つとして、外側の輪郭が正しく描画されずそのまま内側輪郭モードに移行した際、隣接ノードとの角度の判定で正しく判定されなかった外側の輪郭がヒットしてしま

い、内側輪郭の描画がそのノードから始まってしまうこと等があげられる。これを防ぐためには外側輪郭検出のアルゴリズムの改良および軽量化が必要であると思われる。

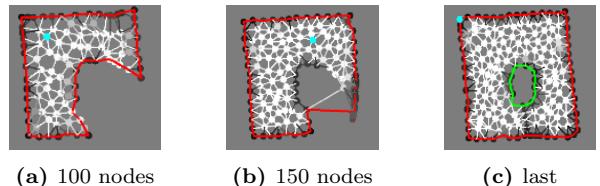


図10: 提案手法を用いた輪郭検出

5.1.3 2つのアルゴリズムの比較

図11に先述した2次元データにおける輪郭検出にかかる時間とターゲットノード数に関するグラフを示す。外側輪郭を形成するノード数を53個と定義すると、図11から提案手法は従来のアルゴリズムと比べて外側輪郭を認識するために1.862倍の時間がかかっていることが読み取られる。これは従来のアルゴリズムに二値化画像における輪郭検出の方法を組み込んだことにより自明である。また赤色で示した提案手法の輪郭ノード推移をみると、値が散乱していることが確認できる。これはアルゴリズム実装上の欠陥により輪郭が検出できなかった際に現れると考えられる。これより、提案手法は従来の手法と比べて時間および安定性に関して劣るとみられる。したがって表1のようにそれぞれのアルゴリズムの特性について表すことができる。

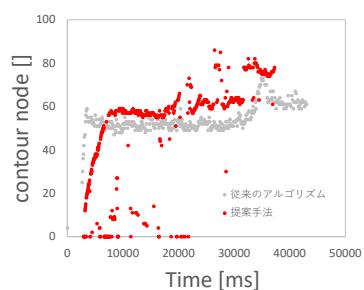


図11: 時間にによる輪郭ノード数の違い

表1: 各アルゴリズムの特性

	従来手法	提案手法
長所	時間、安定性	正確性
短所	正確性	時間、安定性

6 考察

提案手法により、輪郭検出における正確性の向上が見て取れた。しかし、従来の手法よりも外側輪郭を認識するまでに 1.862 倍の時間がかかるうえ、図 11 からわかる通り、輪郭が点滅する等安定性の低下も見られた。原因としては、アルゴリズム実装上の欠陥や大量の計算により内部で予期せぬエラーが起きたことが原因であると考えられる。しかしながら、当初の目標である正確性の向上は確認できたため、まずは 2 次元点群データにおいて、安定性の向上および動作を軽量化する必要がある。また、アルゴリズムに関しては、今回内側輪郭においてすべてのノードを検索して隣接ノードとの角度情報から内側輪郭のスタートを決める手法をとっていた。しかし、この手法では偶然大きな穴がノード空間に生じてしまった場合、無限ループを起こす原因にもなりかねない。そのため、エラー処理をより深く考察することで安定性の向上がみられるとも感じた。

7 結言

今回輪郭検出の正確性を向上させるために、先行研究と二値化画像での手法を組み合わせて新しいアルゴリズムを提案し、2 次元点群データを用いた実験により、その有効性を検証した。その結果、正確性の向上はみられたが、時間と安定性に関しては問題点がみられた。特に、計算時間に関しては、2 次元空間で処理時間の差が顕著にみられたため、3 次元空間へ拡張するとさらに差が開くと考えられる。そこで今後の課題として、アルゴリズムを見直すことによる計算時間の削減があげられる。さらに、3 次元点群データにおいても適用することで、提案手法のさらなる有効性を検証していく予定である。

8 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] Yuichiro Toda, Akimasa Wada, Hikari Miyase, Koki Ozasa, Takayuki Matsuno, Mamoru Mi-nami, “Growing Neural Gas with Different Topologies for 3D Space Perception,” Applied Sciences, Vol. 12, No. 3, 1705, 2022.
- [2] Yuichiro Toda, Takayuki Matsuno, Mamoru Mi-nami, “Multilayer Batch Learning Growing Neural Gas for Learning Multiscale Topologies,” Journal of Advanced Computational Intelligence and Intelligent Informatics, Vol. 25, No. 6, pp.1011-1023, 2021.
- [3] 戸田雄一郎ほか: Growing Neural Gas に基づく環境のトポロジカルマップの構築と未知環境における経路計画, 知能と情報, Vol. 33, No. 4, pp. 872-884, 2021
- [4] イメージングソリューション: ラベリング | イメージングソリューション, 2012.09.02 更新, <https://imagingsolution.net/imaging/labelling/>, (2022.06.17)

連絡先

岡山大学工学部 機械システム系学科システム工学コース
適応学習システム制御学研究室 所属
古田 優泰
E-mail: phxv9301@s.okayama-u.ac.jp