

機械学習を利用したメッシュ地図生成とビジュアルローカリゼーション

ロボット工学研究室 指導教員 黒田洋司
学籍番号 153R173013 4年6組17番 河合響

I. 緒言

本研究では、単眼カメラによる自己位置推定の精度向上を目的として、セマンティックな情報をメッシュ地図生成と自己位置推定の両方に導入する新しい手法を提案する。

ビジュアルローカリゼーションとはカメラからの画像と似た景色を地図から探してマッチングする自己位置推定の一環であり、ロボティクスにおいて広く利用されている。従来手法では SIFT や SURF などの特徴子を地図生成と自己位置推定の両方に導入してきた [1]。しかし、このような特徴子の抽出は季節や時間による環境の変化など光学的な条件に大きく左右され正確な自己位置推定の妨げとなっていた。この問題に対して Stenborg らの先行研究 [2] は機械学習の一種であるセマンティックセグメンテーションを三次元点群の地図生成と自己位置推定の両方に取り入れることで問題解決を図った。豊富な環境で学習したデータを用いることで特徴量ではなく物体としての認識が可能になり、先行研究では季節変化に対してロバストな自己位置推定を実現した。セマンティックな情報を地図に付与することは磁気マーカなどの自己位置推定で使用する人工物を設置する必要があるといった利点を生む。しかし、先行研究ではスパースな三次元点群地図を用いているため実用化する上での大きな問題を抱えていた。本研究ではそれらの問題を解決した自己位置推定手法を提案することを目的とする。

また、本研究で用いたプログラムは Github にて公開している (https://github.com/amslabtech/semantic_mesh_localization)。

II. 関連研究

Stenborg ら [2] は三次元点群地図を構成する点を車や建物などクラスごとに色分けした地図を生成。自己位置推定ではセグメンテーションされたカメラからの画像とカメラの存在し得る位置から見た点群地図の風景を画像化してマッチング、マッチング時のスコアの高かった位置を参考に現在位置を推定するという手法をとっていた。しかし、ある地点から見た三次元点群地図の風景を画像化した場合、近くにある物体は点群の特性によって物体の幾何学的形状が画像に現れず、本来なら物体の裏側にあり隠されていた建物などを構成する点が画像に現れてくる。自己位置推定においてこのような点はマッチング時の大きな妨げになり、多数の物体が存在する環境下での自己位置推定を困難にする。本研究の主な貢献は三次元点群地図の代わりに三次元メッシュ地図を用いることで先述の問題を解決し、多数の物体が存在する環境下での破綻しない自己位置推定を可能にした点である。

III. メッシュ地図生成と提案手法による自己位置推定

A. メッシュ地図生成

ラベル付きの三次元点群データの取得には Semantic KITTI dataset [3] を利用した。三次元点群地図を生成後、PCL の Greedy Triangulation [4] を用いて三次元メッシュ地図を生成した。生成された 2 種類の地図は Fig 1. のとおりである。

B. セグメンテーション手法

カメラ画像のセグメンテーションには Bonnet [5] を使用した。Bonnet はリアルタイムでのセグメンテーションが可能であり ROS 上で使用が可能であることから本研究のセグメンテーション手

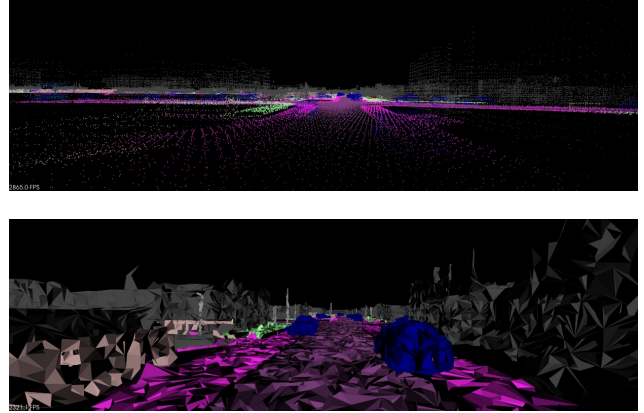


Fig. 1. 生成された点群地図 (上) とメッシュ地図 (下)



Fig. 2. カメラ画像 (上) とセグメンテーションされた画像 (下)

法として選出された。セグメンテーションの様子は Fig 2. のとおりである。

C. メッシュ地図による自己位置推定

自己位置推定の処理の全体の流れを Alg.1 に示す。本手法の自己位置推定手法はパーティクルフィルタ [6] をもとにしたものであり、Alg 1. の (1) の処理であるパーティクルの尤度算出の部分が本研究の特徴的な箇所である。また、(2) のリサンプリングのアルゴリズムには系統的リサンプリング [7] を用いた。

尤度算出の処理を Alg 2. に示す。Alg.2 において、パーティクルの位置から見たメッシュ地図の風景を画像化、セグメンテーションされたカメラからの画像とピクセル単位で比較して、クラスが一致していた場合はクラスの種類に応じたスコアが加算される (Alg.2 中の (3))。

IV. 実験

A. 実験条件

実験にあたっては Semantic KITTI dataset の中から自己位置推定の検証で使用可能な約 8 分ほどの長さの 4 つの sequence を利

Algorithm 1: Semantic Mesh Localization Algorithm

```

initialize particles  $x_0$  and weights  $w_0$ ;
for each time instance  $t$  do
    acquire segmented image  $y_t$ ;
    motion update;
    for number of particles do
        project 3D mesh map to 2D image in the view of
        particle  $i$  as image  $f_t^i$ ;
    end
    measurement update (1);
    normalize weights  $w_t$ ;
    estimate current pose  $x_t$ ;
    if resampling is needed then
        resampling(2);
    end
end

```

Algorithm 2: Calculating Likelihood Algorithms

```

get segmented image as  $y_t$ ;
get mesh map image as  $f_t^i$ ;
 $w_t^i = 0.0$ ;
if The sizes of the  $y_t$  and  $f_t^i$  images match then
    for  $j = 0$ ;  $j < \text{size of image's height}$  do
        for  $i = 0$ ;  $i < \text{size of image's width}$  do
            mesh class = semantic class of image  $y_t$  at
            coordinate (i,j);
            image class = semantic class of image  $f_t^i$  at
            coordinate(i,j);
            if mesh class == image class then
                 $w_t^i = w_t^i + \text{semantic score}(\text{image class})$  (3);
            end
        end
    end
end
return  $w_t^i$ ;

```

用して地図生成, 自己位置推定を行った。

実験にあたって, 比較手法として先行研究 [2] のアルゴリズムを再現実装して提案手法と同じ sequence で自己位置推定を行った。

B. 実験結果

Semantic KITTI dataset の 4 つの sequence において提案手法, 先行研究の 2 つの手法で自己位置推定を行い, 各ループ処理において Ground Truth との推定された XYZ 座標の差, 及び RPY 方向の角度の差を記録した。その結果を Tab.I と Tab.II に示す。

先行研究についてはすべての sequence において自己位置推定が完全に破綻した。これは II 節で述べた「三次元点群地図の風景を画像化した場合, 近くにある物体の幾何学的形状が画像に現れず, 物体の裏側に隠されていた建物などを構成する点が画像に現れる」問題が起きたためである。車両の近くにあるランドマークとなる物体を利用できず, 加えて他の物体によって隠されている物体を構成する点が点群地図を画像化した際に現れ, 適切な尤度算出において妨げになっていたことが自己位置推定の破綻につながったと考えられる。先行研究 [2] の論文内では優れた自己位置推定の精度を示していたが, 本実験においてはそれは認められなかった。

提案手法に関しては sequence02 を除いて自己位置推定が破綻することなく安定した動作を見せた。これはランドマークとなる物体を自己位置推定の時に正確に認識できたため, 進行方向に対して横方向の自己位置推定の精度を確保できたためであると考

TABLE I
LOCALIZATION ERROR

sequence	Proposed Method				Conventional Method			
	00	01	02	03	00	01	02	03
Error Frequency [%]								
<0.5m	1.5	1.5	0.7	1.2	0.7	0.0	0.0	0.0
<1.0m	12.6	8.4	2.7	6.5	0.0	0.4	1.8	0.4
<1.5m	12.6	20.1	10.9	12.7	2.6	2.3	6.1	0.0
<2.0m	18.4	20.1	12.4	13.5	4.6	2.7	9.1	1.6
<2.5m	14.1	17.9	11.5	17.6	9.8	6.1	3.0	4.1
<3.0m	12.1	10.6	10.2	18.0	12.4	3.4	8.5	3.7
<3.5m	5.3	6.1	2.7	12.2	15.7	4.2	8.5	5.7
<4.0m	8.7	3.8	4.1	6.9	13.1	7.2	9.1	2.8
4.0m<	14.6	10.6	44.9	11.4	41.2	73.9	53.9	81.7

TABLE II
ANGLE ESTIMATION ERROR

sequence	Proposed Method				Conventional Method			
	00	01	02	03	00	01	02	03
Error Frequency [%]								
<1.0[deg]	38.3	34.6	36.1	22.9	30.1	33.0	22.6	16.7
<5.0[deg]	61.2	26.2	61.2	51.4	66.0	57.6	24.8	18.7
<10.0[deg]	0.5	37.3	2.7	25.3	3.9	9.5	41.8	0.8
<15.0[deg]	0.0	1.9	0.0	0.4	0.0	0.0	9.7	1.2
<20.0[deg]	0.0	0.0	0.0	0.0	0.0	0.0	0.0	5.7
<25.0[deg]	0.0	0.0	0.0	0.0	0.0	0.0	0.0	50.0
<30.0[deg]	0.0	0.0	0.0	0.0	0.0	0.0	0.0	3.3
30.0[deg]<	0.0	0.0	0.0	0.0	0.0	0.0	0.0	3.7

える。一方, 本手法は単眼カメラのみでの自己位置推定であるため 3 次元空間の情報を入力することができない。そのためランドマーク的な物体の少ない広い直線道路などでは推定される位置が Ground Truth と比較してズレが大きくなった, sequence02 において良好でない結果が得られたのはこのためであると考えられる。

角度推定に関しては, 提案手法は概ね良好な結果が得られたが, 先行研究に対して大きな優位性を持つわけではないことがわかった。

V. 結言

提案手法により, 点群地図特有の尤度算出の問題が解決した。また, 提案手法が先行研究と比べて自己位置推定が破綻しないなどの良好な結果を示したことにより本手法の有用性を示すことができた。

その一方, 実験に使用したソフトウェアが提案手法のような使い方と相性が悪く, メッシュの影の描画など本手法においては不要な処理が数多く行われ大きな負担となった。このため現状ではリアルタイムの推定はできない。処理時間が今後の課題である。

REFERENCES

- [1] Jorge Fuentes-Pacheco, Jose Ascencio, and J. Rendon-Mancha. Visual simultaneous localization and mapping: A survey. *Artificial Intelligence Review*, Vol. 43, , 11 2015.
- [2] Stenborg Erik, Toft Carl, and Hammarstand Lars. Long-term visual localization using semantic segmented images, 2018.
- [3] Behley Jens, Garbade Martin, Milioto Andres, Quenzel Jan, Behnke Sven, Stachniss Cyrill, and Gall Juergen. Semantickitti: A dataset for semantic scene understanding of lidar sequences, 2019.
- [4] <https://pcl.readthedocs.io/projects/tutorials/en/latest/greedyprojection.html>.
- [5] A. Milioto and C. Stachniss. Bonnet: An Open-Source Training and Deployment Framework for Semantic Segmentation in Robotics using CNNs. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2019.
- [6] Dieter Fox, Wolfram Burgard, Frank Dellaert, and Sebastian Thrun. Monte carlo localization: Efficient position estimation for mobile robots. pp. 343–349, 01 1999.
- [7] 上田隆一. 詳解 確率ロボティクス Python による基礎アルゴリズムの実装, pp. 131–132. 講談社, 2019.