

つくばチャレンジ 2022 における 千葉工業大学未来ロボティクス学科チームの取り組み

筑波 太郎 ^{†1}, 筑波 花子 ^{†1}

Development Activity of Advanced Robotics Department Team of Chiba Institute of Technology at Tsukuba Challenge 2021

*Taro TSUKUBA ^{‡1}, Hanako TSUKUBA ^{‡1}

千葉工業大学未来ロボティクス学科チーム, box, box2

Abstract— In this paper, we introduce the activities of the Future Robotics Department, Advanced Engineering Division, Chiba Institute of Technology team in the Tsukuba Challenge 2022. Our team is working on the development of an outdoor autonomous mobile robot, and we are currently tackling several challenges. For example, the development of a new robot, driving using semantic segmentation, and recovery of kidnapped state by combining expansion resetting and GNSS resetting. We will report the results of these efforts.

1. はじめに

本チームは、屋外で安定して自律移動するロボットを目指し、その研究および開発の一環としてつくばチャレンジに参加している。開発したシステムは 2 次元地図と測域センサを用いた自己位置推定により、つくばチャレンジ 2016, 2017 においてマイルストーン 3「横断歩道区間を含まない課題コース (2037m)」を達成した。しかし、つくばチャレンジ 2018 から、ひらけた公園がコースに含まれるようになった。そのため、30m 程度の 2 次元レーザレンジセンサでは検出可能な物体が比較的少ないことから、従来のシステムでは安定して自己位置推定を行うことが困難になった。2018 年度から計測距離が 100m の 3 次元レーザレンジセンサを採用したが、水平面の計測データのみを使用しており、多くのデータを破棄していた。さらに、公園内では場所により地面の傾斜が変化するため、レーザが必ずしも水平の距離を計測していないという問題もあった。例えば、斜めに土が積み上げられた場所では、レーザを照射する位置が上下に変化すると、それに合わせて距離も変化してしまう。そのため、地図生成が容易に行えないという問題もあった。2019 年度は、これらの 3 次元データを有効に活用するため、2 次元地図ではなく、3 次元地図を生成し MCL による 3 次元自己位置推定を行い、2 次元自己位置推定と比べ、自己位置が安定していることを確認した。しかし、公園のような開けた場所では測域センサで検出できる物体が少なく、測域センサ、オドメトリを用いる方法では自己位置が定まらないことがある。本稿では、このような課題を解決するためにつくばチャレンジ 2022 に向けて取り組んだ開発に関して紹介する。

2. ロボットの概要

つくばチャレンジでは、本チームが開発を続けている三台のロボット ORNE- , ORNE-box, ORNE-box2 を用いる。それぞれの方針は以下の通りである。

- ORNE-
2 つの走行の切り替えによる安定した自律走行
- ORNE-box, ORNE-box2

2.1 ハードウェア

本チームは屋外自律移動ロボットとして、ORNE- , ORNE-box, ORNE-box2 の開発を行っており、つくばチャレンジ 2022 にはこれらのロボットが参加する。Fig. 1 に本チームの開発している自律移動ロボットの外観を示す。これらのロボットは i-Cart middle をベースとしており、主なセンサは IMU, 測域センサである。

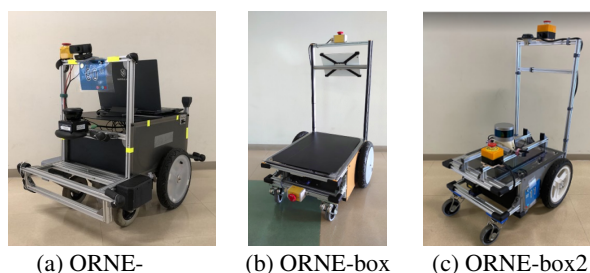


Fig. 1: ORNE-Series

Table 1: Specifications of the robots

| | ORNE- | ORNE-box | ORNE-box2 |
|--------------------|--------------------------------------|--------------------------|----------------------|
| Depth[mm] | 690 | | 600 |
| Wide[mm] | 560 | | 506.5 |
| Height[mm] | 770 | | 957 |
| Wheel diameter[mm] | 304 | | |
| Battery | LONG WP12-12 | | |
| Motor | Oriental motor TF-M30-24-3500-G15L/R | | |
| Driving system | Power wheeled steering | | |
| 2D-LiDAR | URM-40LC-EW (HOKUYO) | None | UTM-30LX-EW (HOKUYO) |
| 3D-LiDAR | None | R-fans-16 (SureStar) | VLP-16 (Velodyne) |
| IMU | ADIS16465 (Analog devices) | ADIS16475 Analog devices | |
| GNSS receiver | None | u-blox SCR-u2t | |
| Camera | CMS-V43BK (Sanwa supply) | None | |

2.2 ソフトウェア

本チームでは、従来より ROS(Robot Operating System) の navigation stack [1] をもとに開発されたシステムである orne_navigation により自律走行させている。Fig. 2 に開発しているロボットのソフトウェアを含むシステム構成を示す。このシステムは、2D-LiDAR を用いた Monte Carlo Localization(MCL) により確率的に自己位置を推定し、経路計画に基づいて自律走行している。また、GitHub の open-rc [2] でプログラムを公開している。

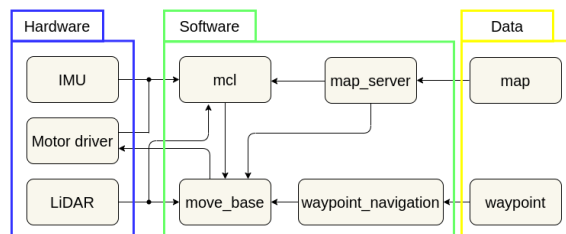


Fig. 2: Structure of the system.

3. 各チームの方針

本チームには、チーム ORNE- , ORNE-box, ORNE-box2 の 3 チームが存在する。従って本章では、各チームごとの活動方針を述べる。ただし、ORNE-box2 は ORNE-box の後継機であるため、活動方針は同じである。

3.1 チーム ORNE-

2D-LiDAR ベースの自律走行時、自己位置推定の結果が不確かになる場合がある。この状態での走行はリタイアの要因の一つになる可能性がある。そこで、チーム ORNE- は、2D-LiDAR ベースの自律走行と機械学習を用いた自律走行の切り替えによる安定した走行を目的としている。昨年度は、orne_navigation による自律走行時、自己位置推定の尤度が低下した場合に、カメラ画像を入力とした end-to-end 学習器を用いた自律走行による切り替えを行った。しかし、意図しない箇所ではカメラを用いた走行へ切り替えが起きてしまうことがあった。そのため、本年度はそれらの問題を解決するために、取り組んだ内容に関して以下で紹介する。

3.1.1 提案手法

提案手法を Fig. 3 に示す。移動ロボットは、2 つの走行方法を持つ。この 2 つの走行方法は、emcl2[3] の α^* を指標として切り替える。

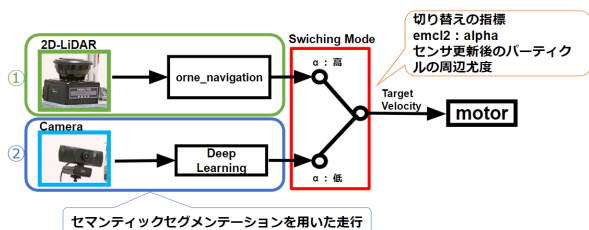


Fig. 3: Developed system of switching action.

- alpha が高い場合
LiDAR ベースのナビゲーション
- alpha が低い場合
セマンティックセグメンテーションを用いた走行

3.1.2 2D-LiDAR ベースの自律走行時について

- 外界センサ: 2DLiDAR
- 自己位置推定: emcl2
- global planner: A*
- local planner: dwa_local_planner
- 地図作成: cartographer
 - resolution: 0.15[m/pixel]
 - 確認走行、駅周辺、公園の 3 つに分割したものを合成

3.1.3 セマンティックセグメンテーションについて

フレームワーク: DeeplabV3 mobilenet

3.1.4 ロボットの制御

セマンティックセグメンテーションを用いて、領域分割を行う。処理を施した例を Fig. 4 に示す。

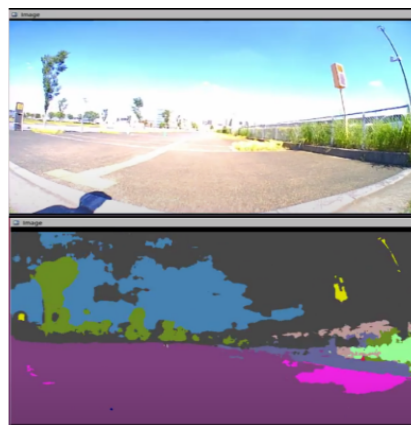


Fig. 4: The top image is the original image, the bottom image is the processed image using semantic segmentation

これらの処理を施した画像を用いて、Fig. 5 の対応する箇所に応じた行動を生成する。具体的には、Fig. 5 の Turn left の範囲に走行可能領域以外のラベルによるピクセル数がある程度多くなると、左に曲がるヨー方向の角速度を出力する。また、GitHub の deeplabv3_plus_pytorch_ros [3] でプログラムとデータセットを公開している。

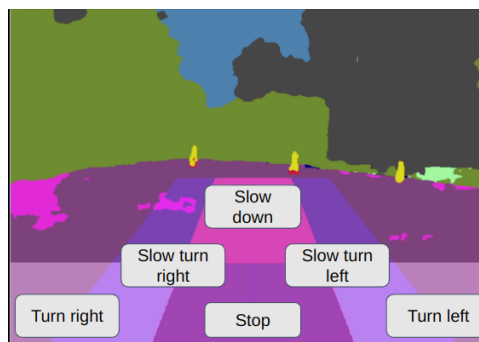


Fig. 5: Types of behavior using semantic segmentation

*1 センサ更新後のパーティクルの周辺尤度

3.1.5 チーム ORNE- の本走行の結果と展望

今年度の本走行の記録は 847.9[m] で、駅構内の手前でリタイアとなった。これは歩行者などによりランドマークが隠され、自己位置が誘拐されるなどの複合的な要因によるものだと考えられる。今後は、自己位置推定に用いるセンサを追加するなどの対策を検討する。

3.2 チーム ORNE-box, ORNE-box2

3.3 本文中での参照

論文に挿入した図は、本文中で必ず参照する。図を参照する際は、以下の例のように `\figref{}` コマンドを用いる。`\figref{}` コマンドは、`jsproceedings.cls` で定義した独自のコマンドである。参照先の図には、`\label{}` コマンドでラベルを付けておく。

LaTeX ソース

```
\figref{fig:matching-concept}に、提案手法の概念  
と各変数の定義を示す。
```

出力

4. 表

本章では、表を挿入する方法、キャプションの書き方や本文中での参照の仕方について述べる。

4.1 Table の挿入

表の挿入には、`table` 環境 (`\begin{table}`, `\end{table}`) と `tabular` 環境 (`\begin{tabular}`, `\end{tabular}`) を用いる。具体的な書き方は LaTeX ソースを参照されたい。

表の罫線は格子状に引く必要はなく、省略できる罫線は引かずに罫線を少なくした表の方が美しい組版と言われる。特に英文の場合、縦罫線は引かない方が良い。

横罫線は、標準の `\hline` コマンドでは上下の間隔が狭く、線の太さのバランスも良くない。

4.2 キャプション

表の上に `\caption{}` コマンドでキャプションを付ける。キャプションを英語で書くか、日本語で書くかを論文中で統一

する。論文を投稿する学会のフォーマットに従うが、和文論文でも英語のキャプションとする場合が多い。図表とキャプションだけを見て論文の内容が類推できるよう、キャプションは単語ではなく文章で書く。よって英語の場合は最初の文字を大文字にし、その後は固有名詞などを除いて小文字にする。また、文末にはピリオドを書く。

4.3 本文中での参照

論文に挿入した表は、本文中で必ず参照する。表を参照する際は、以下の例のように `\tableref{}` コマンドを用いる。`\tableref{}` コマンドは、`jsproceedings.cls` で定義した独自のコマンドである。参照先の表には、`\label{}` コマンドでラベルを付けておく。

LaTeX ソース

```
\tableref{table:corr-dist}に、マッチング後の対応  
点間距離の平均値を示す。
```

出力

5. おわりに

本稿では、千葉工業大学未来ロボティクス学科チームで開発しているロボットの概要とシステムの構成に関して述べた。また、つくばチャレンジ 2022 に向けた取り組みについて紹介した。

参考文献

- [1] ros-planning, navigation レポジトリ
<https://github.com/ros-planning/navigation> (最終閲覧日:2022 年 12 月 5 日)
- [2] Robot Design and Control Lab, openrdc orne_navigation レポジトリ
https://github.com/open-rdc/orne_navigation (最終閲覧日:2022 年 12 月 5 日)
- [3] deeplabv3_plus_pytorch_ros レポジトリ
https://github.com/Tsumoridesu/deeplabv3_plus_pytorch_ros/tree/add_cmd_vel (最終閲覧日:2022 年 12 月 5 日)