

E-5

Autoware を用いた自動運転システム—仮想環境内でのシミュレーション— —仮想環境内でのシミュレーション— Automatic Operation System Using Autoware — Simulation in Virtual Environment—

○小林文俊¹, 入江寿弘²Fumitoshi Kobayashi¹, Toshihiro Irie²

Abstract: One of the indispensable things in our lives is the car, which is convenient for moving, but we take great danger with one operation mistake. We wanted to eliminate traffic accidents on the road. Eliminate human error. As a means for this, there is automatic driving, which is expected to spread in society in the future. We started research on automatic driving systems because we can automatically control the work performed by humans and reduce human error. In order to realize autonomous driving, it is necessary to respond to road conditions such as various traffic rules and weather conditions on motorways and general roads. Therefore, this time we will consider cooperation with Autoware and automatically use virtual space. A driving simulation was performed.

1. 目的

自動運転を実現させるためには、自動車専用道路や一般道のような交通ルール・天候状況などの道路状況に対応し、安全に行えなければならない。現在は公道実験の段階にあるが、実験で事故が起こったニュースを見かけ、安全に実験を行えないかと考えた。そこで今回は、仮想環境内に地図データを取り込んで走行車線にある縁石や白線を認識し車線の真ん中を走行できるか検証させ、また、現在地を予め設定させ目的地を決めて実際に走行させた際に最短ルートを割り出して走行できるかを検証する。

2. Autoware とは

Autoware は Linux と ROS(Robot Operating System)をベースとした自動運転システム用オープンソースソフトウェアである。LiDAR,カメラ,GNSS(Global Navigation Satellite System)などの環境センサを利用して、自車位置や周囲物体を認識しながら、カーナビから与えられたルート上を自律走行できる。

(1) 自動運転のレベル

自動運転レベル1,2は「運転支援技術」であり、レベル3は自動運転レベルと呼べる。レベル4,5では完全自動運転レベルである。Table 1に自動運転のレベルの表を示す。

Table 1. Level of Autonomous Driving

レベル	システム: ハンドル・アクセル・ブレーキ	ドライバー	場所	用途
レベル1	いずれか一つ	主体	/	自動ブレーキ
レベル2	複数	主体		クルーズコントロール
レベル3	すべて (危険回避はドライバー)	あり		公道実験
レベル4	すべて	なし	限定	自動運転バス
レベル5	すべて	なし	全て	自動運転タクシー

(2) 自動運転ソフトウェア

Autoware が対応しているハードウェアには、CPU・GPS やカメラ,GNSS(Global Navigation Satellite System)・LiDARなどのセンシングデバイスがある。基本的に Autoware のプログラムは、マルチコア CPU 上で動作し、計算量の多い処理では GPU を活用する。

(3) ROS

ROS (Robot Operating System)はソフトウェア開発者のロボット・アプリケーション作成を支援するライブラリとツールを提供している。具体的には、ハードウェア抽象化、デバイスドライバ、ライブラリ、視覚化ツール、メッセージ通信、パッケージ管理などが提供されている。ROS はオープンソースの一つである。

(4) 3次元地図

3次元地図データとは、道路の幅、車線を分ける区画線や路肩の線、信号での停止線、横断歩道、標識などが詳細に記されている。平面ではなく立体的なデータになっているのが特徴である。

3. 今回使用したスペック

OS : Linux(ubuntu 18.04) 64bit 版

GPU : GeForce GTX 1080Ti×2

メモリ : 32GB

ROS : ROS melodic

仮想環境 : NVIDIA Docker2

NVIDIA Driver: 440.82

CUDA: 10.2

1 : 日大理工・院生・精機 2 : 日大理工・教員・精機

4. 今回使用するデータ・地図データ

今回使用するデータは,GitHub の Autoware のページに公開されているデータを使用する.データには, LiDAR とカメラのキャリブレーションファイル,3次元地図,走行経路,world座標系と map 座標系の TF,実際に走行して取得したセンシングデータを含む rosbag データファイルを含む.

今回使用した地図データは台湾の自動車ラボ試験施設を使用した. Figure 1 に示す.

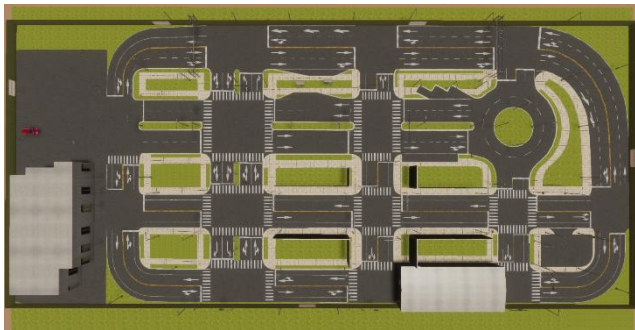


Figure 1. Simulator Destination

5. 実験手順

実験手順の概要を以下に記載

- ①Autoware を実行し, Runtime Manager 内にデータを読み込ませ, ROS の 3次元可視化ツール(RViz)でモデルを表示.
- ②LGSVL シミュレータを実行
- ③使用するマップと車両を設定してシミュレーションを実行
- ④現在地, 目的地を設定

6. 3次元可視化ツール(RViz)及び目的地までの結果

(1) 3次元可視化ツール(RViz)表示結果

ROS の 3次元可視化ツール(RViz)を用いて Figure 1 の地図データを LiDAR やセンサーデータを 3D モデルに表示し,上から見た図を Figure 2 に示す.

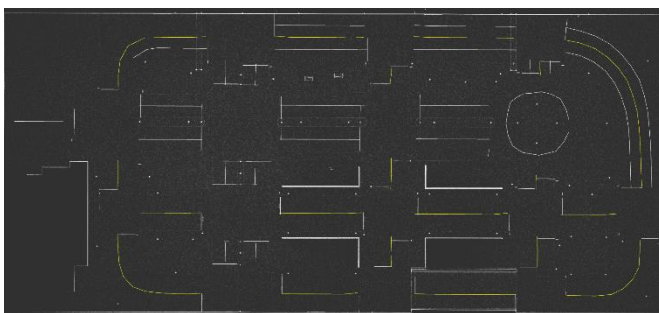


Figure 2. 3D Visualization Tool Display Results

(2) 走行車線内での縁石と白線

自動車から見た走行車線上の縁石や白線の見え方を以下の Figure3 に表す.



Figure3. How the Curb and White Line Look

(3)シミュレータ目的地までの結果

シミュレータを実行して,目的地までの最短経路を表示したのを Figure 4 に示す.出発地から目的地までの最短経路を青い線で示している.

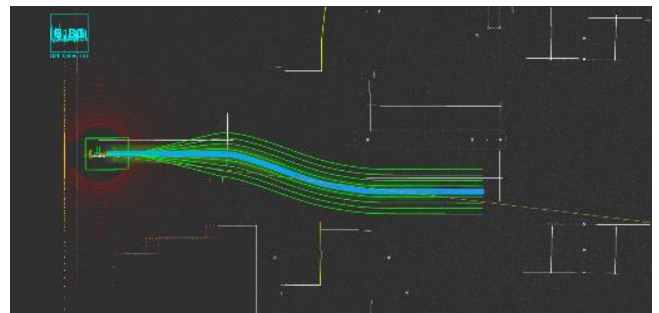


Figure 4. Simulator Destination

7. 結論

仮想環境内に地図データを取り込んで走行車線上にある縁石や白線を認識し車線の真ん中を走行させることができた.また,現在地・目的地を決めて実際に走行させた結果,最短ルート割り出しして走行した.このことから,シミュレーション内で公道実験と同様の実験を安全に行えると結論付けた.

8. 参考文献

- [1] LG Silicon Valley Lab
<https://github.com/lgsvl/autoware-data>
- [2] 著書 加藤真平他 Autoware 自動運転ソフトウェアリックテレコム
- [3] LGSVL Simulator
<https://www.lgsvlsimulator.com/docs/>