

公立はこだて未来大学 2026 年度 システム情報科学実習
グループ報告書

Future University Hakodate 2026 Systems Information Science Practice
Group Report

プロジェクト番号/Project No.

11

プロジェクト名/Project Name

Make Brain Project

グループ名

グループ B

Group Name

Group B

プロジェクトリーダー/Project Leader

角脇輝映 Akira Kadowaki

グループリーダー/Group Leader

太田終生 Syu Ota

グループメンバー/Group Member

大山芳 Kaoru Oyama

佐藤稜隼 Ryoto Sato

指導教員

香取勇一 栗川知己 島内宏和 佐藤直行

Advisor

Yuichi Katori Tomoki Kurikawa Hirokazu Shimauchi Naoyuki Sato

提出日

2026 年 1 月 21 日

Date of Submission

Jul. 21, 2026

概要

近年、画像認識技術や自動運転技術の研究が進展している。一方で、画像認識技術は、標識の認識において、様々な課題を抱えている。具体的には、逆光で標識を認識することなどが大きな課題である。また、車線におけるレーンを認識すること、レーンの判別することも課題である。本プロジェクトでは Raspberry Pi 5 を搭載したラジコンカーを用いて、以下の三つの機能の実装を目的とする。Raspberry Pi 5 とは、本プロジェクトで用いる自動運转向けコンピュータの中で高性能な小型機である。第一に、カメラを使用して標識を認識し、その種類に応じて、車の停止・発進を適切に制御する機能である。画像処理を得意とする Raspberry Pi 5 を用いることで標識を認識する際の逆光についての課題を解決できることが期待できる。第二に、カメラを使用して道路上の線を認識し線を超えないように走行する機能である。また、点線を認識し、障害物の追い越しが可能かどうかを判断する機能を実装する。この機能では走行マップ上に配置した QR コードから現在地を取得し、自律走行を実現する機能である。第三に、LiDAR を使用し障害物を検知し適切に行動をする機能である。LiDAR を使用することで、カメラを使用するよりも、周囲の障害物を高精度ですばやく認識することができる。また、LiDAR は実際の自動運転技術にも使用されるセンサーであるため、本プロジェクトのラジコンカーを実際の自動運転車に近づけることができる。また、第一の機能と第二の機能は、機械学習を用いて実装する。

キーワード 自動運転, 画像処理, 機械学習

Abstract

In recent years, research in image processing and autonomous driving technologies has advanced significantly. However, image processing capabilities face various challenges in recognizing road signs. Recognizing signs in backlight conditions is a major issue. Additionally, distinguishing dashed lines as lines and differentiating solid lines from dashed lines remain challenges for image processing technology. This project aims to implement the following three functions using a radio-controlled car equipped with a Raspberry Pi 5. The Raspberry Pi 5 is a compact computer particularly adept at image processing. First, it involves using a camera to recognize traffic signs and appropriately control the car's stopping and starting based on the sign type. Using the Raspberry Pi 5, which excels at image processing, is expected to resolve challenges related to backlighting when recognizing signs. Second, it involves using a camera to recognize lines on the road and driving without crossing them. It will also implement a function to recognize dotted lines and determine whether overtaking obstacles is possible. This camera will also implement a function to obtain the current location from QR codes printed on the driving map, enabling autonomous driving. Third, it involves using LiDAR to detect obstacles with laser light and take appropriate actions. Using LiDAR enables higher-precision and faster recognition of surrounding obstacles compared to cameras. Furthermore, since LiDAR is a sensor used in actual autonomous driving technology, it brings this project's RC car closer to real autonomous vehicles. Additionally, the first and second functions will be implemented using machine learning.

Keyword autonomous driving, image processing, machine learning

目次

第 1 章	はじめに	1
1.1	背景	1
1.2	研究動機	1
1.3	プロジェクトの目的・重要性	1
第 2 章	関連研究	3
2.1	関連研究	3
2.2	必要なスキル	3
2.2.1	機械学習によるモデルの作成	3
2.2.2	YDLiDAR X2 を使用した障害物検知	4
2.2.3	QR コードを用いた現在地取得	4
第 3 章	プロジェクト学習の最終的な目標	5
3.1	標識認識・レーン認識機能の実装	5
3.2	LiDAR 機能の実装	5
3.3	目的地までの自律走行	5
第 4 章	目的を達成するための手法, 方法	6
4.1	学習モデル	6
4.1.1	標識認識	6
4.1.2	レーン認識	7
4.2	モデルの軽量化	8
4.3	LiDAR による物体検出	8
第 5 章	結果	10
第 6 章	考察	12
	参考文献	13

第 1 章 はじめに

1.1 背景

現在、世界中で自動運転技術の研究開発が急速に進んでいる。交通事故は深刻な社会問題である中、自動運転技術は将来のモビリティ社会において重要な役割を果たす。日本での交通事故発生数は、年々増加している。総合事故総合分析センターによると、2023年の日本の交通事故発生数は、30万7930件であった。これは、前年度の事故件数に比べ、2.4%増加している [1]。事故原因として、標識無視によるなどがあげられる。自動運転車は周囲の環境を正確に認識し、人間と同様に状況判断し、適切な行動をとることで事故を減らすことができる可能性がある。具体的には、標識や車線の認識、障害物の検知を行い、自動運転車の発進や停止を制御することが求められる。これは、今後の自動運転車が直面する重要な課題である。このような課題を解決する手法として画像処理技術の精度の向上がある。画像処理技術の精度を向上させることで、カメラ映像から得られる情報（輪郭・形状・コントラスト・距離情報など）をより正確に解析できるようになる。つまり、標識や実線・点線の認識能力が上がり、発進、停止を行うことができる。また、センサーを用いた障害物検知により、速度の調整や障害物の回避を行うことができる。これらの技術を使うことで、自動運転の安全性を向上させ、社会問題である交通事故に対策を打つことができる。

(※文責：太田)

1.2 研究動機

前年度のプロジェクトである「視覚を持つ AI カー」(公立ほこだて未来大学 2024 年度 システム情報科学実習グループ報告書) という先行研究より、詳細を調査した。その結果、AI カーは、現在地から目的地まで、レーンを認識しながら移動を完了させるものであった。また、標識を認識し、標識通りに動作を行うものであった。そこで本プロジェクトでは、前年度のプロジェクトの成果を踏まえ、機械学習を用いたレーン認識および標識認識の高精度化を図り、前年度のシステムを発展・強化することを目指す。

(※文責：太田)

1.3 プロジェクトの目的・重要性

本プロジェクトの目的は、機械学習を用いて標識・実線/点線の情報をリアルタイムに処理し、発進、停止、線に沿った走行をすることである。カメラを用いて、視覚情報を取得し、処理することで人工知能が統計的推論で、標識を認識したときに停止することや障害物を検知すること、レーンを認識するなどの状況判断をすることができる。これは、自動運転の安全性を向上させ、自動運転車の能力を向上させるために非常に重要である。本プロジェクトの成果が自動運転技術の発展に寄与し、人工知能を利用した交通事故の削減につながることを期待される。

第 2 章 関連研究

2.1 関連研究

本プロジェクトでは、前年度のプロジェクトである「視覚を持つ AI カー」を先行研究として参照する。このプロジェクトでは、作成したマップ上において AI カーが白線や停止線を認識しながら目的地まで自律走行し、目的地到着時に自動で停止するシステムの構築を試みた。その結果、AI カーは概ね安定して走行し、停止動作を適切に行うことが可能であった。また、停止標識の認識についても高精度な結果が得られた。具体的には、停止標識の半分が覆われている場合や、白黒画像の標識に対しても正確に認識を行い、自動停止を実現することができた。それに対し、課題としては大きく三点が挙げられる。第一に、処理負荷や通信環境の影響によりカメラ映像に遅延が生じ、リアルタイムで問題を把握しづらかった。第二に、自動運転カーがラインの中心ではなく端寄りを走行する現象が発生した。これは、ラインが映る本数によってサーボ制御が不安定になる逆透視変換アルゴリズム(カメラ画像を上面視点に変換する手法)に起因する。第三に、コードの処理負荷が大きく、モバイルバッテリーでの電力供給が困難であった。この成果を踏まえ、本プロジェクトでは、ラジコンカーをベースにした小型自動運転車の開発を目指す。このラジコンカーでは、リアルタイム処理や画像処理を得意とする Raspberry Pi 5 を使う。学習モデルを作成し、軽量化することで Raspberry Pi 5 の負荷を抑え、第一、第二の課題を解決する。また、Raspberry Pi 5 に必要な電力を十分に供給できるバッテリーを使用することで第三の課題を解決する。本プロジェクトでは、Python を活用した画像処理技術や機械学習を活用し、自動運転車が自律走行、標識や障害物に対して適切に対応できる機能を実装する。

(※文責：佐藤)

2.2 必要なスキル

2.2.1 機械学習によるモデルの作成

本プロジェクトでは、標識認識およびレーン認識に機械学習を用いたモデルを作成した。標識認識には「分類および検出モデル」を、レーン認識には「セグメンテーションモデル」を使用する。セグメンテーションとは、対象を分割し、画像認識を行う手法である。これにより、さまざまな走行条件下でも適切に判断・動作できる柔軟なシステムを実現することを目的とした。通常の場合、条件分岐のみで構成されたプログラムでは、明度の変化や標識の向きなど、複雑な環境下での判断が困難である。そのため、機械学習によるモデル化を行い、多様な状況に対応可能な認識精度の向上を図った。学習データには、画像の回転によるデータ拡張を実施した約 50 枚の画像を使用し、撮影条件として明度や角度を変化させてデータを収集した。これにより、明度や角度の変化に対応することができる学習を行っている。さらに、作成したモデルは Raspberry Pi 5 上で動作させるため、推論性能を維持しつつ処理負荷を軽減できるよう、モデルの軽量化を行った。これにより、限られたリソース環境下でもリアルタイムに動作可能な認識システムを構築する。

(※文責：佐藤)

2.2.2 YDLiDAR X2 を使用した障害物検知

LiDAR を用いた障害物検知では、周囲の物体までの距離および角度情報を直接取得できる点の特徴である。本プロジェクトでは、2次元 LiDAR である YDLiDAR X2 を用いることで、水平方向の物体分布を距離と角度の組として取得する手法を採用した。これは、カメラによる画像認識は、物体の種類や形状を識別できる反面、周囲の明るさの条件や逆光の影響を受けやすく、また物体の正確な位置座標を直接得ることが難しい。これに対し、LiDAR は周囲の明るさの影響を受けにくく、物体までの距離を安定して計測できるため、衝突回避を目的とした障害物検知に適している。このため、本プロジェクトでは、標識認識やレーン認識を担うカメラに加えて、LiDAR を併用する構成とした。LiDAR から得られた距離及び角度は、前方に障害物が存在するかを判断するために用いられ、あらかじめ設定した条件を満たした場合には停止動作を行う判断に利用される。

(※文責：佐藤)

2.2.3 QR コードを用いた現在地取得

本研究では、自動運転カーを用いた小規模なマップ上での走行実験を行う。そのため、Bluetooth や GPS を用いた位置推定は実験環境の特性上、適用が難しい。そこで、マップ上に QR コードを設置し、自動運転車が自身の位置を常に取得できるようにした。車両前方に搭載したカメラで QR コードを読み取り、走行中も継続的に位置認識を行うことで、自身の位置を正確に把握する。QR コードは交差点などの分岐点にも配置しており、読み取り精度に応じて高い確率で現在地を正確に推定することが可能である。この方法により、小規模環境でも安定した位置情報取得を実現している。

(※文責：佐藤)

第 3 章 プロジェクト学習の最終的な目標

3.1 標識認識・レーン認識機能の実装

自動運転カーは、標識を正しく認識し、それに従って運転する能力が必要である。本プロジェクトでは、機械学習を活用することで標識を正確に認識し、進行・一時停止などの適切な制御の実装を目指す。また、自動運転カーは、レーンの位置を正確に把握し、レーンから逸脱しないように走行する必要がある。特に、実線および点線といったレーンの種類を識別し、走行可能な範囲を判断できることが求められる。これにより、安全性を確保しつつ、走行環境のルールに沿った安定した自律走行を目指す。

(※文責：大山)

3.2 LiDAR 機能の実装

自動運転カーには、周囲に走行の邪魔となる障害物がないか把握し、人や物とぶつからないように安全に走行する機能が求められる。具体的には、前方 80 度以内かつ 40cm 以内に障害物が存在する場合、距離情報を取得し、定めた範囲内に障害物が接近した際には自動的に停止する動作を行うように設定した。障害物の検知には、LiDAR を使用する。LiDAR は、カメラが使えない悪天候でも使用することができ、カメラよりも障害物へのリアルタイム処理が早く正確であるため、安全性を確保できる。安全性を高めることでより信頼性の高い自律走行システムの実現を目指す。

(※文責：大山)

3.3 目的地までの自律走行

本プロジェクトでは、出発地点から目的地までの自律走行を実現することを目標とする。ただし、走行範囲は屋内の小規模なマップ上に限定し、車線の維持や交差点での停止・進行といった部分的な自立走行を対象とする。自律走行を実現するためには、走行中に自車の位置を正確に把握する必要がある。しかし、本プロジェクトの実験環境は屋内であるため、QR コードを車両前方に搭載したカメラでこれを読み取ることで現在地を取得する方式を採用した。読み取った QR コード情報は、現在地の識別および進行方向の判断に利用される。これに基づいて、自動運転カーが次に進むべき経路を判断し、車線認識や標識認識と組み合わせて目的地までの経路を自律的に走行することを目指す。

(※文責：大山)

第 4 章 目的を達成するための手法，方法

4.1 学習モデル

本研究では，AI カーの走行に必要な認識機能を実現するため，標識認識モデルとレーン認識モデルの二種類の学習モデルを実装した．各モデルの学習には，あらかじめ撮影・アノテーションした標識画像およびレーン画像からなる静止画像データセットを用いた．データセットの規模は 997 枚の画像からなる 102MB のものである．一方，走行時の推論では，車両に搭載したカメラからリアルタイムに取得される画像を入力とし，認識結果をもとに走行制御を行う構成としている．標識認識モデルでは，「止まれ」および「車両進入禁止」の二種類の標識を対象として学習を行った．推論では，検出された標識の種類と位置情報が得られ，これらの情報は，一時停止などの走行制御に利用される．レーン認識モデルでは，実線および点線の二種類のレーンを対象とした学習を行った．推論では，カメラ画像中におけるレーンの領域およびその種類が得られ，走行方向の決定に用いられる．これら二つのモデルの学習は，計算資源を十分に確保できるノートパソコン上で実施した．学習後，学習済みモデルを Raspberry Pi 5 に転送し，走行時には推論処理のみを Raspberry Pi 5 上で実行する構成とした．

(※文責：太田)

4.1.1 標識認識

標識認識には，物体検出ライブラリである YOLO を使用した．モデル作成にあたり，まず学習用データセットを構築した．標識画像は，さまざまな撮影角度や照度条件をランダムで自分たちの手で撮影，収集し，それぞれに対してノイズ除去やグレースケール化などの前処理を施した．ノイズ除去は，撮影環境によって生じるランダムな画素の揺らぎを低減し，学習時に不要な特徴を取り込まないようにするために実施した．また，グレースケール化は色情報を削減し，形状やパターンといった構造的特徴を強調することで，学習効率を高める目的がある．前処理後，アノテーション作業として画像中の標識にラベル付けを行った．本研究では，画像内で検出対象を矩形（バウンディングボックス）で囲み，その領域を「標識」として学習させる手法を採用した．ここで，バウンディングボックスとは，物体検出で対象物の位置と大きさを表す最小の長方形領域のことである．学習データの作成後には，データセットの構成やクラス情報，ファイルパスなどを記述した YAML ファイルを作成した．YAML ファイルは，モデルが学習時に参照する設定情報をまとめた定義ファイルであり，データセット構造やパラメータ設定の管理に用いる．このファイルをもとに，YOLO モデルの学習を実行した．



図 4.1 ラベル付けの結果

```
path: ~~~
train: images/train
val: images/val
nc: 2
names: ['tomare', 'circle']
```

図 4.2 作成した YAML ファイル

(※文責：太田)

4.1.2 レーン認識

レーン認識モデルでは、走行中の車両が進行すべきレーンを正確に把握することを目的とし、車両底部に下向きに設置したカメラの画像を用いて走行レーンの認識を行った。下向きに設置することで、地面に描かれたレーン全体を安定して撮影でき、光の反射や周囲環境の影響を最小限に抑えることができる。走行マップ上のレーンは黒色で描かれており、実線および点線の二種類を認識対象とした。本モデルの作成にあたり、まず学習用データセットを構築した。さまざまな走行位置や照明条件で撮影した画像を収集し、それぞれに対してノイズ除去やグレースケール化などの前処理を行った。ノイズ除去は不要な画素の揺らぎを低減し、グレースケール化は線の形状特徴を強調するために行った。その後、レーン領域を画素単位で分類（セグメンテーション）するためにアンノテーション作業を実施し、画像中の実線および点線部分にラベル付けを行った。ここでいうセグメンテーションとは、画像を複数の領域に分割し、各領域がどのクラス（レーン・背景など）に属するかを識別する処理を指す。

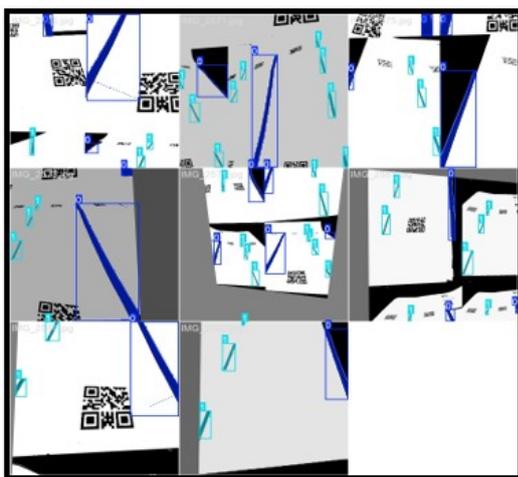


図 4.3 ラベル付けの結果

```
path: ~~~
train: images/train
val: images/val
names:
  0: thick_line
  1: thin_line
```

図 4.4 作成した YAML ファイル

(※文責：太田)

4.2 モデルの軽量化

自動運転車を安定して自律走行させるためには、標識認識およびレーン認識をリアルタイムで実行できる処理速度が必要である。しかし本プロジェクトでは、標識認識 (YOLOv5 s) とレーン認識 (YOLOv8 s) を同時に動作させる構成を採用したため、Raspberry Pi 5 上での推論処理が重くなり、映像処理の遅延が発生した。この遅延は、標識を認識してから停止制御に入るまでの反応を遅らせ、走行の安全性を低下させる要因となる。そこで、認識処理の精度を維持しつつ推論を高速化するため、学習モデルの軽量化を行った。軽量化手法としては知識蒸留を採用した。知識蒸留は、事前に学習済みの大きなモデル (教師モデル) の出力傾向を、小さなモデル (生徒モデル) が模倣するように再学習させることで、モデルを小型化しながら性能を出来る限り維持する方法である。単純にモデル規模を下げるだけでは認識性能が低下しやすいが、知識蒸留では教師モデルが持つ判別の傾向を生徒モデルへ引き継げるため、小型化と安定動作を両立しやすい点が利点である。本プロジェクトでは、標識認識において YOLOv5s を教師モデル、YOLOv5n を生徒モデルとして知識蒸留を行い、レーン認識においては YOLOv8s を教師モデル、YOLOv8n を生徒モデルとして知識蒸留を行った。

(※文責：太田)

4.3 LiDAR による物体検出

自律走行中の衝突を防ぐためには、カメラによる標識認識やレーン認識に加えて、周囲の物体までの距離を正確に計測する仕組みが必要である。本プロジェクトでは、距離情報を取得するために YDLiDAR X2(以下 LiDAR と記載) を使用した。LiDAR は、水平方向を 360 度スキャンし、物体までの距離と角度を取得する 2 次元 LiDAR である。LiDAR は Python プログラムを用いて制御した。LiDAR から取得されるデータは、距離と角度の組として出力される点群情報である。本システムでは、すべての点を扱うのではなく、自動運転車の進行に直接影響を与える前方領域に限定して障害物検知を行うことで、走行判断に不要な情報を除外し、処理負荷の削減を図った。具体的には、検出範囲を前方 80 度以内かつ 40cm 以内に存在する点のみを検出対象とした。前方 80 度という角度範囲は、小型マップ内で走行を行う自動運転車において、衝突の可能性が高い前方領域に限定して障害物を検知することを目的として設定した。また、検出距離を 40cm 以内とした理由は、この距離内に障害物が存在する場合には即座に停止動作を行うことで衝突を回避したうえで、障害物との安全な距離を確保できると判断したためである。検出された点群に対しては、隣接点の連続性に基づく簡易的なクラスタリング処理を行った。まず、前方角度および距離条件を満たした点群の中で、隣接する点同士について、角度差および距離差が小さい場合、または直角座標上で互いに近接している場合に、同一物体に属する点として連結し、一つのクラスタとした。この処理により、一つの物体が LiDAR によって複数の測距点として検出された場合でも、それらを一つのクラスタとしてまとめることができる。各クラスタについては、クラスタ内で LiDAR から最も近い測距点を代表点として選択し、その角度と距離を障害物の位置情報として出力した。この処理により、一つの物体が複数の測距点として検出された場合でも、単一の障害物として安定して扱うことが可能となった。図 4.5 に実験環境の外観を示す。図中の青色の枠は LiDAR を示し、赤枠および黄枠は障害物として配置した二つの物体である。図 4.6 は図 4.5 の環境において LiDAR が

Make Brain Project

ら取得された点群を極座標上に可視化した結果であり、前方 40cm 以内に存在する障害物が検出されている様子を示している。図 4.6 より、前方の検出範囲内に複数の物体が存在し、各物体の自動運転カーから見た距離と角度まで認識していることがわかる。このように、LiDAR を用いることで、カメラでは把握しにくい距離情報を直接取得し、自動運転カーの前方状況を把握する機能を実装した。本来は検知した障害物に対して回避行動を自動運転を行うことを目標としていたが、実装期間および制御の安定性を考慮し、本プロジェクトでは障害物検知に基づく停止動作までの実装に留めた。

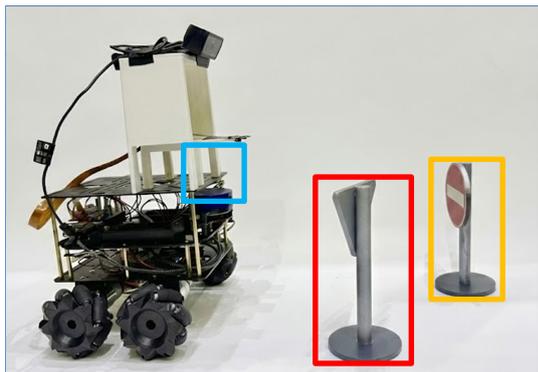


図 4.5 AI カー前方に障害物を配置した様子

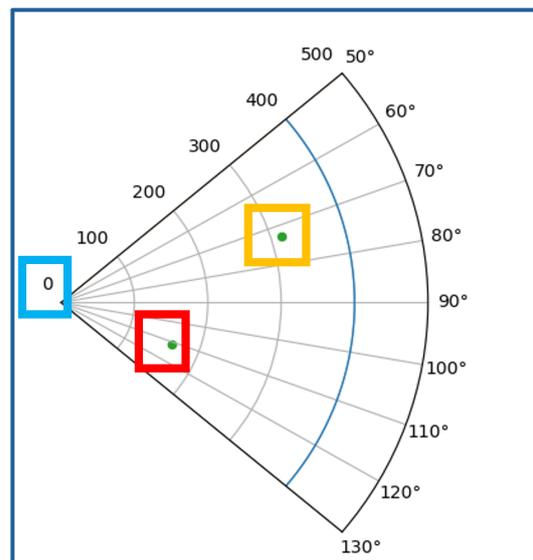


図 4.6 前方 40cm 以内にある障害物を検知した結果

(※文責：太田)

第 5 章 結果

実装予定であった機能について 3 つの機能を作成した。一つ目に、標識認識およびレーン認識に用いた学習モデルの推論結果と、知識蒸留による軽量化の効果について示す。まず、標識とレーンをリアルタイムのカメラ画像から認識するために、YOLO を用いた機械学習を行い、標識認識モデルとレーン認識モデルの二つを作成した。各認識モデルを用いて推論を行った結果、標識認識モデルでは tomare と tuukoukinshi の 2 つのラベルを識別でき、レーン認識モデルでは実線のレーンと点線のレーンを区別できることを確認した。しかし、作成したモデルを用いて標識認識およびレーン認識を同時に実行した場合、Raspberry Pi 5 上では推論処理の負荷が大きくなり、走行制御に遅延が生じる課題が確認された。この課題を解決するため、本プロジェクトでは、推論精度を可能な限り維持しつつモデルを小型化する手法として知識蒸留を適用した。表 5 に、知識蒸留前後における標識認識モデルおよびレーン認識モデルの性能の比較を示す。表 5 に示すように、知識蒸留の適用により、標識認識モデルおよびレーン認識モデルのいずれにおいても、パラメータ数は、約 70% 以上の大幅な削減が確認された。また、この軽量化の効果を確認するため、知識蒸留前後のモデルについて、精度の傾向を比較した。本研究では、認識モデルの性能を評価する指標として、推論時に出力される判定スコアの大きさとその安定性に着目し、これを認識精度として扱った。判定スコアとは、入力画像中の検出領域が特定のクラスに属するとモデルが判断した際に出力される数値であり、0 から 1 の範囲で表される。値が大きいほど認識の精度が強いことを示す。標識認識モデルでは、蒸留前のモデルにおいて、推論結果として出力される精度は概ね 0.8 から 0.9 の範囲に分布していた。一方、知識蒸留後のモデルでは、精度は概ね 0.7 程度であった。このことから、標識認識においては、モデルの軽量化に伴い推論の精度がやや低下することが分かった。レーン認識モデルでは、蒸留前後いずれのモデルにおいても、精度は概ね 0.9 程度を維持しており、知識蒸留による顕著な低下は確認されなかった。この結果から、レーン認識においては、モデルを軽量化した場合でも、推論結果の安定性が保たれることが示された。以上の結果より、知識蒸留によるモデルの軽量化は、Raspberry Pi 5 上で複数の認識処理を同時に実行するために有効であり、走行制御に必要な認識性能を維持しつつ、処理遅延の軽減に寄与することが確認できた。

二つ目に、LiDAR による障害物検知機能について示す。LiDAR を用いて、設定した範囲内に障害物があった際に障害物を検知し、グラフにプロットすることができた。また、単一の物体が複数の点として検出される場合があったため、クラスタリング処理を適用し、一つの物体として認識できるようにした。範囲内に複数の物体を置いた場合でも、複数の物体としてクラスタリングすることができ、それぞれの物体の代表点を 1 つプロットすることができた。

三つ目に、QR コードを読み取り、位置情報を取得する機能について示す。本プロジェクトでは、1 から 48 までの数字を格納した QR コードをマップ上に印刷した。これらの QR コードを読み取り、走行開始地から目的地まで正しく番号が取得できているかを判定するプログラムを作成した。上記の機能を実装したうえで、車体の作成も行った。車体は、OSOYOO FlexiRover ロボットカーキット Raspberry Pi 用を使用し、LiDAR と 2 台のカメラを搭載した。カメラのうち 1 台は QR コードを読み取りをするために真下に向けた。もう 1 台は、標識とレーンの認識を行うために高い台の上に設置した。車体を作成したうえで、基本的な走行実験を行うことができた。

表 5.1 知識蒸留前後におけるモデル性能比較

モデル	構成	パラメータ数 (M)	confidence
標識認識	蒸留前 (YOLOv5s)	7.0	約0.8~0.9
標識認識	蒸留後 (YOLOv5n)	1.7	約0.7
レーン認識	蒸留前 (YOLOv8s)	11.7	約0.9~1.0
レーン認識	蒸留後 (YOLOv8n)	3.2	約0.9

(※文責：佐藤)

第 6 章 考察

本プロジェクトでは、標識認識モデルおよびレーン認識モデルの構築を行い、自動運転に必要な要素技術の一部を実装した。しかし、開発および検証を進める中で、今後の課題が大きく3点あることが明らかとなった。

1つ目は、各機能の統合である。現段階では、標識認識やレーン認識といった機能は個別に動作しており、走行制御や目的地到着を一貫して自律的に行うシステムには至っていない。この課題が生じた要因として、車体の組み立てや Raspberry Pi 5 の環境構築に想定以上の時間を要し、機能統合および十分な動作検証の時間を確保できなかったことが挙げられる。また、技術的課題として、すべての機能を統合した際に

2つ目は、認識精度の安定性である。本研究を通して、認識精度が照明条件やカメラの設置角度といった周囲の環境条件に大きく依存することが確認された。これは、学習データが限られた環境条件に偏っていたことが一因であり、時間的制約により多様なデータ収集が十分に行えなかったことが影響していると考えられる。

3つ目は、走行制御、特に旋回動作の改良である。本システムでは購入した開発キットの使用からメカナムホイールを使用しているが、一般的な二輪駆動車両と同様の制御手法では安定した旋回動作を実現することができなかった。この点についても、制御方法の検討や調整に十分な時間を割くことができなかったことが要因の一つである。

今後の展望としては、これら三つの課題に段階的に取り組むことで、より実環境に近い自動運転システムへと発展させることが考えられる。まず、各認識機能と走行制御を統合したシステムを構築し、全体の動作検証を行う。その上で、学習データの拡充やモデルの高精度化を進めるとともに Raspberry Pi 5 よりも高性能な jetson nano などのハードウェアを導入することで、より遅延の少なく、カメラ映像を取得でき、高いリアルタイム性を持つ認識および制御の実現を目指す。また、メカナムホイールの特性を考慮した制御手法を確立することで、より安定した走行動作が可能になると考えられる。

これらの取り組みにより、ルート検索から目的地到着までを自律的に行う完全自動運転システムの実現に近づけることができると考えられる。

(※文責：大山)

参考文献

- [1] Make Brain Project, 「視覚を持つ AI カー」, 2024, 公立ほこだて未来大学 システム情報科学実習 グループ報告書. Available at: <https://www.fun.ac.jp/wp-content/uploads/2025/05/group21C.pdf>
- [2] OSOYOO, "FlexiRover ベーシックロボットカー (Mega2560 用): メカナムホイール付き 520 モーター搭載", 2024. Available at: <https://osoyoo.com/ja/2024/08/26/osoyoo-flexirover-basic-robot-car-for-mega2560-running-520-motor-with-mecanum-wheel-mod#4>
- [3] 公益財団法人 交通事故総合分析センター, 「交通事故発生状況」, 2025. Available at: https://www.itarda.or.jp/situation_accidents (Accessed: 2025-07-16)