

# 卒業論文

## 視覚と行動の end-to-end 学習により経路追従行動を オンラインで模倣する手法の提案 (オフラインでデータセットを収集して訓練する手法の検証)

A proposal for an online imitation method of path-tracking  
behavior by end-to-end learning of vision and action  
(Validation of a method to collect and train datasets offline)

2022 年 12 月 30 日 提出

指導教員 林原 靖男 教授

千葉工業大学 先進工学部 未来ロボティクス学科

19C1068 高橋祐樹



# 概要

## 視覚と行動の end-to-end 学習により経路追従行動を オンラインで模倣する手法の提案 (オフラインでデータセットを収集して訓練する手法の検証)

近年、自律移動ロボットの研究が盛んに行われている。本研究室においても、2D-LiDAR を用いた自律移動システムの出力を教師信号として与えることで、ロボットの経路追従行動をオンラインで模倣する手法を提案し、実験を行うことで有効性を示してきた。本研究では、従来手法を基に、目標とする経路上及び周辺データを一度に収集し、オフラインで訓練する手法を提案する。提案手法では、経路上にロボットを配置し、カメラ画像と教師データとなる目標角速度を収集する。それらのデータを基にオフラインで学習を行い、学習後はカメラ画像を入力とした学習出力により自律移動する。

キーワード: end-to-end 学習, Navigation, オフライン

# abstract

A proposal for an online imitation method of path-tracking  
behavior by end-to-end learning of vision and action  
(Validation of a method to collect and train dataset offline)

Recently, autonomous mobile robots have been studied extensively. In our laboratory, we have proposed an online imitation method of a robot's path-following behavior by providing the output of a 2D-LiDAR-based autonomous mobile system as a teacher signal, and have demonstrated the effectiveness of the proposed method through experiments. In this study, we propose an off-line training method based on the conventional method by collecting data on and around the target path at a time. In the proposed method, a robot is placed on the path and collects camera images and target angular velocity as teacher data. The robot is trained off-line based on these data, and after training, the robot moves autonomously using the training output from the camera images as input.

keywords: End-to-End Learning, Navigation, Offline

# 目次

|       |                                     |    |
|-------|-------------------------------------|----|
| 第 1 章 | 序論                                  | 1  |
| 1.1   | 背景 . . . . .                        | 1  |
| 1.2   | 関連研究 . . . . .                      | 3  |
| 1.3   | 目的 . . . . .                        | 4  |
| 1.4   | 論文構成 . . . . .                      | 4  |
| 第 2 章 | 要素技術                                | 5  |
| 2.1   | 地図を用いたルールベース制御器によるナビゲーション . . . . . | 5  |
| 2.2   | ディープラーニング . . . . .                 | 6  |
| 2.3   | end-to-end 学習 . . . . .             | 7  |
| 2.4   | データセット . . . . .                    | 8  |
| 2.5   | 模倣学習におけるオフライン . . . . .             | 8  |
| 2.6   | ミニバッチ学習 . . . . .                   | 8  |
| 2.7   | バッチ学習 . . . . .                     | 8  |
| 第 3 章 | 提案手法                                | 9  |
| 3.1   | 手法 1 . . . . .                      | 9  |
| 3.1.1 | データの収集方法 . . . . .                  | 9  |
| 3.1.2 | 訓練 1 . . . . .                      | 11 |
| 3.1.3 | 実験 1 . . . . .                      | 11 |
| 3.2   | 手法 2 . . . . .                      | 14 |
| 3.2.1 | データの収集方法 . . . . .                  | 14 |

|                        |    |
|------------------------|----|
| 目次                     | vi |
| 3.2.2 訓練 2-1 . . . . . | 14 |
| 3.2.3 実験 2-1 . . . . . | 14 |
| 3.2.4 訓練 2-2 . . . . . | 16 |
| 3.2.5 実験 2-2 . . . . . | 17 |
| 3.2.6 訓練 2-3 . . . . . | 18 |
| 3.2.7 実験 2-3 . . . . . | 18 |
| 参考文献                   | 19 |
| 付録                     | 20 |
| 謝辞                     | 21 |

# 目次

|     |  |    |
|-----|--|----|
| 1.1 | Training the neural network from [1] . . . . .                           | 1  |
| 1.2 | Driving with output from map-based navigation indoors from [2] . . . . . | 2  |
| 1.3 | Okada and others proposed method from [2] . . . . .                      | 2  |
| 1.4 | Kiyooka and others proposed method from [3] . . . . .                    | 3  |
| 1.5 | Yunpeng Pan and others proposed method from [4] . . . . .                | 4  |
| 1.6 | Autonomous off-road driving . . . . .                                    | 4  |
| 2.1 | Map based navigation using navigation package . . . . .                  | 5  |
| 2.2 | Structure of Deep Learning . . . . .                                     | 6  |
| 2.3 | Structure of general Learning . . . . .                                  | 7  |
| 2.4 | Structure of end-to-end Learning . . . . .                               | 7  |
| 2.5 | MNIST dataset from [5] . . . . .   | 8  |
| 3.1 | Method of collecting data around the target route . . . . .              | 9  |
| 3.2 | Method of collecting data around the target route . . . . .              | 10 |
| 3.3 | Course to collect data . . . . .   | 10 |
| 3.4 | Structure of network . . . . .   | 11 |
| 3.5 | Turtlebot3 waffle with 3 cameras . . . . .                               | 12 |
| 3.6 | Failure point of the experiment1 . . . . .                               | 13 |
| 3.7 | Histogram of collected angular velocities in the experiment1 . . . . .   | 13 |
| 3.8 | Method of collecting data around the target route . . . . .              | 14 |
| 3.9 | Failure point of the experiment2-1 . . . . .                             | 15 |

---

|      |  |    |
|------|--|----|
| 3.10 | Histogram of collected angular velocities in the experiment2-1 . . . . . | 16 |
| 3.11 | Failure point of the experiment2-2 . . . . .                             | 17 |



# 表目次

|     |  |    |
|-----|--|----|
| 3.1 | Number of successes in the experiment1 . . . . .   | 12 |
| 3.2 | Number of successes in the experiment2-1 . . . . . | 15 |
| 3.3 | Number of successes in the experiment2-2 . . . . . | 17 |
| 3.4 | Number of successes in the experiment2-3 . . . . . | 18 |

# 第 1 章

## 序論

### 1.1 背景

近年, 様々なセンサを用いた自律移動に関する研究が活発に行われており, その中で視覚を入力とした end-to-end 学習により自律走行した例もある. 例えば, Bojaski らは Fig. 1.1 に示すシステムでカメラ画像と人が操作するステアリングの角度を end-to-end 学習することで, 自律走行する手法を提案した [1].

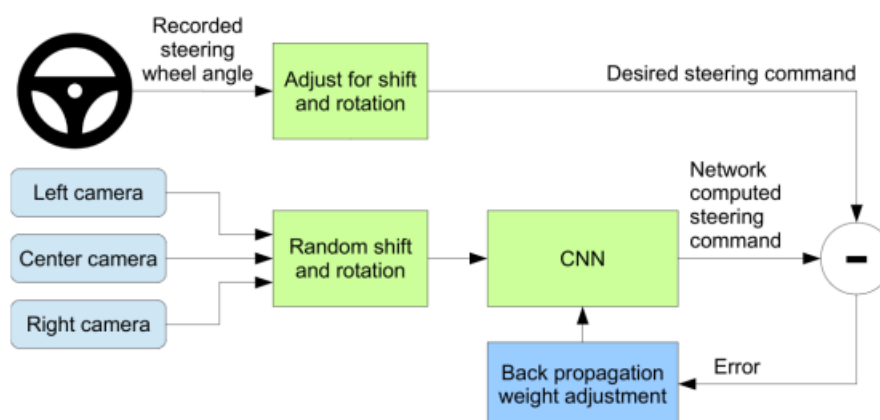


Fig. 1.1: Training the neural network from [1]

岡田らにより Fig. 1.2 のように地図ベースのナビゲーションによる出力を模倣することで、経路追従行動を獲得した [2]. Fig. 1.3 に示すような, LiDAR, オドメトリを入力としたナビゲーションの出力を end-to-end で模倣学習し, 学習後はカメラ画像を入力とした学習器の出力により, 一定の経路において周回が可能であることが確認された.



Fig. 1.2: Driving with output from map-based navigation indoors from [2]

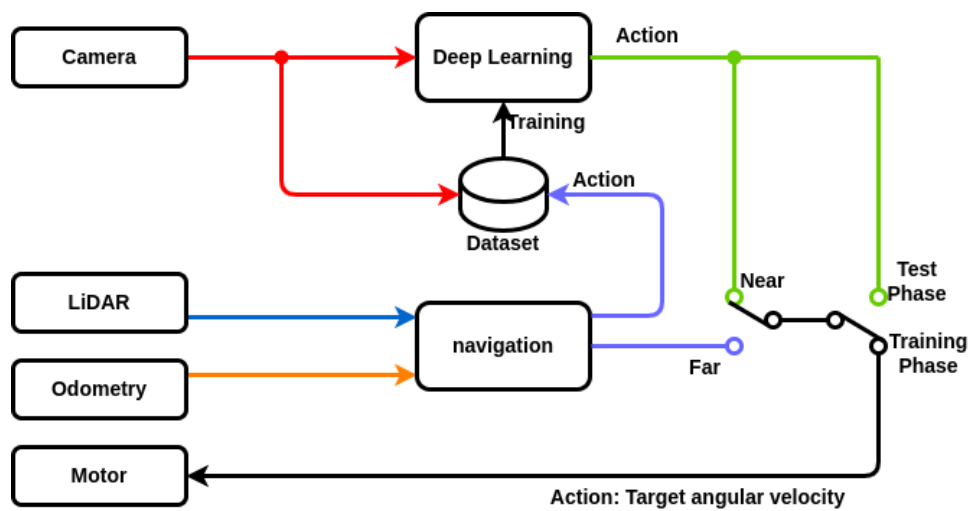


Fig. 1.3: Okada and others proposed method from [2]

また, 清岡ら [3](以下「先行研究」と称する) により, Fig. 1.4 に示すような手法を用いて, 経路上だけでなく経路から離れた状態も学習することが, 経路追従行動を模倣する上で有効であることが示された.

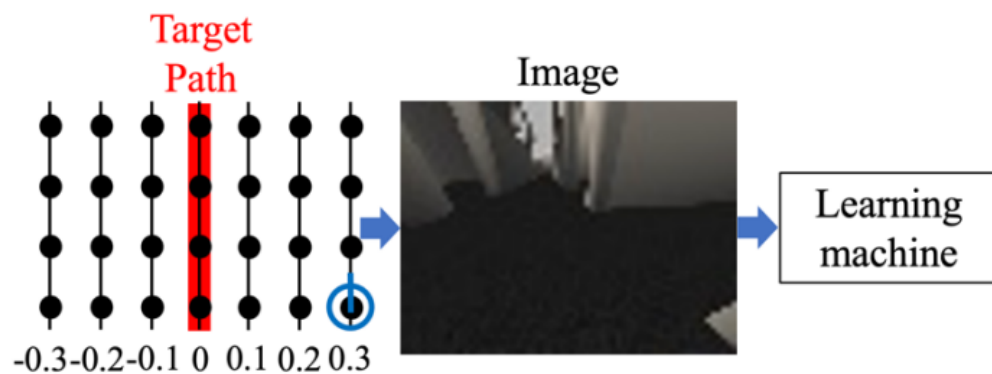


Fig. 1.4: Kiyooka and others proposed method from [3]

以上で述べたように, カメラ画像を入力とした学習器の出力により, ロボットが学習した経路を周回可能であることが示されている. しかし, 先行研究ではロボットを走行させながらオンラインで学習を行っているため, 目標経路周辺のデータを網羅することができない.

これを踏まえて, 先行研究を基に, 新たなデータセットの収集方法を考案する.

## 1.2 関連研究

Yunpeng Pan[4] らは, Fig. 1.5 のようなシステムでバッチ学習とオンライン学習を用いて, 高度なセンサーを備えたモデル予測制御を模倣した. これにより, Fig. 1.6 のように高速でオフロードを自律走行することに成功した.

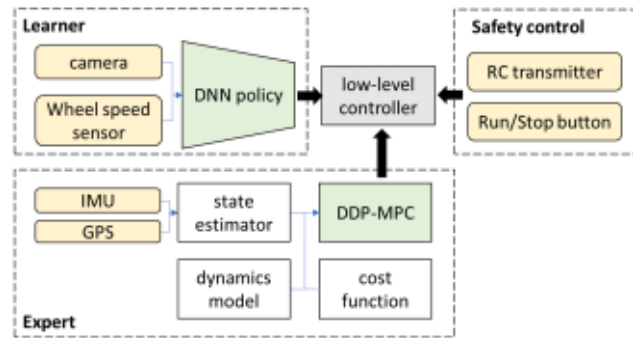


Fig. 1.5: Yunpeng Pan and others proposed method from [4]



Fig. 1.6: Autonomous off-road driving

### 1.3 目的

本研究では, 先行研究を基に, 新たなデータセットの収集方法を提案する. 提案手法の有効性をシミュレータを用いた実験により検証することを目的とする.

### 1.4 論文構成

本論文の構成は以下に述べる通りである. 第 1 章では研究を行う背景や目的を述べた. 第 2 章では研究に関連する要素技術, 第 3 章では提案手法について説明する.

## 第 2 章

# 要素技術

### 2.1 地図を用いたルールベース制御器によるナビゲーション

先行研究と提案手法において、教師信号としている地図を用いたルールベース制御器によるナビゲーションについて説明する。このナビゲーションには、ROS のパッケージである navigation[6] を使用している。移動ロボットは、Fig. 2.1 のように LiDAR のスキャンデータやオドメトリを入力として自己位置推定と経路計画を行い、これらに基づいて自律走行をする。また、自己位置推定には、amcl(Adaptive Monte Carlo Localization)、経路計画とモータ指令には move\_base[6] を使用している。

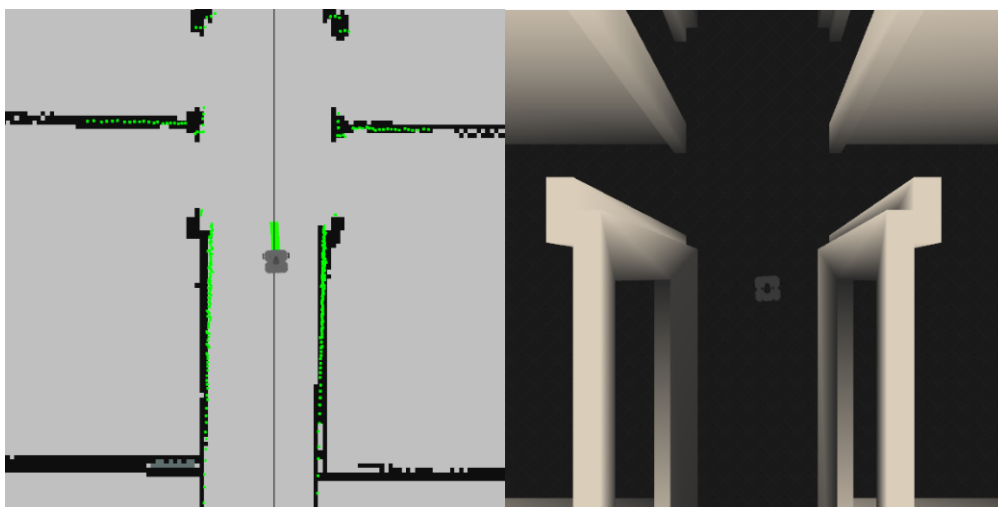


Fig. 2.1: Map based navigation using navigation package

## 2.2 ディープラーニング

ディープラーニングとは、人間の神経細胞を模したネットワーク構造のことである。主に、入力層と出力層、その間に中間層（隠れ層）という構成である。中間層を多層化することで、複雑な入力情報を処理し、パターンを認識することや、ルールを読み解くことができる。近年では、画像や物体認識、自然言語処理などで活用されている。Fig. 2.2 に構造の一例を示す。

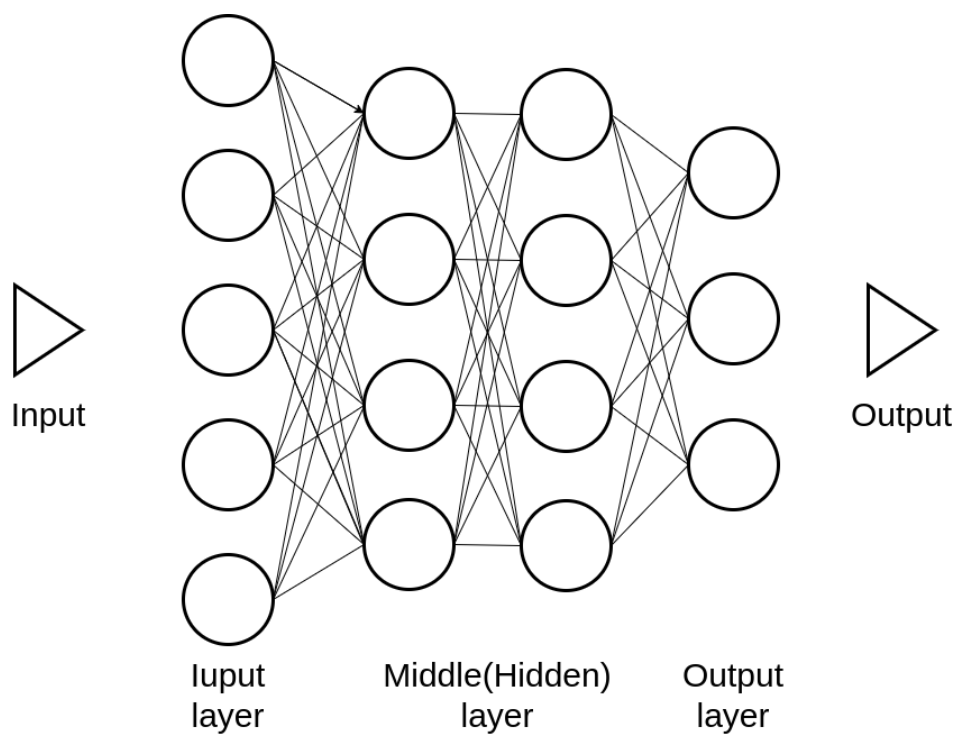


Fig. 2.2: Structure of Deep Learning

## 2.3 end-to-end 学習

end-to-end 学習とは、入力から出力までの流れを一括に学習することができる手法である。例として、画像中からの文字認識を行う処理を挙げる。一般的な処理では、Fig. 2.3 のように画像から文字検出を行い、その後に文字分割、最終的に文字認識をする。しかし、end-to-end 学習では、Fig. 2.4 に示すような入力から出力までの流れを一括して学習することができる。

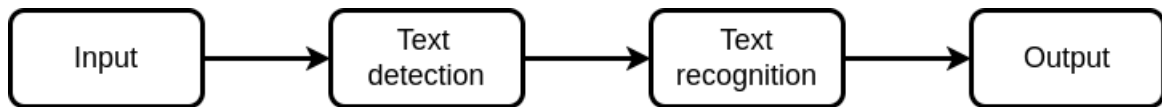


Fig. 2.3: Structure of general Learning

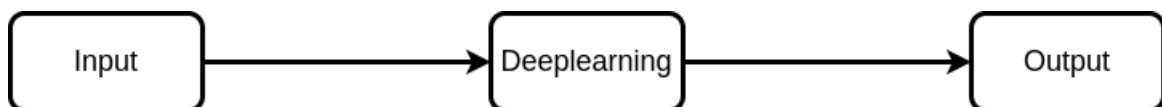


Fig. 2.4: Structure of end-to-end Learning



## 2.4 データセット

データセットとは、学習に使用する学習（訓練）データの集合のことである。例として、Fig. 2.5 に示すような 0 から 9 の手書きで書かれた数字の画像セットである MNIST が挙げられる。機械学習や画像認識において多く利用されており、訓練画像 6000 枚とテスト画像 1000 枚で構成されている。



Fig. 2.5: MNIST dataset from [5]

## 2.5 模倣学習におけるオフライン

模倣学習におけるオフラインとは、あらかじめ用意したデータセットを使用して学習を行うことである。これに対して、先行研究用で用いた模倣学習におけるオンラインとは、タスクを行いながらデータ収集をし、そのデータを使用して学習することを指す。

## 2.6 ミニバッチ学習

ミニバッチ学習とは、訓練データをいくつかのグループ（バッチ）に分けて順番に学習を行うことである。特徴としては、勾配更新の頻度が高く、計算量が少なくて済むといったことが挙げられる。これらの特徴を踏まえて、先行研究ではオンラインで学習を行うことから、ミニバッチ学習を使用している。

## 2.7 バッチ学習

バッチ学習とは、訓練データを一括で処理する学習方法である。特徴として、一度に大量のデータを扱うことができるため学習の進行が安定しやすく、訓練データに異常データが混じっていても受ける影響が小さくて済むなどが挙げられる。

## 第 3 章

# 提案手法

本章では、先行研究を基にした提案手法をデータの収集方法、訓練、実験の 3 節に分けて紹介する。

### 3.1 手法 1

#### 3.1.1 データの収集方法

Fig. 3.1 にデータの収集方法を示す。赤色の線である目標経路から平行に  $\pm 0.10$ ,  $\pm 0.20$ ,  $\pm 0.30\text{m}$  離れた座標にロボットを配置する。そして、その座標ごとに目標経路に沿った向きを基準として  $\pm 5$  度傾けて、 $64 \times 48$  のカメラ画像 (RGB 画像) とルールベース制御器によるナビゲーションの出力である角速度を Fig. 3.2 のように収集する。これを Gazebo の Willow Garage で Fig. 3.3 に示すコースで一周行う。

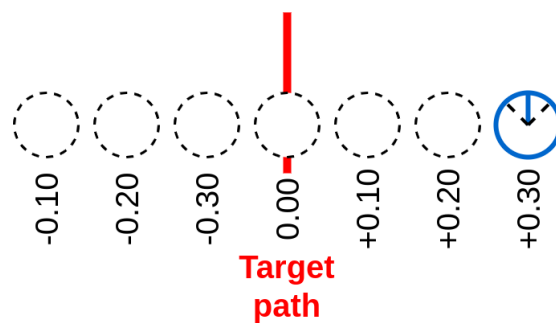


Fig. 3.1: Method of collecting data around the target route

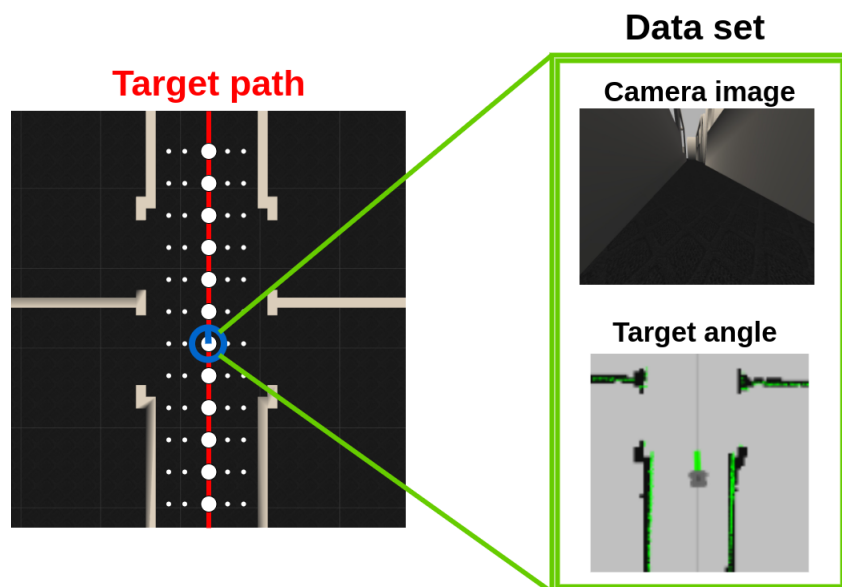


Fig. 3.2: Method of collecting data around the target route

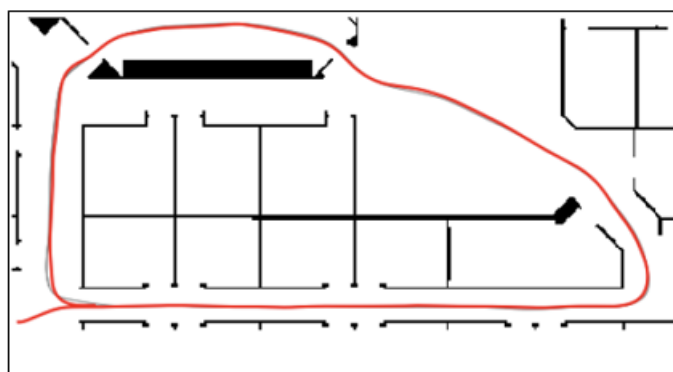


Fig. 3.3: Course to collect data

### 3.1.2 訓練 1

ネットワークの構造 Fig. 3.4 に訓練時に用いたネットワークの構造を示す. 構造は, 入力層 1, 畳み込み層 3, 全結合層 2, 出力層 1 の計 7 層から構成されている. 以下, ネットワークの構造は同じである.

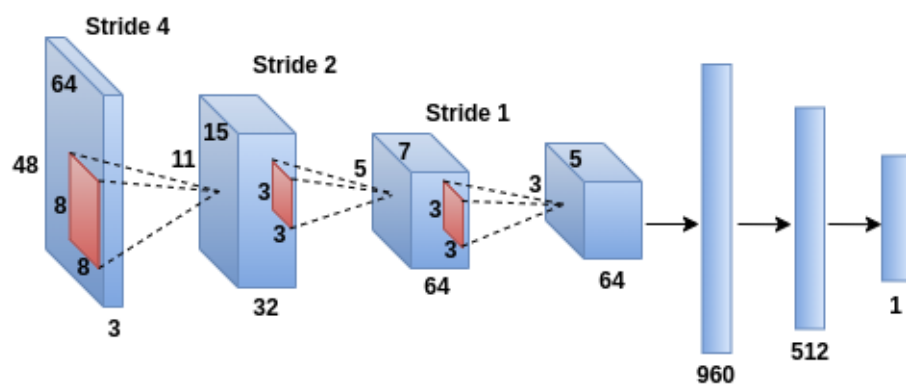


Fig. 3.4: Structure of network

訓練 バッチサイズ 8, データ量 2658, ミニバッチ学習 (先行研究に倣って) で 4000step, 8000step, 10000step 学習した.

### 3.1.3 実験 1

実験目的 シミュレータ上で実験を行い, 提案手法の有効性を検証する.

実験装置 シミュレータを用いた実験では, データの収集方法と同様に Willow Garage を使用した. また, ロボットモデルには Fig. 3.5 に示すようなカメラを 3 つ搭載した Turtlebot3 を用いた.



Fig. 3.5: Turtlebot3 waffle with 3 cameras

**実験方法** シミュレータを用いた実験では, Fig. 3.3 で示した経路で実験を行う. 壁に衝突せずに一周できた場合を成功とし, 壁に激突したり, コースアウトして経路に復帰できなかった場合を失敗とした.

以下, 実験目的, 実験装置, 実験方法は同じである.

**実験結果** 実験結果は, 3.1, 失敗箇所は Fig. 3.6 のようになった. 青色の×は壁への衝突箇所を示している.

| step  | Number of successes |
|-------|---------------------|
| 4000  | 0/10                |
| 8000  | 0/10                |
| 10000 | 0/10                |

Table 3.1: Number of successes in the experiment1

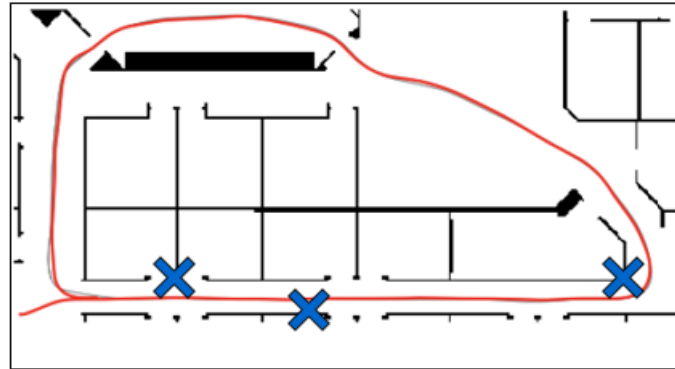


Fig. 3.6: Failure point of the experiment1

考察 学習量を増やしても成功回数は増えなかった. また, 目標経路から離れた際に戻る挙動や, 壁に近づきすぎた際に避ける挙動も見られなかった. そこで, 収集した角速度を Fig. 3.7 のようにヒストグラムにした. これより, 経路上及び経路周辺のデータ (0 から 0.1) が全体の 40 %程度しかないことが分かる. 経路上及び経路周辺のデータが多い方が, 経路追従の成功回数が増えるのではないかと考えた.

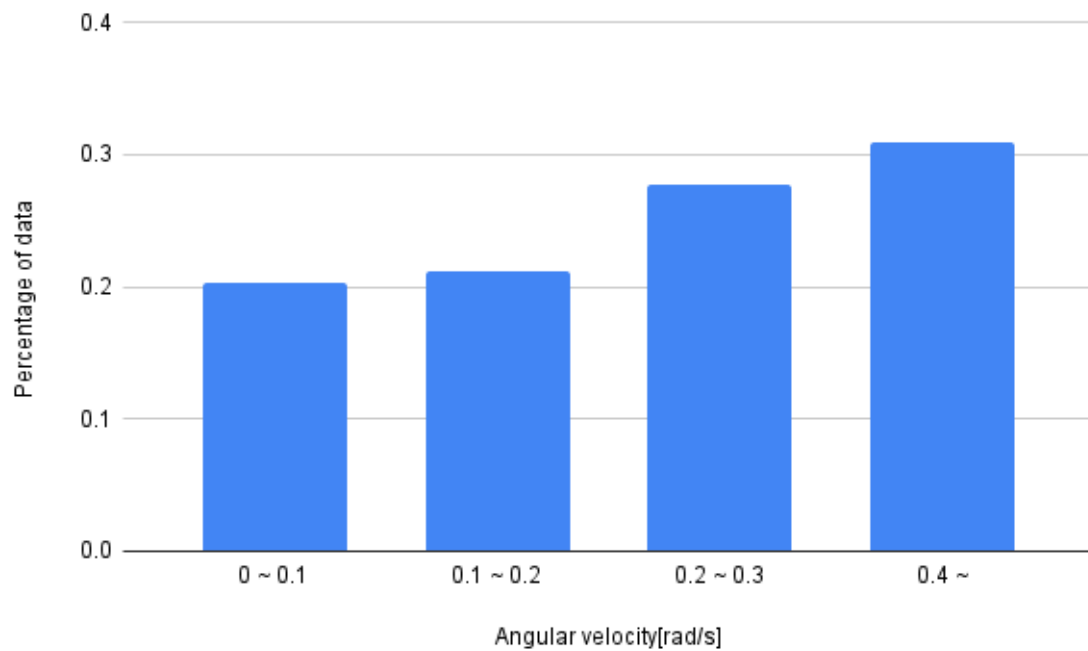


Fig. 3.7: Histogram of collected angular velocities in the experiment1

## 3.2 手法 2

### 3.2.1 データの収集方法

手法 1 を踏まえて、経路周辺のデータを多く取得する手法を試みる。Fig. 3.8 にデータの収集方法を示す。赤色の線である目標経路から平行に  $\pm 0.01$ ,  $\pm 0.02$ ,  $\pm 0.04$ ,  $\pm 0.06$ ,  $\pm 0.08$ ,  $\pm 0.10$ ,  $\pm 0.15$ ,  $\pm 0.20$ ,  $\pm 0.30\text{m}$  離れた座標にロボットを配置する。そして、手法 1 と同様にロボットを傾けて画像と角速度を Fig. 3.2 のように収集する。これを Fig. 3.3 に示すコースで一周行う。

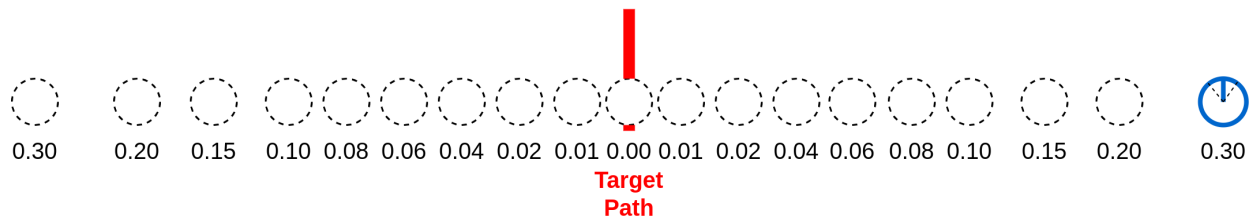


Fig. 3.8: Method of collecting data around the target route

### 3.2.2 訓練 2-1

訓練 バッチサイズ 8, データ量 7242, ミニバッチ学習で 4000step, 8000step, 10000step 学習した。

### 3.2.3 実験 2-1

実験結果 実験結果は、3.2 のようになった。Fig. 3.9 の青×の箇所でコースアウトして、目標経路に復帰できなかった。

| step  | Number of successes |
|-------|---------------------|
| 4000  | 0/10                |
| 8000  | 0/10                |
| 10000 | 0/10                |

Table 3.2: Number of successes in the experiment2-1

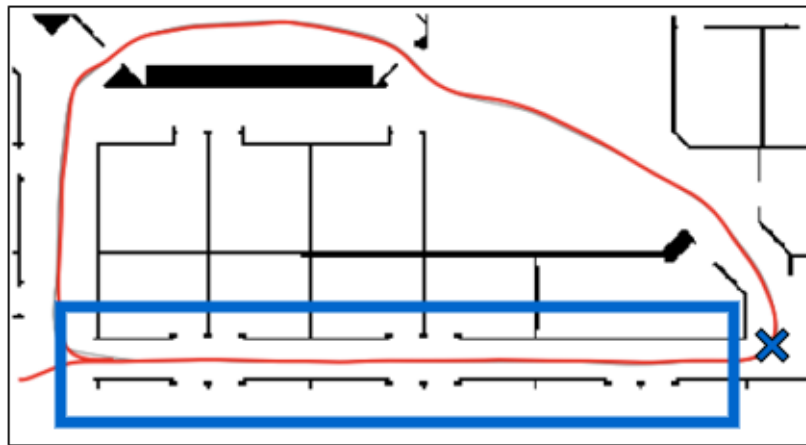


Fig. 3.9: Failure point of the experiment2-1

考察 収集した角速度を Fig. 3.10 のようにヒストグラムにした。経路上及び経路周辺のデータが Fig. 3.7 と比べて多くなっている。これにより, Fig. 3.9 の青枠の区間において, 壁に衝突することなく走行できていることが分かる。しかし, データ量や学習量を変化させても成功回数は増えなかった。そこで, 先行研究のオンライン学習では計算のリソースなどの観点からミニバッチ学習にしていたが, 提案手法ではオフラインで学習を行うため, バッチ学習に変更する。これにより, 一度に大量のデータを扱えるため最適解に辿り着くことができ, 成功回数が増えるのではないかと考えた。



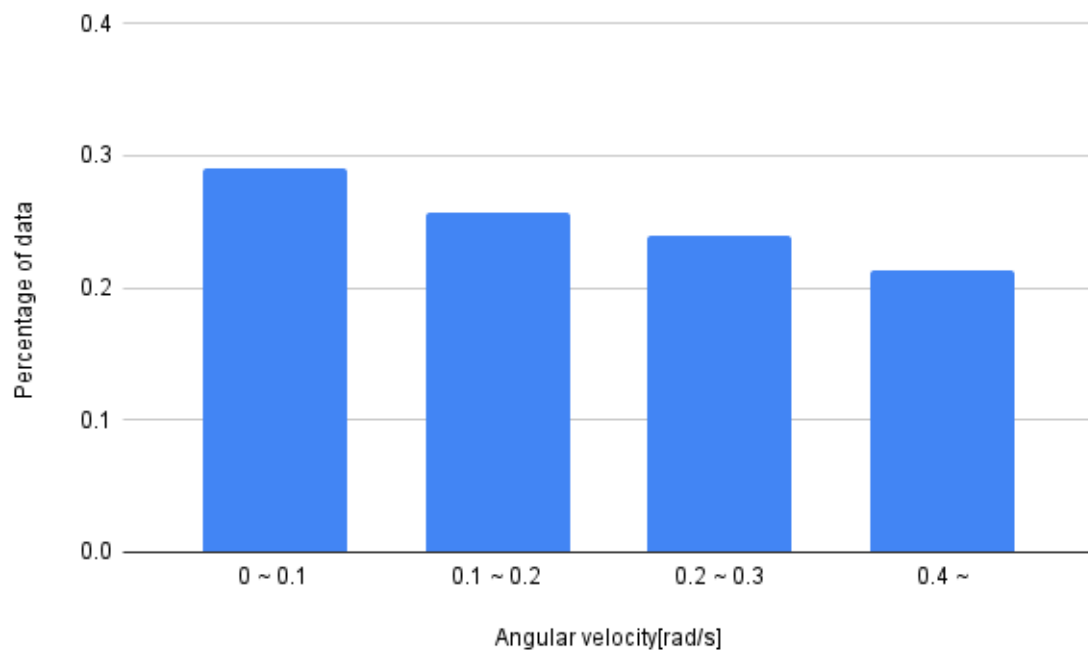


Fig. 3.10: Histogram of collected angular velocities in the experiment2-1

### 3.2.4 訓練 2-2

オフラインで学習を行うメリットを活かして, ここではバッチ学習を用いることで成功回数が増えるか検証する.

訓練 バッチサイズはデータ量と同じ 7242, バッチ学習で 4000step, 8000step, 10000step 学習した.

### 3.2.5 実験 2-2

実験結果 実験結果は, 3.3, Fig. 3.11 のようになった.

| step  | Number of successes |
|-------|---------------------|
| 4000  | 0/10                |
| 8000  | 1/10                |
| 10000 | 1/10                |

Table 3.3: Number of successes in the experiment2-2

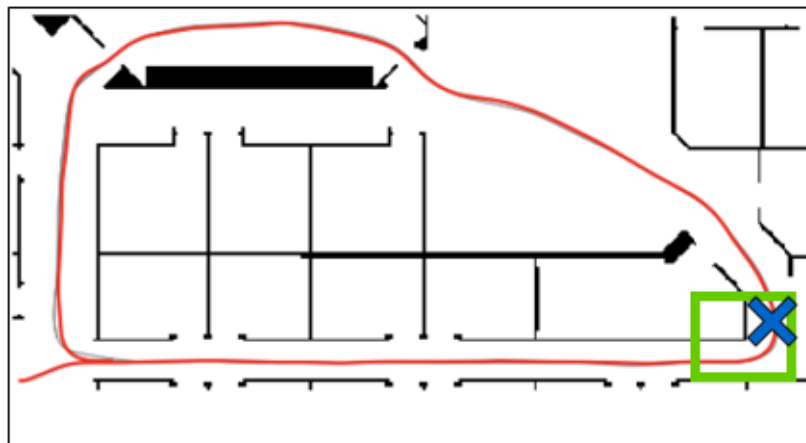


Fig. 3.11: Failure point of the experiment2-2

考察 4000step では成功回数が 0/10 だったが, 8000step, 10000step にすることで成功回数が 1/10 になり, 目標経路を一周することができた. このことから, 訓練する際に異常データから受ける影響が少なく済むバッチ学習を用いて, 学習量を増やすことで成功回数が増えることを示せた. また, Fig. 3.11 の緑枠内の角で曲がりきれず, 青×に示す箇所でコースアウトしたが, 訓練 2-1 と比べて角で曲がる挙動が見られた. よって, 訓練 2-2 では成功回数が十分であるとは言えず, 角を曲がり切れていない. そこで, Fig. 3.11 の緑枠内のデータを水増しすることで角を曲がり切ることができ, 成功回数が増えるのではないかと考えた.

### 3.2.6 訓練 2-3

ここでは、角のデータを水増しすることで成功回数が増えるかどうか検証する。

訓練 バッチサイズはデータ量と同じ 8154, バッチ学習で 4000step, 8000stp, 10000step 学習した。

### 3.2.7 実験 2-3

実験結果 実験結果は、3.4 のようになった。

| step  | Number of successes |
|-------|---------------------|
| 4000  | 0/10                |
| 8000  | 1/10                |
| 10000 | /10                 |

Table 3.4: Number of successes in the experiment2-3

## 参考文献

- [1] Mariusz Bojarski et al. ””end to end learning for self-driving cars.””. arXiv: 1604.07316(2016).
- [2] 岡田眞也, 清岡優祐, 春山健太, 上田隆一, 林原靖男. ”視覚と行動の end-to-end 学習により経路追従行動をオンラインで模倣する手法の提案- “ 経路追従行動の修正のためにデータセットを動的に追加する手法の検討”. 計測自動制御学会 SI 部門講演会 SICE-SI2021 予稿集, pp.1066-1070(2021).
- [3] 清岡優祐, 岡田眞也, 岩井一輝, 上田隆一, 林原靖男. ”視覚と行動の end-to-end 学習により経路追従行動をオンラインで模倣する手法の提案- “ データセットと生成された経路追従行動の解析”. 計測自動制御学会 SI 部門講演会 SICE-SI2021 予稿集, pp.1072-1075(2021).
- [4] Yunpeng Pan, Ching-An Cheng, Kamil Saigol, Keuntaek Lee, Xinyan Yan, Evangelos A. Theodorou, and Byron Boots. ”agile autonomous driving using end-to-end deep imitation learning”. 30332–0250.
- [5] The mnist database of handwritten digits. <http://yann.lecun.com/exdb/mnist/>. 最終閲覧日 2022 年 12 月 30 日.
- [6] ros-planning, navigation. <https://github.com/ros-planning/navigation>. 最終閲覧日 2022 年 12 月 30 日.

## 付録

# 謝辞

本研究を進めるにあたり，1年に渡り，熱心にご指導を頂いた林原靖男教授に深く感謝いたします．