

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
(目標方向による経路選択機能の追加)

A proposal for an online imitation method of path-tracking
behavior by end-to-end learning of vision and action
(Addition of path selection function by target direction)

2022年1月24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

18C1096 春山 健太

概要

カメラ画像と目標方向を用いた end-end 学習による 分岐路でのルート選択可能な navigation 手法の提案

近年, カメラ画像に基づいた自律移動の研究が行われている. 本研究室でも,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, 実験により一定の経路を周回可能であると示されている. 本研究では, 従来手法をベースに, 目標とする進行方向 ("目標方向") をデータセットと学習器の入力へ加えることで, 分岐路で「直進」と「左折」などの経路を選択可能とする機能の追加を提案する. 提案手法では, LiDAR を用いた自律移動システムの出力をカメラ画像と目標方向を用いて模倣学習し, 学習後, カメラ画像と目標方向に基づいて経路を選択可能な自律走行を行う. また, シミュレータを用いた実験により, 提案手法の有効性を検証した.

キーワード: end-to-end 学習, Navigation, 目標方向

abstract

An End-to-End Navigation Method for Route Selection on Branch Roads
Using Camera Images and Target Directions

In this paper,

keywords: End-to-end Learning, Navigation, Target Direction

目次

| | | |
|--------------|--------------------------------------|----|
| 第 1 章 | 序論 | 1 |
| 1.1 | 背景 | 1 |
| 1.2 | 目的 | 7 |
| 1.3 | 論文構成 | 7 |
| 第 2 章 | 要素技術 | 8 |
| 2.1 | Deep learning | 8 |
| 2.2 | end-to-end 学習 | 9 |
| 2.3 | Convolution Neural Network | 10 |
| 2.4 | 地図ベースの制御器 | 11 |
| 2.5 | 従来手法 | 12 |
| 第 3 章 | 提案手法 | 15 |
| 3.1 | 提案手法の概要 | 15 |
| 3.2 | 学習フェーズ | 16 |
| 3.3 | テストフェーズ | 17 |
| 3.4 | 目標方向 | 18 |
| 3.5 | ネットワーク構造 | 19 |
| 第 4 章 | 実験 | 20 |
| 4.1 | 実験目的 | 20 |
| 4.2 | 実験環境 | 20 |
| 4.3 | 実験方法 | 21 |

| | | |
|----------|-------------|----|
| 4.4 | 実験条件 | 21 |
| 4.5 | 実験 1 十字路 | 22 |
| 4.5.1 | 実験環境 | 22 |
| 4.5.2 | 評価 | 23 |
| 4.6 | 実験 2 -8 の字- | 24 |
| 4.6.1 | 実験環境 | 24 |
| 4.6.2 | 評価 | 26 |
| 第 5 章 結論 | | 27 |
| 謝辞 | | 28 |

図目次

| | | |
|-----|--|----|
| 1.1 | Training the neural network from [?] | 1 |
| 1.2 | Okada and others proposed method from [?] | 2 |
| 1.3 | A robot that follows a path using vision based from [?] | 2 |
| 1.4 | Cross road | 3 |
| 1.5 | Topological map from[?] | 4 |
| 1.6 | Scenario of the trajectory from [?] | 5 |
| 1.7 | End-to-end driving via conditional imitation learning from [?] | 6 |
| 1.8 | Overview of Seiya and others proposed method from [?] | 6 |
| 2.1 | Neual network | 8 |
| 2.2 | Structure of end-to-end learning | 9 |
| 2.3 | Alex net from [?] | 10 |
| 2.4 | Map-based controller | 11 |
| 2.5 | Learning phase of conventional method | 12 |
| 2.6 | Collects the rule-based actions the robot's actions from [?] | 13 |
| 2.7 | Test phase of conventional method | 14 |
| 3.1 | Learning phase of proposed method | 16 |
| 3.2 | Overview learning phase | 16 |
| 3.3 | Test phase of proposed method | 17 |
| 3.4 | Overview test phase | 17 |
| 3.5 | Target direction | 18 |

| | | |
|-----|---|----|
| 3.6 | Method network | 19 |
| 4.1 | Turtlebot3 waffle add 3 cameras | 20 |
| 4.2 | Course of experiment1 | 22 |
| 4.3 | Experiment1 dataset | 23 |
| 4.4 | Experiment2 course | 24 |
| 4.6 | Experiment 2 route | 25 |
| 4.7 | Experiment2 dataset | 26 |

表目次

| | | |
|-----|---|----|
| 2.1 | camera | 13 |
| 3.1 | Target direction list | 18 |
| 3.2 | Parameters of deep learning | 19 |
| 4.1 | Number of successes experiment | 23 |
| 4.2 | Number of successes experiment2 | 26 |

第1章

序論

1.1 背景

近年，様々なセンサを用いた自律走行に関する研究が盛んに行われてあり，その中でカメラ画像を用いてロボットへ自律走行を行わせる研究も行われている。Bojaski ら [?] は Fig. 1.1 で示す方法で，人間のハンドル操作によるステアリングの角度の模倣学習を行い，画像を用いて走行を行う方法を提案した。

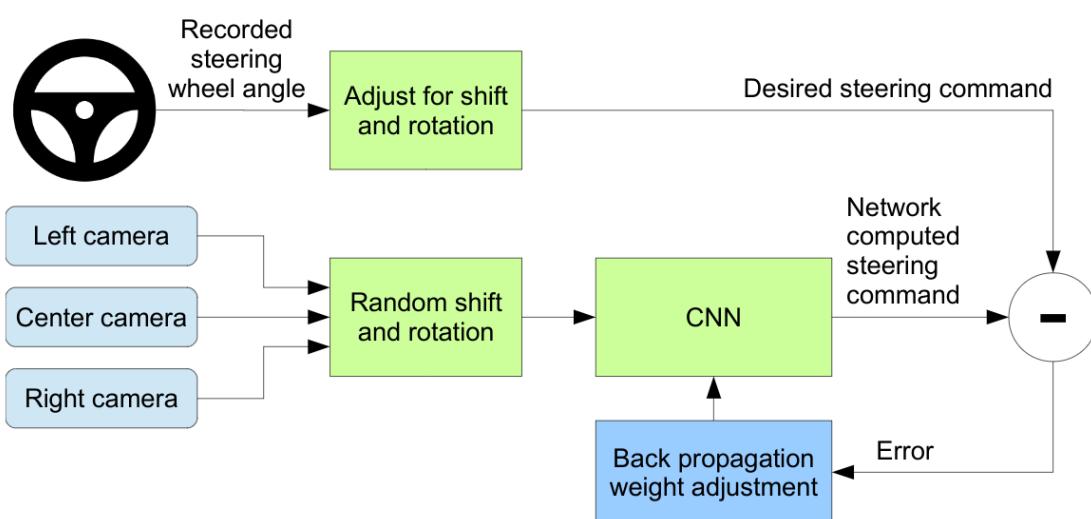


Fig. 1.1: Training the neural network from [?]

本研究室においても岡田ら [?] は Fig. 1.3 に示すような経路追従行動を、行動を模倣し、カメラ画像を用いた経路追従行動を獲得した。研究では Fig. 1.2 に示すように、LiDAR、オドメトリの入力としたルールベース制御器（後述する”地図ベースの制御器”）による経路追従行動を行い、その行動をカメラ画像を用いて end-to-end で模倣学習している。

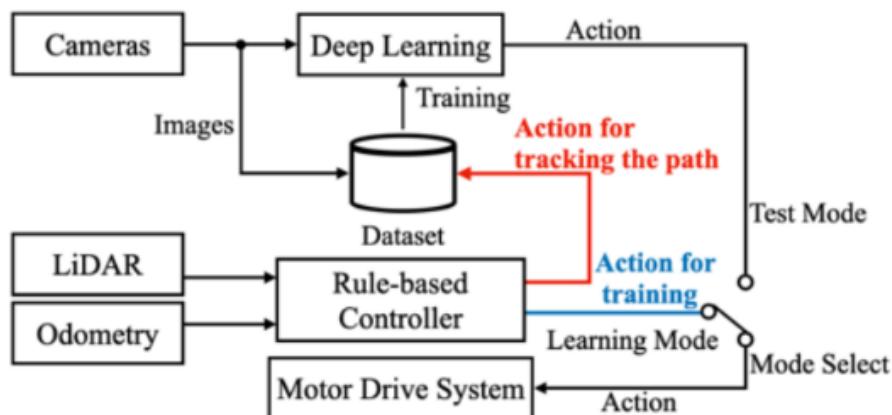


Fig. 1.2: Okada and others proposed method from [?]



Fig. 1.3: A robot that follows a path using vision based from [?]

上記の研究により、カメラ画像用いてロボットが学習した経路を周回可能であることが確認されている。次に岡田ら [?] の研究（本研究では”従来手法”と呼ぶ）をベースに、Fig. 1.4 のような分岐路において経路選択をする機能の追加を検討する。



Fig. 1.4: Cross road

島田らは人が目的地までの移動に用いる道案内の情報を、トポロジカルマップとシナリオを用いて表現する手法を提案している。トポロジカルマップとは Fig. 1.5 に示す通路の特徴とエッジの情報をもち、分岐路などの目印（エッジ）とそのつながり（ノード）を簡易的に表現した地図である、シナリオは 1.6 に示すトポロジカルマップを用いて、「次の角まで」のような「条件」と「直進」などの「行動」の組み合わせた経路の情報である。この研究により、「直進」「右折」などの情報で経路を選択することが可能であると示されている。

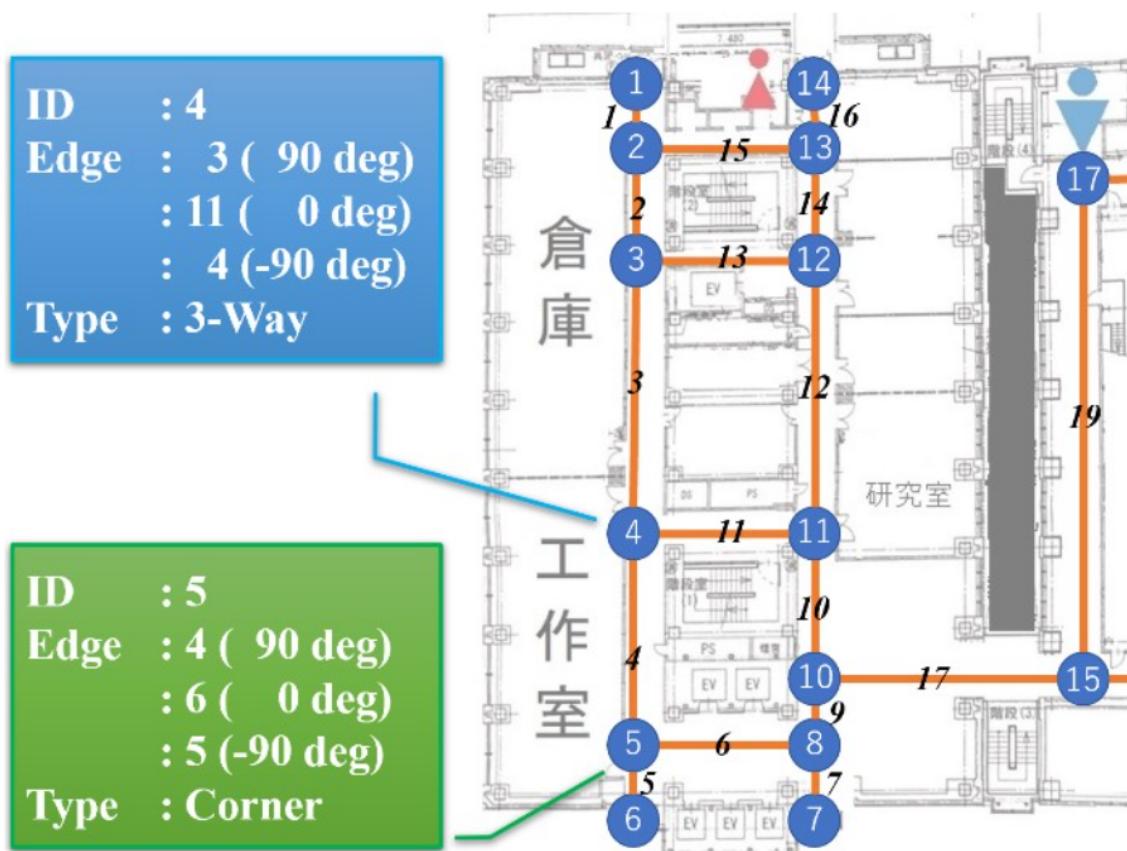


Fig. 1.5: Topological map from[?]

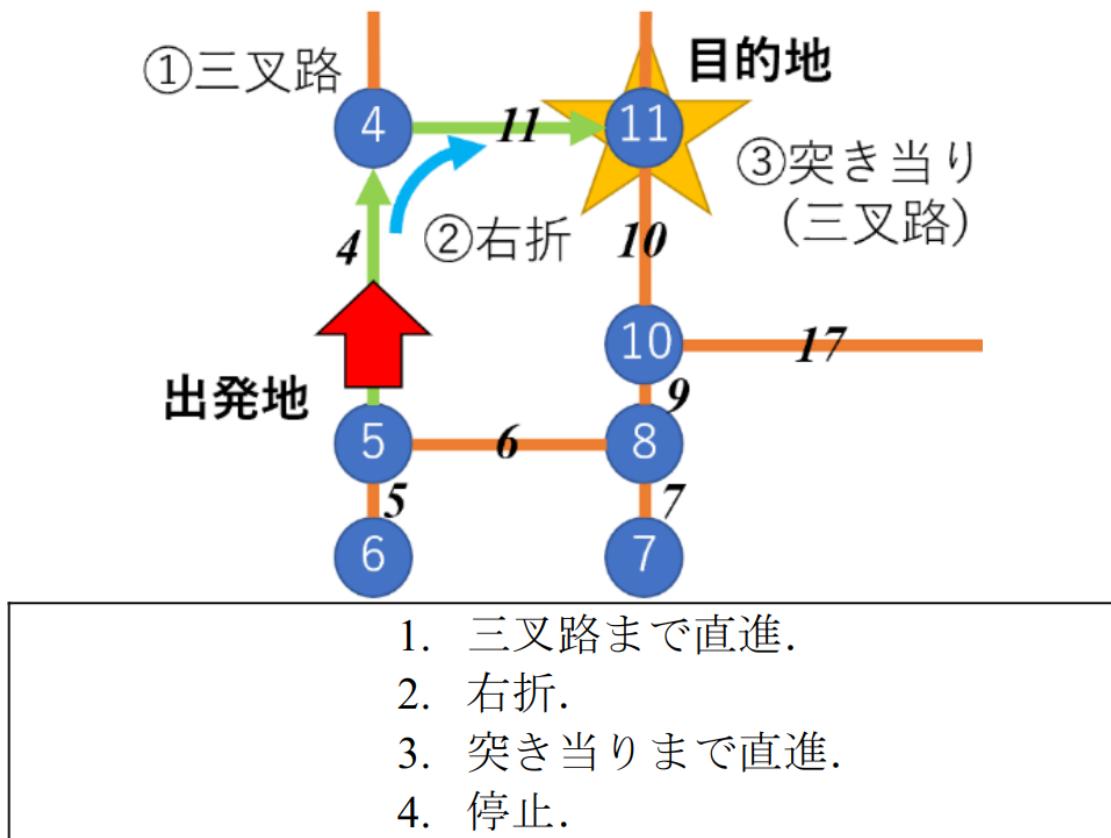


Fig. 1.6: Scenario of the trajectory from [?]

そこで、本研究では従来手法をベースに、「直進」「左折」「右折」などの目標とする進行方向の情報（本研究では”目標方向”と呼ぶ）をデータセットと学習器へ与える。これにより、学習器の出力を用いた走行において、目標方向によって、経路を選択可能とする機能の追加を提案する。

カメラ画像と他の情報を用いて、自律移動を行う研究として Felipe ら [?] はカメラ画像と操舵角と加速度の2次元の制御信号と、*continue*, *left*, *straight*, *right* の4つのコマンドを入力としたネットワークを用いて Fig. 1.7 で示すように実環境と都市環境のシミュレータ上で学習器がコマンドに沿った行動が可能であることを確認している。

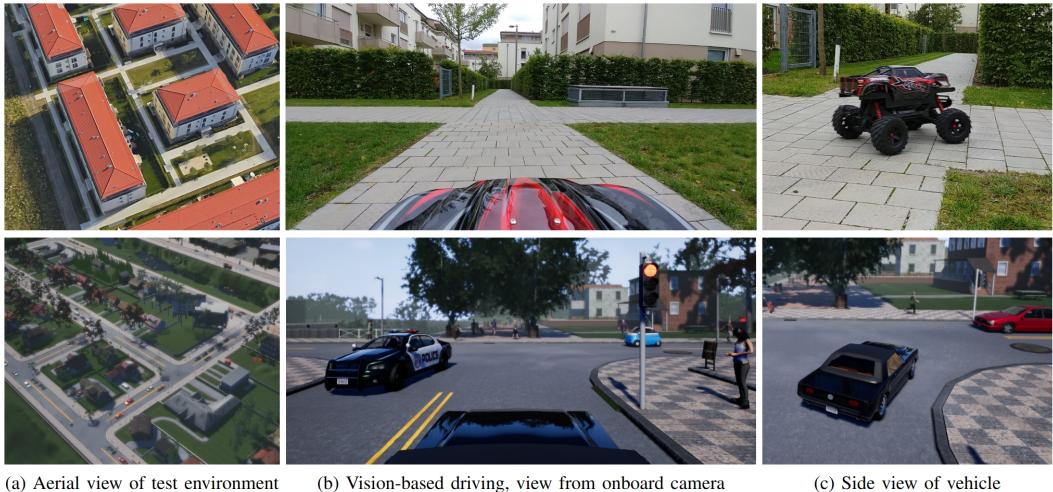


Fig. 1.7: End-to-end driving via conditional imitation learning from [?]

また、Seiya ら [?] は Fig. 1.8 で示すようにカメラ画像と目標方向を入力、ステアリング制御信号を出力とするシステムを用いて、右および左に曲がる屋外の軌道を追跡可能であることを確認している。

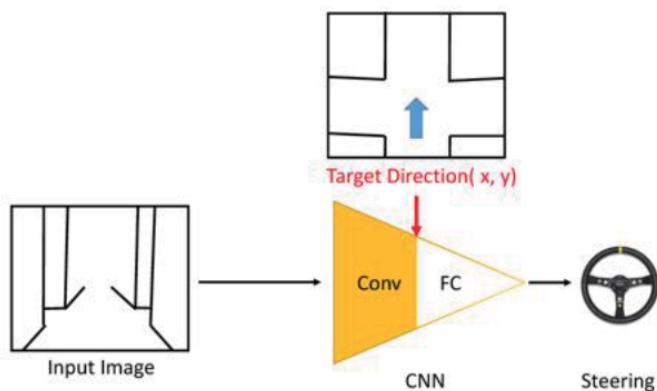


Fig. 1.8: Overview of Seiya and others proposed method from [?]

1.2 目的

本研究では、従来手法をベースに、分岐路で「直進」と「左折」などの経路を選択可能とする機能の追加を提案する。さらに、実験を行い、提案手法の有効性を検証することを目的とする

1.3 論文構成

1章では、本研究における背景、及び目的を述べた。2章では、本研究で用いた深層学習の要素技術とベースとする従来手法について述べる。3章では、本研究で用いた手法と構築したシステムについて述べる。4章では、構築したシステムを用いた実験を行う。5章では、本研究の結論を述べる。

第2章

要素技術

本章では、本研究で用いた深層学習に関連した要素技術と、ベースとする従来手法について述べる。

2.1 Deep learning

Deep learning は近年、自然言語処理など様々な分野で利用されている。人間の脳におけるニューロンの構造を数理モデル（パーセプトロン）を用いて再現したニューラルネットワークを多層構造にすることで、複雑なタスクの解決に必要な関数の表現力を高めたものである。一般的な構造を Fig. 2.1 示す。

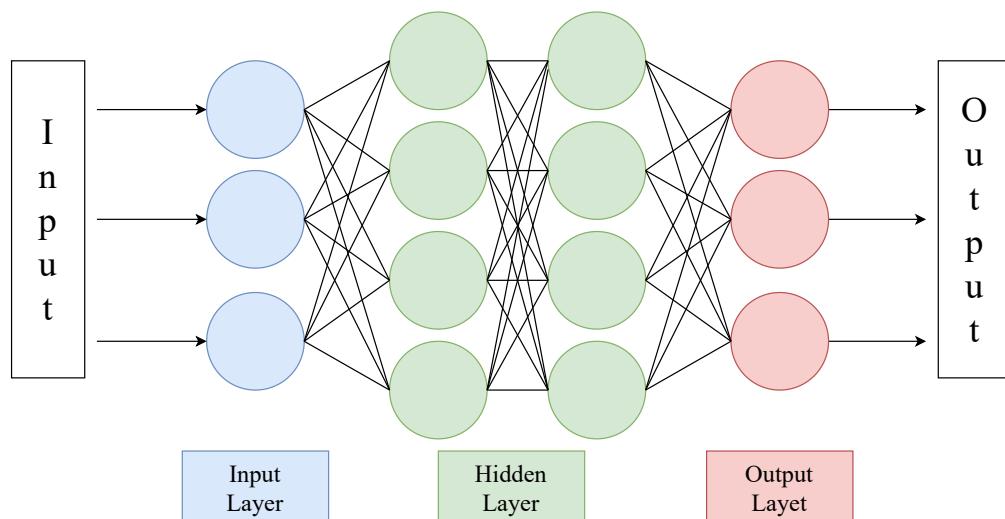


Fig. 2.1: Neural network

2.2 end-to-end 学習

end-to-end とは「端から端まで」という、Fig. 2.2 に示すように生データ（入力）から目的の結果（出力）を得るために必要な多段階の処理を Neural Network を用いて直接学習を行うものである。例として、実世界における自動運転では人物や障害物などの物体認識、走行レンンの検出、経路計画、ステアリング制御などの人間が設定した複数個のタスクを解く必要があるが、End-to-End 学習では先程のタスクを人間が直接設定せずにカメラ画像をニューラルネットワークに入力することで直接ステアリング操作を学習する。

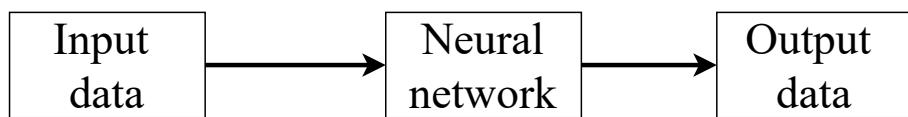


Fig. 2.2: Structure of end-to-end learning

2.3 Convolution Neural Network

畳み込みニューラルネットワーク (convolutional neural network:CNN) は画像認識や音声認識などの多次元の配列で表される複雑なデータを扱う処理へ用いられているニューラルネットワークである。次のような特徴をもつ層で構成されている。

1. 畳み込み層：入力データへフィルタ（カーネル）を用いて畳み込み演算を行うことで、特徴を抽出した特徴マップを取得する
2. プーリング層：Max プーリングなどの演算を用いて特徴マップのダウンサンプリングを行う
3. 全結合層：畳み込み層、プーリング層で行った特徴を抽出したデータを一つのノードに集約を行い、分類などの結果の出力をを行う

Krizhevsk ら [?] は Fig. 2.3 で示すような 3 つの畳み込み層 (convolutional layer), 2 つのプーリング層 (pooling layer) および 3 つの全結合層 (fully connected layer) から構成したネットワークを用いて、120 万の高解像度画像を 1,000 の異なるクラスに分類をエラー率 15.3% で達成し、画像分類コンペティションである ILSVRC (ImageNet Large Scale Visual Recognition Competition) 2012 で優勝をおさめた

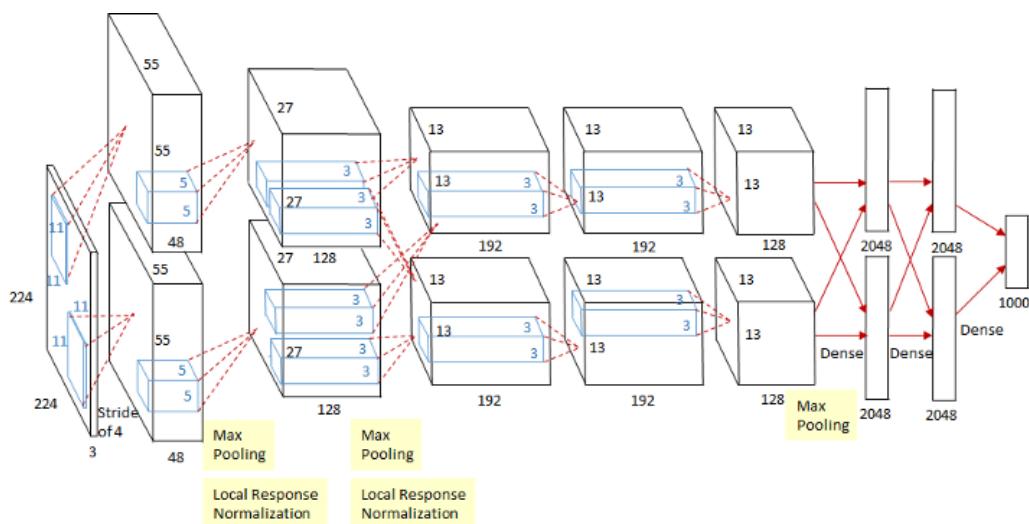


Fig. 2.3: Alex net from [?]

2.4 地図ベースの制御器

従来手法と提案手法において教師信号として用いる地図ベースの制御器について述べる。地図ベースの制御器は、ROS Navigation_stack[?] へ目標への経由地点 (waypoint) の指示を行う waypoint_nav を組み合わせたものである。ROS navigation_stack は 2.4 に示すように、LiDAR とオドメトリを用いて、事前に作成した地図上の自己位置推定を Particle_Filter によって確率的に行う「amcl」。障害物認識などの局所的または地図全体の大域的なコスト計算と、その結果を用いて経路計画、それに従ったモータ指令を行う「move_base」などのパッケージを統合した自律走行用のメタパッケージである。

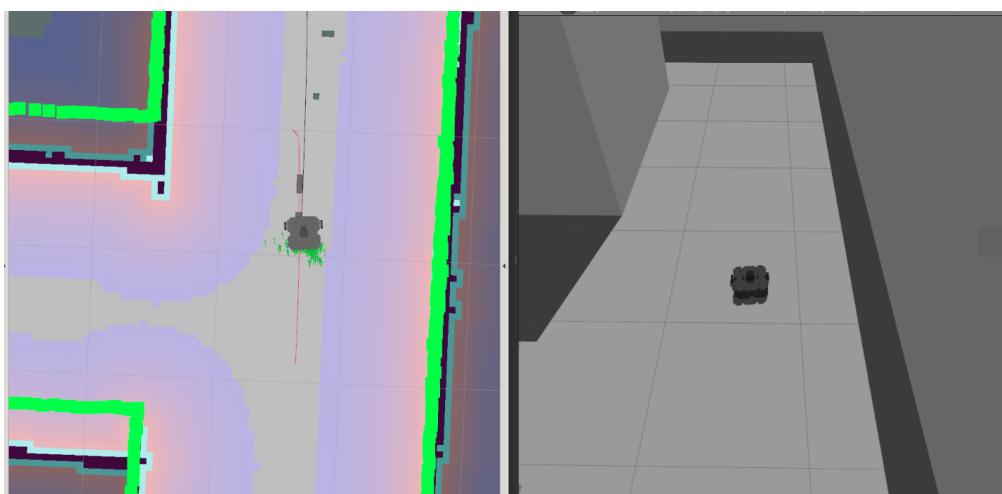


Fig. 2.4: Map-based controller

2.5 従来手法

本研究のベースとする岡田らの研究について述べる。従来手法は、LiDAR を用いた走行を学習し、同じ行動を画像も用いて行う手法であり、学習器の訓練を行う「学習フェーズ」と、学習器の出力を用いて走行を行う「訓練フェーズ」の 2 つにわけられる。なお、2 つのフェーズでの並進速度は固定した同じ値を用いる。学習フェーズを 2.5 に示す。LiDAR とオドメトリを入力する地図ベースの制御器により自律移動をカメラ画像を用いて模倣する。学習器は入力をカメラ画像、出力を自律走行時の角速度として end-to-end 学習を行う。

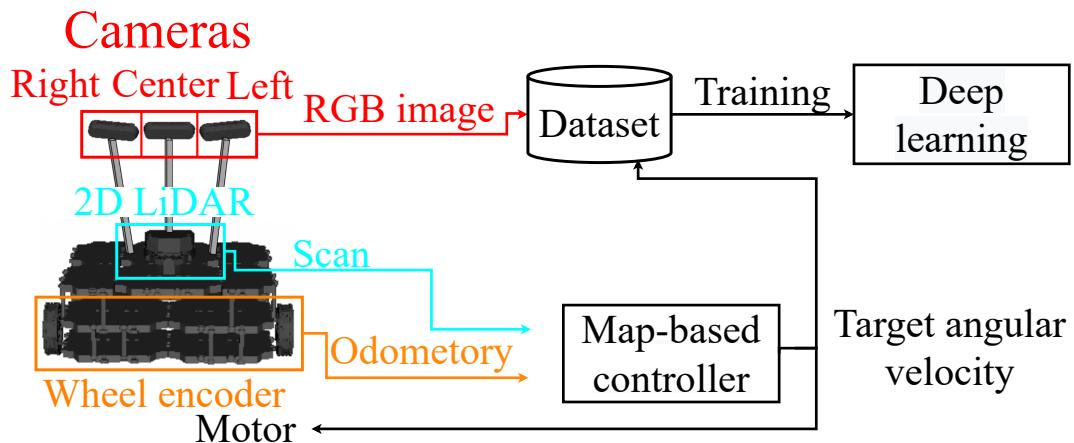


Fig. 2.5: Learning phase of conventional method

学習フェーズでは機体の中央、左、右に傾けて取り付けた3つのカメラを用いる。その際、直進時に、同時に旋回のデータを取得すること及び過学習の抑制を目的としてTable2.1に示すような処理を行う。

Table 2.1: camera

| | |
|---------------|--|
| Left camera | Angular velocity of Map-based controller + 0.2 rad/s |
| Center camera | Angular velocity of Map-based controller |
| Right camera | Angular velocity of Map-based controller - 0.2 rad/s |

また、地図ベースの制御器を教師信号とすることで、学習器の訓練に用いるデータセットと2.6に示すような経路へ戻る行動を自動的に収集可能としている。

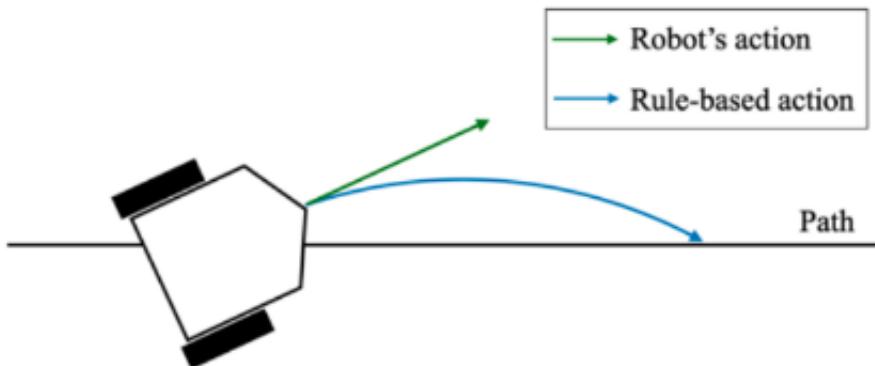


Fig. 2.6: Collects the rule-based actions the robot's actions from [?]

学習器の訓練後，2.7で示すテストフェーズへ移行する．テストフェーズでは，カメラ画像を入力する学習器の出力する角速度を用いて自律移動を行う．なお，テストフェーズでは中央のカメラのみを用いる．

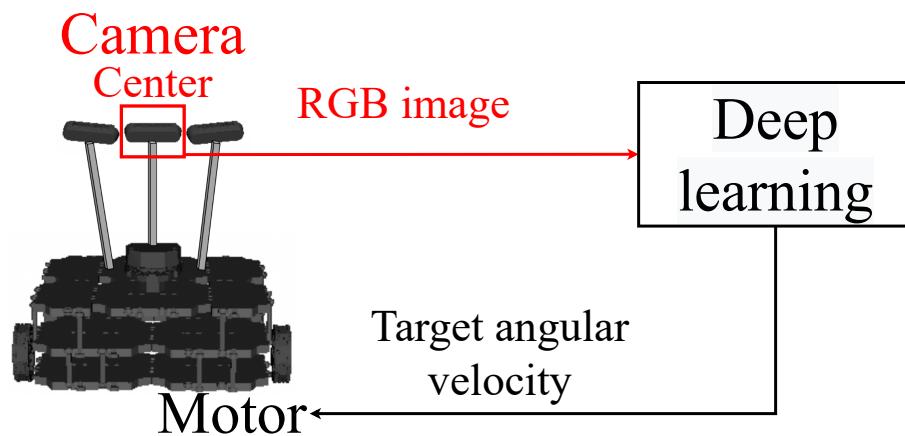


Fig. 2.7: Test phase of conventional method

第3章

提案手法

本章では、従来手法をベースとする提案手法を提案手法の概要、提案手法における学習フェーズ、テストフェーズと用いた目標方向についての4節に分けて述べる。

3.1 提案手法の概要

従来手法で用いたデータセットと学習器の入力へ、「直進」「左折」などの目標方向指令を追加することで、学習器の出力による走行において、経路を選択する機能の追加を行った。なお、追加した要素以外は従来手法と同様である。

3.2 学習フェーズ

学習器の訓練を行う学習フェーズで用いるシステムを Fig. 3.1 示す。経路追従行動を行う地図ベースの制御器へ目標方向の生成機能と、データセットへ目標方向の追加を行った。提案手法では、3.2 に示すように LiDAR とオドメトリを入力とする地図ベースの制御器による経路追従行動を、カメラ画像と目標方向を用いて模倣学習する。

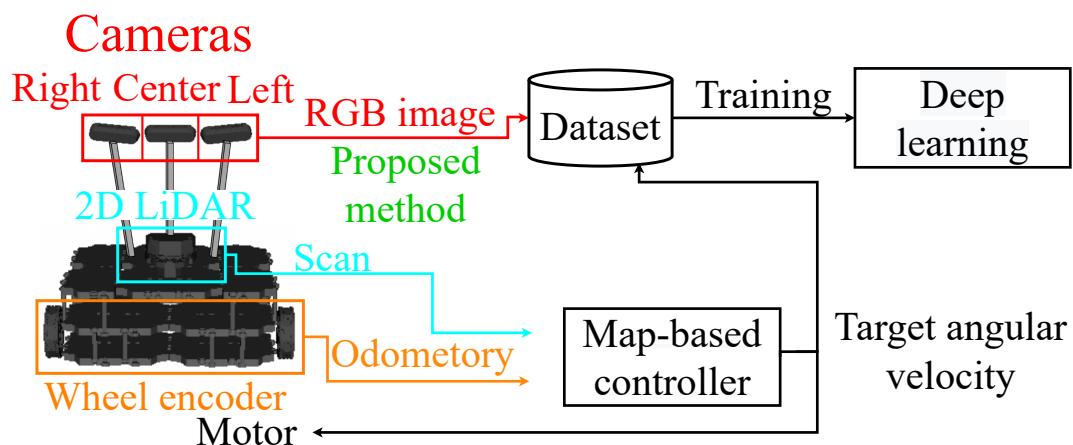


Fig. 3.1: Learning phase of proposed method

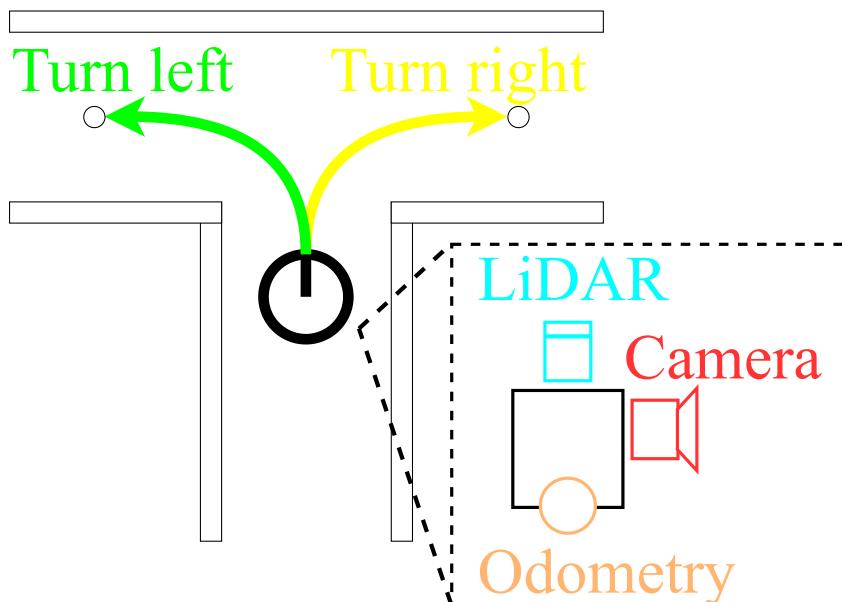


Fig. 3.2: Overview learning phase

3.3 テストフェーズ

提案手法におけるテストフェーズでは Fig. 3.3 で示すように、学習器の入力へ目標方向を追加した。3.4 に動作の様子を示す。カメラ画像と目標方向を用いた学習器の出力による走行において、目標方向によって経路の選択を行う。

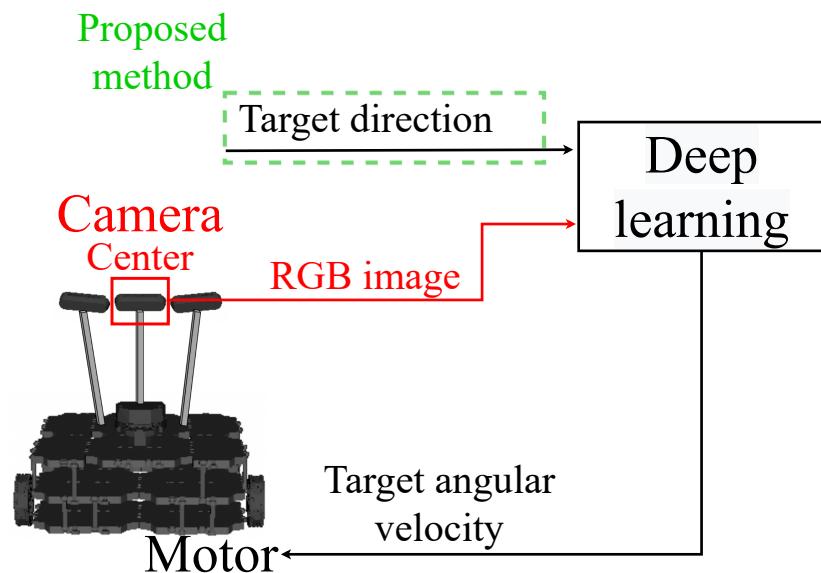


Fig. 3.3: Test phase of proposed method

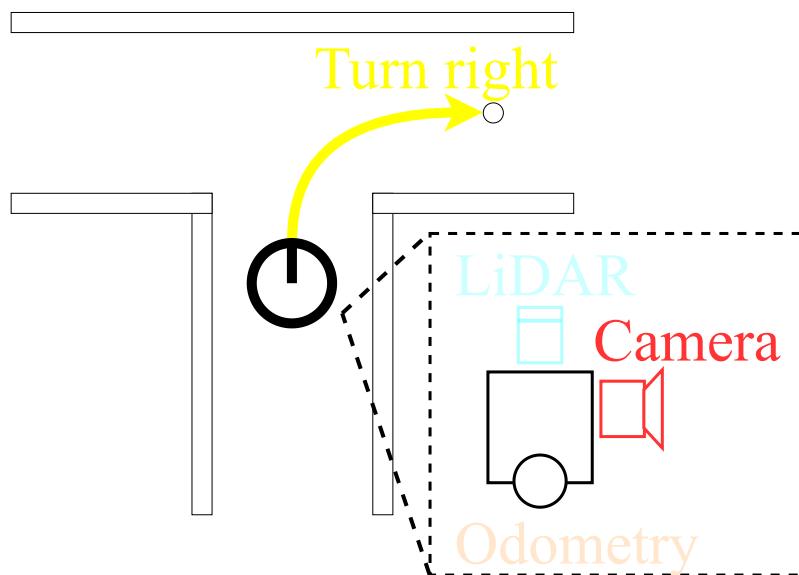


Fig. 3.4: Overview test phase

3.4 目標方向

本研究で用いた目標方向と、そのデータ形式である目標方向指令について述べる。目標方向は Fig.3.5 に示す。経路を「道なり」に走行 (Continue) 分岐路において「直進 (Go straight)」「左折 (Turn left)」「右折 (Turn right)」の 4 つとする。

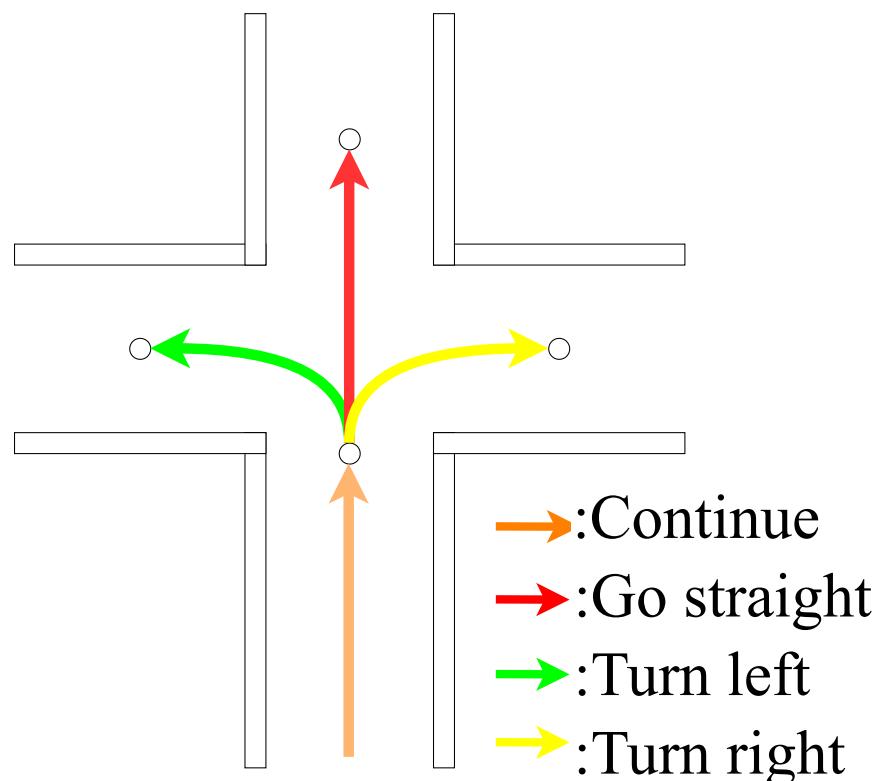


Fig. 3.5: Target direction

学習器には、上記の 4 つの目標方向を、要素数 4, 次元数 1 の int 型の配列 (One-Hot ベクトル) で表現した”目標方向指令”を入力する。目標方向指令のデータ形式を Table 3.1 に示す。

Table 3.1: Target direction list

| Target Direction | Continue | Go straight | Turn left | Turn right |
|------------------|----------------|----------------|----------------|----------------|
| Data | [100, 0, 0, 0] | [0, 100, 0, 0] | [0, 0, 100, 0] | [0, 0, 0, 100] |

3.5 ネットワーク構造

今回の提案手法で用いた、カメラ画像と目標方向を入力とする学習器のネットワークを Fig. 3.6 に示す。また、ハイパーパラメータについて Table 3.2 に示す。64 × 48 の RGB 画像を入力とする入力層 1，畳込み層 3，全結合層 2 層を持つ 6 層の CNN と，CNN の出力と目標方向指令を入力する入力層 1，全結合層 2。出力層 1 の全 10 層の構造になっている。出力はヨー方向の角速度である。

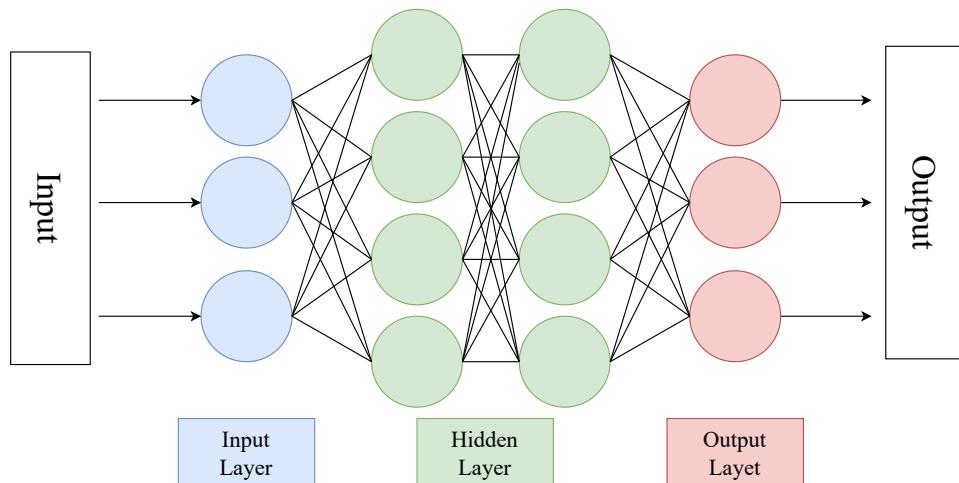


Fig. 3.6: Method network

Table 3.2: Parameters of deep learning

| | |
|---------------|---|
| Input data | Image (64x48 pixels, RGB channels) , Target direction |
| Optimizer | Adam ($\alpha = 0.001, \beta_1 = 0.9, \beta_2 = 0.999, \epsilon = 1e^{-1}$) |
| Loss function | Softmax-cross-entropy |
| Output data | Angular velocity |

第4章

実験

4.1 実験目的

シュミレータを用いて実験を行い，提案手法の有効性の検証を行う．

4.2 実験環境

実験はシミュレータ上で行い，シミュレータ環境としてオープンソースの3DロボットシミュレータGazebo[?]を用いる．実験装置として，turtlebot3_waffle[?]へカメラを3つ追加したモデルFig. 4.1を用いる．



Fig. 4.1: Turtlebot3 waffle add 3 cameras

4.3 実験方法

実験における手順を下記に示す

1. 設定した経路を学習フェーズで走行し，学習器の訓練（経路の学習）行う
2. テストフェーズへ移行，各経路を設定した回数走行する．

なお，学習器の訓練は「訓練データ（カメラ画像，目標方向指令）を学習器へ入力し，結果を出力」を1stepとする．また，データセットの収集及び，学習器への入力は0.25[s]周期で行い，走行に用いる並進速度は0.2[m]とする．

4.4 実験条件

実験条件はテストフェーズにおいて，

1. 成功：壁に衝突せず，指定した目標地点へ到達
2. 失敗：目標方向指令とは異なった経路を選択，または壁に衝突

とした．

4.5 実験1 十字路

4.5.1 実験環境

Fig. 4.2 に環境と経路、入力する目標方向を示す。環境は 2.5m 幅の十字路の環境を用いる。経路は下記の手順を 6000[step] 学習するまで、繰り返し走行する。なお B,C,D の Target point に到達次第、Start の位置へロボットの位置、姿勢を移動させる。

1. Start - A - Target point(B)
2. Start - A - Target point(C)
3. Start - A - Target point(D)

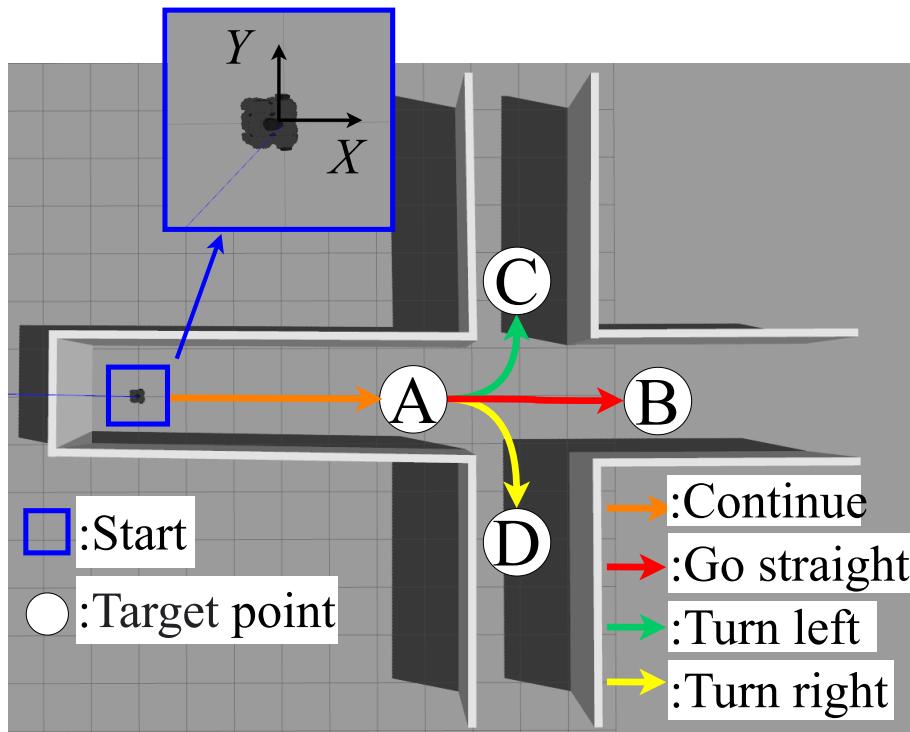


Fig. 4.2: Course of experiment1

4.5.2 評価

テストフェーズで、各経路を 10 回ずつ走行した結果を Table 4.1 に示す。すべての経路において、目標地点へ到達することに成功したまた、学習フェーズで収集したデータセットを Fig. 4.3 に示す。経路内に多い Continue は突出している。それ以外の 3 つはほぼ同数であることが確認できる。

Table 4.1: Number of successes experiment

| Route | Number of successes |
|---------------|---------------------|
| Start - A - B | 10/10 |
| Start - A - C | 10/10 |
| Start - A - D | 10/10 |

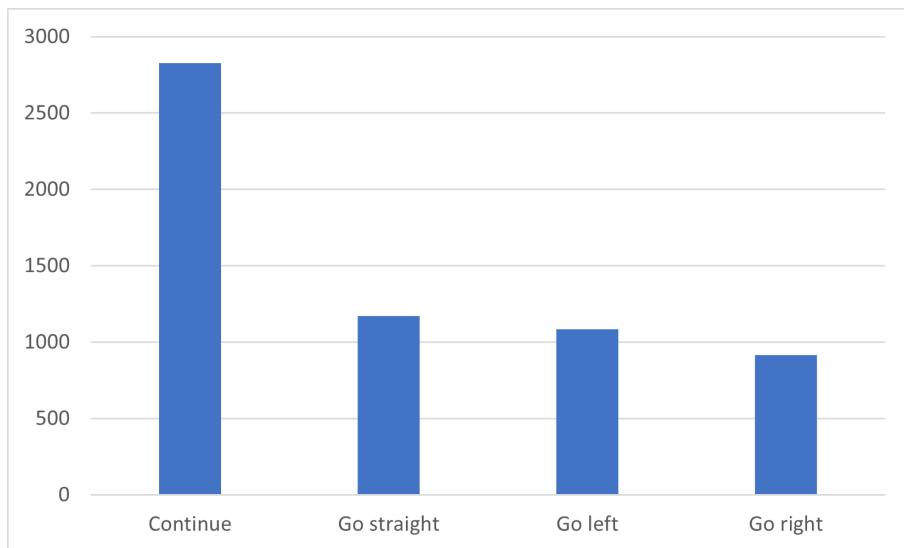


Fig. 4.3: Experiment1 dataset

4.6 実験2-8の字-

4.6.1 実験環境

Fig. 4.4 に示した、縦 8[m] × 横 12[m] で道幅が 2.5[m] の 8 の字型の環境を用いる

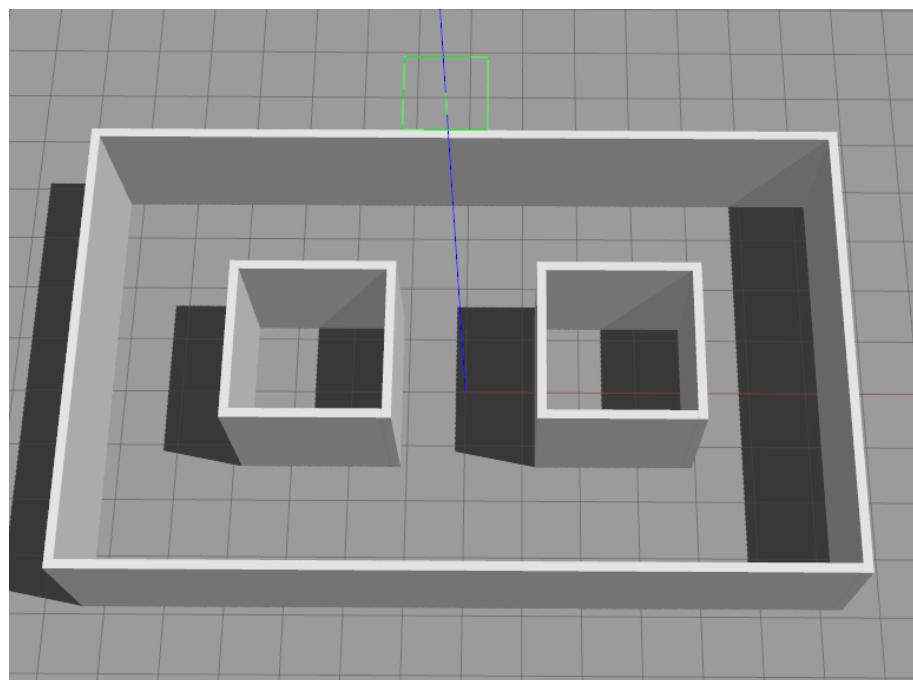
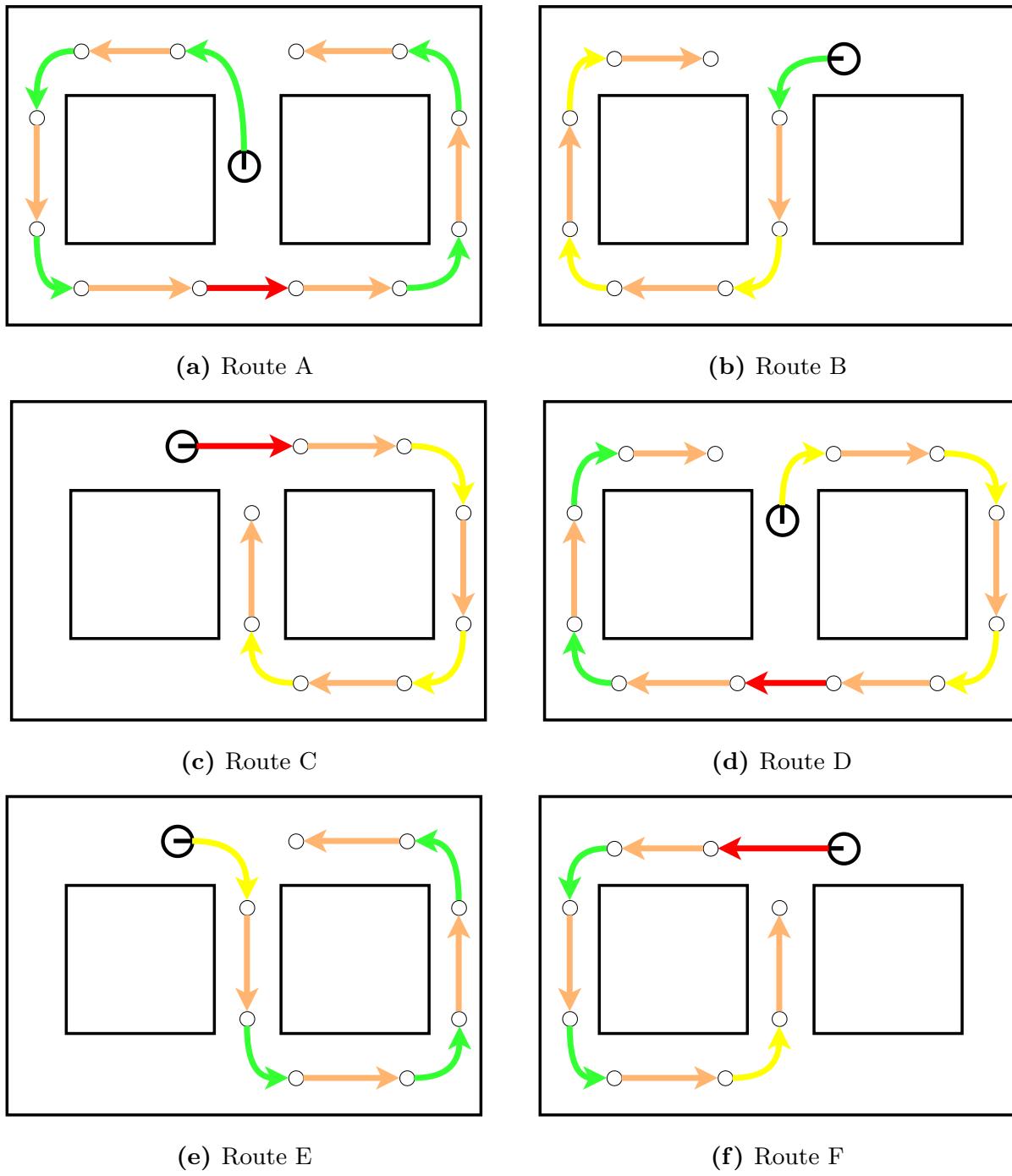


Fig. 4.4: Experiment2 course

Fig. 4.6 に経路と、入力する目標方向を示す。これらの環境を網羅するような経路を、Route A-F の順で 60000[step] 学習するまで、繰り返し走行する。



→:Continue →:Go straight →:Turn left →:Turn right

Fig. 4.6: Experiment 2 route

4.6.2 評価

テストフェーズで、経路をランダムに 60 回走行を行った結果を Table4.2 に示す。結果として経路を約 8 割走行に成功した。また、学習フェーズで収集したデータセットを Fig. 4.7 に示す。Go left, Go right はほぼ同数であるが、Go straight は約半分以下となった。しかし、Go straight を用いた、経路成功は問題なく行うことができた。これは、Go straight が学習する角速度が、Continue と類似しているからだと考えられる。

Table 4.2: Number of successes experiment2

| Route | Number of successes |
|-------|---------------------|
| A-F | 50/60 |

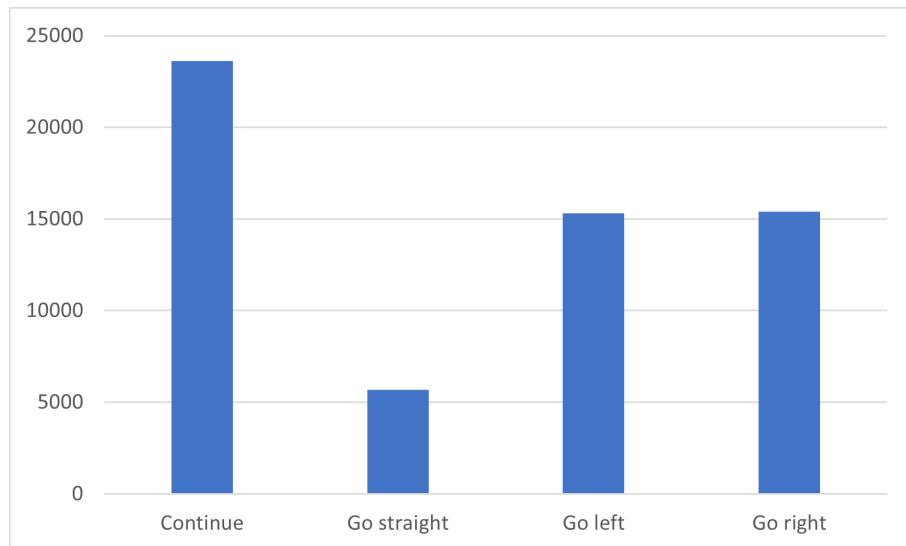


Fig. 4.7: Experiment2 dataset

第5章

結論

本研究では、経路追従行動をカメラ画像を用いた end-to-end 学習で模倣する岡田ら [?] らの従来手法をベースに、データセットと学習器の入力へ目標方向を加えることで、経路選択をする機能の追加を提案した。またシミュレータ上で十字路、8 の字の環境を用いた実験を行い、有効性の検証行った。実験結果より、学習器へ目標方向を入力することで、経路選択が可能であることが示され、提案手法の有効性が確認できた。

謝辞

本研究を進めるにあたり，，熱心にご指導していただいた林原靖男教授に深く感謝いたします。また，研究の礎や研究へのアドバイスなどの様々な面で，指導，サポートしてくださった岡田眞也様，清岡優祐様には返しきれぬ恩をいただきました。日々の生活の中で，議論や意見をしていただいたロボット設計制御研究室の皆様と精神的に辛い際の支えであり，生活に潤いをくださった私の彼女へ感謝いたします。

最後に私を育てていただいた両親へ謝意を表します。