

視覚と行動の end-to-end 学習による経路追従行動の模倣 データセットを収集してオフラインで訓練する手法の検討

A proposal for 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-

学 高橋祐樹 (千葉工大) 白須和暉 (千葉工大) 学 藤原 柁 (千葉工大)
正 上田隆一 (千葉工大) 正 林原靖男 (千葉工大)

Yuuki TAKAHASHI, Chiba Institute of Technology, s19c1068aq@s.chibakoudai.jp

Kazuki SHIRASU, Masaki FUJIWARA,

Ryuichi UEDA and Yasuo HYASHIBARA, Chiba Institute of Technology

In this paper, we propose a method of training a robot by collecting data on and near a target path, based on a conventional method of imitation learning. In the proposed method, the robot is placed on and around the target path and data is collected. Then, the robot is trained off-line using the collected data. After learning, the robot runs according to the output of the trainer using camera images as input. This is performed on a simulator, and the effectiveness of the proposed method is verified through experiments. As a result, it is confirmed that the proposed method can run around the path.

Key Words: End-to-End Learning, Navigation, Offline

1 緒言

本研究室では、従来から end-to-end 学習器により、視覚に基づく経路追従行動をオンラインで生成する手法の提案を行ってきた。そして、その有効性を実験により検証してきた [1][2]。

視覚を入力とした end-to-end 学習により経路追従を模倣する手法はいくつかある。例えば、Bojarski ら [3] は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の手法を提案した。Jing Bi ら [4] はカメラ画像と人がコントローラで操作した入力を模倣学習することで、経路追従と歩行者追従において有効であることを示した。また、岡田ら [1][2] は、図 1 のような測域センサなどの幾何学的なデータを入力として生成された行動を、カメラ画像を入力とする行動に模倣することで経路追従行動を獲得した。この手法は、多くの研究 [3][4] が人の操作を模倣していたのに対して、自動制御の出力を模倣する点に特徴がある。

しかし、岡田らの手法 (以下「従来手法」と称する) では、データ収集及び学習を行う際にロボットを走らせるため、時間が必要となる。

そこで、本研究では従来手法を基にロボットを走らせることなく、目標経路上及びその周辺のデータを一度に収集してオフラインで訓練する手法を提案する。これにより、訓練に要する時間を短縮できることが期待できる。さらに訓練後に、カメラ画像を入力とした学習器の出力で自律走行させることで手法の有効性を検証することを目的とする。

また、本研究の手法を実ロボットに適用する際には、ロボットの進行方向に対して平行な方向にカメラを取り付けることで再現する。その際に、カメラがどの程度あれば経路追従が可能であることをシミュレータ上の実験により確認する。

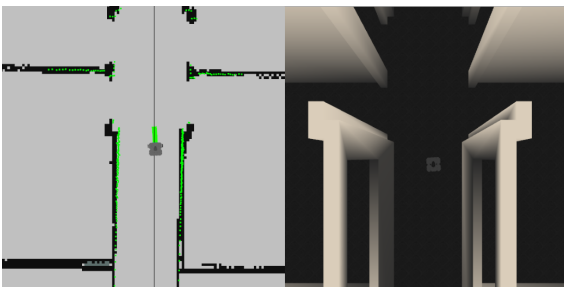


Fig.1: Moving with map-based navigation

2 従来手法

従来手法に関して述べる。従来手法では、地図を用いたルールベース制御器の出力を模倣し、経路追従行動を獲得する。図 2 に示すシステムでは、学習時に測域センサとオドメトリを入力とした navigation[5] の出力である角速度 (以下「目標角速度」と称する) を学習器とモータ駆動系に与える。学習器には、カメラ画像を 64×48 にリサイズした 3 つのカメラ画像 (RGB 画像) を入力し、目標角速度を出力して end-to-end 学習する。左右のカメラ画像に対する角速度には、それぞれ経路に戻るようなオフセットを加える。訓練後は、図 3 のようにカメラ画像を入力とした学習器の出力により走行する。なお、テストフェーズでは 3 台のカメラのうち、中央のカメラのみ使用する。

また、従来手法で用いたネットワークの構造を図 4 に示す。構造は入力層 1、畳み込み層 3、全結合層 2、出力層 1 の全 7 層から構成されている。また、オンラインで学習が行えるように、ネットワークは畳み込みニューラルネットワーク (CNN) を元としている。

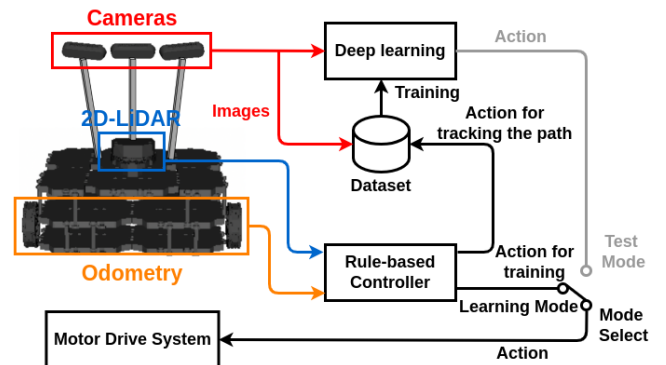


Fig.2: Learning phase

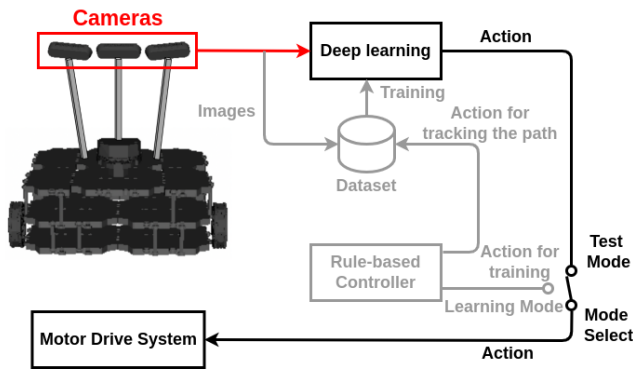


Fig.3: Test phase

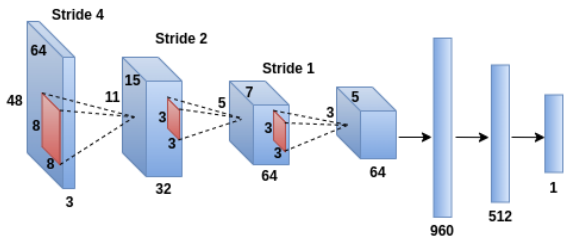


Fig.4: Structure of network

3 提案手法

本研究で提案する手法を述べる。従来手法に対して、提案手法は一度にデータを収集して、オフラインで訓練することが異なる。図5にデータの収集方法を示す。赤線の目標経路から平行に離れた座標にロボットを配置する。ロボットの進行方向に対する並進速度は 0m/s であるが、目標角速度はロボットに与えられる。そして、その座標ごとに 64×48 のカメラ画像 (RGB 画像) と目標角速度を図のように収集する。このように、ロボットを走行させることなく、目標経路上及びその周辺に配置することで、一度に大量のデータを収集することができる。その後、収集したデータを用いてオフラインで学習を行う。

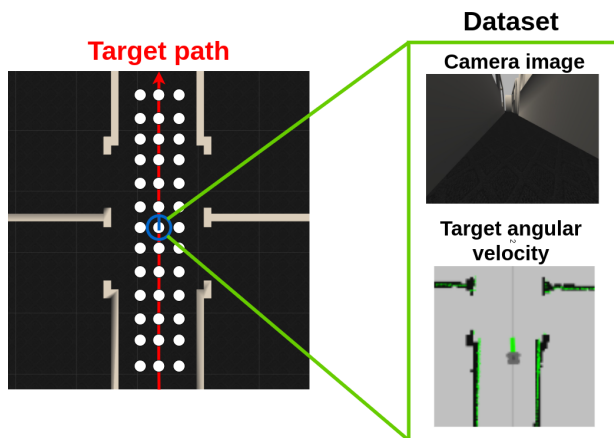


Fig.5: Method of collecting data around the target path

4 シミュレータを用いた実験

4.1 実験装置

シミュレータには Gazebo[6] の Willow Garage[7] を用いて、図6に示すコースで行う。また、ロボットモデルにはカメラを3つ搭載した Turtlebot3 Waffle Pi[8] を用いた。

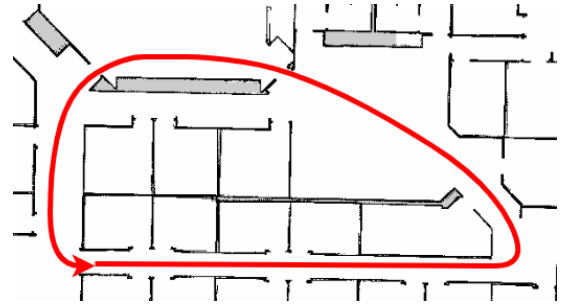


Fig.6: Course to collect data

4.2 実験方法

1. データ収集フェーズ 図7にデータの収集方法を示す。データ収集フェーズでは、従来手法と同様に3台のカメラを使用する。実験1では、赤線の目標経路上のみにロボットを配置する。実験2では、目標経路上には配置せずに、経路から平行に $\pm 0.1\text{m}$ 離れた座標に配置する。そして、実験3と実験4では、目標経路上と経路から平行に $\pm 0.1\text{m}$ 離れた座標にロボットを配置する。ただし、実験1から実験4のロボットの進行方向の配置間隔は 0.1m とする。そして、その座標ごとに経路に沿った向きを基準として ± 5 度傾けて、カメラ画像と目標角速度を収集する。
2. 訓練フェーズ 従来手法では、オンラインで学習を行うため、計算のリソースなどの観点などからミニバッチ学習を用いていた。しかし、本手法はオフラインで学習を行うため、バッチ学習を用いる。実験1から実験3は4000step、実験4は2000step 学習した。なお、4000step は従来手法において、シミュレータの実験に用いられてきたステップ数である。
3. テストフェーズ 図6に示したコースで10個の学習済みモデルを用いて走行させる。経路を3周できた場合を成功とし、壁に衝突したり、目標経路から 10m 以上離れたりした場合を失敗とした。また、並進速度は 0.2m/s で一定の速度をロボットに与える。ただし、テストフェーズでは従来手法と同様に3台のうち、中央のカメラのみを使用する。

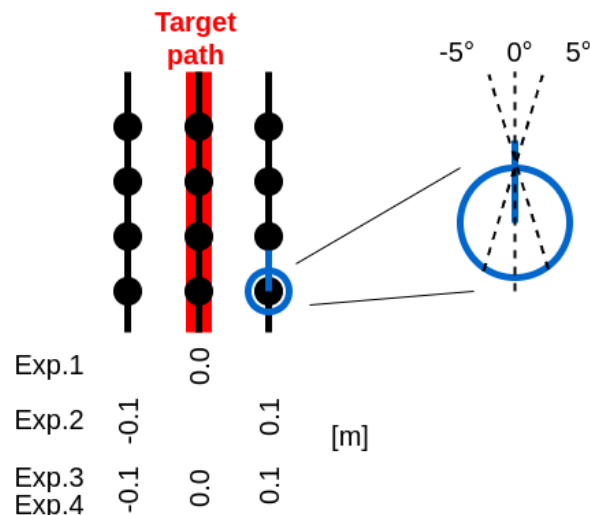


Fig.7: Method of collecting data around the target route

4.3 実験結果と考察

実験結果を表 1 に示す。実験 1 では、角を曲がりきれずにコースアウトすることがほとんどであった。これは、直進時の行動を学習する割合が実験 2 から実験 4 と比べて多く、左折するデータが少ないためと考えられる。実験 2 では、経路上のデータのみがないため、直進時にふらつきながら走行の様子が多く見られた。実験 4 は、実験 3 の半分のステップ数でも成功回数があり変わらなかった。ここで、学習時の loss の一例を図 8 に示す。図から実験 1 は、オーバーフィッティングしていると考えられる。また、実験 2 から実験 4 は学習が収束している様子が確認できる。これらのことから、学習時に経路上及びその周辺のデータを用いる方が、経路追従できる可能性が高いと考えられる。また、学習時に経路上のデータを用いなくても、概ね成功回数は変わらないことも確認できた。因みに、従来手法が訓練に最低 40 分程度必要であったのに対して、実験 3 は訓練時間が 4 分程度であるため、大幅に時間を短縮できることを確認した。

Table 1: Number of successes in the experiment

Experiments	Number of successes
Exp.1(4000step)	0/10
Exp.2(4000step)	9/10
Exp.3(4000step)	10/10
Exp.4(2000step)	9/10

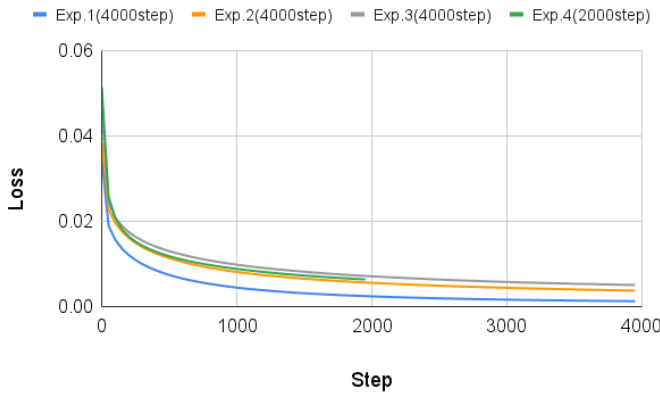


Fig.8: Loss value in the experiments

5 結言

本稿では、従来手法を基にロボットを走行させることなく目標経路及びその周辺のデータを一度に収集して、オフラインで訓練する手法を提案した。実験では、訓練時に経路上だけでなく、経路周辺のデータも訓練データに加えた方が経路追従できる可能性が高いことを確認し、手法の有効性を示した。また、従来手法と比べて訓練時間を大幅に短縮できることを確認した。

参考文献

- [1] 岡田 真也, 清岡 優祐, 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”, 計測自動制御学会 SI 部門講演会 SICE-SI2021 予稿集, pp.1147-1152, 2020.
- [2] 岡田 真也, 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] Bojarski, Mariusz, et al.: “End to End Learning for Self-Driving Cars.”, arXiv: 1604.07316, 2016
- [4] Jing Bi, Tianyou Xiao, Qiuyue Sun, Chenliang Xu. “Navigation by Imitation in a Pedestrian-Rich Environment”, arXiv:1811.00506, 2018
- [5] ros-planning, navigation リポジトリ <https://github.com/ros-planning/navigation> (最終閲覧日 2023 年 2 月 25 日)
- [6] gazebo リポジトリ <http://gazebo.org/> (最終閲覧日 2023 年 2 月 25 日)
- [7] Koenig, Nathan, and Andrew Howard. “design and use paradigms for gazebo, an open-source multi-robot simulator.”. 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)(IEEE Cat. No. 04CH37566). Vol. 3. IEEE, pp.2149-2154(2004). (最終閲覧日 2023 年 2 月 25 日)
- [8] Turtlebot3-robotis emanual.robotis.com/docs/. (最終閲覧日 2023 年 2 月 25 日)