

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 ー データセット収集密度の動的調整による学習の効率化 ー

A Proposal for an Online Imitation Method of Path-tracking Behavior by End-to-end Learning of Vision and Action

- Efficiency Improvement of Learning by Dynamic Adjustment of Dataset Collection Density -

○学 今井悠月 (千葉工大) 正 上田隆一 (千葉工大) 正 林原靖男 (千葉工大)

Yuzuki IMAI, Chiba Institute of Technology, s20c1015as@s.chibakoudai.jp

Ryuichi UEDA, Chiba Institute of Technology

Yasuo HAYASHIBARA, Chiba Institute of Technology

Our objective is to diversify the navigation methods in autonomous mobile robots. To achieve this, we have proposed a method that uses end-to-end learning of images and angular velocity to follow a path. This method involves collecting a dataset while the robot is in operation and using it for learning. However, the translational speed has not been discussed so far, and there is a possibility that data has been over-collected in some parts of the route. As a result, some data may not be learned, leading to a potential failure in path-following. Therefore, this paper first elucidates whether the cause of path-following failure is due to an imbalance in the dataset. Subsequently, we propose a method to adjust the data collection density by varying the translational speed of the robot, and we verify its effectiveness through experiments using a simulator.

Key Words: Navigation, Autonomous mobile robot, End-to-end learning, Dataset

1 緒言

本研究グループは、従来からカメラ画像とロボットの角速度を end-to-end 学習することで、経路追従する手法を提案し、その有効性を検証してきた [1][2][3]。これには、移動ロボットのナビゲーション手段を複数化するというねらいがある。本手法は、地図に基づく経路追従行動を、画像を入力とする行動に模倣する。地図に基づく経路追従とは、あらかじめ作成した地図と LiDAR、オドメトリなどを用いて、自己位置推定、経路計画、制御などの複数のタスクを実行し、経路追従する方法である。本手法を用いることで、ロボットは地図に基づく経路追従と同じ行動を、画像のみで行えるようになる。よって、地図に基づく経路追従と画像に基づく経路追従の 2 つのナビゲーション手段を獲得できる。これらを状況に応じて高い信頼性が見込まれる方を選択することで、経路追従を継続できる可能性が高まる。

本研究と同様に、カメラ画像を入力として、end-to-end 学習により経路追従する手法は、いくつか提案されている。例えば、Muller らは、人のコントローラ操作と画像を end-to-end 学習することで、オフロード環境で障害物を回避して走行できることを確認した [4]。また、Moridian らは、人のコントローラ操作と画像、測域センサのデータを end-to-end 学習することで、一定の経路を走行できることを確認した [5]。さらに、Bojarski らは、画像と人が操作したステアリングの角度を end-to-end 学習することで、自動運転する手法を提案した [6]。これらの手法はすべて人の操作を教師データとしているのに対し、本手法はルールベースの制御器の出力である角速度を教師データとしている。そのため、データセットの収集に人の操作が不要である。それに加えて、ロボットが常に目標経路に復帰する行動を収集できるため、人の操作を教師データとする場合よりも、経路追従を継続しやすいことが確認されている [7]。

ここで、本手法はロボットを走行させながら、各ステップ (0.2 s) ごとにデータセットを収集し、それを用いて学習するが、データセットを収集する密度に関してはこれまで議論されていない。従来は、ロボットの並進速度は常に一定 (0.2 m/s) としており、目標経路に対して、常に同じ密度でデータセットを収集している。しかし、目標経路の多くを占める直線的な経路では、過剰にデータを収集しているおそれがある。収集したデータセットは、各ステップごとに無作為に取り出され、それを学習に用いるため、割

合の多いデータは多く学習され、割合の少ないデータはあまり学習されていない。つまり、データの割合が少ない部分でも経路追従行動を得るには、多くの訓練時間を要する。よって、収集および学習されるデータの偏りを少なくし、学習を効率化することが望まれる。

これを踏まえ、本稿では、まず従来手法における実験結果の一例を取り上げ、収集および学習されるデータセットに偏りがあるかを確認する。そのうえで、データセットの収集密度を調整し、学習を効率化するために、ロボットの並進速度を教師データの値に応じて可変にする手法を提案する。なお、学習を効率化する手法は、これまでに本研究グループでもいくつか提案されている。例えば、藤原らは、オーバーサンプリングによりデータの不均衡を改善する手法を提案した [8]。また、高橋らは、データを事前に収集し、オフラインで学習する手法を提案した [9]。さらに、今井らは、モデルの評価に基づき訓練を打ち切る手法を提案した [10]。いずれの手法も訓練時間の短縮に成功しているが、ロボットの並進速度は常に一定としているため、本稿における提案手法とはアプローチが異なる。

本稿では、今井らの手法を採用し、シミュレータで実験する。ロボットの並進速度を可変にする場合と、そうでない場合で結果を比較することにより、提案手法の有効性を検証する。

2 end-to-end 学習により経路追従行動を模倣する手法

本研究グループが従来から提案する手法について述べる。本手法は end-to-end 学習を用いて、地図に基づく経路追従行動を、視覚を入力とする行動にオンラインで模倣する。はじめに地図に基づく経路追従を模倣学習する手法について述べた後、訓練済みモデルを用いた視覚に基づく経路追従手法について述べる。

2.1 地図に基づく経路追従の模倣学習

地図に基づく経路追従を模倣学習するシステムを図 1 に示す。学習時には、ルールベースの制御器となる ROS navigation パッケージ [11] を用いて、ロボットは自律移動する。それと同時に、 64×48 にリサイズした画像と ROS navigation パッケージから出力されるヨー方向の目標角速度をペアにしたデータセットを収集し、end-to-end で学習する。ROS navigation パッケージは、並進速度と角速度を出力するが、角速度のみを教師データとし、並進速度は常に一定 (0.2 m/s) としている。データセットの収

集と学習は一定時間 (0.2 s) ごとに繰り返すが、これを 1 ステップとしており、1 ステップにつき、収集したデータセットの中から 8 個のデータを取り出して学習する。なお、学習には、効率化の面から Experience Replay[12] を採用している。そのため学習に用いられるデータは毎回のステップで異なる。また、他の研究 [5][6] に倣い、データセットの収集には 3 つのカメラ (左・中央・右) を用いる。この時、表 1 のように、左右のカメラ画像に対応する目標角速度には、それぞれ目標経路に戻るようにオフセットを加える。具体的には、左のカメラには目標角速度に 0.2 rad/s を引いた値を、右のカメラには目標角速度に 0.2 rad/s を足した値を教師データとする。これにより、データの量を増やすことや、過学習を防ぐ効果が期待される。ただし、旋回時 (0.1 rad/s 以上) には、中央のカメラのみを使用する。ロボットのモータを制御するモジュールは、ROS navigation パッケージから出力される角速度と、学習器から出力される角速度の 2 つで構成されており、この 2 つを切り替えることが可能である。学習器の出力をベースに走行すると、学習ができていない部分でロボットは目標経路から外れることになるため、目標経路へ復帰する行動をより効率的に収集できる。ネットワークは、図 2 に示すように、左から順に、入力層 1、畳み込み層 3、全結合層 2、出力層 1 の全 7 層で構成されている。

Fig.1 Method to imitate path-tracking behavior

Cameras	Yaw angular velocity offset (rad/s)
Left camera	ROS navigation system output $- 0.2$
Center camera	ROS navigation system output
Right camera	ROS navigation system output $+ 0.2$

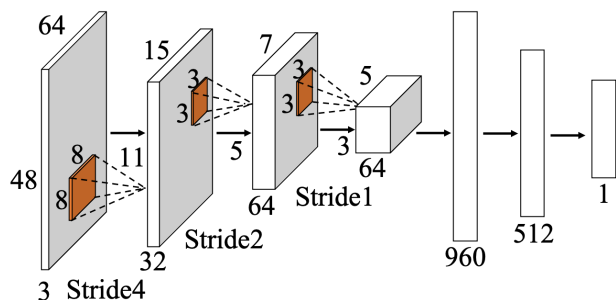


Fig.2 Structure of network

学習が終了したら、訓練済みモデルを用いて、経路追従できるかを確認する。このときのシステムを図 3 に示す。ここでは、ROS の navigation パッケージからの出力は使用せずにロボットを制御する。地図に基づく経路追従の模倣学習時と同様に、画像を学習器に入力し、その推定結果をヨー方向の目標角速度としてロボットを自律移動させる。ロボットのヨー方向の角速度には学習器の出力を用いるが、並進速度は常に一定 (0.2 m/s) とする。また、地図に基づく経路追従の模倣学習時には、3 つのカメラ (左・中央・右) を用いたが、訓練済みモデルを用いた視覚に基づく経路追従時には、3 つのカメラのうち、中央のカメラのみを使用する。

Fig.3 Vision-based path-tracking using learned model

本稿の実験で採用する、今井らが提案した手法について述べる。図 4 に本手法のシステムを示す。2.1 節で述べた地図に基づく経路追従の模倣学習手法に、訓練中のモデルを評価し、望む出力がある程度得られたと判定したら、訓練を終了する仕組みを追加している。具体的には、ROS navigation パッケージの出力である角速度と、学習器の出力である角速度を比較する。比較の結果、その差の絶対値が一定値 (0.05 rad/s) 未満であれば、その地点において望ましい出力が得られていると判定する。この点を DOL (desired output location) と呼び、判定をステップごとに繰り返す。データセットを収集しつつ、学習し、DOL の割合が一定値を超えたところで訓練を打ち切る。これにより、従来は一定のステップ数を用いていた訓練を、訓練の結果に応じて適切なステップ数で打ち切る。ただし、経路の一部でも DOL が少ないと経路追従できないため、DOL の割合は経路全体に対してではなく、経路を分割して算出する。以降、本手法を従来手法と呼ぶこととする。

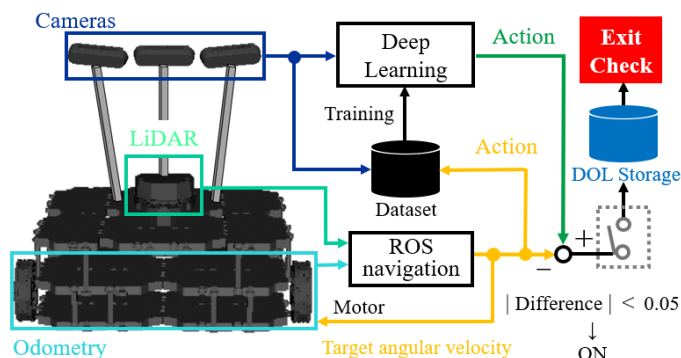


Fig.4 Method to terminate training

4 データセットの偏りの確認

従来手法では、すべての経路点間で DOL の割合が 33 % 以上であれば、99 % 経路追従に成功する様子が見られた [10]。ただし、目標経路に対して、常に同じ密度でデータセットを収集しているため、目標経路の多くを占める直線的な経路では、過剰にデータを収集しているおそれがある。経路の一部でも極端にデータの割合が多いと、割合の少ないデータはあまり学習されないため、訓練打ち切りの指標を満たすのに多くの時間を要する。そこで、再現実験を行い、それによって得た結果の一例から、収集および学習されるデータセットに偏りがあるかを確認する。

4.1 実験装置

実験はシミュレータで行う。シミュレータには Gazebo[13]、環境には Willow Garage を用いた。ロボットには TurtleBot3[14] にカメラを 3 つ取り付け付けたモデルを用いた。

4.2 実験方法

実験では、はじめに地図に基づく経路追従を模倣学習する。このとき、ロボットは図 5 に示すような緑色の経路点を順次移動するが、経路点の配置は先行研究と同じである。ここで、どの地点で得られたデータがどの程度収集されたかを確認するため、各経路点間に位置情報を加えたデータセットを収集する。図の赤で示した部分は、先行研究において経路追従に多く失敗した地点であることを意味する。ただし、ロボットは位置情報は学習せず、従来と同様に画像から角速度のみを予測する。学習後は、ロボットが収集および学習したデータセットの割合を可視化することで、偏りがあるかを確認する。

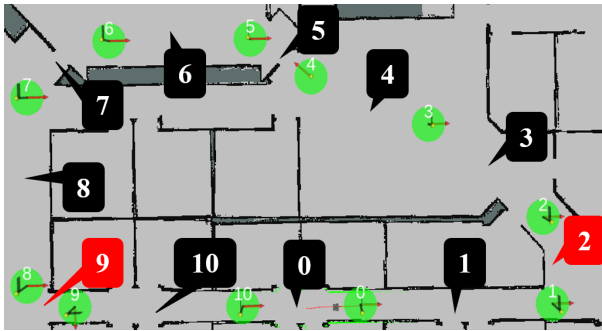


Fig.5 Waypoints used in the experiment

4.3 実験結果

図 6 に収集したデータセットの割合、図 7 に学習に用いられたデータセットの割合を示す。図 6 より、先行研究で経路追従に失敗した地点である 9 番のデータセットが他の地点と比べて少ないことがわかる。また、図 7 より、0 番や 1 番などの直線的な経路のデータが多く学習され、9 番のデータセットがあまり学習されていないことがわかる。よって、目標経路の多くを占める直線的な経路では過剰にデータを収集しており、その結果、割合の少ないデータがあまり学習されていないため、学習は効率的でないといえる。これらの結果から、収集および学習されるデータセットには偏りがあることが一例ではあるが確認された。

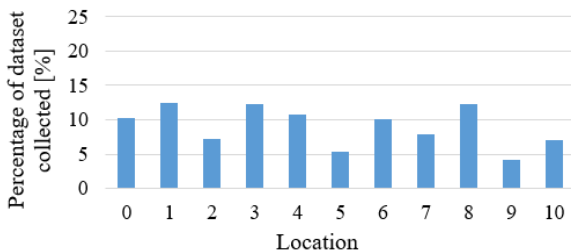


Fig.6 Percentage of dataset collected

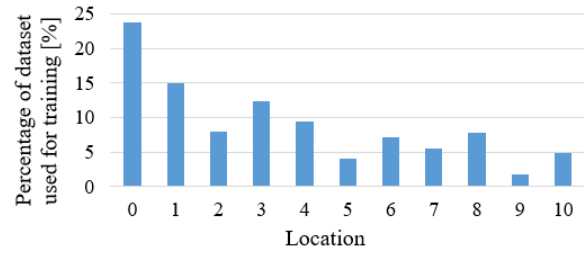


Fig.7 Percentage of dataset used for training

5 提案手法

4 章の結果から、収集および学習されるデータセットには偏りがある。よって、データセットの収集密度を調整する手法を提案する。図 8 に提案手法を示す。従来手法では、学習時のロボットの並進速度は常に一定 (0.2 m/s) であったが、教師データとなる ROS navigation パッケージの目標角速度の値に応じてロボットの並進速度を可変にする仕組みを追加している。具体的には、目標角速度の絶対値が 0.1 rad/s 未満であれば、ロボットの並進速度を従来の 2 倍 (0.4 m/s) とし、0.1 rad/s 以上であれば、従来と同じ (0.2 m/s) とする。なお、0.1 rad/s という値は、3 つのカメラを用いるしきい値に由来する。これにより、目標経路の多くを占める直線的な経路で過剰にデータを収集することを防ぐ効果が期待される。それに伴い、学習されるデータセットの偏りも改善されることで、訓練打ち切りの指標を比較的早い段階で満たすことができると考えられる。

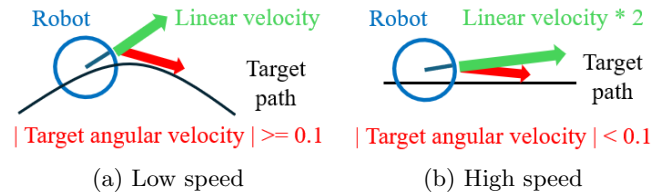


Fig.8 Proposed method

6 提案手法を用いた実験

6.1 実験装置、実験方法

提案手法の有効性を検証するため、実験を行う。実験装置や実験方法は 4 章と同様である。ただし、4 章の実験とは異なり、学習後は訓練済みモデルを用いて経路追従できるかを確認する。壁に衝突することなく、学習した経路を 1 周することができれば成功とみなし、100 回実験する。経路追従の成功率と訓練打ち切りステップ数、収集および学習したデータセットの割合から提案手法の有効性を評価する。

6.2 実験結果

6.2.1 経路追従の成功率

表 2 に経路追従の結果を示す。結果的に、提案手法を用いても、従来手法と経路追従の成功率は変わらなかった。ちなみに失敗した地点は、4.2 節の図 5 の 9 番であり、従来手法において失敗した地点と同じであった。

Table 2 Number of successes in experiments

Experiments	Number of successes
Conventional method	99/100
Proposed method	99/100

6.2.2 訓練打ち切りステップ数

図9に手法ごとの100回の訓練打ち切りステップ数の分布を示す。従来手法では、最短で4000から6000ステップの訓練を必要としたのに対し、提案手法では、最短で2000から4000ステップで訓練を終了しているものがある。つまり、提案手法を用いた方が訓練打ち切りの指標を早くに満たすことができるといえる。ちなみに、従来手法の平均ステップ数は約6262ステップ、提案手法の平均ステップ数は約4742ステップであり、平均で約1500ステップ（約5分）の訓練時間を短縮できた。6.2.1項の結果より、双方で経路追従の成功率は変わらないため、提案手法は経路追従の成功率を損なわずに訓練時間を短縮できるといえる。

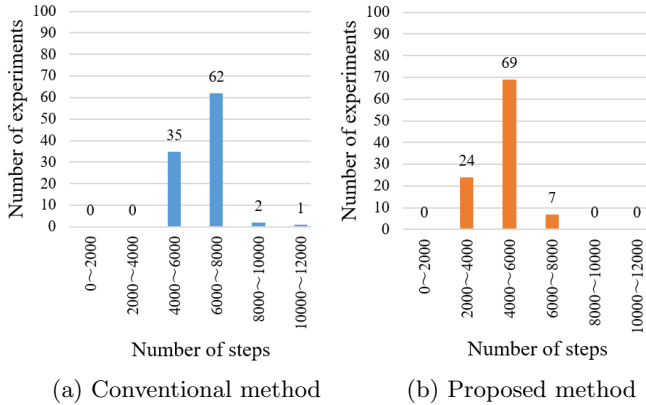


Fig.9 Distribution of the number of steps terminated

6.2.3 収集および学習したデータセットの割合

図10に収集したデータセットの割合、図11に学習に用いられたデータセットの割合を示す。これらは、従来手法と提案手法で成功した10回ずつの結果を平均したものである。図10、図11ともに、提案手法では0番や1番など、目標経路の多くを占める直線的な経路のデータの割合が少なくなり、2番や9番など、曲がり角周辺のデータの割合が多くなったことがわかる。つまり、データセットの収集密度が改善されたことで、学習に用いられるデータセットの偏りも改善することができたといえる。また、それにより学習を効率化することができたといえる。

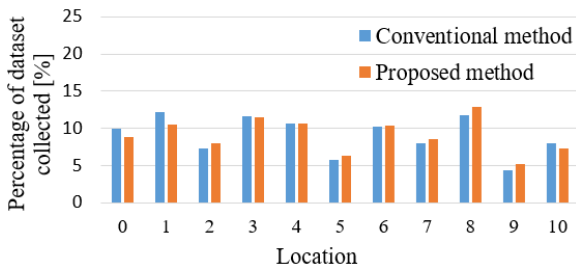


Fig.10 Percentage of dataset collected

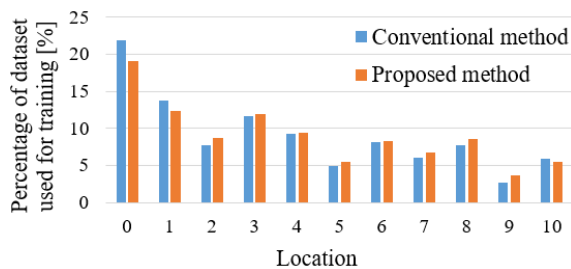


Fig.11 Percentage of dataset used for training

7 結言

本稿では、従来から提案する経路追従行動の模倣手法における実験結果の一例を取り上げ、収集および学習されるデータセットには偏りがあることを一例ではあるが確認した。そのうえで、データセットの収集密度を調整し、学習を効率化するために、ロボットの並進速度を教師データの値に応じて可変にする手法を提案した。その後、シミュレータでの実験により提案手法の有効性を検証した。結果的に、提案手法は従来手法に比べて、データの収集密度を改善し、学習に用いられるデータの偏りを改善できた。それにより、訓練打ち切りの指標を早くに満たすことができる傾向が見られたため、学習を効率化するうえで有効であることが確認された。

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-SI2020 予稿集, pp.1148-1152, 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.1066-1080, 2021.
- [3] 清岡優祐, 岡田真也, 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.1071-1075, 2021.
- [4] U. Muller, J. Ben, E. Cosatto, B. Flepp, and Y. Cun. “Off-Road Obstacle Avoidance through End-to-End Learning.” Advances in neural information processing systems, Vol. 18, 2005.
- [5] Moridian, Barzin, Anurag Kamal, and Nina Mahmoudian. “Learning Navigation Tasks from Demonstration for Semi-autonomous Remote Operation of Mobile Robots.” 2018 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR), pp.1-8, 2018.
- [6] Bojarski, Mariusz, et al., “End to end learning for self-driving cars,” arXiv:1604.08316, 2016.
- [7] 今井悠月, 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 -経路への復帰行動の解析と復帰行動を強化する教師データ収集法の検討-”, 日本機械学会ロボティクス・メカトロニクス講演会'23 予稿集, 2P2-G05(2023).
- [8] 藤原征, 春山健太, 馬場琉生, 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 -環境における経路選択機能の検証と学習時間の短縮化の検討-”, 日本機械学会ロボティクス・メカトロニクス講演会'23 予稿集, 2P2-G06(2023).
- [9] 高橋祐樹, 白須和暉, 藤原征, 上田隆一, 林原靖男, “視覚と行動の end-to-end 学習による経路追従行動の模倣”, 日本機械学会ロボティクス・メカトロニクス講演会'23 予稿集, 2P1-G07(2023).
- [10] 今井悠月, 落合拓海, 白須和暉, 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 -モデルの評価に基づく訓練ステップ数の決定-”, 1B1-04, SI2023, (2023).
- [11] ros-planning, navigation リポジトリ
<https://github.com/ros-planning/navigation>.
(最終閲覧日: 2024 年 3 月 2 日)
- [12] L. Lin, “Self-improving reactive agents based on reinforcement learning, planning and teaching”, Mach. Learn. 8, 293-321 (1992).
- [13] Nathan Koenig 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), Vol.3, pp. 2149-2154, 2004.
- [14] Erico Guizzo and Ackerman Evan, “The TurtleBot3 Teacher [Resources Hands On],” IEEE Spectrum, Vol.54, pp.19-20, August 2017.