

卒業論文

測域センサの反射強度を利用した視覚と行動の end-to-end 学習による人追従行動の模倣

Imitation-based end-to-end learning for human tracking behavior
using reflected intensity of a range sensor

2024 年 1 月 9 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
20C1102 馬場 琉生

概要

測域センサの反射強度を利用した視覚と行動の end-to-end 学習による人追従行動の模倣

近年，機械学習を活用した自律移動に関する研究が盛んに行われている。本研究室では，人追従ロボットに関する2つの手法を提案してきた。1つ目は，引き紐に接続されたリンクの角度を報酬として利用し，強化学習することで，画像に基づく人追従行動を生成する手法である。2つ目は，引き紐を入力とするルールベース制御器の出力を教師信号として深層学習器に与えることで，ロボットの人追従行動をオンラインで模倣する手法である。これらの手法は，実ロボットを用いた実験により，カメラ画像に基づいた人追従行動が可能であることを確認している。

従来の研究は，引き紐を用いて人追従行動を生成してきたが，測域センサの反射強度を利用することで同様の人追従行動を生成できる可能性がある。これにより，学習時のセンサの使用に新たな選択肢が加わる。本研究では，測域センサの反射強度を利用したルールベース制御器による人追従行動を生成する手法を提案し，その出力を教師信号としてカメラ画像に基づいた人追従行動が可能かどうかを，実ロボットを用いた実験により提案手法の有効性を検証する。

キーワード：人追従，end-to-end 学習，移動ロボット

abstract

Imitation-based end-to-end learning for human tracking behavior
using reflected intensity of a range sensor

keywords: Person following, End-to-end learning, Mobile robot

目次

第 1 章	序論	1
1.1	背景	1
1.2	目的	4
1.3	関連研究	5
1.4	論文の構成	7
第 2 章	要素技術	8
2.1	深層学習	8
2.1.1	end-to-end 学習	9
2.1.2	ミニバッチ学習	9
2.1.3	Convolutional Neural Network (CNN)	10
2.2	ROS	12
2.2.1	LiDAR	12
2.2.2	RViz	13
第 3 章	従来手法	14
3.1	学習フェーズ	14
3.2	追従フェーズ	15
3.3	ネットワークの構造	16
第 4 章	提案手法	17
4.1	提案手法の概要	17
4.2	学習フェーズ	19

4.3	追従フェーズ	20
4.4	ルールベース制御器で用いる装置	21
4.5	ルールベース制御器	22
4.6	ネットワーク構造	24
第 5 章	実験	25
5.1	実験の概要	25
5.2	実験装置	27
5.3	実験 1 : 2DLiDAR の反射強度の実験	28
5.3.1	実験目的	28
5.3.2	壁の反射強度	29
5.3.3	再帰反射テープの反射強度	30
5.3.4	学習する場所付近の反射強度	31
5.4	実験 2 : 提案手法による人追従の実験	32
5.4.1	実験目的	32
5.4.2	実験方法	32
5.4.3	結果と考察	34
第 6 章	結論	37
参考文献		38
付録		40
謝辞		41

図目次

1.1	Proposed method	2
1.2	The proposed method for learning of the person-following behavior	3
1.3	Training the neural network	5
1.4	The trained network is used to generate steering commands from a single front-facing center camera.	5
1.5	How the CNN “sees” an unpaved road. Top: subset of the camera image sent to the CNN. Bottom left: Activation of the first layer feature maps. Bottom right: Activation of the second layer feature maps. This demonstrates that the CNN learned to detect useful road features on its own, i.e., with only the human steering angle as training signal. We never explicitly trained it to detect the outlines of roads.	6
2.1	Neural network	8
2.2	Structure of end-to-end learning	9
2.3	Training the neural network	10
2.4	ImageNet classification with deep convolutional neural network	11
2.5	Hokuyo 2DLiDAR (UTM-30LX)	12
2.6	RViz (Display robot model and scan data)	13
3.1	Action according to angles of the joint	14
3.2	Target trajectory that the operator walks	15
3.3	Structure of the network	16

4.1	Output robot actions	17
4.2	The trained network is used to generate the robot's yaw angular velocity from the RGB images	18
4.3	Sequence of proposed method	18
4.4	Proposed method in the learning phase	19
4.5	Wearing retroreflective tape	19
4.6	Proposed method in the test phase	20
4.7	Without retroreflective tape	20
4.8	Image of scan	21
4.9	Turn left toward the retroreflective tape	23
4.10	Turn right toward the retroreflective tape	23
4.11	Architecture of the network	24
5.1	The environment of the experiment	26
5.2	Sequence of the experiment	27
5.3	The developed system	27
5.4	Measure the reflection intensity of the wall	29
5.5	Normal distribution of reflection intensity of wall	29
5.6	Measure the reflection intensity of retroreflective tape	30
5.7	Normal distribution of reflection intensity of retroreflective tape	30
5.8	Measure the reflection intensity of foyer	31
5.9	Histogram of reflection intensity of foyer	31
5.10	Learning and following phase courses	33
5.11	Learning phase in experiment	34
5.12	Following phase in experiment	35
5.13	Failed at the first corner	36
5.14	Histogram of angular velocity	36

表目次

3.1	The control rule and action parameters	15
3.2	Parameters of network configured with chainer	16
4.1	Parameters of <code>urg_node</code>	21
4.2	Output from rule-based controllers	22
4.3	Parameters of network configured with PyTorch	24
5.1	Laptop computer specifications	28
5.2	Experiment result	35

第1章

序論

1.1 背景

近年，機械学習を活用した自律移動に関する研究が盛んに行われている。本研究室でも，機械学習を用いて画像に基づく人追従行動の生成に関する研究を行ってきた。パシンら [1][2][3] は，引き紐を利用して画像に基づく人追従行動を生成する手法を提案している。この手法では，深層強化学習 [4] を活用しており，引き紐に取り付けられたポテンショメータでリンクの角度を取得し，それに応じた報酬をエージェント（ロボット）に与えて強化学習 [5] する。そして，画像と行動を深層学習 [6] することで，画像に基づいて人追従する行動を生成できることを示した。Fig. 1.1 にシステムの概要を示す。入力は画像で，出力は直進，左旋回，右旋回のいずれかの行動である。引き紐は，報酬を与えるためのみに利用され，ロボットの行動選択の入力としては用いていない。目的とする行動は人の追従であるため，選択された行動が適切であれば紐の向きとロボットの進行方向が一致するが，適切でない場合は一致しない。ここで，紐の向きと進行方向が一致しているときに高い報酬を与えている。

さまざまな行動と画像に対して，リンクの角度に応じた報酬を与えることで，徐々に人を追従する行動を選択していった。学習時間は約 20 分であるが，強化学習の特性により行動がランダムに選択されていた。この際に，ロボットは望まない行動を繰り返すため，追従対象者に比較的負担がかかるという問題があった。

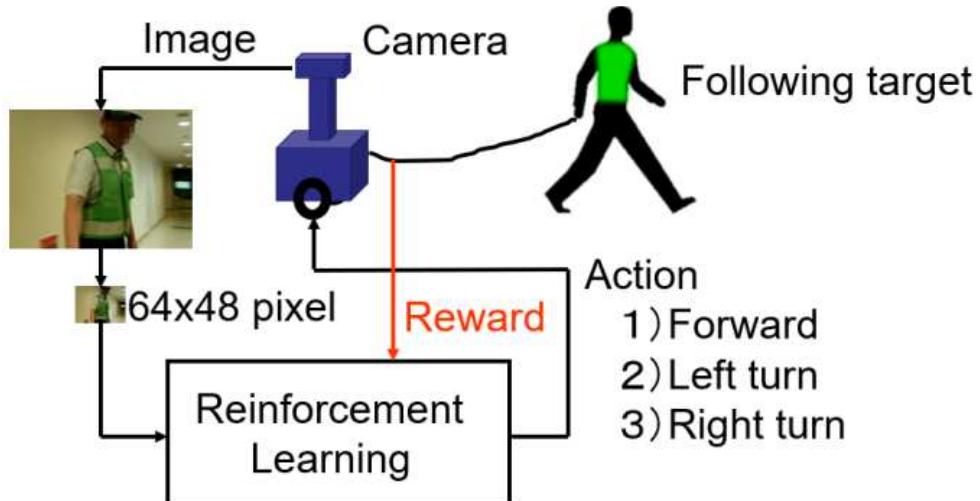
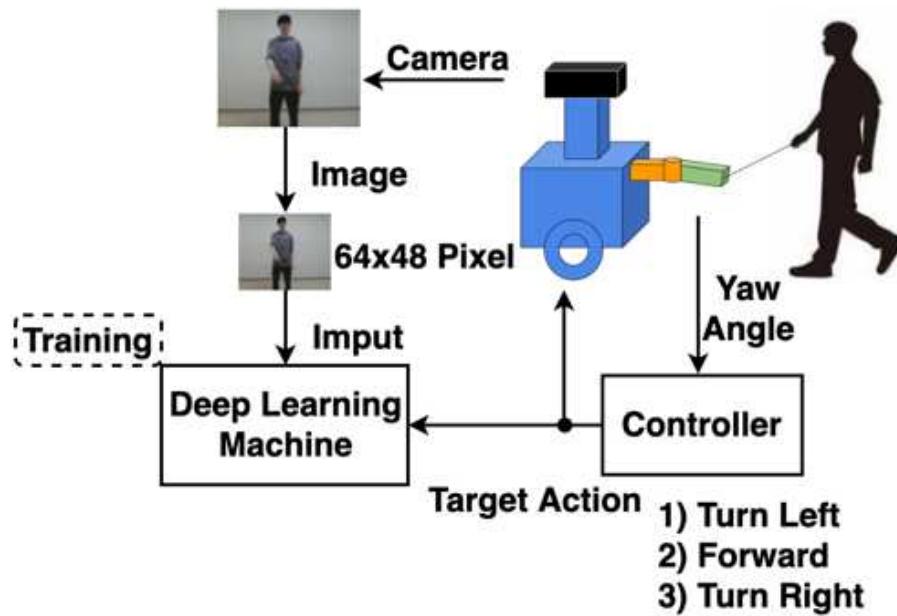


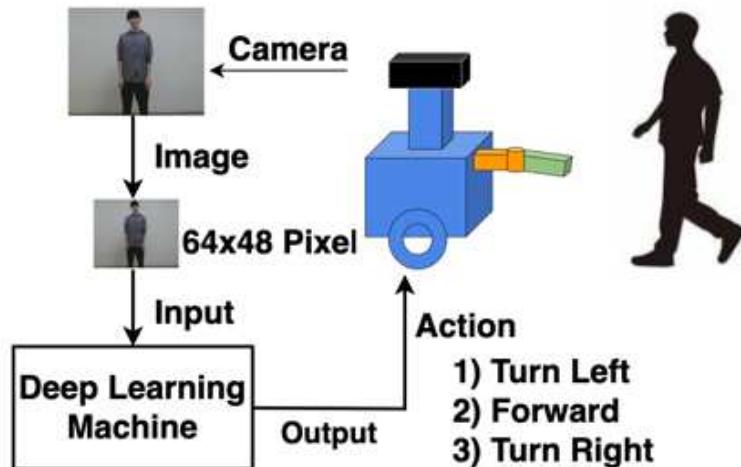
Fig. 1.1: Proposed method (source: [1])

岡田ら [7] は、強化学習のような教師なし学習ではなく、深層学習 [6] という教師あり学習を用いて画像に基づく人追従行動を生成する手法を提案している。この手法は、後述する Bojarski ら [8] の技術 (end-to-end 学習) を人追従問題に応用しており、強化学習を使用していないため、ロボットの行動がランダムに選択されることはない。また、学習時はルールベース制御器でロボットを制御しているので、常に人を追従する。つまり、学習時にも人追従行動を獲得でき、強化学習を採用する手法と比べて追従対象者の負担が少ないというメリットがある。Fig. 1.2 にシステムの概要を示す。まず、学習時には、追従対象者が引き紐を操作する。引き紐にはパシンらと同様にポテンショメータが取り付けられており、ヨー関節の変位角が 0 度となるようにロボットは直進や左旋回、右旋回のいずれかの行動で制御される。並行して、これらの行動とカメラ画像を深層学習器にオンラインで end-to-end 学習する。学習後は、追従対象者が引き紐を操作しなくても、深層学習器によりカメラ画像を入力するだけで、出力は直進や左旋回、右旋回といった行動を選択する。つまり、学習時のルールベース制御器（引き紐による人追従行動）を模倣するような深層学習器（カメラ画像による人追従行動）になっている。

これまで本研究室では、引き紐を用いて人追従行動を生成してきたが、測域センサの反射強度を利用することで同様の人追従行動を生成できる可能性がある。もし本研究の提案手法が有効であることが確認されれば、学習時のルールベース制御器に新たな選択肢が加わり、そのメリットが生まれる。



(a) Learning phase



(b) Following phase

Fig. 1.2: The proposed method for learning of the person-following behavior (source: [7])

1.2 目的

本研究では、2DLiDAR の反射強度を利用したルールベース制御器による人追従行動を生成する手法を提案し、その出力を教師信号として画像に基づいた人追従行動が可能かどうかを、実ロボットを用いた実験により提案手法の有効性を検証する。

1.3 関連研究

Bojarski ら [8] は、カメラ画像と人が操作するステアリングの角度を用いて模倣学習を行うことで、自動車の自動運転に成功している。学習時のシステムを Fig. 1.3 に示す。学習時には、ドライバーが車を運転し、その際に取得したステアリングの角度とカメラ画像を組み合わせて end-to-end 学習が行われる。これにより、学習後は、Fig. 1.4 に示すようにカメラ画像から直接、ステアリングの角度を出力するシステムになっている。すなわち、カメラ画像のみで自動運転を可能にしている。

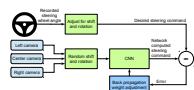


Fig. 1.3: Training the neural network (source: [8])



Fig. 1.4: The trained network is used to generate steering commands from a single front-facing center camera. (source: [8])

Fig. 1.5 は、CNN が白線等のない未塗装道路においても、人が操作するステアリングの角度だけを教師信号として用い、有用な道路の特徴を学習したことを示している。なお、道路の輪郭を検出するような学習は明示的に行っていないことが述べられている。



Fig. 1.5: How the CNN “sees” an unpaved road. Top: subset of the camera image sent to the CNN. Bottom left: Activation of the first layer feature maps. Bottom right: Activation of the second layer feature maps. This demonstrates that the CNN learned to detect useful road features on its own, i.e., with only the human steering angle as training signal. We never explicitly trained it to detect the outlines of roads. (source: [8])

1.4 論文の構成

第1章では、本研究の背景、目的、関連研究について述べた。第2章では、本研究で使用する要素技術について説明する。第3章では、岡田らによる従来手法について紹介する。第4章では、本研究の提案手法を述べる。第5章では、提案手法を用いた実験について述べる。第6章では、本研究の結論をまとめると。

第2章

要素技術

本章では、本研究で用いた深層学習に関する要素技術と ROS に関する要素技術について述べる。

2.1 深層学習

深層学習 (Deep learning) は、画像や音声などの複雑なデータを処理するための機械学習手法であり、人工ニューラルネットワークを基盤としている。Fig. 2.1 に示すように、多層のニューロンが組み合わさり、人間の脳のような階層的な構造をしている。これにより、例えば画像や音声の特徴を学習し、自動的に認識や分類を行うことが可能になる。

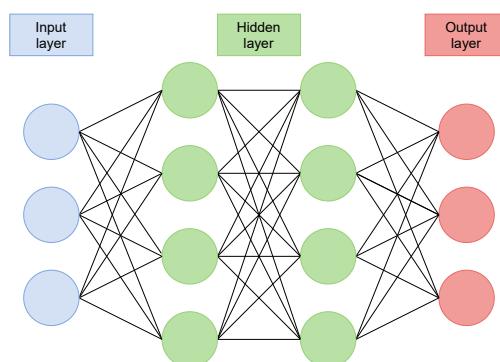


Fig. 2.1: Neural network

2.1.1 end-to-end 学習

end-to-end 学習は、Fig. 2.2 に示すように、システムの入力から出力までの全体の処理を一つのニューラルネットワークで直接学習する機械学習手法である。この手法では、画像や音声などの特徴抽出や前処理の段階を人手で設計する必要がなく、データから直接目標のタスクを学習することができる。

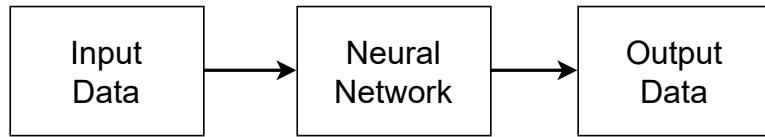


Fig. 2.2: Structure of end-to-end learning

2.1.2 ミニバッチ学習

本研究は、オンライン（データを収集しながら）でミニバッチ学習を行う。利点として、リアルタイムでのデータへの迅速な適応性が挙げられる。新しいデータが入手されるたびに、小さなバッチごとにモデルを即座に更新することで、変動する状況にリアルタイムで対応し、モデルの性能を最新かつ効果的に維持することが可能である。通常、データセット全体を一度に処理するバッチ学習と、一つずつのデータを処理するオンライン学習の中間に位置するアプローチである。本論文では、「オンラインで学習」と「オンライン学習」の使い分けに注意する。

2.1.3 Convolutional Neural Network (CNN)

本研究の学習器は畳み込みニューラルネットワーク (Convolutional Neural Network : CNN) で、これは画像認識などで用いられている [9][10]。畳み込み層とプーリング層を含む構造で、多次元配列の形式データを効率的に処理するように設計されている。例として、LeCun ら [9] は、畳み込み層とプーリング層を連続して接続するネットワークを用いることで、手書き文字を識別できることを示した (Fig. 2.3)。また、Krizhevsky ら [10] は、深い畳み込みニューラルネットワークを用いることで、1000 種類のクラスに分類できることを示し (Fig. 2.4)，ILSVRC(ImageNet Large Scale Visual Recognition Challenge)2012 で優勝した。

CNN は、主に畳み込み層、プーリング層、および全結合層から構成される。以下に、それぞれの特徴を記す。

1. 畳み込み層

入力データに対してフィルターを適用し、特徴を抽出した特徴マップを出力する層

2. プーリング層

特徴マップのサイズを削減し、特徴の位置に対する頑健性を向上させるために、領域内の最大値や平均値を取る操作を行う層

3. 全結合層

畳み込み層とプーリング層で抽出された特徴を組み合わせて、最終的な出力を生成する層

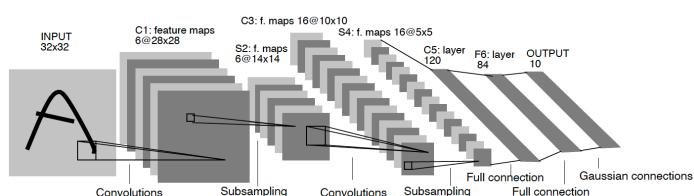


Fig. 2.3: Training the neural network (source: [9])



Fig. 2.4: ImageNet classification with deep convolutional neural network (source: [10])

2.2 ROS

ROS (Robot Operating System) [11] は、オープンソースのロボットソフトウェアフレームワークであり、ロボットアプリケーションの開発や実行をサポートするミドルウェアである。異なるバージョンが存在しているが、本研究では ROS Noetic を使用している。

2.2.1 LiDAR

LiDAR (Light Detection and Ranging) は、光を利用して距離を測定する技術であり、具体的には、レーザ光を発射し、対象物に当たって反射し戻ってくるまでの時間を計測することで距離を推定する。また、使用する装置によっては、物体によってどれだけの光が反射されたかを示す反射強度も測定することができる。したがって、LiDAR は周囲の状況を把握し、環境認識や物体検知などに利用される。

本研究で使用する 2DLiDAR を Fig. 2.5 に示す。これは、北陽電機社製の 2DLiDAR であり、ROS 上で `urg_node`[12] というパッケージが提供されている。この `urg_node` は、検出範囲（最大 270 [deg]）や反射強度の使用有無などのパラメータを変更するだけで簡単にデータのやり取りが可能である。



Fig. 2.5: Hokuyo 2DLiDAR (UTM-30LX) (source: [13])

2.2.2 RViz

RViz (ROS Visualization) [14] は、ROS で提供される三次元ビジュアライゼーションツールであり、数値で表されるロボットの座標や各センサのデータ直感的に理解できる三次元空間上に表示することができる。Fig. 2.6 にその様子を示す。ここでは、ロボットのモデルと 2DLiDAR からのセンサデータ、カメラ画像をリアルタイムに表示している。

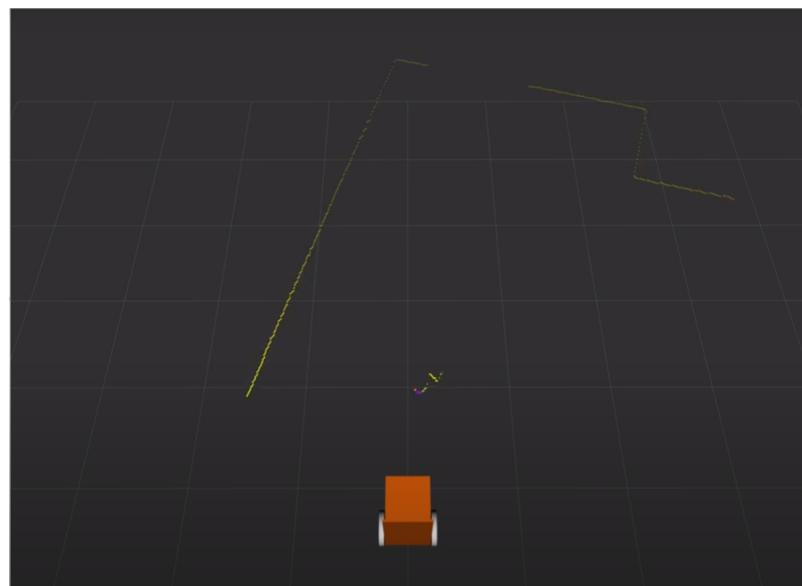


Fig. 2.6: RViz (Display robot model and scan data)

第3章

従来手法

本研究のベースとなる岡田らの研究（以後、従来手法と呼ぶ）について述べる。従来手法は、引き紐を用いたルールベース制御器による行動を模倣学習し、同様の行動を画像を用いて行う手法である。この手法は、学習フェーズと追従フェーズに分かれている。

3.1 学習フェーズ

学習フェーズで使用するルールベース制御器を Fig. 3.1 に示す。ルールベース制御器は、引き紐に取り付けられたポテンショメータからのリンクの角度を入力とし、ルールに従いロボットの行動を選択する。ルールベース制御器からの出力を Table 3.1 に示す。なお、並進速度は 0.1 [m/s] で一定である。

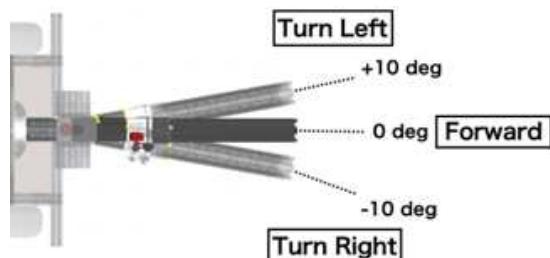


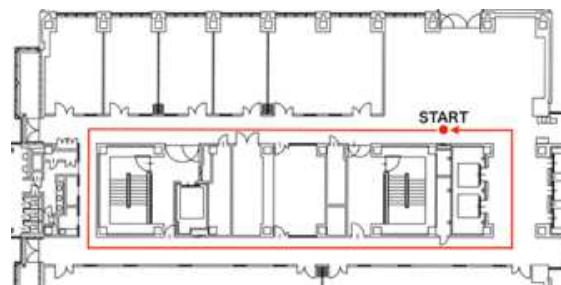
Fig. 3.1: Action according to angles of the joint (source: [7])

Table 3.1: The control rule and action parameters

Action	Control rule	Linear velocity	Angular velocity
Turn Left	$\theta_{\text{yaw}} > 10 \text{ deg}$	0.1 m/s	0.2 rad/s
Forward	$-10 \text{ deg} < \theta_{\text{yaw}} < 10 \text{ deg}$	0.1 m/s	0 rad/s
Turn Right	$\theta_{\text{yaw}} < -10 \text{ deg}$	0.1 m/s	-0.2 rad/s

3.2 追従フェーズ

追従フェーズでは、学習フェーズで学習したモデルを活用し、Fig. 3.2 に示す経路（学習フェーズと同じ経路）を使用してテストを行う。この際、引き紐は不要であり、代わりに深層学習器に画像が入力され、その出力がロボットの行動となる。つまり、学習フェーズで利用された引き紐によるルールベース制御器の出力ではなく、カメラ画像に基づく深層学習器の出力がロボットの行動になる。

**Fig. 3.2:** Target trajectory that the operator walks (source: [7])

3.3 ネットワークの構造

ネットワークを Fig. 3.3 に示す。これは、深層学習フレームワークである Chainer[15] を使用し、CNN をベースとしている。具体的には、入力層、畳み込み層 3、全結合層 2、出力層の 7 層で構成されている。深層学習器は、縮小された画像と選択された行動を 0.6 秒周期で収集して学習する。これに使用されたハイパーパラメータを Table 3.2 に示す。

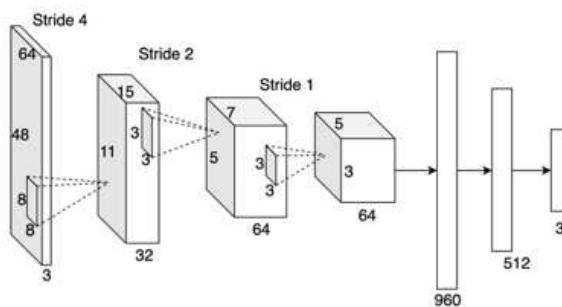


Fig. 3.3: Structure of the network (source: [7])

Table 3.2: Parameters of network configured with chainer

Input data	Image(64x48 pixels, RGB channels)
Optimizer	Adam($\alpha = 0.001, \beta_1 = 0.9, \beta_2 = 0.999, \text{pdf} = 1e^{-2}$)
Loss function	Softmax-cross-entropy
Output data	Action (Forward, Turn Left, Turn Right)

第4章

提案手法

本章では、従来手法をベースとする提案手法の概要、提案手法における学習フェーズ、追従フェーズ、ルールベース制御器で用いる装置、ルールベース制御器、ネットワーク構造についての6節に分けて述べる。

4.1 提案手法の概要

本研究は、ルールベース制御器に引き紐ではなく、2DLiDARの反射強度を利用する。このときのロボットの行動をFig. 4.1に示す。並進速度は、学習時と学習後で共に0.2 [m/s]で一定で、ロボットのヨー方向の角速度 ω のみ変化する。ルールベース制御器は、追従対象者に再帰反射テープを装着し、2DLiDARでそれを検出することで、人に追従する手法である。

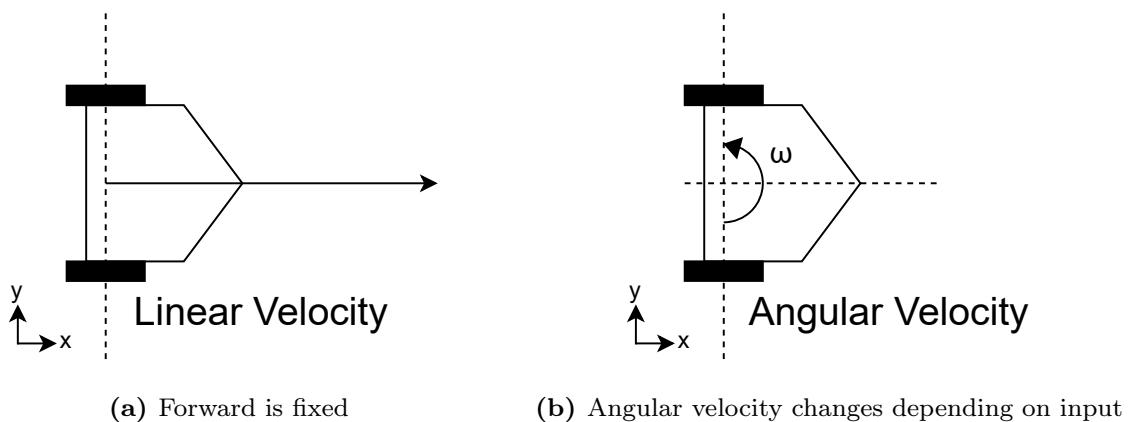


Fig. 4.1: Output robot actions

深層学習器は、ルールベース制御器の出力（ロボットのヨー方向の角速度 ω ）と RGB 画像を end-to-end 学習することで、Fig. 4.2 に示すように、入力を RGB 画像、出力をロボットのヨー方向の角速度 ω として人を追従する。

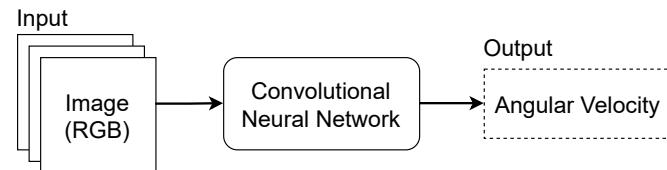


Fig. 4.2: The trained network is used to generate the robot's yaw angular velocity from the RGB images

Fig. 4.3 に示すように、ルールベース制御器を用いてロボットを制御するフェーズを学習フェーズ、深層学習器の出力をロボットの行動にするフェーズを追従フェーズと呼ぶこととする。以下に、学習フェーズと追従フェーズの主な役割を示す。

<学習フェーズ>

2DLiDAR の反射強度を利用したルールベース制御器に従い、ロボットを制御する。制御器の出力と画像を教師信号として深層学習器に与え、オンラインで end-to-end 学習する。

<追従フェーズ>

2DLiDAR は使用せず、画像を入力とした深層学習器の出力をロボットの行動にする。つまり、画像に基づいてロボットが人を追従する。

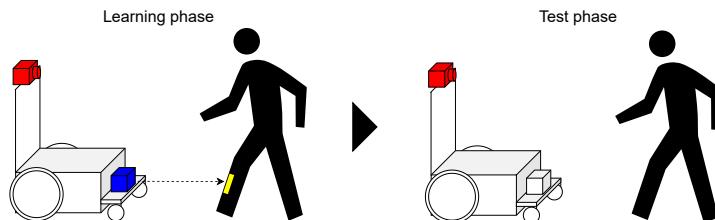


Fig. 4.3: Sequence of proposed method

4.2 学習フェーズ

学習フェーズの概要を Fig. 4.4 に示す。学習フェーズでは、2DLiDAR の反射強度を利用したルールベース制御器を用いて、Fig. 4.5 に示す追従対象者の足に装着した再帰反射テープに向かって、ロボットを制御する。ルールベース制御器の出力は、ロボットのヨー方向の角速度 ω の 1 つとし、角速度 ω が 0 [rad/s] となるようにロボットを制御することで人追従することが可能と考えられる。並行して、この行動とカメラの画像データを深層学習器に入力して、オンラインで学習させる。なお、並進速度は 0.2 [m/s] で一定にしているため、深層学習器に入力しない。

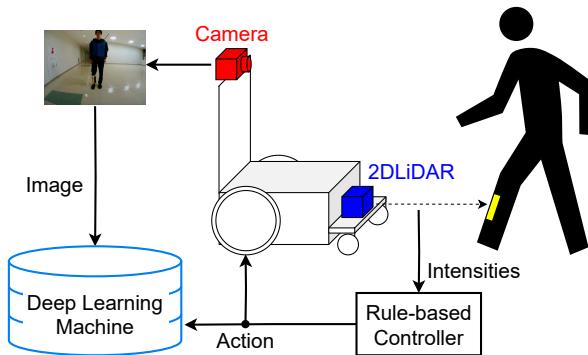


Fig. 4.4: Proposed method in the learning phase



Fig. 4.5: Wearing retroreflective tape

4.3 追従フェーズ

追従フェーズの概要を Fig. 4.6 に示す。追従フェーズでは、学習フェーズで獲得したモデルを用いる。ここでは、2DLiDAR を使用せず、代わりに深層学習器の出力がロボットの行動に影響を与える。つまり、2DLiDAR の反射強度に基づくルールベース制御器の出力ではなく、画像を入力とした深層学習器の出力がロボットの行動を決定する。

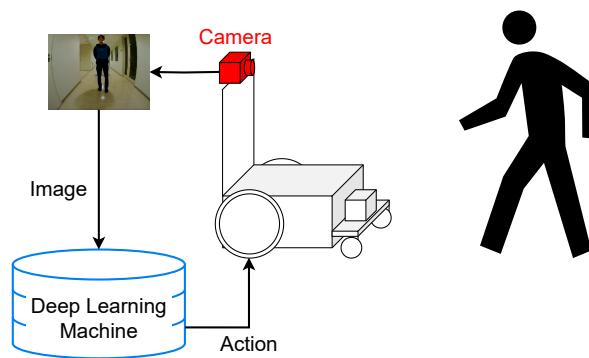


Fig. 4.6: Proposed method in the test phase



Fig. 4.7: Without retroreflective tape

4.4 ルールベース制御器で用いる装置

本研究で使用した 2DLiDAR は北陽電気社製の UTM-30LX[13] である。このセンサは、ROS 上で提供されている `urg_node`[12] というパッケージを使用することで、簡単にデータの取得とやり取りができる。この 2DLiDAR は、物体までの距離情報だけでなく、物体の反射強度の値も取得可能である。基本的には、`urg_node` で提供されているデフォルトのパラメータを使用するが、Table 4.1 に示すように一部のパラメータを変更している。このセンサ自体の最大検出範囲は 270 [deg] であるが、反射強度モードを使用する際には最大検出範囲を 120 [deg] に制限することが推奨されている[12]。そのため、センサの正面を 0 [deg] としたときに、左側に 60 [deg] (1.047 [rad])、右側に -60 [deg] (-1.047 [rad]) の最大 120 [deg] (2.094 [rad]) とした。また、1 回のスキャンは 25 [ms] 時間がかかり、-60 [deg] から 0 [deg] を通り 60 [deg] に向かってレーザが回転する。この動作を Fig. 4.8 に示す。

Table 4.1: Parameters of `urg_node`

Parameter name	Default value	Value to use
intensities	false	true
angle_min	-	-1.047
angle_max	-	1.047

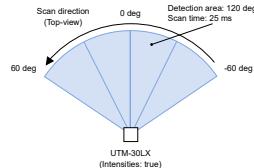


Fig. 4.8: Image of scan

4.5 ルールベース制御器

学習フェーズで使用するルールベース制御器は、前節で述べた 2DLiDAR の反射強度を利用しており、最大反射強度の方にロボットが追従する手法となっている。この制御器の出力は、ロボットのヨー方向の角速度 ω の 1 つであり、角速度 ω が 0 [rad/s] となるようにロボットを制御する。角速度 ω は、以下の式 (4.1) で表され、角度に応じた角速度 ω が式 (4.2) の範囲で出力される。

$$\omega[\text{rad/s}] = \frac{1}{\theta_{\max}} \times \theta \quad (4.1)$$

$$1 \geq \omega \geq -1 \quad (4.2)$$

ルールベース制御器からの出力を Table 4.2 に示す。ここでは、わかりやすくするために行動を左旋回、直進、右旋回の 3 つに分解しているが、実際に出力される行動はロボットのヨー方向の角速度 ω の 1 つであることに注意する。

Table 4.2: Output from rule-based controllers

Action	Control rule [rad]	Linear velocity [m/s]	Angular velocity [rad/s]
Turn Left	$\text{angle_max} \geq \theta > 0$	0.2	$1 \geq \omega > 0$
Forward	0	0.2	0
Turn Right	$0 < \theta \leq \text{angle_min}$	0.2	$0 > \omega \geq -1$

ルールベース制御器によるロボットの行動を Fig. 4.9 と Fig. 4.10 に示す。これは、RViz で部分的に可視化したもので、右足に装着した再帰反射テープに反応し、最大反射強度として周辺の色（黄色）とは異なる色（紫色）で表示している。

Fig. 4.9 の (a) は、2DLiDAR の左前方にいる人（再帰反射テープ）を検出し、その角度に応じた角速度 ω の値が output され、左旋回する。その結果、(b) のように角速度 ω はほぼ 0 [rad/s] に近づき、直進する。

Fig. 4.10 の (a) は、2DLiDAR の右前方にいる人（再帰反射テープ）を検出し、その角度に応じた角速度 ω の値が output され、右旋回する。その結果、(b) のように角速度 ω はほぼ 0 [rad/s] に近づき、直進する。

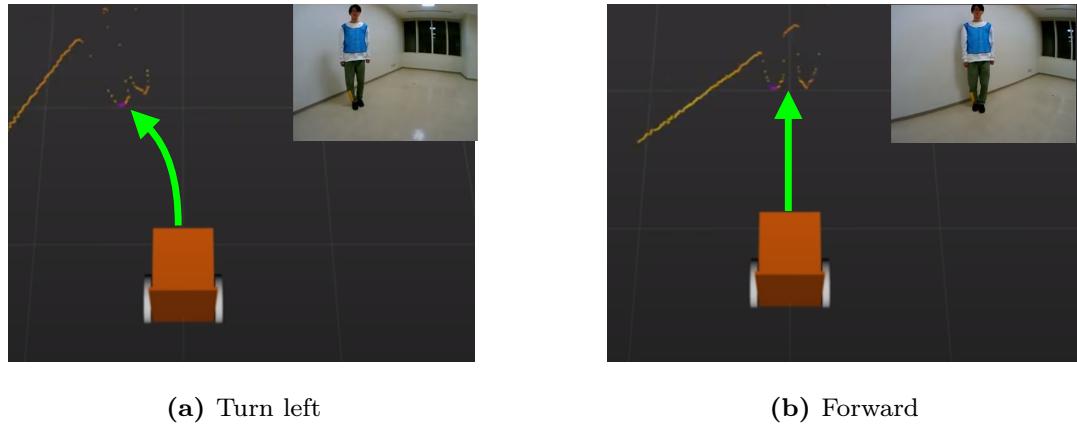


Fig. 4.9: Turn left toward the retroreflective tape



Fig. 4.10: Turn right toward the retroreflective tape

4.6 ネットワーク構造

ネットワークを Fig. 4.11 に示す。これは、深層学習フレームワークである PyTorch[16] を使用し、CNN をベースとしている。具体的には、入力層、畳み込み層 3、全結合層 2、出力層の 7 層で構成している。深層学習器は、縮小された画像と選択された行動を 0.2 秒周期で収集して学習する。これを 1step とする。使用したハイパーパラメータを Table 4.3 に示す。

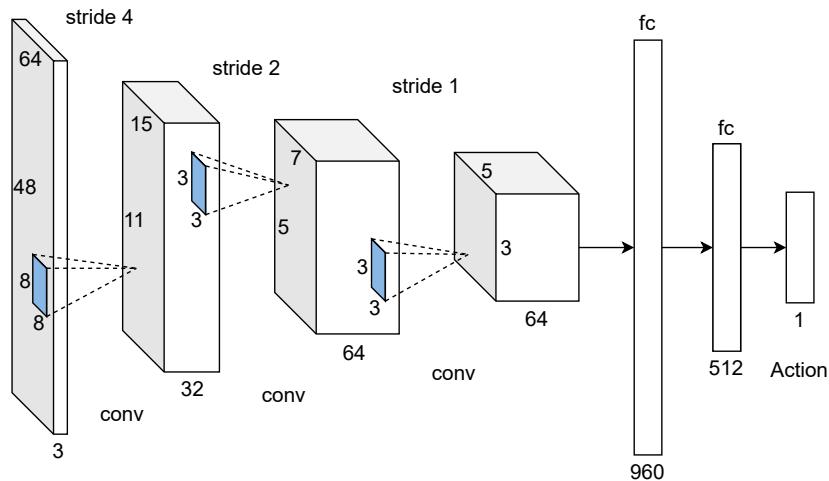


Fig. 4.11: Architecture of the network

Table 4.3: Parameters of network configured with PyTorch

Input data	Image(64x48 pixels, RGB channels)
Optimizer	Adam($\alpha = 0.001, \beta_1 = 0.9, \beta_2 = 0.999, \text{pdf} = 1e^{-2}$)
Loss function	Softmax-cross-entropy
Output data	Angular velocity

第5章

実験

本章では、実験の概要、実験装置、および反射強度の実験について述べた後、実ロボットを用いた実験により提案手法の有効性を検証する。

5.1 実験の概要

本研究では、2DLiDAR の反射強度を利用したルールベース制御器を用いた人追従行動を、カメラ画像で模倣学習することを課題としている。ルールベース制御器は、最大反射強度の方にロボットが追従する手法となっているため、実験で使用する再帰反射テープよりも高い反射強度を取得してしまうと人追従行動を継続できない恐れがある。そこで、2DLiDAR の反射強度の実験と提案手法による人追従の実験の 2 つに分けて実施する。それぞれの実験の概要を以下に示す。

<実験 1：2DLiDAR の反射強度の実験>

- 壁の反射強度を計測する実験
- 再帰反射テープの反射強度を計測する実験
- 学習フェーズで使用する周辺環境の反射強度を計測する実験

<実験2：提案手法による人追従の実験>

ルールベース制御器により10個の学習モデルを作成し、それぞれの学習モデルに対してテストを行う。つまり、実験を10回繰り返し、提案手法の有効性を検証する。

実験環境は、Fig. 5.1に示すように千葉工業大学津田沼キャンパス2号館3階の廊下を使用した。実験は天候による影響を少なくするために、夜間に実施した。また、服装による影響を少なくするために、追従対象者はFig. 5.2に示すような青いビブスを着用し、学習フェーズと追従フェーズに分けて実験を行った。

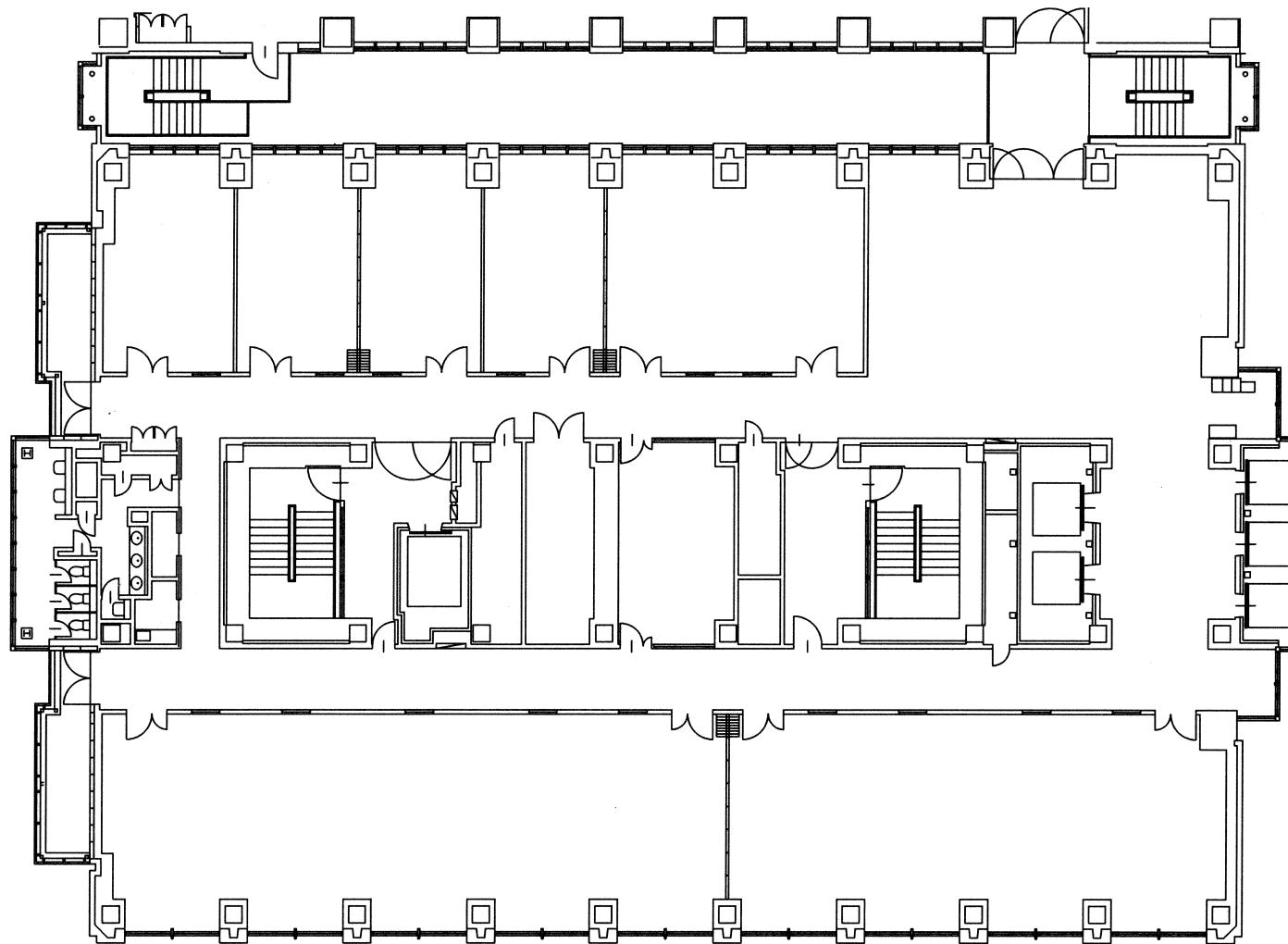


Fig. 5.1: The environment of the experiment

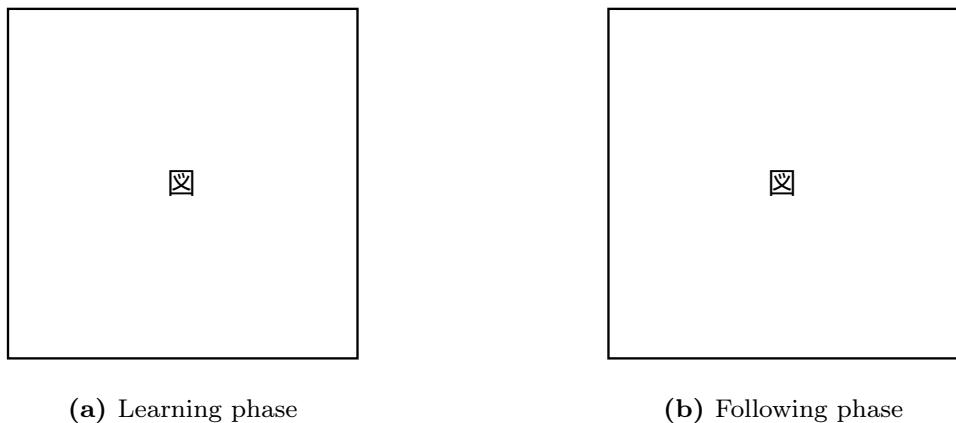


Fig. 5.2: Sequence of the experiment

5.2 実験装置

本研究で使用した実験装置を Fig. 5.3 に示す。ハードウェアは、T-frog プロジェクトの i-Cart mini[17] をベースとしたロボットである、ORNE-box[18][19] を使用する。このロボットは、拡張性に優れており、センサの取り付け位置を自由に変更できる。そのため、カメラを上部のハンドル部分に、2DLiDAR を下部の足元付近に設置した。PC の仕様を Table 5.1 に示す。また、ソフトウェアは ROS を使用して構築し、深層学習のフレームワークには PyTorch を採用している。

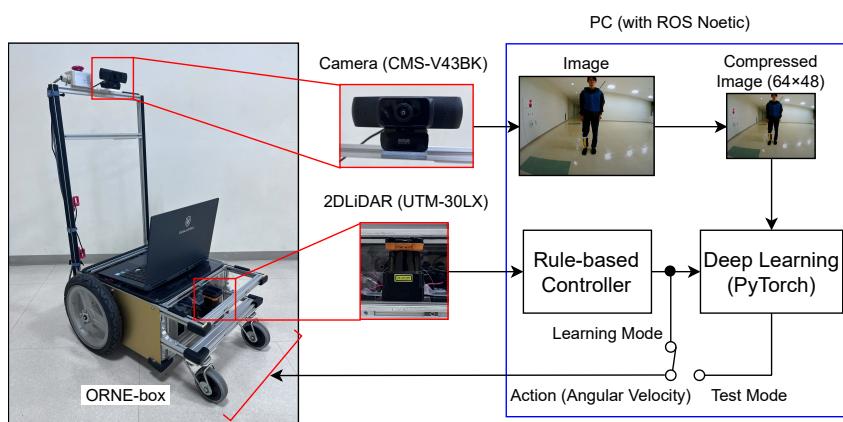


Fig. 5.3: The developed system

Table 5.1: Laptop computer specifications

Processor	Specification
OS	Ubuntu 20.04 LTS
ROS	Noetic
CPU	intel Core i7-10700F(4.8GHz/8コア/16スレッド)
GPU	RTX 2070 Max-Q
DRAM	32GB DDR4(3200/8GB × 4)

5.3 実験 1：2DLiDAR の反射強度の実験

実験 1 では、2DLiDAR の反射強度を利用した実験を行う。ここでは、以下に示す 3 つの事項に関して調査する。なお、以下の実験では、2DLiDAR の検出範囲を 0 [rad] で更に制限している。

1. 壁の反射強度を計測する実験
2. 再帰反射テープの反射強度を計測する実験
3. 学習フェーズで使用する周辺環境の反射強度を計測する実験

5.3.1 実験目的

本研究では、2DLiDAR の反射強度を利用したルールベース制御器を用いた人追従行動を、カメラ画像で模倣学習することを課題としている。ルールベース制御器は、最大反射強度の方にロボットが追従する手法となっているため、実験で使用する再帰反射テープよりも高い反射強度を取得してしまうと人追従行動を継続できない恐れがある。そのため、実験場所として指定した、千葉工業大学津田沼キャンパス 2 号館 3 階の廊下において、再帰反射テープよりも反射強度の高いものが存在しないかを、実験により調査する。

5.3.2 壁の反射強度

実験場所である、千葉工業大学津田沼キャンパス 2 号館 3 階の廊下の壁の反射強度を計測した。実験の様子を Fig. 5.4 に示す。2DLiDAR を壁に向けて正面に配置し、距離によって反射強度の値が変化するのを防ぐため、壁から約 500 [mm] 離れたところに固定した。

計測した結果を Fig. 5.5 に示す。収集したデータは 1 万個で、平均値は約 3600 であり、平均値付近にデータが集中していることがわかった。

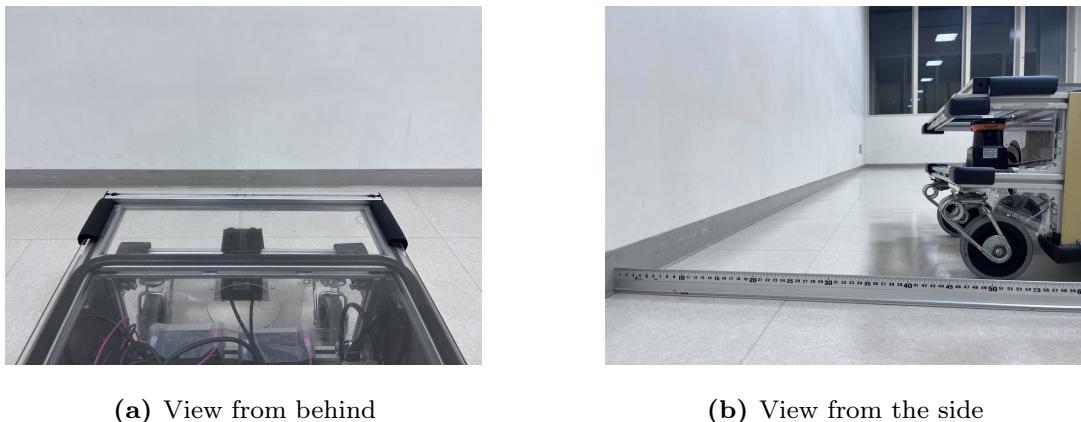


Fig. 5.4: Measure the reflection intensity of the wall

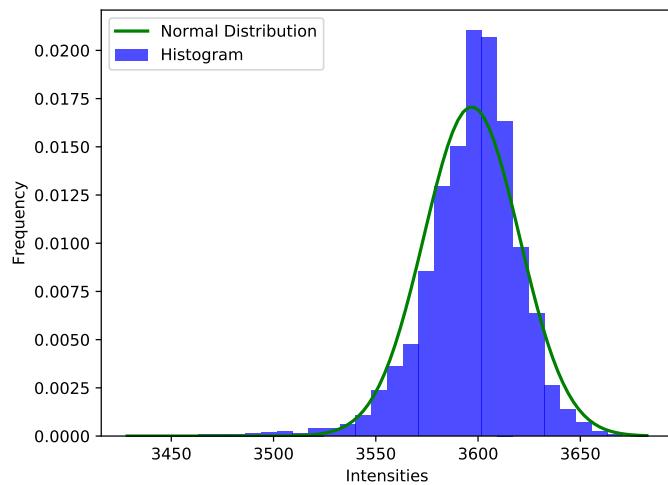
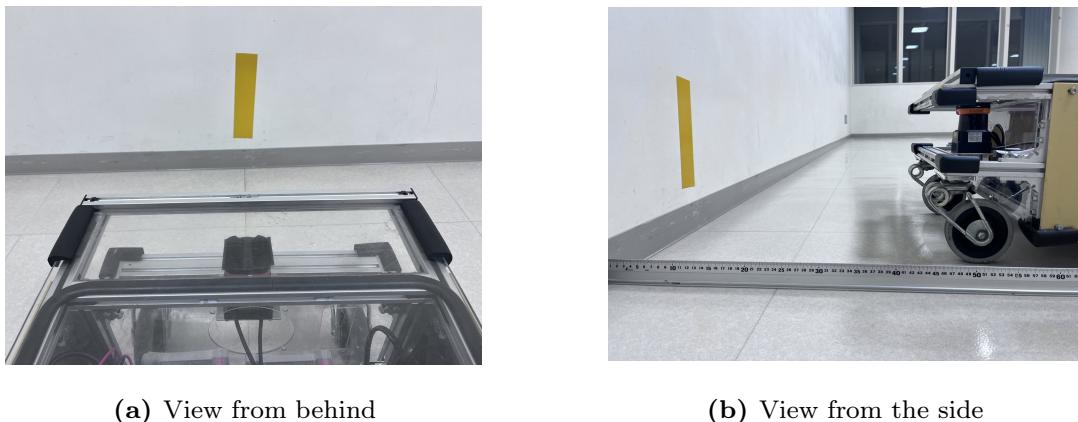


Fig. 5.5: Normal distribution of reflection intensity of wall

5.3.3 再帰反射テープの反射強度

再帰反射テープの反射強度を計測した。実験の様子を Fig. 5.6 に示す。2DLiDAR を再帰反射テープに向けて正面に配置し、距離によって反射強度の値が変化するのを防ぐため、再帰反射テープから約 500 [mm] 離れたところに固定した。

計測した結果を Fig. 5.7 に示す。収集したデータは 1 万個で、平均値は約 15300 であり、平均値付近にデータが集中していることがわかった。



(a) View from behind

(b) View from the side

Fig. 5.6: Measure the reflection intensity of retroreflective tape

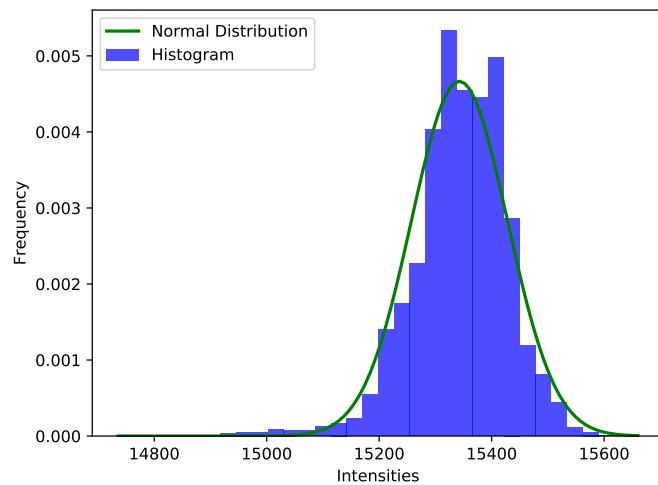


Fig. 5.7: Normal distribution of reflection intensity of retroreflective tape

5.3.4 学習する場所付近の反射強度

学習フェーズで使用する場所を Fig. 5.8 に示す。ここは、千葉工業大学津田沼キャンパス 2 号館 3 階のホワイエと呼ばれるスペースであり、中にはガラスや自動販売機が存在する。それに対して反射強度を計測するのは大変なので、ホワイエをランダムに歩き回ることでデータを収集する。

計測した結果を Fig. 5.9 に示す。収集したデータは 1 万個で、ホワイエには 5000 を超える反射強度の値は存在しないことがわかった。つまり、再帰反射テープが最大反射強度であり、ルールベース制御器が正常に機能すると、人追従行動生成が可能であることを確認した。

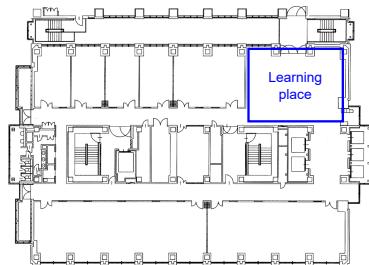


Fig. 5.8: Measure the reflection intensity of foyer

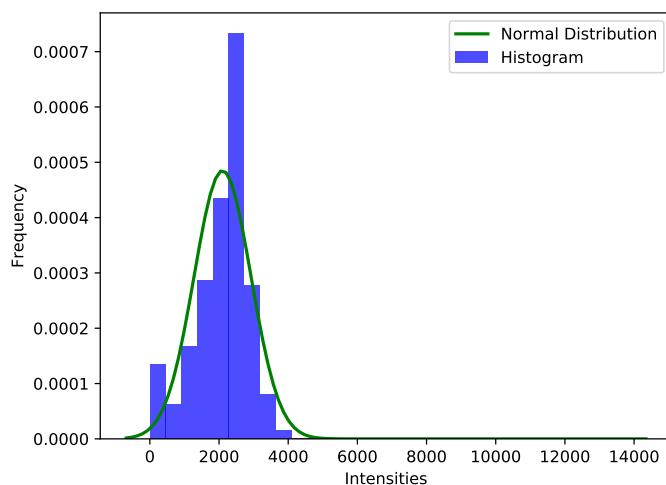


Fig. 5.9: Histogram of reflection intensity of foyer

5.4 実験2：提案手法による人追従の実験

実験2では、提案手法による人追従の実験を行う。

5.4.1 実験目的

提案手法により画像に基づいて人を追従する行動が生成されるかを、10回の実験により有効性を検証する。

5.4.2 実験方法

実験では、学習フェーズの後に追従フェーズに移行する。以下にそれぞれの役割を示す。

<学習フェーズ>

学習フェーズでは、追従対象者が再帰反射テープを装着し、Fig. 5.10における青色で示された場所（ホワイエ）を2DLiDARの最大検出範囲（120 [deg]）に注意しながら、10分間ランダムに歩き回る。

<追従フェーズ>

追従フェーズでは、反射強度を利用していいことをわかりやすくするため、追従対象者は再帰反射テープを取り外す。また、学習フェーズとは異なり、Fig. 5.10における赤色で示されたコースを1周する。このコースは1周約90mであり、学習フェーズで学習したモデルを用いて、2DLiDARは使用せずに画像のみで人追従を行う。その時のロボットの挙動を確認する。

学習フェーズでの10分間の学習と追従フェーズでのテストコースを1周することを1セットとし、10セット実験を行った。

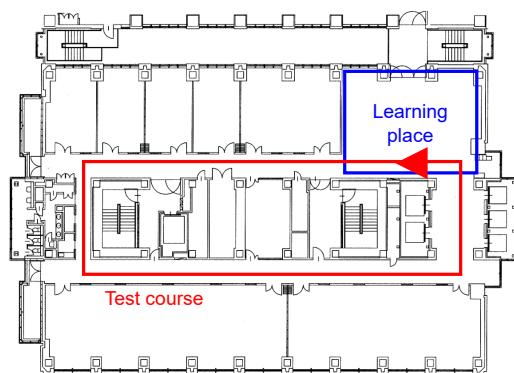


Fig. 5.10: Learning and following phase courses

5.4.3 結果と考察

すべての実験でロボットが人を追従する様子が確認できた。以下にそれぞれのフェーズの様子を記述する。

<学習フェーズ>

学習フェーズにおける実験の様子を Fig. 5.11 に示す。2DLiDAR の反射強度を利用したルールベース制御器を使用することで学習フェーズにおいてロボットが人を追従する様子が確認できた。10 セットの学習を行ったが、全てのセットにおいて人追従が途切れることはなかった。つまり、ルールベース制御器の有効性が確認できた。

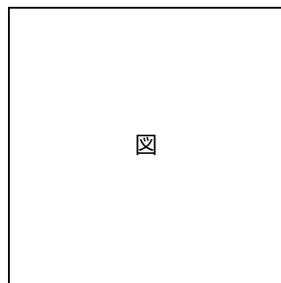


Fig. 5.11: Learning phase in experiment

<追従フェーズ>

追従フェーズにおける実験の様子を Fig. 5.12 に示す。テストコースの道中で、追従対象者のビブスと同じ青色の壁紙が貼られていたが、そちらに釣られることなく人追従行動を継続する様子が確認できた。

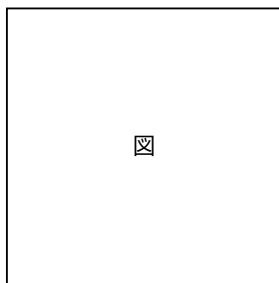


Fig. 5.12: Following phase in experiment

実験結果を Fig. 5.2 に示す。10 回の試行中、9 回は壁に衝突することなくコースを 1 周することができた。学習ステップ数は、0.2 秒周期で学習して 1step としているので、10 分の学習で 3000step となった。つまり、各学習モデルには 3000 個のデータが含まれ、人追従行動を明示的に学習せずに自動的に獲得している。一方で、10 回の試行中、1 回は Fig. 5.13 に示すように、テストコースの最初の曲がり角で壁に衝突した。失敗した場合の学習フェーズでの角速度のヒストグラムを Fig. 5.14 の (b) に示す。成功時の角速度のヒストグラムは (a) で、両者を比較すると、失敗時の左旋回の最大角速度が 0.1 [rad/s] 小さいことがわかる。このことから、学習フェーズで 0.3 [rad/s] 以上の角速度で左旋回する行動を学習していないのにもかかわらず、追従フェーズで 0.3 [rad/s] 以上の角速度で左旋回しなければならない位置に追従対象者が立っていたことが原因で、曲がりきれずに壁に衝突してしまったと考えられる。ただし、これについてはさらなる調査が必要である。

Table 5.2: Experiment result

Step	Result
3000	9/10 (90%)

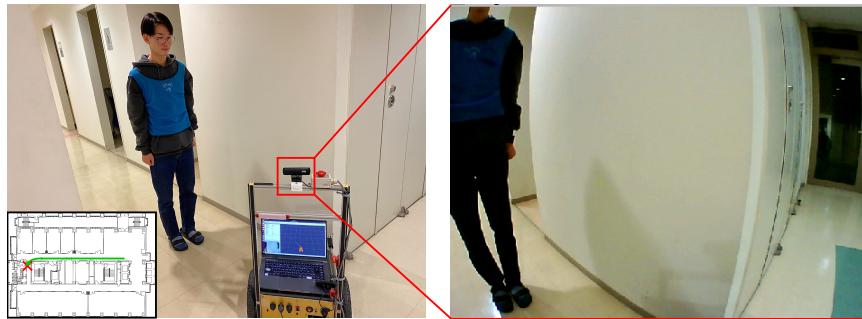


Fig. 5.13: Failed at the first corner

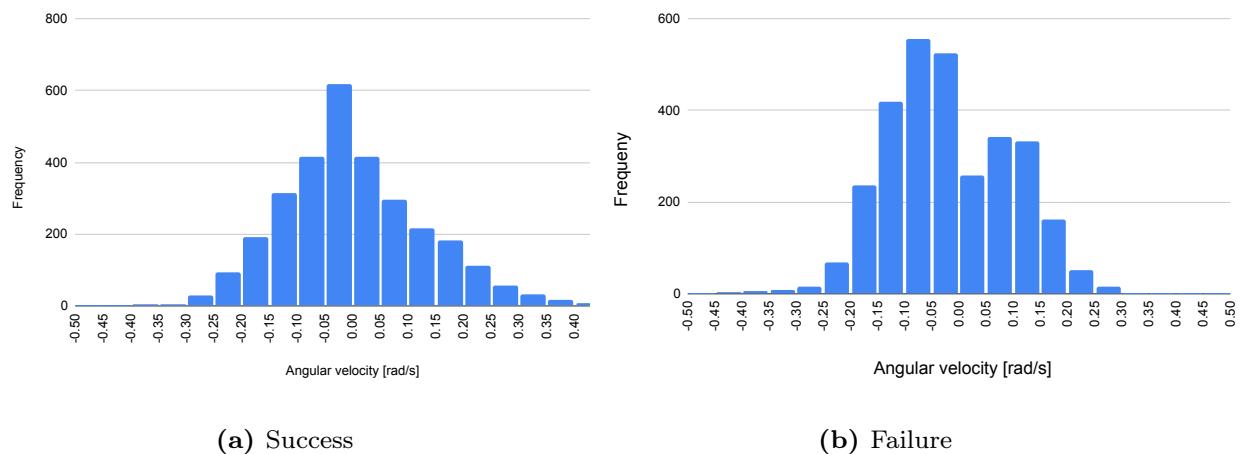


Fig. 5.14: Histogram of angular velocity

第6章

結論

本研究では、引き紐を利用したルールベース制御器による人追従行動をカメラ画像を用いて end-to-end 学習して模倣する岡田らの従来手法をベースに、2DLiDAR の反射強度を利用したルールベース制御器による人追従行動をカメラ画像を用いて end-to-end 学習して模倣する手法を提案した。また、実ロボットを用いた実験により、提案手法の有効性を検証した。その結果、2DLiDAR の反射強度を利用したルールベース制御器の出力により人追従行動が生成できること、カメラ画像を入力とした深層学習器の出力により人追従行動が生成できることを確認した。

参考文献

- [1] ティーラパップパシン, 林原 靖男, “強化学習を用いた移動ロボットの自律化に関する研究—引き紐の角度を報酬とする人追従の提案—”, 3E3-07, SI2017 (2017).
- [2] ティーラパップパシン, 上田隆一, 林原靖男, “強化学習を用いた移動ロボットの自律化に関する研究—引き紐を用いて一定間隔で人追従する学習の性能評価—”, 2B3-14, SI2018 (2018).
- [3] ティーラパップパシン, 林原靖男, 上田隆一, “強化学習を用いた移動ロボットの自律化に関する研究—一定の間隔で人を追従する行動の獲得に関する検討—”, 日本機械学会ロボティクス・メカトロニクス講演会'18 予稿集, 1A1-M11 (2018).
- [4] Hado van Hasselt, Arthur Guez, and David Silver, “Deep reinforcement learning with double q-learning”, Thirtieth AAAI conference on artificial intelligence, 2016.
- [5] Leslie Pack Kaelbling, Michael L. Littman, and Andrew W. Moore, “Reinforcement learning: A survey”, Journal of artificial intelligence research 4, pp.237-285, 1996.
- [6] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. “Deep learning”, nature 521.7553 (2015): 436-444.
- [7] 岡田眞也, 上田隆一, 林原靖男, “引き紐を利用した視覚と行動の End-to-end 学習による移動ロボットの人追従行動の生成”, 2A5-01, SI2019 (2019).
- [8] Mariusz Bojarski et al, “End-to-end Learning for Self-driving Cars”, arxiv: 1604.07316, 2016.
- [9] Yann Lecun et al, “Gradient-based learning applied to document recognition”, Proceedings of the IEEE 86.11 (1998): 2278-2324.
- [10] Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton, “Imagenet classification with deep convolutional neural networks”, Advances in neural information processing

- systems, 2012.
- [11] ROS. Documentation. <https://docs.ros.org/>. (Accessed on 12/31/2023).
 - [12] ROS Wiki. urg_node. https://wiki.ros.org/urg_node. (Accessed on 12/27/2023).
 - [13] 北陽電氣. UTM-30LX. <https://www.hokuyo-aut.co.jp/search/single.php?serial=21>. (Accessed on 12/27/2023).
 - [14] ROS-Visualization. ros-visualization/rviz GitHub repository. <https://github.com/ros-visualization/rviz>. (Accessed on 12/27/2023).
 - [15] Chainer. Documentation. <https://docs.chainer.org/en/latest/>. (Accessed on 12/29/2023).
 - [16] PyTorch. Documentation. <https://pytorch.org/docs/main/>. (Accessed on 12/29/2023).
 - [17] 渡辺敦司. T-Frog Project. http://t-frog.com/products/icart_mini/. (Accessed on 12/29/2023).
 - [18] 井口 鳩人、樋高 聖人、石江 義規、上田 隆一、林原靖男, "屋外自律移動ロボットプラットフォーム ORNE-box の開発", 3H2-03, SI2021(2021).
 - [19] 井口鷦人, 樋高聖人, 野村駿斗, 村林孝太郎, 上田隆一, 林原靖男, "屋外自律移動ロボットプラットフォーム ORNE-box の開発 – ORNE-box の検証・改良–", 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, 1P1-I06(2023).

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。