

# 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 following behavior  
using reflected intensity of a range sensor

2024 年 1 月 23 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，カメラ画像に基づいた人追従行動が可能か，実ロボットを用いた実験によりそ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 following behavior using reflected intensity of a range sensor

In recent years, research on autonomous mobility using machine learning has been actively conducted. In our laboratory, we have proposed two methods for human following robot. The first is a method that uses the angle of the link connected to the pull string as a reward and performs reinforcement learning to generate human following behavior based on images. The second method is to imitate a robot's human following behavior online by feeding the output of a rule-based controller with a pull string as an input to a deep learning machine as a teacher signal. Experiments using real robot have confirmed that these methods are capable of following people based on camera images.

Previous studies have generated human following behavior using a pull string, but it is possible to generate similar human following behavior by using the reflection intensity of a range sensor. This adds new options to the use of sensors during learning. In this research, we propose a method for end-to-end learning and imitation of human following behavior using camera images using a rule-based controller that uses the reflection intensity of a range sensor, and We will verify its effectiveness through experiments using real robot.

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

# 目次

|              |  |    |
|--------------|--|----|
| <b>第 1 章</b> | <b>序論</b>                                    | 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 . . . . .                              | 8  |
| <b>第 2 章</b> | <b>要素技術</b>                                  | 9  |
| 2.1          | 深層学習 . . . . .                               | 9  |
| 2.1.1        | end-to-end 学習 . . . . .                      | 10 |
| 2.1.2        | ミニバッチ学習 . . . . .                            | 10 |
| 2.1.3        | Convolutional Neural Network (CNN) . . . . . | 11 |
| 2.2          | ROS . . . . .                                | 13 |
| 2.2.1        | LiDAR . . . . .                              | 13 |
| 2.2.2        | RViz . . . . .                               | 14 |
| <b>第 3 章</b> | <b>従来手法</b>                                  | 15 |
| 3.1          | 学習フェーズ . . . . .                             | 16 |
| 3.2          | 追従フェーズ . . . . .                             | 17 |
| 3.3          | ネットワークの構造 . . . . .                          | 18 |
| <b>第 4 章</b> | <b>提案手法</b>                                  | 19 |
| 4.1          | 提案手法の概要 . . . . .                            | 19 |
| 4.2          | 学習フェーズ . . . . .                             | 21 |

|              |                         |           |
|--------------|-------------------------|-----------|
| 4.3          | テストフェーズ . . . . .       | 22        |
| 4.4          | 反射強度を計測する装置 . . . . .   | 23        |
| 4.5          | ルールベース制御器 . . . . .     | 24        |
| 4.6          | ネットワーク構造 . . . . .      | 25        |
| <b>第 5 章</b> | <b>2DLiDAR の反射強度の実験</b> | <b>26</b> |
| 5.1          | 実験の概要 . . . . .         | 26        |
| 5.2          | 実験の目的 . . . . .         | 27        |
| 5.3          | 壁の反射強度 . . . . .        | 28        |
| 5.4          | 再帰反射テープの反射強度 . . . . .  | 29        |
| 5.5          | 学習する場所付近の反射強度 . . . . . | 30        |
| 5.6          | 計測した反射強度の比較 . . . . .   | 31        |
| <b>第 6 章</b> | <b>実験</b>               | <b>32</b> |
| 6.1          | 実験の概要 . . . . .         | 32        |
| 6.2          | 実験装置 . . . . .          | 33        |
| 6.3          | 実験方法 . . . . .          | 34        |
| 6.4          | 結果と考察 . . . . .         | 35        |
| <b>第 7 章</b> | <b>結論</b>               | <b>38</b> |
| <b>参考文献</b>  |                         | <b>39</b> |
| <b>付録</b>    |                         | <b>41</b> |
| <b>謝辞</b>    |                         | <b>42</b> |

# 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  |
| 1.6 | Example of point cloud plot for tracking target . . . . .   | 7  |
| 2.1 | Neural network . . . . .  | 9  |
| 2.2 | Structure of general machine learning . . . . .   | 10 |
| 2.3 | Structure of end-to-end learning . . . . .  | 10 |
| 2.4 | Training the neural network . . . . .   | 11 |
| 2.5 | ImageNet classification with deep convolutional neural network . . . . .  | 12 |
| 2.6 | Hokuyo 2DLiDAR (UTM-30LX) . . . . .   | 13 |
| 2.7 | RViz (Display robot model, scan data, and camera images) . . . . .  | 14 |
| 3.1 | Action according to angles of the joint . . . . .   | 16 |

|      |   |    |
|------|---|----|
| 3.2  | Target trajectory that the operator walks . . . . .   | 17 |
| 3.3  | Structure of the network . . . . .  | 18 |
| 4.1  | Outputted robot actions . . . . .   | 19 |
| 4.2  | The trained network is used to generate the robot's yaw angular velocity<br>from the RGB images . . . . . | 20 |
| 4.3  | Sequence of proposed method . . . . .   | 20 |
| 4.4  | System in the learning phase . . . . .  | 21 |
| 4.5  | Attachment retroreflective tape . . . . .   | 21 |
| 4.6  | System method in the test phase . . . . .   | 22 |
| 4.7  | Without retroreflective tape . . . . .  | 22 |
| 4.8  | Condition of scan . . . . .   | 23 |
| 4.9  | Action from rule-based controller . . . . .   | 24 |
| 4.10 | Architecture of the network . . . . .   | 25 |
| 5.1  | The environment of the experiment . . . . .   | 27 |
| 5.2  | Measure the reflection intensity of the wall . . . . .  | 28 |
| 5.3  | Histogram of reflection intensity of wall . . . . .   | 28 |
| 5.4  | Measure the reflection intensity of retroreflective tape . . . . .  | 29 |
| 5.5  | Histogram of reflection intensity of retroreflective tape . . . . .                                       | 29 |
| 5.6  | Measure the reflection intensity of foyer . . . . .   | 30 |
| 5.7  | Histogram of reflection intensity of foyer . . . . .  | 30 |
| 5.8  | Comparison of reflection intensity by histogram . . . . .   | 31 |
| 6.1  | Sequence of the experiment . . . . .  | 32 |
| 6.2  | The developed system . . . . .  | 33 |
| 6.3  | Learning and test phase courses . . . . .   | 34 |
| 6.4  | Learning phase in experiment . . . . .  | 35 |
| 6.5  | Test phase in experiment . . . . .  | 36 |
| 6.6  | Failed at the first corner . . . . .  | 37 |

---

|     |   |    |
|-----|---|----|
| 6.7 | Histogram of angular velocity during learning phase . . . . . | 37 |
|-----|---|----|

# 表目次

|     |   |    |
|-----|---|----|
| 3.1 | The control rule and action parameters . . . . .        | 16 |
| 3.2 | Parameters of network configured with chainer . . . . . | 18 |
| 4.1 | Parameters of <code>urg_node</code> . . . . .           | 23 |
| 4.2 | Output from rule-based controller . . . . .             | 24 |
| 4.3 | Parameters of network configured with PyTorch . . . . . | 25 |
| 5.1 | Comparison of reflection intensity . . . . .            | 31 |
| 6.1 | Laptop computer specifications . . . . .                | 33 |
| 6.2 | Experiment result . . . . .                             | 37 |

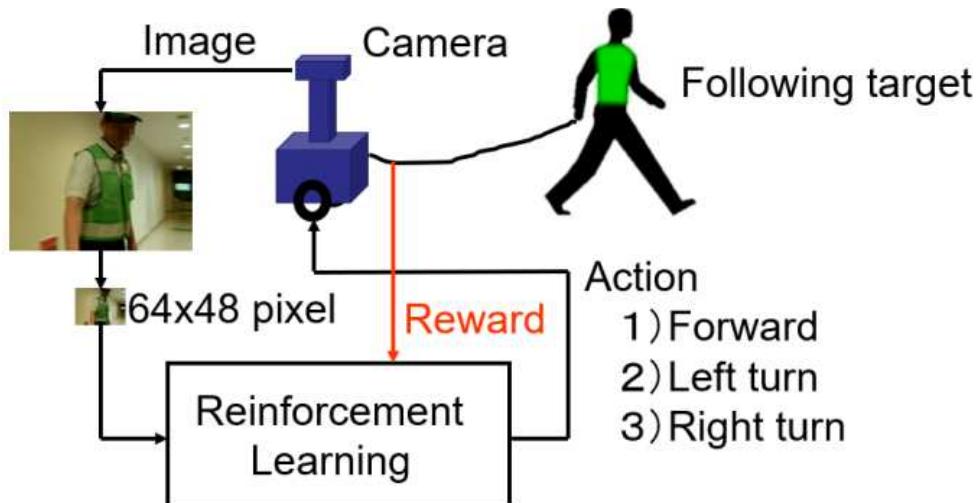
# 第1章

## 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。

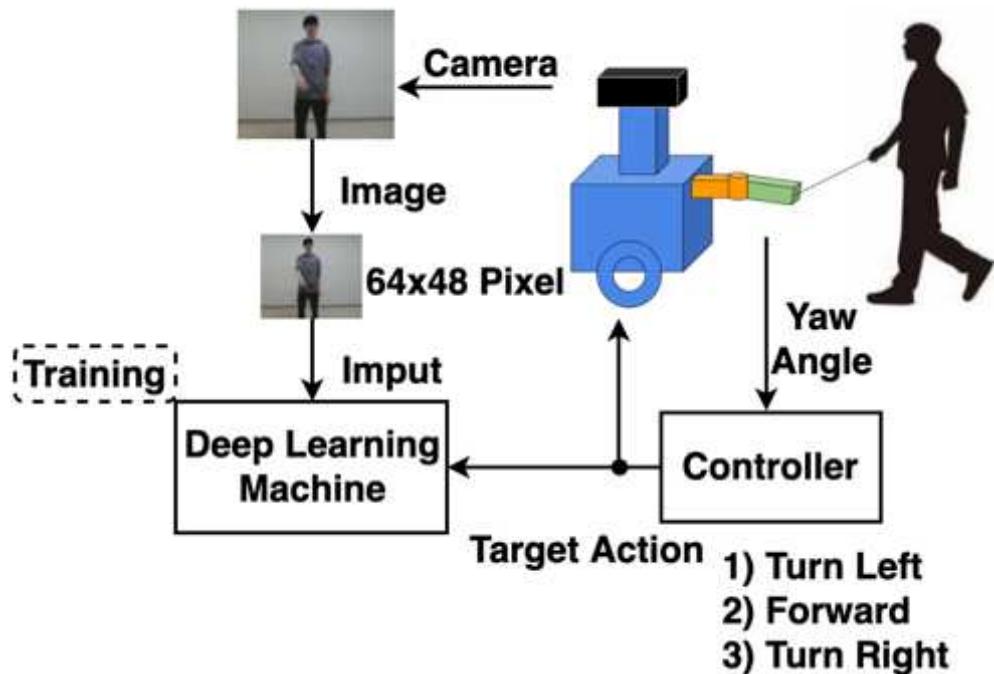
さまざまな行動と画像に対して，リンクの角度に応じた報酬を与えることで，徐々に人を追従する行動を選択していった。学習時間は約 20 分であるが，強化学習の仕組みによりある一定の割合で行動がランダムに選択されていた。この際に，ロボットは望まない行動を繰り返すため，追従対象者に比較的負担がかかるという問題があった。



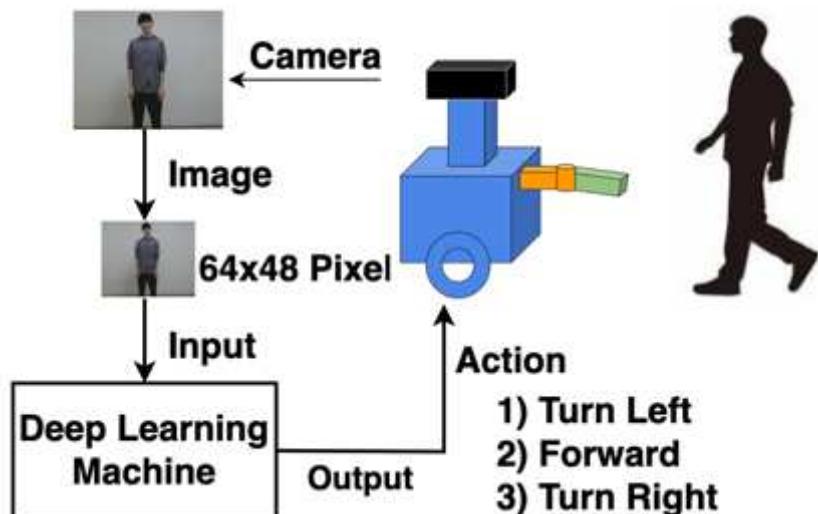
**Fig. 1.1:** Proposed method (source: [1])

岡田ら [7] は、深層学習により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、強化学習を使用していないため、学習時の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。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。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。



(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、カメラ画像に基づいた人追従行動が可能か、実ロボットを用いた実験によりそ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。

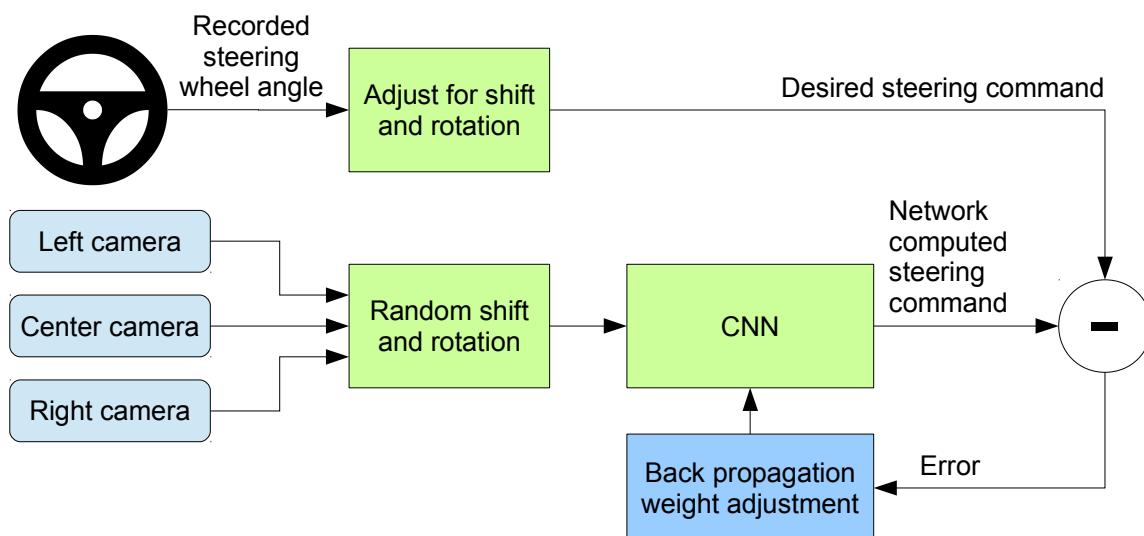


Fig. 1.3: Training the neural network (source: [8])

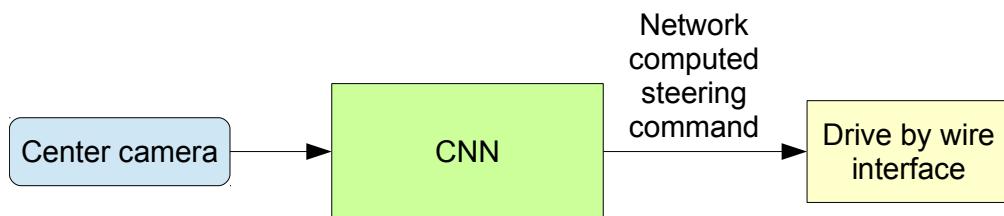
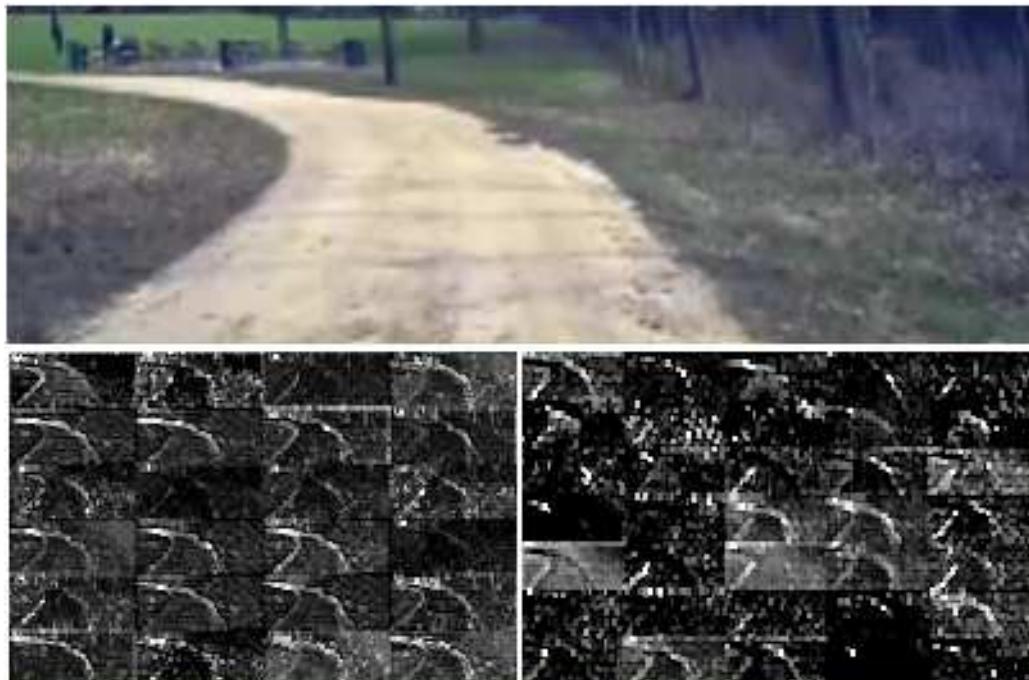


Fig. 1.4: The trained network is used to generate steering commands from a single front-facing center camera. (source: [8])

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])

大島ら [9] は、協働運搬ロボット「サウザー」に取り付けられた一つの 2DLiDAR の入力によって、自動追従走行機能を実現している。ロボットの操作者はロボットの正面に立ち、追従スタートボタンを押すと追従ターゲットとして認識され、歩き出すとロボットはターゲットに追従して走行する。これは、ボタンが押された瞬間の正面一定エリアにある点群をクラスタリングし、正面に存在するクラスタを追従対象としてラベリングしている (Fig. 1.6)。つまり、2DLiDAR の入力のみで人追従行動を実現している。



Fig. 1.6: Example of point cloud plot for tracking target (source: [9])

## 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、2DLiDARの反射強度の実験について述べる。第6章では、提案手法による実ロボットを用いた実験について述べる。第7章では、本研究の結論をまとめると。

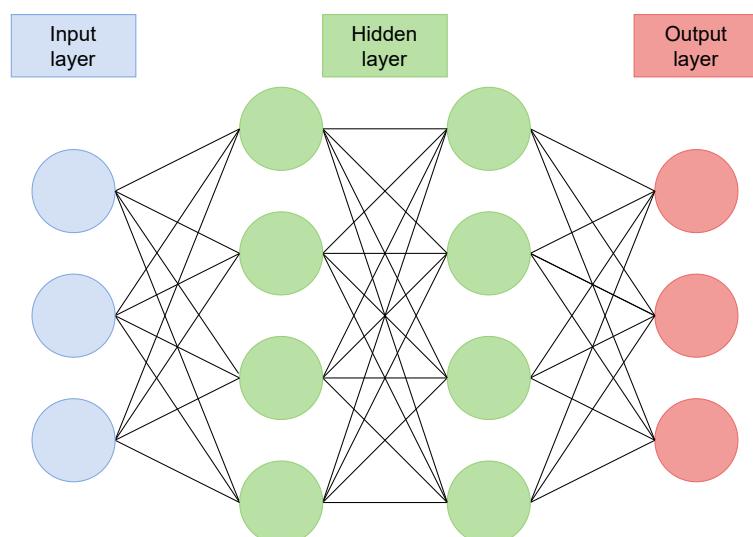
## 第2章

# 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

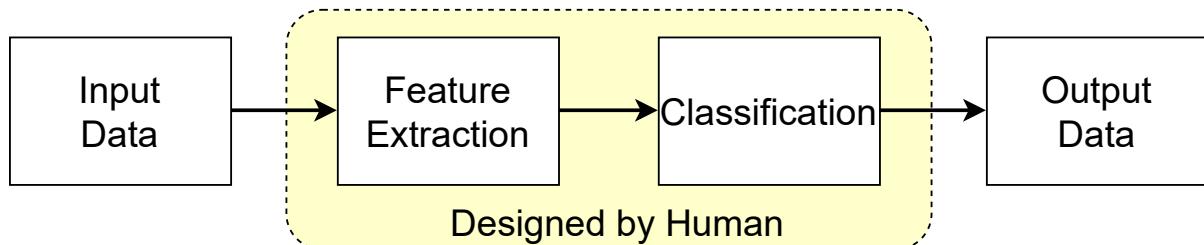
深層学習 (Deep learning) は、画像や音声などの複雑なデータを処理するための機械学習手法であり、人工ニューラルネットワークを基盤としている。Fig. 2.1 に示すように、多層のニューロンが組み合わさり、人間の脳のような階層的な構造をしている。これにより、例えば画像や音声の特徴を学習し、自動的に認識や分類を行うことが可能になる。



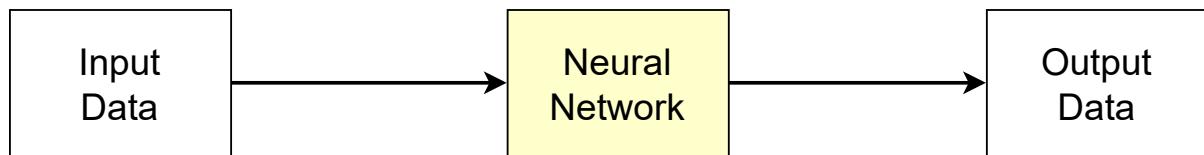
**Fig. 2.1:** Neural network

### 2.1.1 end-to-end 学習

end-to-end 学習は、Fig. 2.2 に示すような一般的な機械学習の構造とは異なり、Fig. 2.3 に示すような、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 general machine learning



**Fig. 2.3:** Structure of end-to-end learning

### 2.1.2 ミニバッチ学習

ミニバッチ学習は、データをミニバッチという小さなグループに分割してモデルを学習するアプローチである。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。

本研究は、オンライン（タスクを行いながらデータを収集）で学習（end-to-end 学習とミニバッチ学習を組み合わせた学習）を行う。

### 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 [10][11]。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 [10] 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. 2.4)。また、Krizhevsky ら [11] 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. 2.5)，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

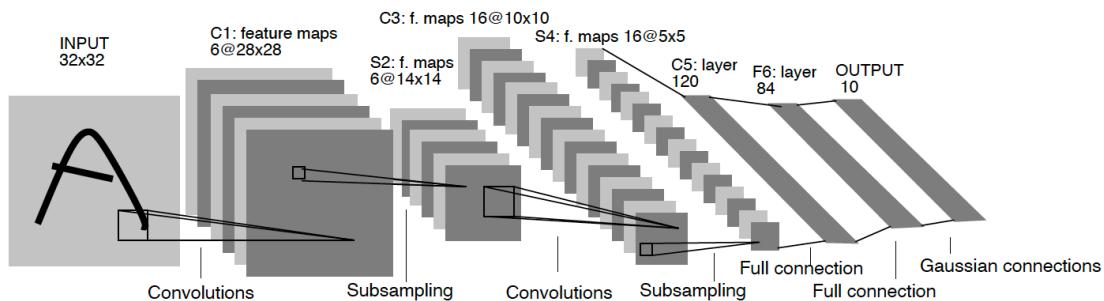
入力データに対してフィルターを適用し、特徴を抽出した特徴マップを出力する層

#### 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. 2.4:** Training the neural network (source: [10])

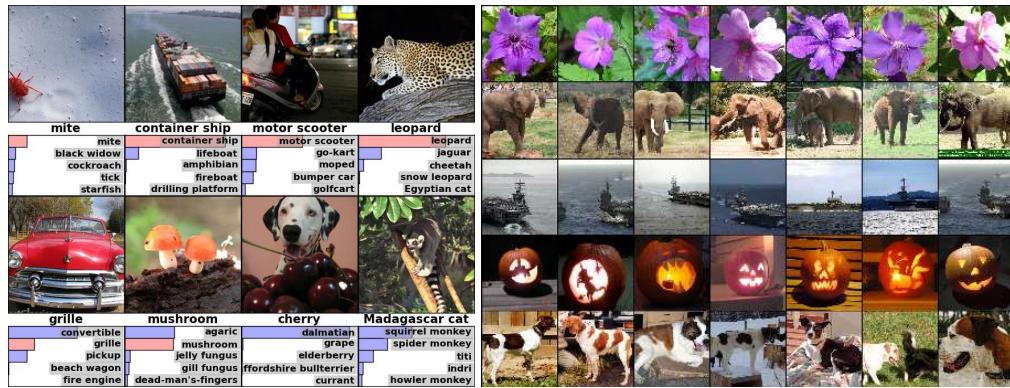


Fig. 2.5: ImageNet classification with deep convolutional neural network (source: [11])

## 2.2 ROS

ROS (Robot Operating System) [12] 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. 2.6 に示す。これは、北陽電機社製の 2DLiDAR であり、ROS 上で `urg_node`[13] 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データの取得が可能である。



Fig. 2.6: Hokuyo 2DLiDAR (UTM-30LX)

## 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を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. 2.7 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。

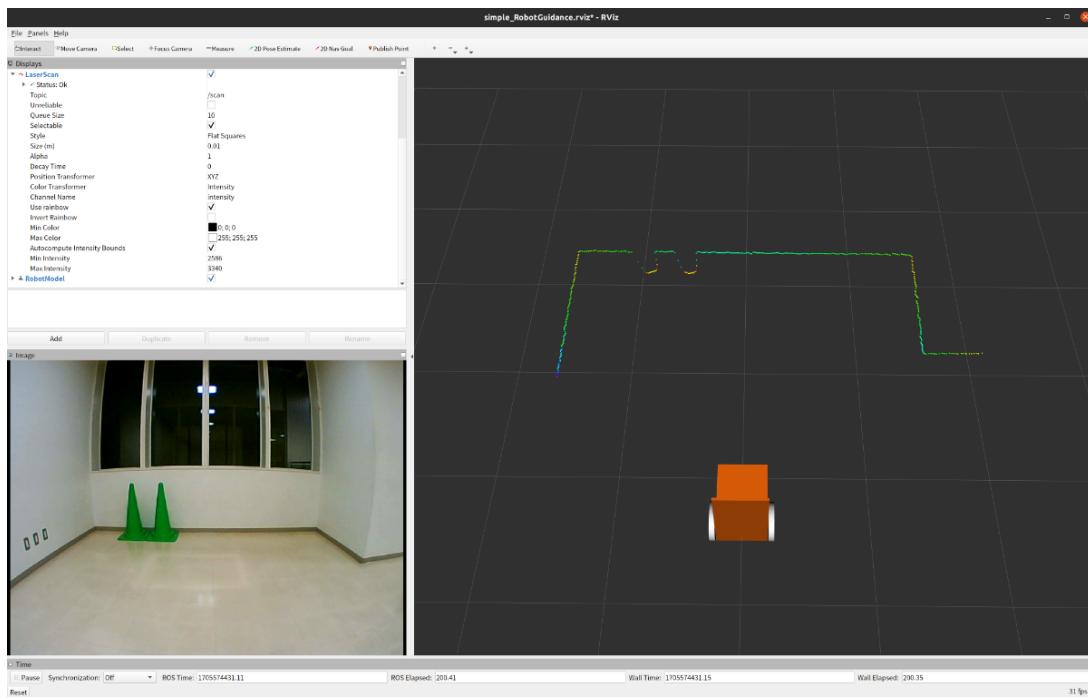


Fig. 2.7: RViz (Display robot model, scan data, and camera images)

## 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、学習時に引き紐を入力としたルールベース制御器の出力で直進や左旋回、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。これにより、学習後はカメラ画像を入力とした深層学習器の出力で同様の行動（直進、左旋回、右旋回）を行う手法である。従来手法では、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、深層学習器の出力でロボットを制御するフェーズを追従フェーズとし、2つのフェーズに分けている。

#### <学習フェーズ>

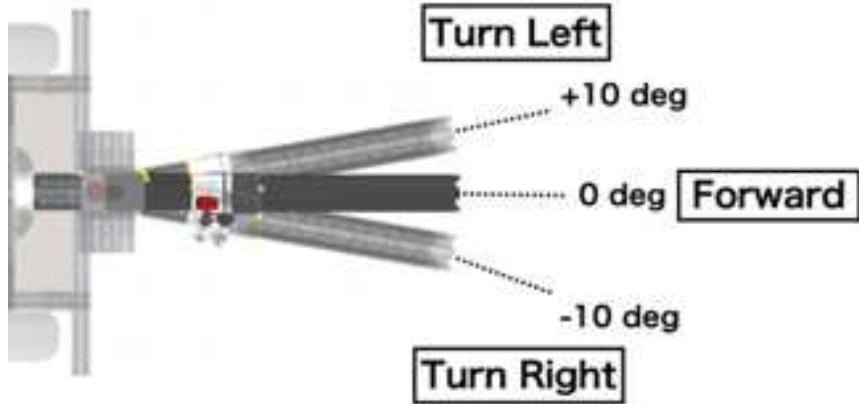
引き紐に接続されたリンクの角度を入力としたルールベースの制御器に従い、ロボットを制御する。その制御器の出力と画像を教師信号として深層学習器に与え、end-to-end 学習する。

#### <追従フェーズ>

引き紐は使用せず、画像を入力とした学習器の出力でロボットを制御する。つまり、画像に基づいてロボットが人を追従する。

### 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は行動を選択する。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。



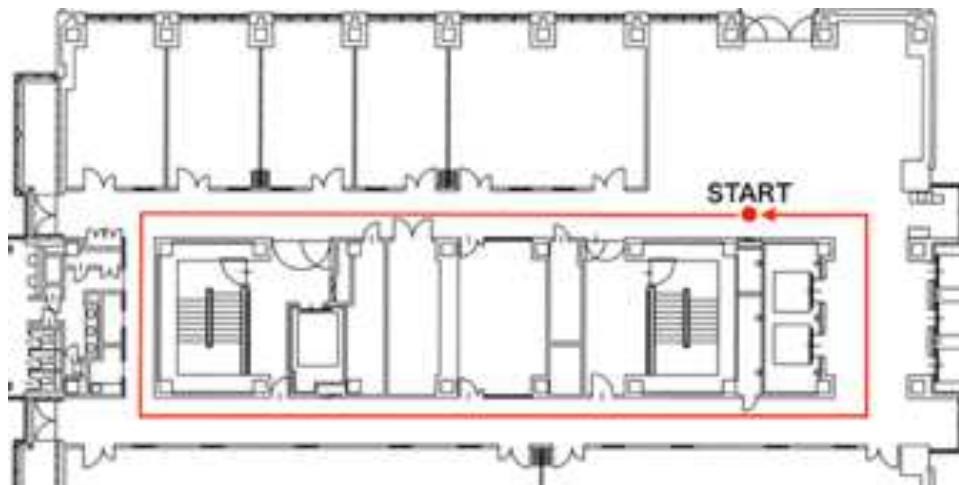
**Fig. 3.1:** Action according to angles of the joint (source: [7])

**Table 3.1:** The control rule and action parameters

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

### 3.2 追従フェーズ

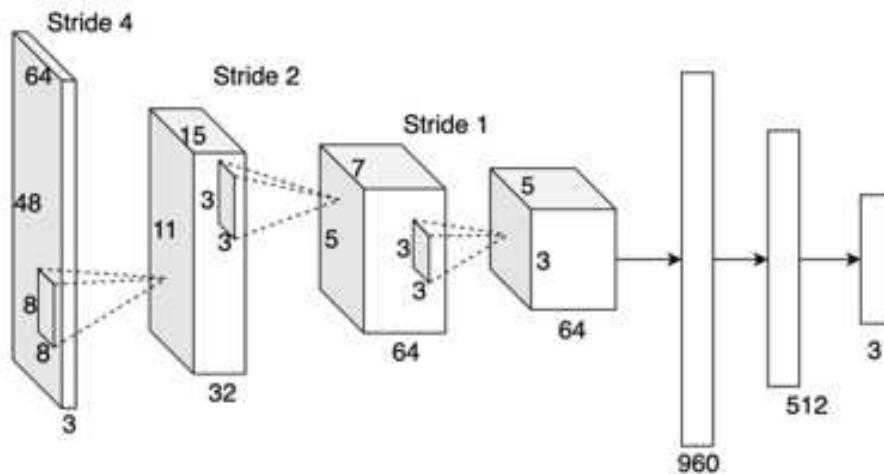
追従フェーズでは、学習フェーズで学習したモデルを活用し、Fig. 3.2 に示す経路（学習フェーズと同じ経路）でテストを行う。この際、引き紐は不要であり、代わりに深層学習器に画像が入力され、その出力がロボットの行動となる。つまり、学習フェーズで利用された引き紐によるルールベース制御器の出力ではなく、カメラ画像に基づく深層学習器の出力がロボットの行動になる。



**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( $\alpha = 0.001, \beta_1 = 0.9, \beta_2 = 0.999, \epsilon = 1e^{-2}$ ) |
| Loss function | Softmax-cross-entropy  |
| Output data   | Action (Forward, Turn Left, Turn Right)                                      |

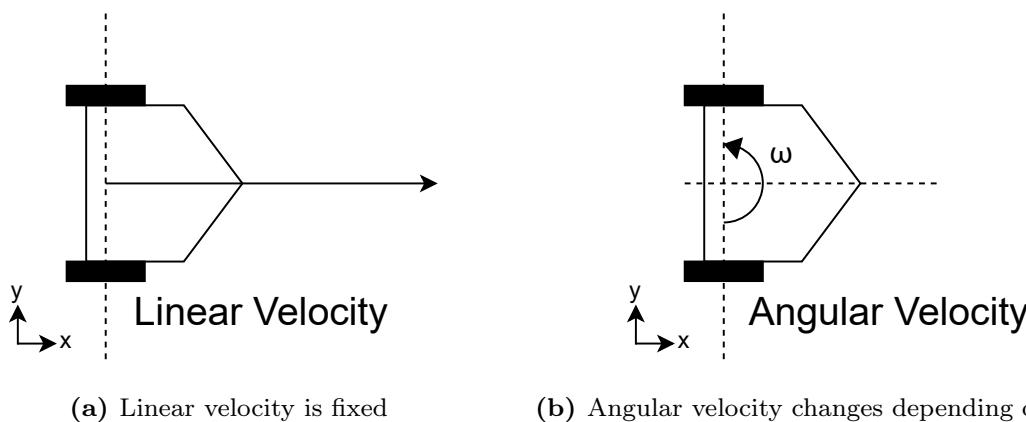
## 第4章

# 提案手法

本章では、従来手法をベースとした提案手法について述べる。

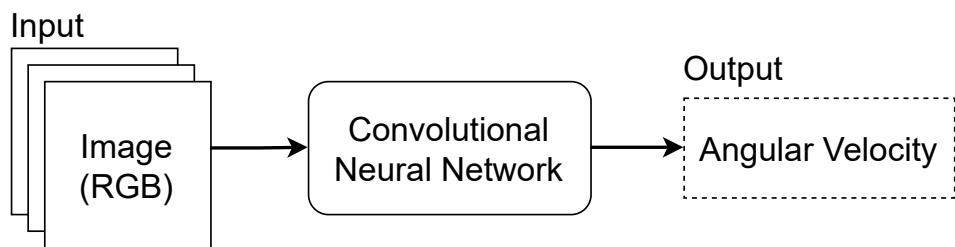
### 4.1 提案手法の概要

本研究は、ルールベース制御器の入力に引き紐ではなく、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] で一定であり、ロボットのヨー方向の角速度  $\omega$  のみが変化する。ルールベース制御器は、反射強度の高い方向にロボットを追従させる制御で、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:** Outputted 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  $\omega$ ）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  $\omega$  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

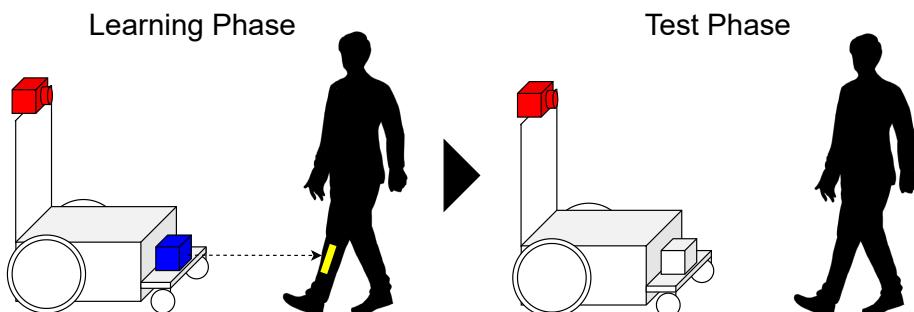
Fig. 4.3 に示すように、ルールベース制御器を用いてロボットを制御するフェーズを学習フェーズ、深層学習器の出力でロボットを制御するフェーズをテストフェーズと呼ぶこととする。以下に、学習フェーズとテストフェーズの主な役割を示す。

#### <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。

#### <テストフェーズ>

2DLiDAR は使用せず、画像を入力とした深層学習器の出力でロボットを制御する。画像に基づいた人追従ができるかを実験により確認する。



**Fig. 4.3:** Sequence of proposed method

## 4.2 学習フェーズ

学習フェーズのシステムを Fig. 4.4 に示す。このフェーズでは、ルールベース制御器を用いてロボットを制御する。ルールベース制御器は、反射強度の高い方向にロボットを追従させる制御で、Fig. 4.5 に示すように、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、ロボットが人に追従する。このとき、入力は 2DLiDAR の反射強度であり、出力はロボットのヨー方向の角速度  $\omega$  となる。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 end-to-end 学習する。

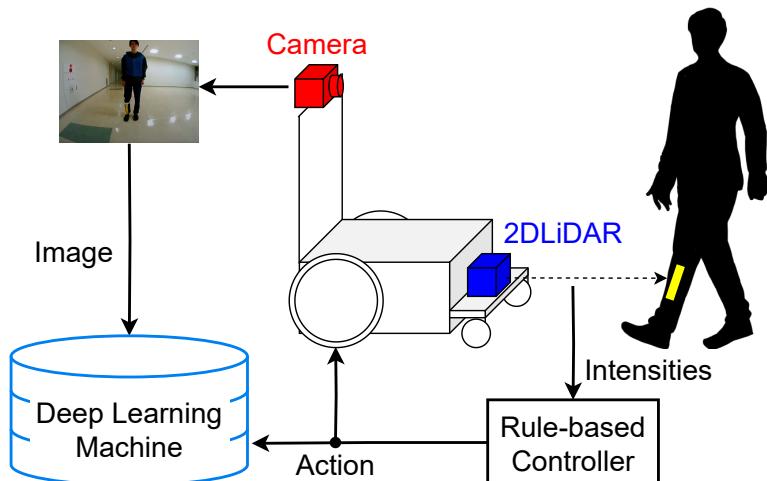


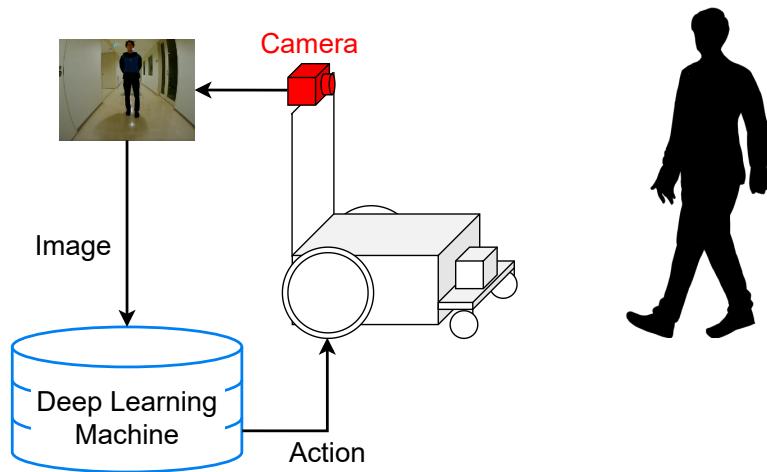
Fig. 4.4: System in the learning phase



Fig. 4.5: Attachment retroreflective tape

### 4.3 テストフェーズ

テストフェーズのシステムを Fig. 4.6 に示す。このフェーズでは、学習フェーズで学習したモデルを用いる。つまり、2DLiDAR の反射強度を入力とした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。そのため、再帰反射テープは必要でない (Fig. 4.7)。



**Fig. 4.6:** System method in the test phase



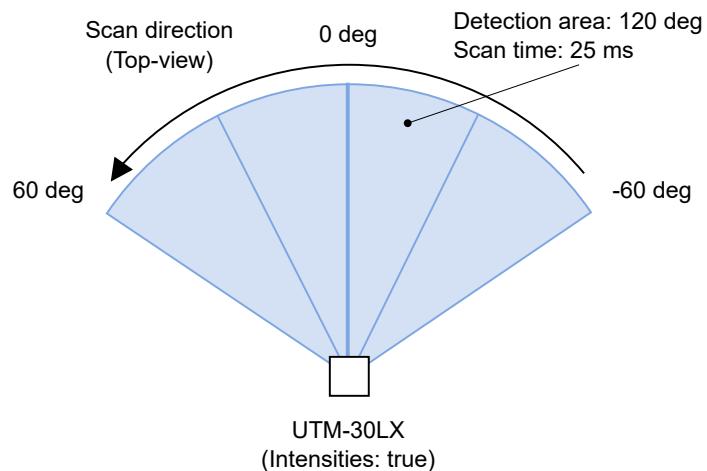
**Fig. 4.7:** Without retroreflective tape

## 4.4 反射強度を計測する装置

本研究で使用した 2DLiDAR は北陽電機社製の UTM-30LX[16] である。このセンサは、ROS 上で提供されている `urg_node`[13] というパッケージを使用することでデータの取得ができる。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 [13]。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]) とした。また、1 回のスキャンは 25 [ms] 時間かかり、-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 | Experiment |
|----------------|---------|------------|
| intensities    | false   | true       |
| angle_min      | -       | -1.047     |
| angle_max      | -       | 1.047      |



**Fig. 4.8:** Condition of scan

## 4.5 ルールベース制御器

ルールベース制御器は、2DLiDARの反射強度を入力とし、式(4.1)よりロボットのヨー方向の角速度を出力する。これにより、ロボットは最大反射強度の方向に追従する。ルールベース制御器からの出力をTable 4.2とFig. 4.9に示す。制御器では、角速度 $\omega$ が0 [rad/s]になると、カメラの中心に追従対象者が来る。

$\omega$  : ロボットのヨー方向の角速度 [rad/s]

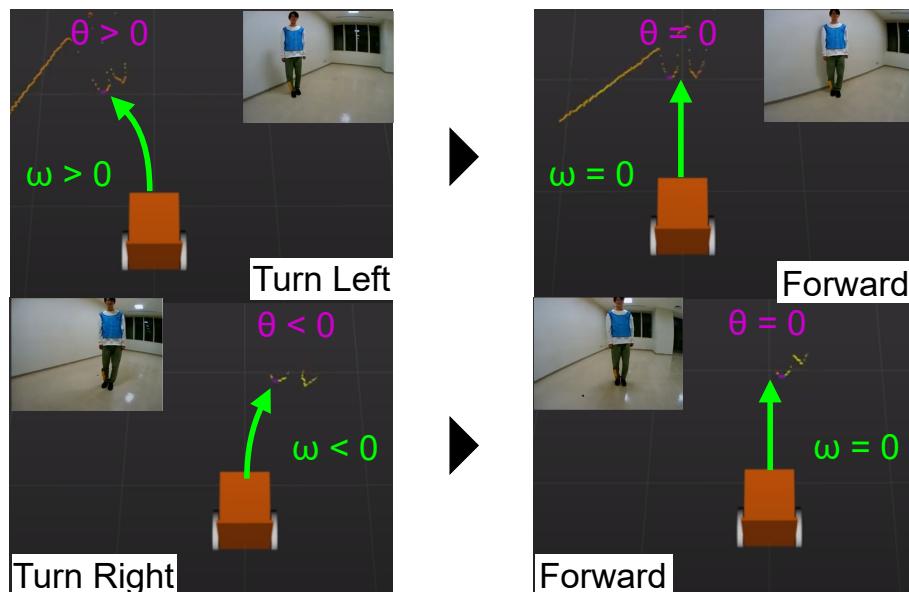
$\theta_{\max}$  : 2DLiDAR の最大検出角度 [rad]

$\theta$  : 2DLiDAR の反射強度が最大の角度 [rad]

$$\omega = \frac{1}{\theta_{\max}} \times \theta \quad (4.1)$$

**Table 4.2:** Output from rule-based controller

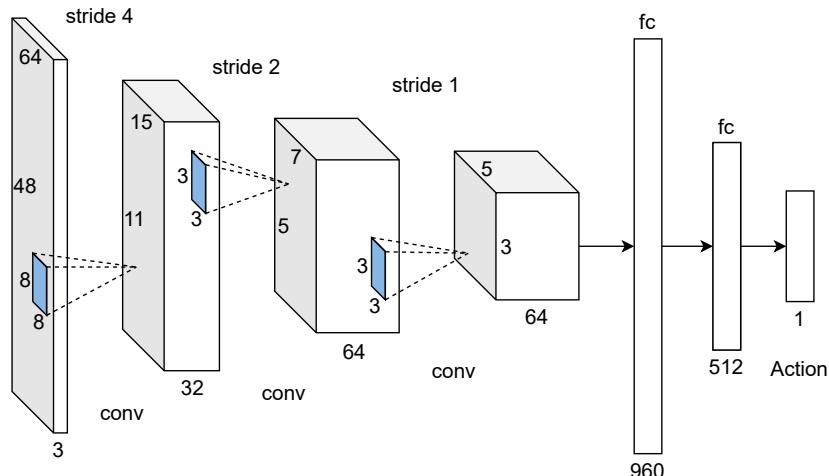
| Control rule [rad]                  | Linear velocity [m/s] | Angular velocity [rad/s] |
|-------------------------------------|-----------------------|--------------------------|
| $\text{angle\_max} \geq \theta > 0$ | 0.2                   | $1 \geq \omega > 0$      |
| 0                                   | 0.2                   | 0                        |
| $0 < \theta \leq \text{angle\_min}$ | 0.2                   | $0 > \omega \geq -1$     |



**Fig. 4.9:** Action from rule-based controller

## 4.6 ネットワーク構造

ネットワークを Fig. 4.10 に示す。これは、深層学習フレームワークである PyTorch[17] 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. 4.10:** Architecture of the network

**Table 4.3:** Parameters of network configured with PyTorch

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

## 第 5 章

# 2DLiDAR の反射強度の実験

本章では、ルールベース制御器で使用する 2DLiADR の反射強度についての実験を行う。

### 5.1 実験の概要

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 の最大検出範囲  $\theta_{\max}$  を 0 [rad] で制限して、前の物体しか検出しないようにしている。

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

## 5.2 実験の目的

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、最大反射強度の方向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検出してしまうと人追従行動を継続できないおそれがある。そのため、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（Fig. 5.1）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。

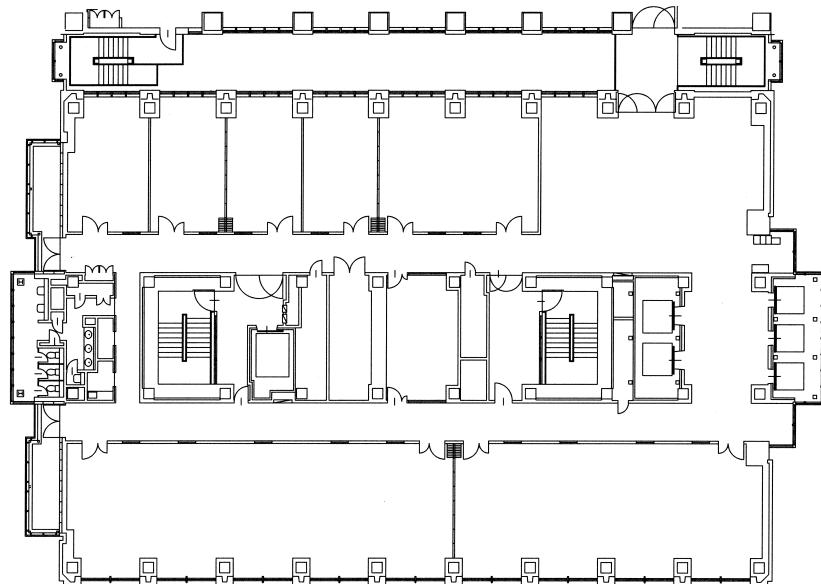


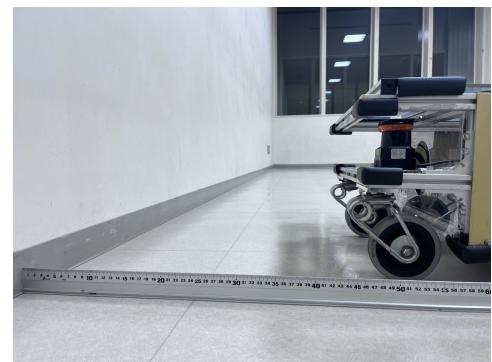
Fig. 5.1: The environment of the experiment

### 5.3 壁の反射強度

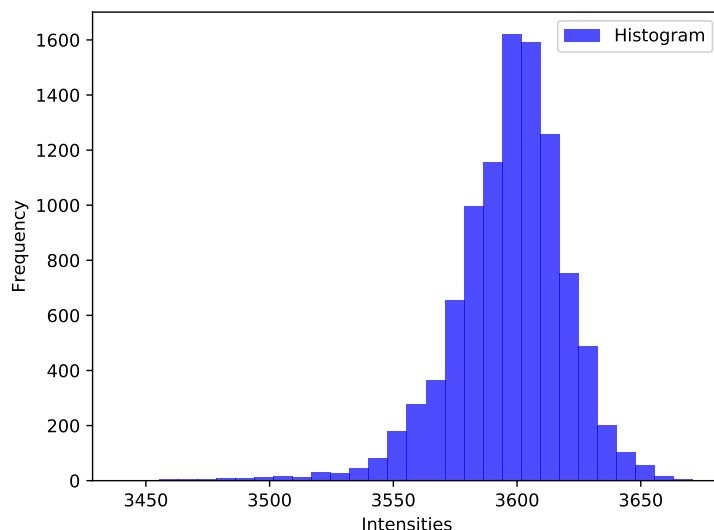
実験場所である、千葉工業大学津田沼キャンパス2号館3階の廊下の壁の反射強度を計測した。実験の様子をFig. 5.2に示す。2DLiDARを壁に向けて正面に配置し、距離によって反射強度の値が変化するのを防ぐため、壁から約500 [mm]離れたところに固定した。また、約250秒間データを収集した。計測した結果をFig. 5.3に示す。平均値は約3600であり、分散は約546であった。



(a) View from behind

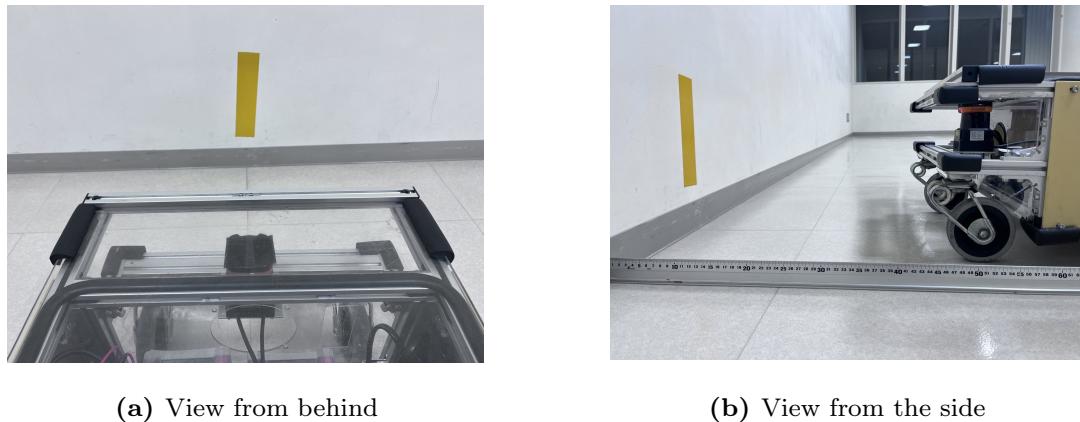


(b) View from the side

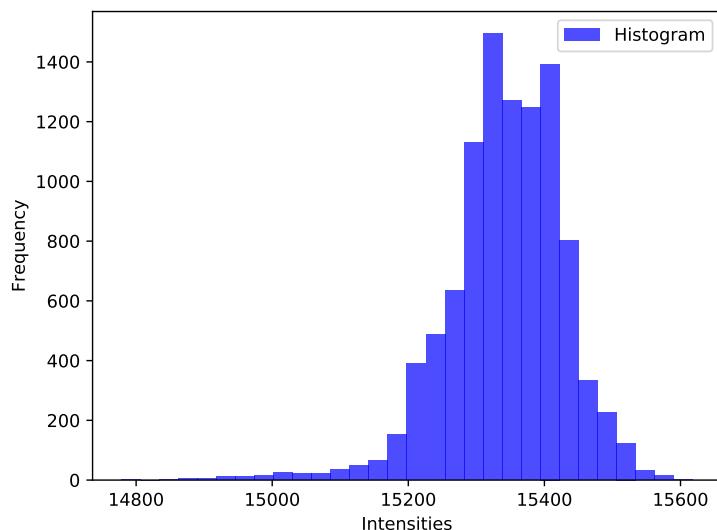
**Fig. 5.2:** Measure the reflection intensity of the wall**Fig. 5.3:** Histogram of reflection intensity of wall

## 5.4 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. 5.4 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。また、約 250 秒間データを収集した。計測した結果を Fig. 5.5 に示す。平均値は約 15300 であり、分散は約 7310 であった。



**Fig. 5.4:** Measure the reflection intensity of retroreflective tape



**Fig. 5.5:** Histogram of reflection intensity of retroreflective tape

## 5.5 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. 5.6 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場所であり、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 250 秒間歩き回ることでデータを収集した。計測した結果を Fig. 5.7 に示す。平均値は約 2102 であり、分散は約 676261 であった。

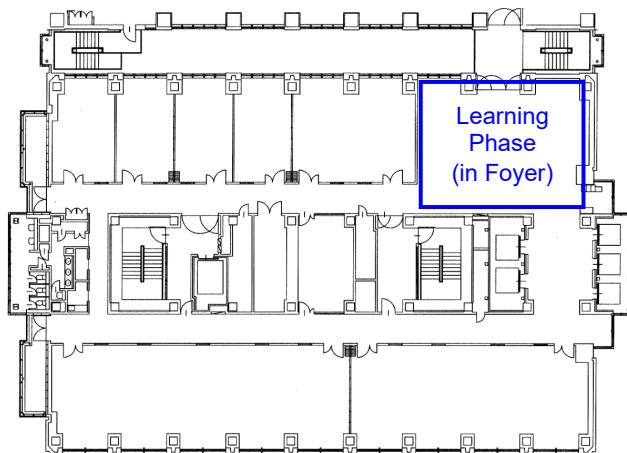


Fig. 5.6: Measure the reflection intensity of foyer

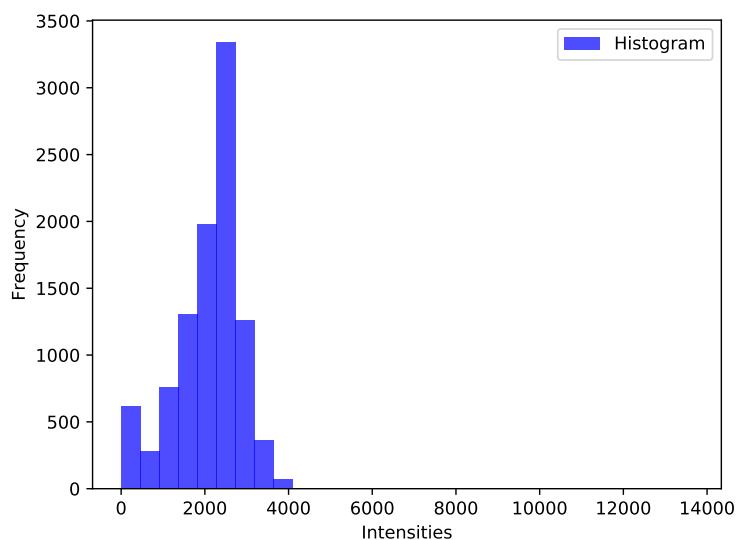
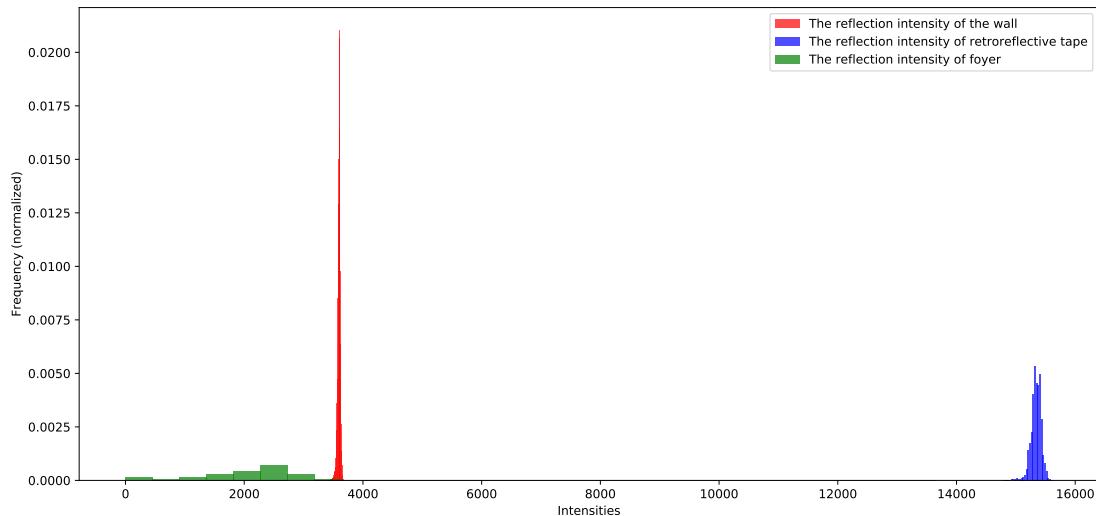


Fig. 5.7: Histogram of reflection intensity of foyer

## 5.6 計測した反射強度の比較

計測した反射強度の比較を Fig. 5.8 と Table 5.1 に示す。これらの結果から、再帰反射テープよりも反射強度の高いものは存在せず、ルールベース制御器が正常に機能すれば、2DLiDAR の反射強度を利用した人追従行動が可能であることを確認した。



**Fig. 5.8:** Comparison of reflection intensity by histogram

**Table 5.1:** Comparison of reflection intensity

| Experiments                                      | Average | Variance |
|--|---------|----------|
| The reflection intensity of the wall             | 3600    | 546      |
| The reflection intensity of retroreflective tape | 15300   | 7310     |
| The reflection intensity of foyer                | 2102    | 676261   |

## 第6章

# 実験

本章では、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。

### 6.1 実験の概要

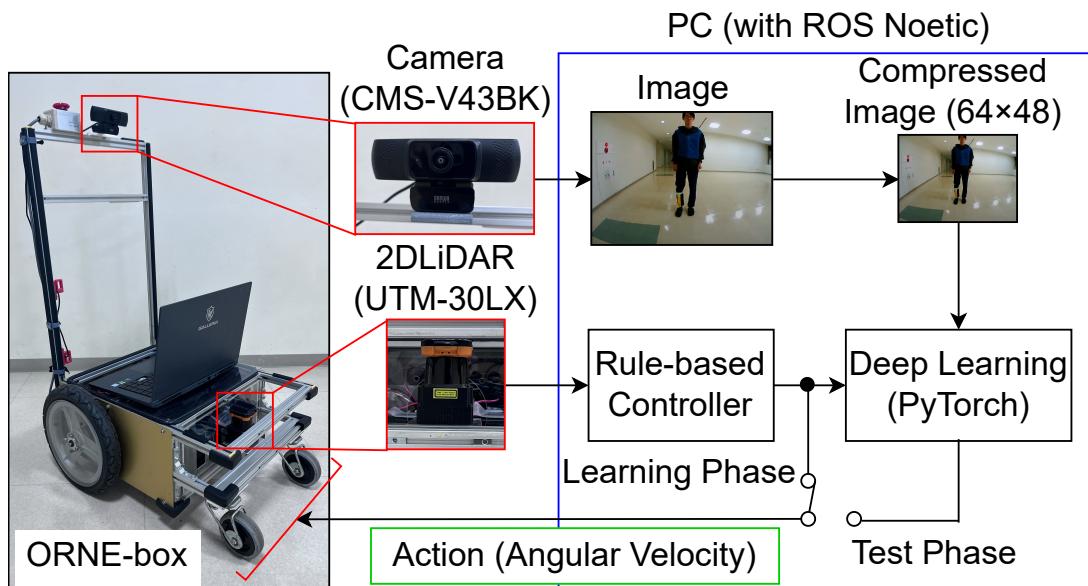
提案手法の有効性を確認するため、実ロボットを用いた実験を 10 回行う。実験では、学習フェーズの後にテストフェーズに移行する。実験環境は、千葉工業大学津田沼キャンパス 2 号館 3 階の廊下であり、天候による影響を少なくするために、夜間に実施した。また、服装による影響を少なくするため、追従対象者は Fig. 6.1 に示す青いビブスを着用して実験を行った。学習フェーズでは、2DLiDAR の反射強度を利用するため、追従対象者に再帰反射テープを装着する必要がある。



**Fig. 6.1:** Sequence of the experiment

## 6.2 実験装置

本研究で使用した実験装置を Fig. 6.2 に示す。ハードウェアは、T-frog プロジェクトの i-Cart mini[18] をベースとしたロボットである、ORNE-box[19][20] 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 6.1 に示す。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. 6.2:** The developed system

**Table 6.1:** Laptop computer specifications

|          |                          |
|----------|--------------------------|
| Computer | GALLERIA GCR2070RGF-QC-G |
| OS       | Ubuntu 20.04 LTS         |
| ROS      | Noetic                   |

### 6.3 実験方法

実験では、学習フェーズの後にテスト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. 6.3 に示す青枠で囲われた場所（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。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 となる。

#### <テストフェーズ>

テストフェーズでは、2DLiDAR の反射強度を利用しないため、再帰反射テープを必要としない。また、学習フェーズとは異なり、Fig. 6.3 に示す赤枠で囲われたコースを壁に衝突せず 1 周できるかテストする。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、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 分間の学習とテスト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。

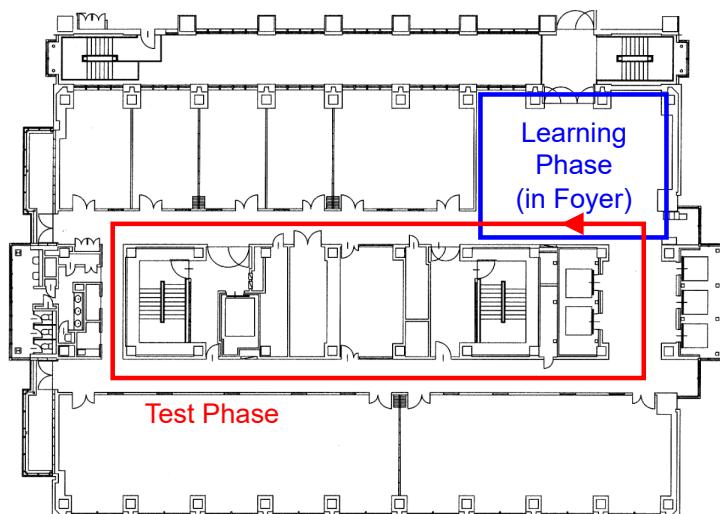


Fig. 6.3: Learning and test phase courses

## 6.4 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. 6.4 に示す。2DLiDAR の反射強度を入力としたルールベース制御器の出力によって、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継続不可能になることはなかった。

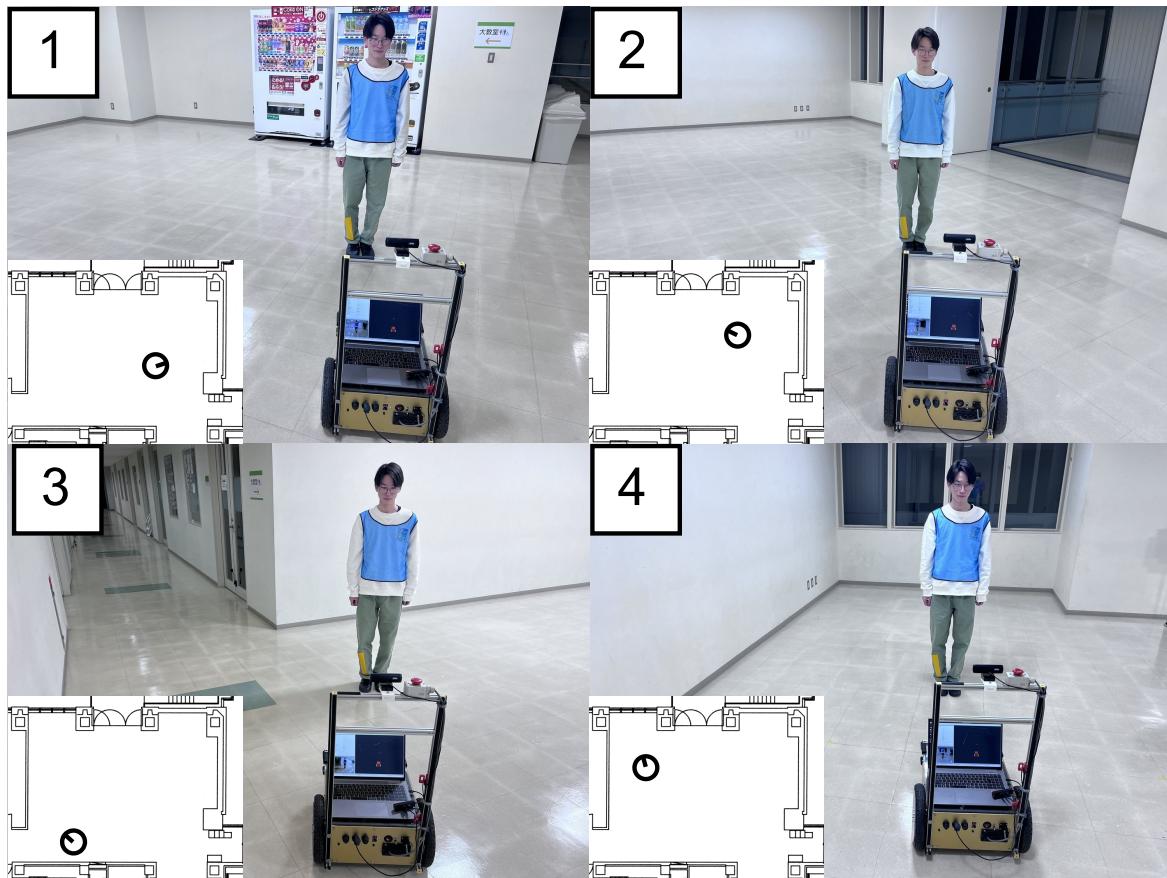


Fig. 6.4: Learning phase in experiment

## &lt;テストフェーズ&gt;

テストフェーズにおける実験の様子を Fig. 6.5 に示す。また、それぞれの番号は以下の 4 つのことを表している。テストコースでは、3 番に示すようにビブスに似た青色の壁紙が貼られていたが、問題なく人追従行動を継続する様子が確認できた。

1. スタート地点（ホワイエ）
2. 1 つ目の曲がり角
3. 青色の壁紙
4. エレベータホール

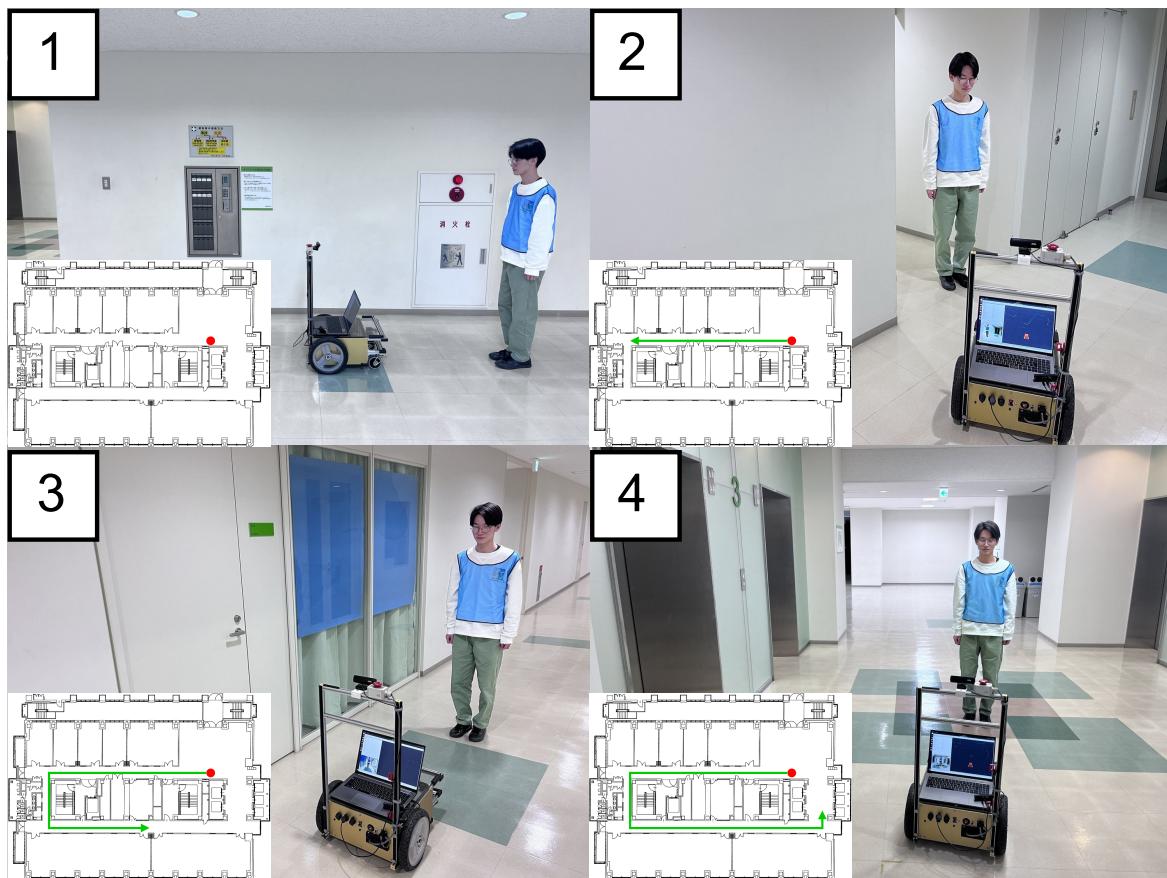
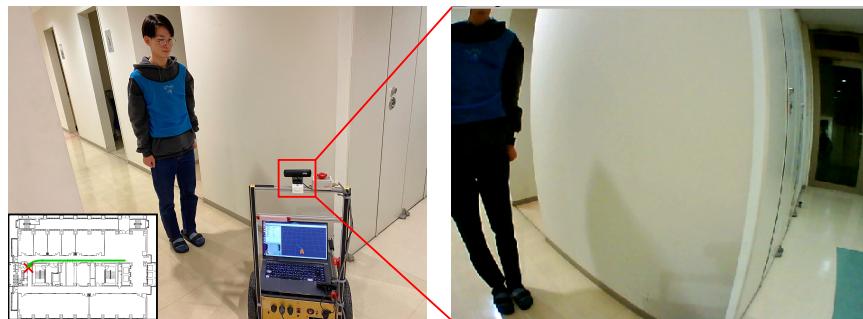


Fig. 6.5: Test phase in experiment

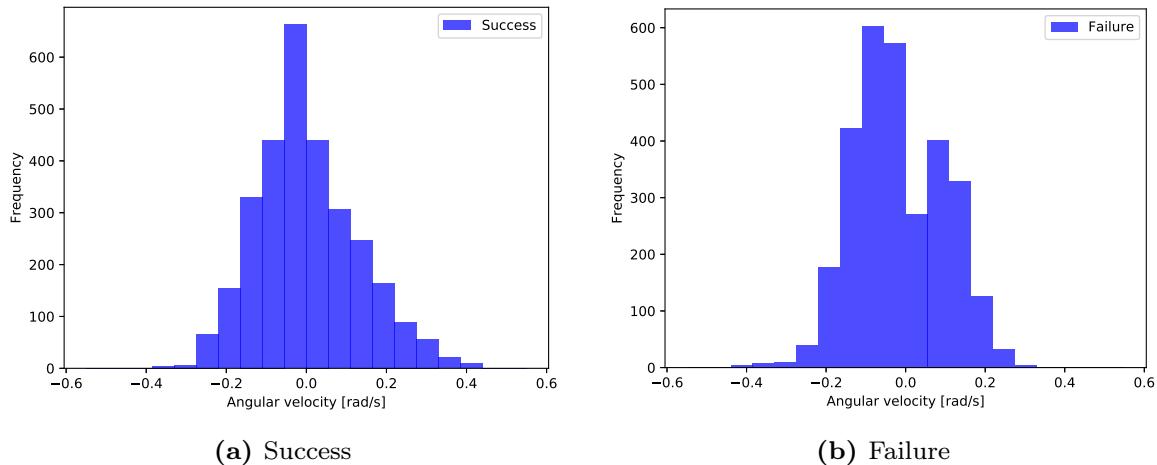
実験結果を Table 6.2 に示す。10回の試行中、9回は人追従行動が継続不可能になることなく、画像を入力とした深層学習器の出力により、コースを1周することができた。一方で、10回の試行中、1回は Fig. 6.6 に示すように、テストコースの1つ目の曲がり角で壁に衝突した。ここで、学習フェーズで取得した、成功時と失敗時の角速度のヒストグラムを Fig. 6.7 に示す。現段階では、学習フェーズでの角速度の違いが失敗の要因と考えているが、これについてはさらなる調査が必要である。

**Table 6.2:** Experiment result

| Experiment      | Number of success |
|-----------------|-------------------|
| Proposed method | 9/10 (90%)        |



**Fig. 6.6:** Failed at the first corner



**Fig. 6.7:** Histogram of angular velocity during learning phase

## 第7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。結果的に、成功率は 90% であったため、提案手法が有効であることを確認した。

## 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] 大島章, 城吉宏泰, 柄川索, 松下裕介, 阪東茂, “既存 AGV を超える特長を持った協働運搬ロボット「サウザー」”, 日本ロボット学会誌, Vol. 39, No. 1, pp.65-66, (2021).
- [10] Yann LeCun et al, “Gradient-based learning applied to document recognition”, Proceedings of the IEEE 86.11 (1998): 2278-2324.

- [11] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton, “Imagenet classification with deep convolutional neural networks”, Advances in neural information processing systems, (2012).
- [12] ROS. Documentation. <https://docs.ros.org/>. (Accessed on 12/31/2023).
- [13] ROS Wiki. urg\_node. [https://wiki.ros.org/urg\\_node](https://wiki.ros.org/urg_node). (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] 北陽電氣. UTM-30LX. <https://www.hokuyo-aut.co.jp/search/single.php?serial=21>. (Accessed on 12/27/2023).
- [17] PyTorch. Documentation. <https://pytorch.org/docs/main/>. (Accessed on 12/29/2023).
- [18] 渡辺敦司. T-Frog Project. [http://t-frog.com/products/icart\\_mini/](http://t-frog.com/products/icart_mini/). (Accessed on 12/29/2023).
- [19] 井口颯人, 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).
- [20] 井口颯人, 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

実験動画を YouTube で公開している。

- リンク

<https://youtu.be/6P8uNoAsuek>

## 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。