

Master's Thesis

# **Human Activity Recognition with Wi-Fi Using Machine Learning for Building Automation and Energy Saving**

**Tanmay Mane**

Karlsruhe Institute of Technology (KIT)  
Institut fuer Technik der Informationsverarbeitung (ITIV)

Heads of the Institute:

Prof. Dr.-Ing. Dr. h. c. Jürgen Becker  
Prof. Dr.-Ing. Eric Sax  
Prof. Dr. rer. nat. Wilhelm Stork

Examiner: Prof. Dr.-Ing. Dr. h. c. Jürgen Becker  
Second Examiner:  
Supervisor: M. Sc. Iris Fürst-Walter, Prof. Dr.-Ing. Adolfo Bauchspiess

November 13, 2023

I declare that I have developed and written the enclosed thesis completely by myself. I have submitted neither parts of nor the complete thesis as an examination elsewhere. I have not used any other than the aids that I have mentioned. I have marked all parts of the thesis that I have included from referenced literature, either in their original wording or paraphrasing their contents. This also applies to figures, sketches, images and similar depictions, as well as sources from the internet.

Karlsruhe, November 13, 2023

---

Tanmay Mane

# Abstract

In recent years, building energy management has emerged as a crucial area of interest due to the escalating energy consumption and carbon emissions attributed to global buildings, accounting for approximately 30 % of total energy usage. This heightened consumption raises profound concerns about energy and the environment. The problem of increasing energy consumption, particularly in Heating, Ventilation, and Air-Conditioning (HVAC) systems in building automation, has been growing rapidly. This study seeks to investigate how variations in design parameters, such as changes in antenna configurations, can influence HVAC energy consumption within building automation scenarios. We aim to address this issue by leveraging modern machine-learning models and integrated engineering techniques. The central focus of our investigation lies in the realm of Human Activity Recognition (HAR) based on Channel State Information (CSI). Wi-Fi CSI data, attainable through 802.11 OFDM devices, offers the potential to enhance Internet of Things (IoT) networks with additional environmental services while ensuring data privacy. Our proposed solution revolves around the analysis of HAR data using Convolutional Neural Networks (CNNs) and Reinforcement Learning (RL). Data acquisition was performed using ESP32 devices and the ESP-Now protocol within a network consisting of 2 senders and 5 receivers, each equipped with a single antenna, totaling 10 links. This network was tested across various locations and with different individuals. Our approach encompasses two machine learning models: 2D Convolutional Neural Networks (CNNs) for activity recognition and Deep Q-learning Networks (DQN) as a feature extraction and classification algorithm. Our study primarily examines medium-scale HAR, focusing on four main activities (sitting, standing, walking, and an empty room) performed by four different individuals under varying orientations and locations. It involves the observation of design parameter changes, including the placement of multiple Wi-Fi devices, antenna orientations, and different locations. Our initial design strategy entails training our model with known individuals and locations, and subsequently assessing its performance when dealing with unknown individuals and locations. Through experiments conducted with four different individuals at location 1, our CNN model achieved an accuracy rate exceeding 93 % in recognizing diverse human activities, while the RL model demonstrated an accuracy of 75 %. When faced with unknown locations while knowing the individual, the CNN accuracy reached 88 %, surpassing RL's 73 %. In situations where both the location and individual are unknown, CNN achieved 79 %, whereas RL's accuracy remained at 69 %. With this developed system, we anticipate a reduction in the human effort required for signal processing and hyper-parameter tuning. Furthermore, the thesis highlights future research directions and challenges in reinforcement learning.

# Zusammenfassung

In den letzten Jahren hat sich das Energiemanagement in Gebäuden zu einem wichtigen Thema entwickelt, da der Energieverbrauch und die Kohlendioxidemissionen von Gebäuden weltweit stark ansteigen und etwa 30 % des gesamten Energieverbrauchs ausmachen. Dieser erhöhte Verbrauch gibt Anlass zu tiefgreifenden Bedenken hinsichtlich Energie und Umwelt. Das Problem des steigenden Energieverbrauchs, insbesondere bei Heizungs-, Lüftungs- und Klimatisierungssystemen (HVAC) in der Gebäudeautomation, hat stark zugenommen. In dieser Studie soll untersucht werden, wie Variationen von Designparametern, wie z. B. Änderungen der Antennenkonfigurationen, den Energieverbrauch von HLK-Systemen in der Gebäudeautomation beeinflussen können. Wir wollen dieses Problem durch den Einsatz moderner Machine-Learning-Modelle und integrierter Engineering-Techniken angehen. Der Schwerpunkt unserer Untersuchung liegt im Bereich der menschlichen Aktivitätserkennung (Human Activity Recognition, HAR) auf der Grundlage von Channel State Information (CSI). Wi-Fi CSI-Daten, die über 802.11 OFDM-Geräte gewonnen werden können, bieten das Potenzial, Internet of Things (IoT)-Netzwerke mit zusätzlichen Umgebungsdiensten zu erweitern und gleichzeitig den Datenschutz zu gewährleisten. Unsere vorgeschlagene Lösung dreht sich um die Analyse von HAR-Daten mithilfe von Convolutional Neural Networks (CNNs) und Reinforcement Learning (RL). Die Datenerfassung erfolgte mit ESP32-Geräten und dem ESP-Now-Protokoll innerhalb eines Netzwerks, das aus 2 Sendern und 5 Empfängern besteht, die jeweils mit einer einzigen Antenne ausgestattet sind, insgesamt also 10 Links. Dieses Netzwerk wurde an verschiedenen Orten und mit verschiedenen Personen getestet. Unser Ansatz umfasst zwei maschinelle Lernmodelle: 2D Convolutional Neural Networks (CNNs) zur Aktivitätserkennung und Deep Q-learning Networks (DQN) als Merkmalsextraktions- und Klassifikationsalgorithmus. Unsere Studie untersucht in erster Linie mittelgroße HAR und konzentriert sich auf vier Hauptaktivitäten (Sitzen, Stehen, Gehen und ein leerer Raum), die von vier verschiedenen Personen in unterschiedlichen Ausrichtungen und an unterschiedlichen Orten ausgeführt werden. Sie umfasst die Beobachtung von Änderungen der Designparameter, einschließlich der Platzierung mehrerer Wi-Fi-Geräte, der Antennenausrichtung und verschiedener Standorte. Unsere anfängliche Entwurfsstrategie sieht vor, unser Modell mit bekannten Personen und Standorten zu trainieren und anschließend seine Leistung bei unbekannten Personen und Standorten zu bewerten. In Experimenten mit vier verschiedenen Personen an Standort 1 erreichte unser CNN-Modell eine Genauigkeit von über 93 % bei der Erkennung verschiedener menschlicher Aktivitäten, während das RL-Modell eine Genauigkeit von 75 % aufwies. Bei unbekannten Orten und bekannten Personen erreichte die CNN-Genauigkeit 88 % und übertraf damit die RL-Genauigkeit von 73 %. In Situationen, in denen sowohl der Ort als auch die Person unbekannt sind, erreichte CNN 79 %, während die Genauigkeit von RL bei 69 % blieb. Mit diesem entwickelten System erwarten wir eine Verringerung des menschlichen Aufwands für die Signalverarbeitung und die Abstimmung der Hyperparameter. Darüber hinaus

zeigt die Arbeit zukünftige Forschungsrichtungen und Herausforderungen im Bereich des Reinforcement Learning auf.

# Contents

<b>Abstract</b>	<b>i</b>
<b>Zusammenfassung</b>	<b>ii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Problem statement . . . . .	1
1.2 Objective . . . . .	2
1.2.1 Device-Free Wi-Fi Sensing . . . . .	2
1.3 Deep Reinforcement Learning . . . . .	3
1.4 The remainder of this Thesis is organized as follows: . . . . .	5
<b>2 Theoretical Background</b>	<b>6</b>
2.1 ESP-32 NodeMCU as a sensing Device . . . . .	6
2.1.1 Features: . . . . .	6
2.2 ESP-NOW Protocol . . . . .	8
2.2.1 ESPNOW Protocol Introduction . . . . .	8
2.2.2 Vender-Specific Action Frame . . . . .	9
2.2.3 Vender Specific Field . . . . .	10
2.3 OFDM-MIMO . . . . .	10
2.3.1 OFDM . . . . .	10
2.3.2 Characteristics of OFDM (Advantages) . . . . .	11
2.3.3 MIMO . . . . .	11
2.3.4 Characteristics of MIMO . . . . .	12
2.4 Channel State Information . . . . .	12
2.4.1 Introduction to CSI . . . . .	12
2.5 Convolutional Neural Network . . . . .	14
2.6 Reinforcement Learning . . . . .	14
2.6.1 Reinforcement Learning Definition and Architecture . . . . .	14
2.6.2 Policy . . . . .	15
2.6.3 Reward . . . . .	15
2.6.4 Value function . . . . .	16
2.6.5 Model of the environment . . . . .	16
2.7 Reinforcement Learning Categories . . . . .	16
2.8 Overview of Deep Reinforcement Learning . . . . .	18
2.8.1 Q-Leraning . . . . .	18
2.8.2 Method to determine the Q-value . . . . .	19
2.8.3 Q-Table . . . . .	19
2.9 Deep Q - Network . . . . .	21

<b>3 State of the Art</b>	<b>22</b>
3.1 Location- and Person-Independent Activity Recognition with WiFi, Deep Neural Networks, and Reinforcement Learning . . . . .	22
3.1.1 Introduction . . . . .	22
3.1.2 Setup . . . . .	23
3.1.3 Design . . . . .	23
3.2 WiAgent: Link Selection for CSI-Based Activity Recognition in Densely Deployed Wi-Fi Environments . . . . .	24
3.2.1 Introduction . . . . .	24
3.2.2 Experimental setup . . . . .	25
3.2.3 Evaluation . . . . .	26
3.3 Evaluation of State of Art methods . . . . .	27
<b>4 Framework of the Implemented Human Activity Recognition</b>	<b>28</b>
4.1 Data Acquisition . . . . .	28
4.2 Data Pre-processing . . . . .	29
4.3 Feature Extraction Algorithm . . . . .	29
4.4 Reinforcement Learning . . . . .	29
4.5 Application . . . . .	30
<b>5 Implementation</b>	<b>31</b>
5.1 Data Acquisition . . . . .	31
5.1.1 Configuration of Setup . . . . .	31
5.1.2 Channel State Information Transmission workflow . . . . .	32
5.1.3 Training and Testing Environment . . . . .	33
5.1.4 Gathering process . . . . .	34
5.2 Data Preprocessing . . . . .	35
5.2.1 Data Cleaning . . . . .	35
5.2.2 Amplitude and Phase Mapping . . . . .	37
5.2.3 PCA Transformation . . . . .	39
5.3 Feature Representation . . . . .	40
5.3.1 Adoptions and Improvements . . . . .	41
5.4 Reinforcement Learning . . . . .	44
5.4.1 Q - Learning approach . . . . .	44
5.4.2 Adaptation and Improvements . . . . .	45
5.4.3 Flowchart of DQN model . . . . .	45
5.4.4 Validation of RL using DQN . . . . .	47
<b>6 Results and Discussion</b>	<b>48</b>
6.1 Transfer Learning Results . . . . .	48
6.1.1 Cross Environment Validation using CNN . . . . .	49
6.1.2 Cross Environment Validation using Reinforcement Learning (DQN) . . . . .	49
6.2 Overall result . . . . .	51
6.2.1 HAR with Transfer Learning using Vertical Antenna Setup . . . . .	51
6.2.2 HAR with Transfer Learning using Horizontal Antenna Setup . . . . .	52
<b>7 Summary</b>	<b>54</b>
7.1 Objective . . . . .	54
7.2 Methodology . . . . .	54

7.3	Key findings . . . . .	54
7.3.1	Data Acquisition . . . . .	54
7.3.2	Convolutional Neural Networks (CNN) . . . . .	54
7.3.3	Reinforcement Learning (RL) . . . . .	55
7.3.4	Transfer Learning . . . . .	55
7.3.5	Device-free Wi-Fi Sensing . . . . .	55
7.4	Overall Significance . . . . .	55
7.5	Limitations . . . . .	56
<b>8</b>	<b>Conclusion</b>	<b>57</b>
<b>9</b>	<b>Perspectives</b>	<b>58</b>
9.1	Future prospects for Human Activity Recognition (HAR)-based building automation . . . . .	58
9.2	Additional areas where it can be applied . . . . .	58
	<b>Abbreviations</b>	<b>60</b>
	<b>Bibliography</b>	<b>61</b>

# 1 Introduction

## 1.1 Problem statement

Meeting global energy needs in an efficient and sustainable manner is one of the most pressing issues facing society today. The energy consumption from buildings accounts for a large portion of the total global energy consumption and total carbon emissions in the world [1, 2, 3, 4, 5, 6]. For example, global buildings consumed 36% of total energy and generated 37% of total carbon emissions in 2020. [7] Residential buildings had the highest share at 22%, while non-residential buildings had 8%, and the final 6% were related to the construction industry. Moreover, the energy demand for buildings is expected to increase by 50% in the next 30 years [8], [9]. Fifty percent of building energy consumption is used in heating, ventilation, and air-conditioning (HVAC) systems [10]. Under the above background, smart buildings have received more and more attention in recent years, which can provide sustainable, economical, and comfortable operational environments for occupants using many advanced technologies. e.g., Internet of Things (IoT), cloud computing, machine learning, and big data analytics [11, 12, 13]. This can be achieved by developing efficient, smart, and adaptive buildings that go beyond the conventional role of passive energy consumers. Developing advanced Smart Building Energy Management (SBEM) technologies [14], is crucial for optimizing energy consumption, reducing carbon emissions, minimizing energy costs, and enhancing user comfort [15], [16], [17]. However, there are several challenges to address:

- Creating an accurate and efficient building thermal dynamics model is complicated due to complex and random factors [18].
- Uncertain parameters, such as renewable energy generation, electricity prices, and environmental conditions, add complexity [19].
- Coordinating various energy subsystems, like HVAC, Smart grids and energy storage systems, which are temporally and spatially coupled [20].
- Real-time optimization of large-scale building energy problems with traditional methods is time-consuming [19].
- Developing a universal building energy management approach suitable for diverse building environments is challenging [21].

In summary, the development of SBEM technologies must overcome these challenges to achieve energy-efficient and sustainable building operations.

As a general artificial intelligence technology, deep reinforcement learning (DRL), and WiFi sensing for HAR is promising to address the above challenges and have been applied in many fields [22], e.g., games autonomous driving, autonomous IoT, smart buildings, smart

city, wireless networks, Internet of energy, unmanned aerial vehicles, smart microgrids, edge computing, and manufacturing systems [23] [10].

## 1.2 Objective

### 1.2.1 Device-Free Wi-Fi Sensing

The emergence of WiFi sensing is a compelling solution for intelligent building automation due to its non-intrusive and cost-effective nature [24], [25]. WiFi signals have the unique ability to traverse physical barriers such as walls and floors, enabling the monitoring of human activities and environmental changes without the necessity of deploying physical sensors [26]. This offers a "device-free" HAR sensing, eliminating the need for individuals to carry specific devices for tracking [27]. Within this context, researchers have explored HAR utilizing WiFi signals. Their primary focus has been on Received Signal Strength Indication (RSSI) and Channel State Information (CSI) [28]. Early research efforts in WiFi-based HAR employed feature engineering techniques to manually extract relevant features and subsequently train recognition models [29]. They emphasized both time-domain and frequency-domain information for training models geared toward CSI-based activity recognition tasks. Multiple-Input Multiple-Output (MIMO) and Orthogonal Frequency Division Multiplexing (OFDM) are pivotal technologies that enhance the performance of contemporary WiFi systems [23]. MIMO-OFDM provides Channel State Information (CSI), which details the power attenuation and phase shifts between transmitters and receivers at specific carrier frequencies. Beyond its role in optimizing network performance, CSI also lends itself to WiFi-based sensing applications [30]. It captures alterations in WiFi signals resulting from human movements and activities in the vicinity of WiFi transmitters or receivers [31], [32]. Variations in CSI amplitude and phase, induced by these activities, can be harnessed by predefined models or machine learning algorithms for the detection of human motion and recognition of activities [22], [33]. In this Thesis, the use of ESP32 modules with ESP-Now protocol, a MAC-based connection, will investigate MIMO ESP32 sensing networks as a smarter sensing alternative, in comparison to Multi-antenna Access Points. To make WiFi-based activity recognition practical in real-world scenarios,

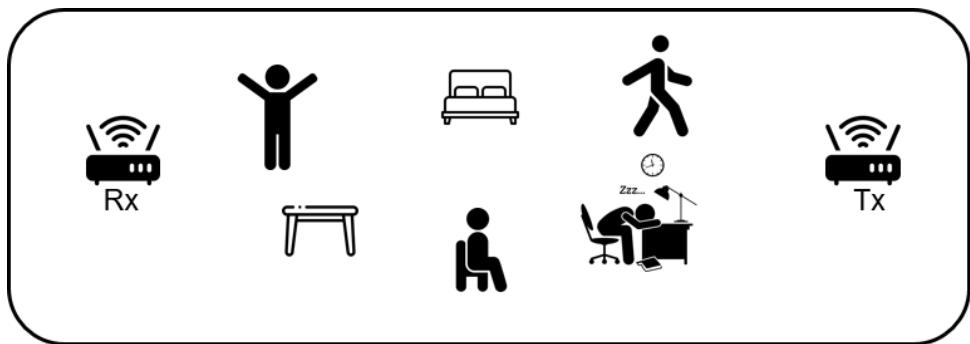


Figure 1.1: CSI Transmission affected by surrounding and different activities

recognition algorithms must be location- and person-independent. While these algorithms are typically trained in controlled environments, they must contend with unknown locations and orientations of WiFi devices during real-world deployments. This challenge

arises because WiFi signals are highly sensitive to various factors. The influence of human activities and the static or dynamic states of WiFi transmitters, receivers, and their surroundings can significantly affect CSI as shown in Figure 1.1. For instance, the positioning and orientation of WiFi receivers relative to target individuals can substantially alter CSI amplitude and phase [34]. Consequently, recognition algorithms trained with data from one WiFi receiver may not perform optimally when applied to another receiver situated differently with distinct antenna orientations. Furthermore, variations in human motion and activity patterns between individuals may render models trained on one person less effective for others whose data was not part of the training or modeling process [34], [35] [36].

### 1.3 Deep Reinforcement Learning

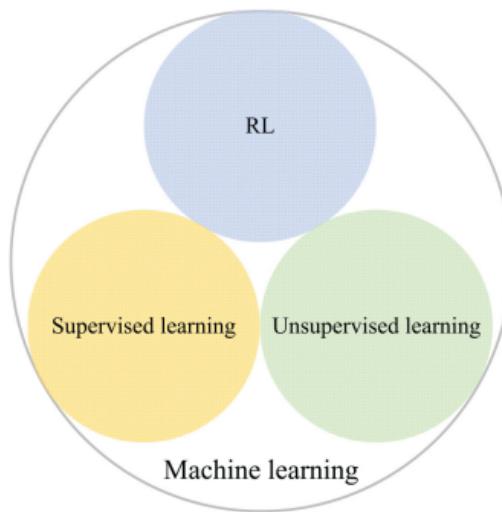


Figure 1.2: Machine Learning

Machine learning can be categorized into three primary types based on feedback mechanisms Figure 1.2: supervised learning, unsupervised learning, and reinforcement learning (RL). In supervised learning, immediate feedback is available as predictions are compared to labeled data, facilitating the improvement of predictors. In contrast, unsupervised learning lacks feedback since input data isn't labeled. In RL, interaction with the environment introduces delayed feedback, where actions in the current state influence future states and actions, and the value of current actions is learned gradually. Supervised and unsupervised learning are typically applied to single-stage problems like regression, classification, clustering, and dimension reduction. RL, on the other hand, excels at solving multistage decision problems. In the context of Sustainable Building Energy Management (SBEM), supervised learning is useful for developing models of building thermal dynamics and reward structures [10]. RL can subsequently leverage these models to reduce the number of interactions with the environment [1], increasing sampling efficiency. Deep Reinforcement Learning (DRL) is a combination of deep learning and RL, where Deep Neural Networks (DNNs) approximate optimal value functions or policies in RL. This

endows DRL with powerful representation capabilities and robust decision-making abilities, particularly in situations involving uncertainty.

Reinforcement Learning (RL) methods have shown their potential to improve image classification [37]. RL is essentially an Exploration and Exploitation learning approach employed to teach an agent how to achieve a specific goal, typically an optimal final state, through a sequence of actions. Researchers have applied RL techniques in various image classification experiments to address challenges and enhance the classification process. This study's primary objective is to offer a comprehensive review of how RL-based methods are employed in image classification procedures. The underlying motivation is to assess the impact of integrating RL techniques on the image classification process [38]. In this thesis, our approach centers around Convolutional Neural Network (CNN) and Q-learning, a foundational reinforcement learning algorithm that facilitates the systematic process of guiding agents in sequential decision-making with the aim of optimizing cumulative rewards. Q-learning leverages a Q-table to assess the anticipated rewards associated with specific actions undertaken within distinct states and progressively refines an optimal policy through an iterative process involving trial and error [37]. This technique finds widespread application in diverse domains, where it effectively balances exploration and exploitation to facilitate well-informed decision-making over time. Additionally, this

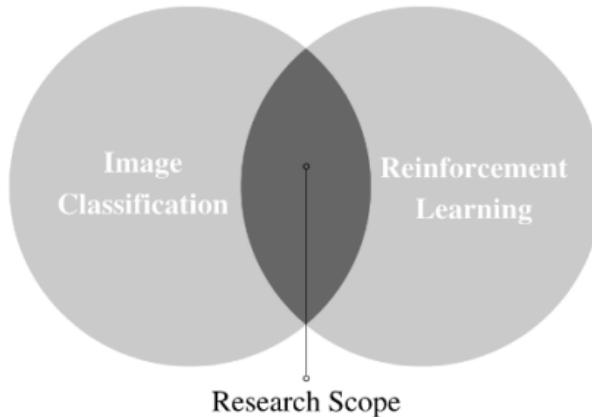


Figure 1.3: Research Scope area Image classification with RL

thesis explores the efficacy of Convolutional Neural Network (CNN) and Deep Q-Network (DQN), a sophisticated reinforcement learning methodology that integrates Q-learning with deep neural networks. DQN harnesses the capabilities of deep learning to approximate Q-values, particularly in scenarios characterized by high-dimensional state spaces. Image classification using DQN is a widely known research topic Figure 1.3 to analyze whether it enhances the accuracy of image classification for building automation and energy saving [39].

## **1.4 The remainder of this Thesis is organized as follows:**

- chapter 2 - Theoretical Background
- chapter 3 - State of the art
- ?? - Framework of the Overall Human Activity Recognition
- chapter 5 - Implementation
- chapter 6 - Result and Discussion
- chapter 7 - Summary
- ?? - Future Scope

# 2 Theoretical Background

The main Theoretical aspects will be presented in this chapter which are as follows:

- ESP32-WROOM-32UE Microprocessor
- ESPNOW Protocol
- MIMO-OFDM
- Convolutional Neural Network (CNN)
- Reinforcement Learning (Q-Learning)

## 2.1 ESP-32 NodeMCU as a sensing Device

The ESP32 is a low-cost microcontroller, the successor of the well-known ESP8266. Designed and produced in 2016 by the Espressif System, a company located in Shanghai, ESP32 has proven its ability for a self-containing Wi-Fi working solution:

We are using the ESP32-WROOM-32UE Microprocessor as a sensing device for data acquisition in this Thesis Figure 2.1. ESP32-WROOM-32E and ESP32-WROOM-32UE are two powerful, generic Wi-Fi + Bluetooth + Bluetooth LE MCU modules that target a wide variety of applications, ranging from low-power sensor networks to the most demanding tasks, such as voice encoding, music streaming and MP3 decoding. ESP32-WROOM-32E comes with a PCB antenna and ESP32-WROOM-32UE with a connector for an external antenna.

### 2.1.1 Features:

#### 2.1.1.1 CPU and On-Chip Memory

- ESP32-D0WD-V3 or ESP32-D0WDR2-V3 embedded,
- Xtensa dual-core 32-bit LX6 microprocessor, up to 240 MHz
- 448 KB ROM
- 520 KB SRAM

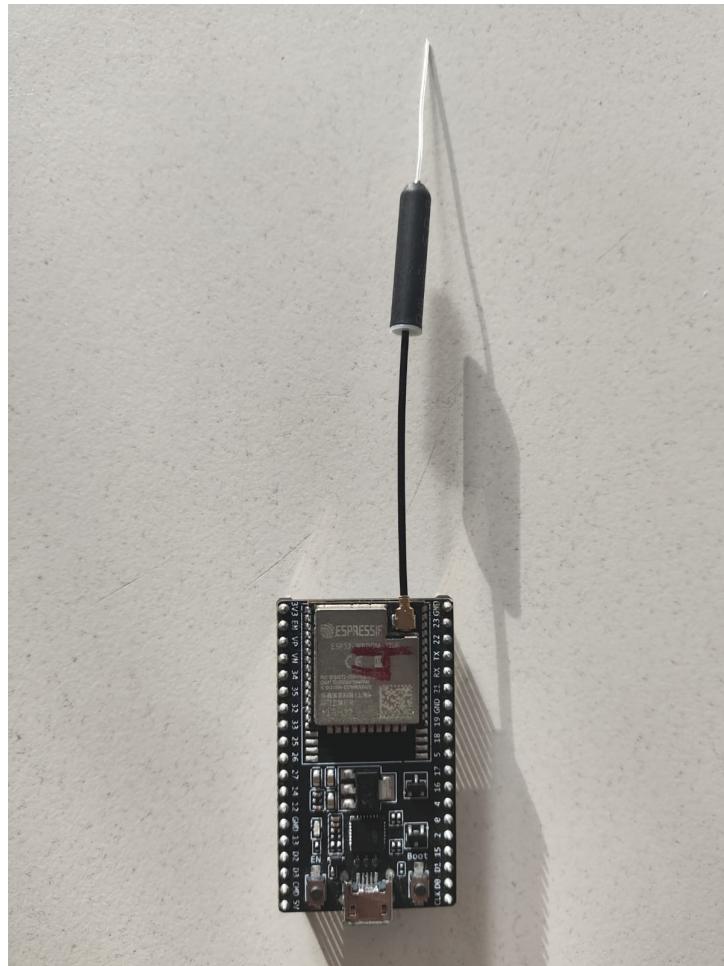


Figure 2.1: ESP32 NodeMCU micro-controller ESP32-WROOM-32UE

### 2.1.1.2 Wi-Fi

- 802.11b/g/n
- Bit rate: 802.11n up to 150 Mbps
- 0.4  $\mu$ s guard interval support
- Center frequency range of operating channel: 2412 – 2484 MHz

### 2.1.1.3 Integrated Components on Module

- 40 MHz crystal oscillator
- 4/8/16 MB SPI flash

### 2.1.1.4 Antenna Options

- ESP32-WROOM-32E: On-board PCB antenna

- ESP32-WROOM-32UE: external antenna via a Connector (Used in this Thesis Figure 2.1)

#### 2.1.1.5 Operating Conditions and Temperature

- Operating voltage/Power supply: 3.0 – 3.6 V
- 85 °C version: –40 – 85 °C
- 105 °C version: –40 – 105 °C

## 2.2 ESP-NOW Protocol

### 2.2.1 ESPNOW Protocol Introduction

ESP-NOW, developed by Espressif, is a connectionless Wi-Fi communication protocol. It facilitates the transmission of application data by encapsulating it within a vendor-specific action frame, allowing for communication between Wi-Fi devices without establishing a formal connection.

This protocol operates at a low data rate, ensuring high efficiency. It is particularly well-suited for battery-powered IoT devices and sensor networks. ESP-NOW facilitates direct and rapid communication between devices, eliminating the need for a conventional Wi-Fi network infrastructure, which simplifies connections and reduces power consumption.

To ensure security, the action frame in ESP-NOW is protected using the CTR with CBC-MAC Protocol (CCMP). This makes ESP-NOW a prevalent choice in applications like smart lighting, remote control, and various sensor-related tasks.

As ESP-NOW is connectionless, the MAC header is a little different from that of standard frames. The FromDS and ToDS bits of the Frame Control field are both 0. The first address field is set to the destination address. The second address field is set to the source address. The third address field is set to broadcast address (0xff:0xff:0xff:0xff:0xff:0xff).

Using the vendor-specific protocol ESP-NOW, created by Express, several devices can be connected without needing Wi-Fi. Many wireless devices use the 2.4 GHz low-power wireless connection. As a result, before the devices may communicate, pairing is required. The connection between two devices is safe and peer-to-peer after pairing is complete, meaning there is no need for a handshake, and the connection is durable. To put it another way, if one of the boards experiences an unexpected power outage or reset, it will immediately establish a connection with its peer and then resume the conversation. The ESP-NOW protocol can transmit 250 bytes of data each time.

The data can be single-duplex or full-duplex (i.e.) it can be unidirectional or bi-directional. Most of the data types are usable. There is no need for a router or an external Wi-Fi source, and data can be encrypted or unencrypted. Depending on the application, one can have anywhere from 2 to 20 nodes interacting with each other. The surroundings

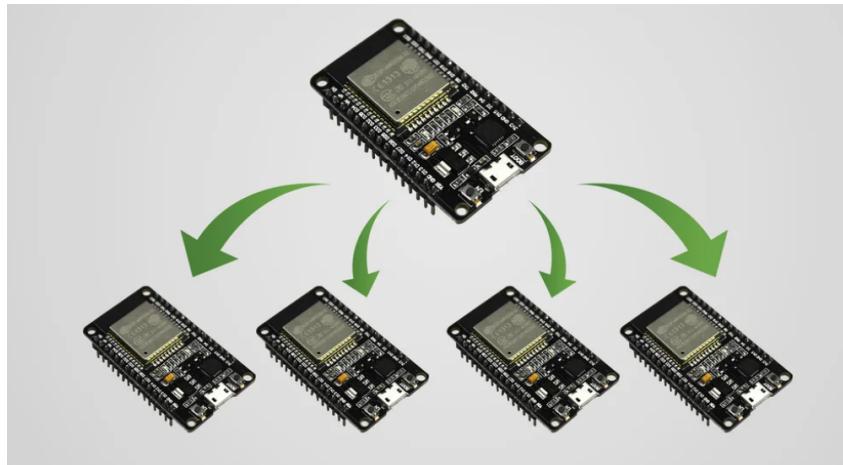


Figure 2.2: Broadcast Connection or one to many

can significantly impact the range, but in the right circumstances (and with suitable antennas), you can obtain over 400 meters.

### 2.2.2 Vender-Specific Action Frame

ESP-NOW employs a vendor-specific action frame to transmit data, with a default bit rate of 1 Mbps. The format of this vendor-specific action frame is as follows Figure 2.3:

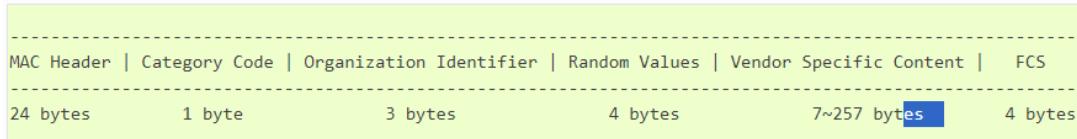


Figure 2.3: vendor-specific action frame

**Category Code:** The Category Code field is set to the value (127) indicating the vendor-specific category.

**Organization Identifier:** The Organization Identifier contains a unique identifier (0x18fe34), which is the first three bytes of the MAC address applied by Espressif.

**Random Value:** The Random Value field is used to prevent relay attacks.

**Vendor Specific Content:** The Vendor Specific Content contains vendor-specific fields as follows Figure 2.4:

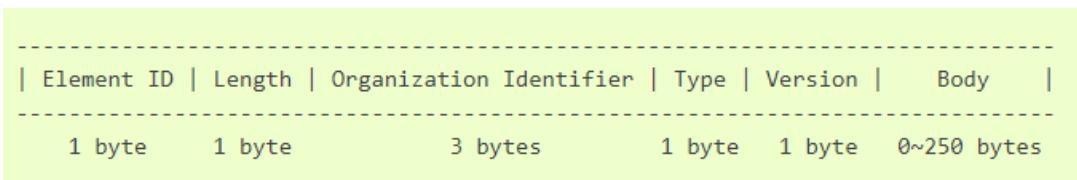


Figure 2.4: Vendor specific Field

### 2.2.3 Vender Specific Field

Element ID: The Element ID field is set to the value (221), indicating the vendor-specific element.

Length: The length is the total length of Organization Identifier, Type, Version, and Body.

Organization Identifier: The Organization Identifier contains a unique identifier (0x18fe34), which is the first three bytes of the MAC address applied by Espressif.

Type: The Type field is set to the value (4) indicating ESP-NOW.

Version: The Version field is set to the version of ESP-NOW.

Body: The Body contains the ESP-NOW data.

## 2.3 OFDM-MIMO

The wireless world today, demands applications supporting high data transmission rates with high spectral efficiency and reliability. This requirement calls for a technology that increases spectral efficiency and offers high quality of service (QoS) to multiple users at the same time. MIMO plus OFDM can offer increased spectral efficiency through the use of multi-carrier modulation and spatial-multiplexing gain offered by MIMO. This technology can also improve link reliability and offer high quality of service (QoS) through the diversity gain achievable by MIMO technology.

### 2.3.1 OFDM

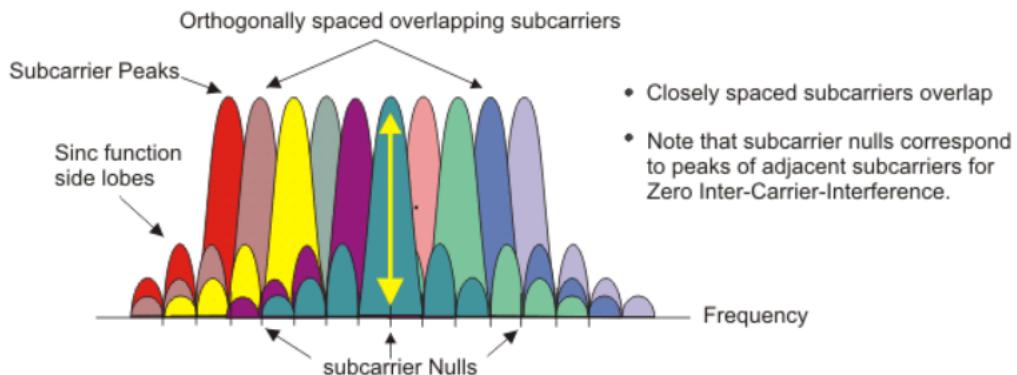


Figure 2.5: OFDM Signal Frequency Spectra

It is a method of transmitting data parallelly on multiple orthogonal carriers for transmitting digital data at a very high rate with high QoS. This principle of transmitting data or messages parallelly on different orthogonal sub-carriers is called orthogonal frequency division multiplexing and was first explained by [40]. OFDM is multi-carrier modulation, where data is transmitted at a lower rate using several parallel sub-carriers or sub-channels. In multi-carrier modulation, data or message to be transmitted is first split and superimposed onto multiple frequency carriers or sub-channels before its transmission, and finally, it is combined at the receiver. In the context of this research, OFDM allows

for the transmission of CSI across multiple frequency carriers, which is then collected and combined at the receiver. This process results in high-speed data transfer and enhances the accuracy and efficiency of the CSI-based human activity recognition system, which is a focal point of this thesis Figure 2.5.

### 2.3.2 Characteristics of OFDM (Advantages)

- Reduced inter-carrier interference (ICI).
- Reduced inter-symbol interference (ISI).
- Carries CSI signal into multiple subcarriers.
- High Spectral efficiency.
- Easy and less complex hardware implementation through the use of Fast Fourier Transform (FFT) is possible.
- Sensitivity to synchronization errors is less.
- (unlike conventional FDM) Tuned filters at the receiver for sub-carriers are not required.
- Possible to adapt to vast varying channel conditions easily without the need for time domain equalization.

### 2.3.3 MIMO

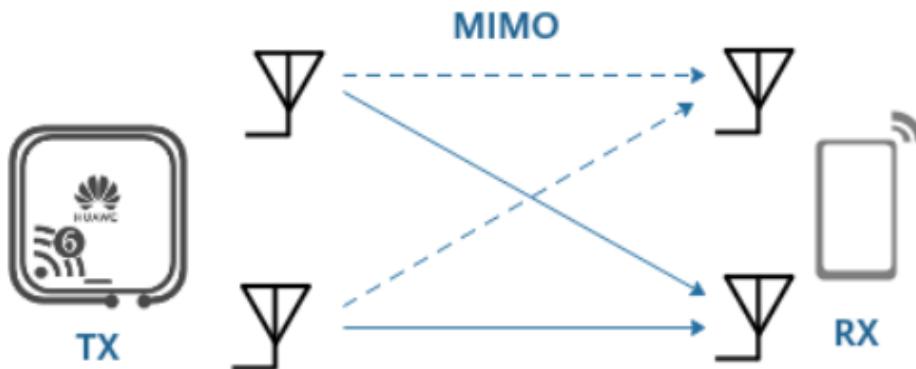


Figure 2.6: Basic structure of MIMO

Multiple-input multiple-output (MIMO) is a multiple antenna technology. In the MIMO system, multiple antennas are used at the transmitter end and also at the receiver end to provide multiple data streams for carrying data. It is an antenna technology that is used in many technologies, such as Wi-Fi(wireless fidelity), LTE(Long Term Evolution), Wi-Max, and many others. This antenna technology offers spatial diversity and increases link capacity and spectral efficiency. It also improves link stability and reliability. Therefore data transmission with extremely high data rate and with high reliability is possible.

### 2.3.4 Characteristics of MIMO

- MIMO can be used to increase spectral efficiency or link reliability.
- Spatial multiplexing: Transmit independent data streams simultaneously to increase spectral efficiency.
- Spatial diversity: Transmit multiple versions of the same transmitted signal to improve link reliability.
- The fundamental trade-off between multiplexing and diversity is possible.
- 802.11n and 802.11ac support both spatial multiplexing and spatial diversity. It also supports using hybrid mode.

## 2.4 Channel State Information

### 2.4.1 Introduction to CSI

CSI represents how wireless signals propagate from the transmitter to the receiver at certain carrier frequencies along multiple paths. For a Wi-Fi system with MIMO-OFDM, CSI is a 3D matrix of complex values representing the amplitude attenuation and phase shift of multi-path Wi-Fi channels. A time series of CSI measurements captures how wireless signals travel through surrounding objects and humans in time, frequency, and spatial domains, so it can be used for different wireless sensing applications. For example, CSI amplitude variations in the time domain have different patterns for different humans, activities, gestures, etc., which can be used for human presence detection, fall detection, activity recognition, gesture recognition, and human identification/authentication. CSI phase shifts in the spatial and frequency domains, i.e., transmit/receive antennas and carrier frequencies, are related to signal transmission delay and direction, which can be used for human localization and tracking. Different Wi-Fi sensing applications have their specific requirements for signal processing techniques and classification/estimation algorithms.

CSI characterizes how wireless signals propagate from the transmitter to the receiver at certain carrier frequencies. CSI amplitude and phase are impacted by multi-path effects including amplitude attenuation and phase shift. Each CSI entry represents the Channel Frequency Response (CFR) as shown in Equation 2.1:

$$H(f; t) = \sum_{n=1}^N a_n(t) e^{-j2\pi f \tau_n(t)} \quad (2.1)$$

where  $a_i(t)$  is the amplitude attenuation factor,  $\tau_{in}(t)$  is the propagation delay, and  $f$  is the carrier frequency. The CSI amplitude  $|H|$  and  $\angle H$  are impacted by the displacements and movements of the transmitter, receiver, and surrounding objects and humans. In other words, CSI captures the wireless characteristics of the nearby environment. These characteristics, assisted by mathematical modeling or machine learning algorithms, can be used for different sensing applications. This is the rationale for why CSI can be used for Wi-Fi sensing. A Wi-Fi channel with MIMO is divided into multiple sub-carriers by

OFDM. To measure CSI, the Wi-Fi transmitter sends Long Training Symbols (LTFs), which contain pre-defined symbols for each sub-carrier, in the packet preamble. When LTFs are received, the Wi-Fi receiver estimates the CSI matrix using the received signals and the original LTFs. For each sub-carrier, the Wi-Fi channel is modeled by

$$\mathbf{y} = \mathbf{Hx} + \mathbf{n} \quad (2.2)$$

where  $\mathbf{y}$  is the received signal,  $\mathbf{x}$  is the transmitted signal,  $H$  is the CSI matrix, and  $\mathbf{n}$  is the noise vector. In real-world Wi-Fi systems, the measured CSI is impacted by multi-path channels, transmit/receive processing, and hardware/software errors. The measured baseband-to-baseband CSI is

$$\mathbf{H}_{i,j,k} = \underbrace{\sum_{n=1}^N a_n e^{-j2\pi \frac{d_{i,j,n} f_k}{c}}}_{\text{Multi-Path Channel}} \underbrace{e^{-j2\pi \tau_i f_k}}_{\text{Cyclic Shift Diversity}} \underbrace{e^{-j2\pi \rho f_k}}_{\text{Sampling Time Offset}} \underbrace{e^{-j2\pi \eta \left( \frac{f'_k}{f_{k-1}} \right) f_k}}_{\text{Sampling Frequency Offset}} \underbrace{q_{i,j} e^{-j2\pi \zeta_{i,j}}}_{\text{Beamforming}} \quad (2.3)$$

where  $d_{i,j,n}$  is the path length from the  $i$ -th transmit antenna to the  $j$ -th receive antenna of the  $n$ -th path  $f_k$  is the carrier frequency,  $\tau_i$  is the time delay from Cyclic Shift Diversity (CSD) of the  $i$ -th transmit antenna,  $\rho$  is the Sampling Time Offset (STO),  $\eta$  is the Sampling Frequency Offset (SFO), and  $q_{i,j}$  and  $\zeta_{i,j}$  are the amplitude attenuation and phase shift of the beamforming matrix. WiFi sensing applications need to extract the multi-path channel that contains the information of how the surrounding environment changes

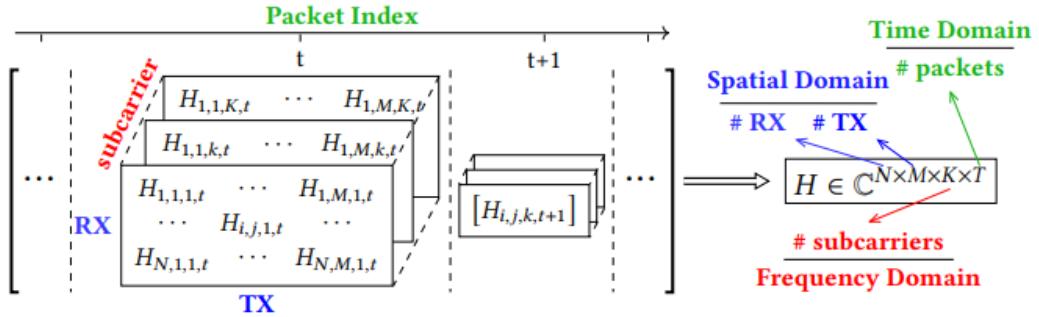


Figure 2.7: CSI matrices for MIMO-OFDM channels. It captures multi-path channel variations, including amplitude attenuation and phase shifts, in spatial, frequency, and time domains

A time series of CSI matrices characterizes MIMO channel variations in different domains, i.e., time, frequency, and spatial, as shown Figure 2.7. For a MIMO-OFDM channel with  $M$  transmit antennas,  $N$  receive antennas, and  $K$  sub-carriers, the CSI matrix is a 3D matrix  $\mathbf{H} \in \mathbb{C}^{N \times M \times K}$  representing amplitude attenuation and phase shift of multi-path channels. CSI provides much more information than the Received Signal Strength Indicator (RSSI). The 3D CSI matrix is similar to a digital image with spatial resolution of  $N \times M$  and  $K$  color channels, so CSI-based WiFi sensing can reuse the signal processing techniques and algorithms designed for computer vision tasks. The 4D CSI tensor provides additional information in the time domain. CSI can be processed, modeled, and trained in different domains for different WiFi sensing purposes, e.g., detection, recognition, and estimation.

## 2.5 Convolutional Neural Network

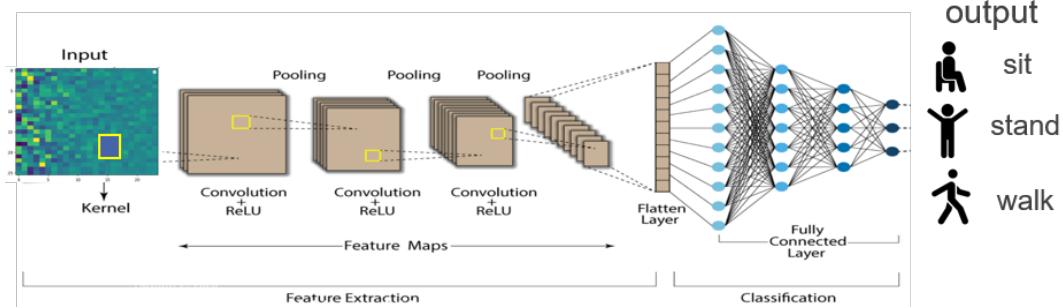


Figure 2.8: CNN architecture

CNN is a type of deep learning model for processing data that has a grid pattern, such as images, which is inspired by the organization of animal visual cortex and designed to automatically and adaptively learn spatial hierarchies of features, from low- to high-level patterns. CNN is a mathematical construct that is typically composed of three types of layers (or building blocks) Figure 2.8: convolution, pooling, and fully connected layers. The first two, convolution and pooling layers, perform feature extraction, whereas the third, a fully connected layer, maps the extracted features into final output, such as classification.

Within the scope of this thesis, CSI images serve as inputs to a Convolutional Neural Network (CNN). The CNN is tasked with performing two key operations: feature extraction and classification Figure 2.8. These operations collectively enable the network to effectively identify and recognize various human activities. This approach leverages the information contained in the CSI images to distinguish and categorize different activities, which constitutes a central aspect of the research conducted in this thesis.

A convolution layer plays a key role in CNN, which is composed of a stack of mathematical operations, such as convolution, a specialized type of linear operation. In digital images, pixel values are stored in a two-dimensional (2D) grid, i.e., an array of numbers, and a small grid of parameters called the kernel, an optimizable feature extractor, is applied at each image position, which makes CNNs highly efficient for image processing, since a feature may occur anywhere in the image. As one layer feeds its output into the next layer, extracted features can hierarchically and progressively become more complex. The process of optimizing parameters such as kernels is called training, which is performed so as to minimize the difference between outputs and ground truth labels through an optimization algorithm called backpropagation and gradient descent, among others.

## 2.6 Reinforcement Learning

### 2.6.1 Reinforcement Learning Definition and Architecture

RL is an Exploration and Exploitation learning method used to learn how to reach a goal, i.e., the final optimal state, based on performing actions by the agent. RL's main aim is

to give machines the ability to learn that exceeds the limitations of the supervised and unsupervised methods. The main elements in RL are (i) agent, (ii) environment, (iii) states, and, (iv) actions.

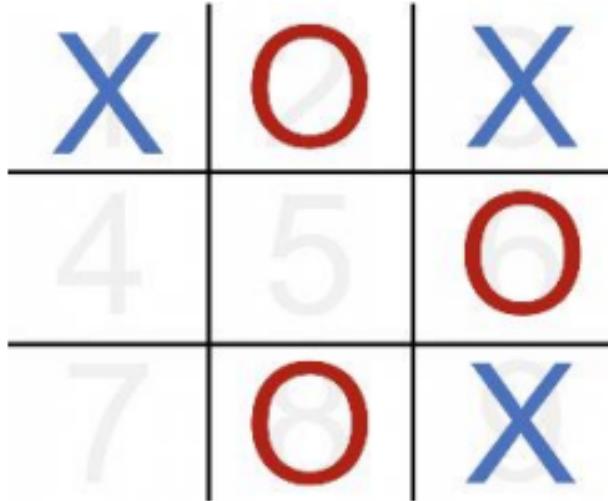


Figure 2.9: A situation(state)in the Tic-Tac-Toe Game

To explain RL elements, think about the situation of Figure 2.9 in the Tic-Tac-Toe game environment. It is agent-X turn to play. Agent-X has to decide where to place the X that will make him win the game or push him closer to its goal state, i.e., winning the game. Agent-X could take any position, i.e., move or action that will change the current game state to another. For instance, if Agent X played in position 5, he would win the game. However, Agent X would lose if he played in any other place, and Agent O would win the game. Moreover, there are some sub-elements of the RL system, as stated, which are:

### 2.6.2 Policy

A policy is when the agent is behaving in a specific situation based on previously gained knowledge. A policy could be in many forms, such as a table or a function. An agent will perform actions based on the generated policies. In general, the policy is considered the core of the RL process.

### 2.6.3 Reward

The agent gets a reward based on each action. The reward gives an immediate sense of the performed action if it was the right step, i.e., it made the agent closer to the goal or not. Thus, the agent seeks to maximize the reward. Besides, rewards could be assigned directly or delayed.

### 2.6.4 Value function

Value function: indicates the right steps in the long run. The value of a state is the total amount of rewards the agent gets to reach this state.

### 2.6.5 Model of the environment

The model will help to conclude how the environment will behave, e.g., the model could predict the next action and state given the current action and state. The RL system's

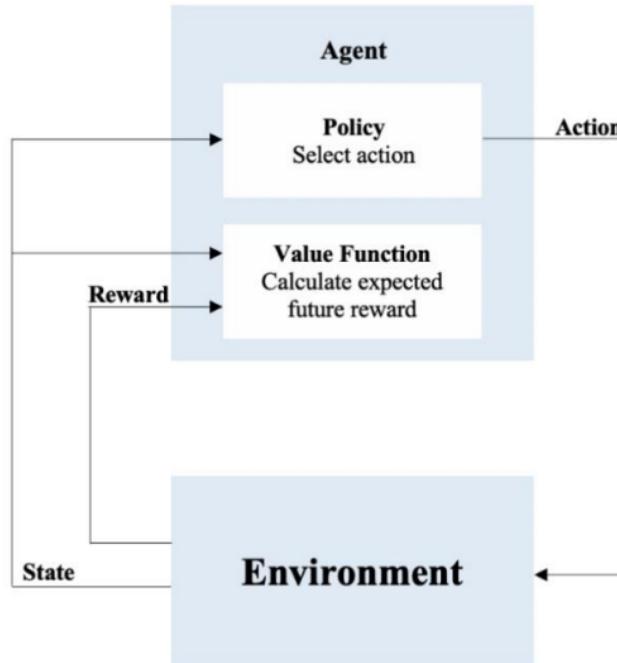


Figure 2.10: Reinforcement Learning Architecture

architecture is presented in Figure 2.10, which contains the agent, environment, action, and state. As there is no prior knowledge nor supervision provided to the agent, it will first perform a random action to discover the environment. Then the agent will be rewarded or punished based on the action it took. The agent will be rewarded if the performed action helped it to step closer to the goal. However, the agent will receive punishment when performing an action that makes it far from the goal.

## 2.7 Reinforcement Learning Categories

Many of the proposed RL algorithms require large amounts of training data before achieving acceptable performance. There are two kinds of RL algorithms, model-free (direct) and model-based (indirect) when continuous actions are available. Model-free RL algorithms learn a policy or value function without explicitly representing a model of the controlled system. In contrast, model-based approaches learn an explicit model of

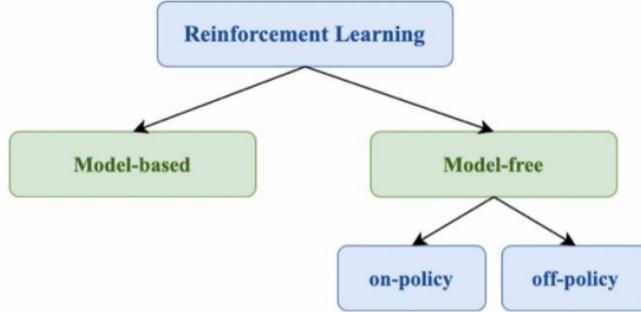


Figure 2.11: Caption

the system simultaneously with a value function and policy. Moreover, model-based RL prevents unexpected actions and states. Hence, model-free RL learns by exploring, while model-based RL learns by simulating.

## 2.8 Overview of Deep Reinforcement Learning

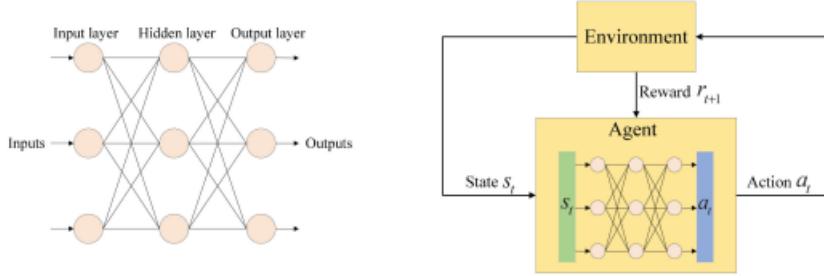


Figure 2.12: Deep Neural Network and Deep Reinforcement Learning

Machine learning can be categorized into three primary types based on feedback mechanisms: supervised learning, unsupervised learning, and reinforcement learning (RL). In supervised learning, instant feedback is available by comparing predictions with actual label data, aiding in model refinement. On the other hand, unsupervised learning lacks feedback as input data isn't labeled. RL, on the other hand, involves delayed feedback, as actions taken in the current state impact future states and actions, with the value of actions learned gradually over time.

Supervised and unsupervised learning are commonly employed for single-stage problems like regression, classification, clustering, and dimension reduction. In contrast, RL specializes in solving multi-stage decision problems. In the context of Sustainable Building Energy Management (SBEM), supervised learning is used to develop models for building thermal dynamics and reward functions. RL, based on these models, reduces interactions with the environment, enhancing sampling efficiency. Deep Reinforcement Learning (DRL) combines deep learning and RL, employing Deep Neural Networks (DNNs) to approximate optimal value functions or policies in RL, offering robust decision-making capabilities under uncertainty.

### 2.8.1 Q-Learning

Q-learning is a machine learning approach that enables a model to iteratively learn and improve over time by taking the correct action. Q-learning is a type of reinforcement learning. Q-learning models operate in an iterative process that involves multiple components working together to help train a model. The iterative process involves the agent learning by exploring the environment and updating the model as the exploration continues. The multiple components of Q-learning include the following:

- **Agents** - The agent is the entity that acts and operates within an environment.
- **States** - The state is a variable that identifies the current position in the environment of an agent.
- **Actions** - The action is the agent's operation when it is in a specific state.
- **Rewards** - A foundational concept within reinforcement learning is the concept of providing either a positive or negative response to the agent's actions.

- **Episodes** - An episode is when an agent can no longer take a new action and ends up terminating.
- **Q-values** - The Q-value is the metric used to measure an action at a particular state.

### 2.8.2 Method to determine the Q-value

Bellman's equation - Mathematician Richard Bellman invented this equation in 1957 as a recursive formula for optimal decision-making. In the q-learning context, Bellman's equation Equation 2.4 is used to help calculate the value of a given state and assess its relative position. The state with the highest value is considered the optimal state.

Q-learning models work through Exploration and Exploitation experiences to learn the optimal behavior for a task. The Q-learning process involves modeling optimal behavior by learning an optimal action-value function or q-function. This function represents the optimal long-term value of action  $a$  in state  $s$  and subsequently follows optimal behavior in every subsequent state.

#### Bellman's equation

$$Q(s, a) = Q(s, a) + \alpha \cdot (r + \gamma \cdot \max(Q(s', a')) - Q(s, a)) \quad (2.4)$$

- $Q(s, a)$  represents the expected reward for taking action  $a$  in state  $s$ .
- The actual reward received for that action is referenced by  $r$  while  $s'$  refers to the next state.
- The learning rate is  $\alpha$  and  $\gamma$  is the discount factor.
- The highest expected reward for all possible actions  $a'$  in state  $s'$  is represented by  $\max(Q(s', a'))$ .

### 2.8.3 Q-Table

The Q-table includes columns and rows with lists of rewards for the best actions of each state in a specific environment. A Q-table helps an agent understand what actions are likely to lead to positive outcomes in different situations. The table rows represent different situations the agent might encounter, and the columns represent the actions it can take. As the agent interacts with the environment and receives feedback in the form of rewards or penalties, the values in the Q-table are updated to reflect what the model has learned. Reinforcement learning aims to gradually improve performance through the Q-table to help choose actions. With more feedback, the Q-table becomes more accurate so the agent can make better decisions and achieve optimal results.

### 2.8.3.1 What is the Q-learning algorithm process?

The Q-learning algorithm process is an interactive method where the agent learns by exploring the environment and updating the Q-table based on the rewards received.

- **Q-table initialization** - The first step is to create the Q-table as a place to track each action in each state and the associated progress.
- **Observation** - The agent needs to observe the current state of the environment.
- **Action** - The agent chooses to act in the environment. Upon completion of the action, the model observes if the action is beneficial to the environment.
- **Update** - After the action has been taken, it's time to update the Q-table with the results.
- **Repeat steps 2-4** until the model reaches a termination state for a desired objective.

### 2.8.3.2 advantages of Q-learning

The Q-learning approach to reinforcement learning can potentially be advantageous for several reasons, including the following:

- **Model-free** - The model-free approach is the foundation of Q-learning and one of the biggest potential advantages for some uses. Rather than requiring prior knowledge about an environment, the Q-learning agent can learn about the environment as it trains. The model-free approach is particularly beneficial for scenarios where the underlying dynamics of an environment are difficult to model or completely unknown.
- **Off-policy optimization** - The model can optimize to get the best possible result without being strictly tethered to a policy that might not enable the same degree of optimization.
- **Flexibility** - The model-free, off-policy approach enables Q-learning flexibility to work across a variety of problems and environments.
- **Offline training** - A Q-learning model can be deployed on pre-collected, offline data sets.

## 2.9 Deep Q - Network

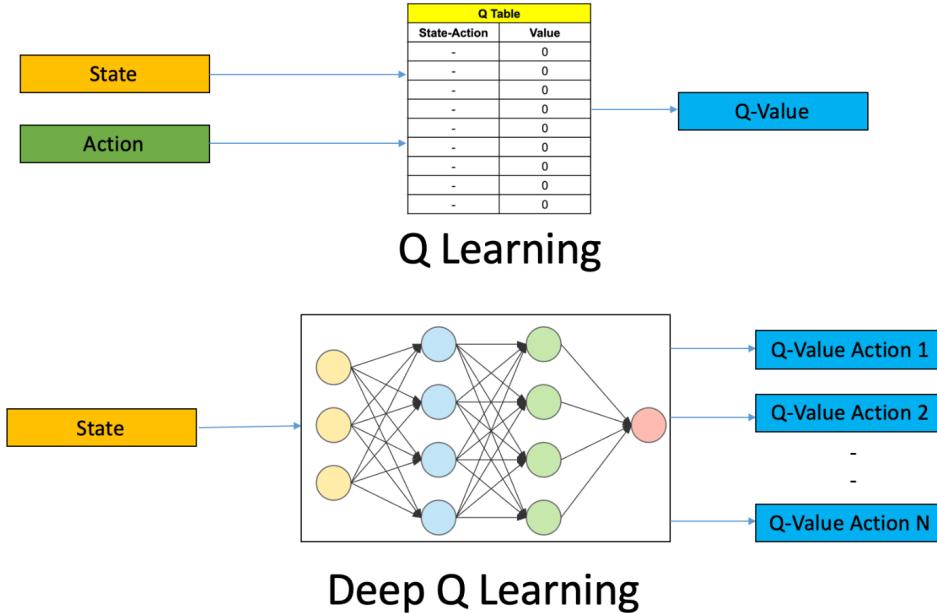


Figure 2.13: Overview of Q-Learning and Deep Q-learning Network

DQN's use of neural networks allows it to generalize Q-values to unseen states. The neural network can provide estimates for Q-values even in states that were not explicitly visited during training, making it more adaptable to new situations.

So now we use a Deep Neural Network that gets the state as input and produces different Q values for each action. Then again we choose the action with the highest Q-value. The learning process is still the same with the iterative update approach, but instead of updating the Q-Table, we update the weights in the neural network to improve the outputs. It works on the Epsilon-Greedy exploration strategy.

# 3 State of the Art

CSI Sensing has been a very active research field in recent years. Two very relevant papers that background the present research, will be shortly presented in this chapter.

## 3.1 Location- and Person-Independent Activity Recognition with WiFi, Deep Neural Networks, and Reinforcement Learning

### 3.1.1 Introduction

The paper [23] discusses a deep learning approach for recognizing human activities using WiFi Channel State Information (CSI). The proposed design comprises three neural networks: a 2D Convolutional Neural Network for recognition, a 1D CNN for temporal dependency, and a reinforcement learning agent for architecture optimization. It achieves 97 % average accuracy for unseen devices and persons in a lab environment and 80 % to 83 % accuracy on public datasets. Importantly, this design minimizes the need for human effort in labeling, feature engineering, and tuning parameters. This article

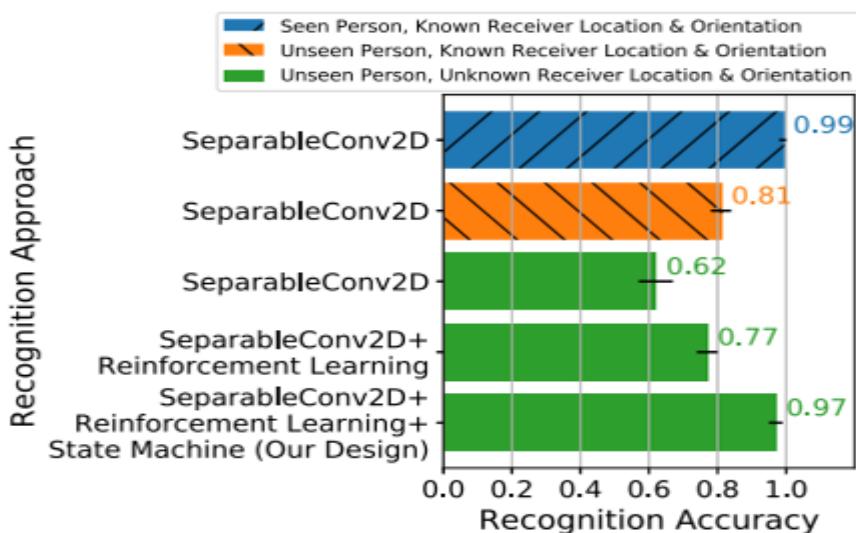


Figure 3.1: Acuuracy comparison of different Deep Learning solutions

introduces a novel deep learning approach for robust WiFi-based activity recognition, comprising three neural networks: a 2D CNN for recognition, a 1D CNN for modeling temporal dependencies, and an RNN with LSTM for neural architecture optimization. The

proposed design has three key components: Recognition Algorithm: Utilizes a 2D CNN to learn location- and person-independent features from various perspectives of 4D CSI tensors (in time, spatial, and frequency domains). State Machine: Employs a 1D CNN to capture temporal dependencies from previous classification results, enhancing recognition performance for both static and transition activities. Neural Architecture Search: Utilizes an RNN with LSTM for optimizing the neural architecture of the recognition algorithm through reinforcement learning.

### 3.1.2 Setup

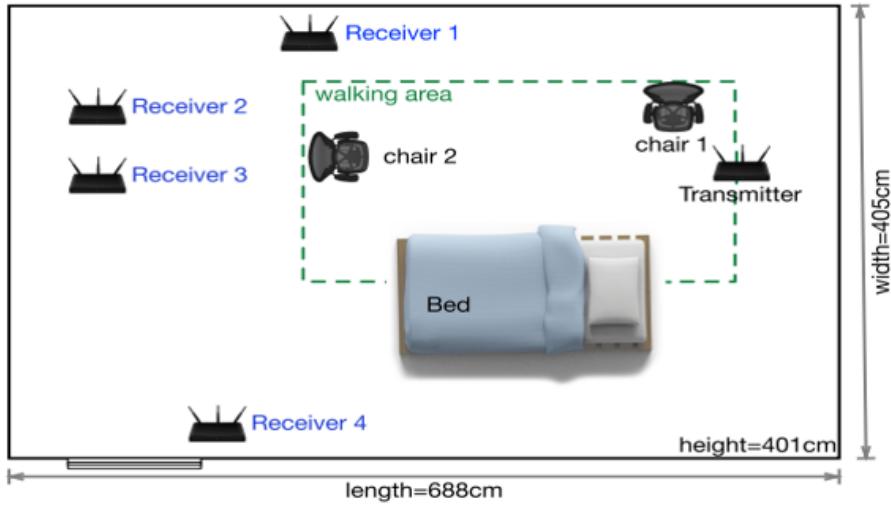


Figure 3.2: Experiment setup. There are 4 WiFi receivers placed at different locations with different heights and antenna orientations

The proposed design Figure 3.2 is tested using real-world CSI measurements, involving 5 activities (sitting, standing, sit-down, stand-up, and walking) performed by 7 individuals. There are 4 WiFi receivers positioned at various locations and antenna orientations. Participants can sit or stand at two locations with different facing directions and walk in a confined area. Through reinforcement learning, the recognition accuracy improves from 62 % to 77 % for unknown receiver locations, orientations, and unseen individuals. When both reinforcement learning and the state machine are used, the accuracy further increases to 97 %. The proposed design is also evaluated with two public datasets, S.Yousefi-2017 and FallDeFi, achieving accuracies of 80 % and 83 %, respectively.

### 3.1.3 Design

The proposed design for WiFi-based activity recognition is outlined in Figure 3.3 and includes pre-processing, a recognition algorithm, a state machine, and a neural architecture search. Data collection involves participants following audio scripts while collecting CSI measurements with WiFi receivers. Motion-tracking devices and audio scripts are used for offline training but are not required during the inference stage, where only CSI measurements are used.

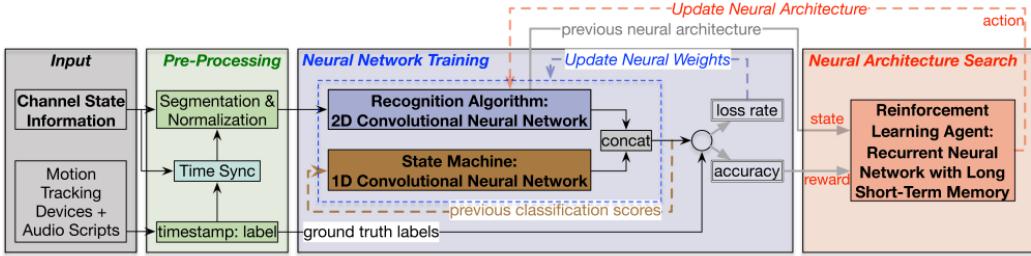


Figure 3.3: The training process of the proposed design

**The design consists of four key components:**

- Data Pre-processing: CSI matrices are synchronized, segmented by ground truth labels, and normalized based on training data.
- Recognition Algorithm: Utilizes a 2D or 3D CNN to learn features from various perspectives of 4D CSI tensors, focusing on location- and person-independent features within one CSI segment.
- State Machine: Learns temporal dependencies across multiple CSI segments using a 1D CNN.
- Neural Architecture Search: Optimizes the neural architecture of the recognition algorithm through a reinforcement learning agent, an RNN with LSTM.

This combination enables robust WiFi-based activity recognition that is location- and person-independent. The recognition algorithm focuses on extracting relevant features, the state machine models temporal dependencies, and the reinforcement learning agent maximizes accuracy. The proposed design performs well in scenarios with unknown WiFi device locations, orientations, and unseen individuals. Importantly, it minimizes the need for human efforts in various aspects of the recognition process.

## 3.2 WiAgent: Link Selection for CSI-Based Activity Recognition in Densely Deployed Wi-Fi Environments

### 3.2.1 Introduction

The work presented in this paper addresses the challenge of WiFi-based Human Activity Recognition (HAR) in environments where WiFi networks are densely deployed. While traditional WiFi CSI-based HAR systems typically use a single WiFi transmitter and one or more WiFi receivers to capture activity-related data, they often overlook the interactions and communication between multiple WiFi devices in real-world scenarios. In response to this limitation, the paper introduces a novel WiFi link selection model, named WiAgent. This model employs a continuous state decision-making process in which the Channel State Information (CSI) is integrated as part of the system state. WiAgent takes actions to select specific WiFi links based on the current state and continuously updates the state to inform subsequent decisions. The results of extensive experiments indicate that WiAgent outperforms other existing solutions, especially in environments

with multiple Wi-Fi transmitters. It demonstrates the effectiveness of the proposed model in improving the accuracy and reliability of Wi-Fi-based Human Activity Recognition, particularly in complex and densely deployed Wi-Fi networks.

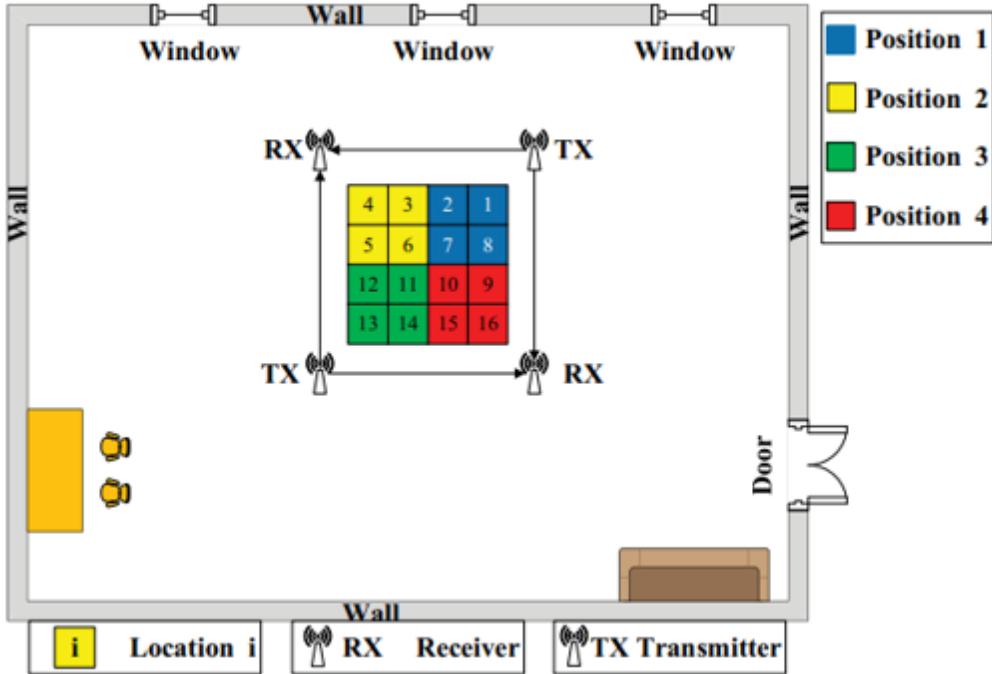


Figure 3.4: Experimental Setup Configuration

### 3.2.2 Experimental setup

Employed two mini-PCs as transmitters and two mini-PCs as receivers, each equipped with Intel Wireless Link 5300 Network Interface Cards (NICs). We utilized the CSI tool to capture Channel State Information (CSI) data. To facilitate multi-device collaboration, both of the two receivers were configured to simultaneously receive packets from the two transmitters, effectively creating a total of four distinct Wi-Fi links. The experimental area was divided into 16 locations, as depicted in Figure 3.4, which were defined by the boundaries of these Wi-Fi links. The target individual was engaged in five different daily activities: "walk," "run," "stand," "sit," and "bend." We gathered a dataset consisting of 2,749 real CSI samples within a room measuring 8 meters by 7 meters. Of these samples, 1,949 were allocated as training sets, and the remaining 800 samples were reserved for testing. In the test sets, there were 50 samples collected at each of the 16 locations. As input data for the reinforcement learning agent, the normalized CSI streams were processed through an LSTM-based context network. This setup enabled the agent to effectively learn and make decisions based on the received CSI data in the context of human activity recognition.

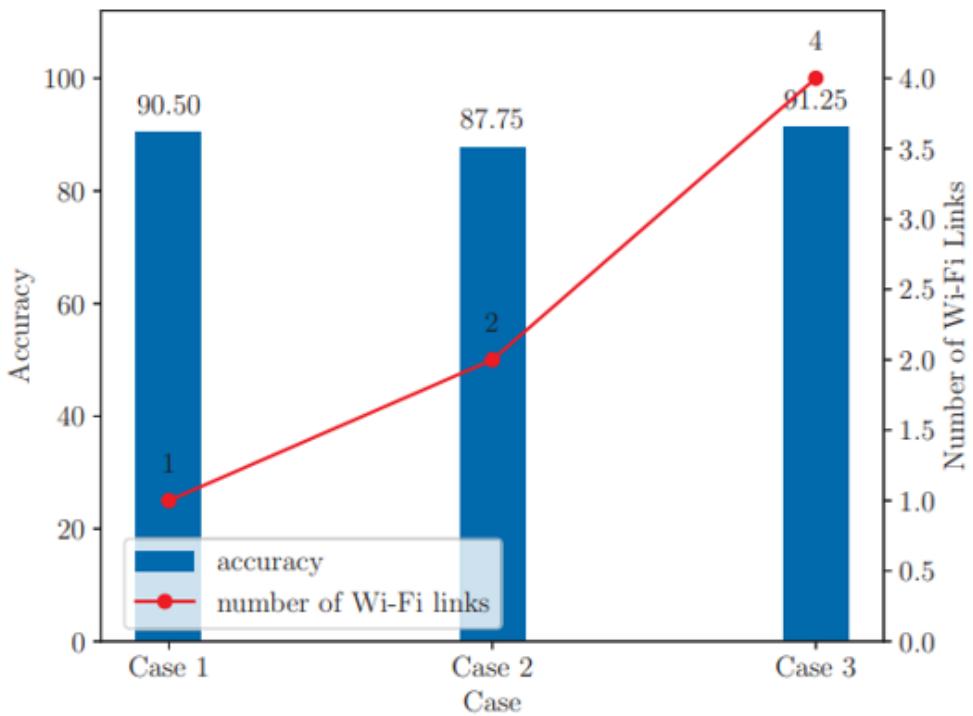


Figure 3.5: The overall accuracy of the three cases and the numbers of Wi-Fi links used in these cases

### 3.2.3 Evaluation

In this study, a comparison of three cases of Human Activity Recognition (HAR) using Wi-Fi data:

- Case 1: WiAgent intelligently selects a Wi-Fi link for training.
- Case 2: Traditional use of two orthogonal Wi-Fi links.
- Case 3: Utilizes all available Wi-Fi links.

The experiment results show that Case 1 (WiAgent selection) and Case 3 (using all links) outperform Case 2. Case 1, with intelligent link selection, has excellent performance and reduces computational complexity. This approach is highly practical for large-scale Wi-Fi deployments, enhancing both recognition accuracy and efficiency.

### **3.3 Evaluation of State of Art methods**

In the first state-of-the-art [23], the data collection setup includes one transmitter and four receivers in a relatively small area of interest. Seven individuals perform five different activities with various orientations. The machine learning model employed in this scenario combines CNN and LSTM architecture for neural architecture search. In the second scenario [33], the area of interest is divided into 16 locations, and two transmitters and three receivers are used. Only one person participates in performing five different activities. For machine learning classification and neural architecture search, the model utilizes a combination of CNN and LSTM.

The thesis explores two distinct approaches: Convolutional Neural Networks (CNN) and Reinforcement Learning, particularly using Q-learning. This research represents a novel and emerging area in device-free Human Activity Recognition (HAR) by incorporating a deep Q-learning network for image classification. The study also delves into the realm of transfer learning, investigating its application and the results obtained.

# 4 Framework of the Implemented Human Activity Recognition

The ?? which is presented below, illustrates the framework for Human Activity Recognition using Wi-Fi and a Machine Learning model for building automation. The structure of this framework is as follows:

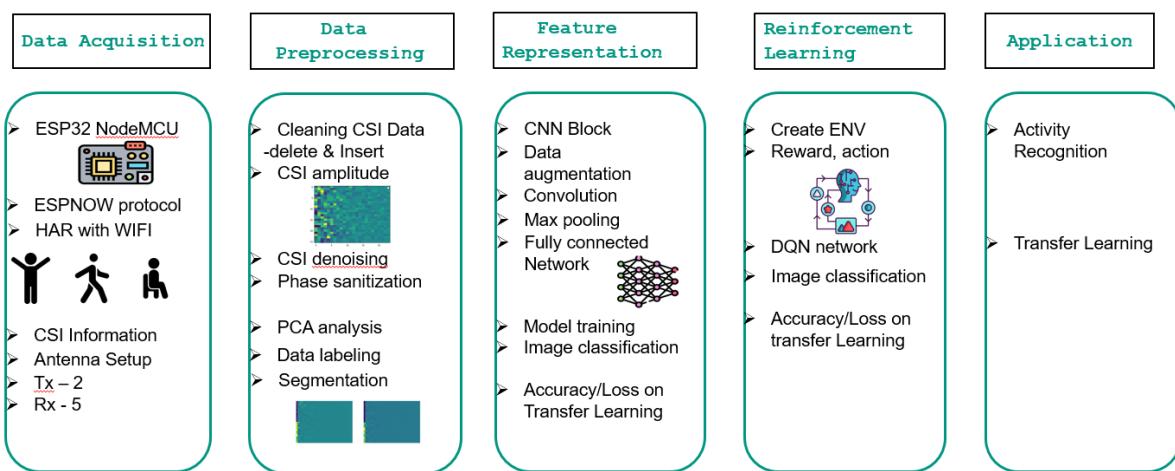


Figure 4.1: Framework of Overall Human Activity Recognition using Machine Learning

## 4.1 Data Acquisition

For data acquisition, we employ the ESP32, a low-cost microcontroller that operates on the 802.11n wireless network standard. ESP32 is utilized for data acquisition purposes, and data transmission is accomplished using the ESPNOW protocol. ESPNOW protocol operates based on MAC addresses and employs callback functions to ensure the reliability of data transmission.

The Channel State Information (CSI) is acquired from the sensors, encompassing amplitude attenuation and phase shift related to human activity and the surrounding environment. We employ a total of 7 sensors, consisting of 2 transmitters and 5 receivers, configured in a 2:5 ratio. Data acquisition is carried out by four individuals, each engaging in four different activities, in two distinct locations with varying antenna orientations.

## 4.2 Data Pre-processing

After acquiring the desired Channel State Information (CSI) data, the next step involves data preprocessing and data cleaning. The CSI data is initially stored in CSV file format, and several operations are performed to enhance data quality and ensure a seamless acquisition process. These operations include the insertion and deletion of data lines that may be considered as misbehaving or erroneous. Furthermore, the CSI data undergoes preprocessing procedures such as CSI denoising, phase sanitization, and the application of a Butterworth filter to refine its quality. The cleaned CSI data is then transformed into two distinct representations: the Amplitude Image and the Phase Image of the CSI data.

To manage the data effectively, Principal Component Analysis (PCA) is employed for dimensionality reduction, resulting in an image size of 24 \* 24 pixels. This reduction in dimensionality simplifies the data while retaining its essential characteristics. Subsequently, the processed and dimensionally reduced data is subjected to Data Labeling and Segmentation. These steps are integral in preparing the CSI data for Feature Extraction and classification within the Machine Learning model, facilitating accurate analysis and recognition of human activity patterns.

## 4.3 Feature Extraction Algorithm

Feature extraction in this study was performed using a Convolutional Neural Network (CNN). The CSI input image underwent a series of processing steps through multiple convolution layers, max-pooling layers, and dropout layers. These layers were instrumental in identifying and emphasizing essential patterns within the CSI data. The classification task was carried out by utilizing two fully connected dense layers. These layers played a pivotal role in distinguishing and categorizing various human activities based on the features extracted earlier. To enhance the performance of the classification task, various data augmentation methods were employed, resulting in improved accuracy and reduced validation loss.

At the final stage of the network, a Softmax classifier was employed to make conclusive classifications. This classifier provided a comprehensive and accurate analysis of the recognized human activities, thereby contributing to the overall success of the classification process.

## 4.4 Reinforcement Learning

The reinforcement learning model begins with the creation of a custom environment using the OpenAI Gym framework. This environment forms the foundation for the learning process. In this learning process, several key components are defined:

- Reward Function: The reward function is established to provide feedback to the agent based on its actions. It plays a critical role in guiding the agent's learning process.

- Agent: The agent, often implemented as a reinforcement learning algorithm, is responsible for making decisions and taking actions within the environment.
- Environment: This is the simulated world where the agent interacts and learns. In this context, the environment models the problem of human activity recognition.
- Action: Actions refer to the choices or decisions that the agent can make within the environment. These actions are determined by the problem being addressed.
- Q-Values: Q-values represent the expected cumulative reward associated with taking a specific action in a given state. They are used to guide the agent's decision-making process.

Additionally, hyperparameters are adjusted to optimize the learning process and achieve improved results in the classification of human activities. The agent employs a combination of exploration and exploitation strategies to interact with the environment. Exploration involves trying new actions to discover potentially better strategies, while exploitation involves selecting actions that have previously yielded positive rewards. This balance between exploration and exploitation is crucial for the agent to learn and improve its decision-making over time. By iteratively adapting its actions based on the rewards received in different states, the reinforcement learning model aims to enhance its classification results for human activity recognition.

## 4.5 Application

The fusion of Human Activity Recognition using Reinforcement Learning and Convolutional Neural Networks presents exciting opportunities for various applications, including transfer learning for building automation. In this context, activity identification is achieved by focusing on achieving the best accuracy and minimizing validation loss during the classification of CSI images. By prioritizing accuracy and minimizing loss, the system becomes more reliable in recognizing and categorizing human activities. This approach has the potential to revolutionize building automation, as it can enable the system to adapt and respond effectively to various human activities within a building environment. It opens the door to numerous possibilities, such as optimizing energy consumption, enhancing security, and improving the overall efficiency and user experience in smart buildings. In summary, the integration of these advanced technologies not only advances the field of Human Activity Recognition but also holds great promise for applications beyond it, particularly in the realm of building automation.

# 5 Implementation

The Wi-Fi Sensing procedure from the experimental assembly will be described in this chapter.

## 5.1 Data Acquisition

Data acquisition steps are as follows:

- Configuration of the Setup
- CSI transmission procedure
- Training and Testing Environment
- Data Gathering Process

### 5.1.1 Configuration of Setup

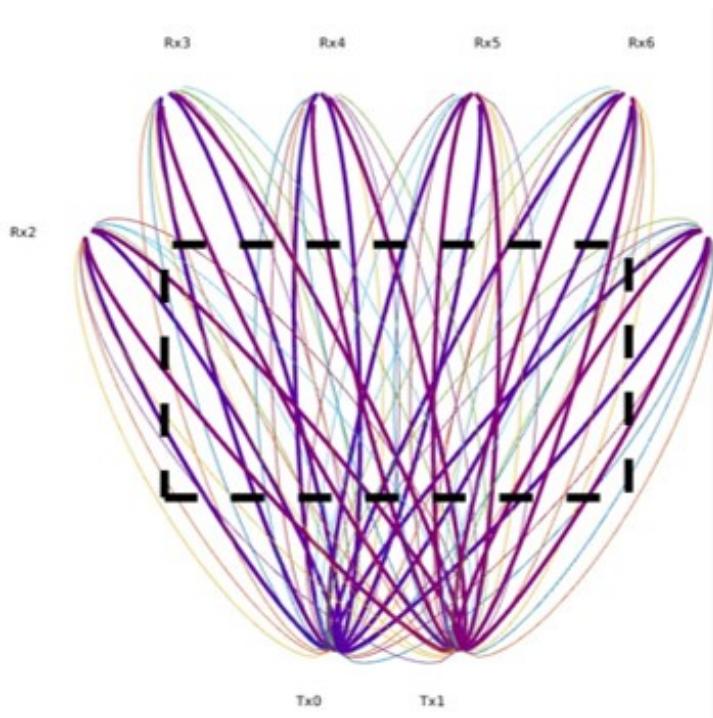


Figure 5.1: Configuration setup of the ESP32 sensors

In the sensor configuration setup, a total of 8 sensors have been deployed. This setup comprises two ESP32 transmitters and six ESP32 receivers, as depicted in Figure 5.1. The configuration is designed to cover the entire area of interest, ensuring comprehensive coverage during both the training and testing phases. Specifically, the two transmitters are positioned on one side, while the six receivers are located on the other side. The sensor configuration involves the use of 10 links, each equipped with an omnidirectional antenna featuring a gain of 2dBi in the 2.4 GHz band. Omnidirectional antennas are known for their ability to radiate or intercept radio-frequency (RF) electromagnetic fields equally well in all horizontal directions within a flat, two-dimensional (2D) geometric plane. This characteristic results in a 360° donut-shaped radiation pattern, providing wide signal coverage suitable for various indoor and outdoor wireless applications. Regarding the 802.11n physical standard, it comprises a total of 56 subcarriers, with 52 of them being considered useful samples when utilizing a 40 MHz bandwidth. The area of interest in this setup encompasses the entire near-field region, often referred to as the Fresnel region. During the setup, all antennas were positioned in both vertical and horizontal orientations to capture data under different antenna orientations and readings.

### 5.1.2 Channel State Information Transmission workflow

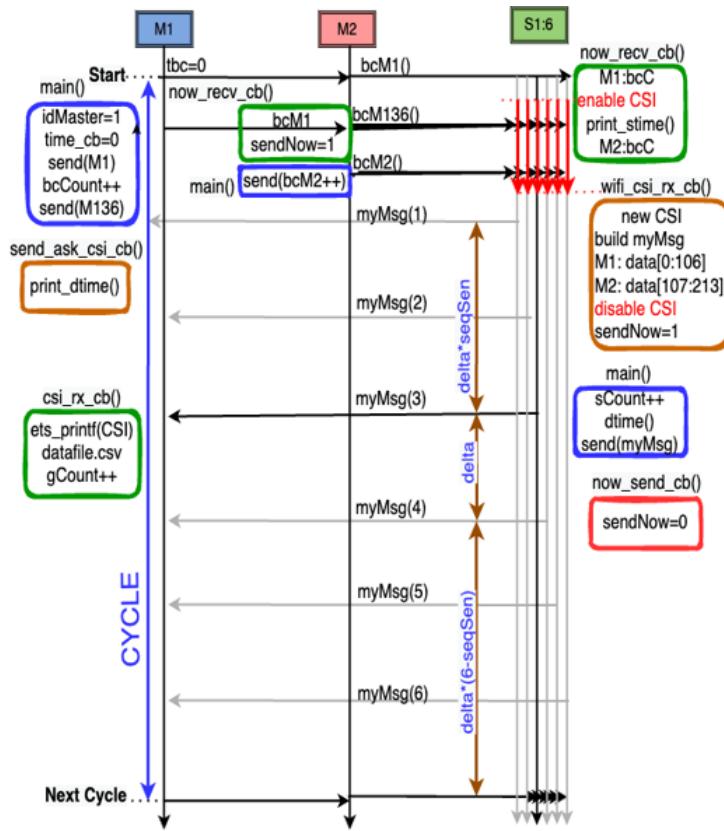


Figure 5.2: Workflow of CSI transmission

The transmission and collection of Channel State Information (CSI) data are facilitated through the use of the ESPNOW connectionless protocol. This protocol operates based on

the MAC addresses of the devices involved, eliminating the need for a Wi-Fi connection establishment. Once devices are paired, the connection between them is secure and peer-to-peer, obviating the requirement for a handshake mechanism. CSI, in this context, represents how wireless signals propagate from the transmitter to the receiver at specific carrier frequencies across multiple paths using the OFDM-MIMO technique. The entire operation of CSI transmission is illustrated in Figure 5.2 Here's a step-by-step breakdown of the CSI transmission process:

- Transmitter M1 initiates the broadcast signal and sends it to all sensors.
- All receivers receive the broadcast from M1 and enable the CSI.
- Transmitter M2 sends the broadcast signal to all the receivers.
- As soon as receivers receive the broadcast from M2, they disable the CSI and send a signal (`sendNow=1`) back to the M1 transmitter.
- M1 begins collecting the CSI data from all the receivers.
- The transmission employs Time Division Multiple Access (TDMA) to allocate specific acquisition times for each device.
- The receiver signal sends (`sendNow=0`) after successfully transmitting the CSI message and data to M1.

The complete cycle, which includes the collection of CSI data from all the sensors, requires approximately 750 milliseconds. In practice, only 350 milliseconds are typically needed for one cycle, while an additional 400 milliseconds are allocated as buffer time to account for potential Jamming signals or data transmission collisions.

### 5.1.3 Training and Testing Environment



Figure 5.3: Training and Testing Environment of CSI data Acquisition

Two different training and testing environments are employed for data acquisition, as illustrated in Figure 5.3. The primary objective of using these two environments is to ensure the presence of shared features such as windows, shelves, benches, chairs, and other elements. This similarity in features holds significant importance in the context of

data acquisition for transfer learning applications in building automation. The testing environment replicates the configuration of the training environment, encompassing these common elements. However, it is important to note that the testing environment is of a smaller scale compared to the training environment. This approach, which involves utilizing similar features in both environments, facilitates effective and accurate transfer learning, allowing the model to confidently apply the knowledge gained during training to the testing environment, despite the differences in scale.

#### 5.1.4 Gathering process

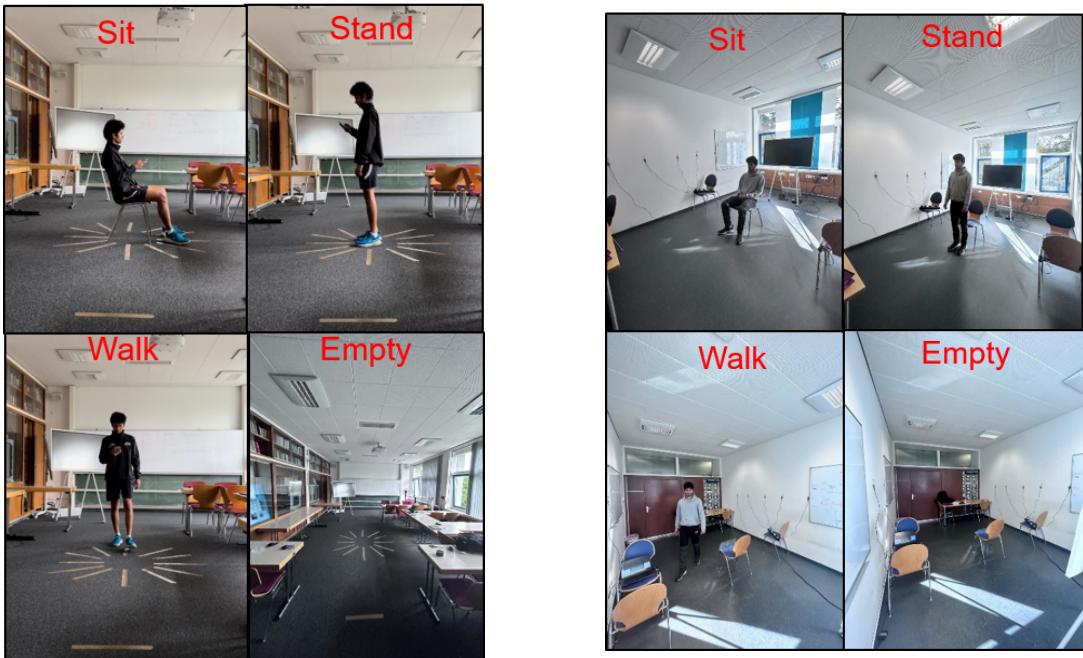


Figure 5.4: Person 1 performing activities in training and Testing Environment

Following the setup and configuration implementation, the data-gathering process commences, involving four distinct activities: sitting, standing, walking, and an empty state. These activities are performed by four different individuals, each engaging in a specific activity for a duration of 30 seconds Figure 5.4. This sequence continues, resulting in a total experimental duration of two minutes. The experiments are conducted with two different antenna orientations, specifically vertical and horizontal, for both the training and testing environments. The sequence of activities typically starts with an individual sitting, transitions to standing after 30 seconds, proceeds to walking for the next 30 seconds, and culminates with the person leaving the area of interest for the empty room activity, which lasts for the final 30 seconds of the two-minute experiment.

As depicted in Figure 5.5, a cycle of CSI data acquisition is carried out by each individual, illustrating the process for person 1. It's important to note that during the data acquisition phase, sensor number 8 was not operational and, consequently, was excluded from the process. As a result, data was collected from only five receivers. The data acquisition from all sensors was completed in less than 270 milliseconds for each cycle, both in the training and testing environments. Notably, the same individuals participated in data gathering

1	CSI_DATA	3	1790.961	1798.517	17.678	1974.01	248.3	19.82	[7,-20]
2	CSI_DATA	4	1802.608	1810.164	58.179	2032.788	306.817	77.869	[14,-1]
3	CSI_DATA	5	1787.181	1794.738	108.352	2095.982	370.11	141.63	[0,16,0]
4	CSI_DATA	6	1787.885	1795.441	158.338	2152.666	426.695	197.747	[3,6,8,6]
5	CSI_DATA	7	1790.696	1798.253	213.75	2211.172	485.201	256.253	[5,2,5,1]
1	CSI_DATA	3	239.914	246.939	16.946	413.862	225.428	18.559	[4,-14]
2	CSI_DATA	4	231.761	238.786	64.467	463.969	275.535	68.666	[5,21]
3	CSI_DATA	5	223.726	230.751	114.639	527.77	339.336	132.467	[4,-5,3]
4	CSI_DATA	6	230.946	237.97	164.512	592.185	403.751	196.882	[3,-15]
5	CSI_DATA	7	238.842	245.867	212.69	661.189	472.755	265.886	[10,10]

Figure 5.5: 1 cycle of CSI data

for both environments, ensuring consistency in the data collection process. These four participants exhibit a diverse range of heights, spanning from 175 to 190 centimeters, and weights, ranging from 60 to 85 kilograms. Before the data undergoes preprocessing and cleaning procedures, each experiment yields a total of 1,500 CSI lines. These lines represent the raw data acquired during the experiments and will be subject to further processing to ensure data quality and relevance for subsequent analysis.

## 5.2 Data Preprocessing

Data Preprocessing steps are as follows:

- Data Cleaning
- CSI Amplitude and Phase Mapping
- PCA Transformation

### 5.2.1 Data Cleaning

Following the acquisition of CSI data for human activities, the subsequent step involves data cleaning. CSI data typically contains unwanted noise and sequences of misbehaving lines, which need to be addressed to ensure the data's quality and suitability for further processing. To achieve this, an insert and delete lines mechanism is employed. This mechanism inserts missing lines in place of the ones that were deemed problematic and deletes redundant lines. Although this process can be time-consuming, it is essential to maintain the integrity of the sensor sequence and corresponding time data. In this data-cleaning approach, it is crucial to minimize the impact of data editing on the overall dataset. Therefore, the maximum acceptable edit percentage for the conducted data acquisition is set at 7%. To identify and handle outliers within each data line, a median filter is applied.

For addressing unwanted noise, a Butterworth low-pass filter is utilized. This type of filter allows low-frequency components of the signal to pass through while attenuating or blocking higher-frequency components. The Butterworth filter is configured as follows:

Sampling period: 1.0 Sampling rate: 10 Hz Cutoff frequency: 2 Hz Nyquist frequency:  $0.5 * fs$  (where  $fs$  is the sampling frequency) Total number of samples ( $n$ ) is calculated as the integer part of  $T * fs$ , where  $T$  represents the time period.

Additionally, to ensure that phase values remain within a specific range, phase sanitization is performed by adding or subtracting  $2\pi$ . This step is crucial for later processes that involve creating CSI amplitude and phase images.

Listing 5.1: Phase Sanitization algorithm

```

1 if oAbs == False:
2     for i in range(l1):
3         for c in range(co-1):
4             delta = ab[i+li][c+1]-ab[i+li][c]
5             if delta > 3.1415926:
6                 ab[i+li][c+1]=ab[i+li][c]+delta-2*3.1415926
7             if delta < -3.1415926:
8                 ab[i+li][c+1]=ab[i+li][c]+delta+2*3.1415926

```

These data-cleaning procedures are essential to ensure the reliability and accuracy of the CSI data for subsequent analysis and modeling. As Figure 5.6 shows the Border time

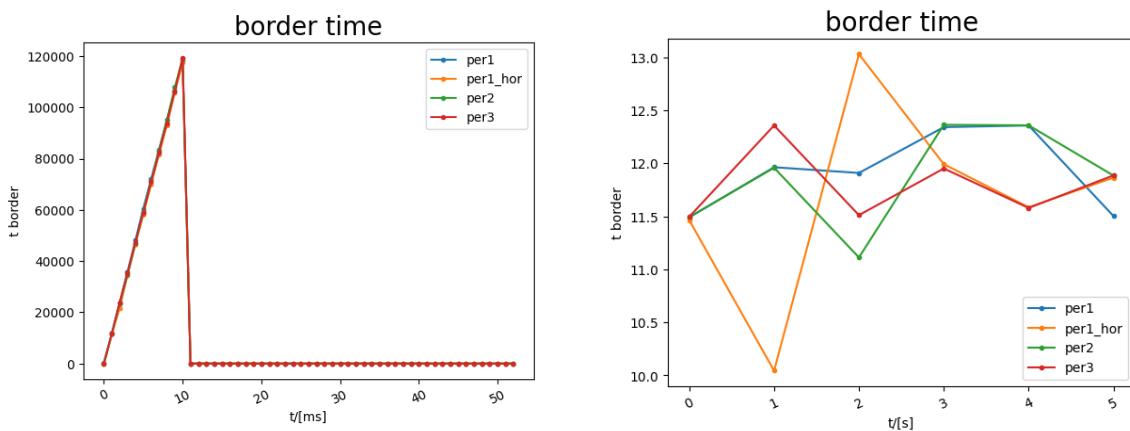


Figure 5.6: Border Time

between consecutive data points and stores them in the "difborde" array. The border time between consecutive CSI (Channel State Information) points, also known as the time interval between two consecutive CSI measurements, is a critical parameter in the context of Wi-Fi-based human activity recognition (HAR) and wireless signal analysis. This time interval determines how frequently CSI measurements are taken. A shorter time interval results in more frequent measurements and higher temporal resolution, which can be useful for capturing fast and dynamic activities. However, it may also increase the computational load and power consumption. Conversely, a longer time interval may save computational resources but could miss fine-grained details of human activities.

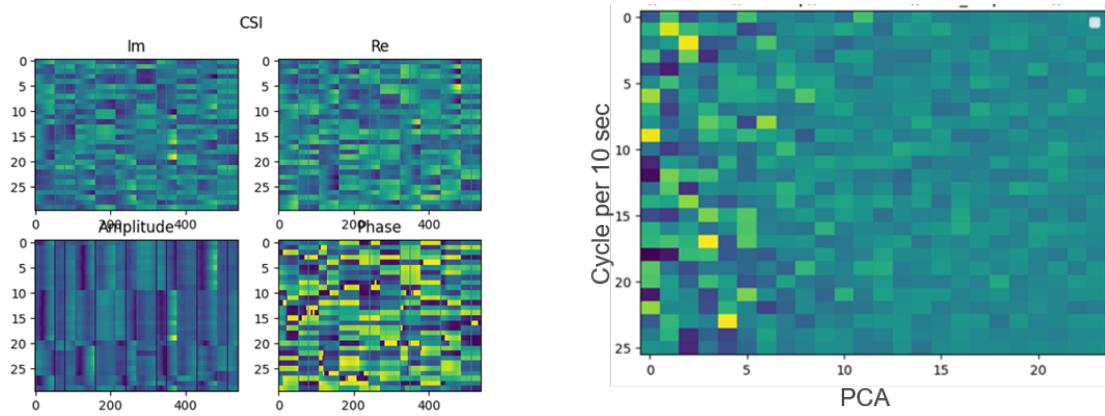


Figure 5.7: CSI Amplitude and Phase mapping, PCA Image

### 5.2.2 Amplitude and Phase Mapping

From each CSI captured by the ESP32, we have selected 54 carriers, resulting in a total of  $54 \times 10 = 540$  samples per acquisition cycle. By excluding the pilot ("guard") frequencies, we are left with 520 useful (Im, Re) pairs of data. Each acquisition cycle produces 5 CSI lines in the .csv file, with each line sent by a sensor corresponding to the CSI received from both M1 and M2. With an image period of 10 seconds, each .csv file contains approximately 1500 lines of data. Within each 10-second period, 150 lines are utilized for processing. Given that data from the 5 sensors (M1, M2) are concatenated to achieve a common broadcast and sampling time, images are constructed with 24 lines each. The Real and Imaginary parts of the data are separated into two distinct images, as illustrated in Figure 5.7. Subsequently, they are mapped to create Amplitude and Phase images, which are then combined into a single image consisting of 24 lines. To compress

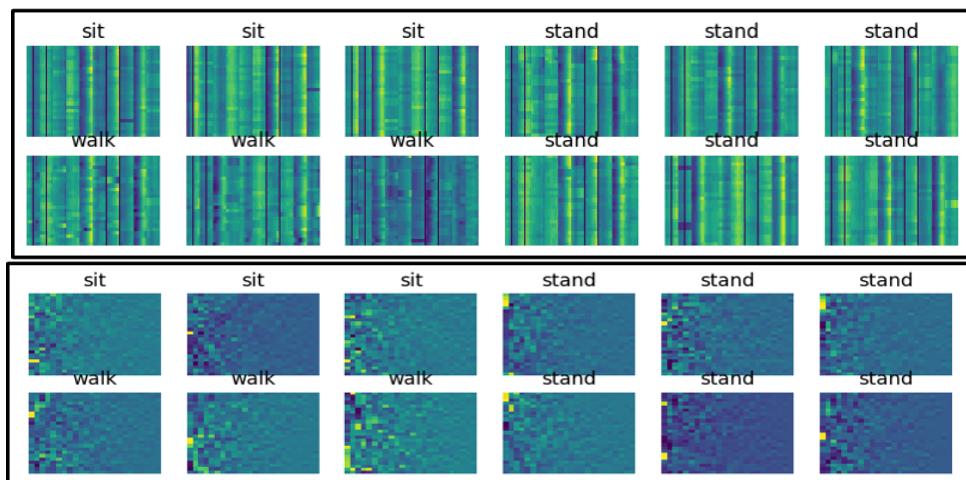


Figure 5.8: Individual Activity with Amplitude and Phase CSI images

and reduce redundancy in the information while retaining its essential characteristics, a Principal Component Analysis (PCA) routine is applied. This process involves the use of 24 components to transform the data into a new coordinate system known as the principal component space. The objective is to reduce the dimensionality of the data while preserving as much of the original data's variance as possible. This results in the

creation of standard 24 x 24 images that are employed in the classification task. Utilizing 24 Principal Component Analysis (PCA) components, the Imaginary and Real parts of the data are separated and mapped to create Amplitude and Phase images. This results in a total of 25 cycles within a 10-second timeframe. As depicted in Figure 5.8, the activities

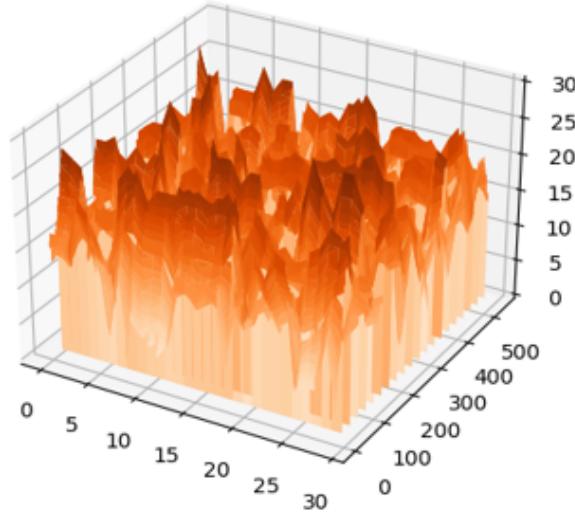


Figure 5.9: Surface Visualisation of CSI Image and Data

are visually represented using CSI Amplitude Images and Phase Images. In the context of a 30-second duration for each activity, a total of 75 cycles is generated per 10 seconds, allowing for a comprehensive visualization of the activities. Figure 5.9 provides a surface CSI visualization, where the x-axis represents the number of useful samples, the y-axis represents the number of cycles per 10 seconds, and the z-axis represents the amplitude of the image. This visualization offers a three-dimensional representation of the CSI data, enhancing the understanding of the data's characteristics and variations.

In Figure 5.10 , all carriers from M1 to S3 are visualized for a complete acquisition cycle involving the activities of sitting, standing, walking, and being in an empty room. To enhance readability and improve visualization, the first 7 carriers are displayed separately. This allows for a clear observation of different activity patterns during each 30-second interval. The second figure, depicted in Figure 5.11, illustrates the combination of all carriers from M1 to S3. Notably, these carriers exhibit very similar behavior patterns. However, when considering all 52 carriers, the amplitude displays a significant spread across the carriers, while the phase spread is notably smaller in comparison to the amplitude. In analyzing these patterns, it's interesting to observe that for the "sit" activity, the carrier pattern remains quite stable, whereas for the "walk" activity, the pattern becomes abruptly unstable. This variation in carrier spread serves as a valuable indicator for recognizing patterns in CSI behavior and its corresponding human activities. This observation allows for the interpretation that "sit" and "walk" activities exhibit higher CSI variance, indicating more pronounced changes in the wireless signals, while "stand" activity presents a lower CSI variance, signifying greater signal stability. These insights contribute to the understanding and interpretation of human activities based on the observed CSI patterns.



Figure 5.10: Subcarriers from M1 to sensor S3 involving different activities and pattern

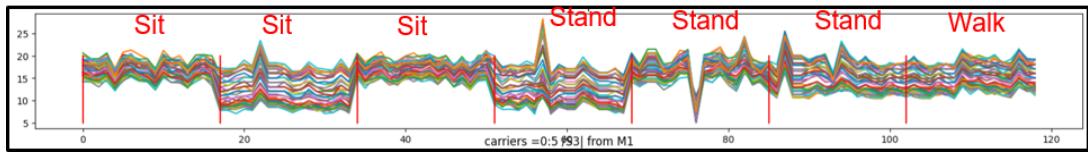


Figure 5.11: All Subcarriers from M1 to sensor S3 involving different activities and pattern

### 5.2.3 PCA Transformation

At the final stage of data processing, the amplitude and phase CSI images are merged for each 2-minute experiment to create composite images with a size of 24x24 pixels. This results in a total of 12 images for each 2-minute experiment Figure 5.12. In the course of the study, four individuals were involved, and each of them participated in two 2-minute experiments. These experiments encompass four different activities: sitting, standing, walking, and an empty room. Data was gathered for both vertical and horizontal antenna orientations during the data collection process. The CSI PCA images, which have been created through Principal Component Analysis (PCA), serve as the input data for the neural network training. This step represents a crucial phase in the overall process, where the machine learning model is trained to recognize and classify human activities based on the processed CSI data.

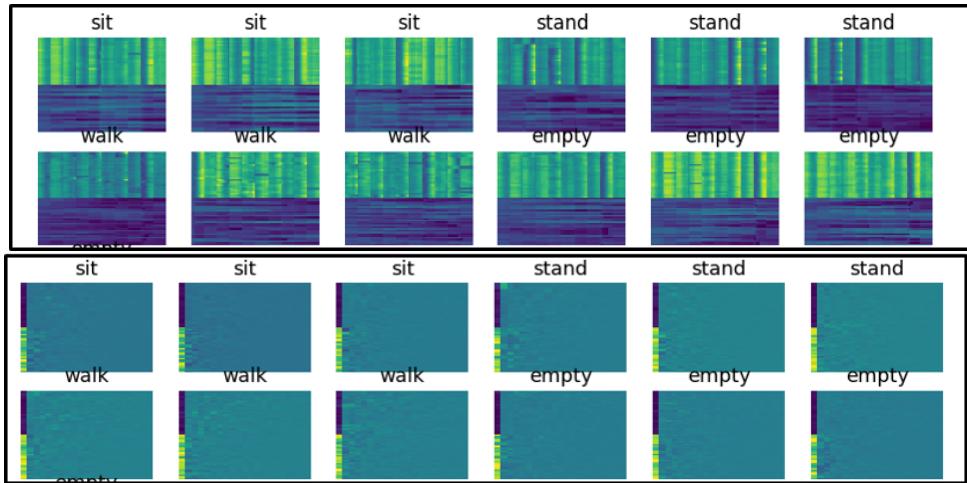


Figure 5.12: Amplitude and Phase CSI images get combined for PCA transformation

### 5.3 Feature Representation

For feature extraction and classification, the chosen method is the Convolutional Neural Network (CNN) classification approach. In total, there are 96 CSI data images for the Training environment and 96 CSI data images for the Test environment, encompassing all activities and experiments. These activities have been labeled numerically as follows: ["Empty room" = 0, "Sit" = 1, "Stand" = 2, "Walk" = 3]. In the initial run, a 1D convolution layer with a kernel size of 3x3 and padding = 1 is employed. This is followed by a max-pooling layer of size 2x2 with a stride of 2, and subsequently, a fully connected dense layer for classification with a size of 4. The training process was conducted over 50 epochs, with a learning rate set at 0.0001.

Listing 5.2: CNN layer configuration

```

1 class CNN(nn.Module):
2     def __init__(self):
3         super(CNN, self).__init__()
4         self.conv1 = nn.Conv2d(1, 32, kernel_size=3, padding
5             =1)
6         self.relu = nn.ReLU()
7         self.maxpool = nn.MaxPool2d(kernel_size=2, stride=2)
8         self.fc1 = nn.Linear(32 * 12 * 12, 128)

```

The best accuracy achieved with this simple CNN architecture is 42.553 %, with a training loss of 1.03 and a validation loss of approximately 1.40. However, it's important to note that there is a substantial difference between the training and validation loss, as indicated in Figure 5.13. This suggests that there may be issues related to overfitting or the model's generalization capability that need to be addressed.



Figure 5.13: Training vs Validation loss after Initial run

### 5.3.1 Adoptions and Improvements

#### 5.3.1.1 Adaptation approach 1

To address the overfitting issue observed in the initial CNN approach, a series of adaptation strategies has been implemented. The primary approach involves the use of data augmentation techniques, incorporating four specific methods:

- Random Crop Size: This technique introduces variability by randomly cropping different portions of the image during training.
- Horizontal Flip: The image is flipped horizontally to enhance the dataset and improve feature learning.
- Vertical Flip: Similar to horizontal flipping, this technique flips the image vertically for increased data diversity.
- Rotation Range: The images are rotated within a specified range to provide additional variations for training.

The application of data augmentation increases the size of the dataset, thereby enhancing the model's ability to generalize. The number of epochs is increased to further improve training. A dropout layer with a rate of 0.2 is introduced between the max-pooling and the fully connected dense layer. This dropout layer contributes to the model's generalization by reducing overfitting. It decreases the model's sensitivity to weight initialization and accelerates the learning process. Lastly, the resizing of images to a consistent size of 24x24 is applied to ensure uniformity in input image sizes, thereby enhancing the classification process. These adaptations collectively serve to address the overfitting issue and improve the model's performance. As illustrated in Figure 5.14, the model has demonstrated a remarkable performance improvement. The best accuracy achieved stands at an impressive 83.33 %, with the validation loss reaching a value close to 1.0, and the training loss at 1.05.



Figure 5.14: Training vs Validation loss after first adaptation run

These results unequivocally signify a substantial reduction in the overfitting problem, establishing the model as more robust and capable of accurate human activity classification based on CSI data.

### 5.3.1.2 Adaptation approach 2

Additional enhancements have been incorporated into the CNN architecture to further boost classification accuracy. The neural network's depth has been increased by introducing an extra fully connected dense layer equipped with a softmax classifier. Additionally, another Convolution 2D layer and Max-pooling layer have been integrated for more robust feature mapping.

Layer	Name	size	Parameters	Activation
1	Conv 2D	Kernel 3 * 3 (16)	Stride = 2	ReLU
2	Max-pooling	Kernel 2 * 2	Stride = 2	none
3	Conv 2D	Kernel 3 * 3 (32)	stride = 2	ReLU
4	Dropout	none	Rate = 0.2	none
5	Max-pooling	Kernel 2 * 2	Stride = 2	none
6	Fullu-connected	4	none	Softmax
7	fully-connected	4	none	Softmax

Table 5.1: CNN layer configuration

As evident from Figure 5.15, the model has exhibited remarkable progress following the implementation of two key adaptations. The best accuracy attained now stands at an impressive 93.055 %, with the validation loss reaching a value of 0.98 and the training loss hovering around 1.0. These results strongly suggest a significant improvement in human

activity recognition classification, affirming the model's enhanced capability and accuracy in this task.

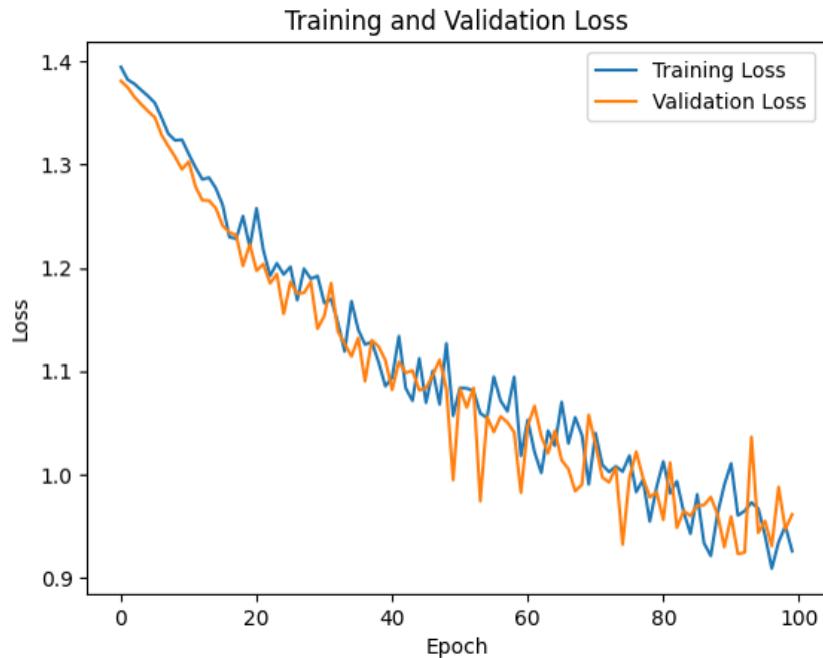


Figure 5.15: Training vs Validation loss after second adaptation run

## 5.4 Reinforcement Learning

### 5.4.1 Q - Learning approach

In an effort to enhance the accuracy of human activity classification, the approach of Reinforcement Learning (RL), particularly Q-learning, was explored. Initially, input CSI data images were utilized as input to the RL algorithm, along with a data augmentation approach. However, it's important to note that the initial accuracy achieved using only the Q-learning approach was relatively low at 13.55

Q-learning is traditionally designed for addressing problems with discrete action spaces, whereas image classification tasks typically involve continuous data and are better suited for deep learning techniques under supervised learning methods. Despite the challenges posed by Q-learning in image classification, it's worth acknowledging that RL algorithms hold promise in the context of transfer learning for applications related to building automation and energy conservation. While the initial results may not be as high in accuracy for image classification, RL has the potential to bring valuable insights and benefits in different areas of application. Q-Learning parameters are as follows:

Listing 5.3: Training loop for Q - learning algorithm

```

1 # Define the training loop
2 num_episodes = 1000
3 for episode in range(num_episodes):
4     # Initialize the state (corresponding to a CSI image
        class)
5     state = np.random.randint(0, num_states)
6
7     while True:
8         # Choose an action
9         action = choose_action(state)
10
11        # Simulate the environment (receive a reward, e.g.,
            based on correctness)
12        # In practice, you'd use a reward function based on
            classification accuracy
13        if action == state:
14            reward = 0.3 # Correct classification
15        else:
16            reward = -0.3 # Incorrect classification
17
18        # Update the Q-value for the current state-action
            pair
19        Q[state, action] = Q[state, action] + learning_rate *
            (reward + discount_factor * np.max(Q[action, :])
            - Q[state, action])
20
21        # Move to the next state (corresponding to a new CSI
            image class)

```

```

22     state = np.random.randint(0, num_states)
23
24     # Check if the episode is done
25     if episode_done:
26         break

```

### 5.4.2 Adaptation and Improvements

The concept involves the development of a Deep Q-learning Network (DQN) that integrates a 2D convolutional neural network and a Q-learning model. In the DQN model, the neural network replaces the traditional Q-table, aiming to achieve improved classification results while implementing the epsilon-greedy exploration strategy. Additionally, a custom environment is created using the openAI gym framework, designed to accommodate four distinct classes for training and testing. Further adjustments and hyperparameters have been fine-tuned, as detailed in the following Table 5.2, for comparative analysis and evaluation of the model's performance. These adjustments and the incorporation of DQN reflect an innovative approach to enhancing the classification accuracy for human activity recognition using CSI data.

Parameters	Initial Values	Updated Values
Batch Size	32	30
Discount Factor	0.9	0.8
Learning Rate	0.001	0.00025
Target update Freq.	20	10
Reward for correct action	+0.3	+0.1
Reward for Negative action	-0.3	-0.1
Optimizer	none	Adam

Table 5.2: Initial vs updated parameters

### 5.4.3 Flowchart of DQN model

The DQN algorithm is explained as follows

- **Initialization:** Initialize the DQN model with random weights. The model typically consists of a neural network with input units representing the state, output units representing Q-values for each action, and hidden layers.
- **Replay Memory:** Create a replay memory, also known as a replay buffer, to store experiences. This memory will store tuples of (state, action, reward, next state, done) from past interactions with the environment.
- **Define Hyperparameters:** Discount Factor (gamma): Determines the importance of future rewards. A higher value gives more weight to future rewards. Learning Rate (alpha): Controls the size of the updates to the Q-values during training. Batch Size: Specifies the number of experiences to sample from the replay memory for each

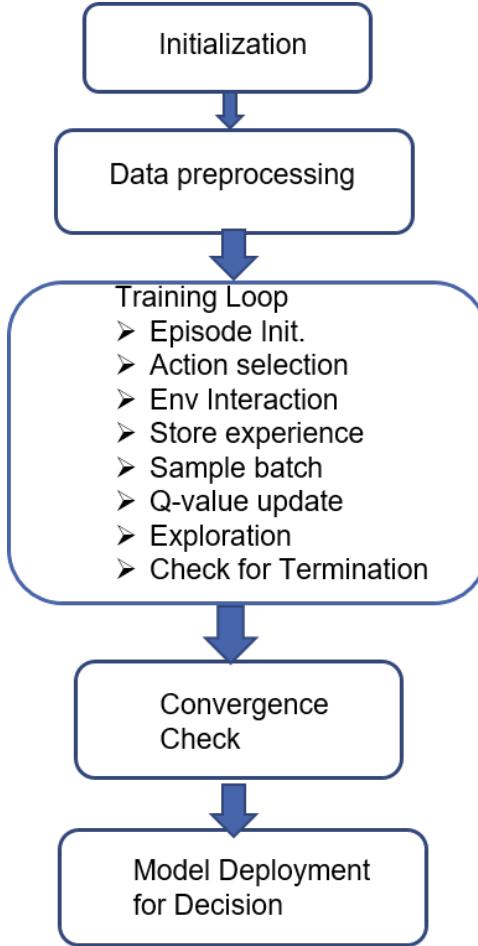


Figure 5.16: Flowchart of DQN model

training iteration. **Exploration Strategy:** Typically, an epsilon-greedy strategy is used. With probability  $\epsilon$ , select a random action; otherwise, select the action with the highest Q-value from the DQN model.

- **Training Loop:** Train for a specified number of episodes or until convergence is achieved.
- **Action Selection:** Choose an action using the current policy. With probability  $\epsilon$ , select a random action; otherwise, select the action with the highest Q-value from the DQN model.
- **Experience Storage:** Store the experience (state, action, reward, next state, done) in the replay memory.
- **Calculate Target Q-values:** Calculate the target Q-values for the next states in the batch using the target network. The target network is a copy of the DQN model with fixed weights.
- **Backpropagation and Update:** Backpropagate the error and update the weights of the DQN model using an optimizer (e.g., stochastic gradient descent).

- **Exploration Decay:** Decay the exploration parameter (epsilon) to gradually shift from exploration to exploitation over time. This can be done by reducing epsilon at a constant rate.
- **Convergence Check:** Periodically check for convergence using some criteria, such as the average return over episodes or a change in the Q-value estimates.

#### 5.4.4 Validation of RL using DQN



Figure 5.17: Training vs validation loss of DQN model

The DQN model has achieved a training accuracy of 75% in the training environment. In addition, the validation loss is recorded at 1.09, and the training loss stands at 1.13, as depicted in Figure 5.17. These results indicate that the DQN model has made substantial progress in human activity classification based on CSI data. While the accuracy may not be as high as previous approaches, the incorporation of reinforcement learning techniques like DQN provides a unique and promising avenue for further improvements in this domain.

# 6 Results and Discussion

## 6.1 Transfer Learning Results

The automated building system consists of training it in one environment and then using it in another environment (Transfer Learning). So long training phases are avoided by learning the common building features. Transfer learning is a valuable practice in building automation, involving the utilization of pre-trained machine learning models or knowledge gained in one building to enhance and tailor automation tasks in another building. This approach enables the efficient and effective customization of automation processes by leveraging existing expertise and adapting it to the unique requirements of different environments. Transfer learning can lead to accelerated and more data-efficient model development, ultimately enhancing automation tasks such as occupancy prediction, energy management, and more. In the context of this study, two environments within the same building have been employed. Initially, the model is exclusively trained at location 1 using training environment data. In the transfer learning phase, the model is trained at location 1 and then tested at location 2, encompassing both known and unknown locations. The results of this transfer learning process are presented below, showcasing the model's performance and adaptability to different building environments.



Figure 6.1: Person 1 performing the activity at location 1 and location 2

### 6.1.1 Cross Environment Validation using CNN

#### 6.1.1.1 At an Unknown location and a Known person

In this specific scenario where the location is unknown but the same individual from the training location is utilized, the model achieves a commendable accuracy of 88 %. The validation loss stands at 1.00, and the training loss is 0.98, as depicted in Figure 6.2. These results illustrate the adaptability and robustness of the model in recognizing human activities, even in an environment where the location is unfamiliar, provided the same individual is involved.

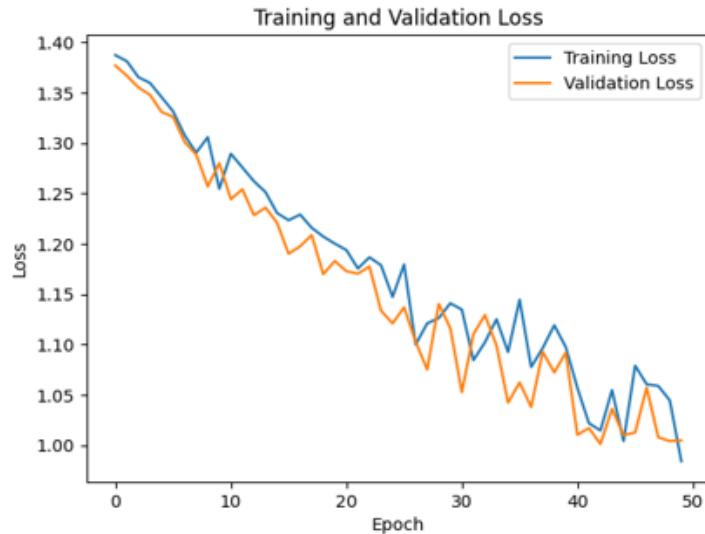


Figure 6.2: Validation vs Training loss at unknown location and Known person

#### 6.1.1.2 At an Unknown location and a Unknown person

In this particular case, both the location and the individual are unknown, as Person 1 from the training location and Person 3 from the test location are used. Despite these additional challenges, the model achieves a respectable accuracy of 79 %. The validation loss is recorded at 1.09, and the training loss is at 1.05, as depicted in Figure 6.3. These results highlight the model's adaptability to handle scenarios with unknown individuals and locations.

### 6.1.2 Cross Environment Validation using Reinforcement Learning (DQN)

#### 6.1.2.1 At an Unknown location and a Known person

In the first case where the location is unknown but the person is known, and reinforcement learning is used for transfer learning, the model achieves an accuracy of 73 %. The validation and training loss is recorded at 1.16 and 1.15, respectively as shown in Figure 6.4.



Figure 6.3: Validation vs Training loss at an unknown location and Unknown person

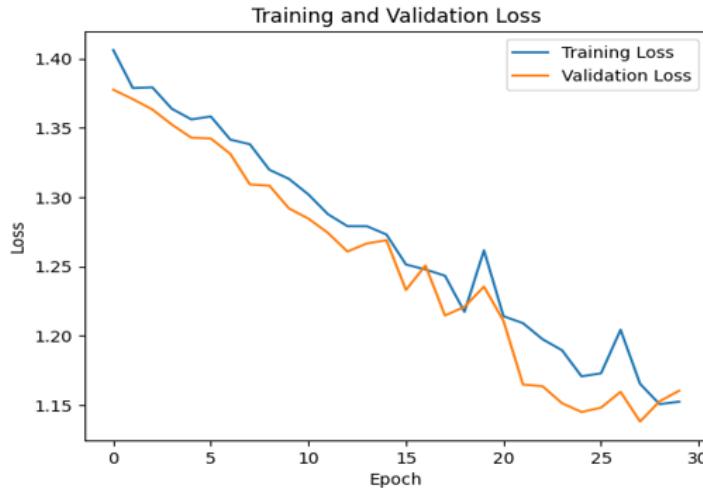


Figure 6.4: Validation vs Training loss at an unknown location and known person

#### 6.1.2.2 At an Unknown location and a Unknown person

In the second case, where both the location and the person are unknown, and specifically, Person 4 from the training location and Person 2 from the test location are used, the model attains an accuracy of 69 %. The validation and training loss are 1.17 and 1.18, respectively as shown Figure 6.5.

These results indicate that the model's performance can vary based on the level of information available regarding the location and individual, but it still demonstrates its adaptability to handle these challenging scenarios.



Figure 6.5: Validation vs Training loss at an unknown location and Unknown person

## 6.2 Overall result

### 6.2.1 HAR with Transfer Learning using Vertical Antenna Setup

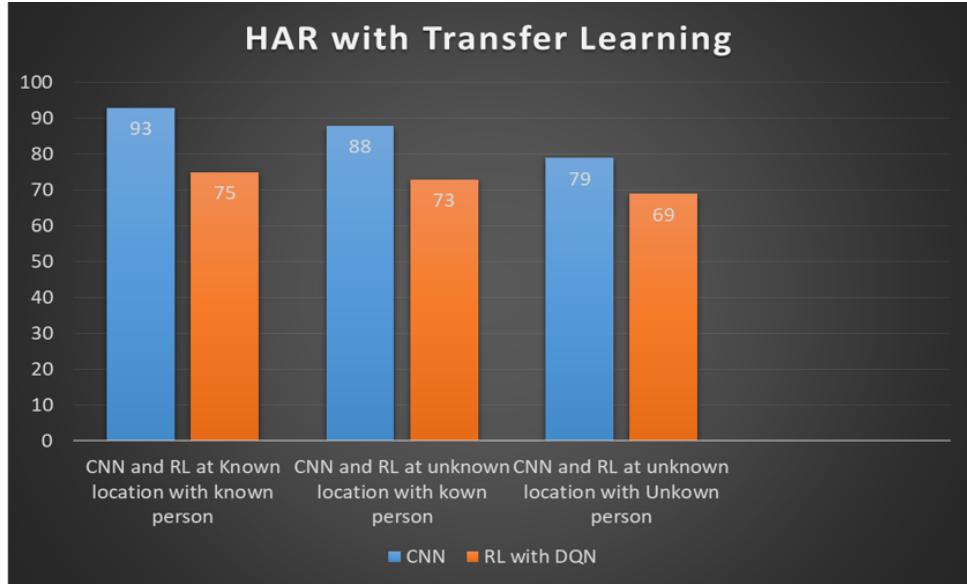


Figure 6.6: HAR with transfer learning result comparison using Vertical Antenna

The experiment involved four different individuals performing four distinct activities with varying antenna orientations. From the results presented in Figure 6.6, it's evident that the Convolutional Neural Network (CNN) achieved an impressive accuracy of 93 % in the scenario where both the location and person are known. Furthermore, when the location is unknown, CNN outperformed Reinforcement Learning (RL) by a substantial margin. In the case where RL achieved 69 % accuracy with both the location and person being unknown, the CNN model delivered a significantly higher accuracy of 79 % for the classification of human activity.

These results underscore the efficacy of CNN in human activity recognition, especially in scenarios with unknown locations and individuals, demonstrating its robustness and superiority over RL in these challenging conditions. CNNs are specifically designed for image analysis tasks. It excels at capturing spatial features, patterns, and structures in images. In the case of CSI image classification, the primary focus is identifying spatial features or abnormalities in the Images. Also, for RL to perform at its maximum potential requires a large dataset so agents can perform exploration and exploitation in the environment much better.

### 6.2.2 HAR with Transfer Learning using Horizontal Antenna Setup

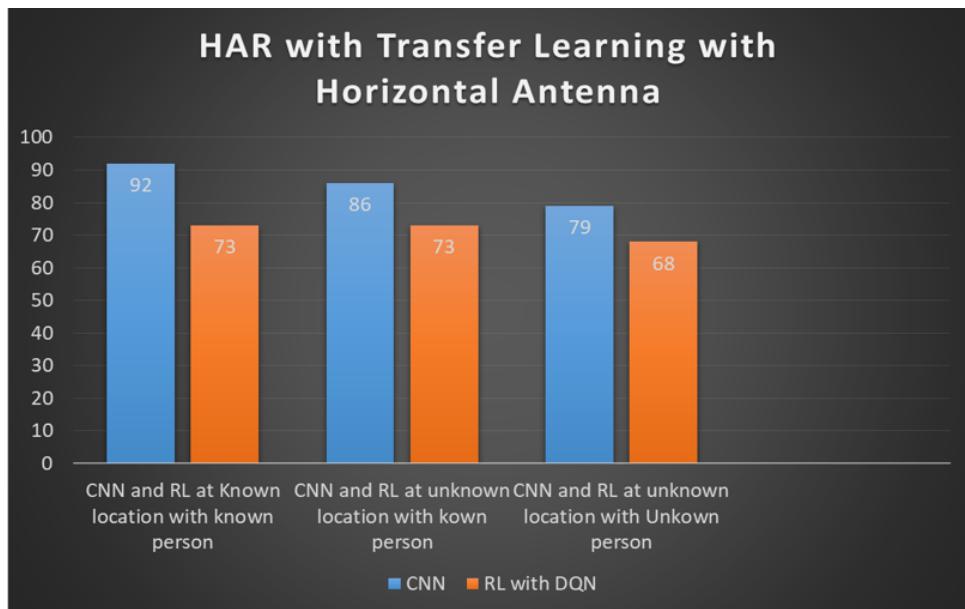


Figure 6.7: HAR with transfer learning result comparison using Horizontal Antenna

In the same experiment with a horizontal antenna setup Figure 6.7, the performance differences between CNN and RL were found to be negligible. CNN continued to outperform RL when a person and location are known, and in scenarios where both the location and person are unknown, CNN achieved a notable accuracy of 79 % compared to RL's 68 %. These results demonstrate the consistency of CNN's performance across different antenna setups, highlighting its effectiveness in human activity recognition tasks.

The primary objective of this thesis is to achieve human activity recognition for building automation and energy conservation using machine learning techniques. Transfer learning serves as a crucial component of this research, allowing the model to leverage knowledge from one context and apply it effectively to another. Furthermore, this thesis explores the relatively new research area of reinforcement learning (RL) for human activity recognition, specifically using image classification techniques. While convolutional neural network (CNN) machine learning models have demonstrated superior performance, RL, particularly when employing Q-learning, is a novel approach to image classification.

The findings in this research indicate that CNN models excel in image classification and outperform RL models. However, RL shows substantial promise in transfer learning, especially after adapting hyperparameters, leading to significant improvements in model

performance. Overall, this thesis highlights the significance of transfer learning, the potential of RL in image classification, and the proven effectiveness of CNN in human activity recognition, underscoring the importance of these techniques in building automation and energy-saving applications.

# **7 Summary**

## **7.1 Objective**

Revisit the objective adding some valuable observations. The primary objective of the Thesis is to research and analyze a robust system for Human activity recognition using Machine learning models within the context of building automation with a focus on energy-saving conservation.

## **7.2 Methodology**

Revisit methodology adding some experience. The study employs CSI data acquisition using low-cost microcontrollers, and machine learning techniques, with a particular emphasis on Convolutional Neural Networks, transfer learning, and reinforcement learning. The research explores the application of these techniques using Device-free Wi-Fi sensing to image classification tasks, specifically for human activity recognition.

## **7.3 Key findings**

### **7.3.1 Data Acquisition**

The data acquisition process consists of four individuals engaging in various activities for 30 seconds each, resulting in a two-minute experiment. These activities encompass sitting, standing, walking, and an empty state. The experiments are conducted with two antenna orientations, namely vertical and horizontal, within both training and testing environments. To gather Channel State Information (CSI), ESPNOW protocol is employed, utilizing a configuration of seven ESP32 modules, comprising two transmitters (Tx) and five receivers (Rx). The sequence of activities starts with sitting, transitions to standing, proceeds to walk, and concludes with the empty room activity. This data acquisition process aims to collect human activity data for subsequent analysis and classification.

### **7.3.2 Convolutional Neural Networks (CNN)**

CNN, a conventional approach for image classification and feature extraction, significantly outperforms RL models in human activity recognition with a small dataset. Its proven effectiveness in handling image-based tasks makes it a strong choice for building automation.

### 7.3.3 Reinforcement Learning (RL)

RL, particularly using Q-learning, is a relatively new research area in image classification for human activity recognition. While RL results show promise, especially after hyperparameter tuning and adaptation, they also reveal the need for further research and optimization in this domain. Reinforcement learning (RL) often requires a larger dataset to effectively train an agent. This dataset is crucial for the agent to perform epsilon-greedy exploration in the environment, ultimately leading to better classification results. The larger dataset enables the agent to explore a wider range of actions and states, allowing it to learn and improve its decision-making abilities over time.

### 7.3.4 Transfer Learning

Transfer learning plays a vital role in building automation. By leveraging pre-trained models and knowledge from one location to another, the research demonstrates its effectiveness in adapting and customizing automation tasks in different environments. This approach leads to more efficient and data-efficient model development.

### 7.3.5 Device-free Wi-Fi Sensing

Firstly, it enables device-free Wi-Fi sensing for building automation, providing a non-intrusive and privacy-conscious method for monitoring and managing activities within a building. In a world where concerns about privacy and the usage of cameras are growing, this approach circumvents those issues, as it doesn't rely on cameras or intrusive devices. Instead, it leverages existing Wi-Fi infrastructure and channel state information for activity recognition. Furthermore, the absence of RFID chips or dedicated devices is a significant advantage. This eliminates the need for additional hardware, reducing costs and simplifying implementation. Overall, this approach aligns with the contemporary focus on privacy and non-intrusive technology solutions while providing a cost-effective and efficient means of building automation. Performance Variability: The study highlights that the performance of machine learning models can vary depending on factors such as the level of knowledge about the location and the individuals involved in the activities. RL and CNN exhibit differing performances in scenarios with known or unknown locations and individuals.

## 7.4 Overall Significance

The research underscores the critical role of transfer learning in Building Automation and the potential of RL and CNN in image classification. While CNN is the preferred choice for human activity recognition due to its established image classification capabilities, RL presents opportunities for further exploration and optimization in this domain. The study's findings emphasize the importance of applying machine learning techniques to building automation for energy conservation. It highlights the strengths and limitations of transfer learning, RL, and CNN, providing insights into their performance in varying scenarios related to location and individual knowledge.

## 7.5 Limitations

The pursuit of advancing human activity recognition in building automation reveals several key limitations. First, the reliance on a relatively small dataset hampers the models' ability to generalize to broader real-world scenarios, particularly for reinforcement learning models, which thrive on larger datasets. Expanding the dataset's size would provide a more robust foundation for training and evaluating machine learning models, enhancing their applicability in diverse building automation contexts. Second, privacy and security concerns, although addressed with a privacy-conscious approach, remain a variable challenge contingent on legal regulations and individual circumstances. This underscores the importance of thorough privacy assessments when implementing the proposed solutions in various settings. Recognizing and addressing these limitations are crucial steps toward the effective deployment of building automation systems in practical applications.

## 8 Conclusion

In conclusion, this thesis has made significant contributions to the field of building automation and human activity recognition. By leveraging low-cost microcontrollers, Channel State Information (CSI) data acquisition, and machine learning models, we have advanced the state of device-free Wi-Fi sensing for image classification tasks, particularly in the domain of human activity recognition. Convolutional Neural Networks (CNNs) have demonstrated their superior performance, making them a practical choice for this application. While reinforcement learning (RL) exhibits potential, our research indicates the need for further exploration and adaptation, along with larger datasets for improved performance. Moreover, the pivotal role of transfer learning has been showcased in adapting automation tasks for different environments, aligning with contemporary priorities of privacy-conscious and cost-effective building automation solutions.

Our findings emphasize the practical importance of understanding contextual factors in machine learning models' performance and underscore the critical role of transfer learning in building automation. As we navigate an era of heightened privacy concerns and increasing demand for energy-efficient solutions, our research offers a balanced approach that leverages technology for automation while respecting privacy and cost-effectiveness. This thesis serves as a stepping stone for future research and practical implementations in the realm of building automation, enriching our understanding of human activity recognition and its role in energy conservation.

# 9 Perspectives

## 9.1 Future prospects for Human Activity Recognition (HAR)-based building automation

- **Multi-Environment Transfer Learning:** Extending transfer learning to multiple and diverse environments is a powerful concept. It allows the system to adapt to a wider range of building automation scenarios, making it more versatile and applicable in different contexts.
- **Real-Time Monitoring with Reinforcement Learning:** Real-time monitoring and decision-making using reinforcement learning can significantly enhance the system's ability to respond immediately to human activities. This offers the potential for dynamic and adaptive building automation in response to changing conditions.
- **Edge Computing and IoT Integration:** Leveraging edge computing and IoT technologies is pivotal for faster data processing and more efficient real-time decision-making. This integration can result in more responsive and agile building automation systems.
- **Reinforcement Transfer Learning for Customization:** The application of reinforcement transfer learning can be pivotal for customizing WiFi-sensed HAR-based room automation without the need for extensive data acquisition and training. This approach streamlines the implementation process and can lead to quicker deployment and adaptation.
- **Energy Efficiency and IoT Integration:** Energy savings, especially in areas like illumination and thermal comfort, are a direct and practical application of this technology. The integration of IoT within building automation can optimize energy consumption, making it a sustainable solution. The technology can play a pivotal role in optimizing Heating, Ventilation, and Air Conditioning (HVAC) systems, resulting in more energy-efficient and environmentally sustainable building management. It can dynamically adjust HVAC settings based on occupancy and activity patterns.

## 9.2 Additional areas where it can be applied

- **Additional Sensors for Enhanced Data Acquisition:** Incorporating a wider array of sensors can provide richer and more detailed data for human activity recognition. For example, environmental sensors (e.g., temperature, humidity) can be integrated to improve energy-saving strategies and indoor climate control.

- **Vehicle Applications:** Extending the technology to vehicle interiors can enhance automotive safety and comfort. It can be employed for occupant monitoring, recognizing different activities inside the vehicle, and personalizing the in-car experience.
- **Security Solutions:** The same technology can be harnessed for security applications, such as intrusion detection, access control, and surveillance. It can identify suspicious activities and raise alerts when necessary.

By expanding into these areas, the technology has the potential to revolutionize not only building automation but also various other domains, contributing to improved efficiency, safety, and user experience.

These future directions highlight the potential of HAR-based building automation to evolve and provide more responsive, efficient, and adaptable solutions for smart buildings. The convergence of transfer learning, real-time monitoring, edge computing, and IoT integration can result in intelligent and energy-efficient building automation systems.

# Abbreviations

Abbreviation	Meaning
CSI	Channel State Information
CNN	Convolutional Neural Network
CCMP	CBC MAC Protocol
CSD	Cyclic Shift Diversity
DQN	Deep Q-Learning Network
DRL	Deep Reinforcement Learning
DNN	Deep Neural Network
FDMA	Frequency Division Multiple Access
FEC	Forward Error Correction
FFT	Fast Fourier Transform
HVAC	Heating, Ventilation, and Air-Conditioning
HAR	Human Activity Recognition
HLK	Heizungs-, Lüftungs- und Klimatisierungssystemen
IoT	Internet of Things
LTE	Long Term Evolution
LTS	Long Training Symbols
LSTM	Long Short Term Memory
MAC	Medium Access Control
MIMO	Multiple Input Multiple Output
OFDM	Orthogonal Frequency Division Multiplexing
QoS	Quality of Service
RSSI	Received signal strength Indicator
RL	Reinforcement Learning
RNN	Recurrent Neural Network
SBEM	Smart Building Energy Management
SFO	Sampling Frequency Offset
Wi-Fi	Wireless Fidelity

# Bibliography

- [1] Liang Yu et al. “A review of deep reinforcement learning for smart building energy management”. In: *IEEE Internet of Things Journal* 8.15 (2021), pp. 12046–12063.
- [2] Shushan Hu et al. “Environmental and energy performance assessment of buildings using scenario modelling and fuzzy analytic network process”. In: *Applied Energy* 255 (2019), p. 113788.
- [3] Shushan Hu et al. “Building performance evaluation using OpenMath and Linked Data”. In: *Energy and Buildings* 174 (2018), pp. 484–494.
- [4] Amit Prasad and Ivana Dusparic. “Multi-agent deep reinforcement learning for zero energy communities”. In: *2019 IEEE PES innovative smart grid technologies Europe (ISGT-Europe)*. IEEE. 2019, pp. 1–5.
- [5] Xiangxiang Dong et al. “Optimal scheduling of distributed hydrogen-based multi-energy systems for building energy cost and carbon emission reduction”. In: *2020 IEEE 16th International Conference on Automation Science and Engineering (CASE)*. IEEE. 2020, pp. 1526–1531.
- [6] Ehsan Haggi et al. “Assessing the potential of surplus clean power in reducing GHG emissions in the building sector using game theory; a case study of Ontario, Canada”. In: *IET Energy Systems Integration* 1.3 (2019), pp. 184–193.
- [7] ABC Global. “Global status report for buildings and construction”. In: *Global Alliance for Buildings and Construction* (2020).
- [8] Sumedha Sharma et al. “Time-coordinated multienergy management of smart buildings under uncertainties”. In: *IEEE Transactions on Industrial Informatics* 15.8 (2019), pp. 4788–4798.
- [9] Nan Zhou et al. “Scenarios of energy efficiency and CO<sub>2</sub> emissions reduction potential in the buildings sector in China to year 2050”. In: *Nature Energy* 3.11 (2018), pp. 978–984.
- [10] Karl Mason and Santiago Grijalva. “A review of reinforcement learning for autonomous building energy management”. In: *Computers & Electrical Engineering* 78 (2019), pp. 300–312.
- [11] Daniel Minoli, Kazem Sohraby, and Benedict Occhiogrosso. “IoT considerations, requirements, and architectures for smart buildings—Energy optimization and next-generation building management systems”. In: *IEEE Internet of Things Journal* 4.1 (2017), pp. 269–283.
- [12] Basheer Qolomany. “Ala Al-Fuqaha, Ajay Gupta, Driss Benhaddou, Safaa Alwajidi, Junaid Qadir, and Alvis C Fong. 2019. Leveraging machine learning and big data for smart buildings: A comprehensive survey”. In: *IEEE Access* 7 (2019), pp. 90316–90356.

- [13] Xiangyu Zhang et al. “An IoT-based thermal model learning framework for smart buildings”. In: *IEEE Internet of Things Journal* 7.1 (2019), pp. 518–527.
- [14] Wei Feng et al. “A conditional value-at-risk-based dispatch approach for the energy management of smart buildings with HVAC systems”. In: *Electric Power Systems Research* 188 (2020), p. 106535.
- [15] Bin Yang et al. “Non-invasive (non-contact) measurements of human thermal physiology signals and thermal comfort/discomfort poses-a review”. In: *Energy and Buildings* 224 (2020), p. 110261.
- [16] Fei Wang et al. “Multi-objective optimization model of source–load–storage synergetic dispatch for a building energy management system based on TOU price demand response”. In: *IEEE Transactions on Industry Applications* 54.2 (2017), pp. 1017–1028.
- [17] Rufeng Zhang et al. “Stochastic optimal energy management and pricing for load serving entity with aggregated TCLs of smart buildings: A Stackelberg game approach”. In: *IEEE Transactions on industrial informatics* 17.3 (2020), pp. 1821–1830.
- [18] Tianshu Wei, Yanzhi Wang, and Qi Zhu. “Deep reinforcement learning for building HVAC control”. In: *Proceedings of the 54th annual design automation conference 2017*. 2017, pp. 1–6.
- [19] Liang Yu et al. “Multi-agent deep reinforcement learning for HVAC control in commercial buildings”. In: *IEEE Transactions on Smart Grid* 12.1 (2020), pp. 407–419.
- [20] Chi Zhang et al. “A cooperative multi-agent deep reinforcement learning framework for real-time residential load scheduling”. In: *Proceedings of the International Conference on Internet of Things Design and Implementation*. 2019, pp. 59–69.
- [21] Elena Mocanu et al. “On-line building energy optimization using deep reinforcement learning”. In: *IEEE transactions on smart grid* 10.4 (2018), pp. 3698–3708.
- [22] Niloofar Bahadori, Jonathan Ashdown, and Francesco Restuccia. “ReWiS: Reliable Wi-Fi sensing through few-shot multi-antenna multi-receiver CSI learning”. In: *2022 IEEE 23rd International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*. IEEE. 2022, pp. 50–59.
- [23] Yongsen Ma et al. “Location-and person-independent activity recognition with WiFi, deep neural networks, and reinforcement learning”. In: *ACM Transactions on Internet of Things* 2.1 (2021), pp. 1–25.
- [24] Jiang Xiao et al. “A survey on wireless indoor localization from the device perspective”. In: *ACM Computing Surveys (CSUR)* 49.2 (2016), pp. 1–31.
- [25] Jieming Yang et al. “A framework for human activity recognition based on WiFi CSI signal enhancement”. In: *International Journal of Antennas and Propagation* 2021 (2021), pp. 1–18.
- [26] Vinicius Galvao Guimaraes et al. “A novel IoT protocol architecture: Efficiency through data and functionality sharing across layers”. In: *2019 28th International Conference on Computer Communication and Networks (ICCCN)*. IEEE. 2019, pp. 1–9.

- [27] Kamran Ali et al. “Recognizing keystrokes using WiFi devices”. In: *IEEE Journal on Selected Areas in Communications* 35.5 (2017), pp. 1175–1190.
- [28] Chunhai Feng, Sheheryar Arshad, and Yonghe Liu. “MAIS: Multiple activity identification system using channel state information of WiFi signals”. In: *Wireless Algorithms, Systems, and Applications: 12th International Conference, WASA 2017, Guilin, China, June 19-21, 2017, Proceedings* 12. Springer. 2017, pp. 419–432.
- [29] Adolfo Bauchspies, Alexandre Saran Rodrigues, and Mariana Pimentel Martins da Silva. “Controle Antecipativo por Estimativa de Carga Térmica em Vídeo”. In: (2019).
- [30] Keerthana Sivamayil et al. “A systematic study on reinforcement learning based applications”. In: *Energies* 16.3 (2023), p. 1512.
- [31] Andrii Zhuravchak, Oleg Kapshii, and Evangelos Pournaras. “Human activity recognition based on wi-fi csi data-a deep neural network approach”. In: *Procedia Computer Science* 198 (2022), pp. 59–66.
- [32] Sheheryar Arshad et al. “Wi-chase: A WiFi based human activity recognition system for sensorless environments”. In: *2017 IEEE 18th International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM)*. IEEE. 2017, pp. 1–6.
- [33] Xinbin Shen et al. “WiAgent: Link selection for CSI-based activity recognition in densely deployed wi-Fi environments”. In: *2021 IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE. 2021, pp. 1–6.
- [34] Yong Zhang et al. “CSI-based human activity recognition with graph few-shot learning”. In: *IEEE Internet of Things Journal* 9.6 (2021), pp. 4139–4151.
- [35] Chen Chen, Gang Zhou, and Youfang Lin. “Cross-Domain WiFi Sensing with Channel State Information: A Survey”. In: *ACM Computing Surveys* 55.11 (2023), pp. 1–37.
- [36] Yongsen Ma, Gang Zhou, and Shuangquan Wang. “WiFi sensing with channel state information: A survey”. In: *ACM Computing Surveys (CSUR)* 52.3 (2019), pp. 1–36.
- [37] Norah Alrebdi et al. “Reinforcement Learning in Image Classification: A Review”. In: *2022 2nd International Conference on Computing and Information Technology (ICCIT)*. IEEE. 2022, pp. 79–86.
- [38] Ayas Shaqour and Aya Hagishima. “Systematic Review on Deep Reinforcement Learning-Based Energy Management for Different Building Types”. In: *Energies* 15.22 (2022), p. 8663.
- [39] Zidong Zhang, Dongxia Zhang, and Robert C Qiu. “Deep reinforcement learning for power system applications: An overview”. In: *CSEE Journal of Power and Energy Systems* 6.1 (2019), pp. 213–225.
- [40] Ming-Xian Chang and Yu T Su. “Model-based channel estimation for OFDM signals in Rayleigh fading”. In: *IEEE Transactions on Communications* 50.4 (2002), pp. 540–544.

# List of Tables

5.1	CNN layer configuration . . . . .	42
5.2	Initial vs updated parameters . . . . .	45

# List of Figures

1.1	CSI Transmission affected by surrounding and different activities . . . . .	2
1.2	Machine Learning . . . . .	3
1.3	Research Scope area Image classification with RL . . . . .	4
2.1	ESP32 NodeMCU micro-controller ESP32-WROOM-32UE . . . . .	7
2.2	Broadcast Connection or one to many . . . . .	9
2.3	vendor-specific action frame . . . . .	9
2.4	Vendor specific Field . . . . .	9
2.5	OFDM Signal Frequency Spectra . . . . .	10
2.6	Basic structure of MIMO . . . . .	11
2.7	CSI matrices for MIMO-OFDM channels. It captures multi-path channel variations, including amplitude attenuation and phase shifts, in spatial, frequency, and time domains . . . . .	13
2.8	CNN architecture . . . . .	14
2.9	A situation(state)in the Tic-Tac-Toe Game . . . . .	15
2.10	Reinforcement Learning Architecture . . . . .	16
2.11	Caption . . . . .	17
2.12	Deep Neural Network and Deep Reinforcement Learning . . . . .	18
2.13	Overview of Q-Learning and Deep Q-learning Network . . . . .	21
3.1	Acuuracy comparison of different Deep Learning solutions . . . . .	22
3.2	Experiment setup. There are 4 WiFi receivers placed at different locations with different heights and antenna orientations . . . . .	23
3.3	The training process of the proposed design . . . . .	24
3.4	Experimental Setup Configuration . . . . .	25
3.5	The overall accuracy of the three cases and the numbers of Wi-Fi links used in these cases . . . . .	26
4.1	Framework of Overall Human Activity Recognition using Machine Learning	28
5.1	Configuration setup of the ESP32 sensors . . . . .	31
5.2	Workflow of CSI transmission . . . . .	32
5.3	Training and Testing Environment of CSI data Acquisition . . . . .	33
5.4	Person 1 performing activities in training and Testing Environment . . . . .	34
5.5	1 cycle of CSI data . . . . .	35
5.6	Border Time . . . . .	36
5.7	CSI Amplitude and Phase mapping, PCA Image . . . . .	37
5.8	Individual Activity with Amplitude and Phase CSI images . . . . .	37
5.9	Surface Visualisation of CSI Image and Data . . . . .	38
5.10	Subcarriers from M1 to sensor S3 involving different activities and pattern	39
5.11	All Subcarriers from M1 to sensor S3 involving different activities and pattern	39

5.12 Amplitude and Phase CSI images get combined for PCA transformation . . . . .	40
5.13 Training vs Validation loss after Initial run . . . . .	41
5.14 Training vs Validation loss after first adaptation run . . . . .	42
5.15 Training vs Validation loss after second adaptation run . . . . .	43
5.16 Flowchart of DQN model . . . . .	46
5.17 Training vs validation loss of DQN model . . . . .	47
6.1 Person 1 performing the activity at location 1 and location 2 . . . . .	48
6.2 Validation vs Training loss at unknown location and Known person . . .	49
6.3 Validation vs Training loss at an unknown location and Unknown person	50
6.4 Validation vs Training loss at an unknown location and known person .	50
6.5 Validation vs Training loss at an unknown location and Unknown person	51
6.6 HAR with transfer learning result comparison using Vertical Antenna .	51
6.7 HAR with transfer learning result comparison using Horizontal Antenna .	52