ABSTRACT

Title of Dissertation: WIRELESS SENSING AND ANALYTICS FOR MOTION MONITORING AND MAPPING

Guozhen Zhu
Doctor of Philosophy, 2023

Dissertation Directed by: Professor K. J. Ray Liu
Department of Electrical and Computer Engineering

Environmental perception is pivotal for intelligent systems, enabling them to adeptly capture, interpret, and act upon contextual cues. Grasping the intricacies of the environment—its objects, occupants, floor plan, and dynamics—is fundamental for the effective deployment of technologies, including robotics, the Internet of Things (IoT), and augmented reality. Traditional perception mechanisms, such as video surveillance and sensor-based monitoring, are often hampered by privacy concerns, substantial infrastructural costs, energy inefficiencies, and limited coverage. In contrast, WiFi sensing stands out for its non-intrusive, cost-effective, and pervasive attributes. Capitalizing on ubiquitous WiFi signals that permeate both indoor and outdoor spaces, WiFi sensing delivers unparalleled advantages over its traditional counterparts, sidestepping the need for extra hardware yet offering profound environmental insights. Its capability to penetrate walls and other obstructions further broadens its range, covering areas beyond the reach of conventional sensors. These unique edges of WiFi sensing elevate its value across diverse applications, spanning smart homes, health monitoring, location-based services,

and security systems. Amplifying environmental perception via WiFi sensing is more than just an innovation in ubiquitous computing; it's a leap towards forging safer, more efficient, and smarter environments. This dissertation explores monitoring and mapping environments leveraging motion analytics based on commodity WiFi.

In the first part of this dissertation, we introduce an efficient and cost-effective system for precise floor plan construction by integrating RF and inertial sensing techniques. The proposed system harnesses detailed insights from RF tracking and broad context from inertial metrics, such as magnetic field strength, to produce an accurate map. The system employs a robot for trajectory collection and requires only a single Access Point to be arbitrarily installed in space, both of which are widely available nowadays. Impressively, the system can produce detailed maps even with minimal data, making it adaptable for diverse structures such as shopping centers, offices, and residences without significant expenses. We validated the efficacy of the proposed system using a Dji RoboMaster S1 robot equipped with standard WiFi across three distinct buildings, demonstrating its capability to produce reliable maps for the intended regions. Given the widespread presence of WiFi setups and the increasing prevalence of domestic robots, the proposed approach paves the way for universal intelligent systems offering indoor mapping services.

In the second and third parts, we present two innovative strategies leveraging WiFi to identify the motion of human and various non-human subjects. Initially, we detail a novel passive, non-intrusive methodology tailored for edge devices. By extracting and analyzing motion's physically and statistically plausible features, our system recognizes human and diverse non-human subjects through walls using a singular WiFi link. Experimental results from four distinct buildings with various moving subjects validate its efficiency on edge devices. Advancing to

more intricate cases, we put forth a deep learning-based WiFi sensing paradigm. This delves into the efficacy of diverse deep learning models on human and non-human object recognition and probes the feasibility of transferring image-trained models to fulfill the WiFi sensing task. Designed with a robust statistic invariant to the environment and position, this system efficiently adapts to new surroundings. Comprehensive experimental evaluations affirm our framework's precision in pinpointing intricate human and non-human subjects, and readiness for integration into prevalent intelligent systems, thereby boosting their perceptual capacities.

In the final part of this dissertation, we propose a pioneering through-wall indoor intrusion detection system that adeptly filters out interference from non-human subjects using ubiquitous WiFi signals. A novel deep learning architecture is proposed for single-link WiFi signal analysis. It employs a ResNet-18-based module to extract features of indoor moving subjects and an LSTM-based module to incorporate temporal information for efficient intrusion detection. Notably, the system is invariant to environmental changes, angles, and positions, enabling swift deployment in new environments without additional training. Evaluation in five indoor environments with various interference yielded high intrusion detection accuracy and a low false alarm rate, even without model tuning for unseen settings. The results underscore the system's exceptional adaptability, positioning it as a top contender for widespread intelligent indoor security applications.

WIRELESS SENSING AND ANALYTICS FOR MOTION MONITORING AND MAPPING

by

Guozhen Zhu

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2023

Advisory Committee:
Professor K. J. Ray Liu, Chair/Advisor
Professor Gang Qu, Co-Chair
Professor Min Wu
Dr. Beibei Wang
Professor Lawrence C. Washington

Dedication

To my family —

Xinhua Zhu, and Wenchuan Guo

Acknowledgments

I am immensely grateful for the collective support, mentorship, and camaraderie that have contributed to the completion of this dissertation.

First and foremost, my heartfelt gratitude goes out to my advisor, Professor K. J. Ray Liu, for his belief in my abilities, which paved the way for the exploration of challenging yet profoundly intriguing projects throughout my four-year journey. His continual availability and prompt response for guidance and inquiries never wavered - there hasn't been an instance when I reached out for advice and he hasn't generously lent his time. The experience of collaborating with and learning from such an extraordinary individual has been an unparalleled privilege and pleasure.

I am sincerely thankful to my mentors, Dr. Chenshu Wu and Dr. Beibei Wang, for their invaluable advice, patience, mentorship, and for constantly challenging me to achieve more. I am truly inspired by their commitment and dedication to academic excellence. Their suggestions, discussions, and dedication to my progress have been a beacon throughout this journey.

I would like to thank my co-authors, Dr. Yuqian Hu, Dr. Xiaolu Zeng, and Weihang Gao for their valuable contributions, intellectual stimulation, and spirited discussions. They have enriched my research experience and boosted my learning curve.

I would like to extend my gratitude to my committee members, Prof. Min Wu, Prof. Gang Qu, and Prof. Lawrence C. Washington, for their precious time and efforts in serving on my

dissertation committee and reviewing the manuscript.

My journey would not have been the same without the camaraderie and support from my lab mates, Wei-Hisang Wang, Dr. Sai Deepika Regani, Dr. Muhammed Zahid Ozturk, Dr. Fengyu Wang, and Sakila S. Jayaweera. Their collective contribution in providing a collaborative, intellectually stimulating environment was invaluable. I am fortunate to have shared this journey with such a dedicated and friendly group of individuals.

I wish to express my special thanks to my best friend Mengting Xing, for her support, love, and understanding. Her constant motivation and companionship provided comfort and strength during challenging times. I also want to thank my wonderful friends here, Ke Gong, Yexin Cao and Xiaojue Chen, for their friendship and unwavering support during this journey.

Finally, I would like to express my deepest gratitude to my parents, Mr. Xinhua Zhu and Dr. Wenchuan Guo. Your unconditional love, encouragement, advice, and faith in my abilities have been my rock. Thank you for instilling in me the values of hard work and perseverance, and for your endless sacrifices to provide me with the best opportunities. This achievement is as much yours as it is mine.

To all mentioned and those unmentioned who have contributed to my academic journey in any form, I am eternally grateful. I am reminded of an African proverb, “If you want to go fast, go alone. If you want to go far, go together.” This journey has been far and enriching, and I could not have done it without each and every one of you.

Thank you all.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

A-ACF	Amplified Autocorrelation Function
ACF	Autocorrelation Function
AoA	Angle of Arrival
AP	Access Point
CDF	Cumulative Distribution Function
CFO	Channel Frequency Offset
CFR	Channel Frequency Response
CIR	Channel Impulse Response
CNN	Convolutional Neural Network
CPU	Central Processing Unit
CSI	Channel State Information
DTW	Dynamic Time Warping
DFS	Doppler Frequency Shift
EM	Electromagnetic
FMCW	Frequency Modulated Continuous Wave
FNN	Forward Neural Network
FFT	Fast Fourier Transform
FPR	False Positive Rate
GHz	Gigahertz
GPU	Graphics Processing Unit
GRUNet	Gated Recurrent Unit Network
HMM	Hidden Markov Model
IDM	Intrusion Detection Module
IoT	Internet of Things
IMU	Inertial Measurement Unit
LBS	Location-Based Services

LOS	Line of Sight
LSTM	Long Short Term Memory
MFS	Magnetic Field Strength
MHz	Megahertz
MIMO	Multiple Input Multiple Output
MLP	Multilayer Perceptron
mmWave	Millimeter-Wave
MRC	Maximum Ratio Combine
MRM	Motion Recognition Module
NLOS	Non Line of Sight
NMI	Normalized Mutual Information
PCA	Principal Component Analysis
RAM	Random-Access Memory
RF	Radio Frequency
RNN	Recurrent Neural Network
RSS	Received Signal Strength
RSSI	Received Signal Strength Indicator
Rx	Receiver
SFO	Sampling Frequency Offset
SLAM	Simultaneous Localization and Mapping
SNR	Signal-to-Noise Ratio
STFT	Short-Time Fourier Transform
STO	Symbol Timing Offset
SVM	Support-Vector Machines
ToA	Time of Arrival
ToF	Time of Flight
TPR	True Positive Rate
TRRS	Time Reversal Resonance Strength
Tx	Transmitter
UWB	Ultra-Wide Band
ViT	Vision Transformer

Chapter 1: Introduction

1.1 Motivation

Environmental perception is integral to the effectiveness of intelligent systems. It enables these systems to understand, map, and monitor their surroundings, which is crucial for their proper functioning and optimization. Mapping the environment allows a system to accurately determine its position and plan routes, a fundamental aspect for applications like autonomous vehicles and robotics. Similarly, monitoring enables a system to detect changes or anomalies in the environment, key for predictive maintenance, security, or hazard detection. In essence, the ability to perceive the environment underpins the core functionalities of intelligent systems, enhancing their effectiveness, safety, and efficiency.

WiFi signals are ubiquitous in our modern digital world, providing wireless connectivity for a plethora of devices ranging from smartphones to smart home appliances. These signals, which invisibly traverse our environment, are a cornerstone of wireless communication and have opened up a new avenue of WiFi sensing. By detecting alterations in WiFi signal characteristics due to interactions with objects or individuals, WiFi sensing provides a new layer of environmental awareness. It offers numerous advantages, including: 1) non-intrusive and privacy-preserving sensing, 2) lower costs and energy consumption by leveraging existing infrastructure, 3) the ability to penetrate penetrate walls and other opaque objects for hard-to-reach sensing areas, 4)

high scalability due to the ubiquity of WiFi, and 5) resilience to lighting conditions for continuous operation.

Motivated by the compelling benefits of WiFi-based sensing and the critical need to augment the perceptual capabilities of intelligent systems, this dissertation explores the innovative use of ubiquitous WiFi signals for environmental perception enhancement through motion analytics, with a specific focus on mapping and monitoring. We present an advanced suite of solutions that includes an *automated algorithm for floor plan construction*, two distinctive approaches for *identifying human and non-human subjects*, and a robust *indoor intrusion detection system*. By pushing the frontiers of Wi-Fi sensing, this research marks a significant advancement in the realm of intelligent systems, fostering their evolution towards heightened perceptual precision and superior situational awareness.

- *Automatic floor plan construction* is a vital component for intelligent systems. Providing a bird's eye view of an environment, floor plans are crucial for navigation, localization, and planning. For systems such as autonomous vehicles and mobile robots, they enable efficient path planning and collision avoidance. Furthermore, they assist in resource allocation for indoor positioning systems, improving services like indoor navigation and emergency response. Therefore, the development of automated floor plan construction techniques significantly enhances the perceptual capabilities and operational efficiency of intelligent systems.

- *Human and non-human motion identification* is a critical capability for intelligent systems, particularly for applications in security, automation, and user interaction. By distinguishing between human and non-human entities, systems can perform tasks more

efficiently and accurately, ensuring proper response based on the detected subject type. Given the prevalence of pets, robotic vacuum cleaners, and electrical appliances, especially in residential environments, it is crucial to develop a reliable system that can accurately recognize human and non-human subjects.

- *Indoor intrusion detection* is integral for maintaining safety and security in various environments. An advanced intrusion detection system can provide real-time alerts and minimize the false alarms caused by non-human subjects. In the context of intelligent systems, the incorporation of robust intrusion detection can significantly elevate their utility and effectiveness, making them indispensable tools for proactive security management. By enhancing the ability of these systems to detect intrusions and mitigating the interference from non-human motion, we can ensure a higher level of security and peace of mind for individuals and organizations alike.

1.2 Related Works

The related works of this dissertation cover floor plan construction, human and non-human motion identification, and indoor intrusion detection, which are reviewed in the following subsections, respectively.

1.2.1 Floor Plan Construction

Although projects like Google Indoor Maps, Point Inside, and Micello Indoor Map aim at collecting indoor maps, the availability of indoor maps is still very limited considering the huge number of buildings worldwide. Furthermore, most floor plans are manually generated and

uploaded in these projects. The traditional manual methods require professional technicians to draw the floor plan using specialized measurement devices, which is time-consuming, costly, and thus unaffordable to cover all buildings. In addition, potential changes in the indoor environment would require much effort to update the maps.

Efforts have been taken to generate indoor maps automatically. Most indoor map construction approaches are based on Simultaneous Localization and Mapping (SLAM), which is widely used in robotics to locate robots in unknown scenarios by mapping the environment simultaneously. Among numerous SLAM systems, many of them are based on vision [1–5]. These systems could construct 3D indoor maps by modeling the environment through images obtained by cameras. However, the cameras have a high requirement for light, making these systems unnable to work in an environment with poor lighting. Besides, there are many privacy concerns in vision-based SLAM systems. Lasers are commonly used to combat the concerns of vision-based systems. However, laser-based systems are costly and need special equipment. Inertial Measurement Unit (IMU) is free of these drawbacks, but it suffers from odometry deviation, especially over a long distance.

Pedestrian Dead Reckoning (PDR) algorithm is often employed to estimate the trajectories from inertial sensor data [6–14]. However, PDR estimated trajectories by inertial sensors usually suffer from large deviation, especially in direction. Thus, when these trajectories are employed for floor plan construction, several methods have been proposed to correct the deviation error caused by dead reckoning. In CrowdInside [7] and SenseWit [6], a large number of anchor points, such as water dispensers and doors, are utilized to overcome the derivation of IMU. However, in many buildings, these anchor points are too sparse to correct the IMU errors. Thus, the accuracy and generation performance would decrease in buildings without sufficient anchor

points employed in their methods. Furthermore, to construct a map with erroneous trajectories, inertial sensor-based systems require a vast amount of data to estimate the reachable area in the environment. Different from current works based on PDR using inertial sensors, *EZMap* requires neither any anchor points nor a large number of trajectories. We propose a hierarchical matching scheme to accurately match trajectories at the exact location by considering both the geometric shape estimated from the RF tracking and ambient properties such as RSSI and the MFS.

Different approaches have been developed to generate maps from crowdsourced data, which can be classified into two categories: grid-based and topological-based. For grid-based map construction, Gaussian distribution is often used to compute the accessible probability of each cell to create the occupancy grid map. Grids in the matrix are described by values between two states of 0 and 1, where 0 represents that the grid is accessible and 1 represents that objects fully occupy the grid. The value between 0 and 1 represents the percentage of the area that an object occupies the grid [8, 9]. These approaches are easy to build and maintain but suffer from enormous storage space and are time-consuming [15].

Since the topological-based approach is more efficient than the grid-based approach, it attracts more attention from researchers. Walkie-Markie [16] applies the spring relaxation concept, treating the encountered WiFi landmarks as nodes and the user trajectories as springs. As more user trajectories are collected, the node positions are adjusted so as to achieve the minimum system potential energy. CrowdInside [7] uses identifiable acceleration patterns of the elevators, escalators, and stairs to correct dead reckoning, hence constructing the map. This approach fails when the users mainly stay on the same floor, e.g., an exhibition in a one-storey hall or a single-floor office such as that in our experiments.

In [17], users have to hold the Arduino sensor, and need to wait for the servomotor to sweep

the surroundings. [18] requires the users to walk in a straight line, and the corridors should be perpendicular to each other, which means it cannot be applied to a structure with curved paths. A preliminary version of this work has been published in [19] based on crowdsourcing with carts. In this paper, we improve the work by tackling curve paths and providing detailed information about the environment, such as open spaces and rooms.

In addition, existing approaches mainly crowdsource data from a large number of mobile users and targets public spaces like malls and office buildings. Small private spaces like homes are less explored. Differently, *EZMap* employs the increasingly popular home robots for data collection and extends the scope to home environments.

1.2.2 Human and Non-Human Motion Identification

While there have been numerous indoor activity recognition and monitoring methods proposed over the years, only a few have taken into account the impact of non-human motion. Some researchers have attempted to filter out pet motion from human motion due to its potential interference with their systems. In this section, we review the current methods used to differentiate pets from humans, focusing on non-RF-based and RF-based methods.

Non-RF-based methods distinguishing pet motion and human motion mainly rely on heat maps obtained from thermal sensors [20] or image/face recognition techniques using cameras [21]. These methods have limited coverage, and are often restricted to LOS, and camera-based methods further introduce privacy concerns [22].

As for RF-based methods, recent attempts to differentiate between humans and pets have employed radar [23] and ultra-wideband (UWB) [24], focusing on analyzing biometric indicators

such as heartbeat and respiration. But these methods necessitate the pet to remain stationary during recognition and impose constraints regarding their spatial position and distance from the device, thereby rendering them impractical and ineffective for identifying moving subjects. While [25] enables the recognition of canine movement distinct from human movement through radar, it demands that the device be installed on the ceiling, and restricts the movement of humans and pets to a limited area under the device. Furthermore, the method presented in [26] can differentiate fan motion from human respiration, yet it remains inapplicable to human activities like walking and running.

WiFi-based sensing systems are attractive for indoor activity recognition and monitoring due to their ubiquity and low cost [27,28]. Numerous WiFi-based systems have been designed to monitor human motion and the environment. WiFi-based motion detection systems [29,30] detect the occurrence of human motion within an environment, while WiFi-based activity recognition methods [31–35] attempt to recognize human daily activities like running, walking, and sitting. Identification systems like [36–39] recognize human identity through gait or other biometric features extracted from WiFi signals. In addition, mapping and tracking systems that leverage WiFi [40–43] plot human movement paths and generate floor maps. Nonetheless, these systems fail to account for the influence of non-human entities, like pets and vacuum robots, on their operational accuracy. These non-human subjects, capable of affecting WiFi signals similarly to humans, introduce a potential source of interference. Consequently, these systems' reliability for indoor monitoring and recognition can be compromised, particularly when non-human subjects are present in the environment.

Some researchers have proposed to detect intrusion using WiFi. Among the proposed systems, PerFree [44] and M-WiFi [45] proposed methods to reduce false alarms caused by

pets. For example, PetFree distinguishes pets from humans by mapping the subject's effective interference height (EIH) to CSI measurements. This method, however, requires the transmitter and receiver to be positioned at the same height and it operates solely under LOS conditions. Moreover, its performance can be affected by the multipath effect in environments with complex layouts. M-WiFi differentiates between pets and humans by examining the RF disturbance. However, this system encounters issues when a pet moves close to the wireless link. This movement can induce signal fading akin to that of a person moving farther from the wireless link, resulting in a significant number of false alarms, regardless of differences in RF disturbance.

Unlike the existing RF-based approaches, we have no restriction on pet movement, device placement, or environment complexity. Notably, most existing works only consider the interference of pets, and few tackle the motion of indoor robots and household appliances. In this paper, we systematically differentiate various typical non-human motions from human motions.

1.2.3 Indoor Intrusion Detection

Human detection systems not based on Radio Frequency (RF) primarily utilize technologies such as video cameras [21, 46], sound [47, 48], and Infrared sensors [49, 50]. Though camera- and sound-based detection systems can effectively identify intruders in an environment, their practicality as ideal intrusion detection systems is marred by privacy concerns. Furthermore, camera-based detection devices demand high environmental lighting conditions and struggle under Non-Line-of-Sight (NLOS) scenarios.

Infrared intrusion detection devices, due to their relatively better privacy protection and cost-effectiveness, are quite popular. However, these devices come with their own set of

limitations. They require precise positioning, offer smaller coverage areas, and fail under NLOS conditions. Most importantly, their detection accuracy can be influenced by environmental temperature, and they lack the ability to distinguish between human and non-human targets like pets.

As for RF-based Intrusion Detection, there has been an uptick in indoor intrusion detection projects using devices like millimeter-wave radars recently. While these devices have superior accuracy and lower false-positive rates compared to infrared-based devices, their high cost and extremely limited coverage area, along with inability to operate under NLOS conditions, pose significant limitations.

As IoT devices and WiFi become increasingly ubiquitous, a surge in the development of WiFi-based sensing systems has been observed [51, 52], thanks to their omnipresence and cost-effectiveness. Of late, numerous studies have attempted to leverage WiFi for human detection [29, 53–55]. However, these systems fail to eliminate false alarms triggered by non-human movements in the environment, such as pets, cleaning robots, fans, etc.

M-WiFi [45] proposed to distinguish pet from human based on the difference of the RF disturbance, but it failed to reduce false alarm rates, as pets approaching the wireless connection could produce similar interference to humans distancing themselves from it. PetFree [44] introduced a method to eliminate pet interference but required all equipment to be placed at the same height and its performance decreases when applied to in complex environments. Moreover, these existing approaches only accounted for pet interference, incapable of excluding disturbances from other non-human entities like cleaning robots or moving household appliances.

Our previous work [56] proposed an SVM-based method to differentiate various non-human targets from humans, offering a more comprehensive way to discern between human and

non-human movements in the environment. However, as a classification methodology, this work did not consider the temporal correlation of targets' appearances in the environment. In this paper, we leverage a more robust deep learning model, designed specifically for WiFi-based intrusion detection, and combine both current target detection results with historical information, enabling an intrusion detection system capable of eliminating interference from various non-human targets in the environment.

1.3 Dissertation Outline and Contributions

In this dissertation, we initially lay out the fundamentals of WiFi sensing in Chapter 1.4, encompassing the Received Signal Strength Indicator (RSSI) and Channel State Information (CSI). Following this, in Chapter 2, we delineate a method for automatic floor plan construction. Chapters 3 and 4 respectively detail a Support Vector Machine (SVM) based method for distinguishing human and non-human motion using edge computing, as well as a framework for human and non-human subject recognition with deep learning methods. Chapter 5 introduces a robust indoor intrusion detection system resilient to non-human movement, underpinned by deep learning. Finally, in Chapter 6, we conclude the dissertation and discuss the future work.

1.3.1 Automatic Floor Plan Construction (Chapter 2)

In this chapter, we present *EZMap*, a high-accuracy, low-cost floor plan construction system that fuses RF and inertial sensing. *EZMap* combines the fine-grained yet local information from RF tracking with the coarse-grained but global contexts from inertial sensing (e.g., magnetic field strength), which together makes for an accurate map. Our system employs a robot for trajectory

collection and requires only a single Access Point to be arbitrarily installed in the space, both of which are widely available nowadays. Furthermore, it can generate a map even only a small amount of data is available, allowing it to scale for different buildings like malls, office buildings, and homes with little cost. We validate the performance using a Dji RoboMaster S1 robot with commodity WiFi in three different buildings. The results show that our system can efficiently generate faithful maps for the targeted areas. With the ubiquity of WiFi infrastructure and the rise of home robots, we believe our approach will pave the way for pervasive indoor maps services.

1.3.2 Human and Non-human Motion Discrimination (Chapter 3)

In this chapter, we propose a novel system, *Wi-MoID*, that passively and unobtrusively distinguishes moving human and various non-human subjects using a single pair of commodity WiFi transceivers, without requiring any device on the subjects or restricting their movements. *Wi-MoID* leverages a novel statistical electromagnetic wave theory-based multipath model to detect moving subjects, extracts physically and statistically explainable features of their motion, and accurately differentiates human and various non-human movements through walls, even in complex environments. In addition, *Wi-MoID* is suitable for edge devices, requiring minimal computing resources and storage, and is environment-independent, making it easy to deploy in new environments with minimum effort. We evaluate the performance of *Wi-MoID* in four distinct buildings with various moving subjects, including pets, vacuum robots, humans, and fans, and the results demonstrate that it achieves 97.34% accuracy and 1.75% false alarm rate for identification of human and non-human motion, and 92.61% accuracy in unseen environments without model tuning, demonstrating its robustness for ubiquitous use.

1.3.3 Deep Learning for Human and Non-human Motion Identification (Chapter 4)

In this chapter, we design a deep learning framework to recognize human and non-human subject with WiFi signals through the wall. Our system extract environment independent features from single-link WiFi. We investigate the performance of popular deep neural networks and explore the efficacy of transfer models trained with image dataset on WiFi sensing tasks. We implement our framework and evaluate the performance in five environments with commodity WiFi devices. With a challenging dataset considering large pets and multiple subjects coexiting cases, our proposed framework achieves an average validation accuracy of 95.84% and an average testing accuracy of 91.71% in unseen environments without further training or parameter tuning. These results underline the robustness of our approach and its readiness for integration into ubiquitous intelligent IoT systems and applications.

1.3.4 Through-the-wall Indoor Intrusion Detection (Chapter 5)

In this chapter, we propose *Wi-IntruNet*, the first robust through-wall indoor intrusion detection system that mitigates the interference from non-human indoor objects based on widely available WiFi signals. *Wi-IntruNet* introduces a novel deep learning framework for single-link WiFi signal analysis, employing a ResNet model for extracting features of indoor moving targets and an LSTM to incorporate temporal data for efficient intrusion detection. Notably, *Wi-IntruNet* is invariant to environmental changes, angle, and position, enabling swift deployment in new environments without additional training. We implement and extensively evaluate *Wi-IntruNet*

with commodity WiFi devices in 5 typical indoor environments with various interference from pets, cleaning robots and fans. The results revealed that *Wi-IntruNet* achieved an average of 98.91% intrusion detection accuracy and 2.84% false alarm rates in unseen environments without model tuning, underscoring its robustness and ready for ubiquitous indoor security applications

1.4 Preliminaries on Wireless Sensing

Wireless sensing represents a pivotal advancement in the field of remote detection and measurement technologies. It leverages the pervasive and invisible radio waves to detect, identify, track, and understand phenomena within their range without requiring a physical connection or proximity. One notable subset of this technology is WiFi sensing, which utilizes the omnipresent WiFi signals as a source of information to enable context-aware applications. WiFi sensing is capable of detecting minute changes in the wireless signals caused by various objects and movements in the environment, providing rich information about the surroundings. This transformative technology not only offers an innovative way to interact with the environment but also opens up a plethora of opportunities for applications in various sectors such as healthcare, security, home automation, and more.

5 GHz WiFi, part of the dual-band WiFi capabilities present in many modern devices, provides a faster, less congested frequency range compared to its counterpart, 2.4 GHz WiFi. This frequency band not only allows for greater data transmission speeds but also supports a wider range of channels, reducing the potential for interference from other devices. When utilized for WiFi sensing, the 5 GHz spectrum can offer improved performance and accuracy due to its characteristics. WiFi sensing on this band leverages the scattering, reflection, and absorption

of the 5 GHz signals by objects and people in the environment to create a rich tapestry of data about the surroundings. This information can be analyzed and interpreted to enable a range of applications, such as detecting movement or occupancy in a room, monitoring vital signs for healthcare, or even identifying gestures for interaction with smart devices. The use of 5 GHz WiFi sensing thus combines the benefits of higher frequency wireless communication with the growing potential of context-aware technologies.

1.4.1 Received Signal Strength Indicator

Received Signal Strength Indicator (RSSI) is a metric used in wireless communications to measure the power level received by a device from a source transmitter. In an IEEE 802.11 system, RSSI is the relative received signal strength in a wireless environment. RSSI is an indication of the power level being received by the receiving radio after the antenna and possible cable loss. A higher RSSI value denotes a stronger received signal. This metric is crucial for various applications, including network diagnostics, link quality estimation, and location-based services. Factors affecting RSSI include the distance from the transmitter and physical obstructions.

RSSI plays a vital role in various WiFi sensing applications. One of the primary applications of RSSI in WiFi sensing is indoor positioning. By measuring the RSSI values from multiple access points, a device can estimate its location based on signal strength triangulation. The stronger the signal from a particular access point, the closer the device is likely to be to that access point. Beyond positioning, RSSI also aids in detecting movement, gauging link quality, assisting in network management, and offering insights into human-device interactions.

While RSSI offers a singular measure of overall signal strength, indicative of the total received power from a transmitter, there are some cases—such as motion detection, through-wall sensing, gait recognition—where finer granularity is required to capture the intricate nuances of the wireless environment. Therefore, Channel State Information (CSI) comes into play, providing detailed information on the channel's response across different frequencies. CSI captures the amplitude and phase of each signal path, allowing for a deeper understanding of environmental dynamics. In scenarios demanding precision and comprehensive insights into the wireless channel, CSI is the preferred choice over RSSI.

1.4.2 Channel State Information

CSI represents an intricate matrix containing the properties of the wireless communication channel. It captures comprehensive information regarding the propagation path between each transmit-receive antenna pair, including the path loss, multipath fading, phase shift, and Doppler effect due to relative movement. The CSI thus provides a detailed spectral view of the WiFi signal as it traverses through, and interacts with, the complex wireless medium. The CSI is often calculated as the frequency response of the channel, which can be modeled as

$$H(t, f) = \sum_{l=1}^{L} a_l(t) \exp\left(-j2\pi f \tau_l(t)\right)$$

where $a_l(t)$ and $\tau_l(t)$ denote the complex amplitude and propagation delay of the l-th multipath component (MPC), respectively, and L stands for the number of MPCs.

Due to the timing and frequency synchronization offsets and additive thermal noise, the

real measurement of CFR $\tilde{H}(t,f)$ is expressed as

$$\tilde{H}(t,f) = \exp(-j(\alpha(t) + \beta(t)f))H(t,f) + n(t,f),$$

where $\alpha(t)$ and $\beta(t)$ are the random initial and linear phase distortions at time t, respectively. Define the channel power response $G(t,f)$ as the square of the magnitude of $\tilde{H}(t,f)$

$$\begin{aligned} G(t,f) &\triangleq |\tilde{H}(t,f)|^2 \\ &= |H(t,f)|^2 + 2\,\mathrm{Re}\left\{n^*(t,f)H(t,f)\right. \\ &\left.\exp(-j(\alpha(t) + \beta(t)f))\right\} + |n(t,f)|^2 \\ &\triangleq |H(t,f)|^2 + \varepsilon(t,f), \end{aligned} \tag{1.1}$$

where the superscript * denotes the operator of complex conjugate, the operator Re denotes the real part of x, and $n(t,f)$ is defined as the noise term, which can be approximated as additive white Gaussian noise (AWGN) with variance σ_n^2 and is statistically independent of $H(t,f)$ [57].

Chapter 2: Automatic Floor Plan Construction

Location-Based Services (LBS) are gaining increasing popularity, such as navigation, location-based searching, and social network services, etc., thanks to the ever-growing presence of smart devices with built-in location systems and maps. However, most of these services are only available in outdoor environments. One of the most critical constraints to the development of indoor LBS is the lack of digital indoor maps [58].

Recent advances in RF-based tracking enable an opportunity for this breakthrough [59–62]. Notably, a recent work RIM [63], an RF-based Inertial Measurement system, has enabled centimeter-level distance tracking on commodity WiFi in both Line-of-Slight (LOS) and Non-LOS (NLOS) conditions with only a single Access Point. RIM allows us to collect high-accuracy trajectories with commodity WiFi clients (e.g., home robots) and further construct precise floor plans from the gathered trajectories. There are multiple advantages if we could leverage this opportunity for floor plan construction: It only uses commodity WiFi, particularly without expensive hardware like lidar or privacy-intrusive modules like cameras. It promises high accuracy for broad coverage with only one AP, including both LOS and NLOS areas throughout the space being covered by the WiFi signals [64]. Moreover, different from previous works that assume a number of mobile users to contribute a lot of data, we could achieve high-quality construction with a small amount of data efficiently collected by a single home robot.

However, leveraging RF-based tracking like RIM for high-precision floor plan construction entails great challenges. First, RIM only provides distance estimation (using a linear array available on most devices). To recover a trajectory, we still need the direction information, for which we can only utilize the inaccurate inertial sensors. Second, despite its high moving distance estimation accuracy, the RF-based tracking lacks global reference information, a necessity to connect different trajectories and construct a floor plan. And it becomes very difficult to obtain reliable global reference information while there is only one single AP in the environment [65]. Third, given the imperfect information, it is non-trivial to fuse trajectories collected by a robot to recover a floor plan.

In this chapter, we overcome the above challenges and propose *EZMap*, a novel system to boost the automatic construction of indoor floor plans. *EZMap* employs a commodity home robot, equipped with WiFi and inertial sensors to roam around the environment and collect data. It requires only a single AP and can map not only public spaces such as malls and office buildings but also home environments that are barely explored by existing works. The reconstructed map represents hallway structure, room layout, as well as their sizes (i.e., corridor widths and room sizes).

This chapter is organized as follows. Section 2.1 elaborates on the system design and detailed solutions. Section 2.2.3 shows the implementation, experiments, and evaluation results. In Section 2.3, we conclude this chapter.

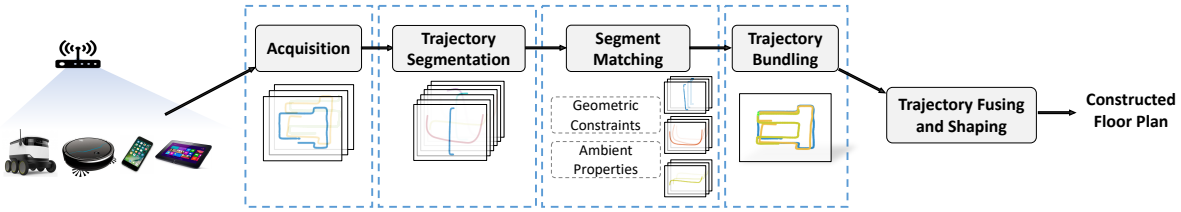


Figure 2.1: System overview.

2.1 System Design

2.1.1 System Overview

In this part, we present an overview of *EZMap* system as Fig. 2.1 shows. The traces can be collected by humans or robots with one AP deployed in the environment. The traces contain the distance information obtained by RIM and direction information derived from inertial sensors. RSSI of the AP and the magnetic field strength (MFS) are also recorded. Then, the collected trajectories, which could be of arbitrary lengths, are divided into short segments, named as *atomic segments*, to overcome the potential errors accumulated in orientation while leveraging the accurate distance estimation (*Trajectory Segmentation*). The atomic segments are then clustered by their intrinsic geometric constraints as well as the accompanying time series of RSSI and MFS (*Segment Matching*). The clustered segments are then positioned by bundling the long trajectories and thereby inferring their relative positions (*Trajectory Bundling*). Finally, all trajectories are fused, taking their original shapes into account, to output the reconstructed map with corridor widths and room sizes (*Trajectory Fusion and Shaping*).

(a) Cart. (b) Robot.

Figure 2.2: Crowdsourcing trajectory collection devices.

2.1.2 Trajectory Acquisition

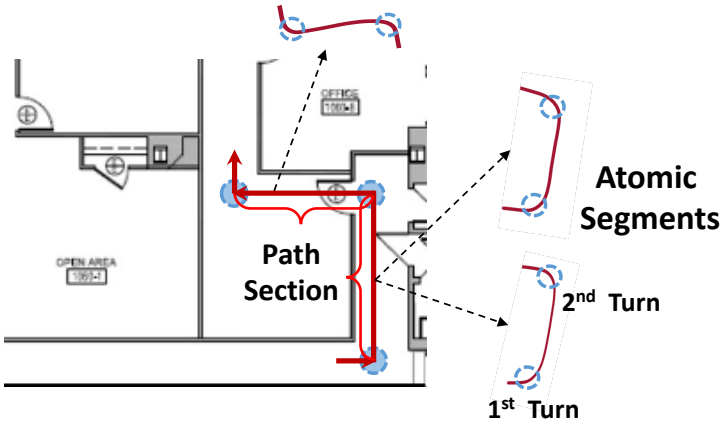
The crowdsourcing trajectory tracking is based on RIM [63] for distance estimation and inertial sensors for orientation estimation. RIM is good at estimating the moving distance of wheeled platforms and robots [40]. Leveraging multipath profiles as virtual antennas with a super-resolution virtual antenna alignment algorithm, RIM achieves centimeter accuracy in moving distance over a multipath-rich area. RIM needs only a single AP, and captures Channel State Information (CSI) from data packets for its moving distance calculation. *EZMap* relies on RIM's algorithm to estimate the moving distance with high accuracy while using inertial sensors to measure the turning angles and heading information, which together shape the geometric properties. The device setup can be found in Fig. 2.2, where we can use a pushing cart in Fig. 2.2a and a Dji robot in Fig. 2.2b. In both cases, the blue bot is customized hardware with a commodity WiFi chipset and IMU on it. We use a customized hardware because CSI extraction is not yet ready on Dji robot. *EZMap* itself is a software solution making no changes to hardware.

2.1.3 Trajectory Segmentation

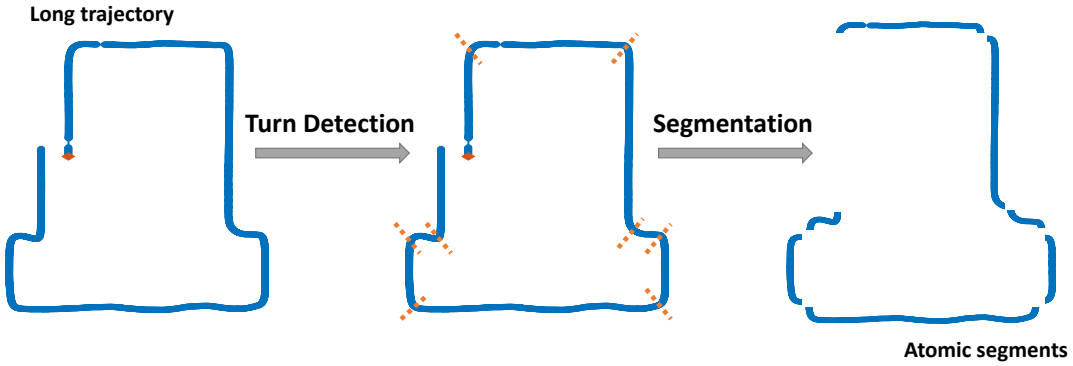
The acquired trajectories come in various lengths from meters to tens of meters or even longer. *EZMap* prefers long trajectories in general, as they likely cover and connect wider areas and thus contain more information to be uniquely identified. However, long trajectories suffer from large accumulative errors particularly in orientation. To overcome the issues, we propose to decompose all collected trajectories into short pieces in a novel form of atomic segments that better preserve the accurate geometric shape information.

We divide each long trajectories into the structures that consist of a straight segment with two turns on the two ends, as shown in Fig. 2.3(a). We name such a structure an atomic segment and use it as the basic unit of trajectories in *EZMap*. Each atomic segment contains distance information of the path section and angle information of two turns at its two ends. Such atomization can be accomplished by a simple turn detection based on inertial sensor readings. In *EZMap*, we calculate the change rate of angle (the first-order reciprocal of angle to time) of accelerometer and gyroscope data to detect turns. The procedure of trajectory segmentation is shown in Fig. 2.3(b).

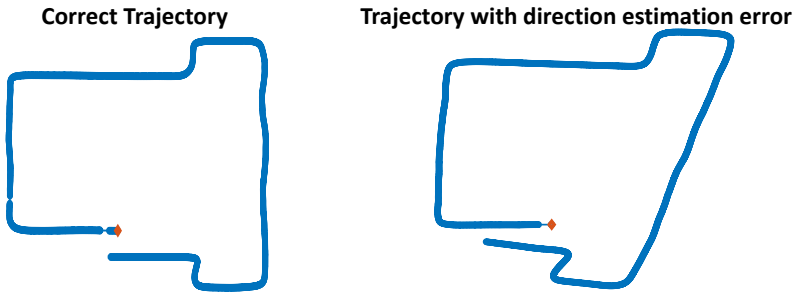
The design of atomic segment is inspired by the below insights: First, the atomic segments disconnect the original trajectories at each turn, because orientation errors easily accumulate during a turn (as seen in the example in Fig. 2.3(c)). As a result, it preserves the accurate distance information while resets significant orientation errors before they could accumulate over multiple turns. Second, the two turns are kept as parts of each atomic segment because they provide more information to cluster the segments that have similar lengths. Taking the segments in Fig. 2.3(a) as an example, two atomic segments, ending with a left turn and a right turn respectively, can



(a) Example atomic segments.



(b) Procedure of trajectory segmentation.



(c) Errors accumulated at turns.

Figure 2.3: Atomic segments.

be separated even they have the same length. Third, considering real-world building layouts, such atomic segments can be frequently obtained. And as will be demonstrated later, the global location information (i.e., RSSI and MFS time series) associated with such segments suffice to

uniquely cluster and reconnect them, be jointly considering the precise geometric shape of the segments.

Note that some trajectories will not be segmented by the atomization. For example, if the target circles around and moves along a curved pathway, the resultant trajectory would be continuously "turning" and will never be cut. These curved segments are still useful, as they possibly provide information about some open spaces and/or rooms, and will be handled separately in area shaping, as in Section 2.1.6.2.

2.1.4 Segment Matching

To generate an accurate floor plan with tracking traces, we need to accurately estimate the relative position of each trajectory. However, it is very challenging because we do not have global reference or the start point of each trajectory. As we discussed in the previous section, we divide the long trajectories into atomic segments and resort to determining the relative position of each atomic segment. To accurately predict the relative position of each segment, we take two steps. We first cluster the atomic segments belongings to the same location. Then the atomic segments in the same cluster are positioned and bundled by inferring its relative position from the long trajectory. In this section, we will focus on the first step, that is identifying the segments at the same location and distinguishing those at different locations.

We present a trajectory matching algorithm to recognize the segments belonging to the same location and distinguish those belonging to different locations by their geometric shapes and time series of RSSI and MFS. The matching process is shown in Fig. 2.4.

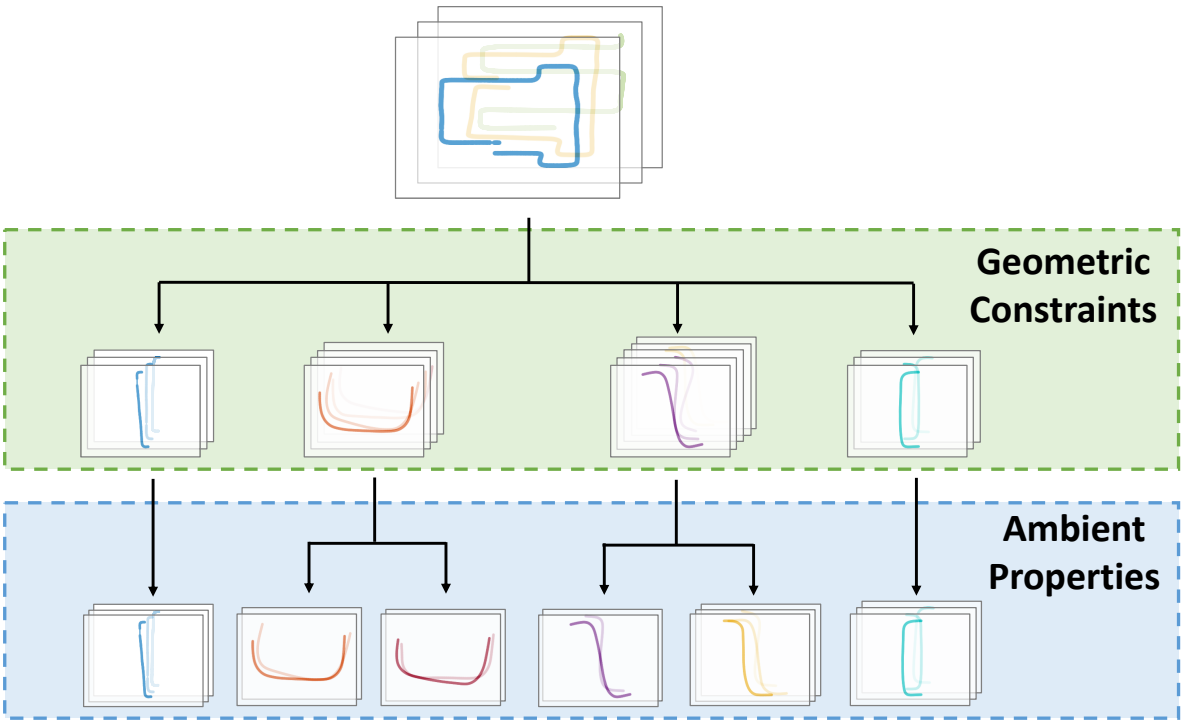


Figure 2.4: An example of matching process.

2.1.4.1 Geometric Constraints

With the unique structure that we designed for segments, each atomic segment has a geometric shape which contains the angle information of two turns and distance information of the path section in between. Atomic segments related to same path show similar geometric shapes. Geometric constraints are designed to sort the segments with similar shapes together based on their geometric shape information. Example of atomic segments belonging to the same path are shown in Fig. 2.3(a).

The constraint of geometric shapes has two parts, distance constraint and angle constraint. As for the distance constraint, based on experimental observations and the distance tracking accuracy of the tracking client, we limit the distance difference between segments on the same

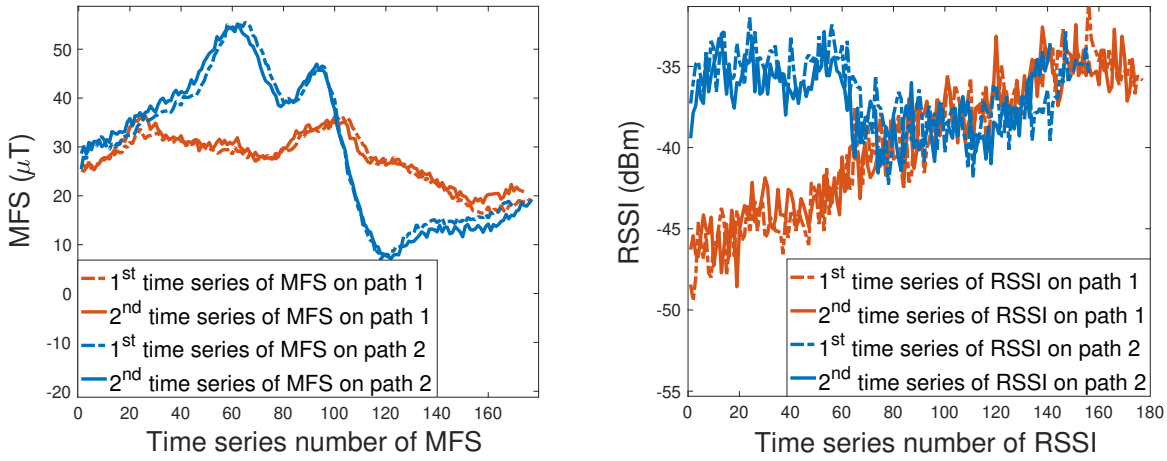
path to be no more than 3 m. In other words, if the difference of estimated distances of two path sections are greater than 3 m, these two segments are unlikely to be at the same location. As for the angle constraint, considering certain angle errors, the angle difference between the same turn is limited to be no more than 30 degrees. We choose 30 degrees because it has tolerance for the orientation estimation error from IMU, but will not match the trajectories of different turns into one category. If the constraint is too strict, the segments of the same path may not be matched together as they have orientation deviations caused by IMU. If the constraint is too loose, the segments with different turning angles would be mistakenly matched.

Noted that atomic segments from the same location with different turnings (left and right) would not be clustered together by this method. But this will not affect the final reconstructed map because they would be recognized and merged together with trajectory fusion later at Section 2.1.6.1.

To verify the effectiveness of the geometric constraints, we conduct a comparative experiment where over 200 atomic segments are used to compare the matching accuracy with and without geometric constraints. The experimental results validate that the trajectory constraint can effectively improve the accuracy of segment matching, which will be shown later in Section 2.2.3.1.

2.1.4.2 Ambient Properties

The geometric constraints alone are insufficient to support segment matching with high accuracy since it cannot separate the segments from different paths but having similar geometric shapes. Therefore, a robust global reference is crucial to the matching accuracy and the



(a) Time series of MFS.

(b) Time series of RSSI.

Figure 2.5: Time series of MFS and RSSI on two path.

construction performance. In *EZMap*, we leverage the time series of MFS and RSSI along the segment, denoted as *ambient properties* for global reference information and propose a robust algorithm to match them.

As described in CrowdMap[8], the magnetic field anomalies caused by ferromagnetic construction materials (e.g., reinforcing steel bars) on the indoor path are generally stationary and unique over time. The time series of MFS are collected when a user or a robot walks along a specific indoor path and are used as the representative feature of the corresponding path. As shown in Fig. 2.5(a), the time series of MFS data along the same indoor path is similar, while those along different indoor paths differ considerably. Similar characteristics also appear on the time series of RSSI values measured for a single AP, as shown in Fig. 2.5(b). Although RSSI values at a single location suffer from very limited space resolution, a series of RSSI along a path provide more distinctive information.

A cluster can be generated when the similarity value of two atomic segments based on the time series of MFS and RSSI is below a threshold. A similarity vector can be calculated for an

atomic segment, measuring the similarity between itself and each existing cluster of the atomic segments. For an atomic segment j, we apply dynamic time warping (DTW) [66] to calculate its RSSI similarity vector $D_{j,R}$ and MFS similarity vector $D_{j,M}$, respectively. $D_{j,R}$ and $D_{j,M}$ are normalized with min-max normalization and summed together to obtain the similarity vector as follows:

$$D_j = \overline{D_{j,R}} + \overline{D_{j,M}} = [d_{j,1}, d_{j,2}, ..., d_{j,i}, ..., d_{j,N}], \tag{2.1}$$

where $d_{j,i}$ denotes the similarity value between atomic segment j to the i^{th} cluster, and N denotes the total number of clusters. If $d_{j,i}$ is below a pre-defined threshold (e.g., 0.2), this atomic segment is clustered to the i^{th} cluster and can be matched with other segments from the i^{th} cluster. If an atomic segment cannot be matched with any existing cluster, we generate a new cluster for it. Based on our experimental observations, two segments with a similarity value below 0.2 can be considered matched. As we have normalized the RSSI vector and MFS vector during fusion, the threshold can be generalized to various environments.

In some corner cases, the threshold may not work well. For example, when: 1) the similarity value of two matched segments (i.e., two segments from the same location) is greater than 0.2, and 2) there are multiple similarity values lower than 0.2 but only one segment should be matched. In the first case, although the segments are mistakenly treated as belonging to two clusters, we will be able to merge them together by trajectory fusion in Section 2.1.6.1. As for the second case, we will match the target segment with the most similar one, meaning the one with the smallest similarity value. The threshold is tested to be valid in our experiments in three scenarios as will be shown in Section 2.2.

Note that for the matched segments within the same cluster, we not only know that they

should come from the same location but also have the alignment information between each pair of the segments, e.g., we have information about how each turn on the atomic segment corresponds to each other on the other matched segments.

2.1.5 Trajectory Bundling

In the previous section, we identify the segments belonging to the same location. To reconstruct an accurate floor plan with these segments, we also need to identify their relative positions. In this section, a trajectory bundling algorithm is designed to determine the position of atomic segments.

The long trajectories are bundled leveraging their matched atomic segments. For every two long trajectories that have matched segments, they are bundled in the following two steps: 1) The first turning points of two matched segments are stitched together. 2) The long trajectories are rotated so that the direction of the path section of the two matched atomic segments are consistent. Every long trajectory is bundled to others via aligning their matched segments. After long trajectories are bundled, we can determine the coordinates of atomic segments by utilizing their spacial relationship on their original long trajectories. An example of trajectory bundling is shown in Fig. 2.6, where two trajectories are bundled by aligning their matched segments.

However, as we treat the long trajectory as a rigid body for trajectory bundling, the long trajectories containing large accumulative direction errors, as shown in Fig. 2.3(c), can reduce the accuracy of the generated map. In the next section, we will adjust the positions of atomic segments on the long trajectory to avoid the impact of accumulated errors on the generated map.

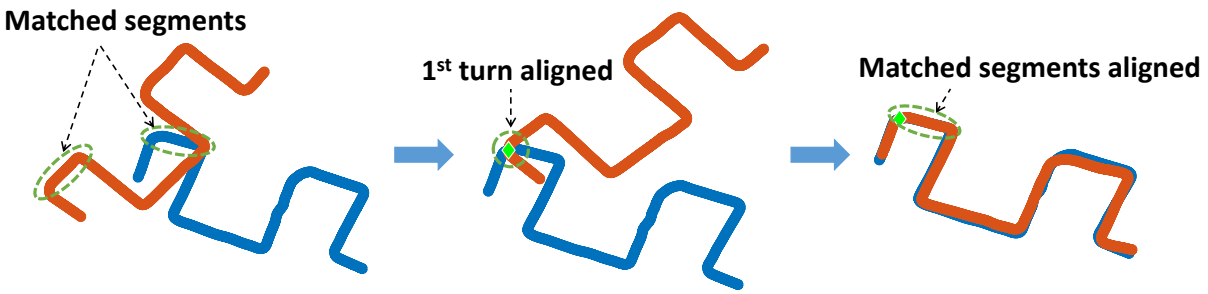


Figure 2.6: Bundling process.

2.1.6 Trajectory Fusion and Area Shaping

2.1.6.1 Trajectory fusion

Although the rough position of atomic segment can be inferred by bundling long trajectories, atomic segments belonging to the same path have severe coordinate deviation, as shown in Fig. 2.7, which is undesirable. Hence, we propose a trajectory fusion algorithm to generate a more accurate and well-shaped map next.

Leveraging the characteristics of matched segments, we fuse matched atomic segments in two steps. First, for matched atomic segments at the same path, the inner-cluster constraints, including angle and positions of their turns, are utilized to update their positions. We calculate the medium coordinate of their endpoints at the same turn and update the coordinate of each endpoint by

$$\begin{aligned} New\_location =& (1-b)^{*} cluster\_medium\_location \\ &+ b^{*} original\_location, \end{aligned} \tag{2.2}$$

where $b \in (0, 1)$ is a parameter balancing the cluster median location and original location.

Then, the intra-cluster global information can be leveraged to merge the turns that are

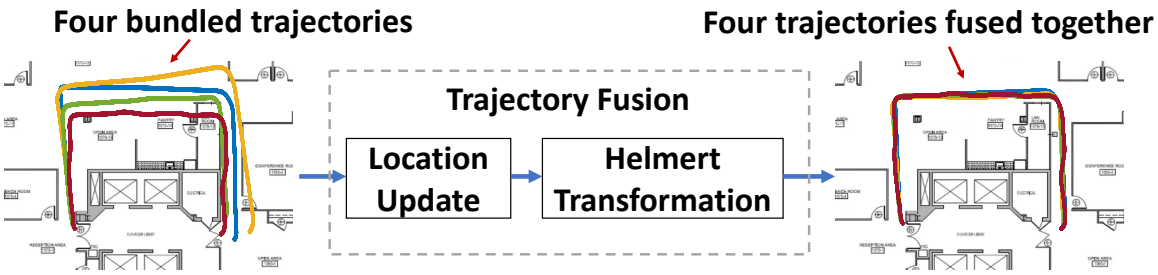


Figure 2.7: Trajectory fusion process.

mistakenly separated, which would happen when atomic segments on the same path are classified into different clusters, as discussed at the end of Section 2.1.4.2. DTW can be applied again to merge the mistakenly separated turns. After the trajectories are bundled, we search for the turns within a small area. Different from last time that we applied DTW on all pairs of atomic segments, this time we only apply the DTW on the turns close to each other. If the DTW distance of MFS and RSSI of two turns are below a threshold (e.g. 0.05 according to our experiments), the two turns are considered to be at the same location and merged together. Thus, the mistakenly separated segments belonging to the same path are correctly merged.

Finally, with the new coordinates of two endpoints, each atomic segment is transformed to its new position by Helmert transformation [67], a similarity transformation frequently used in geodesy to produce datum transformations between datums. Two dimensional Helmert transformation is performed here. With the knowledge of the original coordinate of two turning points and the updated coordinates of each points on the segments, Helmert Transformation calculates the updated coordinates of each point on the segments. The transformed segments preserve the shape and 2D information of the original atomic segments. Fig. 2.7 shows an example of trajectory fusion process, where Helmert Transformation is performed on four trajectories.

2.1.6.2 Area Shaping

When segmenting the long trajectories, there are atomic segments without straight path section in between. Instead, they contain curved sections with continuous "turnings". While the straight segments, as discussed before, reflect the major layout of the building, the curved segments also contain useful information about rooms, curved corridors, and open spaces. Therefore, we separately process them to add more details to the recovered floor plan.

The position of the curved segment is deduced from its spatial relationship with straight segments. We first find out straight segments that directly connect with the curved segment. Then, the positions of straight segments are utilized to estimate the position of the curved segment and Helmert transformation is then performed on the curved segment. This process is shown in Fig. 2.8. Without losing the shape information, curved segments are grouped with straight segments to reconstruct the curve corridors, rooms and open spaces. Hence, the map construction algorithm can not only generate straight corridors, but also estimate the location and area of the rooms, making the floor plan more detailed and comprehensive.

Finally, to provide a well-shaped floor plan, we use alpha-shape [68] method to extract the contour of trajectories, which creates a bounding area that envelops the set of points on all trajectories. Specifically, it finds a convex hull using Delauney Triangulation and then deletes some triangle borders according to parameter α to make a better fitting hull.

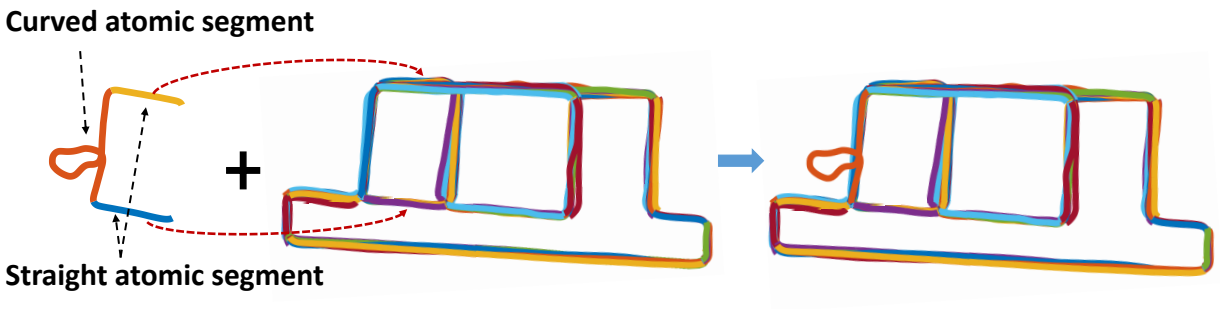


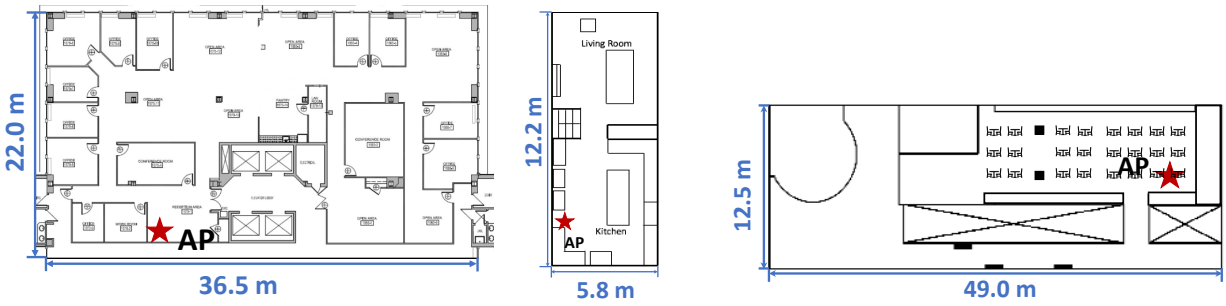
Figure 2.8: Curved trajectory positioning process.

2.2 Experiment

2.2.1 Experimental Setup and Data Collection

The proposed algorithm is evaluated in three scenarios: Scenario I is an office with an area of 22.0 m $\times$ 36.5 m, which consists of corridors, rooms, doors, and elevators as shown in Fig. 2.9(a); Scenario II is the second floor of a townhouse with an area of 5.8 m $\times$ 12.2 m, consisting of a kitchen and a living room as shown in Fig. 2.9(b); and Scenario III is a dining court of a campus building with an area of 49.0 m $\times$ 12.5 m, which consists of a dining hall and hallways around it, as shown in Fig. 2.9(c). The goal is to reconstruct the detailed floor plan using the proposed system.

In Scenario I, the tracking device is installed on a cart pushed by human as shown in Fig. 2.2(a). Traces are collected on 12 different days in 5 months, with a total of 64 crowdsourcing trajectories collected. Since the different turning directions are considered, there are 18 clusters in this data set. In Scenarios II and III, the tracking device is equipped on a robot, Dji RoboMaster S1, as shown in Fig. 2.2(b), which collect 49 trajectories with 8 clusters over 16 different days in 5 months for Scenario II, and 54 trajectories with 19 clusters on 7 different days in 2 months



(a) (b) (c)

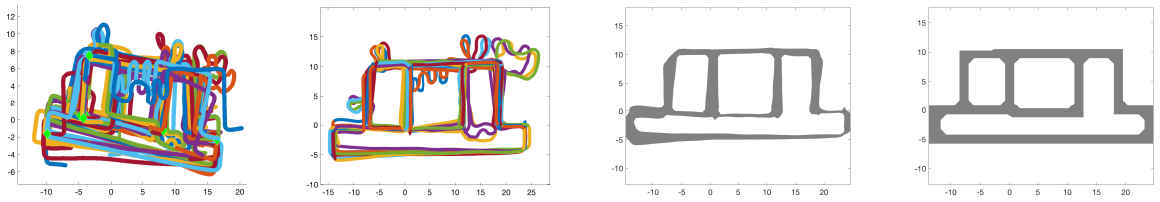
Figure 2.9: Ground truth floor plans of (a) office, (b) home, and (c) campus court.

for Scenario III. We collect data over a long course to validate the robustness of *EZMap* over environmental dynamics, while we only obtain a small amount of trajectories to demonstrate the effectiveness of the proposed system.

EZMap employs a commodity home robot to collect data because it is labor-saving and efficient. Moreover, home robots are widely available nowadays. However, *EZMap* is not limited to high-precision robotic tracking based on RIM. Our proposed map construction algorithm is also applicable to high-precision pedestrian tracking, such as WiBall [69].

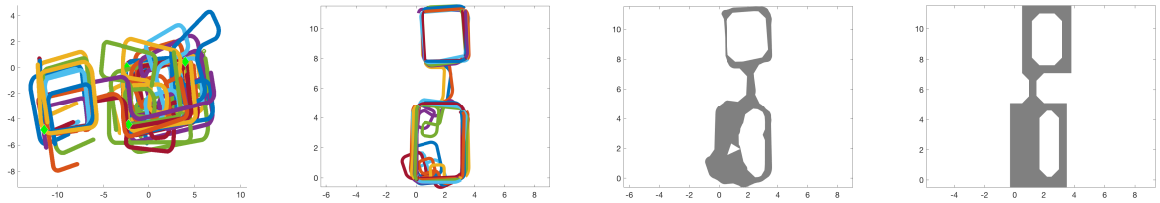
2.2.2 Reconstructed Floor Plans

The results of the three scenarios are presented in Fig. 2.10, Fig. 2.11, and Fig. 2.12, respectively. For all three scenarios, the first subfigure (i.e., Fig. 2.10(a), Fig. 2.11(a), and Fig. 2.12(a)) shows the bundled trajectories with color lines. The reconstructed floor plan of Scenario I is shown in Fig. 2.10(b), where we can see the accessible area of rooms and open space are well reconstructed with curved trajectories. The reconstructed hallway plan of Scenario I is shown in Fig. 2.10(c), while the ground-truth hallway plan extracted by alpha-shape is shown



(a) (b) (c) (d)

Figure 2.10: Construction result of Scenario I. (a) Trajectories bundling result. (b) Reconstructed floor plan. (c) Alpha-shape of reconstructed floor plan. (d) Alpha-shape of ground-truth floor plan.



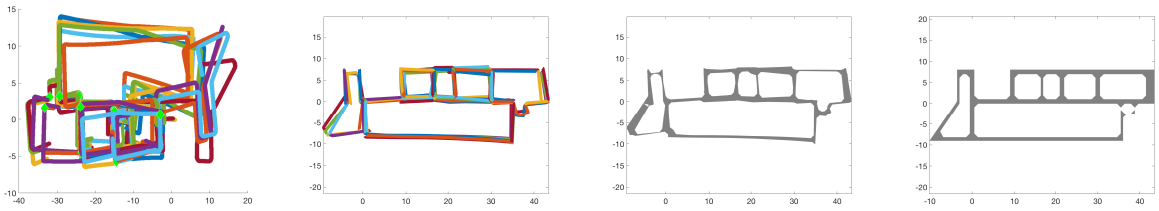
(a) (b) (c) (d)

Figure 2.11: Construction result of Scenario II. (a) Trajectories bundling result. (b) Reconstructed floor plan. (c) Alpha-shape of reconstructed floor plan. (d) Alpha-shape of ground-truth floor plan.

in Fig. 2.10(d). In Scenario II, the reconstructed floor plan is shown in Fig. 2.11(c), where the open space in Kitchen is well reconstructed by our system. Without the need of anchor points required by most related works, our system demonstrate the capability to reconstruct the floor plan for not only large and public areas, but also small and private environments like homes. Fig. 2.12(c) shows the well reconstructed floor plan of Scenario III. The result shows our system can accurately reconstruct the floor plan for a variety of areas with only a single AP.

2.2.3 Performance Evaluation

The matching performance, room size accuracy, hallway shape accuracy and the construction efficiency of *EZMap* are evaluated. As for hallway shape and room size, we compare our system with SenseWit [6] and CorwdInside [7] using the evaluation result reported



(a) (b) (c) (d)

Figure 2.12: Construction result of Scenario III. (a) Trajectories bundling result. (b) Reconstructed floor plan. (c) Alpha-shape of reconstructed floor plan. (d) Alpha-shape of ground-truth floor plan.

in [6]. We do not implement the two systems for comparison because the prerequisites make them inapplicable to our settings.

2.2.3.1 Matching performance

The matching performance is evaluated by purity measure and Normalized Mutual Information (NMI) score. The purity score s_{c} is defined as

$$s_{\mathrm{c}} = \frac{1}{M} \sum_{i=1}^{c} \mathrm{C}_i, \tag{2.3}$$

where C_i is the number of atomic segments in the largest class inside each cluster, c is the number of clusters and and M is the total number of atomic segments.

The NMI is defined as

$$NMI(Y, C) = \frac{2 \times I(Y; C)}{[H(Y) + H(C)]}, \tag{2.4}$$

where Y denotes class labels, C denotes cluster labels, $H(.)$ is Entropy, and $I(Y; C)$ is Mutual Information between Y and C.

In this experiment, we collect 206 atomic segments belonging to 18 true clusters, which are clustered into 20 clusters as our algorithm incorrectly separates one of the clusters into three. All the atomic segments belong to the largest class inside the cluster, except for one, resulting in a purity score of 99.51% and a normalized mutual information score of 95.05%.

To justify that the geometric constraint is very beneficial to atomic trajectory matching, a comparison experiment is conducted, where the matching performance is calculated without adopting the geometric constraint. The purity score drops to 87.38%, and the NMI score becomes 81.80%, proving that the geometric constraint can improve the matching performance significantly. On the other hand, it also demonstrates, surprisingly, that a reasonable accuracy of over 80% of segment matching could be achieved by using magnetism and RSSI of a single AP.

2.2.3.2 Room size

The room size is evaluated by the error between the reconstructed room size and ground truth area, which is calculated as:

$$Error = \frac{|reconstructed\_room\_size - ground\_truth\_size|}{ground\_truth\_size}. \tag{2.5}$$

We compare the error of our system with SenseWit and CrowdInside with the data collected in office. The average error of SenseWit is 31.4%, and that of CrowdInside is 40.6%. Our system has an average error of 36.1%, which is lower than CrowdInside and higher than SenseWit. Without requiring lots of anchor points as SenseWit, the accuracy of our system is still comparable. The estimation error is mainly due to obstacles like tables, drawers, and chairs in the office, which prevent the cart or robot from walking through the entire room.

2.2.3.3 Hallway shape

We evaluate the hallway shape with the same metric as SenseWit [16], which is shown below:

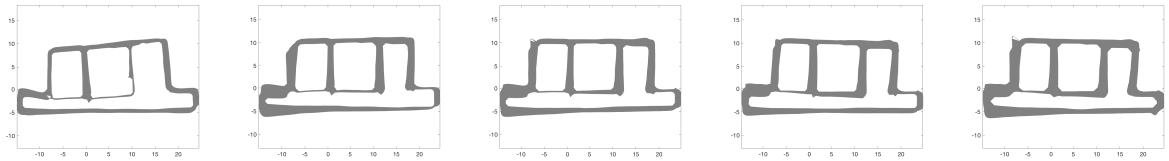
$$\begin{aligned} \mathcal{P} &= \frac{|S_{gen} \cap S_{true}|}{|S_{gen}|}, \\ \mathcal{R} &= \frac{|S_{gen} \cap S_{true}|}{|S_{true}|}, \\ \mathcal{F} &= 2 * \frac{\mathcal{P} * \mathcal{R}}{\mathcal{P} + \mathcal{R}}, \end{aligned} \tag{2.6}$$

where $\mathcal{P}$ is the precision of the hallway shape, $\mathcal{R}$ is the recall, and $\mathcal{F}$ is the harmonic mean of precision and recall. $\mathcal{P}$ is defined as the overlapped area divided by the reconstructed hallway area. $\mathcal{R}$ is defined as the overlapped area divided by the ground-truth hallway area.

Table 2.1: Evaluation results of hallway shape

	EZMap	**SenseWit**	**CrowdInside**
$\mathcal{P}$	78.18%	75.3%	59.5%
$\mathcal{R}$	75.10%	82.4%	47.1%
$\mathcal{F}$	76.61%	78.69%	52.0%

The evaluation result is shown in Table 2.1. The origin point and the orientation of ground truth hallway and the generated hallway are aligned. From the table, we can find that the $\mathcal{P}$, $\mathcal{R}$, and $\mathcal{F}$ of our system are better than CrowdInside and comparable with SenseWit. The precision of our system is higher than SenseWit, but recall rate is lower than SenseWit, which is because we do multi-trajectory fusion so that the hallway widths are less than the ground truths. Note that SenseWit needs various anchors to achieve comparable performance with the proposed system, showing that the proposed system can generate accurate hallway plans for various environments,



(a) (b) (c) (d) (e)

Figure 2.13: Reconstructed hallway plan results of Scenario I using different number of trajectories. (a) Result of using 24 trajectories. (b) Result of using 35 trajectories. (c) Result of using 46 trajectories. (d) Result of using 52 trajectories. (e) Result of using 61 trajectories.

especially for the private environments without anchor points, such as home.

2.2.3.4 Construction efficiency

Different from most crowdsourcing based indoor map construction system, *EZMap* can generate hallway map with much fewer trajectories while achieving comparable precision. We evaluate the construction efficiency of *EZMap* in Scenario I. Fig. 2.13 shows the reconstructed hallway with a number of 24, 35, 46, 52, and 61 trajectories, respectively. The hallway construction accuracy with different number of trajectories is 53.76%, 67.44%, 71.99%, 73.46%, and 76.61%, respectively.

We compare the construction efficiency of *EZMap* with SenseWit, and CrowdInside, shown in Table 2.2. SenseWit uses over 300 trajectories to cover a campus library with 464 m^2 area, and CrowdInside employs over 150 trajectories to cover an environment with 448 m^2 area. We can find that *EZMap* achieves comparable accuracy to SenseWit with only one fourth of the number of trajectories used in a twice larger environment, and *EZMap* is more accurate than CrowdInside with much less trajectories needed. The efficiency evaluation result shows that *EZMap* can rapidly reconstruct accurate hallway. In addition, our design employs low-cost home robots, which also save significant manpower for data collection.

Table 2.2: Evaluation results of construction efficiency

	EZMap	**SenseWit**	**CrowdInside**
Trajectory number	61	300	150
Area	803 m^2	464 m^2	448 m^2
Trajectory number/area	0.08 /m^2	0.65 /m^2	0.33 /m^2

2.3 Summary

In this chapter, we present *EZMap*, a universal automatic floor plan construction system that does not need any prerequisite knowledge of buildings in advance. *EZMap* benefits from recent advances in centimeter-accuracy indoor tracking using RF signals. It leverages commodity WiFi to estimate accurate moving distances and employs inertial sensors for orientation reckoning. *EZMap* then processes crowdsourced trajectories with a novel pipeline of trajectory segmentation, matching, bundling, and shaping, which ultimately reconstructs a floor plan with not only skeletal layouts but also detailed area sizes of straight/curved corridors, open spaces, and rooms. Requiring minimal infrastructure and a small amount of data, *EZMap* can scale to a number of various buildings including public malls, offices, as well as home environments. To our best knowledge, *EZMap* is the first RF-based system that can accurately reconstruct map of private environments such as home.

Chapter 3: Human and Non-human Motion Discrimination

With the proliferation of Internet of Things (IoT) devices, indoor intelligent applications such as security surveillance, intruder detection, occupancy monitoring, and activity recognition have gained significant attention [28]. However, these applications frequently suffer from an elevated rate of false alarms due to the inability to recognize human and non-human subjects, such as pets, robotic vacuum cleaners, and electrical appliances, as depicted in Fig. 3.1. This ability to differentiate is essential, especially for applications related to security, health monitoring, automation and energy management. Misidentification can lead to user frustration, erode trust, and hamper the practical and widespread adoption of these technologies.

Pets, vacuum machines, and electrical appliances such as fans are prevalent in indoor environments, especially in homes. According to the American Pet Products Association [70], about 70% of families in the United States (about 85 million families) have pets in 2022, and the percentage is increasing year by year. The global robotic vacuum cleaners market is expected to grow from $5.59 billion in 2021 to $7.83 billion in 2026, at a compound annual growth rate (CAGR) of 6.9%. Therefore, given the prevalence of pets, robotic vacuum cleaners, and electrical appliances, especially in residential environments [70], it is crucial to develop a reliable system that can accurately recognize human and non-human subjects.

In this chapter, we introduce the first system ("WI-MOID": **Wi**Fi-based human and

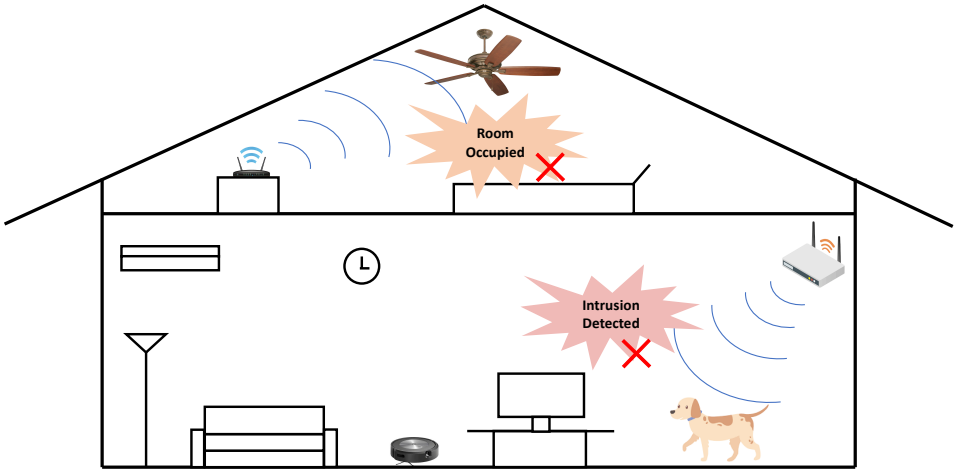


Figure 3.1: An illustration of non-human motion interference with sensing systems.

non-human **mo**tion **id**entification) that can accurately identify various human and non-human subjects through walls. The system utilizes ubiquitous WiFi signals and operates unobtrusively and without contact, eliminating the need for additional instrumentation or environmental restrictions. In contrast to existing systems that require pets, robots, and humans to move along predefined paths/directions/areas, *Wi-MoID* can detect freely moving subjects, allowing them to walk/turn/stand/stay without any predefined restrictions. *Wi-MoID* automatically detects the movement of the subject and extracts context-independent features, allowing it to function effectively in diverse environments without requiring additional training or parameter tuning.

The structure of this chapter is organized as follows: Section 3.1 provides an overview of *Wi-MoID*, including challenges, the insight and a system overview. Section 3.2 delves into the details of motion detection and speed estimation. Section 3.3 introduces feature extraction and motion recognition. Experimental setups and evaluations are described in Section 3.4. We discuss the impact of various factors in Section 3.5 and draw our conclusions in Section 3.6.

3.1 Overview

3.1.1 Challenges

To realize such a flexible system is not an easy task, several major challenges need to be addressed.

The first difficulty lies in deriving motion features in both Line-Of-Sight (LOS) and Non-Line-Of-Sight (NLOS) settings using only a single-link WiFi signal. Speed pattern, a key component of motion features, presents its own unique challenge. While various WiFi-based speed estimation approaches, such as those using Doppler Frequency Shift (DFS), have been attempted, they generally require multiple links or are only effective under LOS conditions. In this paper, we draw inspiration from WiSpeed [57] and adopt a rich-scattering model to overcome the challenge by treating subjects as multiple scatters.

Secondly, developing a system that operates effectively in unseen contexts without requiring additional training or parameter adjustments is a substantial challenge. Most existing WiFi-based activity recognition methods rely on features extracted directly from raw CSI, which are susceptible to environmental changes with a degraded performance in new environments without retraining. To address this issue, we leverage a second-order statistic, the autocorrelation function (ACF), of the CSI and only extract features caused by the change of the multipath profiles. Unlike raw CSI, the ACF is insensitive to the static subjects and solely reflects the movement of moving subjects, making it well-suited for motion recognition in different environments.

Third, accurately recognizing pet motion in real-world scenarios is challenging, especially

when their movement path, area, or activities are not restricted. Previous works that aim to recognize pet motion often make unrealistic assumptions about pets' movement, such as assuming a specific path or confined area, considering only walking activity, or requiring pets to remain close to devices quietly for extended periods [44, 45]. In this paper, we propose a new approach that combines physical and statistical features to classify human and non-human motion, and further enhance the classification using a Hidden Markov Model (HMM)-based state machine that leverages temporal information to boost the accuracy. By incorporating temporal information, *Wi-MoID* can accurately and reliably distinguish pet motion from human motion, even when pets move freely and engage in different activities.

3.1.2 Insight

A moving subject can be categorized as either human or non-human, the latter encompassing a range of entities such as pets or vacuum robots. When we refer to human motion, we are talking about the movement of the human body, encompassing activities such as walking, running, and sneaking. In contrast, non-human motion pertains to the movement of animate or inanimate subjects, which includes animals, robots, and other physical objects. We classify any motion originating from non-human entities as non-human motion.

Interestingly, though pets, cleaning robots, and even electrical appliances such as fans can all generate motion that can be picked up by sensing systems, their movements are distinct from those of humans. The differences in gait patterns are a key aspect of this distinction, with humans, pets, and robots each demonstrating unique movement patterns due to their bipedal, quadrupedal, and wheeled locomotion, respectively. These differences are visually represented in Fig. 3.2. As

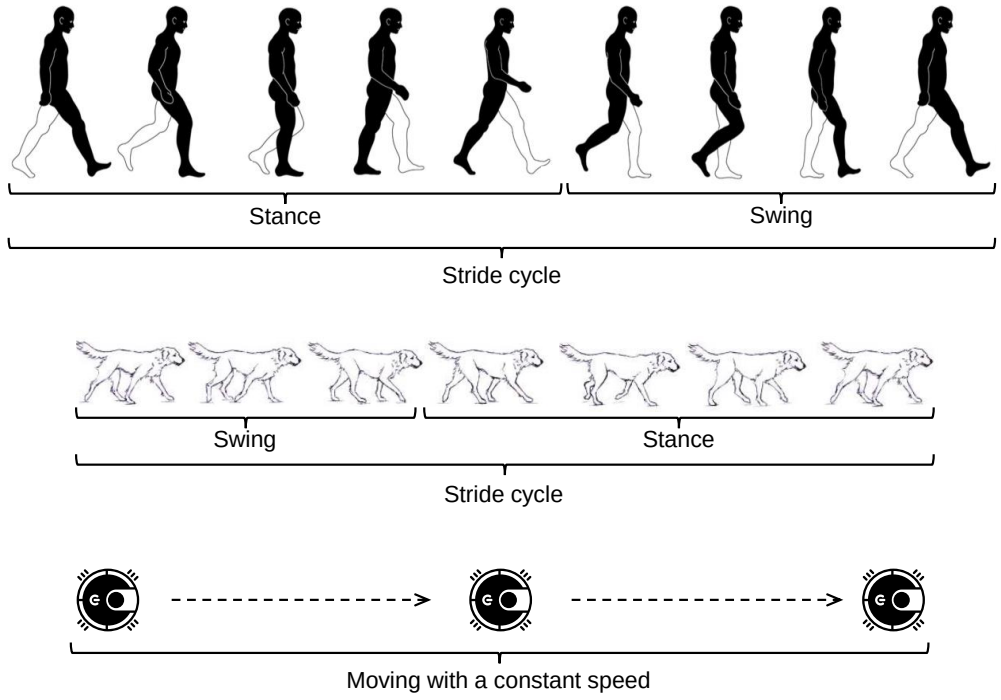


Figure 3.2: An illustration of moving patterns of human, dog and cleaning robot.

a result of these disparate movement patterns, the analysis of features like speed patterns becomes an essential factor in differentiating the motion of these subjects.

3.1.3 System Overview

The overview of *Wi-MoID*, as illustrated in Fig. 3.3, is primarily composed of the following three steps:

1) Firstly, we preprocess the CSI received by the receiver. We compute the ACF of the CSI and, based on this, calculate the motion statistic to determine the presence of moving targets in the environment. Additionally, we derive the speed of the moving target based on the ACF.

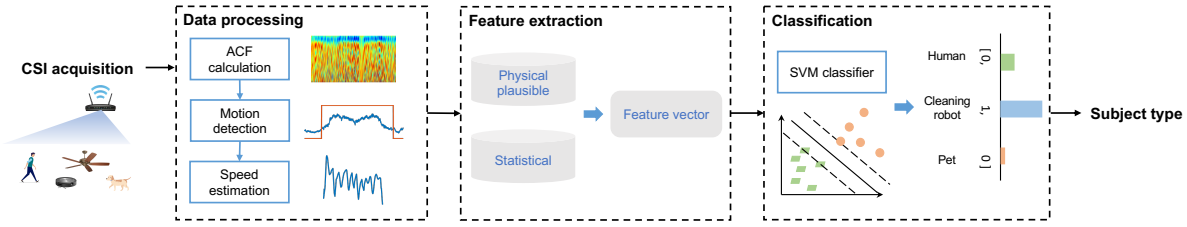


Figure 3.3: System overview.

2) Secondly, we effectively extract a series of features from the moving target. Using the ACF, motion statistic, and speed, we extract physically interpretable features and signal statistical features of the moving target. These features are independent of the environment, location, and direction.

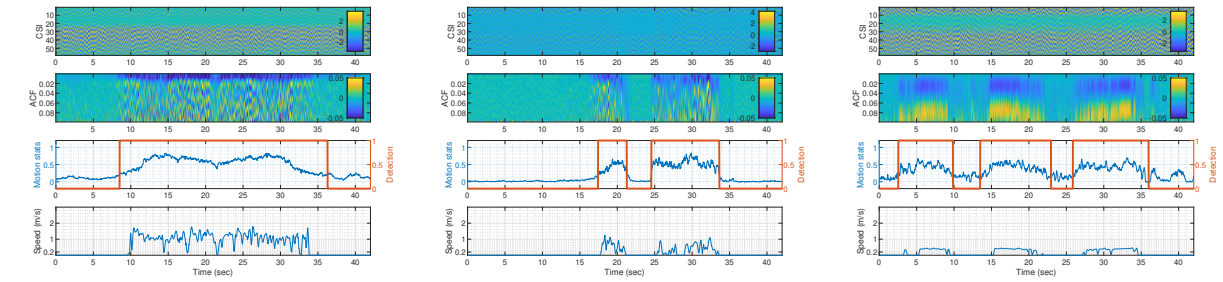
3) Lastly, using SVM and a state machine, we output the final identification result of the system. We initially use SVM to classify the features of the target and then employ a state machine to adjust any potential misclassifications based on the transitional features between states, finally outputting the result.

3.2 Motion Detection and Speed Estimation

This section presents our approach to detecting the motion of subjects and estimating their speeds through walls, using a single WiFi link.

3.2.1 Preliminary

In order to accurately detect the motion of a subject in an indoor environment, it is crucial to consider the presence of numerous multipath signals arising from rich scattering. A statistical model that account for all these multipath components can be employed to estimate the speed of



(a) Human (b) Cleaning robot (c) Pet

Figure 3.4: CSI, ACF, motion statistics and speed estimation for (a) human, (b) pet, and (c) cleaning robot.

the target [29, 36, 57].

In indoor environments, rich scattering occurs due to both static and dynamic scatterers. Static scatterers comprise walls, floors, and stationary furniture, while dynamic scatterers include moving individuals or objects. The superposition principle of electromagnetic (EM) waves allows us to decompose CSI as follows:

$$H(t,f) = \sum_{i \in \Omega_s(t)} H_i(t,f) + \sum_{j \in \Omega_d(t)} H_j(t,f) + \varepsilon(t,f), \tag{3.1}$$

where $\Omega_s(t)$ represents the set of static scatterers, $\Omega_d(t)$ corresponds to the set of dynamic scatterers. $H_i(t,f)$ and $H_j(t,f)$ represent the contributions of the i-th and j-th scatterers respectively. The noise term, $\varepsilon(t,f)$, is statistically independent of $H_i(t,f)$ and $H_j(t,f)$ [57]. Each scatterer acts as a "virtual transmitter", scattering its received EM waves around. The CSI represents the aggregate of the electric fields of all incoming EM waves.

The ACF of the CSI $H(t,f)$ with a time lag τ, denoted as $\rho_H(\tau,f)$, can be determined by calculating the covariance between $H(t,f)$ and $H(t+\tau,f)$ divided by the variance of $H(t,f)$

itself. Mathematically, it can be expressed as:

$$\begin{aligned}\rho_H(\tau, f) &= \frac{\mathrm{Cov}[H(t,f), H(t+\tau,f)]}{\mathrm{Cov}[H(t,f), H(t,f)]} \\ &= \frac{\sum_{i\in\Omega_d} \sigma^2_{F_i}(f) J_0\left(k v_i \tau\right) + \sigma^2(f)\delta(\tau)}{\sum_{i\in\Omega_d} \sigma^2_{F_i}(f) + \sigma^2(f)}.\end{aligned} \tag{3.2}$$

Here, $J_0(\cdot)$ is the Bessel function of the first kind, given by $J_0(x) = \frac{1}{2\pi}\int_0^{2\pi} \exp(-jx\cos(\theta))d\theta$. The term $\delta(-)$ represents the Dirac delta function.

3.2.2 Motion Detection and Speed Estimation

The motion statistic $\phi(f)$ for a subcarrier with frequency f is defined as ACF of the CSI $H(t, f)$ with a time lag of $\tau = 1/F_s$, where F_s is the sounding rate [29]. That is,

$$\phi(f) \triangleq \rho_H\left(\tau = \frac{1}{F_s}, f\right). \tag{3.3}$$

Motion statistics function as a reliable gauge of movement presence or lack thereof within a given environment. In a stationary environment, the motion statistic $\phi(f)$ is close to 0, whereas in dynamic environments with movement, $\phi(f) > 0$. Fig. 3.4 illustrates the motion statistics of human, pet, and robot movements.

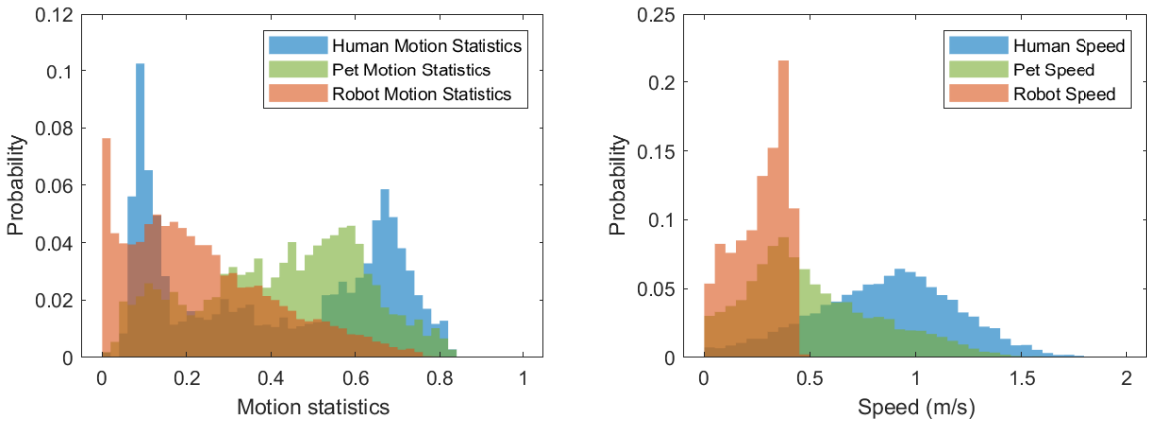
Despite the discrepancies in size, various subjects are capable of producing similar disruptions across different distances. This is evident from the analogous motion statistics exhibited in Fig. 3.4 and Fig. 3.5(a). Consequently, relying solely on motion statistics cannot produce reliable recognition results.

Considering that speed can effectively capture the distinct walking patterns exhibited by

humans and non-human entities, we extend our feature set by extracting speed information from WiFi signals.

To approximate the motion speed of a target, we simplify the ACF of the CSI by employing a motion model that assumes the torso contributes most of the strong scatterers, resulting in $\rho_H(\tau, f) = \alpha(f)J_0(kv\tau)$ [36], where $\alpha(f)$ represents the gain of each subcarrier. This function bears resemblance to the well-known Bessel function $J_0(x)$, where $x = kv\tau$. By aligning the position of the first peak or valley of $J_0(kv\tau)$ with that of the Bessel function, we can estimate the subject's speed. Specifically, we detect the first peak and calculate the speed as $\hat{v} = \frac{x_0}{k\hat{\tau}} = \frac{x_0\lambda}{2\pi\hat{\tau}}$, where x_0 denotes the constant value corresponding to the first peak of $J_0(x)$, and $\hat{\tau}$ represents the time lag corresponding to the first peak in $J_0(kv\tau)$. Estimated speeds of human, pet, and robot motions derived from the ACF are depicted in Fig. 3.4.

Although estimating the speed of a moving subject provides significant insights for subject differentiation, it is inadequate to rely solely on speed values to distinguish between human and non-human subjects, due to the potential overlap in their speed values, as shown in Fig. 3.5. This is because their movements may exhibit significant overlap in terms of speed values. Therefore, a comprehensive analysis of motion statistics, speed, and ACF patterns over time is imperative for the successful distinction between moving subjects. By extracting features that have physical and statistical explanations from these patterns, we can enhance discrimination and accurately identify human and non-human subjects. The subsequent section will elaborate on these features and their role in motion classification.



(a) Motion statistics (b) Speed

Figure 3.5: Histograms of (a) motion statistics and (b) speed estimations for human and pet.

3.3 Feature Extraction and Motion Recognition

This section outlines the methodology for distinguishing between the motion of human, pet, and cleaning robot. First, a set of features is extracted, followed by the development of a recognition model that utilizes the selected features.

3.3.1 Feature Extraction

In our pursuit to discern features that can effectively identify movements of human and non-human subjects, we adopt a two-pronged approach: examination of physical motion characteristics and statistical properties gleaned from CSI. On the physical front, we delve into gait-related attributes and speed patterns exhibited by the subjects. Statistically, we scrutinize the statistical features of ACF and changes in motion statistics. This combined approach aims to encapsulate the holistic nature of movement and its impact on the WiFi signal, facilitating accurate identification across varied subjects.

3.3.1.1 Physical Features Extraction

The term 'gait' signifies a pattern of limb movement during locomotion on a solid surface and has proven an effective human identification feature. Humans walk bipedally, pets such as dogs and cats predominantly operate as quadrupeds, and cleaning robots rely on the wheel-based motion. These inherent differences result in unique gait and speed patterns, providing differentiation between humans, pets, and robots.

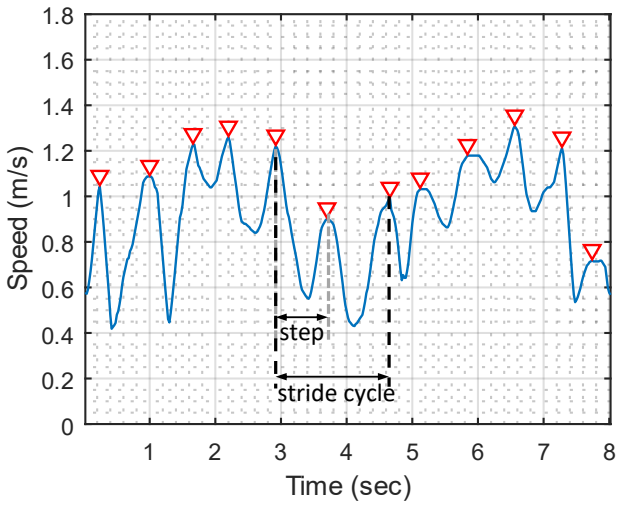


Figure 3.6: Example of human stride cycle.

In locomotion, the movement pattern of animals' limbs generates the gait, where humans and pets alternate their feet. A stride cycle encompasses a stance phase (foot-ground contact) and a swing phase (foot lift and forward motion). Speed changes are integral to gait, decreasing during the stance phase and increasing during the swing phase, as shown in Fig. 3.6. In contrast to humans and pets, vacuum robots don't exhibit significant speed fluctuations, enabling their differentiation via gait detection. We estimate motion speed using the method outlined in Section 3.2, identifying patterns of speed through peak and valley analysis. Gait presence is determined

by detecting speed fluctuations with a distinct ups and downs regularity.

Stride Cycle Time and Stride Length. Stride cycle time and stride length are essential gait indicators. Stride cycle time refers to the duration between the same foot contacting the ground twice, while stride length is the distance covered by a person or animal between two consecutive footfalls. These definitions are applicable to quadruped pets like dogs and cats. Since pets generally have shorter legs than humans, their stride lengths are smaller than that of humans during normal indoor movement.

We extract these two features from the speed curves to distinguish the movement of human and pet movements. Specifically, the stride cycle time is extracted by calculating the time duration of three consecutive speed peaks. Stride length is the integral of speed value over stride cycle time. Fig. 3.7 illustrates the average stride lengths of human and dog across ten motion instances. It can be observed that although humans and pets may have similar stride cycle times, their stride lengths are significantly different. Pets exhibit much shorter stride lengths compared to humans due to their shorter legs.

Speed Mean and Deviation. As wheeled platforms lack feet, vacuum robots typically maintain a constant speed while moving from one location to another and slow down when encountering obstacles. Therefore, speed-based features are analyzed to differentiate between the movements of vacuum robots and humans. The average speed is calculated as the mean of the instantaneous speeds of a moving target during a particular time window. Additionally, speed deviations are taken into account to distinguish between the motions of humans, pets, and robots, as depicted in Fig. 3.8. Specifically, we estimate the variance, 25th percentile value, and 75th percentile value of the speed in the time window as features of speed deviations.

Vacuum cleaners, being wheeled, lack feet and maintain relatively constant speeds, slowing

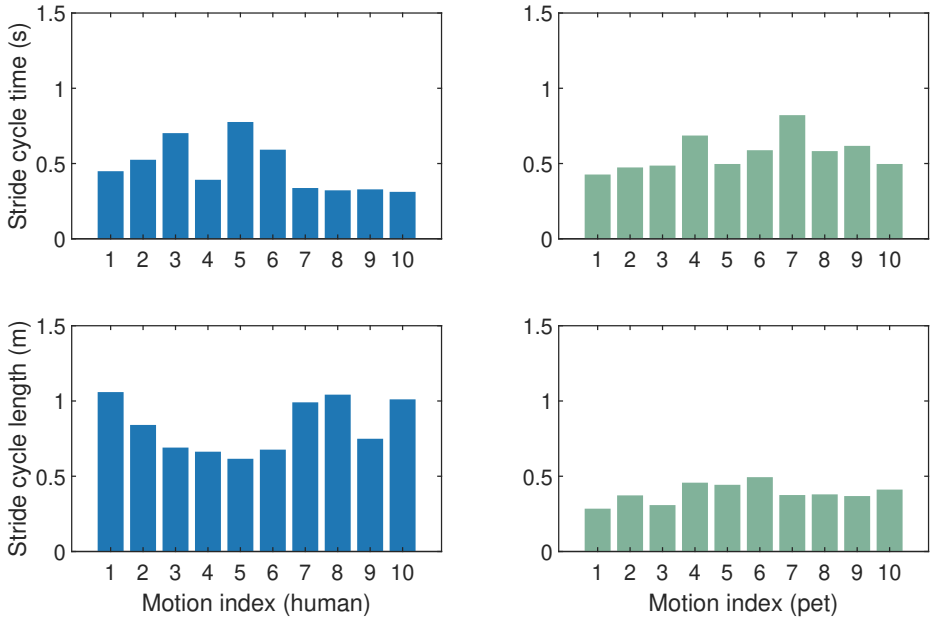
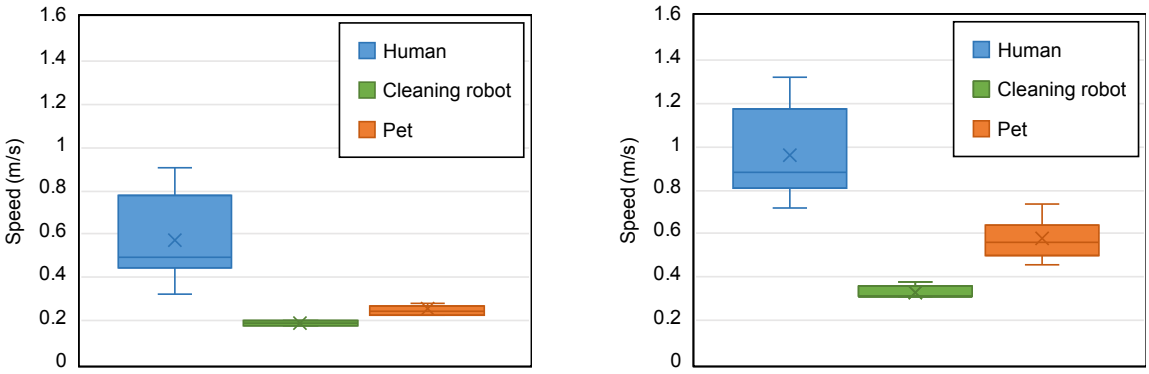


Figure 3.7: Stide cycle time and stride length of human and pet.

only when meeting obstacles. Hence, we employ speed patterns to distinguish between vacuum robot and human movements. Specifically, the average speed and speed deviations are extracted. The average speed is derived from the mean of a target's instantaneous speeds within a specific time window. As for speed deviation, we estimate the variance, 25th and 75th percentile value of speed, as shown in Fig. 3.8.

3.3.1.2 Statistical Feature Extraction

In most cases, human and pet movements exhibit distinctive gait features. However, certain actions such as drinking, eating, and tail wagging do not demonstrate gait characteristics. Therefore, to augment our distinction capability, we further exploit other features such as statistical properties derived from the ACF and motion statistics of CSI. These attributes prove valuable in situations where speed estimation is not reliable, such as when subjects are too



(a) (b)

Figure 3.8: (a) 25 percentile and (b) 75 percentile of speed estimations for human, robot, pet.

distant for accurate gait estimation.

ACF Features. ACF of CSI serves as an advantageous statistic since it encapsulates a wealth of information regarding the subject's motion, while its sensitivity to location and environment is minimal. As such, we extract features based on ACF. Our observations reveal that the movements of humans, pets, and robots uniquely impact the amplitude and frequency of ACF peaks at each time step. This distinction is evidenced by the variances in the amplitude and time lag of ACF peaks and valleys across different types of movements, as depicted in Fig. 3.9(a). To leverage these distinctions, we extract the mean values of ACF peaks and valleys separately for each movement type, as shown in Fig. 3.9(b, c). These features provide a comprehensive representation of the unique characteristics of human, pet, and robot movements and enable effective differentiation between different motion types in various environments.

Motion Statistic Features. In addition to the ACF-based features, motion statistics also provide valuable insights into the statistical characteristics of the impact of the subject's movement on the CSI. This assists in refining our ability to identify human and non-human movements. To capture these statistical characteristics, we compute motion statistics according

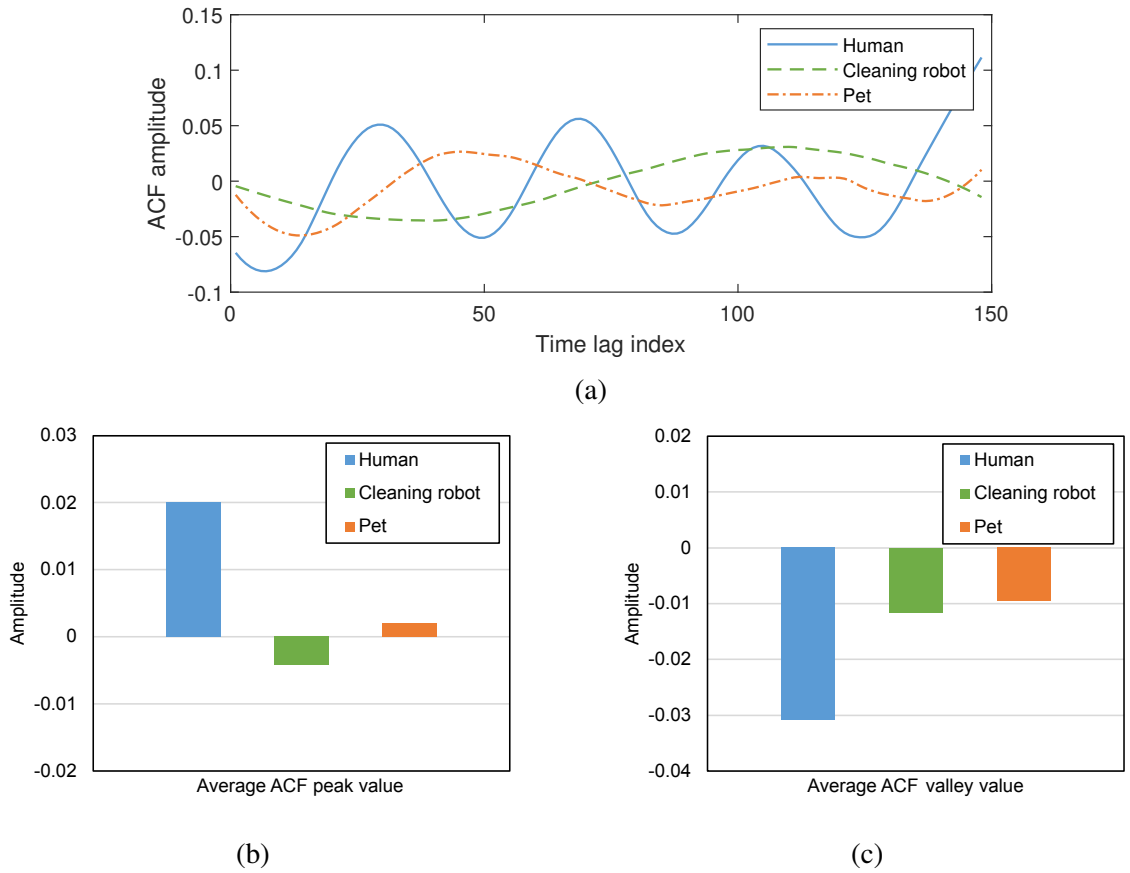


Figure 3.9: (a) ACF at a time instance, (b) average ACF peak value, and (c) average ACF valley values of human, robot, pet.

to (5.3) and extract their mean and variance as additional features. The statistical features enhance the system's resilience to variations in the environment, making it more adaptable to real-world scenarios.

3.3.2 Recognition Model Design

SVM is a supervised learning model that utilizes associated learning algorithms to classify and perform regression analysis on data. Our SVM model uses a feature vector consisting of 11 features, including gait existence, stride length, stride cycle time, average speed, speed variance, speed 25th percentile, speed 75th percentile, ACF peaks mean, ACF valleys mean,

motion statistic mean, and motion statistic variance. During the training phase, we trained SVM models employing Linear kernel or Gaussian kernel, and the best model is chosen via a grid search complemented by 5-fold cross-validation. Concurrently, the features underwent standardization. The trained SVM model is subsequently tested in unknown environments, with no further training or parameter adjustments.

3.3.3 State Machine

In some cases, distinct continuous gait features are not available due to irregular motion during walking/running/slow moving, such as stopping, turning around, etc. Consequently, the classifier may misclassify humans as non-human due to the lack of obvious gait features and the presence of small motions. To address this challenge, we propose a state machine based on HMM.

In the proposed state machine, we estimate the parameters using the maximum likelihood estimation method to determine the most likely probability of the observation sequence occurring. Let the emission probability $b_j(k)$ represent the probability of observing symbol k in state j. It can be estimated as:

$$b_j(k) = \frac{c_j(k)}{\sum_{k=1}^{M} c_j(k)}, \tag{3.4}$$

where $c_j(k)$ denotes the number of times symbol k is observed in state j, and M represents the total number of symbols. The transition probability $a_{i,j}$ represent the probability of transitioning from state i to state j, and can also be estimated as

$$a_{i,j} = \frac{c_{i,j}}{\sum_{j=1}^{N} c_{i,j}}, \tag{3.5}$$

where $c_{i,j}$ is the number of transitions from state i to state j, and N denotes the total number of states.

By leveraging the Hidden Markov State Machines, we can enhance the performance of the proposed classifier and minimize the occurrence of misclassifications due to irregular human motions, as will be detailed in Section 3.5.

3.4 Evaluation

This section elaborates the evaluation of *Wi-MoID*. First, we introduce the evaluation methodologies, including experiment settings, data collection, and metrics. Then, we discuss the recognition performance from various aspects.

3.4.1 Methodology

Experimental Setting. We utilize off-the-shelf WiFi devices, as shown in Fig. 3.11(a), to implement *Wi-MoID* and performed experiments in four representative indoor environments. Specifically, Scenario I corresponds to a 76.96 m^2 apartment, Scenario II a 70.64 m^2 townhouse, Scenario III a 40.27 m^2 single-family house, and Scenario IV a 200.75 m^2 office building, as illustrated in Fig. 3.10. To obtain data from different locations and heights, we employ one transmitter and two to three receivers in every data gathering session. Receivers are positioned 4 to 10 meters from the transmitter at heights varying between 0.3 m and 1 m, considering the presence of walls. Note that although we used multiple transceiver pairs in each session to improve efficiency and diversity, the CSI data from different pairs are not combined. The feature extraction and recognition processes are performed based on a single-pair WiFi transceiver. The

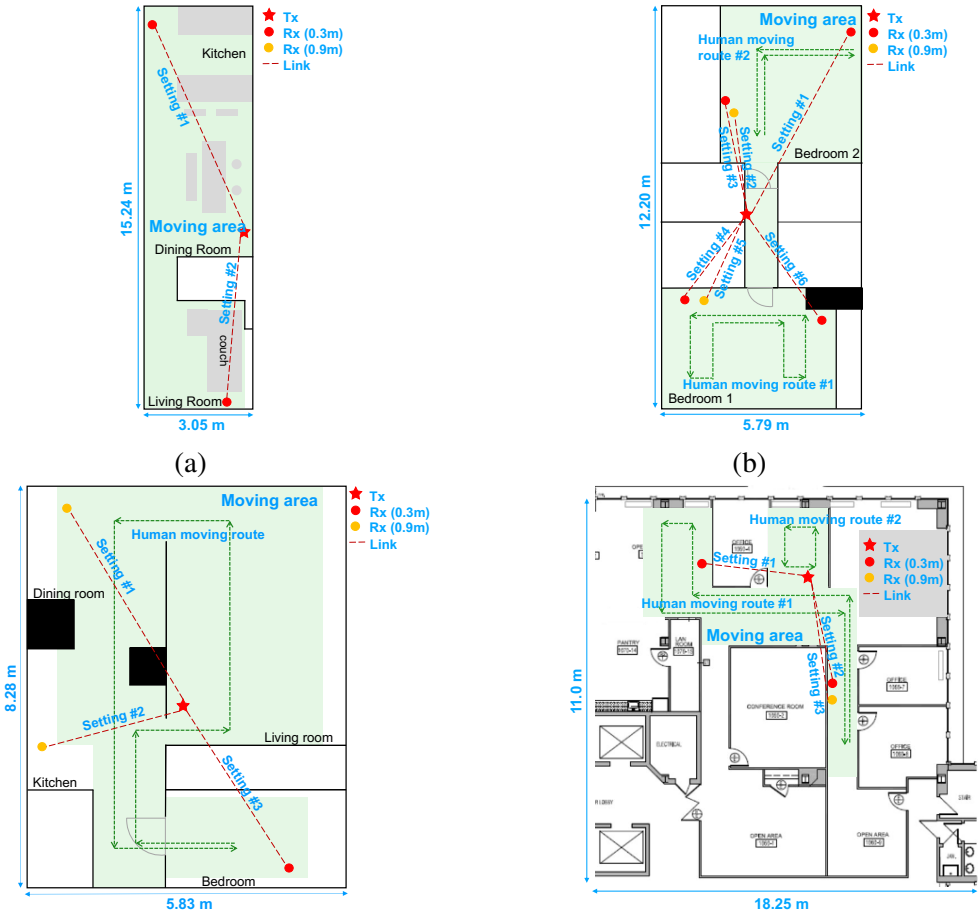


Figure 3.10: Floor plans of (a) Scenario I: an apartment, (b) Scenario II: a townhouse, (c) Scenario III: a single family house, and (d) Scenario IV: an office building. Multiple pairs of transceiver are adopted to enhance data collection efficiency and data diversity, but only data from single transceiver pair is used for feature extraction and recognition.

CSI data is collected on 5.8 GHz channels with a bandwidth of 40 MHz and a sounding rate of 1500 Hz. In each scenario, the movement areas for humans, pets, and vacuum robots are delineated in green on Fig. 3.10.

Data Collection. We gather motion data from two human subjects of respective heights 170 cm and 187 cm, one male and one female, alongside a dog approximately 1 meter in length, and an iRobot i3 vacuum cleaner. All data collection is conducted in a natural manner without any restrictions. Specifically, the human movement data is captured while individuals is freely

(a) (b)

Figure 3.11: (a) Off-the-shelf WiFi device, (b) pet in Scenario I.

walking around the environments, with the freedom to roam and stop at will. They engage in various activities such as using mobile phones or playing games. Continuous data collection is performed for the dog's movement throughout the day as it walk around, ate, drank, and wag its tail as usual. Fig. 3.11(b) shows the pet in Scenario I. We gather the vacuum cleaner's motion data when it is cleaning the floors furnished with items like tables and chairs, with the robot moving freely within the accessible spaces to perform its task.

The experiments are conducted over a period of six months, where the subjects initiate their movements from arbitrary locations. Movement durations range from 2 seconds to several minutes during each data collection session. In total, we accumulate approximately 287 minutes of human motion data, 185 minutes of pet motion data, and 100 minutes of robot motion data. We apply motion statistics to detect motion and segment the data into 6-second instances. This yields 2318 instances of human motion data, 493 instances of pet motion data, and 796 instances of robot motion data. For the training and validation process, we randomly allocate 90% of

the data from Scenarios I and II to form the training dataset, reserving the remaining 10% for validation. We use Scenarios III and IV data exclusively to test the model's resilience to unknown environments, excluding them from the training or validation phases.

To evaluate the model's adaptability to a variety of human motion types, we gather 356 instances of sneaking and 367 instances of running data in Scenarios II and III. Additionally, 306 instances of ceiling fan motion are collected in Scenarios I and III to evaluate the model's versatility in recognizing various non-human subject movements. The diverse human and fan motion data, collected in Scenarios I and II, are incorporated into our training and validation datasets. Conversely, data from Scenario III are leveraged to assess the model's performance in an unknown environment, offering a comprehensive evaluation of its ability to generalize and adapt.

Evaluation Metrics. The effectiveness of *Wi-MoID* is gauged through the following key metrics:

- Recall or True Positive Rate (TPR) represents the likelihood of the system correctly identifying the target.
- Precision, alternatively termed as Positive Predictive Value (PPV), signifies the proportion of accurate recognition results for a specific class.
- Accuracy is the ratio of correctly identified data to the total data pool.
- False alarm or False Positive Rate (FPR) measures the probability of the system misidentifying the target's motion.

The system's performance is considered improved when there is an increase in accuracy,

precision, and recall, along with a decrease in the false alarm rate. These metrics collectively serve as a measure of the system's efficacy.

3.4.2 Recognition Performance

To evaluate the effectiveness of *Wi-MoID*, we conduct experiments using data collected from Scenarios I and II as the training and validation dataset. Firstly, we extract features from the data and train SVM models with different kernels. To ensure the model's ability to generalize, we incorporate a 5-fold cross-validation strategy to randomize the training and validation dataset. The model that demonstrates the highest validation accuracy is chosen as the final model. The optimal SVM model's performance is gauged by its accuracy, precision, recall, and false alarm rate, the details of which are presented in Table 3.1. The table shows that *Wi-MoID* achieves high validation accuracy of 97.34%, paired with a minimal false alarm rate of 1.75%, indicating its ability to accurately identify the motion of human and non-human subjects.

Table 3.1: Human and non-human motion identification performance

Subject	**Validation (%)**			**Testing (%)**		
	Recall	**Precision**	**False Alarm**	**Recall**	**Precision**	**False Alarm**
Human	98.31	99.96	3.33	99.04	91.78	3.81
iRobot	94.12	99.21	1.69	100.00	100.00	0.00
Pet	98.08	80.45	0.23	49.32	90.00	7.18
Overall Accuracy	97.34			92.61		

3.4.2.1 Adaptivity to Environment Changes

To assess the generalizability of the trained model across new contexts, we employ a testing dataset gathered from previously unobserved settings (Scenarios III and IV). The model's

testing accuracy is ascertained without further retraining is documented in Table 3.1. We observe that our pre-trained model achieves a high testing accuracy and precision of 92.61% and 93.93%, respectively, along with a low false alarm rate of 3.66%. These results demonstrate the robustness of *Wi-MoID* in recognizing humans, pets, and robots through walls in new environments. Furthermore, they confirm that our trained model is resilient to environmental changes and independent of the training environment. This attribute facilitates quick and effortless deployment in new contexts, obviating the need for additional user exertion.

3.4.2.2 Adaptivity to Other Non-human Subject

While pets and robots are typically the primary sources of non-human motion in indoor scenarios, electrical appliances such as fans can also introduce motion. Therefore, it is necessary to evaluate the robustness of our algorithms in recognizing motion caused by electrical appliances. In particular, we assess the robustness of *Wi-MoID* to fan motions. To achieve this, we gather 306 instances of CSI data while the fan is operating in the environment. The results of validation accuracy, testing accuracy, and false alarm rate are presented in Table 3.2. Even with the inclusion of fan motion, the validation and testing accuracies remain high, and the false alarm rate is low. These results indicate that our model is adaptable to other electrical appliances' motion, such as fan motion, due to the effectiveness of the proposed features. Our approach encompasses the extraction of attributes that carry physical significance and simultaneously encapsulates statistical elements. These elements effectively mirror the influences of both human and non-human movements on the CSI within the environment. Since electrical appliances typically operate continuously and periodically, they affect the statistics differently from human

motions, allowing our model to robustly distinguish electrical appliances' motion from human motion.

Table 3.2: Adaptivity to other non-human subjects

	Validation accuracy(%)	**Testing accuracy(%)**	**False alarm(%)**
Human, iRobot, Pet	97.34	92.61	1.75
Human, iRobot, Pet, Fan	97.58	90.52	1.48

3.4.2.3 Adaptivity to Human Motion Types

To gauge our model's flexibility in handling other human motion types, we incorporate sneaking and running activities into our experimental setup. A total of 106 minutes of CSI data encapsulating these motions is collected, allowing us to extract 723 6-second instances representing various human movements. We train and validate the SVM model using various human motion data from Scenario I and II. The model's robustness is further assessed by testing it with data from Scenarios III and IV, which is not part of the initial training set. All data subsets utilized for training, validation, and testing comprise an all-encompassing array of human movements, inclusive of walking, running, and sneaking. The results, illustrated in Fig. 3.12, reveals a high recognition accuracy of 92.55% even with the inclusion of more diverse human motion types. This underscores the adaptability of *Wi-MoID* to accommodate a variety of human movements.

Figure 3.12: Adaptivity to human motion types.

3.5 Discussion

3.5.1 Effectiveness of State Machine

To evaluate the effectiveness of the proposed state machine, we compare the recognition accuracy of *Wi-MoID* with and without the state machine. Table 3.3 presents the accuracy and false alarm rates for human, robot, and pet recognition. The results demonstrate that the state machine improves the recognition accuracy of human, robot, and pet by 1.67%, 1.85%, and 17.98%, respectively, compared to the system without the state machine. The state machine improves the accuracy of pet recognition significantly by addressing the challenges posed by pet motion, which are difficult for SVM classifiers to handle. Specifically, pets can be misclassified as humans due to their similar gait patterns, and they can also be mistaken for cleaning robots when they move slowly or wave their tails, generating small motions that can interfere with CSI signals in a manner similar to cleaning robots. In addition, the proposed state machine

Table 3.3: State machine performance evaluation

Subject	Performance	Without state machine	With state machine	Improvement
Human	Accuracy(%)	98.29	99.96	1.67
	False Alarm(%)	4.49	3.33	1.16
iRobot	Accuracy(%)	97.36	99.21	1.85
	False Alarm(%)	4.25	1.69	2.56
Pet	Accuracy(%)	62.47	80.45	**17.98**
	False Alarm(%)	2.2	0.23	1.97

reduces the false alarm rates of human, robot, and pet recognition by 1.16%, 2.56%, and 1.97%, respectively. The reduction in false alarms and the increase in human detection accuracy are particularly noteworthy since the state machine helps avoid misclassifying cases where human motion lacks an obvious gait pattern. These results demonstrate that the state machine effectively improves recognition accuracy and reduces the false alarm rate, enhancing the robustness and reliability of *Wi-MoID*.

3.5.2 Latency

We assess the computational performance of *Wi-MoID* by running the MATLAB code on a desktop computer equipped with an Intel Core i7 processor and 16-GB memory. For a 6-second segment of motion data, the feature vector extraction process takes approximately 3.076 seconds. The training and testing of the SVM with 1,000 instances take 1.084 seconds and 0.019 seconds, respectively. Since the feature vectors are environment-independent, the training process can be done offline, and the time required for training is negligible.

3.5.3 Feature Efficiency

To evaluate the effectiveness of the features designed in our model, we conduct a comparative analysis utilizing a model trained with a 95% Principal Component Analysis (PCA)-driven approach, which is designed to decrease the feature vector's dimensionality. This strategy preserves only the principal components accounting for 95% of the total variance. The SVM model underwent retraining with the condensed feature set, using the identical dataset. The comparison results are recorded in Table 3.4, which demonstrates that the accuracy of the PCA-trained model is lower than that of the original model that utilized our designed features. This observation indicates the effectiveness of the designed features, which contributes to the high accuracy of our model.

Table 3.4: Recognition performance with feature selection

	Overall accuracy(%)	**Average recall(%)**	**Average precision(%)**
All features	97.34	96.84	93.21
Feature selection	94.52	91.79	86.19

3.5.4 Length of Motion Segments

We investigate the impact of the length of motion segments on the performance of *Wi-MoID*, as shown in Fig. 3.13. Our results indicate that the length of motion segments does not have a significant effect on the accuracy of the system. While shorter motion segments may provide more specific information about the motion, longer motion segments may offer a broader perspective of the overall motion. However, our findings suggest that neither length leads to a significant difference in the accuracy of the system. Therefore, we conclude that *Wi-MoID* is

robust to variations in the length of motion segments, ranging from 6 seconds to 25 seconds, and can accurately analyze motions of different durations.

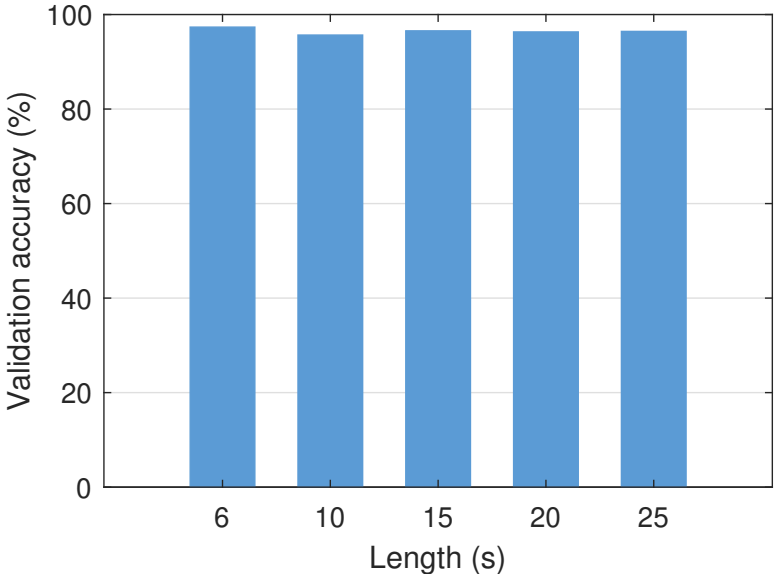


Figure 3.13: Impact of length of motion segments.

3.5.5 Sounding Rate

In this study, we also explore the impact of different sounding frequencies on performance. We compare the detection rate and false alarm rate of *Wi-MoID* at four different sounding frequencies: 1500 Hz, 500 Hz, 100 Hz, and 30 Hz, as shown in Table 3.5. The results demonstrate that the proposed algorithm achieves high accuracy and low false alarm rates across a range of sounding rates, with accuracy improving as the sounding rate increases. Notably, we also observes that the algorithm's accuracy remains satisfactory even at low sounding rates, indicating its robustness to variations in data acquisition frequency. This suggests that the algorithm can be effectively used in resource-constrained scenarios where high sounding rates may not be feasible due to memory or power limitations, without compromising its accuracy.

Table 3.5: Performance at difference sounding rates

Sounding rates (Hz)	**1500**	**500**	**100**	**30**
Accuracy (%)	97.34	94.36	94.31	91.80
Recall (%)	96.84	94.57	95.43	91.94
Precision (%)	93.21	88.48	87.97	84.16
False alarm (%)	1.75	3.06	3.38	4.68
Accuracy w/o state machine (%)	93.58	88.48	85.88	86.03

Therefore, the proposed algorithm offers a scalable solution for real-time data processing applications on resource-constrained edge devices.

3.5.6 Edge Device Deployment

The computational efficiency and storage requirements of the proposed algorithm are evaluated on commercialized edge devices with Marvell chipset, as shown in Fig. 3.14. Our results indicate that the algorithm is highly optimized for edge computing, with minimal CPU consumption and storage usage. This makes it an ideal candidate for deployment on resource-constrained edge devices. Moreover, the algorithm's scalability enables it to adapt to varying requirements and device specifications, making it a versatile solution for real-time data processing at the edge.

3.6 Summary

In this chapter, we introduce an unprecedented system that identifies the movements of various human and non-human subjects utilizing readily available WiFi devices. The system identifies moving subjects by analyzing both physically interpretable and statistical attributes of

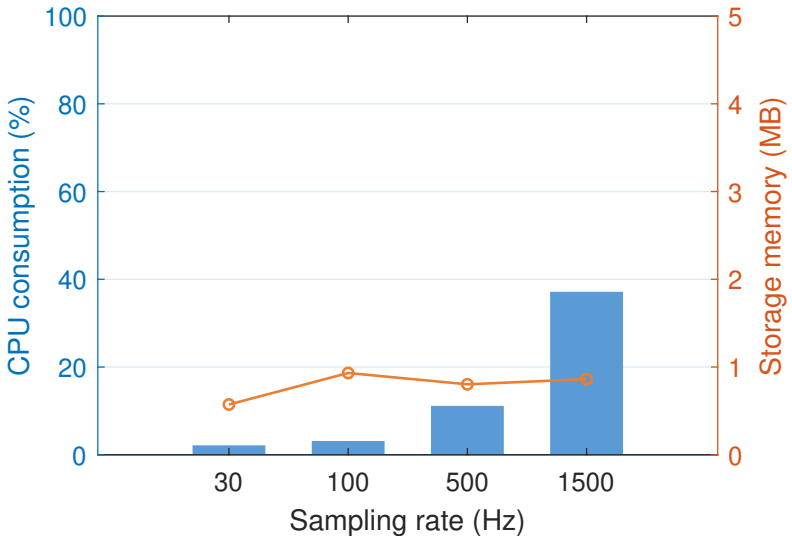


Figure 3.14: CPU and memory requirements.

the movement. A comprehensive assessment in diverse indoor scenarios validates the resilience of *Wi-MoID* in the real-world against various non-human subjects and changing conditions. *Wi-MoID* has been implemented on edge devices and is ready to be deployed ubiquitously for extensive use.

Chapter 4: Deep Learning for Human and Non-human Motion Identification

Present efforts to distinguish between human and non-human subjects using WiFi generally fall into two categories: traditional model-based and machine learning-based. The traditional model-based methods, although providing good interpretability, demand precise placement and height of equipment, and their effectiveness diminishes with increasing environmental complexity. In contrast, traditional machine learning-based methods, utilizing extracted meaningful features, can mitigate the limitations imposed by equipment placement and environmental constraints, thereby exhibiting a wider range of applicability. Relying on hand-crafted features, these methods retain as good interpretability as model-based methods. However, traditional machine learning models' performance hinges on the efficacy of these manually designed features. Their performance is compromised when these features can't be accurately derived from the data. Moreover, models based on traditional machine learning classifiers such as Support Vector Machines (SVM) tend to perform poorly with classification issues exhibiting considerable overlap and struggle with data that is not linearly separable.

In recent years, deep learning models have achieved remarkable success in domains such as image and speech processing, showcasing their potent ability to extract and classify features from high-complexity and high-dimensional data. This has sparked interest in the realm of WiFi sensing, with many studies, such as [52, 71–74], increasingly employing deep learning.

However, these works often merely adopt a specific pre-existing network framework. Although various types of deep learning models have excelled in fields like image processing and language learning, current WiFi sensing studies have not extensively compared the performance of different models on their tasks, leading to a dearth of exploration on the suitability of different models for WiFi sensing tasks. Additionally, most deep learning-based WiFi sensing tasks at present rely on raw CSI as input, unable to mitigate the impact of environmental factors on feature extraction. Consequently, trained networks often falter when transposed into new environments. While various strategies for domain adaptation are proposed, they necessitate extensive user data to retrain or tweak the model in a new environment.

In this chapter, we design a framework utilizing deep neural networks to recognize motion from human and non-human subjects with WiFi signals. Our method utilizes the Amplified Auto-correlation Function (A-ACF) extracted from CSI as the network input, enabling the network to extract only the features associated with the subject's movement, independent of the subject's location, orientation, and environmental changes. We investigate two major categories of deep learning models, convolution-based ones from image processing and recurrent neural network-based ones from natural language processing. We evaluate the performance of these models in human and non-human recognition tasks, as well as their computing resource requirements. In addition, we investigate the effectiveness of transferring pre-trained models in the image processing field to WiFi sensing tasks. Extensive experiments are conducted in five real world scenarios to demonstrate the effectiveness of our framework.

The structure of this chapter is organized as follows: Section 4.1 covers the preprocessing of WiFi signal, including environment-independent statistics extraction, motion segmentation, as well as the network input preparation. Section 4.2 introduces various deep neural networks and

the transfer learning. Experimental setups are detailed in Section 4.3. Section 4.4 evaluates the performance and we draw our conclusions in Section 4.5.

4.1 WiFi Signal Preprocessing

As CSI embodies the cumulative impact of the environment on signal propagation, it incorporates not just the influence of dynamic entities in the environment on the signal, but also the impact of static objects such as walls, furniture, and floors. Consequently, WiFi sensing systems that utilize CSI as direct input [52, 71, 73–75] are heavily subject to environmental influences. They exhibit subpar domain adaptation capabilities and necessitate data collection and retraining in new environments. This hampers their ability to be swiftly deployed and utilized in new environments.

Fig. 4.1 contrasts the CSI of different moving subjects: human movement (a,b), pet activity (c,d), and a robot vacuum cleaner operation (e,f). The first set (a, b, c) comes from Environment A, while the second set (d, e, f) from Environment B. The furniture in Environment B is moved between each data collection session. It's evident that environmental factors dominate CSI variations, substantially overriding the effects of different moving subjects, making it hard to extract motion-related features. Therefore, it's crucial to eliminate the dependency on environmental contexts for WiFi sensing systems to extract intrinsic statistics related to motion analytics and to enhance accuracy and deployment feasibility.

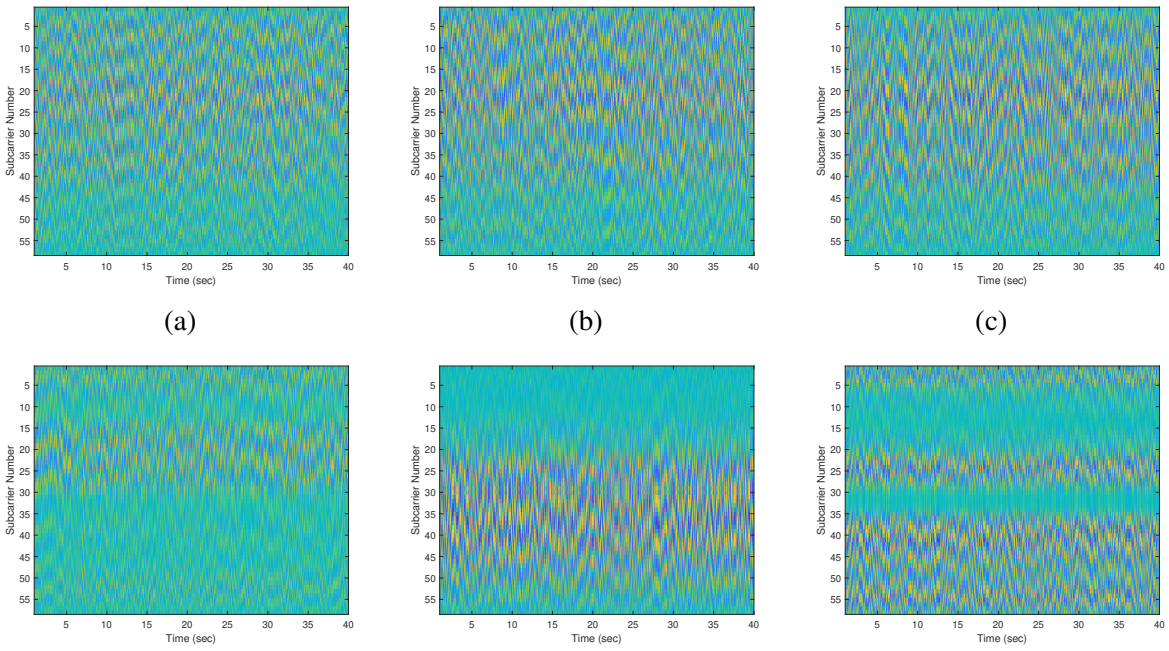


Figure 4.1: CSI of (a) human motion in Environment A, (b) pet motion in Environment A, (c) cleaning robot motion in Environment A, (d) human motion in Environment B, (e) pet motion in Environment B, and (f) cleaning robot motion in Environment B.

4.1.1 Environment-independent Statistic Extraction

In order to segregate the influences of static environments and dynamic entities on the signal, we first calculate the Auto-Correlation Function (ACF) of CSI that solely encapsulates the characteristics of dynamic subjects. Then, we designed a robust statistic, A-ACF, to amplify the characteristics of dynamic subjects and facilitate the analysis of motion patterns.

Specifically, we extract the ACF of $G(t, f)$ as:

$$\begin{aligned} \rho_G(\tau, f) &= \frac{\text{cov}[G(t, f), G(t+\tau, f)]}{\text{cov}[G(t, f), G(t, f)]} \\ &= \frac{\sigma_s^2(f)}{\sigma_s^2(f) + \sigma_n^2(f)} \rho_s(\tau, f) + \frac{\sigma_n^2(f)}{\sigma_s^2(f) + \sigma_n^2(f)} \delta(\tau), \end{aligned} \tag{4.1}$$

where τ is the time lag. $\sigma_s^2(f)$ and $\rho_s(\tau)$ is the variance and ACF of the propagated signal,

respectively [76]. The normalized channel gain at frequency f is defined as $w(f) = \frac{\sigma_s^2(f)}{\sigma_s^2(f)+\sigma_s^2(f)}$.

In order to better extract features of the moving subjects, we employ a Maximum Ratio Combine (MRC) approach on the ACF. This amalgamation across all subcarriers enhances the Signal-to-Noise Ratio (SNR) of the ACF, thereby accentuating the impact of the moving subject on the signal. The aggregated ACF after MRC is represented as follows:

$$\begin{aligned}\hat{\rho}_s(\tau) &= \sum_{i=1}^{N_s} w(f_i)\rho_G(\tau, f_i) \\ &= \sum_{i=1}^{N_s} \frac{\sigma_s^2(f)}{\sigma_s^2(f)+\sigma_s^2(f)}\rho_G(\tau, f_i),\end{aligned} \tag{4.2}$$

where N_s is the total number of subcarriers. When $\tau \to 0$, we have:

$$\lim_{\tau\to 0} \rho_G = w(f_i) \lim_{\tau\to 0} \rho_s(f_i). \tag{4.3}$$

Since the movement of the subject is continuous, we have $\lim_{\tau\to 0} \rho_s(f_i) = 1$ and then $w(f_i) = \lim_{\tau\to 0} \rho_G$. When the sounding rate F_s is high, we can estimate $w(f_i)$ by $\rho_G(\tau = \frac{1}{F_s}, f_i)$ [77]. Then the aggregated ACF can be estimated by:

$$\hat{\rho}_s(\tau) = \sum_{i=1}^{N_s} \rho_G(\tau = \frac{1}{F_s}, f_i)\rho_G(\tau, f_i). \tag{4.4}$$

Inspired by [57], we take the differential of aggregated ACF $\hat{\rho}_s(\tau)$ to amplify the speed information. Using $\Delta\rho(\tau)$ to denote $\frac{d\rho(\tau)}{\rho(\tau)}$, we express the A-ACF as $\Delta\hat{\rho}_s(\tau)$.

As depicted in Fig. 4.2, we present the results after extracting A-ACF from the CSI data shown in Fig. 4.1. By comparing Fig. 4.2 and Fig. 4.1, it becomes apparent that the motion characteristics of the same type of subject in different environments are similar. Meanwhile,

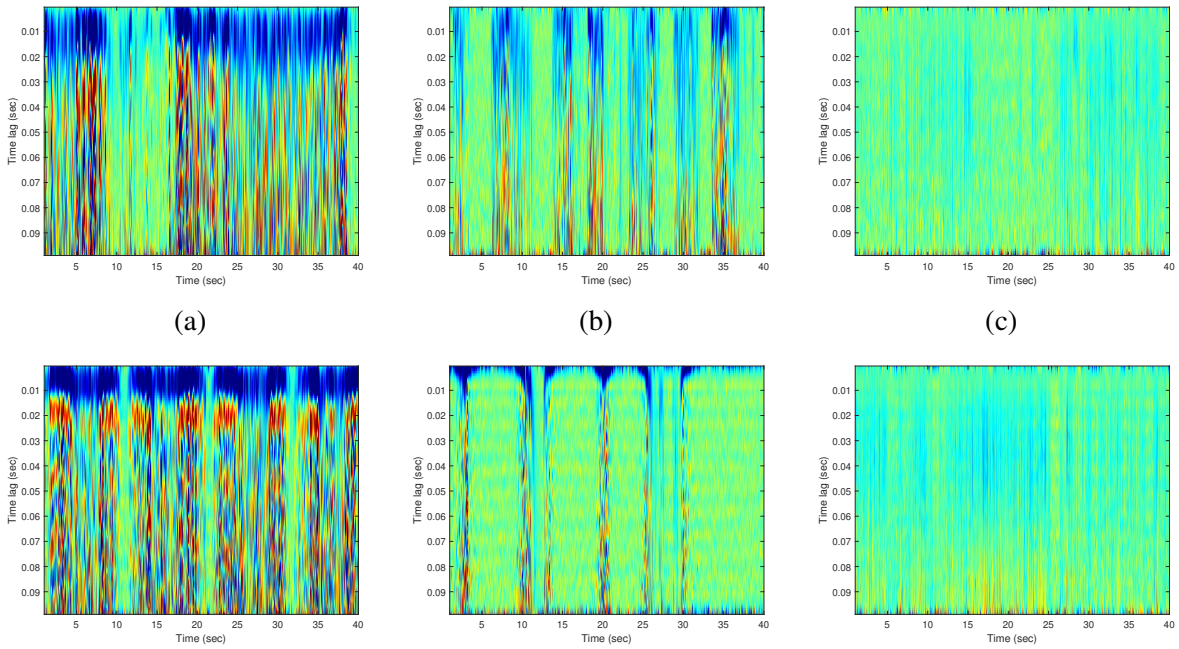


Figure 4.2: A-ACF of (a) human motion in Environment A, (b) pet motion in Environment A, (c) cleaning robot motion in Environment A, (d) human motion in Environment B, (e) pet motion in Environment B, and (f) cleaning robot motion in Environment B.

the A-ACF patterns of different moving subjects show significant discrepancies, irrespective of whether they are in the same or different environments.

As the A-ACF exclusively encapsulates the dynamic features associated with subject movement, devoid of any environmental or directional information of the subject, our WiFi sensing framework, utilizing A-ACF as input, is immune to variations in the environment and the positioning of subjects. Our framework focuses on extracting the more intrinsic features correlated to the movement of subjects, disregarding aspects such as the subject's orientation, position, and the surrounding environment. This independence empowers our framework to be rapidly and efficiently deployed within new environments.

4.1.2 Motion Detection and Segmentation

Subsequently, we derive the robust motion statistic from the A-ACF and utilize it to detect and segment motion-containing fragments. For the power response $G(t, f)$, the robust motion statistic derived from its A-ACF at time t over subcarrier f_i is defined as

$$\phi_G(f) \triangleq \rho_G\left(\tau = \frac{1}{F_s}, f\right), \tag{4.5}$$

where F_s is the sounding rate.

The robust motion statistic functions as a reliable gauge of movement presence or lack thereof within a given environment. In a stationary environment, the robust motion statistic $\phi_G(f)$ is close to 0, whereas in dynamic environments with movement, $\phi_G(f) > 0$. Following this, we partition the A-ACF into 5-second segments that are detected to encompass motion, as shown in Fig. 4.3. The extracted A-ACF segment is a 2D (two-dimensional) matrix with size $\mathbb{R}^{N\times T}$, where N is the total number of time lags at each time instance, and T is the number of time instances in a segment.

4.1.3 Input Preparation

In response to the varied model types, we further transform the 2D A-ACF segment into inputs that align with respective models. This mainly includes the following four cases:

Input for FNN. In feed-forward neural networks (FNNs), each sample is typically represented as a flattened, one-dimensional vector. The reason for this is that FNNs are designed to process each feature independently, without considering any inherent structure or correlation

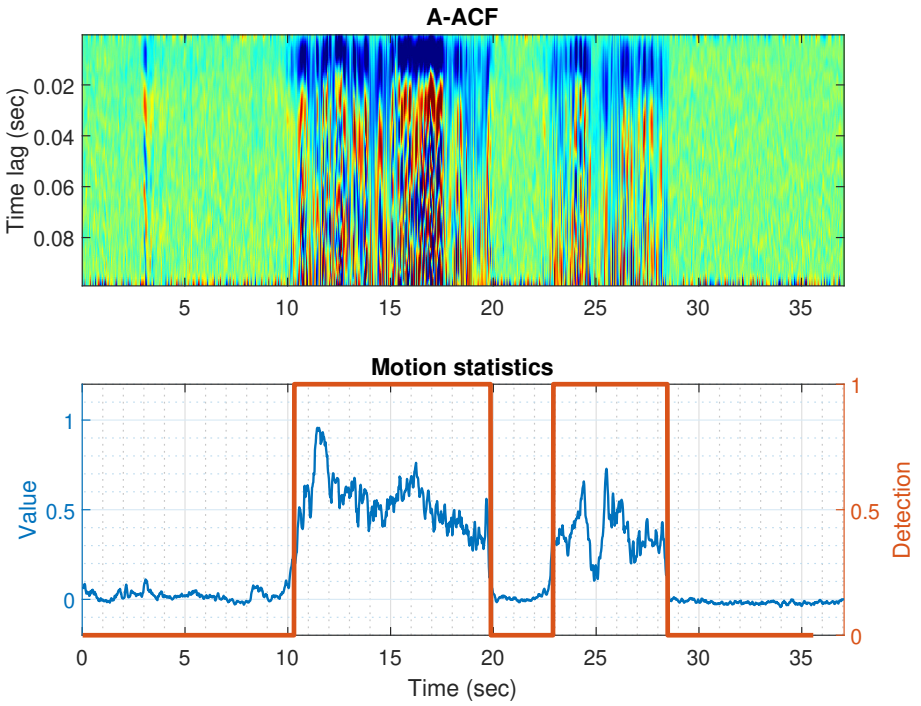


Figure 4.3: Motion detection and A-ACF segmentation with motion statistics.

between features. Hence, the 2D A-ACF segment $\mathbb{R}^{N\times T}$ is flattened into a 1D vector $\mathbb{R}^{NT}$ to fit the input requirement of FNNs.

Input for Image-based Models. Unlike FNNs, image-based models like convolutional neural network (CNNs) are designed to handle multi-dimensional data particularly, such as images, while preserving spatial relationships between pixels or features. Thus, we can directly feed the 2D A-ACF segment into image-based models.

Input for Language-based Models. Language-based models like recurrent neural networks (RNNs) are particularly suited to handle sequential data, where the order of inputs matters. They maintain a hidden state that can theoretically capture information about past elements in the sequence. Hence, the typical input for an RNN is a sequence. In this setup, the sequence of A-ACF vectors $\mathbb{R}^{N}$ is fed into the network one at a time.

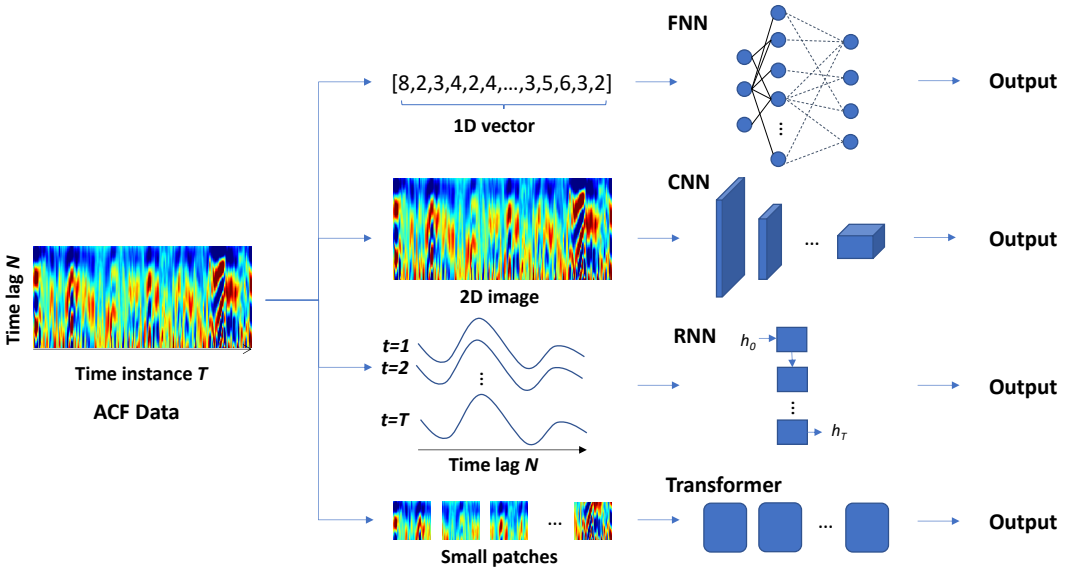


Figure 4.4: Illustration of data input preparation.

Input for Transformer-based Models. Transformer-based models have been applied for both Natural Language Processing (NLP) tasks and vision tasks [78]. For both kinds of tasks, the transformer models handle data in a similar sequential manner. For a 2D A-ACF segment, it is divided into P small patches $\mathbb{R}^{h \times w}$. These patches are then flattened into a 1D vector $\mathbb{R}^{hw}$. Positional embeddings are added to the vectors to provide information about the relative positions of patches in the original image.

Fig. 4.4 provides an intuitive depiction of the process by which A-ACF data is integrated into these models.

4.2 Deep Learning for Human and Non-human Recognition

In this section, we provide a succinct overview of various deep learning models that have demonstrated promising performance in computer vision and natural language processing domains. We delve into their advantages and limitations when applied to the identification

of human and non-human motions based on WiFi signals. Moreover, the concept of transfer learning, as well as its application to our problem domain, is discussed.

4.2.1 Forward Neural Network

FNN is one of the simplest types of artificial neural network. In an FNN, information moves in one direction only: from the input layer, through the hidden layers, and to the output layer. There are no loops in the network – information is always fed forward. Each neuron in a layer receives inputs from all neurons of the previous layer, processes the inputs using a weighted sum and a bias, applies an activation function to this sum, and passes the result to all neurons in the next layer.

The Multilayer Perceptron (MLP) [79] is a classic feed-forward neural network architecture and has been widely used in machine learning tasks. It takes the flattened A-ACF as input and maps the latent features into the categorical space with nonlinearity introduced by the activation function. Although MLPs are capable of modeling complex non-linear relationships in the data, and the hidden layers of the MLP extract increasingly abstract and meaningful features, MLPs don't explicitly consider spatial or temporal relationships in the data. This limitation makes MLPs less suited for modeling the spatial and temporal dependencies of human and non-human motion from A-ACF, but it is incorporated in other networks like CNNs as a classifier.

4.2.2 Convolutional Neural Network

CNNs are predominantly used in the field of computer vision, where they have achieved state-of-the-art results in image classification, object detection, and many other tasks. CNNs are

designed to automatically and adaptively learn spatial hierarchies of features from the input data. They consist of one or more convolutional layers, often followed by pooling layers, and then fully connected layers for classification or regression at the end. The convolutional layers are designed to recognize small, simple patterns, while the later layers recognize more complex structures.

For CNNs, the A-ACF is fed into the network as a 2D image, and a 2D convolutional kernel is applied with a sliding window mechanism to extrapolate feature maps from the A-ACF segment. Utilizing 2D CNNs for feature extraction from the A-ACF segment of motion offers numerous advantages. Initially, it captures motion characteristics with a 2D kernel that adeptly discerns the structure and features of the A-ACF segment from two dimensions. Moreover, by employing parameter sharing, they significantly reduce the number of parameters, boosting computational efficiency and mitigating the risk of overfitting. However, it's noteworthy that CNNs predominantly focus on local patterns and might overlook long-range dependencies or the global context present in the data. In this paper, we have primarily investigated the efficacy of LeNet [80], ResNet-18, ResNet-50, and ResNet-101 [81] in discerning human and non-human motion subjects based on WiFi.

4.2.3 Recurrent Neural Network

RNNs are specifically designed to work with sequence data. They achieve this by including loops in the network, which allow information to flow from one step in the sequence to the next. This makes RNNs capable of processing data such as time series, sentences, and more. However, standard RNNs often struggle to learn long-range dependencies due to vanishing and exploding gradients, which has led to the development of more advanced types of RNNs like LSTM [82]

and GRU (Gated Recurrent Unit) [83].

For RNNs, the A-ACF is interpreted as a time series signal. The network treats the A-ACF at each instance as a distinct token. RNNs are apt for understanding temporal dependencies within the A-ACF segment, making them an ideal choice for analyzing time-varying features of A-ACF induced by motion. However, they fall short in recognizing the characteristics of A-ACF at a specific moment, thereby neglecting the features of A-ACF resulting from the immediate short-term motion. In this study, we evaluated several popular RNN architectures, including the standard RNN and its variants such as GRUNet and LSTM, on our WiFi-based human and non-human recognition task.

4.2.4 Transformer

Transformers are a type of model that use self-attention mechanisms and have achieved state-of-the-art results on a variety of natural language processing and computer vision tasks. Unlike RNNs, transformers process the input data in parallel rather than sequentially, which are more efficient. They are designed to handle sequences and can capture complex patterns in the data, making them suitable for challenging tasks like machine translation, text summarization, and more.

To better capture the two-dimensional features of the A-ACF segment, we employ a Transformer model designed for vision tasks, Vision Transformer (ViT) [84]. ViT processes the input A-ACF segment by splitting it into fixed-size small image patches and reshaping them into vectors. These patch vectors are then linearly transformed into D-dimensional embeddings. To retain positional information, position embeddings are added to these patch embeddings. These

patch embeddings are then passed into a transformer encoder, consisting of several layers of multi-head self-attention modules and position-wise fully connected feed-forward networks. Finally, the output corresponding to the first token is used by the classification head to generate the final prediction for human and non-human motion classification.

4.2.5 Transfer from Pre-trained Models

Transfer learning is a machine learning method where a model, often pre-trained on a large dataset, is used as the starting point for a task of interest. It leverages knowledge learned from one problem domain (the source domain) to another related but different problem domain (the target domain). This is particularly useful when the target task has limited labeled data. It's a standard practice in deep learning, where models trained on large-scale image datasets are used for other vision tasks.

Due to the scarcity of labeled WiFi data and our transformation of WiFi signals into a 2D A-ACF image segment, we aim to investigate the feasibility of leveraging pre-trained image models in WiFi sensing tasks. Utilizing A-ACF segment images as the input for the transfer learning network, we transfer characteristics learned by deep neural networks from image data to the WiFi-based human and non-human motion identification task, to augment WiFi sensing performance. We assess the efficacy of transferring a ResNet-18 model pre-trained on the ImageNet dataset [85] to identify human and non-human motion based on WiFi signals, and juxtapose this with the performance of the ResNet-18 model trained explicitly with WiFi signals. Furthermore, we also examine the effect of fine-tuning different layers on the transfer performance. Detailed results are presented in Section 4.4.5.

4.3 Experimental Details

This section elaborates on the evaluation of our system. First, we introduce the evaluation methodologies, including experiment settings, data collection, and metrics.

4.3.1 Hardware

Our framework is composed of a pair of apparatuses, both equipped with commercially procurable WiFi network interface cards. As delineated in Fig. 4.5, one device plays the role of a transmitter (Tx), whereas another acts as the receiver (Rx). Each device is equipped with dual omnidirectional antennas, culminating in a total of four communication links for every transceiver unit. These pairs of antennas facilitate the streaming of CSI throughout 58 subcarriers. The framework operates within WLAN channel 153, which takes advantage of a carrier frequency positioned at 5.18 GHz and employs a bandwidth of 40 MHz. The Tx proactively deploys sounding frames at a sounding rate of 1500 Hz.

(a) Tx (b) Rx

Figure 4.5: (a) Tx, and (b) Rx in Scenario III.

4.3.2 Experimental Environment

We evaluate the proposed framework in 5 typical environments with a total of 29 different configurations, including a compact residential apartment, a townhouse, two single family houses, and an office building. The floor plans of these environments, together with the designated locations of the Tx (indicated in blue) and the Rx (highlighted in orange), are presented in Fig. 4.6. We collect data under both LOS and NLOS conditions to gauge the through-the-wall capabilities of the system. It is noteworthy that although we implement multiple configurations within a single scenario to bolster data diversity, we do not perform data fusion. The analysis solely employs single-link WiFi data. The varying distances between the Tx and Rx range from 2 to 8 meters.

4.3.3 Dataset and Metrics

We collect CSI data of four kinds of subjects from the 5 scenarios. The details of each kind of subjects' data is summarized as follows:

- **Human.** The human participants consist of 10 males and 3 females. The age of participants ranges from 23 to 34, and the height ranges from 154 cm to 194 cm. Participants are free to walk, run, sneak, or stop and have small motions during data collection such as using their phones while walking.

- **Pet.** The pet dataset is collected with 11 different pets, including 10 dogs and a cat, with the weight ranges from 17 lb to 85 lb. Pets are allowed to move freely.

- **Cleaning robot.** The cleaning robot's data is collected while it is doing routine cleaning

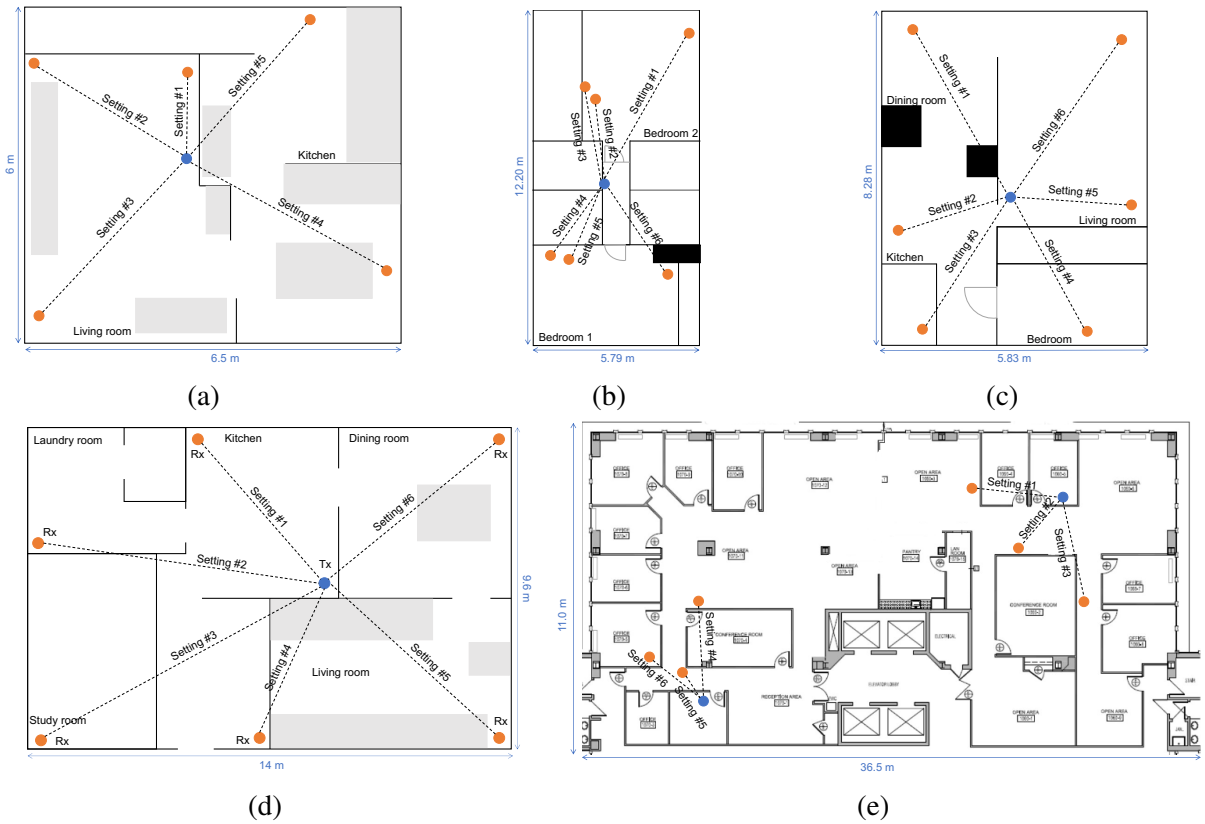


Figure 4.6: Floor plan of (a) Scenario I, an apartment, (b) Scenario II, a townhouse, (c) Scenario III, a single family house, (d) Scenario IV, a single family house, and (e) Scenario V, an office building.

tasks. An iRobot V3 vacuum machine is used as a cleaning robot in the experiments.

- **Fan.** The CSI of two kinds of fans, including the ceiling fan and the rotation fan, are collected while the fan is running.

In our data collection, certain human and pet subjects are involved in multiple scenarios, while others are restricted to a single scenario. Table 4.1 provides a summary of the data collected across five different scenarios. In this table, 'M' represents male, 'F' signifies female, 'D' stands for dog, and 'C' is used for cat. Any subjects that are exclusive to a particular scenario and do not appear in others are indicated in italics. The data collection duration varied across scenarios, amounting to approximately 150 days of total data.

Table 4.1: Summary of dataset

Scenario	Human	Pet	Cleaning robot	Fan
I	F1, M1, M2, M3, M4, M5, *M6, M7, M8, F2, F3*	D1, D2, *D3, D4, D5, D6, D7, D8*	iRobot V3	Rotation
II	F1	–	iRobot V3	–
III	M2, M3, *M9*	C1	–	–
IV	M1, M3, *M10*	D2	–	–
V	F1, M1, M2, M3, M4, M5	D1, *D9, D10*	iRobot V3	Rotation, *Ceiling*

To assess the accuracy of our model, we use the top-1 accuracy metric, defined as the percentage of motion segments correctly classified in the dataset. We also measure the model's computational requirements and complexity by recording the CPU prediction time. Peak memory usage and model size are monitored to estimate memory consumption. Given that our goal is to deploy the deep learning method for WiFi sensing on edge devices with limited computational resources, it's crucial to understand these computational and memory requirements thoroughly.

4.3.4 Implementation Details

We develop the neural networks using PyTorch [86], and conduct the training on a single NVIDIA GTX 2080. The models are optimized using the Adam optimizer [87], incorporating a warm-up strategy and a weight decay factor set at 0.001. The warm-up period lasts for 5 epochs, during which the learning rate gradually increases from 0 to $1e^{-4}$. We adopt a batch size of 64 during the training phase. To prevent overfitting, we implement an early stopping mechanism with a patience setting of 20 epochs. This means training would cease if there is no improvement observed over 20 consecutive epochs.

4.4 Evaluation

4.4.1 Classification Performance Evaluation

We thoroughly evaluate a variety of deep learning models by conducting five experiments based on different environmental settings. In each experiment, we designate data from four scenarios for training and validation, and the other one scenario for testing. The training dataset and validation dataset are randomly divided at a ratio of 8:2.

Table 4.2: Classification accuracy of deep learning models for human and non-human motion identification

Method	Exp. I	Exp. II	Exp. III	Exp. IV	Exp.V	Average
MLP	92.84%	93.35%	91.88%	86.48%	93.39%	91.59%
LeNet	94.64%	95.51%	93.90%	89.03%	94.91%	93.60%
ResNet-18	96.26%	96.34%	96.32%	93.24%	97.03%	**95.84%**
ResNet-50	97.19%	96.98%	96.45%	91.84%	97.65%	**96.02%**
ResNet-101	97.22%	97.14%	97.23%	92.86%	97.44%	**96.38%**
RNN	88.62%	84.50%	86.72%	81.25%	92.13%	86.64%
GRUNet	92.72%	92.74%	90.73%	81.25%	91.39%	89.77%
LSTM	86.34%	85.26%	84.86%	82.72%	89.78%	85.79%
ViT	94.21%	94.22%	93.05%	85.97%	94.54%	92.40%

We compare the accuracy of deep learning models on the unseen validation dataset, as illustrated in Table 4.2. The rightest column presents the average accuracy of each model across all five experiments. The MLP model exhibited consistent performance with an average accuracy of 91.59%. LeNet performed slightly better, with an average accuracy of 93.60%. The ResNet family of models demonstrates superior performance, especially ResNet-101, which achieves the highest average accuracy of 96.38%. Other deep learning models like RNN, GRUNet, and

LSTM performed comparatively lower, with average accuracies of 86.64%, 89.77%, and 85.79%, respectively. The ViT model also exhibited commendable accuracy, averaging 92.40%.

Overall, the results compellingly demonstrate the superiority of our proposed deep learning-based framework in accurately identifying human and non-human subjects through the wall with WiFi. The remarkable performance of the ResNet models, in particular, affirms the efficacy of the deep learning approach in complex detection tasks.

Next, we assess the model's performance in unseen environments from three distinct perspectives: recognizing seen subjects in unseen scenarios, identifying unseen subjects in unseen scenarios, and handling settings with multiple coexisting subjects.

4.4.2 Recognition in Unseen Environments for Seen Subjects

We begin by assessing our model's capacity to accurately identify seen subjects in unseen environments. We follow the leave-one-environment-out methodology and thus get five testing experiments. Fig. 4.7 depicts a box chart of the evaluation results, from which we discern that ResNet-18 achieves an impressive average testing accuracy of 91.71%. All ResNet models surpass the 90% mark in average testing accuracy. These outcomes firmly validate that our proposed architecture minimizes the impact of factors such as environment, position, or direction on recognition performance. This enables effective differentiation between human and non-human subjects in novel environments without necessitating additional training or parameter adjustments.

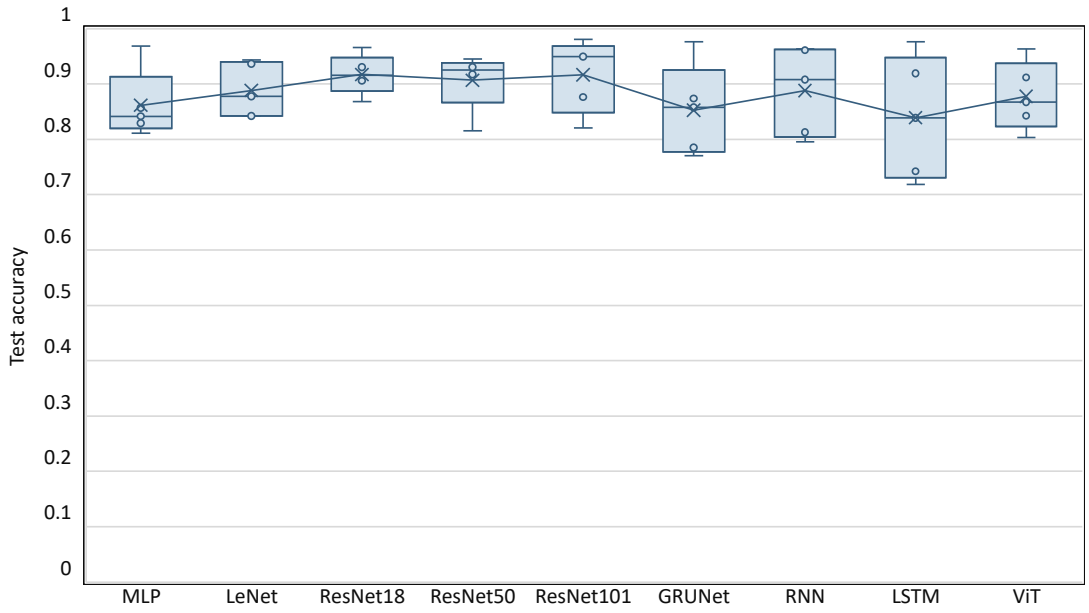


Figure 4.7: Recognition accuracy in unseen environment with familiar subjects.

4.4.3 Recognition in Unseen Environment for Unseen Subjects

Our evaluation continues by exploring the model's resilience to unseen subjects in unseen environments. We adhere to the same leave-one-environment-out training approach, and the subjects incorporated in the testing data have not been present in either the training or validation data sets. The testing accuracy is shown in Fig. 4.8.

Impressively, our proposed framework maintains high accuracy in differentiating between human and non-human subjects. This robustness can be credited to the A-ACF-based framework's efficiency in extracting universal patterns across different subjects. Despite slight variations in the motion patterns of different subjects, it captures universal features to distinguish between human and various non-human subjects. This robust recognition capability underpins our model's ability to be swiftly deployed in new environments with minimal user intervention.

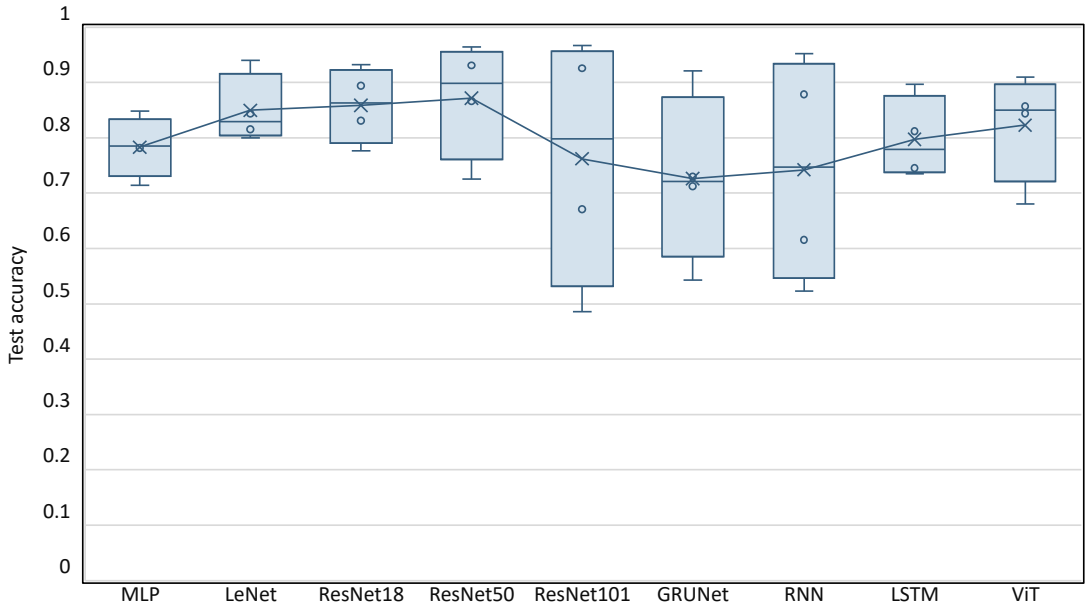


Figure 4.8: Recognition accuracy in unseen environment with unseen subjects.

4.4.4 Recognition in Unseen Environment for Coexisting Multiple Subjects

We further examine our framework's robustness when multiple subjects are moving together in unseen environments. We gather data from two scenarios, I and II, where two individuals or two dogs are simultaneously moving. Employing the same leave-one-environment-out approach, we assess the framework's ability to recognize multiple subjects in unknown environments, with the results presented in Fig. 4.9. The ViT model achieves a testing accuracy of 82.86%,

This performance signifies that our model can identify multiple subjects-whether they be humans or pets-even when they are moving simultaneously, eliminating the need for additional retraining or fine-tuning. This robustness further underscores the practical value and adaptability of our model in real-world scenarios.

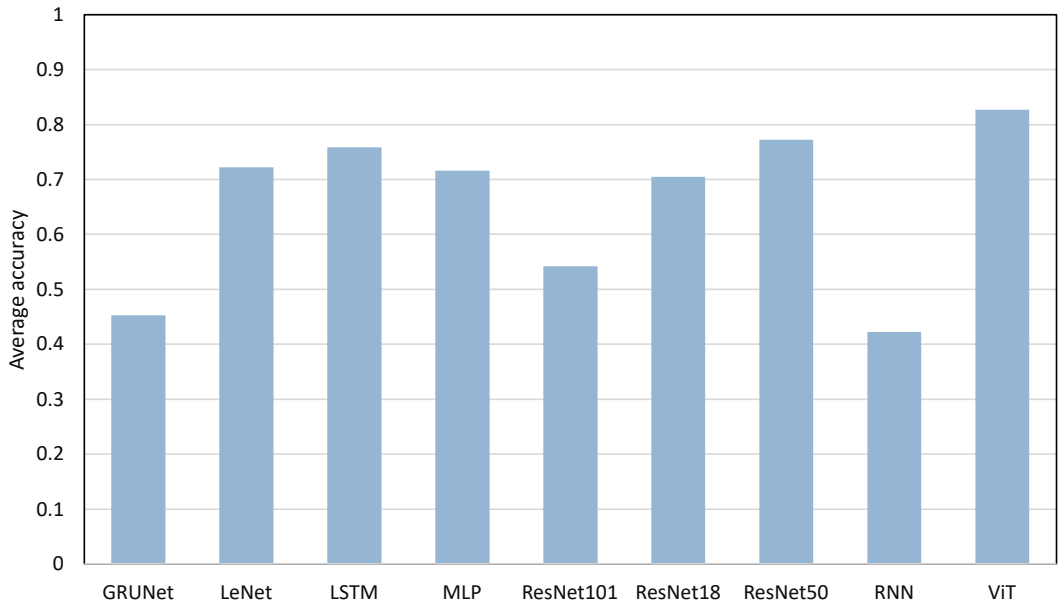


Figure 4.9: Recognition accuracy in unseen environment with multiple subjects coexisting.

4.4.5 Performance of Transfer Learning

To investigate whether transfer learning can enhance human and non-human target recognition based on WiFi by leveraging features learned from image recognition, we evaluate the accuracy of the ResNet-18 model, pre-trained on ImageNet, in five experiments consistent with Section 4.4.1. This is juxtaposed against the base ResNet-18 model trained exclusively with WiFi data. Furthermore, we examine the effects of model freezing at varying epochs on transfer learning outcomes, as presented in Table 4.3. The results suggest a slight accuracy increase when the number of frozen epochs is set at 10 and 50, improving by 0.65% and 0.48%, respectively. However, indefinite model freezing lead to a decline in accuracy. This negative transfer is attributed to the distributional differences between the target dataset, WiFi data, and the source dataset, image data.

Table 4.3: Classification accuracy evaluation on transfer learning

	Transferred			Plain
Freeze epochs	**10**	**50**	**All**	
Exp. I	96.44%	96.59%	94.74%	96.26%
Exp. II	97.04%	97.36%	94.52%	96.34%
Exp. III	97.09%	96.21%	94.61%	96.32%
Exp. IV	94.52%	94.45%	91.90%	93.24%
Exp. V	97.38%	97.01%	96.24%	97.03%
Average	**96.49%**	**96.32%**	94.40%	**95.84%**

4.4.6 Convergence

To ensure model selection, performance optimization, and robust generalization, we conduct a comprehensive evaluation and comparison of the convergence behavior among these models. In the evaluation of our various models over 100 epochs, we observe a compelling convergence pattern that is consistently exhibited across different architectures, as depicted in Fig. 4.10, Fig. 4.11, and Fig. 4.12.

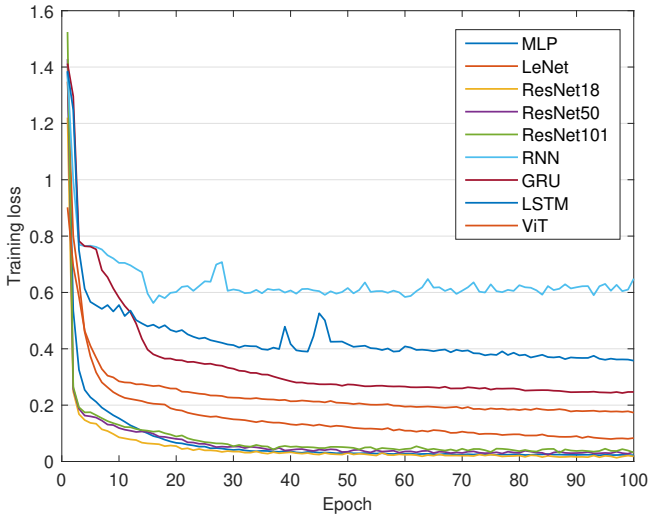


Figure 4.10: Training losses of deep learning models over 100 epochs.

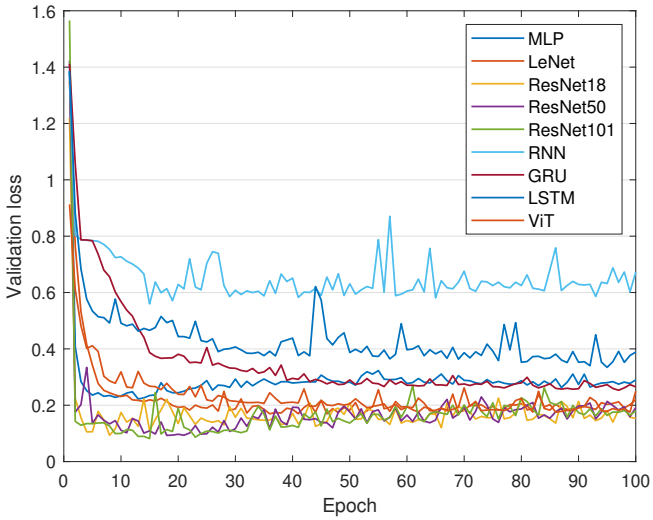


Figure 4.11: Validation losses of deep learning models over 100 epochs.

Fig. 4.10 illustrates a stable and gradual decline in training loss for all models, with CNNs showing lower training losses compared to RNNs, which may indicate a more efficient capture of spatial patterns by the CNNs. This pattern is mirrored in Fig. 4.11, where the validation loss follows a similar downward trajectory for all models, reflecting a well-balanced bias-variance trade-off and a controlled approach to overfitting. The consistency between training and validation losses across different architectures emphasizes the models' robustness in generalizing beyond the training data.

Fig. 4.12 further corroborates this trend, depicting a continuous increase in training accuracy for all evaluated models. Together, these patterns suggest that each model learns the underlying data patterns effectively, with nuanced differences like the lower loss in CNNs potentially pointing to architecture-specific advantages. The consistent convergence, represented across these three figures for all models, substantiates our confidence in the successful training of the models, tailored to the human and non-human motion identification with WiFi signals.

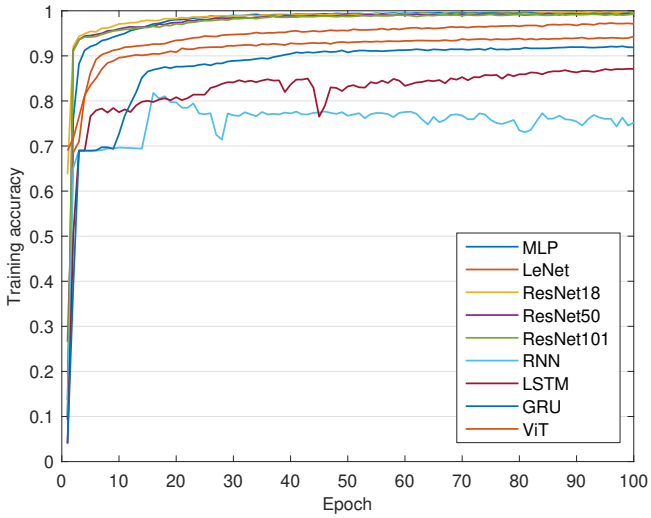


Figure 4.12: Training procedures of deep learning models regarding training accuracy.

4.4.7 Window Length

In Section 4.1.2, we elucidate the use of 5-seconds segments of A-ACF as network input. Here, we delve into the implications of using different lengths of A-ACF as network input across various networks. We examine the influence of A-ACF input segments with varying lengths of 5 s, 10 s, 15 s, and 20 s on the framework's accuracy and its ability to generalize. The performance of deep learning models with different lengths of A-ACF inputs is assessed in five unseen environments, with the average accuracy visualized in Fig. 4.13.

From Fig. 4.13, the models' performance generally increase with the length of the A-ACF segment. The ResNet-50 model achieves an accuracy of 99.77% when the segment length is 20 s. In practical use, the selection of A-ACF length needs to consider both the model's accuracy and the available computational resources. Longer segments consume more memory and computational resources, making them less practical for large-scale applications. Our findings suggest that the model's performance is robust to variations in segment length, and can be further

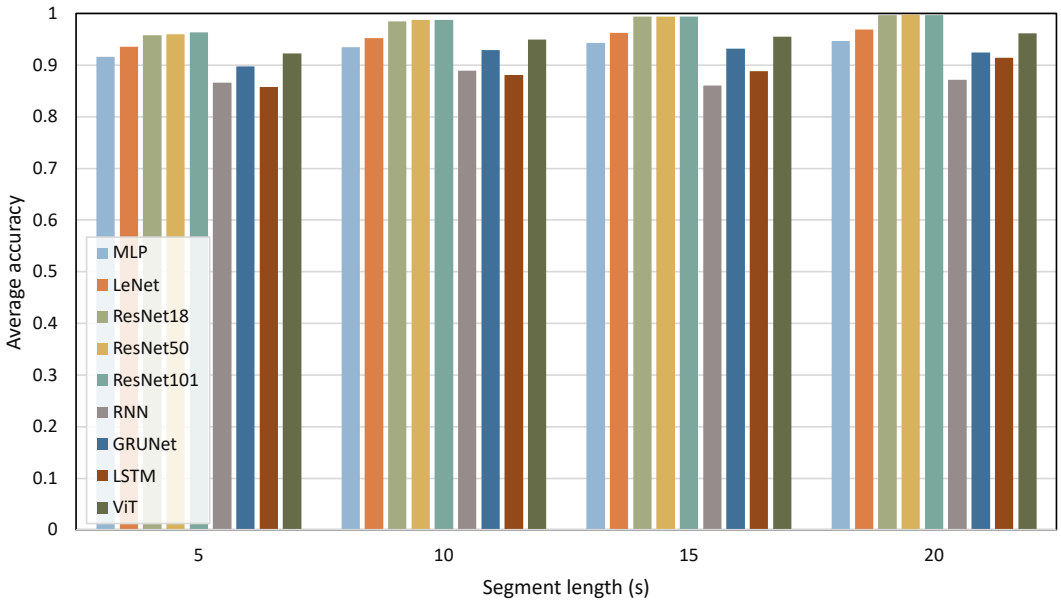


Figure 4.13: Evaluation on impact of motion segment length.

improved with longer segment length, offering practical flexibility for different applications.

4.4.8 Sounding Rate

In practical applications, different devices may operate at various sounding rates, necessitating an evaluation of the deep learning framework's robustness across these rates. Fig. 4.14 depicts the average accuracy of the models at 30 Hz, 150 Hz, and 1500 Hz. Remarkably, within the range of 150 Hz to 1500 Hz, models such as LeNet, ResNet and ViT consistently achieve high recognition accuracies, thereby substantiating the framework's robustness over a considerable span of sounding rates. This flexibility lends itself to applications with divergent sounding rate demands. Conversely, when the sounding rate plunges by a factor of 50 from 1500 Hz to 30 Hz, certain models exhibit a decline in recognition performance. This downturn is attributable to the substantial diminution of information content within the signal. Nevertheless, strategies such as prolonging the segment length or employing state machines could ameliorate

this degradation, underscoring the framework's adaptability and potential applicability across various scenarios.

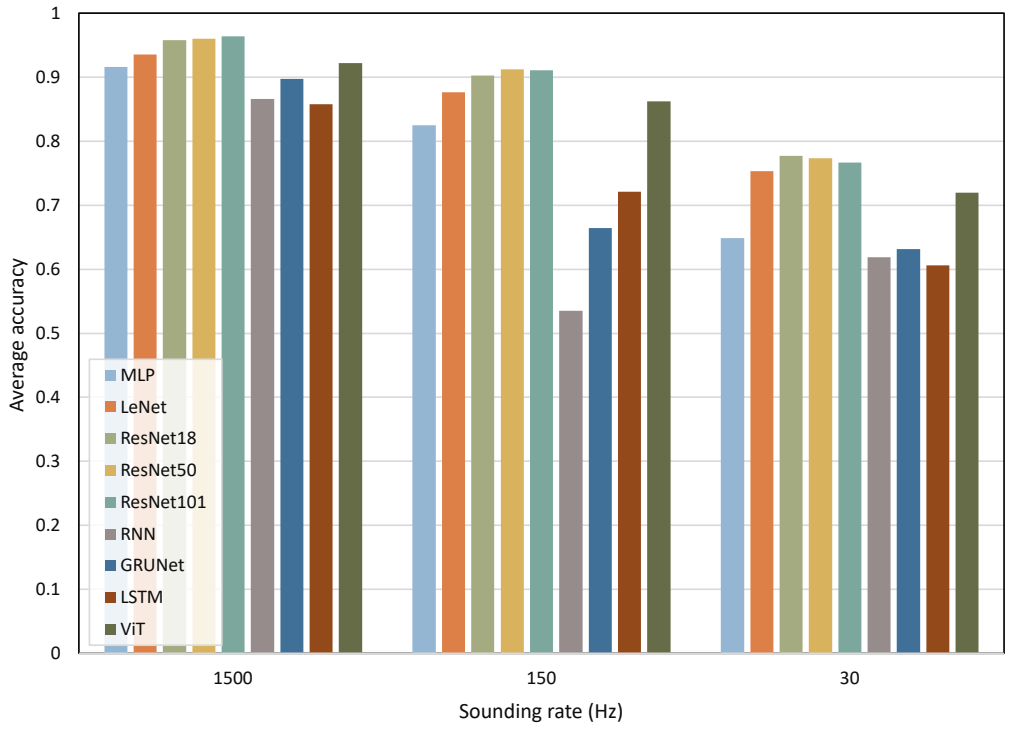


Figure 4.14: Evaluation on impact of sounding rate.

4.4.9 Computational Complexity and Model Parameter

We further assess the computational resource requirements of various identification networks for a 5-second motion segment with an i7 CPU, as displayed in Table 4.4. ResNet family, owing to its extensive parameters, necessitates greater CPU and memory resources. In contrast, shallower networks like LeNet exhibit lesser demand for CPU and storage. Hence, during real-world deployment, it is recommended to undertake a comprehensive assessment of computational capabilities alongside system performance.

Table 4.4: Computational resource requirements and model size

Model name	CPU prediction time (ms)	Peak memory usage (MB)	Model size (MB)
MLP	5.612	0.053	145
LeNet	15.847	0.082	14.7
ResNet-18	21.57	0.841	42.7
ResNet-50	50.896	1.496	81.3
ResNet-101	84.439	2.87	158
GRU	29.514	11.36	0.24
RNN	6.269	3.151	0.082
LSTM	13.607	0.055	0.055
ViT	8.515	0.571	42.7

4.5 Summary

In this chapter, we present a pioneering deep learning framework capable of distinguishing between human and non-human entities through walls using single-link WiFi. While prevailing intelligent systems grapple with interference stemming from non-human movements, our system adeptly identifies a spectrum of non-human subjects. It achieves this by leveraging a deep neural network and using the robust, environment-, location-, and direction-agnostic statistic, A-ACF, as input. Rigorous experiments conducted in diverse settings with an array of subjects scrutinize the performance of prominent deep learning models and the effectiveness of transfer learning. Our evaluation results not only affirm the system's capability to discern human and non-human entities with high accuracy in challenging scenarios but also offer insights into selecting appropriate deep learning models and utilizing transfer learning for WiFi sensing tasks.

Chapter 5: Through-the-wall Indoor Intrusion Detection

Indoor intrusion detection systems, crucial for indoor security, have been variously developed using camera surveillance, audio detection, and near-infrared technologies. However, each approach bears inherent limitations. Camera and audio-based systems, while accurately detecting intrusions, present significant privacy concerns. Near-infrared systems, better for privacy, require precise placement and are sensitive to environmental factors like temperature, resulting in a high false alarm rate and reducing user confidence. Additionally, both camera and near-infrared devices work primarily under Line-Of-Sight (LOS) conditions, thus limiting coverage, and necessitating supplementary equipment, making the setup process time-consuming and labor-intensive.

With the proliferation of IoT devices, WiFi has become pervasive, leading to innovative attempts at utilizing WiFi signals for indoor intrusion detection. In contrast to camera, sound, and near-infrared-based systems, WiFi-based intrusion detection devices offer superior privacy protection, extensive coverage, and eliminate the need for extra installations. However, these WiFi-based systems often overlook that disturbances in WiFi signals can be caused not only by humans but also by non-humans, resulting in a heightened probability of false alarms. While some systems attempt to mitigate false alarms from such non-human disturbances, they impose strict requirements on device placement and environmental conditions. In addition, they are

mainly effective under LOS conditions, not accounting for performance under Non-Line-Of-Sight (NLOS) conditions.

More importantly, the current attempts to filter non-human interference from intrusions fail to adequately consider non-human factors within indoor environments. They struggle to effectively differentiate between human and various non-human movements, which results in a high incidence of false alarms in practical use, deterring widespread adoption. In any indoor setting, various non-human elements can cause movement, including pets, robotic vacuum cleaners, and household appliances such as fans. Given that over 70% of American households have pets, robotic vacuum cleaners are growing in popularity, and fans are a staple in many homes, the ability to discern between the movements of these non-human entities and human movements is paramount to the robustness and practicality of an indoor intrusion detection system.

In this chapter, we introduce the first WiFi and deep-learning-powered indoor intrusion detection system, *Wi-IntruNet*, enhanced by human and non-human object differentiation. *Wi-IntruNet* discerns human intrusions through walls with single-link commercial WiFi, eliminating the need for supplementary devices and adeptly handling multiple concurrent human intruders. Utilizing a neural network-based model for distinguishing between human and non-human entities, *Wi-IntruNet* effectively mitigates interference caused by non-human movements. By integrating current spatial information with historical temporal information, *Wi-IntruNet* provides a comprehensive assessment of potential intrusions. Moreover, *Wi-IntruNet* focuses on extracting movement characteristics that are independent of the environment, ensuring that the performance of *Wi-IntruNet* is unaffected by variables such as location, angle, direction, and environmental changes.

The structure of this chapter is organized as follows: Section 5.1 introduces the challenges and framework of *Wi-IntruNet*. The preprocessing of WiFi signals and the preparation of network inputs are described in Section 5.2. Section 5.3 presents the human and non-human feature extractor based on ResNet-18 [81] and the intrusion detector based on Long Short-Term Memory (LSTM) [82]. System development and performance evaluation are covered in Section 5.4, followed by a discussion of the system in Section 5.5. Finally, the chapter is concluded in Section 5.6.

5.1 Overview

5.1.1 Challenges

Achieving a comprehensive indoor intrusion detection system poses numerous challenges, three of which are highlighted here.

First, both human and non-human moving objects within the environment influence the WiFi signal, making it particularly challenging to use single-link commercial WiFi under LOS and NLOS conditions to filter out non-human interference to the intrusion system. Although different sizes of human and non-human objects lead to varying influences on the CSI when moving at the same location, non-human objects can cause interference comparable to humans when they are closer to the WiFi device. In this study, we tackle this issue by leveraging deep learning with the ResNet-18 model to effectively discern human and non-human movement features, thereby filtering out non-human movements.

Second, current WiFi-based deep learning networks exhibit significant dependence on the environment and targets, rendering them ineffective when introduced to new environments or

targets. Although a multitude of domain adaptation techniques has been employed to counteract environmental changes, these techniques require model training within the new environment, necessitating significant user participation—a process that is both laborious and time-consuming. To tackle this issue, we use the A-ACF of CSI as our network input. This approach makes the extracted features environment-independent, ensuring that our trained network remains unaffected by environmental changes and can be readily deployed in new settings.

Third, classification-based neural networks predominantly focus on identifying the current target, thereby overlooking the temporal correlation of the target. It's challenging to merge information from both temporal and spatial dimensions to accurately determine the identity of subjects presented currently. While most existing models consider signal correlation across time and spatial dimensions, their objectives are typically confined to identification tasks, neglecting the temporal correlation of identification results. In this study, we design a state machine based on LSTM that integrates the temporal and spatial correlations of movement. This allows for accurate determination of the current target based on temporal information, even under conditions where target identification may be challenging.

5.1.2 WiFi-based Intrusion Detection Framework

Fig. 5.1 illustrates the comprehensive workflow of our system. Initially, we preprocess the WiFi signals by computing the A-ACF of the CSI. This allows us to derive the statistical information encapsulating the influence of dynamic targets on the propagated signal. During this preprocessing phase, we segment the A-ACF fragments associated with motion based on the statistical characteristics of the A-ACF. Subsequently, the A-ACF serves as the input of the

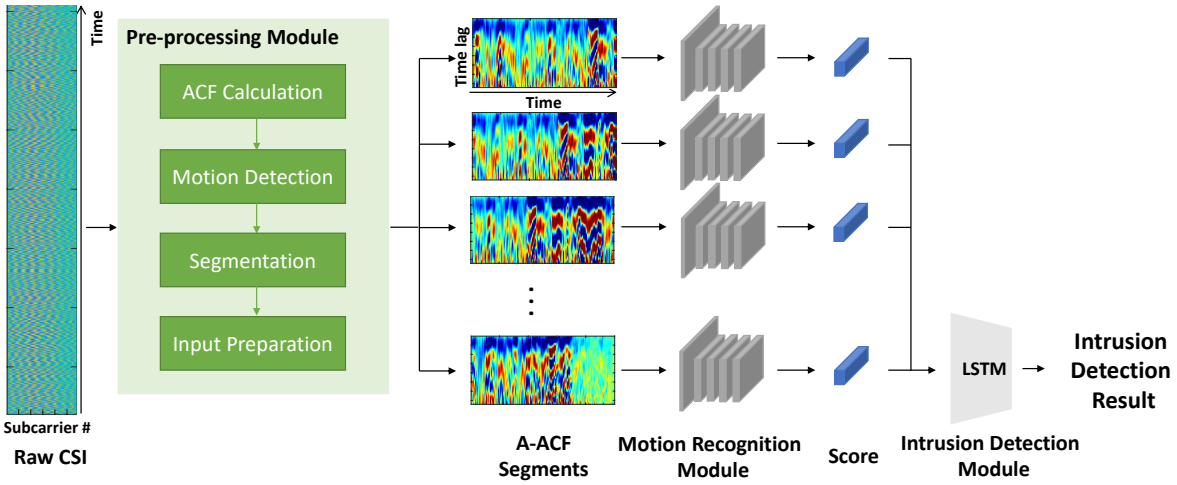


Figure 5.1: Overview of *Wi-IntruNet*.

motion recognition module (MRM) to distinguish the subject. This module employs a neural network based on ResNet-18 to analyze the impact of various moving subjects-such as humans, pets, and robot cleaners-on signal propagation, and extract their motion characteristics for identification purposes. Ultimately, we design an intrusion detection module (IDM) based on the LSTM network that integrates the currently identified spatial features with temporal features of the target's presence, thereby enabling efficient detection of intrusions in the environment.

5.2 WiFi Signal Preprocessing

5.2.1 A-ACF Calculation

In order to segregate the influences of static environments and dynamic entities-people and objects-on the signal, we extract the A-ACF that solely encapsulates the characteristics of dynamic objects to serve as the network input. Specifically, we initiate by calculating the ACF of

power response $G(t, f)$ by

$$\begin{aligned}\rho_G(\tau, f) &= \frac{\text{cov}[G(t, f), G(t+\tau, f)]}{\text{cov}[G(t, f), G(t, f)]} \\ &= \frac{\sigma_s^2(f)}{\sigma_s^2(f) + \sigma_n^2(f)}\rho_s(\tau, f) + \frac{\sigma_n^2(f)}{\sigma_s^2(f) + \sigma_n^2(f)}\delta(\tau),\end{aligned} \tag{5.1}$$

where τ is the time lag. $\sigma_s^2(f)$ and $\rho_s(\tau)$ is the variance and ACF of the propagated signal, respectively [76]. Then, we employ a Maximum Ratio Combine (MRC) approach on the ACF [77] and the aggregated ACF can be estimated by:

$$\hat{\rho}_s(\tau) = \sum_{i=1}^{N_s} \rho_G(\tau = \frac{1}{F_s}, f_i)\rho_G(\tau, f_i). \tag{5.2}$$

Inspired by [57], we take the differential of aggregated ACF $\hat{\rho}_s(\tau)$ to amplify the speed information. Using $\Delta\rho(\tau)$ to denote $\frac{d\rho(\tau)}{\rho(\tau)}$, we express the A-ACF as $\Delta\hat{\rho}_s(\tau)$.

As the A-ACF exclusively encapsulates the dynamic features associated with object movement, devoid of any environmental or directional information of the object, our WiFi sensing system, utilizing A-ACF as input, is immune to variations in the environment and the positioning of targets. Our system focuses on extracting the more intrinsic features correlated to the movement of objects, disregarding aspects such as the object's orientation, position, and the surrounding environment. This independence empowers our system to be rapidly and efficiently deployed within new environments.

5.2.2 Motion Detection and Segmentation

Subsequently, we derive the motion statistic via the ACF and utilize it to detect and segment motion-containing fragments. The motion statistic $\phi(f)$ for a subcarrier with frequency f is defined as ACF of the CSI $H(t, f)$ with a time lag of $\tau = 1/F_s$, where F_s is the sounding rate [29]. That is,

$$\phi(f) \triangleq \rho_H \left(\tau = \frac{1}{F_s}, f \right). \tag{5.3}$$

Motion statistics function as a reliable gauge of movement presence or lack thereof within a given environment. In a stationary environment, the motion statistic $\phi(f)$ is close to 0, whereas in dynamic environments with movement, $\phi(f) > 0$. We partition the A-ACF fragments, detected to encompass motion, into segments with a time length of Ts. Experimental results led us to opt for $T = 5$ for each segment, a decision made to maintain system performance while also shortening real-time detection time and maximally conserving computational resources.

5.3 Intrusion Detection Network

5.3.1 Motion Recognition Module

Upon partitioning the A-ACF of the CSI into segments, we utilize these segments as input for MRM designed to distinguish between human and non-human targets, as illustrated in Fig. 5.1. The size of an A-ACF input segment is $\mathbb{R}^{N_t \times T}$, where T denotes the number of time instances and N_t denotes the number of time lag of one instance. This matrix can be interpreted as a 2D (two-dimensional) spectrum image or as an encoded time series. Modern deep neural networks have demonstrated impressive capabilities in feature extraction from both images and time series

[88]. Therefore, we propose to employ a deep neural network for WiFi-based human and non-human feature extraction.

However, given the distinct nature of the A-ACF spectrum, which diverges from conventional images or time series, existing network architectures may not always yield satisfactory results in WiFi recognition tasks. Contrary to images, which typically possess three RGB channels, the A-ACF spectrum is limited to a singular channel and suffers from lower spatial resolution. Unlike standard encoded time series, the A-ACF function at each instant encapsulates the movement information of the target at that specific moment, deviating from embedding. Consequently, it becomes essential for us to evaluate various network architectures in order to pinpoint an appropriate deep learning model, one that is optimally compatible with a WiFi sensing system that employs the A-ACF spectrum as its input.

The time series of A-ACF vectors encapsulate critical information regarding the speed, volume size, and motion intensity of moving targets in the environment [36, 57]. Different networks can extract information from distinct facets of the training data. Our objective is to identify the optimal deep neural network model and the corresponding neural architecture that can offer robust WiFi-based human and non-human recognition across diverse environments and targets. We have evaluated the performance of popular neural networks, including CNNs, RNNs and Transformer, in discerning between human and non-human targets, with a detailed analysis presented in Section 5.4.3. Ultimately, we select the ResNet-18 model, which demonstrats consistent performance across a variety of environments and targets and requires relatively less computational resources.

5.3.2 Intrusion Detection Module

In the prior step, we segment the WiFi signal, identifying the moving targets within each segment based on the characteristics of their A-ACF. However, we overlook the temporal correlation amongst the A-ACF segments. The MRM does not consider the time dependency that exists between A-ACF segments. For instance, if a moving object is detected and recognized in an environment, it is highly probable that the motion detected subsequently will be due to this same object. By incorporating the historical information on target detection in the environment, we can more accurately ascertain the presence of intrusions and filter out false alarms triggered by animals.

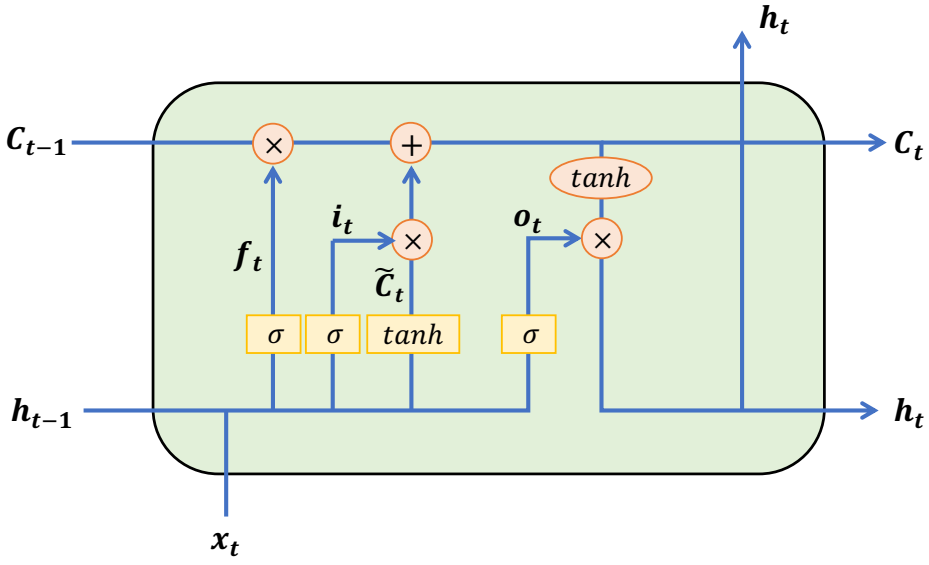


Figure 5.2: An illustration of LSTM.

Inspired by the impressive performance of ConvLSTM [89] in video classification and LSTM's effectiveness in time series signal classification, we design a temporal information extractor rooted in LSTM, as depicted in Fig. 5.2. Its input comprises the probability values

outputted by MRM from current and past A-ACF segments, while its output is a determination of the presence of an intrusion. An LSTM unit is composed of a cell with a memory state C_t, an input gate i_t, an output gate and a forget gate f_t. Denoting the embedding from MRM at current time t as x_t, the forward pass of the LSTM is:

$$
\begin{aligned}
i_t &= \sigma\left(x_t U^i + h_{t-1} W^i\right) \\
f_t &= \sigma\left(x_t U^f + h_{t-1} W^f\right) \\
o_t &= \sigma\left(x_t U^o + h_{t-1} W^o\right) \\
\tilde{C}_t &= \tanh\left(x_t U^g + h_{t-1} W^g\right) \\
C_t &= \sigma\left(f_t * C_{t-1} + i_t * \tilde{C}_t\right) \\
h_t &= \tanh\left(C_t\right) * o_t,
\end{aligned}
\tag{5.4}
$$

where $W \in \mathbb{R}^{h \times d}$ and $U \in \mathbb{R}^{h \times h}$ are weight matrices, with superscripts d and h referring to the number of input features and number of hidden units, and σ is the Sigmoid function.

The LSTM network learns the relationship between the current and past probability outputs of MRM, thereby enabling a more robust assessment of whether an intrusion is present. In Section 5.5.1, we evaluate the performance of the LSTM-based IDM, demonstrating that it can substantially reduce the likelihood of false alarms triggered by non-human subjects.

5.4 Implementation and Evaluation

This section elaborates on the implementation and evaluation of our system. First, we introduce the implementation details and evaluation methodologies, including experiment settings, data collection, and metrics. Then, we reveal the intrusion detection performance from

(a) Tx (b) Rx

Figure 5.3: (a) Tx, and (b) Rx in Scenario III.

various aspects.

5.4.1 Hardware and Experimental Environment

Our system consists of two prototype devices, each outfitted with commercially available WiFi network interface cards. As illustrated in Fig. 5.3, one prototype serves as the transmitter (Tx), while the other functions as the receiver (Rx). Each prototype is equipped with two omnidirectional antennas, yielding a total of four links for each transceiver. Each pair of antennas streams CSI over 58 subcarriers. The system operates on WLAN channel 153, utilizing a carrier frequency of 5.18 GHz and a bandwidth of 40 MHz. To capture precise motion information, the Tx dispatches sounding frames at a high channel sampling rate of 1500 Hz.

We deploy *Wi-IntruNet* across five diverse scenarios, spanning 29 unique setups. These range from compact apartments to multi-roomed office buildings. The floorplans for these test scenarios along with the locations of the Tx (blue) and the Rx (orange) are shown in Fig. 5.4. We collect both LOS and NLOS data to test the through-the-wall performance of *Wi-IntruNet*. Please note that we employ multiple setups in one scenario to enhance the data diversity, but evaluate

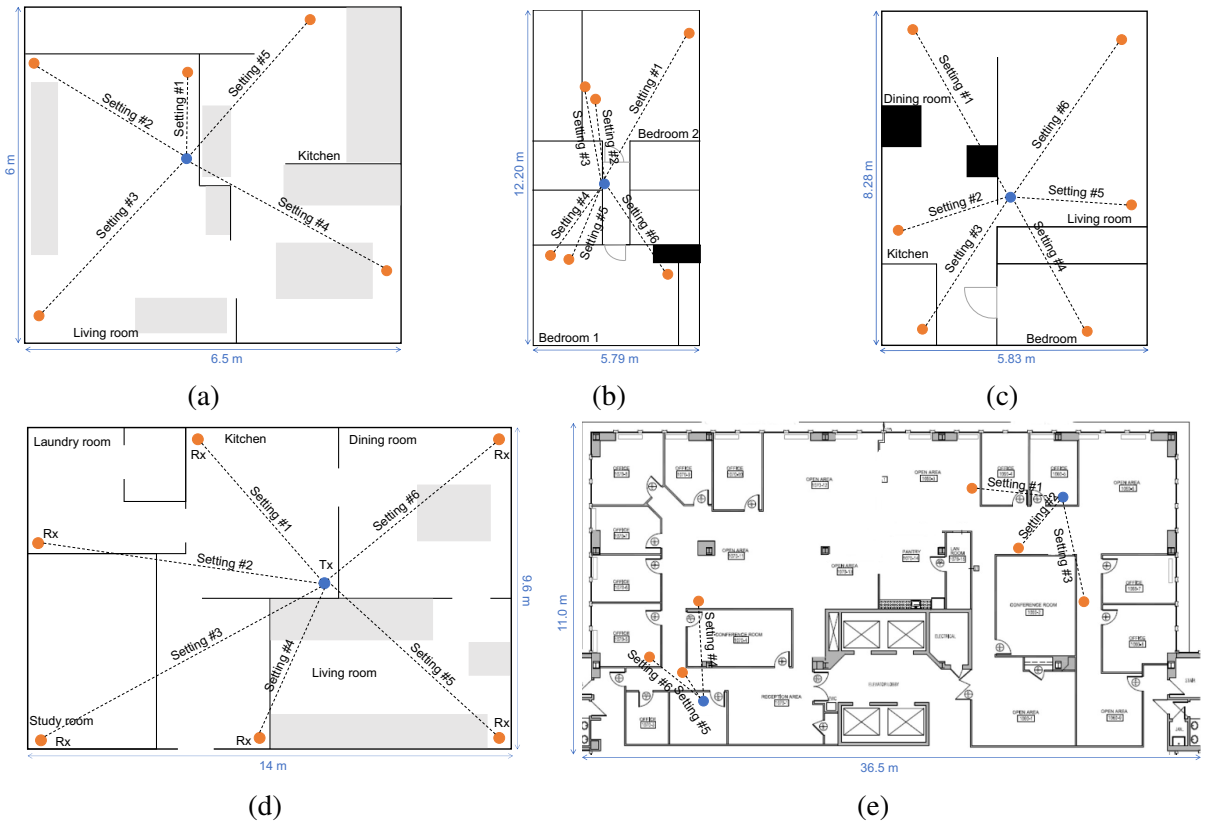


Figure 5.4: Floor plan of (a) Scenario I, an apartment, (b) Scenario II, a townhouse, (c) Scenario III, a single family house, (d) Scenario IV, a single family house, and (e) Scenario V, an office building.

the performance only using single-link data without data fusion. The distances between Tx and Rx vary from 2 to 8 m.

5.4.2 Dataset and Metrics

We collect CSI data of four kinds of subjects from the 5 scenarios. The details of each kind of subjects' data is summarized as follows:

- **Human.** The human participants consist of 10 males and 3 females. The age of participants ranges from 23 to 34, and the height ranges from 154 cm to 194 cm. Participants are free to walk, run, sneak, or stop and have small motions during data collection such as using

their phones while walking.

- **Pet.** The pet dataset is collected with 11 different pets, including 10 dogs and a cat, with the weight ranges from 17 lb to 85 lb. Pets were allowed to move freely.

- **Cleaning robot.** The cleaning robot's data is collected while it is doing routine cleaning tasks. A iRobot V3 vacuum machine is used as a cleaning robot in the experiments.

- **Fan.** The CSI of two kinds of fans, including the ceiling fan and the rotation fan, are collected while the fan is running.

Several humans and pets participate in multiple scenarios, whereas some appear in only one. A comprehensive summary of the data from the five scenarios can be found in Table 5.1. The data collection duration fluctuates across scenarios, cumulatively accounting for approximately 160 days of data. We utilize the data from Scenarios I, II, and III for training and validation purposes, whereas the datasets from Scenarios IV and V serve to assess the performance of *Wi-IntruNet* in unseen environments.

Table 5.1: Summary of dataset

Scenario	Dataset	Human	Pet	Cleaning robot	Fan
I	Train Validation	3 Females and 7 Males	8 Dogs	iRobot V3	Rotation
II		1 Female	–	iRobot V3	–
III		3 Males	1 Cat	–	–
IV	Test	2 Males	1 Dog	–	–
V		1 Female and 3 Males	3 Dogs	iRobot V3	Rotation, Ceiling

We measure the classification efficacy of *Wi-IntruNet* using top-1 accuracy. For its intrusion detection capabilities, we assess using the Intrusion Detection Rate and False Alarm Rate metrics.

The computational demands and complexity are gauged via CPU prediction time and FLOPS. Memory efficiency is evaluated by examining peak memory usage, parameter size, and overall model size.

5.4.2.1 Network Implementation

The networks are implemented with PyTorch [86] and are trained on one NVIDIA GTX 2080. Table 5.2 shows the networks' architecture. The models are optimized with the Adam optimizer [87] with a warm-up strategy and a weight decay of 0.001. For the ResNet-18-based motion recognition network, we employ a batch size of 32, while for the LSTM-based state machine, the batch size is set as 64. The warm-up epoch is 5 and the learning rate increases from 0 to 1e-4. We apply early stopping with a patience value of 20 to prevent over-fitting.

5.4.3 Evaluation on Classification Performance

The target identification capability of a model reflects its proficiency in extracting motion features of the target. To assess this, we first gauge the identification accuracy of various neural network models, as presented in Table 5.3. This accuracy is the average over five distinct environments. For testing accuracy evaluation, we adopt a leave-one-environment-out methodology, offering a true representation of the model's performance in unfamiliar scenarios. Note that we do not make any parameter adjustments to the model when assessing its performance in unseen environments. The average testing accuracy is derived by averaging each model's testing accuracy across the five different unseen environments.

Upon comparison, we discover that the ResNet family consistently exhibits the most stable

Table 5.2: Architecture of *Wi-IntruNet*

Layer name	**MRM**	**IDM**
1	Conv, $7 \times 7, 64$, Stride 2	LSTM 200 cells
2	Max Pool, 3×3, Stride 2	–
	Res-block $\begin{bmatrix} \text{Conv}, 3 \times 3, 64 \\ \text{Conv}, 3 \times 3, 64 \end{bmatrix} \times 2$	
3	Res-block $\begin{bmatrix} \text{Conv}, 3 \times 3, 128 \\ \text{Conv}, 3 \times 3, 128 \end{bmatrix} \times 2$	–
4	Res-block $\begin{bmatrix} \text{Conv}, 3 \times 3, 256 \\ \text{Conv}, 3 \times 3, 256 \end{bmatrix} \times 2$	–
5	Res-block $\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 2$	–
6	Average Pool, 7×7	–
7	Fully Connections, 512×1000	–
8	Softmax	–

performance across various environments, showing average accuracies of over 95%. This can be attributed to its superior extraction of target motion features, such as speed and gait, from the A-ACF. As these features bear no relation to the environment or the environmental noise, the model maintains high accuracy even in unknown settings. Considering both accuracy and computational complexity, we selecte the ResNet-18 to undertake the intrusion detection task.

5.4.4 Evaluation on Intrusion Detection Performance

We further probe into the intrusion detection accuracy of Scenario IV and Scenario V. We utilize the ResNet-18 model, trained with data from Scenario I, II, and III, and evaluate its intrusion detection capabilities in the unseen Scenario IV and V. In the unseen environment, we

Table 5.3: Classification accuracy of MRM for human and non-human motion identification

Method	**Validation**	**Testing**
MLP	91.59%	86.14%
LeNet	93.60%	88.83%
ResNet-18	**95.84%**	**91.71%**
ResNet-50	96.02%	90.67%
ResNet-101	96.38%	91.66%
RNN	86.64%	88.82%
GRUNet	89.77%	85.25%
LSTM	85.79%	83.90%
ViT	92.40%	87.77%

gather intrusion data from four volunteers, three dogs moving around, a robotic vacuum cleaner, and a rotating fan. The specifics of the intrusion detection dataset can be seen in Table 5.1. The six volunteers participate in varied activities such as free movement, running, and item searching within the environment. The three dogs are engaged in free movement, running, food searching, and barking. The data for the robotic vacuum cleaner is collated during its cleaning operations, and the fan's data is gathered while it is operational. Each volunteer's intrusion CSI segment has varied durations ranging from 1 to 5 minutes. Among them, three volunteers and two dogs are observed in training scenarios and thus are seen by the model in training and validation environments, while the data from the other three volunteers and two dogs is unseen by the model.

5.4.4.1 Intrusion Detection Performance in Unseen Environment with Seen Subjects

We first assess the model's ability to detect intrusions of known targets in unfamiliar environments. As detailed in Table 5.4, the model boasts an impressive intrusion detection rate of 97.92% and maintains a false alarm rate of 0%. Notably, the presence of pets and robotic vacuum cleaners in the environment didn't result in any false alerts. This is because *Wi-IntruNet* adeptly recognize the unique features of different targets, extracting only those features tied to target movement. Even without additional training or parameter adjustments for the new setting, the system adeptly identifies different targets and discerns intrusions accurately.

Table 5.4: Evaluation of *Wi-IntruNet* in unseen environments with seen, unseen and coexisting subjects

Subject Type	**Intrusion detection rate**	**False alarm rate**
Seen	97.92%	0.00%
Unseen	98.81%	8.53%
Two	100%	0.00%
Average	98.91%	2.84%

5.4.4.2 Intrusion Detection Performance in Unseen Environment with Unseen Subjects

Subsequently, we probe the model's detection performance for unknown target intrusions in unseen environments, as well as the system's false alarm rate concerning unknown pets, as shown in Table 5.4. The system showcases a high detection rate of 98.81% and a modest false alarm rate of 8.53%. However, compared to known targets, there's a slight uptick in the false

alarm rate. This can be attributed to the presence of unusually large unseen dogs, including one weighing 85 lb, which the model hadn't encountered during its training and validation phases. In addition, the movement and behavior of a pet are different from other pets that the model has seen, which may trigger false alarms in a minimal number of instances.

5.4.4.3 Intrusion Detection Performance in Unseen Environment with Multiple Subjects

Following this, we evaluate the system's performance when faced with simultaneous intrusions by multiple individuals, as shown in Table 5.4. The results reveal that the system retains its ability to accurately detect human movements and maintain a high detection rate, even with multiple concurrent intruders. This proficiency stems from the network's acute discernment of the nuanced differences between multi-person intrusions and movements of non-human subjects, like pets or robotic vacuums. Notably, the system can pinpoint human-specific movement signatures, such as distinct gaits and speed patterns, even when several individuals move concurrently. This is a capability currently beyond the reach of most WiFi-based human sensing systems, particularly in unseen environments.

5.5 Discussion

5.5.1 Effectiveness of Intrusion Detection Module

An ablation study is conducted to assess the performance of the IDM. Using the dataset described in Section 5.4.2, we evaluate IDM's impact on both the Intrusion Detection Rate and

False Alarm Rate, as shown in Fig. 5.5. The results indicate that IDM significantly reduces the False Alarm Rate. This enhancement is credited to IDM's capability to leverage historical environmental data to rectify the MRM's occasional misinterpretations, commonly observed during complex motions such as the rapid movement of larger pets.

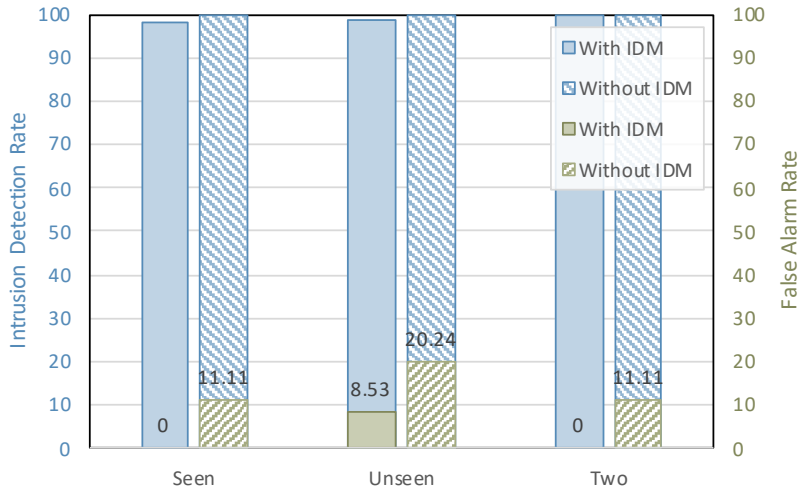


Figure 5.5: Effectiveness evaluation of IDM.

5.5.2 Latency

We analyze the computational efficiency of *Wi-IntruNet* by measuring the training time required by data preprocessing, MRM and IDM modules on a desktop computer, outfitted with an Intel Core i7 processor, an NVIDIA GTX 2080 GPU, and 16GB of RAM. For a segment of motion data spanning 5 seconds, the preprocessing phase consumes about 2.51 seconds. The training durations for the MRM and IDM modules are 160.39 seconds and 44.56 seconds, respectively. Given that *Wi-IntruNet* operates independently of specific environments, its training can be conducted offline, rendering the training time practically negligible.

5.5.3 Computational Complexity and Memory Requirements

We delve deeper into the computational complexity and memory demands of *Wi-IntruNet* by gauging the CPU time and peak memory usage during inference. Floating point operations per second (FLOPS), number of parameters, as well as model size are also measured for both MRM and IDM modeules. These findings are tabulated in Table 5.5. The MRM module necessitates 16.95 ms to categorize a 5-second motion segment, whereas the LSTM-based IDM module requires just 0.76 ms to integrate historical data for intrusion detection. The minimal CPU computation time, paired with the modest peak memory consumption of the MRM and IDM modules, underscores the feasibility of scaling *Wi-IntruNet* for widespread deployment on prevalent edge devices.

Table 5.5: Computational complexity and memory requirements

Module	**MRM**	**IDM**
FLOPS(G)	1.37	$1.85e^{-4}$
Parameters(M)	11.17	$1.82e^{-2}$
CPU inference time(ms)	16.95	0.76
Peak memory usage (MB)	0.65	0.06
Model size (MB)	42.7	0.06

5.6 Summary

In this chapter, we present *Wi-IntruNet*, the first robust through-wall indoor intrusion detection system that mitigates the interference from non-human indoor objects using commercial WiFi. *Wi-IntruNet* propose a novel deep learning framework that integrates a ResNet-18-based feature extractor for distinguishing human from non-human motions and

an LSTM-driven detector for historical data assimilation. With A-ACF input, *Wi-IntruNet* is versatile across various environments and orientations. Comprehensive evaluations validate its superior performance in real-world scenarios and readiness for swift and widespread deployment.

Chapter 6: Conclusion and Future Work

6.1 Conclusion

In this dissertation, we begin with an introductory overview of WiFi sensing, detailing concepts like WiFi-based distance estimation and the extraction of environment-invariant statistics from CSI for motion analysis. Subsequently, we present four WiFi-based solutions using motion analytics for tasks related to environmental mapping and monitoring.

1) In chapter 2, we present *EZMap*, a universal automatic floor plan construction system that does not need any prerequisite knowledge of buildings in advance. *EZMap* benefits from recent advances in centimeter-accuracy indoor tracking using RF signals. It leverages commodity WiFi to estimate accurate moving distances and employs inertial sensors for orientation reckoning. *EZMap* then processes crowdsourced trajectories with a novel pipeline of trajectory segmentation, matching, bundling, and shaping, which ultimately reconstructs a floor plan with not only skeletal layouts but also detailed area sizes of straight/curved corridors, open spaces, and rooms. Requiring minimal infrastructure and a small amount of data, *EZMap* can scale to a number of various buildings including public malls, offices, as well as home environments. To our best knowledge, *EZMap* is the first RF-based system that can accurately reconstruct map of private environments such as

home.

2) In chapter 3, we introduce an unprecedented system, *Wi-MoID* that identifies the movements of various human and non-human subjects utilizing readily available WiFi devices. The system identifies moving subjects by analyzing both physically interpretable and statistical attributes of the movement. A comprehensive assessment in diverse indoor scenarios validates the resilience of *Wi-MoID* in the real-world against various non-human subjects and changing conditions. *Wi-MoID* has been implemented on edge devices and is ready to be deployed ubiquitously for extensive use.

3) In chapter 4, we present a pioneering deep learning framework for robustly distinguishing between human and non-human subjects through walls using single-link WiFi. Unlike many prevailing intelligent systems that face challenges with interferences from non-human movements, our method stands out by effectively identifying a wide range of non-human subjects. We leverage a deep neural network to extract features from a designed statistic, A-ACF, which ensures robust performance regardless of environment, location, or direction. Rigorous experiments conducted in diverse settings with an array of subjects scrutinize the performance of prominent deep learning models and the effectiveness of transfer learning. Our evaluation results not only affirm the system's capability to discern human and non-human subjects with high accuracy in challenging scenarios but also offer insights into selecting appropriate deep learning models and utilizing transfer learning for WiFi sensing tasks.

4) In chapter 5, we presented *Wi-IntruNet*, the first robust through-wall indoor intrusion detection system that mitigates the interference from non-human indoor objects using

commercial WiFi. *Wi-IntruNet* propose a novel deep learning framework that includes a ResNet-18-based feature extractor for distinguishing human from non-human motions and an LSTM-driven detector for historical data assimilation. With A-ACF input, *Wi-IntruNet* is versatile across various environments and orientations. Comprehensive evaluations validate its superior performance in real-world scenarios and readiness for swift and widespread deployment.

6.2 Future Work

In this dissertation, we introduce a suite of systems that utilize WiFi-based motion analysis for the purposes of environmental monitoring and map construction. Utilizing the foundations provided by these technologies, we seek to advance related applications, thereby significantly augmenting their environmental sensing capabilities and ultimately elevating their level of computational intelligence.

- Drawing on our work of automatic map reconstruction, we can further delve deeper into the realization of a WiFi-based simultaneous tracking and mapping system. By converging the objectives of tracking and map construction, we can leverage the auto-generated map to refine tracking trajectories, leading to a notable enhancement in tracking precision.

- With the continual advancement of deep learning, an increasing number of models and learning methods are emerging, capable of solving more complex problems. Given these new developments, we can explore learning techniques, network architectures, and parameters settings for feature extraction from WiFi signals. For instance, we could delve further into the capacity of self-supervised learning and unsupervised learning for WiFi

signal feature extraction. By leveraging a wealth of unlabeled WiFi data, it opens the possibility of learning a more potent feature extractor.

- Additionally, the architecture we proposed for human and non-human subjects differentiation can be suitably adapted to address other intricate WiFi sensing tasks, thus providing solutions to challenges currently encountered in associated systems. This includes WiFi-based activities recognition in multi-person scenarios, human identification via WiFi unhindered by environmental constraints or directional limitations, as well as indoor positioning and proximity verification involving multiple individuals.

Bibliography

[1] Robert Sim, Pantelis Elinas, Matt Griffin, and James J Little. Vision-based slam using the rao-blackwellised particle filter. In *IJCAI Workshop on Reasoning with Uncertainty in Robotics*, volume 14, pages 9–16, 2005.

[2] WooYeon Jeong and Kyoung Mu Lee. CV-SLAM: A new ceiling vision-based slam technique. In *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3195–3200, 2005.

[3] Elena López, Rafael Barea, Alejandro Gómez, Álvaro Saltos, Luis M Bergasa, Eduardo J Molinos, and Abdelkrim Nemra. Indoor slam for micro aerial vehicles using visual and laser sensor fusion. In *Robot 2015: Second Iberian Robotics Conference*, pages 531–542, 2016.

[4] Koray Celik, Soon-Jo Chung, Matthew Clausman, and Arun K Somani. Monocular vision slam for indoor aerial vehicles. In *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1566–1573, October 2009.

[5] Jaehoon Jung, Sanghyun Yoon, Sungha Ju, and Joon Heo. Development of kinematic 3d laser scanning system for indoor mapping and as-built bim using constrained slam. *Sensors*, 15(10):26430–26456, 2015.

[6] Yuan He, Jiaqi Liang, and Yunhao Liu. Pervasive floorplan generation based on only inertial sensing: Feasibility, design, and implementation. *IEEE Journal on Selected Areas in Communications*, 35(5):1132–1140, 2017.

[7] Moustafa Alzantot and Moustafa Youssef. CrowdInside: Automatic construction of indoor floorplans. In *Proceedings of the 20th International Conference on Advances in Geographic Information Systems*, pages 99–108, 2012.

[8] Ruipeng Gao, Guojie Luo, and Fan Ye. VeMap: Indoor road map construction via smartphone-based vehicle tracking. In *2016 IEEE Global Communications Conference (GLOBECOM)*, pages 1–6, 2016.

[9] Chen Qiu and Matt W Mutka. iFrame: Dynamic indoor map construction through automatic mobile sensing. *Pervasive and Mobile Computing*, 38:346–362, 2017.

[10] Damian Philipp, Patrick Baier, Christoph Dibak, Frank Dürr, Kurt Rothermel, Susanne Becker, Michael Peter, and Dieter Fritsch. MapGENIE: Grammar-enhanced indoor map construction from crowd-sourced data. In *2014 IEEE International Conference on Pervasive Computing and Communications (PerCom)*, pages 139–147, 2014.

[11] Ricardo Santos, Marília Barandas, Ricardo Leonardo, and Hugo Gamboa. Fingerprints and floor plans construction for indoor localisation based on crowdsourcing. *Sensors*, 19(4):919, 2019.

[12] Akin Ayanoglu, Daniel M Schneider, and Ben Eitel. Crowdsourcing-based magnetic map generation for indoor localization. In *2018 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pages 1–8, 2018.

[13] Michael Hardegger, Sinziana Mazilu, Dario Caraci, Frederik Hess, Daniel Roggen, and Gerhard Tröster. ActionSLAM on a smartphone: At-home tracking with a fully wearable system. In *International Conference on Indoor Positioning and Indoor Navigation*, pages 1–8, 2013.

[14] Heba Abdelnasser, Reham Mohamed, Ahmed Elgohary, Moustafa Farid Alzantot, He Wang, Souvik Sen, Romit Roy Choudhury, and Moustafa Youssef. SemanticSLAM: Using environment landmarks for unsupervised indoor localization. *IEEE Transactions on Mobile Computing*, 15(7):1770–1782, 2015.

[15] Sebastian Thrun. Learning metric-topological maps for indoor mobile robot navigation. *Artificial Intelligence*, 99(1):21–71, 1998.

[16] Guobin Shen, Zhuo Chen, Peichao Zhang, Thomas Moscibroda, and Yongguang Zhang. Walkie-markie: Indoor pathway mapping made easy. In *10th Symposium on Networked Systems Design and Implementation*, pages 85–98, 2013.

[17] Divya Vavili, Dilip Gudlur, Pallav Vyas, Faisal Luqman, and Pei Zhang. SMILAS: Sensor based map generation for indoor location aware systems. In *Proceedings of the Fourth ACM SIGSPATIAL International Workshop on Indoor Spatial Awareness*, pages 33–36, 2012.

[18] Hyojeong Shin, Yohan Chon, and Hojung Cha. Unsupervised construction of an indoor floor plan using a smartphone. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 42(6):889–898, 2011.

[19] Guozhen Zhu, Chenshu Wu, Beibei Wang, and K. J. Ray Liu. Floor plan reconstruction with high-precision rf-based tracking. In *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5073–5077, 2022.

[20] Abdallah Naser, Ahmad Lotfi, Junpei Zhong, and Jun He. Human activity of daily living recognition in presence of an animal pet using thermal sensor array. In *Proceedings of the 13th ACM International Conference on PErvasive Technologies Related to Assistive Environments*, pages 1–6, 2020.

[21] Neilly H. Tan, Richmond Y. Wong, Audrey Desjardins, Sean A. Munson, and James Pierce. Monitoring pets, deterring intruders, and casually spying on neighbors: Everyday uses of smart home cameras. In *CHI Conference on Human Factors in Computing Systems*, pages 1–25, 2022.

[22] Jian Gong, Xinyu Zhang, Ju Ren, and Yaoxue Zhang. The invisible shadow: How security cameras leak private activities. In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, pages 2780–2793, 2021.

[23] Emanuele Cardillo, Changzhi Li, and Alina Caddemi. Vital sign detection and radar self-motion cancellation through clutter identification. *IEEE Transactions on Microwave Theory and Techniques*, 69(3):1932–1942, 2021.

[24] Pengfei Wang, Yang Zhang, Yangyang Ma, Fulai Liang, Qiang An, Huijun Xue, Xiao Yu, Hao Lv, and Jianqi Wang. Method for distinguishing humans and animals in vital signs monitoring using IR-UWB radar. *International Journal of Environmental Research and Public Health*, 16(22), 2019.

[25] Abhijit Bhattacharya and Rodney Vaughan. Deep learning radar design for breathing and fall detection. *IEEE Sensors Journal*, 20(9):5072–5085, 2020.

[26] Mohamad Forouzanfar, Mohamed Mabrouk, Sreeraman Rajan, Miodrag Bolic, Hilmi R. Dajani, and Voicu Z. Groza. Event recognition for contactless activity monitoring using phase-modulated continuous wave radar. *IEEE Transactions on Biomedical Engineering*, 64(2):479–491, 2017.

[27] Chenshu Wu, Beibei Wang, Oscar C. Au, and K. J. Ray Liu. Wi-Fi can do more: Toward ubiquitous wireless sensing. *IEEE Communications Standards Magazine*, 6(2):42–49, 2022.

[28] K. J. Ray Liu and Beibei Wang. *Wireless AI: Wireless Sensing, Positioning, IoT, and Communications*. Cambridge University Press, 2019.

[29] Feng Zhang, Chenshu Wu, Beibei Wang, Hung-Quoc Lai, Yi Han, and K. J. Ray Liu. WiDetect: Robust motion detection with a statistical electromagnetic model. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 3(3):1–24, 2019.

[30] Dan Wu, Daqing Zhang, Chenren Xu, Hao Wang, and Xiang Li. Device-free WiFi human sensing: From pattern-based to model-based approaches. *IEEE Communications Magazine*, 55(10):91–97, 2017.

[31] Fangxin Wang, Wei Gong, and Jiangchuan Liu. On spatial diversity in WiFi-based human activity recognition: A deep learning-based approach. *IEEE Internet of Things Journal*, 6(2):2035–2047, 2019.

[32] Jie Wang, Yunong Zhao, Xiaorui Ma, Qinghua Gao, Miao Pan, and Hongyu Wang. Cross-scenario device-free activity recognition based on deep adversarial networks. *IEEE Transactions on Vehicular Technology*, 69(5):5416–5425, 2020.

[33] Fangxin Wang, Wei Gong, Jiangchuan Liu, and Kui Wu. Channel selective activity recognition with WiFi: A deep learning approach exploring wideband information. *IEEE Transactions on Network Science and Engineering*, 7(1):181–192, 2020.

[34] Yongsen Ma, Sheheryar Arshad, Swetha Muniraju, Eric Torkildson, Enrico Rantala, Klaus Doppler, and Gang Zhou. Location- and person-independent activity recognition with WiFi, deep neural networks, and reinforcement learning. *ACM Trans. Internet of Things*, 2(1), Jan. 2021.

[35] Dazhuo Wang, Jianfei Yang, Wei Cui, Lihua Xie, and Sumei Sun. Multimodal csi-based human activity recognition using gans. *IEEE Internet of Things Journal*, 8(24):17345–17355, 2021.

[36] Chenshu Wu, Feng Zhang, Yuqian Hu, and K. J. Ray Liu. GaitWay: Monitoring and recognizing gait speed through the walls. *IEEE Transactions on Mobile Computing*, 20(6):2186–2199, 2021.

[37] Yuanying Chen, Wei Dong, Yi Gao, Xue Liu, and Tao Gu. Rapid: A multimodal and device-free approach using noise estimation for robust person identification. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 1(3), Sep. 2017.

[38] Yangjie Cao, Zhiyi Zhou, Chenxi Zhu, Pengsong Duan, Xianfu Chen, and Jie Li. A lightweight deep learning algorithm for WiFi-based identity recognition. *IEEE Internet of Things Journal*, 8(24):17449–17459, 2021.

[39] Yang Xu, Wei Yang, Min Chen, Sheng Chen, and Liusheng Huang. Attention-based gait recognition and walking direction estimation in WiFi networks. *IEEE Transactions on Mobile Computing*, 21(2):465–479, 2022.

[40] Chenshu Wu, Feng Zhang, Beibei Wang, and K. J. Ray Liu. EasiTrack: Decimeter-level indoor tracking with graph-based particle filtering. *IEEE Internet of Things Journal*, 7(3):2397–2411, 2019.

[41] Yue Zheng, Yi Zhang, Kun Qian, Guidong Zhang, Yunhao Liu, Chenshu Wu, and Zheng Yang. Zero-effort cross-domain gesture recognition with wi-fi. In *Proceedings of the 17th Annual International Conference on Mobile Systems, Applications, and Services*, pages 313–325, 2019.

[42] Guozhen Zhu, Chenshu Wu, Beibei Wang, and K. J. Ray Liu. Floor plan reconstruction with high-precision RF-based tracking. In *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5073–5077, 2022.

[43] Guozhen Zhu, Chenshu Wu, Beibei Wang, and K. J. Ray Liu. EZMap: Boosting automatic floor plan construction with high-precision robotic tracking. *IEEE Internet of Things Journal*, 10(8):6988–6998, 2023.

[44] Yuxiang Lin, Yi Gao, Bingji Li, and Wei Dong. Revisiting indoor intrusion detection with WiFi signals: Do not panic over a pet! *IEEE Internet of Things Journal*, 7(10):10437–10449, 2020.

[45] Elahe Soltanaghaei, Rahul Anand Sharma, Zehao Wang, Adarsh Chittilappilly, Anh Luong, Eric Giler, Katie Hall, Steve Elias, and Anthony Rowe. Robust and practical WiFi human sensing using on-device learning with a domain adaptive model. In *Proceedings of the 7th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation*, BuildSys '20, pages 150–159, 2020.

[46] Leila Takayama, Caroline Pantofaru, David Robson, Bianca Soto, and Michael Barry. Making technology homey: Finding sources of satisfaction and meaning in home automation. page 511–520, 2012.

[47] Wei Wang, Alex X. Liu, and Ke Sun. Device-free gesture tracking using acoustic signals. In *Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking*, pages 82–94, 2016.

[48] Wenguang Mao, Jian He, and Lili Qiu. CAT: High-precision acoustic motion tracking. In *Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking*, pages 69–81, 2016.

[49] Leibo Liu, Weilong Zhang, Chenchen Deng, Shouyi Yin, and Shaojun Wei. BriGuard: A lightweight indoor intrusion detection system based on infrared light spot displacement. *IET Science, Measurement & Technology*, 9(3):306–314, 2015.

[50] Ju Han and B. Bhanu. Human activity recognition in thermal infrared imagery. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Workshops*, pages 17–17, 2005.

[51] Beibei Wang, Qinyi Xu, Chen Chen, Feng Zhang, and K.J. Ray Liu. The promise of radio analytics: A future paradigm of wireless positioning, tracking, and sensing. *IEEE Signal Processing Magazine*, 35(3):59–80, 2018.

[52] Zhenghua Chen, Le Zhang, Chaoyang Jiang, Zhiguang Cao, and Wei Cui. WiFi CSI based passive human activity recognition using attention based BLSTM. *IEEE Transactions on Mobile Computing*, 18(11):2714–2724, 2019.

[53] Zimu Zhou, Zheng Yang, Chenshu Wu, Longfei Shangguan, and Yunhao Liu. Towards omnidirectional passive human detection. In *2013 Proceedings IEEE INFOCOM*, pages 3057–3065, 2013.

[54] Tong Xin, Bin Guo, Zhu Wang, Mingyang Li, Zhiwen Yu, and Xingshe Zhou. FreeSense: Indoor human identification with WiFi signals. In *2016 IEEE Global Communications Conference (GLOBECOM)*, pages 1–7, 2016.

[55] Chenshu Wu, Zheng Yang, Zimu Zhou, Xuefeng Liu, Yunhao Liu, and Jiannong Cao. Non-invasive detection of moving and stationary human with WiFi. *IEEE Journal on Selected Areas in Communications*, 33(11):2329–2342, 2015.

[56] Guozhen Zhu, Chenshu Wu, Xiaolu Zeng, Beibei Wang, and K. J. Ray Liu. Who moved my cheese? human and non-human motion recognition with WiFi. In *2022 IEEE 19th*

International Conference on Mobile Ad Hoc and Smart Systems (MASS), pages 476–484, 2022.

[57] Feng Zhang, Chen Chen, Beibei Wang, and K. J. Ray Liu. WiSpeed: A statistical electromagnetic approach for device-free indoor speed estimation. *IEEE Internet of Things Journal*, 5(3):2163–2177, 2018.

[58] Axel Küpper. *Location-based services: Fundamentals and operation*. John Wiley & Sons, 2005.

[59] Chen Chen, Yan Chen, Yi Han, Hung-Quoc Lai, and K. J. Ray Liu. Achieving centimeter-accuracy indoor localization on WiFi platforms: A frequency hopping approach. *IEEE Internet of Things Journal*, 4(1):111–121, 2017.

[60] Chen Chen, Yan Chen, Yi Han, Hung-Quoc Lai, Feng Zhang, and K. J. Ray Liu. Achieving centimeter-accuracy indoor localization on WiFi platforms: A multi-antenna approach. *IEEE Internet of Things Journal*, 4(1):122–134, 2017.

[61] Chen Chen, Yi Han, Yan Chen, and K. J. Ray Liu. Indoor global positioning system with centimeter accuracy using Wi-Fi. *IEEE Signal Processing Magazine*, 33(6):128–134, 2016.

[62] Zhung-Han Wu, Yi Han, Yan Chen, and K. J. Ray Liu. A time-reversal paradigm for indoor positioning system. *IEEE Transactions on Vehicular Technology*, 64(4):1331–1339, 2015.

[63] Chenshu Wu, Feng Zhang, Yusen Fan, and K. J. Ray Liu. Rf-based inertial measurement. In *Proceedings of the ACM Special Interest Group on Data Communication*, pages 117–129. 2019.

[64] K. J. Ray Liu and Beibei Wang. *Wireless AI: Wireless Sensing, Positioning, IoT, and Communications*. Cambridge University Press, 2019.

[65] Beibei Wang, Qinyi Xu, Chen Chen, Feng Zhang, and K. J. Ray Liu. The promise of radio analytics: A future paradigm of wireless positioning, tracking, and sensing. *IEEE Signal Processing Magazine*, 35(3):59–80, 2018.

[66] Donald J Berndt and James Clifford. Using dynamic time warping to find patterns in time series. In *KDD workshop*, volume 10, pages 359–370, 1994.

[67] GA Watson. Computing helmert transformations. *Journal of computational and applied mathematics*, 197(2):387–394, 2006.

[68] H. Edelsbrunner, D. Kirkpatrick, and R. Seidel. On the shape of a set of points in the plane. *IEEE Transactions on Information Theory*, 29(4):551–559, 1983.

[69] Feng Zhang, Chen Chen, Beibei Wang, Hung-Quoc Lai, Yi Han, and K. J. Ray Liu. WiBall: A time-reversal focusing ball method for decimeter-accuracy indoor tracking. *IEEE Internet of Things Journal*, 5(5):4031–4041, October 2018.

[70] 2023-2024 APPA national pet owners survey.

[71] Siamak Yousefi, Hirokazu Narui, Sankalp Dayal, Stefano Ermon, and Shahrokh Valaee. A survey on behavior recognition using WiFi channel state information. 55(10):98–104, 2017.

[72] Chunjing Xiao, Daojun Han, Yongsen Ma, and Zhiguang Qin. CsiGAN: Robust channel state information-based activity recognition with GANs. *IEEE Internet of Things Journal*, 6(6):10191–10204, 2019.

[73] Han Zou, Yuxun Zhou, Jianfei Yang, Hao Jiang, Lihua Xie, and Costas J. Spanos. DeepSense: Device-free human activity recognition via autoencoder long-term recurrent convolutional network. In *2018 IEEE International Conference on Communications (ICC)*, pages 1–6, 2018.

[74] Biyun Sheng, Fu Xiao, Letian Sha, and Lijuan Sun. Deep spatial–temporal model based cross-scene action recognition using commodity WiFi. 7(4):3592–3601.

[75] Jianfei Yang, Xinyan Chen, Han Zou, Dazhuo Wang, Qianwen Xu, and Lihua Xie. EfficientFi: Towards large-scale lightweight WiFi sensing via CSI compression. *IEEE Internet of Things Journal*, 9(15):13086–13095, 2022.

[76] Yuqian Hu, Feng Zhang, Chenshu Wu, Beibei Wang, and K. J. Ray Liu. DeFall: Environment-independent passive fall detection using WiFi. *IEEE Internet of Things Journal*, 9(11):8515–8530, 2022.

[77] Feng Zhang, Chenshu Wu, Beibei Wang, Min Wu, Daniel Bugos, Hangfang Zhang, and K. J. Ray Liu. SMARS: Sleep monitoring via ambient radio signals. *IEEE Transactions on Mobile Computing*, 20(1):217–231, 2021.

[78] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6000–6010, 2017.

[79] Simon Haykin. *Neural networks: A comprehensive foundation*. Prentice Hall PTR, 1994.

[80] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[81] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[82] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[83] Junyoung Chung, Caglar Gulcehre, Kyunghyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. In *NIPS 2014 Workshop on Deep Learning*, 2014.

[84] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021.

[85] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255, 2009.

[86] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. 2019.

[87] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2015.

[88] Ian J. Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, Cambridge, MA, USA, 2016.

[89] Xingjian Shi, Zhourong Chen, Hao Wang, Dit-Yan Yeung, Wai-Kin Wong, and Wang-chun Woo. Convolutional lstm network: A machine learning approach for precipitation nowcasting. *Advances in neural information processing systems*, 28, 2015.