Efficient Deep Learning Framework Using Convolutional Recurrent Neural Network for Elderly Care Centers for Fall Detection

Aamer Khan

A thesis submitted in fulfilment of the degree of Master of Philosophy

Auckland University of Technology

August 2024

Abstract

Each year, about a third of people over 65 years old experience falls, with one out of five sustaining significant injuries such as head trauma or fractures. Even those who do not suffer injuries often struggle to get up unassisted, leading to a fear of falling, loss of confidence, reduced physical activity, poor social interaction, and depression. Monitoring the movements of older adults is crucial to address these issues without compromising their privacy or hindering daily activities.

Device-free sensing has emerged as a reliable method for monitoring presence, location, motion, activity, and gestures without requiring attached devices. This technique leverages wireless signals such as Wi-Fi and 4G/5G to detect movements, offering a dependable alternative in conditions where other technologies like vision-based cameras might fail, such as in smoke or darkness. Additionally, device-free sensing respects privacy, making it ideal for pervasive sensing applications, particularly fall detection.

This research aims to develop a prototype for elderly care that is effective and affordable. The objective is to create a system that does not require frequent recharging, can promptly alert caregivers about falls, and is within the financial reach of most elderly care centers. The study explores the effectiveness of different machine learning models—Convolutional Neural Networks (CNN), Long Short-Term Memory (LSTM) networks, and a hybrid Convolutional LSTM (Conv-LSTM) model—in detecting activities using Channel State Information (CSI) data, both with and without automatic data labeling.

Experiments conducted without data labeling revealed that the CNN model, trained on pre-processed CSI data, achieved a validation accuracy of 85.33% and a testing accuracy of 84.52%, though it displayed instability during training. The LSTM model performed slightly better, with a validation accuracy of 87.12% and a testing accuracy of 86.87%, benefiting from its suitability for time-series data. However, the Conv-LSTM model, which combines CNN and LSTM layers, demonstrated superior stability and performance, with a validation accuracy of 89.34% and a testing accuracy of 90.16%, effectively capturing spatial and temporal information from the data. This reliability of the Conv-LSTM model provides a strong foundation for the research findings.

When automatic data labeling was applied, all models showed significant performance improvements. The CNN model achieved a validation accuracy of approximately 93% and a

testing accuracy of around 92%. The LSTM model reached a validation and testing accuracy of approximately 94%, demonstrating robust training behavior. The Conv-LSTM model outperformed both, achieving exceptional results with a validation accuracy of around 97% and a testing accuracy of approximately 98%.

These findings underscore the significant impact of data preprocessing and labeling on the performance of machine learning models in activity detection using CSI data. The proposed automatic data labeling method proved particularly effective, removing extraneous frames and applying heuristic-based labels. This method enhances the models' ability to interpret data and significantly reduces the complexity and noise within the dataset, leading to more accurate and reliable activity detection.

In conclusion, developing and applying effective data labelling and preprocessing techniques are crucial for improving the accuracy and reliability of device-free sensing systems for elderly care. This research demonstrates that with proper data handling, machine learning models can significantly enhance fall detection and other monitoring capabilities, providing a valuable tool for ensuring the safety and well-being of older adults.

Table of Contents

Page No

List of Figures

List of Tables

Page No

List of Abbreviations

CNN – Convolutional Neural Network

Conv-LSTM – Convolutional Long Short-Term Memory

CSI – Channel State Information

DL – Deep Learning

DNN – Deep Neural Network

IoT – Internet of Things

LSTM – Long Short-Term Memory

ML – Machine Learning

RNN – Recurrent Neural Network

Attestation of Authorship

I hereby declare that the submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person nor material which, to a substantial extent, has been accepted for qualification of any other degree or diploma of a university or other institution of higher learning, except where due acknowledgement is made in the acknowledgements.

Aamer Khan

30th August 2024

Acknowledgements

I wish to express my gratitude to my supervisor, Dr. Jeff Kilby, for his patience, support, guidance, and advice throughout this research.

I want to thank my parents for their support, patience, and love in completing this work.

Sincere thanks are also extended to helpful friends, participants, and all staff of the School of Engineering for their kindness and encouragement during my studies at Auckland University of Technology.

Chapter 1
Introduction

The Internet of Things (IoT) is a phenomenon that has attained immense popularity among entrepreneurs, researchers, and tech-savvy people worldwide (Kummitha & Crutzen, 2019). Also, with rapid growth in the functionality of computing and communication technologies and how IoT devices operate with low-power, low-cost sensors and electronic components, many newer opportunities for IoT applications have been on the rise. One such example includes the geo-technology concept (Agjei & Orangeh, 2023), which is associated with e-health and assisted living technology for the aged population that may improve the quality of healthcare systems.

Statistically, injuries due to falls are one of the significant public health problems globally. A study suggests that 646,000 persons throughout the world ("Falls, WHO - Fact sheet," 2021) face fatality. while falling, which makes it the second leading cause of unintentional injury or death (Mall et al., 2023). The trend using Google shows that fall detection has gained immense popularity among academia and industry, especially in the last few years.

According to the World Health Organization (WHO) findings, adults older than 65 are more prone to falling, the repercussions of which are huge. Death rates due to falling are highest among adults who are over 60 years of age ("Falls, WHO - Fact sheet," 2021). Detecting a fall enables immediate assistance from a caregiver and minimizes the fatality. One issue is that no such data is available, as falling is a random accidental act that happens unintentionally or involuntarily. Gathering information and data relating to falls and their fatality is daunting for researchers who want to build their dataset to study and investigate.

There have been many approaches and techniques, including wearable sensor-based and computer vision-based approaches (W. Chen et al., 2020; Dang et al., 2020). Wearable sensors are the first type of fall detection method, consisting of an accelerometer, gyroscope, barometer, and global positioning system (W. Chen et al., 2020). This approach was gradually improved, especially in providing a better device and battery size. However, older people still need to be equipped with it all the time to detect falls. Still, carelessness and forgetfulness can stop older people from wearing such a sensor, which is one major limitation of this approach.

1.1 Background

Understanding the evolution and significance of fall detection systems is crucial for improving patient outcomes in healthcare. This section delves into the historical development and the impact of accurate fall detection systems.

1.1.1 Evolution of fall detection systems

Fall detection technology has significantly progressed, from early reliance on wearable sensors for detecting falls through threshold values of peak acceleration or angle of inclination to sophisticated methods that reduce false alarms and enhance accuracy. (Alazeb et al., 2023) noted the limitations of early systems while (Singh et al., 2020) improved upon these with a two-stage algorithm, achieving 98% accuracy in distinguishing falls from daily activities using a combination of inclination angle thresholding and supervised machine learning on data from 12 participants. For seamless integration into everyday life and clinical settings, ideal fall detection systems must be unobtrusive, cost-effective, highly accurate, and capable of real-time data processing with minimal false alarms. Traditional methods often fall short in one or more aspects, but recent advancements leverage device-free technologies, such as ubiquitous wireless sensors, to meet these demands. (Nahian et al., 2021) highlight the potential of Wi-Fi and CSI-based approaches, offering fine granularity, cost-effectiveness, and scalability through hybrid machine learning and deep learning algorithms.

In addition to wearable sensors, rule-based algorithms on multisensory data were explored, with (Bourke et al., 2007) achieving 97.5% accuracy using accelerometers and gyroscopes. However, the requirement for individual threshold tuning and reliance on empirical formulas limit their applicability in real-world scenarios. (Palipana et al., 2018) demonstrated the effectiveness of recurrent models like LSTMs in capturing temporal patterns in time-series data for fall detection, with accuracies up to 96%, underscoring deep learning's ability to learn from raw data without explicit feature engineering. The advent of smartphone-based fall detection leverages built-in accelerometers and cloud-based machine learning for efficient and accessible solutions. (Wang et al., 2020) reported a 96% accuracy using machine learning classifiers on data from elderly participants.

Channel State Information (CSI) has emerged as a valuable source for device-free sensing, with early CSI-based studies focusing on feature engineering and machine learning for fall detection. Initial efforts like WiFall and research by (Y. Wang et al., 2016) showed accuracies of up to 94.1% using handcrafted features and traditional classifiers. Despite these advancements, challenges remain, including the lack of contextual awareness and the need for robustness across diverse populations and environments. Often overlooked, transactional activities that precede falls require attention for comprehensive fall detection solutions.

Recent research has broadened fall detection's scope, incorporating models like CNN-LSTM for classifying falls and localizing fall locations, with WiDance achieving an 88.7% accuracy in location-aware fall detection (Palipana et al., 2018). Ensemble methods and transfer learning have further enhanced accuracy and generalization capabilities, with notable successes in adapting models to different environments, as demonstrated by (Pan et al., 2008) with over 90% accuracy. This cohesive narrative illustrates the evolution from simple sensor-based systems to sophisticated, context-aware models that leverage the latest in machine learning and sensing technologies, highlighting the ongoing quest for unobtrusive, accurate, and reliable fall detection systems.

1.1.2 Significance of Accurate Fall Detection in Healthcare

Falls are a significant health concern, particularly among the elderly, with the World Health Organization ("Falls, WHO - Fact sheet," 2021) noting that one in four elderly individuals globally experiences a fall each year, leading to severe injuries such as fractures and head traumas. These incidents increase mortality risk and significantly diminish the quality of life by reducing mobility and independence. With the elderly population expected to double by 2050, the socioeconomic impact of fall-related injuries is set to increase. In the United States, an estimated one in three adults aged 65 and above falls each year, highlighting falls as a leading cause of both fatal and non-fatal injuries worldwide for this demographic ("Falls, WHO - Fact sheet," 2021; Ramachandran & Karuppiah, 2020). The delayed detection of falls further exacerbates injury severity, underscoring the importance of timely intervention. However, traditional methods reliant on self-reporting by the elderly suffer from significant detection latencies, often worsening outcomes.

Enhancing fall detection is crucial for prompt medical response, initiating preventive measures, and mitigating the psychological impact on elderly individuals living alone. Research indicates that even a 30-minute delay in fall detection can dramatically increase the risk of death (Karlsson et al., 2013), while early notification systems can activate safety calls to family or caregivers, aiding non-pharmaceutical fall prevention strategies (W. Chen et al., 2020). Accurate fall detection thus becomes fundamental in improving health outcomes, ensuring safety monitoring, and supporting the independence of vulnerable groups. Moreover, continuous, real-time monitoring of falls assists in preventive interventions and rehabilitation, allowing for a data-driven assessment of fall risks and the development of personalized treatment plans based on fall characteristics (Liu et al., 2019; Min et al., 2018). This approach facilitates insights into fall risk factors. It supports proactive geriatric care models that reduce fall frequency through targeted rehabilitation strategies, enabling detection of abnormal post-fall behaviors requiring urgent assistance.

The traditional clinical solutions for fall detection, often reliant on obtrusive wearables, face challenges with user compliance. Ambient fall detection through ubiquitous sensing technologies offers a less intrusive alternative, enabling consistent environmental monitoring in the homes of elderly individuals. Emerging technologies like Wi-Fi Channel State Information (CSI) have shown promise in unobtrusive sensing, leveraging existing wireless infrastructure for reliable fall detection with machine learning. This reflects a significant shift from wearables and vision-based systems to non-intrusive methods such as Wi-Fi-based fall detection, which benefits from the widespread deployment of Wi-Fi and the sensitivity of CSI to human motion (Mansoor et al., 2022). This evolution towards unobtrusive, ubiquitous sensing aligns with a broader healthcare trend aiming to improve access, enhance the quality of life, and promote wellness among the ageing population.

1.1.3 Overview of Channel State Information and its Advantages

Channel State Information (CSI) represents a critical aspect of modern Wi-Fi technology, particularly in orthogonal frequency division multiplexing (OFDM) systems utilized in 802.11n/ac Wi-Fi standards. CSI measures the channel frequency response at the receiver for each subcarrier, capturing both amplitude attenuation and the multipath effects—scattering, diffraction, and reflections—that signals undergo between the transmitter and receiver. This capability enables the extraction of rich spatial and temporal signatures from the propagation environment, offering a detailed view of wireless channel characteristics at the sub-carrier level, including combined effects of scattering, fading, and power decay (Halperin et al., 2011; He et al., 2015). Distinguished from received signal strength (RSS), which is measured at a higher, more abstracted layer, CSI operates at the physical layer, providing more precise and location-specific channel state information. Its ability to offer finer-grained localization potential, with accuracies below the 60 cm mark at a 5 GHz Wi-Fi frequency, underscores its superiority over RSS, especially in complex indoor settings where multipath richness is prevalent (Abuhoureyah et al., 2023). Furthermore, CSI has proven to be highly effective in indoor localization and human activity recognition, leveraging the fine-grained information it provides on channel amplitude and phase at the subcarrier level to detect subtle motion details, such as those caused by human movements affecting Wi-Fi signals (Guo et al., 2024).

An additional strength of CSI lies in its inherent immunity to minor environmental variations that do not significantly alter signal propagation characteristics. For instance, activities with minimal impact on Wi-Fi-CSI trajectories, such as opening and closing doors or the movement of non-metal fixtures, highlight the reliability of CSI-based human sensing systems over RSS-based approaches, which are more susceptible to noise (Zafari et al., 2019). Moreover, CSI measurements maintain their integrity independently of factors like transmit power level or Link Quality Indicator (LQI) values, which fluctuate significantly over time (Bibbò et al., 2022). Despite its advantages, the effective utilization of CSI data for applications like fall detection presents challenges, including sensitivity to environmental factors like object reflections, temperature variations, and humidity levels. These elements introduce variability and noise into CSI measurements, complicating accurately detecting human activities or movements. Additionally, different human body parts reflect Wi-Fi signals to varying degrees, which, coupled with the distortion of phase information by multipath reflections, can lead to inaccuracies and unpredictable fluctuations in CSI data (Janidarmian et al., 2017).

Recent technological advancements, such as channel bonding and wider bandwidth transmissions in line with the 802.11ax standard, have improved CSI's capabilities, allowing for the extraction of information across bandwidths up to 80 MHz channel bandwidths. These enhancements are pivotal for applications that detect subtle environmental perturbations, like device-free activity classification, and further solidify CSI's role in robust indoor localization and sensing applications (Wang et al., 2024). Coupled with its resistance to noise and attenuation and its ability to leverage frequency diversity, CSI's fine resolution facilitates the extraction of distinct features associated with delicate activities, such as falls, making it a reliable carrier for ambient activity sensing applications that demand nuanced motion recognition. (Abbate et al., 2012; Kraft et al., 2020).

1.2 Importance of Machine Learning and Deep Learning

Machine Learning (ML) and Deep Learning (DL) play pivotal roles in enhancing the capabilities of detection systems. Their integration can significantly improve the accuracy and efficiency of fall detection. The subsections below discuss the importance of ML and DL in enhancing fall detection systems.

1.2.1 Integration of Machine Learning and Deep Learning in Enhancing Detection Systems

Machine Learning (ML), particularly through Deep Learning (DL) techniques, has significantly advanced the capabilities in various domains, such as computer vision and natural language processing, by enabling automatic learning of task-specific representations directly from raw data. These advancements have similarly impacted CSI-based human sensing, where supervised learning algorithms have progressed beyond the early rule/threshold-based heuristic approaches traditionally employed in healthcare. Early machine learning applications in this sector utilized algorithms like Support Vector Machines (SVM), Decision Trees, and K-Nearest Neighbors (KNN), leveraging handcrafted features by domain experts. This approach saw applications such as Boosted Decision Trees enhancing fall risk prediction from accelerometry data (Wang et al., 2022). However, the advent of deep learning in the mid-2010s introduced the capability for automatic representation learning from raw data, offering a significant leap forward. Deep learning models, particularly Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs), including Long Short-Term Memory (LSTM) networks, have proven exceptionally suited for analysing time-series data like CSI measurements. CNNs, leveraging techniques adapted from computer vision, perform temporal analysis through 1D convolution operations, while LSTMs capture long-range dependencies within CSI data, facilitating effective activity classification (Mekruksavanich & Jitpattanakul, 2021). The integration of CNNs and RNNs into hybrid models, such as ConvLSTM, and the incorporation of attention mechanisms have further optimized the modeling of spatiotemporal dependencies and focused computational resources on critical signal segments for improved human activity recognition (HAR) (Şengül et al., 2022). Novel approaches to data representation, treating CSI's real and imaginary components as analogs to color channels in image processing, have also emerged, bypassing the need for manual feature engineering and enhancing recognition accuracy (Wang et al., 2020).

Advancements continue with convolutional LSTM and attention mechanisms enriching the models' ability to discern salient features in sequence data, thereby bolstering HAR tasks. Ensemble methods that combine CNNs, RNNs, and other classifiers through a two-stage training process have shown superior accuracy on CSI datasets compared to standalone models, marking a substantial improvement over initial methodologies that relied heavily on manually extracted features (Zafari et al., 2019). These deep learning approaches simplify feature extraction and present a robust framework for device-free fall detection using CSI, overcoming the challenges of high dimensionality and temporal variability inherent in CSI signals (Hussain et al., 2019). Recent research has explored transfer and federated learning to enhance CSI-based human sensing. Transfer learning, which involves initializing models with weights from pre-trained networks, helps mitigate overfitting risks in smaller datasets. On the other hand, Federated learning offers a distributed learning approach that ensures privacy while allowing for model improvements across multiple devices, a promising avenue for fall detection applications (Shiri et al., 2021). Despite these advancements, the unstable nature of CSI measurements remains a significant challenge, with standard preprocessing steps like filtering only partially addressing issues related to phase fluctuations, subcarrier limitations, and sampling inconsistencies. This underscores the necessity for robust noise mitigation techniques to ensure deep learning models can effectively learn reliable patterns from CSI data, emphasizing the critical role of preprocessing in enhancing data quality for successful fall detection applications (Dang et al., 2020).

1.3 Motivation

This research aims to address the growing need for advanced fall detection methodologies. This section highlights the driving factors and expected outcomes from integrating various technologies.

1.3.1 Advanced Fall Detection Methodologies

Despite the significant strides made in fall detection research, existing systems still confront substantial challenges, underscoring the necessity for further advancements in methodologies to cater to the healthcare needs of vulnerable populations effectively. Initial investigations into fall detection employed traditional machine learning techniques such as Support Vector Machines (SVMs), decision trees, and nearest neighbors to classify handcrafted Channel State Information (CSI)-based features. However, the complexity due to the high dimensionality and temporal dynamics inherent in CSI signals made manual feature extraction a daunting task (K. Wu et al., 2022). In response, the focus has shifted towards deep learning approaches, notably Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) like Long Short-Term Memory (LSTM) networks, which automate the feature learning process, efficiently extracting spatial and temporal features from sequential CSI data (Igual et al., 2013; Yadav & Gurjar, 2021). The global demographic shift towards an ageing population has amplified the socioeconomic impacts of falls and related injuries, highlighting the limitations of current clinical standards that often rely on obtrusive wearable devices, leading to suboptimal user compliance. The focus of previous studies on basic daily activities, rather than on the crucial transitional activities preceding falls, necessitates a shift towards unobtrusive ambient sensing methods for swift detection and response (Koshmak et al., 2016). The goal of continuous, real-time monitoring is not only to facilitate prompt emergency interventions but also to enable preventive measures through the analysis of fall incidents, thereby supporting proactive care models to reduce fall incidences and improve post-fall interventions.

However, the reliability of existing fall detection algorithms, primarily threshold or rule-based, is compromised in real-world settings due to user mobility, diverse living environments, and complex human behaviors. Although the fusion of heterogeneous data streams promises enhanced reliability, it introduces increased computational demands. Furthermore, the dependence on manual feature engineering in traditional machine learning methods limits generalizability, underscoring the need for more sophisticated sensing and learning strategies. Recent advancements demonstrate the potential of integrating machine learning and deep learning to enhance fall detection accuracy through ensemble models that combine algorithms such as SVM, CNN, and LSTM. These approaches leverage multi-task learning and late fusion techniques to capitalize on the strengths of both handcrafted and deep-learned features, capturing complex spatiotemporal patterns unattainable from singular data perspectives (Dang et al., 2020).

Nevertheless, the clinical adoption of fall detection technologies is hindered by the absence of substantial validation evidence, raising concerns about patient safety and the reliability of these systems under true ambulatory conditions. This gap highlights the critical need for benchmark datasets and evaluation methodologies to foster credible accreditation of assistive technologies within the healthcare sector (Karlsson et al., 2013). Thus, a significant opportunity exists to advance fall prevention strategies by leveraging the potential of ubiquitous wireless sensing and artificial intelligence. A multidisciplinary approach encompassing biomedical engineering, computer science, and artificial intelligence is essential for advancing fall detection technologies. The integration of wireless sensing can reveal crucial insights into human poses and motions. At the same time, the clinical validation of these technologies requires expertise in physiology to ensure precision across various settings (Salau et al., 2022). Moreover, AI innovations must navigate data privacy and interpretability challenges, ensuring that technological advancements respect human dignity and are guided by ethical and policy considerations. Ultimately, the success of assistive technologies will hinge on collaborative innovation, prioritizing community welfare and well-being over profit.

1.3.2 Role of Channel State Information in Fall Detection

Channel State Information (CSI) has emerged as a promising data modality for fall detection systems, offering unique advantages over conventional Received Signal Strength (RSS)-based approaches through the utilization of the ubiquitous wireless infrastructure present in most homes and consumer devices. Integrating CSI with complementary sensing sources, alongside leveraging recent advancements in machine learning (ML) and deep learning (DL), holds promise for achieving unobtrusive, precise, scalable, and practically viable fall detection systems. This approach addresses the limitations of individual sensing techniques and meets the performance criteria needed for widespread healthcare deployment (Nahian et al., 2021). The increased availability of multi-modal data from co-located wireless and inertial devices has fueled interest in fused ML-DL strategies, enhancing fall detection and activity recognition capabilities. CSI distinguishes itself by capturing finer-grained multipath signal propagation behaviors, enabling precision in indoor localization and resilience against shadowing effects, thus enhancing robustness under diverse user contexts (Qian et al., 2020). Its ability to reflect complex wireless propagation phenomena allows for the deduction of ambient parameters without specialized hardware, facilitating the sensing of diverse environmental traits such as crowd flows, weather conditions, and indoor pollution through fluctuations caused by moving objects (Bibbò et al., 2022; Yacchirema et al., 2019).

Recent technological advances have further bolstered CSI's appeal for fall detection. For instance, introducing 802.11ax channel bonding has expanded CSI dimensionality, significantly improving the spatiotemporal resolution necessary for detecting subtle environmental changes during postural transitions. These enhancements render CSI an ideal candidate for continuous, device-free fall monitoring through common wireless devices, offering a less invasive alternative to wearables (Rathor & Joshi, 2021; Wang et al., 2024). Falls pose a significant health concern, particularly among the elderly. Over the years, various detection methods have been explored, from wearable sensors to vision-based systems, each with challenges related to user adherence, privacy, and environmental limitations. Recently, device-free approaches using ubiquitous wireless signals have emerged as less intrusive options, with Wi-Fi-based fall detection via CSI standing out due to its fine-grained sensitivity to human motions and the widespread availability of Wi-Fi infrastructure (Wang et al., 2024).

Combining CSI's ambient sensing capabilities with advanced learning techniques opens the door to next-generation fall detection solutions that overcome many limitations of previous methods, such as non-stationary user contexts and the need for explicit user cooperation. However, the practical implementation of Wi-Fi-based fall detection faces hurdles related to CSI's instability. Standard signal preprocessing steps, while helpful, do not fully mitigate issues like fluctuating phase information and inconsistent sampling intervals. Robust noise mitigation techniques are thus critical for enhancing the quality and integrity of data, enabling deep learning models to effectively learn reliable spatiotemporal patterns from CSI sequences for fall detection applications (Palipana et al., 2018; Shi et al., 2022).

1.3.3 Expectations from Hybrid Machine Learning and Deep Learning Application

The integration of machine learning (ML) and deep learning (DL) techniques with emerging data modalities like Channel State Information (CSI) has shown significant promise in enhancing fall detection systems. Recent advancements in Wi-Fi sensing, alongside ML and DL, have not only improved fall detection capabilities but have also equipped these systems with rich spatial and temporal insights, surpassing the capabilities of traditional RSS-based approaches (Şengül et al., 2022)Recent developments have addressed some of the key challenges in fall detection, such as limited labeled data, class imbalance, and domain shifts, through hybrid ML and DL frameworks, which leverage the strengths of unsupervised and supervised learning.

ML/DL techniques have revolutionized the ability to automatically learn representations from raw data in fields like computer vision. This approach has been applied to CSI-based human sensing, where supervised algorithms have overcome the limitations of early rule-based heuristics by robustly modeling spatiotemporal signatures from large observational datasets. Ensemble deep models, combining CNN and RNN architectures, have shown enhanced accuracies, demonstrating the potential of integrating diverse algorithms like SVM, CNN, and LSTM for higher accuracy through the aggregation of diverse representations (Şengül et al., 2022; Yao et al., 2018).

Furthermore, recent efforts in employing multi-task learning, which jointly optimizes related tasks such as Human Activity Recognition (HAR) and vital monitoring alongside fall classification, have further enhanced performance. The late fusion of hand-crafted features with deep features, as well as exploiting multi-domain correlations within multi-modal data, has been effective in capturing intricate spatiotemporal patterns that are challenging to learn from isolated views (Shi et al., 2022). However, the reliance on large, labeled datasets for supervised Activities of Daily Living (ADLs) classification remains a bottleneck. Transfer learning and self-supervised methods offer partial solutions by utilizing unlabeled data for auxiliary predictive tasks, addressing issues of dataset bias, lack of generalization, and data shifts between controlled and free-living conditions (Pan et al., 2008). Advanced unsupervised, dimensionality reduction and generative models are addressing scalability challenges, with transfer learning and multitask learning improving localization accuracy and generalizability through shared low-level features (Bibbò et al., 2022; Yao et al., 2018).

Despite these advancements, most existing works have focused on limited datasets and basic isolated activities, highlighting the potential for further improvements in robustness, detection accuracy, and reliability. Integrating diverse data sources through sensor fusion and leveraging hybrid learning techniques capable of learning semantics from data heterogeneity are key areas for development. Establishing benchmark evaluation frameworks is also crucial for promoting advancements across datasets, underscoring the significant potential of ML-DL integration efforts in developing next-generation fall alerting systems that combine unobtrusiveness, precision, scalability, and ease of deployment (Wei et al., 2015).

1.3.4 Parameters of Fall Detection Technologies

Existing fall detection technologies optimize key performance parameters crucial for clinical adoption and enhancing user experiences. These parameters include detection accuracy, sensitivity (true positive rate), specificity (true negative rate), precision, and fall detection response time, which is the latency between the actual fall onset and the generation of a system alarm/alert. Advanced approaches aim to maximize these metrics through intelligent sensing and learning from heterogeneous data streams, addressing critical performance metrics such as sensitivity, specificity, latency, deployment costs, and scalability that govern real-world adoption of fall detection systems (Bagala et al., 2012; Palmerini et al., 2020). Recent shifts from supervised classification based on statistical features to unsupervised techniques leveraging spectral-temporal representations highlight the evolving landscape of technology aimed at precise localization of fall incidents for timely medical response and enhanced fall risk assessment through advanced activity (Guo et al., 2024).

Hybrid sensor fusion offers promising solutions to meet these criteria, though it introduces complexity due to multi-stream modeling (Kong et al., 2021). Other important measures of system effectiveness include minimal obtrusiveness, extensive fall-type coverage across diverse scenarios, and user compliance over prolonged periods, which are essential for preventive healthcare. Predictive capabilities that accurately distinguish accidental falls from other activities of daily living (ADL) movements are paramount for a system's clinical relevance (Casilari et al., 2020; Kraft et al., 2020).

Continuous, real-time monitoring of falls expedites emergency responses and aids in preventive interventions by offering insights into risk factors, activity patterns before falls, and individual treatments' effectiveness. This approach supports proactive geriatric care models aimed at reducing fall frequency through targeted rehabilitation strategies and enables the detection of abnormal post-fall lying behaviors that require immediate assistance. Such advancements are increasingly important given the ageing global population and the consequential healthcare and economic implications (Carretero, 2015).

However, large-scale evaluations of these technologies under real-world conditions are scarce, and most prior works have assessed single parameters in isolation rather than demonstrating reliability across diverse temporal, spatial, and contextual factors. This underscores the need for rigorous methodologies that address these heterogeneous factors simultaneously. Additionally, CSI data collected from Wi-Fi devices, a key component of many fall detection systems, presents inherent challenges such as significant variability and noise influenced by environmental factors, varying reflection patterns from different parts of the human body, and distortion of phase information by multipath reflections (Abuhoureyah et al., 2023). These issues introduce unpredictable fluctuations in the raw CSI data, negatively impacting system performance and underscoring the importance of sophisticated data processing and analysis techniques for accurate fall detection.

1.3.5 Channel State Information as a Transformative Tool in Fall Detection

Channel State Information (CSI) technology offers a promising avenue to enhance fall detection methods. Its unique signal propagation characteristics provide ambient spatiotemporal cues at frequencies below the typical radio spectrum. Unlike wearables that depend on online-of-sight and may suffer from user adherence issues, CSI utilizes pervasive wireless infrastructures, enabling device-free monitoring that can penetrate walls using subtle multipath reflections modulated during postural shifts. This innovation in fall detection methodologies comes as falls remain a significant health concern, particularly among the elderly population, prompting decades of research into various detection techniques (Sposaro & Tyson, 2009). Traditional methods have included wearable sensors and vision-based systems, which, while groundbreaking, encountered limitations such as adherence, convenience, blind spots, and privacy concerns (Noury et al., 2007).

In contrast, recent shifts toward device-free approaches using ubiquitous wireless signals, including RFID tags, sound, and Wi-Fi, have emerged due to their non-intrusive nature. Wi-Fi-based fall detection, leveraging channel state information (CSI), has become increasingly promising because of Wi-Fi's extensive deployment and CSI's sensitivity to human motion (Rathor & Joshi, 2021)Compared to RSSI, CSI's resilience to shadowing effects and its ability to capture fine-grained multipath behaviors with sub-wavelength resolution enhance the robustness and precision of fall detection across diverse user contexts (Xue et al., 2020). Advanced machine learning frameworks further support this, filtering noise from mobility and link variability and distilling meaningful patterns from raw sensory data.

Moreover, CSI's resistance to noise and attenuation, attributed to its exploitation of frequency diversity and its high resolution that allows for extracting features unique to delicate activities like falls, underscore its reliability for incident detection (X. Wu et al., 2022). Introducing 802.11ax channel bonding, which extends CSI dimensionality up to 80MHz, enhances the potential for improved spatiotemporal 1 feature representation, paving the way for next-generation assistive safety solutions. These solutions aim to overcome challenges related to user compliance, device form factors, and the reliability of fall detection under real-world conditions.

However, the practical implementation of Wi-Fi-based fall detection faces significant challenges due to the unstable nature of CSI measurements. Standard signal preprocessing methods like filtering only partially mitigate issues such as fluctuating phase information, limited usable subcarriers, and inconsistent sampling intervals. These challenges prevent deep learning models from effectively learning spatiotemporal dependency patterns from raw CSI data, highlighting the critical need for robust noise mitigation techniques. Such preprocessing enhancements are essential for improving data quality and integrity, enabling the full potential of deep neural networks in fall detection applications (Dang et al., 2020). This comprehensive blend of advanced signal processing, machine learning, and the unique attribute of CSI forms a solid foundation for the future of non-intrusive, reliable fall detection systems.

1.4 Limitations of Current Systems

Over the last decade, significant innovations in fall detection research have been observed, yet these systems still encounter critical limitations that hinder their trustworthy clinical adoption and practical deployment. Despite advancements towards real-world applications, the research community grapples with gaps preventing widespread clinical usage. Initially, studies were often limited to laboratory-simulated scenarios that did not accurately represent the variability found in in-home settings, leading to environmental sensitivity, adherence issues, or limited expressivity in single-modality systems (Bagala et al., 2012; Y. Wang et al., 2016).

Early methodologies predominantly relied on accelerometer-based detections using basic thresholds or heuristic rules, which were not robust against the uncertainties in variable living contexts and human activities. More recent efforts have employed sophisticated machine learning and deep learning (ML/DL) models to address these shortcomings. However, these approaches often depend on large, manually labeled datasets that are prone to biases and challenging to scale in heterogeneous and dynamic environments (X. Wu et al., 2022). Innovative strategies have evolved to include temporal modeling, integrating feature extractors like CNNs with sequential models such as RNN/LSTMs to leverage historical and future context, thereby capturing fine-grained motion dynamics for robust Human Activity Recognition (HAR) (Bibbò et al., 2022). Further advancements have seen the hybridization of CNN-LSTM-Attention networks, focusing on distinguishing motion patterns for enhanced activity recognition and the utilization of semi-supervised learning frameworks to improve classification with limited labeled samples (Hur et al., 2018).

However, most datasets have focused on limited scenarios, predominantly in-lab simulated falls and isolated Activities of Daily Living (ADLs), overlooking the behavioral transitions critical for preventive healthcare. The absence of reliable validation frameworks and benchmark assessments under real-world, uncontrolled conditions remains a significant barrier to the credible evaluation of system proofs. Key performance metrics for the real-world adoption of fall detection systems include sensitivity, specificity, latency, deployment costs, and scalability. Recent shifts towards unsupervised techniques that leverage spectral-temporal representations, the precise localization of fall incidents, advanced activity recognition for subtle postural transitions, and hybrid sensor fusion strategies indicate progress, though challenges such as complexity in multi-stream modeling persist (Bagala et al., 2012; Guo et al., 2024).

The reliance on wearables alone also poses a threat to long-term user compliance, highlighting the need for minimally disruptive ambient sensing and assistive approaches through cross-domain data integration (Islam et al., 2020; Tahir et al., 2022). To overcome these limitations, research must invest in technologies that facilitate viable clinical deployments. The potential of leveraging edge-cloud computational architectures to balance resources, latency, and privacy, combined with learning from multimodal pervasive sensing signals, presents significant opportunities. Dedicated studies focusing on distributed sensing, clinical integration, integrating explainable AI techniques, and continual learning methodologies are essential for overcoming data uncertainty and advancing towards human-aware AI safety and trust, moving closer to realizing fall prevention in real-world scenarios.

1.5 Objectives

The objectives of this thesis are centered around improving fall detection systems through innovative approaches. These goals aim to address the current limitations and enhance overall system performance. This research work targets the following three objectives.

1.5.1 Automatic Abrupt Labeling of Raw Signals Data

Develop an automated system that would label raw Channel State Information (CSI) signals based on abrupt changes in the observed data stream. This approach will be intended to detect various human activities, such as falls, through the recognition of significant changes in signal patterns.

This is paramount in facilitating accurate data analysis later in the stages without human intervention, ensuring that the dataset is labeled consistently to train deep learning models.

Expected Impact

It manages the issue of labor-intensive, error-prone manual data labeling and enhances the scalability of the data processing workflow, making it possible to handle larger datasets effectively.

1.5.2 Efficient Data Preprocessing to Filter Out Useless Information

Develop a strong data preprocessing pipeline to clean the raw CSI data from cluttering sources and noise. In this phase, the process will be signal-conditioned using filtering, normalization, and segmentation techniques.

Thus, clean data will be used for more accurate and efficient analysis, especially by machine learning models, where the quality of the input data influences performance significantly.

Expected impact

It reduces the computational overhead by dropping the unnecessary data before the learning process starts. This, therefore, increases the model's accuracy in providing cleaner, more relevant data for training.

1.5.3 Custom and Ensemble Deep Learning Models for Better Prediction Performance

Develop an ensemble of deep learning models optimized to predict the activity's fall accurately. This approach highlights the strengths of both Long-Short-Term Memory (LSTM) networks and One-dimensional Convolutional Neural Networks (1D-CNN) to enhance overall predictive performance.

Models

The LSTM captures temporal dependencies in the time series data, while the CNN handles spatial feature extraction from the sequential data. Our system also provides a strong solution to real-time fall detection and can be implemented in healthcare and elderly care monitoring systems. Integrating automatic data labeling and preprocessing steps to prepare the dataset.

1.6 Thesis Layout

The thesis structure is designed to explore the research domain comprehensively. Following the introduction, Chapter 2 delves into related works and outlines the specific research objectives. A thorough literature review and an in-depth introduction to the suggested feature selection technique are provided in Chapter 3, providing a framework for the ML and DL approaches used in the following studies. The outcomes of the suggested approaches are shown in Chapter 4. In Chapter 5, the thesis provides in-depth analyses of the categorization outcomes and a comprehensive assessment of the model's effectiveness. The conclusive insights and implications drawn from this study and future directions are discussed in Chapter 6.

Chapter 2
Literature Review

The number of people over 65 in the UK has been increasing gradually, and the Office for National Statistics (ONS) projects that this number will increase by an additional two million by 2025 ("Overview of the UK population: January 2021," 2021). Falls are the most frequent reason for hospitalization in this age range, and they can potentially be fatal. Most fall victims remain immobile for an hour or longer after the fall and cannot call for assistance, with over half of the falls affecting older persons' mobility (Fleming & Brayne, 2008). In the United States, over 40% of the elderly population who live at home will fall at least once a year and roughly 2.5% will end up in a hospital (Rubenstein, 2006). One of the leading causes of disability and decreased independence in the elderly is falls. Unintentional falls have been linked to 29% of disabilities in the UK for those 65 years of age or older and 32% for those 75 years of age or older (Beswick et al., 2010). In older persons, fear of falling and difficulty getting help after a fall are significant contributors to discomfort, diminished self-esteem, diminished independence, and mortality. For this reason, an effective fall detection system is critical to older individuals' physical and mental well-being.

Fall detection is important enough that many systems have already been developed. Numerous gadgets, including watches and sticky patches, are used in the context of wearable technologies (Delahoz & Labrador, 2014). These gadgets track the wearer's posture by analyzing data from their sensors—such as the accelerometer and gyroscope—and identify falls. Considering that most contemporary smartphones come with various sensors built-in, fall detection systems have also been created for them (Habib et al., 2014). The majority of wearable devices are accurate and useful. However, older persons are less likely to use them since they need to be attached to the body and require regular battery changes. When it comes to non-wearable technology, vision-based techniques are crucial for identifying falls utilizing information gathered from sensors like infrared and RGB cameras (Gutiérrez et al., 2021; Keskes & Noumeir, 2021; Shu & Shu, 2021)The most accurate systems rely on vision. However, the sensor's field of view typically limits the coverage area, and some older persons may find vision-based monitoring intrusive. Additionally, privacy concerns are associated with this kind of monitoring.

Numerous studies have already investigated fall detection methods employing Wi-Fi CSI (Damodaran et al., 2020; Nguyen & Nguyen, 2020; Palipana et al., 2018; Wang et al., 2018). Typically, these methods use three steps to identify falls: (1) CSI pre-processing, which includes resampling, filtering, and principal component analysis (PCA) (Smith, 2002); (2) CSI feature extraction, which identifies relevant data by examining features like CSI spectrogram, normalized standard deviation, entropy, and power decline ratio; and (3) event classification, which uses classifiers based on support vector machine (SVM), convolutional neural network (CNN), or long short-term memory (LSTM) to determine whether the collected CSI involves fall or non-fall events. Nevertheless, the training dataset significantly impacts these methods' detection accuracy. For instance, FallDeFi (Palipana et al., 2018) attains a maximum accuracy of 88.9% and offers a dataset containing hundreds of falls and everyday activities in various situations. The study by Damodaran et al. shows 100% accuracy with a dataset of just 80 falls carried out in a single room (Damodaran et al., 2020).

Using a ResNet-based architecture and stochastic residual blocks instead of residual blocks, ResFi uses CSI data to achieve indoor localization (Qian et al., 2020). An attention-based bidirectional LSTM system is proposed by (Khan et al., 2018) to identify various daily human actions such as walking, running, sitting, and standing up using CSI data. Yousefi et al. proposed a human behavior identification system that recognizes six distinct behaviors, including falls, using recurrent neural networks (RNNs) and long short-term memory (LSTM) (Yousefi et al., 2017). To detect six distinct hand activities and give indoor localization using CSI data, E2EDLF (Alazrai et al., 2020) detects human-to-human interactions (including high-fiving, hugging, and handshaking).

RT-Fall (H. Wang et al., 2016) and WiFall (Y. Wang et al., 2016) are the pioneering works that established the foundation for fall detection with CSI data. An SVM-based classifier is used in both methods to identify fall occurrences. By extracting features using a power burst curve (PBC) and a short-time Fourier transform (STFT) spectrogram, FallDeFi (Palipana et al., 2018) enhances the performance of WiFall and RT-Fall. FallDeFi has integrated several everyday acts that are not falling but have characteristics similar to falls, such as sitting and picking things up, in order to lower false alarms and, in some cases, reach an accuracy of up to 88.9%. A CSI-based fall detection system with a classifier combining SVM and LSTM is proposed by Damodaran et al. (Damodaran et al., 2020). Although the dataset is gathered in a single room and the sample size of fall occurrences is very small (80 samples), our system achieves 100% accuracy in fall detection. Nakamura et al. (Nakamura et al., 2020) suggest a fall detection system that uses spectrogram images produced from CSI data to identify fall events. A ResNet-based image classifier receives the spectrogram images as input to recognise falls. The accuracy of the suggested task is up to 96% in some settings. Nonetheless, the CSI data is gathered in a restricted number of contexts (two small rooms with Wi-Fi modules positioned adjacent to one another for LoS propagation), much like the work of (Damodaran et al., 2020).

Several works achieve impressive accuracy but share similar shortcomings. According to (Mattela et al., 2022) for instance, the suggested LSTM-based classifier can achieve up to 99% accuracy; however, the dataset only contains CSI collected in two contexts, and the only non-fall actions recorded were sitting up and walking. Hu (Hu et al., 2021) suggests utilizing CSI to estimate moving objects' speed and acceleration to detect falls. The work uses a dataset that includes three interior conditions and achieves a maximum detection rate of 95%. However, as the authors note, the detection rate decreases with lower-speed falls, and the work is focused on heavy falls.

2.1 Background

This section explains the ageing population growth and the causes of falls in elderly people. It further describes the current monitoring systems and limitation of fall detection such as wearable device based and vision-based systems. In the end, this section talks about the device free sensing system for fall detection which has been implemented for this research.

2.1.1 Growth Rate in Ageing Population

Due to rising human life expectancy, the proportion of persons aged 65 and up in the global population is steadily increasing. By 2040, the population of this age group will be about a fifth of that of the 20-64 age group (Palipana et al., 2018). According to the United Nations, 25% of the world population will be elderly by 2050. According to data published by the United Nations Department of Economic and Social Affairs Population Dynamics, the global population of adults aged 60 has more than doubled from 8.0 percent to 13.5 percent between 1950 and 2020. The estimated results reveal that the ageing population is more likely to increase in the years to come. By 2035, the population of people older than 65 years in New Zealand will be around a quarter of the entire population. Some elderly people experience some decline in their functional capacity. In terms of functional capacity, one major element is related to the injury. In elderly people, various injuries can affect their quality of life. One such common injury is falling for specific reasons such as balance problems, muscle weakness, poor vision, low blood pressure, etc. If the falling cannot be determined promptly, then it might cause serious repercussions, which include but are not limited to fracture, loss of blood, and even fatality in the worst cases. In case of such an event, the elderly person cannot even inform others about the incident due to the severe effect of the falling; either they are unconscious or cannot move or act due to injury. The Centers for Disease Control and Prevention found that one-third of elderly people aged 65 and above experience falls yearly at home (Burns et al., 2016). According to the United Nations, nearly one-third of persons in this age range are at risk of falling each year, and one out of every five people who fall suffers head injuries or fractured bones in the wrists, arms, ankles, and hips. After a fall, many people who are not injured cannot get up on their own. This can lead to a fear of falling, a loss of confidence in their ability to live independently, a lack of physical activity, a lack of social connections, depression, bad quality of life, and even mortality (Y. Wang et al., 2016). In addition to physical injuries and expensive medical costs, falls may also inflict psychological harm to the elderly, which fall experts refer to as the fear of falling cycle.

Unless faced with severe health hazards and issues that might require elderly people to be admitted to the hospital, they are advised to stay indoors as much as possible. However, this approach is applicable and suitable only if their well-being is ensured through an effective health system that would help in keeping a close check on their activities at home (Shen et al., 2024). If instant medical assistance is not provided after a fall, then that can result in increasing the likelihood of more deaths of elderly people; nearly 50% of such people whose medical treatment was delayed were found to be dead within six months after such an incident (Y. Wang et al., 2016). Moreover, falls not only cause physical damage and increased medical costs but also result in psychological damage.

This trend is named the fear of falling cycle by the fall researchers. In addition to physical injuries and high medical costs, falls can cause psychological damage to the elderly as well, which is known as the fear of falling cycle by researchers (Koshmak et al., 2016). With the population ageing faster than ever, monitoring the movements of elderly people has become necessary without compromising their privacy or hindering their day-to-day activities.

2.1.2 Causes of Elderly Falls

For the elderly, falls are the primary cause of both fatal and nonfatal injuries. As a result, a system that can detect a fall and alert caretakers or family in a timely manner might be a valuable healthcare tool. A reliable fall detection system benefits the older population in two ways: it minimizes the time that persons who have fallen spend on the floor and lessens the fear of falling (Patil, 2015). Looking after the elderly population and enabling them to have a functional healthcare system is the need of the hour (Kim et al., 2020), as falling can inhibit the ageing population from having an independent life and make them rely on other individuals' support for their remaining lives. One of the ways this can be dealt with is by ensuring the presence of a person for continuous care and assistance; however, it is not a convenient option as a person cannot be there for another person all the time. At times, old-aged people reside with their relatives, but other times, they are left alone in their homes or elderly care centers. Falling while living alone can be dangerous as the rescue time can take longer than otherwise. Thus, to address this problem, the development and implementation of automatic fall detection systems must be implemented to detect falls and ensure the older population's safety. Such a system will help caregivers and relatives detect a fall when it happens, and it has proven to be an excellent tool for assisting them within a short time.

2.1.3 Current Monitoring Systems

Currently, many health-monitoring approaches are already adopted and popular in people's daily lives, and a variety of applications are widely developed (Maia et al., 2018). If you look around, technological development has provided many devices which aim to care for elderly people. One such available device in the market is the wearable sensor, and another is computer vision technology; each has certain limitations. Wearable devices are impossible to be worn throughout the whole day because many elderly people can have a weak memory and they tend to forget things easily (Hu et al., 2021). Another issue related to wearable devices is that they need to be recharged after every 8 to 10 hours. If anything happens to elderly people while charging the wearable devices, there is no point in using them. Computer vision technology has its limitations as well. Not everyone can behave naturally in front of it due to a psychological reason for always looking and feeling fine, which causes them to alter their behavior, and the results will be misleading. Another reason for being not much feasible for computer vision technology is its cost, and one cannot easily afford to install it in every corner of the room/house, which requires constant maintenance as well (H. Wang et al., 2016).

The human fall detection systems can only work effectively if the difference between a fall and other activities of daily life (ADL for short), such as walking, standing, sitting, and lying, is made clear to them by providing training. However, differentiating between a human fall and an ADL can be tricky as certain ADLs, like lying on the floor or getting down from a standing position, are quite like general falls. Thus, proper training about falls and ADLs needs to be provided to these systems by collecting relevant data to get the best advantage of these systems. Data collection can be done with the help of numerous kinds of sensors installed in the environment, such as pressure sensors, floor vibration sensors, infrared sensors, microphones, and cameras. This data is then presented in the form of acceleration signals, pressure signals, audio, or videos, which are further processed and moved on to the classifier that basically makes the distinction of it being either a fall or an ADL.

Categorization of human fall detection systems generally depends on the source of data collection, and they are further classified into two major categories: i) wearable device-based systems and ii) vision-based systems.

2.1.4 Wearable Device-based Fall Detection Systems

These electronic devices are worn by a person either on top or beneath their clothes. Such devices contain sensors like gyroscopes, accelerometers, tilt meters, and oscilloscopes, which help detect a fall (Singh et al., 2020). A person must always carry these devices with them, thus providing location independence for elderly people to move around freely. However, it is quite possible that the old-aged person may not carry this device for various reasons like forgetfulness, uneasiness, or poor charging of a device's battery. Keeping in view these constraints of such devices, researchers have come up with a novel detection system known as the contact-free fall detection system, which does not require a person to wear or carry anything along with them to ensure their safety (Islam et al., 2020).

2.1.5 Vision-based Fall Detection Systems

These systems track real-time human movements with the help of either a regular video camera or a depth video camera. Moreover, an algorithm is run in the background to determine a person’s posture, and an alarm goes off as soon as a fall is detected to inform the caretakers about any help that the falling person might need (Gutiérrez et al., 2021)The problem with the vision-based system is that it is quite expensive to implement in elderly care centres. Moreover, if the video camera malfunctions at night, it is difficult to fix it, and most elderly accidents happen at night.

2.1.6 Device-free Sensing Approaches for Fall Detection Systems

Device-free sensing is a method recently introduced to monitor the presence, location, motion, activity, and gestures of a person without the help of any attached device (Zhang et al., 2020). This approach functions with wireless signals, which are a great communication tool as these signals are all around us regardless of where we are; for instance, Wi-Fi, 3G/4G, FM, and TV are some of the most common signals that happen to be everywhere. Any movement done by the person present within the premises of the deployment area of the wireless network can be instantly gauged by the wireless signal patterns and characteristics. Moreover, unlike other state-of-the-art sensing techniques, such as vision-based systems, device-free sensing works exceptionally well in smoky or dark conditions. This technique also prevents privacy intrusion. This approach's features make it a desirable option for pervasive sensing applications.

Device-free sensing techniques mainly support diffraction, reflection, and scattering phenomena that a person forces on wireless links to observe the state of human activity. A person present within the region of a wireless network will inevitably disrupt the propagation of wireless signals. Moreover, researchers claim that the influence of a person on wireless signals is conventional and repetitive, making it easy to determine the human state by examining the wireless signal patterns and characteristics (Y. Wang et al., 2016).

The said technique also provides context information, offering immense opportunities for the latest services. Such budding applications can be classified into two major sections: The first category involves assessment of the human state patterns that can aid in helping public services like security monitoring and emergency rescue, while the second one entails analysis of human state in order to give personal services such as intelligent interaction and intelligent monitoring; thus, it can be deduced that the device free sensing technique is a meaningful tool by the use of which a great amount of data about the human state can be extracted through numerous effective data analysis methods (Kianoush et al., 2016). The important advantages of device-free sensing-based applications include that they can improve their sensing tasks by efficiently updating the firmware and protocol of traditional wireless networks, thereby reducing the deployment of new hardware and taking advantage of the ubiquitous nature of wireless signals, hence sensing and gauging human state patterns anywhere and everywhere.

Data collection methods largely comprise video cameras and/or wearable sensors that may enhance the chance of privacy intrusion. Furthermore, the active participation of the volunteers can also be a hindrance in the data collection and identification process as the anonymity of the people is compromised, which may make them feel uneasy and unnatural in responding to the situations they would otherwise if the cameras were not around. Therefore, the selection of an appropriate method that can function passively is mandatory for the data collection and people identification process. Due to the very same reason, adoption of device-free sensing is the most apt option as the volunteers do not necessarily have to equip themselves with any sensing devices and can actively participate in the data collection process by keeping their anonymity intact (Chen et al., 2022). This, in turn, would avoid all the sensing coverage and key management hassles caused when sensor-based approaches are used. According to some researchers, utilization of Wi-Fi signals can be done in device-free sensing systems as Wi-Fi networks are extensively used solely because they are openly accessed and for ease of installation. These networks operate as a communication tool and work as generalized heterogeneous sensors.

Moreover, a lot of research work has been done on device-free sensing, such as indoor localization, activity recognition, etc., but they are largely built on received signal strength indicator (RSSI), which is highly dependent on multipath effect (Ngamakeur et al., 2020). In recent times, it has been explored by researchers that Channel State Information (CSI) is a measurement in depth which propagates signals from transmitter to receiver (Xiao et al., 2016). It is a measurement in the physical layer of wireless networks, and amplitude and phase information are also included in CSI. Consequently, it works quite stably in static situations, fluctuates when environmental conditions change, and can be withdrawn with few firmware changes from commodity Wi-Fi devices. Moreover, many research studies have explored pervasive sensing using CSI, such as human detection (Alam et al., 2020). Therefore, this work uses Wi-Fi as the sensing technology for fall detection.

According to current research, shallow machine learning algorithms cannot handle human sensing tasks at higher scales, where deep learning has significant potential. Practical RF sensing systems will be anticipated to perform consistently over a broad user population, activities, physical surroundings, and RF devices, demonstrating that human sensing can scale in many dimensions. As a result, all these characteristics could be explored in deep learning research.

The more data used, the better the model will be trained and the more effective the results will be. It has been observed that the data was collected in a short span of time, i.e., a few days or weeks. This is because data collection is a lengthy process that requires a great amount of time, and dedicating that much ample time is not possible for every researcher. For example, the data used for fall detection was collected in just 6 or 7 days (Y. Wang et al., 2016). If it were to be expanded, then more efficient results could be produced. If this data is further expanded using the Generative Adversarial Networks Algorithm and the difference is calculated, then the chances of having improved results increase many times. Previous studies have gathered limited data, and with such limited information available, the results generated cannot be regarded as accurate, which also causes less training in Deep Learning. Generative Adversarial Networks Algorithms can convert small datasets into larger ones, improving the results' reliability.

2.2 Machine Learning and Deep Learning Frameworks for Fall Detection

Machine learning algorithms harness historical data to detect patterns, enabling predictions on new, unlabeled samples. Supervised algorithms learn mappings between inputs and outputs using labeled training datasets, while unsupervised techniques uncover underlying structures without labeled responses, setting a foundation for diverse applications, including fall detection (Koshmak et al., 2016). The quality and standardization of Channel State Information (CSI) data are crucial for training deep neural networks, aiming to achieve human-level intelligence in practical fall detection scenarios. These deep models, which consist of millions of trainable parameters, learn complex hierarchical representations directly from the data, bypassing the need for handcrafted heuristics. This ability allows them to potentially excel in recognizing intricate patterns in raw CSI data that are influenced by environmental factors, although these same intricacies present challenges in generalizing the underlying temporal patterns associated with human movements (Patil, 2015).

Deep learning models employ multilayer nonlinear processing, which improves performance in direct proportion to problem complexity and the availability of training resources. Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) are particularly effective, with CNNs learning hierarchical feature representations from grid-structured data like images and RNNs processing variable-length sequences through stateful cycles. This makes them well-suited for tasks in computer vision and natural language processing (NLP), achieving human-level performance by learning representation hierarchies directly from high-dimensional input spaces. Similarly, RNN variants have been instrumental in modeling time-series healthcare data, including sensor streams and clinical records, through their ability to handle sequential and temporal patterns (Kim et al., 2020).

2.2.1 Data Preprocessing and Feature Extraction

The quest for enhancing the precision and reliability of fall detection systems using Wi-Fi sensing pivots around the meticulous preprocessing and analysis of Channel State Information (CSI) traces. The normalization of raw CSI traces to standardized distributions through zero-mean unit-variance transformations effectively addresses scale disparities, setting the foundation for more effective learning processes by bounding amplitudes. The stability of channel strength and phase over time is instrumental in tracking spatiotemporal patterns, which are crucial for accurately identifying falls (Liu et al., 2019). However, the inherently unstable nature of CSI measurements poses significant challenges, including fluctuating phase information, limited usable subcarriers, and inconsistent sampling intervals. Although standard signal preprocessing steps like filtering mitigate channel noise to some extent, they do not fully overcome these obstacles, underscoring the necessity for robust noise mitigation techniques to improve the reliability of deep learning models in capturing spatiotemporal dependencies from CSI sequences (Palipana et al., 2018).

Advancements in signal processing techniques further refine the extraction of features encoding contextual information inherent to human activities. Methods such as the Discrete Fourier Transform (DFT) and Short-Time Fourier Transform (STFT) convert CSI time series into joint Time-Frequency representations, capturing the periodic nature of gait and the transient aberrations associated with impacts. These techniques facilitate the classification of activities by highlighting characteristic periodicities and frequency transitions during postural shifts, thus reducing the reliance on manual feature engineering (Palipana et al., 2018)The emergence of Deep Spectrum Learning exemplifies the strides made in utilizing frequency patterns for activity recognition. It offers superior performance compared to traditional time or cepstral coefficients-based models.

In parallel, the implementation of cross-validation and dimensionality reduction techniques rescales and refines the feature set to enhance generalizability against site-specific characteristics, thereby minimizing bias and variance (Chowdhury, 2018). The landscape of CSI-based fall detection is further broadened by integrating advanced machine learning strategies such as transfer learning and multitask learning. These approaches address dataset bias, enhance generalization across different environments, and facilitate the joint execution of localization and activity classification tasks. By leveraging shared low-level features, these methodologies underscore the potential for creating next-generation fall detection systems that marry unobtrusiveness with precision, scalability, and ease of deployment (Rezaei, 2023). This holistic approach to preprocessing, feature extraction, and machine learning integration illuminates a path forward for the development of sophisticated fall alerting mechanisms, poised to revolutionize the landscape of elderly care and mobility assistance.

2.2.2 Model Selection and Training

In the evolving landscape of healthcare technology, the application of machine learning (ML) algorithms, particularly in the domain of fall detection, showcases a significant transformation from reliance on traditional methods to the adoption of advanced deep learning techniques. Supervised ML algorithms have been foundational, utilizing labeled training samples to classify and predict outcomes with methods such as K-nearest neighbors, which classify unlabeled vectors based on distances to known exemplars, decision trees that recursively split multi-dimensional spaces via entropy tests, and Bayes' classifiers that compute class membership likelihoods from attribute distributions (Damodaran et al., 2020). Early applications in healthcare leveraged traditional ML algorithms like support vector machines (SVM), decision trees, and K-nearest neighbors (KNN), utilizing handcrafted features from domain experts. Notably, Boosted Decision Trees were employed to enhance fall risk prediction from accelerometry data (Mattela et al., 2022), illustrating the initial forays into employing ML for predictive healthcare.

The advent of deep learning from the mid-2010s onwards marked a paradigm shift towards automatic representation learning from raw data, bypassing the need for hand-engineered features. This approach, characterized by training models end-to-end, revolutionized the field by enabling convolutional neural networks (CNNs) to extract spatial patterns and recurrent neural networks (RNNs) to capture temporal dynamics efficiently. Additionally, attention-based models have been developed to selectively focus on and learn relevant attributes, thus enhancing recognition capabilities (Chu et al., 2023). The emphasis on high-quality and standardized channel state information (CSI) data became pivotal for training deep neural networks effectively, aiming to achieve human-level intelligence in practical fall detection applications (Surasakhon et al., 2022).

Ensemble methods and the exploration of novel ML approaches have further expanded the scope of fall detection. Techniques such as boosted decision trees and random forests enhance prediction robustness by aggregating diverse models' opinions and decorrelating outputs through iterative fitting and bootstrap aggregation (Chowdhury, 2018). Innovations that employ CNN-LSTM models for classifying falls versus activities of daily living and localizing fall locations have achieved significant accuracy improvements, demonstrating the potential of integrating diverse methodologies. Moreover, the application of transfer learning and ensemble methods combining CNNs, RNNs, and gradient boosting has not only improved accuracy but also addressed challenges posed by dataset variability and the need for models to generalize across different environments (Surasakhon et al., 2022). The comprehensive evolution from traditional ML algorithms to sophisticated deep learning models and ensemble methods underscores the dynamic nature of fall detection technologies. By harnessing the power of advanced ML techniques and ensuring data integrity, the field is moving closer to developing systems that can accurately predict and detect falls, offering promising enhancements to healthcare monitoring and elderly care.

2.3 Deep Learning Innovations in CSI Utilization

This section describes the enhancing features of the different algorithms and how to use them in real-time applications.

2.3.1 Enhancing Detection Algorithms

Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) represent the cornerstone of modern pattern recognition, driving advancements in human activity and fall detection technologies. CNNs excel in extracting localized features through translation-invariant convolutional filters applied across space and time domains, facilitating learning spatial patterns in 2D images and 1D temporal sequences. This capability is further enhanced by backpropagated errors that drive adaptation, improving classification outcomes (Chu et al., 2023). On the other hand, RNNs capture temporal patterns effectively by maintaining memory states that are continuously updated from input sequences and feedback, showcasing their strength in sequence data processing (Cheng et al., 2019).

The application of these deep learning methods has been revolutionized further by leveraging Wi-Fi signals for human activity and fall detection. Notably, the use of CNNs has been extended to analyze Wi-Fi signals, where architectures incorporating attention mechanisms have demonstrated remarkable accuracy. For instance, a CNN architecture with attention to identifying hand gestures using Wi-Fi achieved a 97.65% accuracy rate (Moshiri et al., 2021), while another application of CNNs on time-frequency representation of Doppler shift in Wi-Fi for human activity recognition reported an 89.3% accuracy (Moshiri et al., 2021). Attention mechanisms, such as self-attention and multi-head attention, play a pivotal role in these advancements by enabling models to focus selectively on informative data areas, thereby enriching feature representation and learning diverse patterns in parallel.

Hybrid architectures combining CNNs and RNNs have shown promising results in enhancing the efficacy of recognition systems. ConvLSTM models, for example, treat Channel State Information (CSI) as a sequence of frames, applying CNN filters along both temporal and feature dimensions to learn spatiotemporal dependencies end-to-end. These models benefit significantly from attention mechanisms and bolster performance by focusing model capacity on discriminative portions of the signal indicative of different human motions. Novel data representations, such as treating CSI's real and imaginary components as color channel inputs to CNNs, have bypassed manual feature engineering, further enhancing recognition efficacy. (Shang et al., 2021).

Transfer learning and domain adaptation are vital strategies in improving the generalization and performance of activity recognition systems. Pre-training models on related auxiliary tasks and aligning distributions between source and target domains help transfer gains and improve generalization, especially when sampling protocols differ (Mattela et al., 2022). Data augmentation techniques, including time-warping, scaling, and the use of Generative Adversarial Networks (GANs) to synthesize realistic samples, play a crucial role in expanding datasets and reducing overfitting (Wang et al., 2021). These approaches, along with curriculum learning, which sequences samples in increasing order of difficulty, optimize learning processes and enhance model efficacy for activity recognition and fall detection (Huang et al., 2021).

2.3.2 Deep neural networks for real-time processing

Efficient model architectures are pivotal in optimizing the trade-offs between speed and accuracy, particularly in resource-constrained contexts. Techniques such as the squeeze-and-excitation modules, which recalibrate feature maps to emphasize more informative channels, and the use of MobileNets that leverage depth-wise separable convolutions to decompose standard kernels, are at the forefront of these advancements (Y. Chen et al., 2020). Additionally, Channel State Information (CSI) plays a crucial role in capturing small-scale fading statistics, enabling link adaptation through the flexible adjustment of transmission parameters to match short-term channel conditions, thus optimizing throughput and boosting spectral efficiency by allowing modulation level, coding rate, and spatial stream number selection to be tailored on a per-carrier basis (Chu et al., 2023).

To further enhance efficiency, model compression techniques reduce model sizes by pruning dispensable connections, applying quantization to restrict weights to low-bit representations, and employing hybrid approaches. Knowledge distillation, for instance, facilitates the transfer of softened outputs from large, ensembled teacher models to smaller, more nimble student models, streamlining the complexity of deep learning models without significant losses in performance. In the realm of continuous monitoring applications such as elderly care, where real-time processing is paramount, which has significantly reduced computational complexity (Chen et al., 2024). Their work on a Knowledge Distillation framework demonstrates this by training a student LSTM model that is 60x faster than standard LSTMs for fall detection and achieves 95% accuracy using Wi-Fi.

Edge intelligence further enhances the deployment of trained models directly on end devices, supported by dedicated hardware accelerators like ARM ML Processors, which expedite embedded inference. This approach is complemented by federated learning, which iteratively updates global models by coordinating decentralized optimizers, thus preserving privacy while ensuring efficient model updates (Vedula, 2021). The strategic co-design of hardware and software, leveraging embedded accelerators, facilitates efficient embedded and edge processing for streaming analytics, addressing the compute, memory, and energy constraints inherent in edge deployments. Moreover, advances in decentralized, incentive-compatible, and privacy-preserving federated learning techniques help reconcile performance, trustworthiness, and sustainability needs in modern AI applications.

2.4 Challenges with CSI Data

Recent advancements in Deep Learning (DL) methodologies have significantly impacted the analysis and utilization of Channel State Information (CSI) for various applications, including accurate fall detection. Notably, DL approaches that operate directly on raw CSI sequences have gained prominence. Hybrid Convolutional Neural Network-Recurrent Neural Network (CNN-RNN) architectures, for example, have been systematically capturing both spatial and temporal patterns in sequential CSI data, thereby obviating the need for manual feature engineering steps. Such innovations also include novel data representations, where CSI's real and imaginary components are treated similarly to color channels in CNNs, thereby strengthening the learning of discriminative features (Xu et al., 2018). These techniques have begun to address significant challenges, such as dataset bias and the lack of generalization that have previously hampered CSI-based approaches.

CSI data, however, is not without its challenges. It suffers from multi-path effects and device differences that limit the generalization capabilities of plug-and-play training models. Signals from Wi-Fi transmitters, for instance, can reflect unpredictably off environmental obstructions, distorting their amplitude and phase. Moreover, distinct wireless hardware brands exhibit divergent measurement properties, which hinders model transferability (Damodaran et al., 2020). CSI data collected from Wi-Fi devices, therefore, exhibits high variability and noise due to factors like reflections, temperature variations, and humidity levels, among others. These issues introduce unpredictable fluctuations and inconsistencies in raw CSI data, negatively impacting system performance and accuracy in tasks such as fall detection.

The dynamic nature of environments further complicates modeling efforts through abrupt context switches that confound expectations. Changes in the arrangement of people, partitions, and furniture modify RF propagation properties, necessitating adaptive modeling strategies that balance exploration and exploitation (Yang et al., 2022). Recent studies have expanded the scope of fall detection by utilizing CNN-LSTM models to classify falls versus activities of daily living and even localize the fall location with significant accuracy. Ensemble methods combining CNNs, RNNs, and gradient boosting have improved accuracy through model diversity. Additionally, the application of transfer learning across datasets has enabled models to achieve high accuracy levels even when trained and tested across different indoor environments, showcasing the potential of DL approaches in overcoming the inherent challenges of CSI data for fall detection.

Integrating recurrent, convolutional, or self-attention architectures has been crucial in addressing the processing challenges posed by streaming time-series data. Hybrid architectures combining CNNs and RNNs, such as ConvLSTM models, leverage the sequential nature of CSI data, applying CNN filters along both temporal and feature dimensions to learn spatiotemporal dependencies end-to-end (Chowdhury, 2018). Attention mechanisms further enhance performance by focusing model capacity on discriminative portions of the signal indicative of different human motions. These advancements in data representation and analysis techniques, bypassing traditional manual feature engineering, underscore the evolving landscape of DL applications in effectively utilizing CSI for robust and accurate fall detection.

2.5 Deep Learning

Deep learning is a sort of machine learning that extracts valuable information from significant volumes of data using artificial neural networks (ANNs) with numerous layers of linked neurons. Almost every neuron employs an activation function to create an output signal, frequently the consequence of a series of weighted inputs supplied by neurons in neighboring layers. When all these weights are modified in line with the input of data accessible in bulk quantities into the network during the training phase, successful learning typically happens. Earlier, these deep neural networks were not deemed feasible as they require a very long time and extensive computing resources to train. However, with the advancement of computing architectures like graphical processing units (GPUs) and algorithmic breakthroughs during the training processes, for instance, work by (Mahdi et al., 2021). This phenomenon is now more affordable, which has aided research efforts to investigate and uncover novel deep-processing learning architectures and their applications in many fields, such as face recognition, image processing, natural language processing, etc.

Recent research has resulted in the establishment of various deep learning architectures, each with its own set of characteristics and benefits. Some are very particular, developed for extremely specific uses, while others encompass broad application principles. This section goes through some of the most common generic designs that have lately been related to RF sensing. However, before describing specific deep learning architecture, clarifying a few critical ideas linked to training and application is crucial. When the data used for training is not required to be labelled, a deep learning architecture does not need to be supervised to operate. Data labeling, on the other hand, need supervision, particularly for deep learning, because such systems demand a large quantity of data for training. Some use cases require supervised learning, while others may be solved using unsupervised deep learning. The term "generative" refers to deep learning architectures that are designed and trained to produce fresh data samples. They include image-to-image translation, text-to-image translation, clothing translation, and 3D object development, among other things. Various aspects and use cases of widely used deep learning architectures will be investigated in the next part, which will be employed in the research.

Recurrent Neural Networks (RNNs) were first created to handle sequence prediction issues utilizing a feedback technique in each recurrent unit. Because of the concealed nature of these unit connections, temporal aspects of the inputs can be remembered. RNNs, on the other hand, have two drawbacks: The vanishing gradient problem occurs when gradient updates become obsolete, resulting in a decrease in network learning (Ding & Wang, 2020). Exploded gradient problem is the other issue caused when the cumulative weights' gradients give a large amount of update in backpropagation. These loopholes raised many questions among researchers, and they declared RNNs to be a less favorable option as they are challenging to train. Their views changed after the evolution of variants called Long Short-term Memory (LSTM) and Gates Recurrent Unit (GRU) (Shang et al., 2021). With the help of these variants, multiple functions and copying/concatenation were included to learn long-term dependencies of the inputs, unlike before, where single non-linear activation functions were used. The amount of internal activation functions and how the interconnections are maintained are the differences between LSTM and GRU. RNN successors have been used to solve various sequence detection issues, including natural language processing.

Convolutional Neural Networks (CNN) or ConVets are neural networks used to analyze visual pictures of rows and columns of pixels. They intend to collaborate with 2D grid-like inputs with spatial relationships between them. CNNs use a set of filters (or kernels) to convolve in the inputs to learn spatial features (Shalaby et al., 2022)When numerous layers are formed, CNNs adapt to hierarchical representations from the given data set. Pooling layers are also implemented to reduce the learnt dimensionality during network development. Although CNNs are designed to deal with pictures, they are extremely good at learning spatial connections in one-dimensional objects, like the order relationship among words in a text document or the steps in a time series.

The second type of unsupervised generative deep learning architecture designed to interpret any data distribution from a training set is Generative Adversarial Networks (GANs) (Wang et al., 2021). There are two networks: generator and discriminator. The former network tries to deceive the latter by creating false samples that seem identical to the actual ones. A discriminator distinguishes the genuine sample from the created false sample. The functionality of the two networks improves with time, and the training sessions end when the discriminator can no longer distinguish between the produced and actual ones. With many variants of GAN models in state-of-the-art, GANs have revolutionized deep learning research (Huang et al., 2021).

2.5.1 Why To Use Deep Learning in Device-Free Sensing

Because of the many objects and people in the surroundings, mapping Radio frequency signals to humans and their activities can be difficult. Often, a problem cannot be figured out mathematically, thus resulting in the adoption of machine learning for Radiofrequency human sensing. Deep learning enables the feature of great flexibility for the researchers to gauge the depth of the learning networks until the sensing application is not fully developed and learned. Radio Frequency data for any human scene can be produced with better radio hardware and protocols such as multi-input-multi-output (MIMO) systems, multi-antenna radar sensors, etc. (Damodaran et al., 2020). This data, in turn, helps to train deep neural networks. Thus, it can be concluded that deep learning is a new phenomenon that not only enables RF sensing on multiple fronts, like improving the accuracy and scale of current sensing applications, but also develops generalized models that are adaptive to different and even unseen environments (Klein Brinke, 2018; Zou et al., 2019).

Chapter 3

Methodology

The methodology chapter explains the overall methodology proposed by this research. First, the raw Channel State Information (CSI) signals are pre-processed with the standard preprocessing pipeline. Automatic data labeling is proposed along with the pre-processed signals that are given to One-dimensional (1D) Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM) models. The classification results of these two models are compared with those of some state-of-the-art research works. The overall methodology is shown in Figure 3.1.

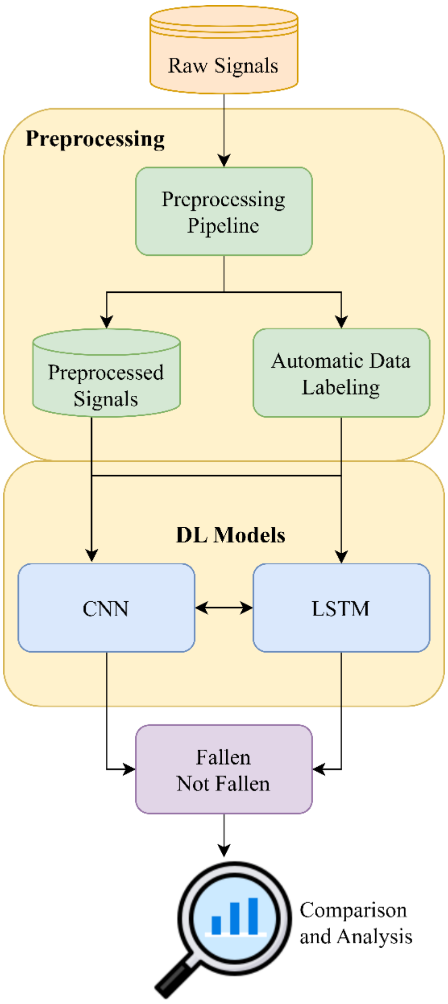

Figure 3.1. Overall proposed methodology workflow. Raw input signals are preprocessed and given to the input of CNN and LSTM models, followed by the comparative analysis.

3.1 Data Description and Analysis

The dataset used in this research was provided by Brinke and Meratnia back in 2019 (Brinke & Meratnia, 2019). The authors have created a unique device called a transceiver node. This device mainly used a Gigabyte Brix IoT as its core. They made some changes to the Brix hardware to make it work with an Intel Ultimate Wi-Fi Link 5300 NIC, shown in Figure 3.2. The Intel NIC has been chosen because the authors wanted to use a platform called open CSI by D. Halperin (Halperin et al., 2011). Moreover, the authors initially tried using small computers (micro-PCs), but they did not work well with the Intel 5300, likely because the Intel 5300 is quite old. You can find the detailed specifications of the final design in Table 3.1.

Table 3.1. The hardware specification table

Component	**Specifications**
Processor	Intel Apollo Lake N34500
RAM	1x HyperX 8GB DRR3L-SO DIMM 1866 MHz
Hard drive	Transcend MTS800 SSD 128 GB (M.2 2280)
Graphics card	None
Wireless adapter	Intel N Ultimate Wi-Fi Link 5300
Size	165x105x27mm
Operating System	Ubuntu 14.04.4

The software created for the receiver node was a tool to gather CSI data. It allowed us to collect this data for different durations and at various sampling rates, depending on the specific task. The receiver node sends a ping to the access point to collect data. When the access point responded, the authors recorded the CSI, which includes details about the signal's amplitude and phase, for 30 subcarriers. So, the sampling rate in the dataset does not refer to how many data frames we collect per second but rather how often the receiver node pings the access point. Once the data was collected, the receiver node connected to a Raspberry Pi server with an external hard drive. It stored the compiled files on the hard drive in an organized manner. This synchronization with the server was done over the same network used during data collection activities.

Figure 3.2. The hardware used for data collection

3.1.1 Experimental Setup for Data Collection

To create a dataset reminiscent of day-to-day living, the authors used an actual (small) living room in student housing, as shown in Figure 3.3. The living area is approximately 379×345cm and enclosed by two concrete walls (379cm), an entire glass wall (305 cm), and an "open space," partly blocked by a rigid plastic toilet box (179cm), leading into the kitchen, and sleeping area. The total dimensions of the studio are 861 x 345cm. The dataset is unique in that data is collected from nine participants over three days, while two participants repeated the experiments over three days. The participants on the first three days are not the same, while for the last three days, the participants remain the same. The activities performed by the participants include i) jumping, ii) walking, iii) falling, iv) clapping, v) waving, and vi) sitting/doing nothing. The number of jumping samples is 450, while other activities contain almost 750 samples, as depicted in Figure 3.4. It allows researchers to test their model on i) the same participant (50 trials per activity), ii) different participants on the same day, iii) different participants over different days and iv) the same participants over different days.

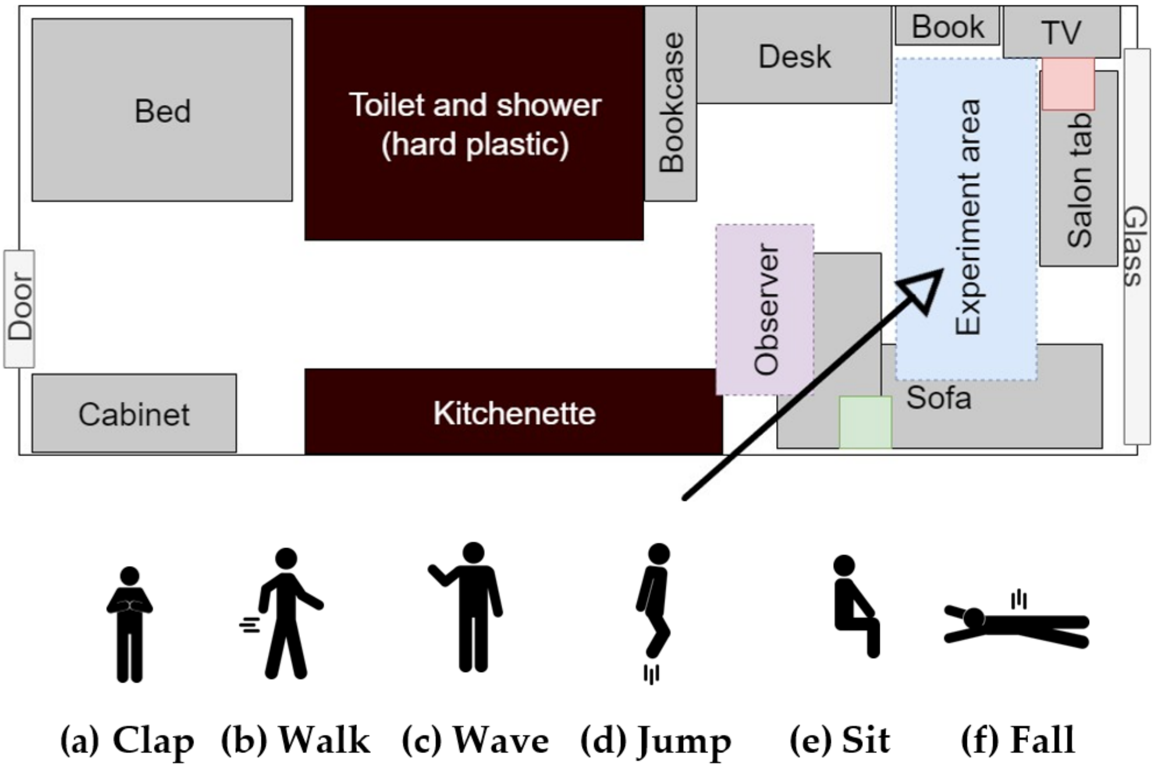

Figure 3.3. The experiment studio layout and a depiction of the activities conducted are illustrated. Red and green squares represent the transmitter and receiver.

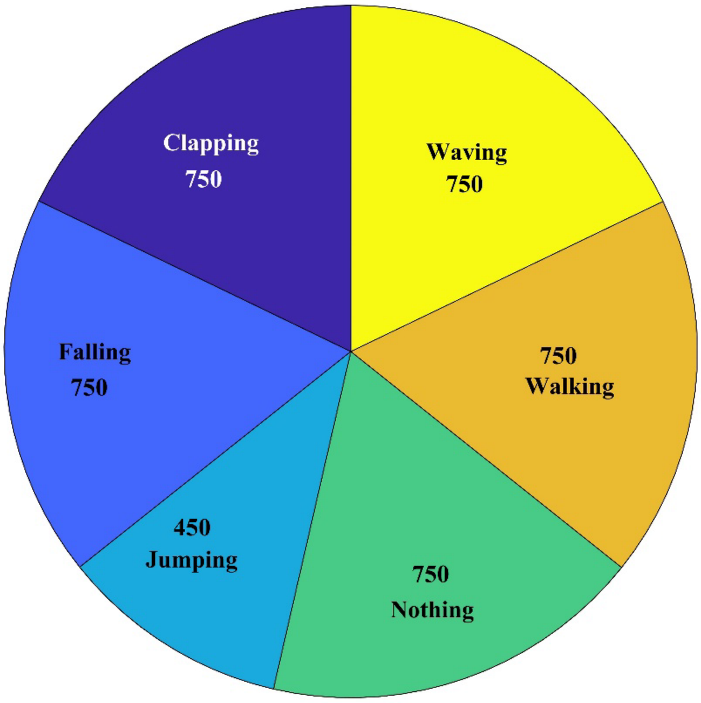

Figure 3.4. The distribution of all activities in the provided dataset. The number of “Jumping” is less than the other activities.

Furthermore, three different transmitters and receivers produce nine signals overall for a single experiment. A raw falling and nothing signal of Participant 2, for day 3, trail 50, is depicted in Figure 3.5 and Figure 3.6. The X-axis shows the time taken (number of frames with time) for the activities, while the Y-axis shows the CSI signals' amplitude. The difference between the maximum and minimum amplitude of the activity “Falling” is ~75, and “Sitting”, or non-falling, has a difference of ~25 between maximum and minimum amplitude, as depicted in Figure 3.5 and Figure 3.6, respectively.

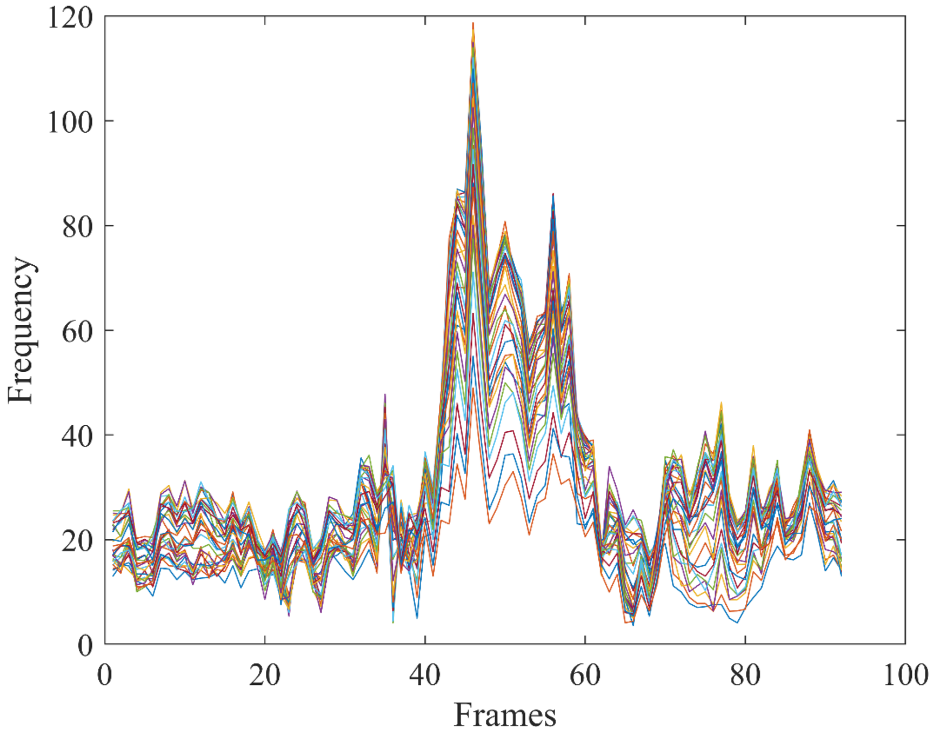

Figure 3.5. Raw “Falling” activity CSI signal.

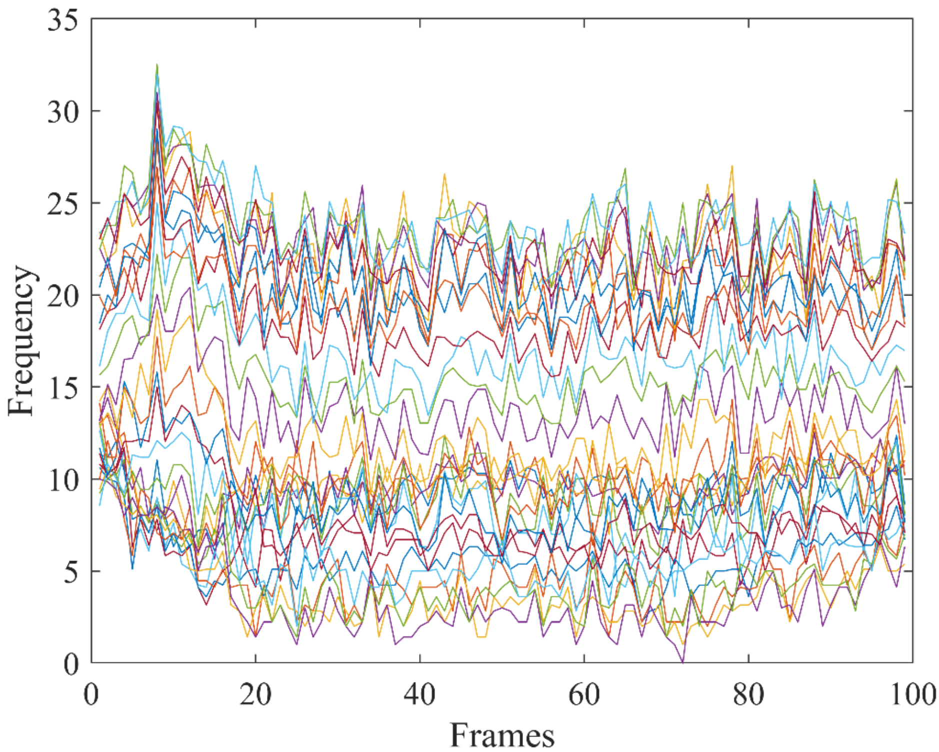

Figure 3.6. Raw “Sitting/Nothing” activity CSI signal.

The total number of participants, days, activities, samples, and average frames per sample are depicted in

Table 3.2. Similarly, the number of samples in each activity and the minimum, maximum, and mean frames in each activity are presented in Table 3.3. Calculating mean frames helps obtain a specific length of input signals for ML and DL models. Our average size is 97 frames per sample.

Table 3.2. The details of the dataset

S. No.	Thing	Quantity
1	Total Participants	9
2	Days	6
3	Activities	6
4	Transmitters	3
5	Receivers	3
6	Total samples	4199
7	Average frames per sample	97.23

Table 3.3. The statistics of the frames performed by different activities in the dataset. These statistics aim to equalize the length (number of frames) for DL model training.

S. No.	Activity Performed	Minimum Frames	Maximum Frames	Average
1	Clapping	42	106	96.71
2	**Falling**	**42**	**113**	**97.57**
3	Jumping	71	106	97.8
4	**Nothing/Sitting**	**35**	**106**	**98.02**
5	Walking	0	135	96.01
6	Waving	28	142	97.11

3.2 Preprocessing

Data is food for Machine Learning (ML) and Deep Learning (DL). Food is dependent on ingredients and cooking. If one of these is not good, the food is not tasty. Similarly, ML consists of two essential steps, i.e., preprocessing (collection of ingredients) and training (cooking). Most researchers focus on training only and leave the preprocessing behind, which is not good practice in AI and ML. This research has focused on data preprocessing followed by training and testing of the LSTM model and 1D CNN.

This preprocessing algorithm selects the best signal out of 9 signals from 3 transmitters and three receivers. The selection of the best signal is done by calculating the signal power. A signal with the highest power is considered the best signal and selected. The signal power is calculated according to equation (1).

$$P_{avg} = \frac{1}{T}\int_{0}^{T} |x(t)|^2 d(t) \tag{1}$$

P_{avg} represents the average power of the time series signal $x(t)$. It measures the signal's power content over a given time interval T. Squaring the signal is necessary because power is a non-negative quantity, ensuring that both positive and negative values contribute to the power calculation. After finding the best signal, the proposed research has focused on the absolute part of the input CSI signal to avoid the complex-valued information using equation (2). z is the output signal of real part a and complex part b. The total number of bands in each signal is 30. We have taken the average of all the bands, as stated in equation (3), to get a better single value of the input signal. A is the average of the input signal $x(t)$ starting from time 1 to n.

$$|z| = \sqrt{a^2 + b^2} \tag{2}$$

$$A = \frac{1}{n}\sum_{i=1}^{n} x(t)_i \tag{3}$$

Different lengths are used for different subjects and experiments. According to equation (4), all the other signals are equalized using a proposed padding algorithm, which paddings the values using a simple moving average of the last values.

$$SMA = \frac{1}{n}\sum_{i=1}^{n} X_i \tag{4}$$

n is the number of data points in the window of the input signal, X, The last five values. The sample pre-processed 'falling' and 'sitting/nothing' signals are depicted in Figure 3.7 and Figure 3.8, respectively. The signals are now clean, and the difference between the two activities is visible. There is a clear spike in falling activity, while sitting activity has no sharp spike. Finally, the average of all the signals of a single subject is taken, and these average values are stacked, resulting in the pre-processed file.

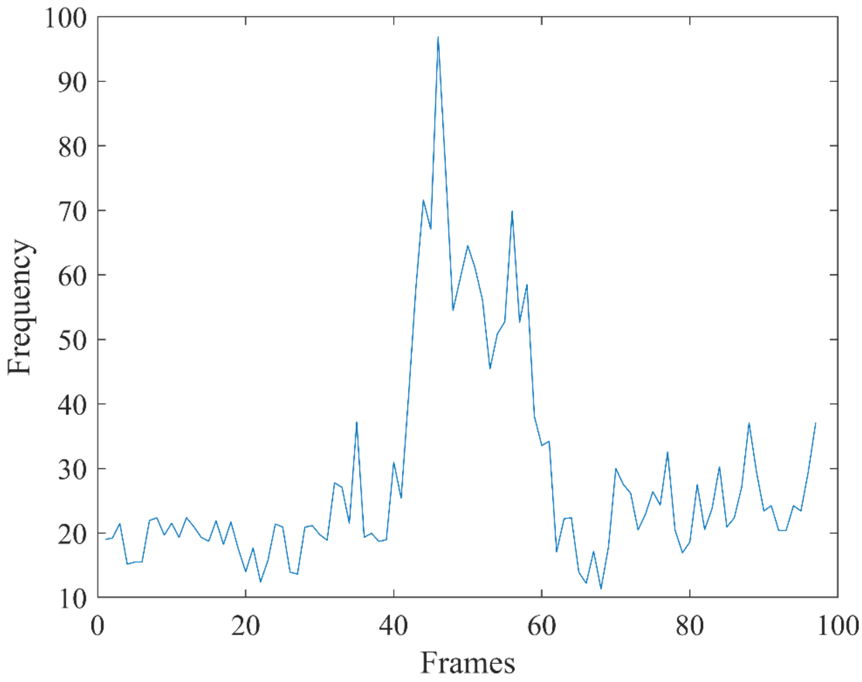

Figure 3.7. Preprocessed 'Falling' activity signal.

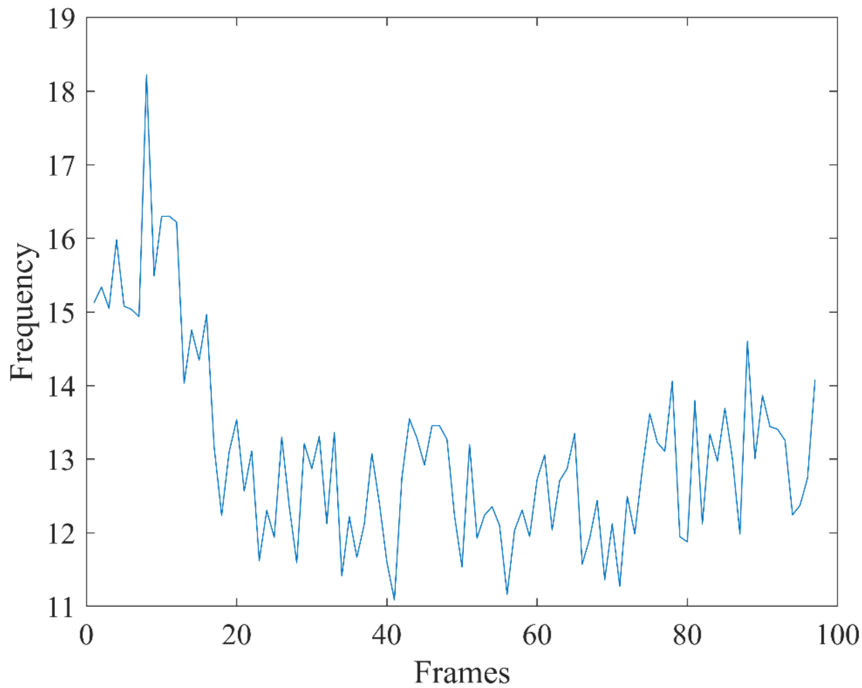

Figure 3.8. Preprocessed 'Sitting/nothing' activity signal.

This pre-processed file is provided for inputting different models, including a CNN and an LSTM model. The algorithm of the overall methodology proposed by this research is given in Figure 3.9 (Algo 1).

Algo 1
for every day in all days   *for each file in the day*   *for all the signals of every single file*   *if a file contains falling*   *then it is considered as a falling label*   *if a file contains nothing*   *then it is considered a non-falling label*   *Find best stream (having highest signal power)*   *for best stream of each frame*   *Do the padding to equalize the length*   *Calculate the magnitude of the signals*   *Stack the average of all the signals*   *end*   *end*   *end* *end*

Figure 3.9. Proposed Algorithm for Data Preprocessing

3.3 Automatic Data Labeling

After successful preprocessing, this research proposes automatic data labeling with the help of an integral aspect, the precise labeling of signal windows, as shown in Figure 3.10 (Algo 2). According to the algorithm, the proposed data labeling algorithm has three main parts: signal segmentation, variance calculation, and threshold selection.

The first step in the data labeling process involves segmenting CSI signals into manageable windows. Given the standard length of 97 frames for each signal, a comprehensive approach is taken to capture varying temporal dynamics. A sliding window technique is employed, considering windows of different sizes and offsets. Specifically, windows of 30 frames are considered, starting from the initial frame (1 to 30), followed by subsequent windows with offsets of 15 frames, i.e., 16 to 45 and 31 to 60. This approach enables the extraction of diverse temporal features, catering to different durations of activities within the signal. The overlapping factor in our proposed methodology is 50%.

Algo 2
for every signal in preprocessed signals   *for windowIndex = 1 to length(windowSizes) do*   *for startIdx = 1 to signalLength - windowSize + 1 step slideSize do*   *endIdx = startIdx + windowSize - 1*   *window = signal[startIdx:endIdx]*   *variance = calculateVariance(window)*   *up_threshold = mean(variance) + threshold * std(variance)*   *low_threshold = mean(variance) – threshold * std(variance)*   *if variance > up_threshold*   *then Fallen*   *Elseif variance < low_threshold*   *then Not-Fallen*   *end*   *end*   *end* *end*

Figure 3.10. Proposed Algorithm for Automatic Data Labeling

The variance is calculated to quantify the degree of signal fluctuation within that temporal segment. This variance computation measures the dynamic patterns present in the CSI signals. The resulting variances serve as the foundation for subsequent thresholding, contributing to the accurate classification of activities. A heuristic thresholding method is employed to facilitate the labeling process. The threshold is established based on the variance distribution across all windows, as depicted in equation (5, 6).

$$T_{up} = \mu_{var} + k \,.\, \sigma_{var} \tag{5}$$

$$T_{low} = \mu_{var} + k \,.\, \sigma_{var} \tag{6}$$

where T_{up} is the upper threshold, while T_{low} is the lower threshold. μ_{var} is the mean of variance of all signals while σ_{var} is the standard deviation of all signal's variance. The mean of the variances, coupled with a constant multiplier times the standard deviation, forms the threshold. This heuristic approach allows for adaptability, accommodating the diverse nature of activities within the CSI signals. The constant multiplier, denoted as k, is a tunable parameter, enabling researchers to fine-tune the threshold's sensitivity based on the data's specific characteristics.

The labeling decision is made by comparing the calculated variance for each window with the determined threshold. If the variance exceeds the threshold, the corresponding window indicates a falling activity; otherwise, it is labelled as a non-falling activity. This binary labeling approach provides a straightforward yet effective means of categorizing signal windows, forming the basis for subsequent machine learning techniques.

The proposed data labeling methodology, encompassing sliding window analysis and heuristic thresholding, lays the groundwork for accurate human activity recognition using CSI signals. The samples that fell and did not fall are shown in Figure 3.11 and Figure 3.12, respectively.

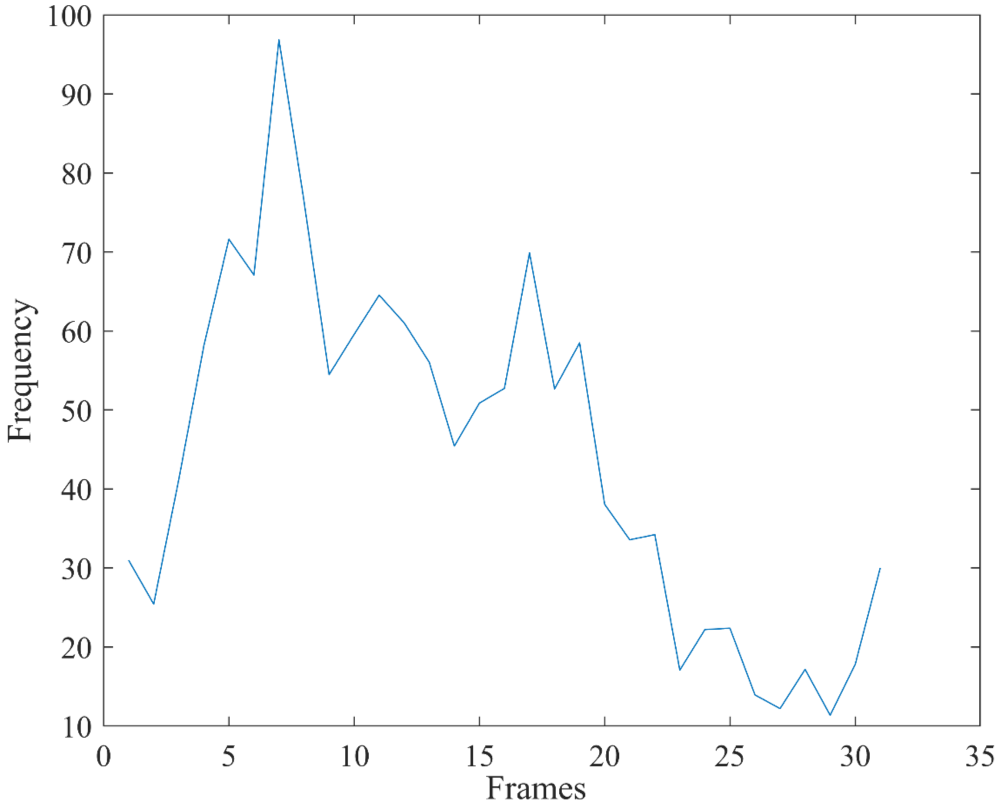

Figure 3.11. Sample Fallen signal resulted from the proposed automatic data labeling algorithm.

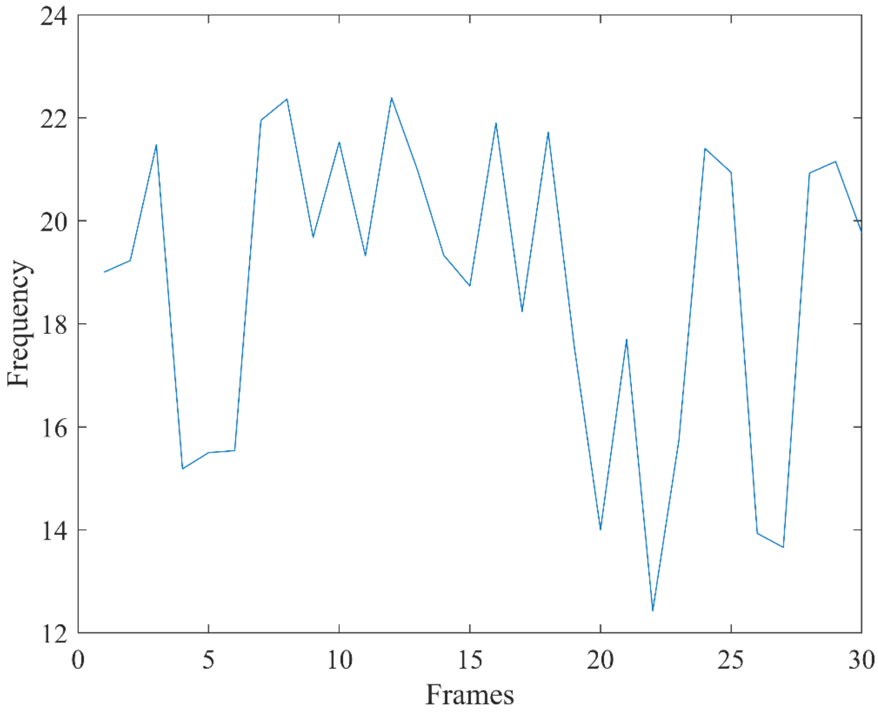

Figure 3.12. Sample non-fallen signal resulted from the proposed automatic data labeling algorithm.

After the proposed data labeling, the total number of fallen signals was 1435, while not-falling was 1495. By systematically considering different window sizes and offsets, capturing diverse temporal dynamics, and employing a flexible thresholding approach, the labelled data becomes enriched with nuanced information crucial for subsequent stages in the research. This approach ensures the adaptability and robustness of the labeling process, contributing to the overall efficacy of the human activity recognition system.

3.4 Features Selection

It was proposed that the features selected from the CSI signals be used along with the pre-processed signals. As the CSI signals are in complex numbers, the magnitude and direction from the signals were first extracted, and then equations were applied to extract features from those real numbers. Trigonometric functions can extract magnitude and direction (phase) information from complex numbers (including imaginary numbers). The complex number can be represented as equation (7).

$$z = a + ib \tag{7}$$

z is the complex number with the real part a and the imaginary part b of the complex number.

3.4.1 Magnitude (Amplitude)

The magnitude of a complex number is equivalent to the Euclidean distance of the number from the origin in the complex plane. It is calculated according to equation (2), where a is real part, and b is the imaginary part.

3.4.2 Phase (Direction)

A complex number's phase (also known as argument or angle) represents its angle with the positive real axis in the complex plane. To calculate the phase using the arctangent function, typically in radians. The phase is given by equation (8).

$$\theta = arctan\frac{a}{b} \tag{8}$$

The phase angle provides direction information and indicates how much the complex number deviates from the real axis. They allow you to analyze and understand the magnitude and direction of complex data, which can be crucial for various applications, to calculate seven different parameters for both the magnitude and direction, including minimum, maximum, mean, standard deviation, variance, skewness, and kurtosis. Each of these is explained in the following subsections.

3.4.2.1 Minimum

Minimum value helps identify the lowest data point in a set, which is valuable in various contexts.

3.4.2.2 Maximum

Maximum value helps identify the lowest data point in a set, which is valuable in various contexts.

3.4.2.3 Mean

The mean (average) measures the central tendency in the data. It gives you a typical or representative dataset value and is widely used for summarizing data. The equation for calculating the mean value is given in (3).

3.4.2.4 Standard Deviation

Standard deviation quantifies the spread or dispersion of data. It indicates how data points are scattered around the mean. A higher standard deviation suggests greater variability. It is calculated according to equation (9).

$$\sigma = \sqrt{\frac{1}{N}\sum_{1}^{n}(x_i - A)^2} \tag{9}$$

N is the number of data points, x_i is the i-th data point, A is the mean, as calculated in (3).

3.4.2.5 Variance

Variance is the square of the standard deviation and measures the average squared deviation from the mean. It is useful for understanding data variability and is a component in various statistical tests and analyses. It is calculated according to equation (10).

$$variance = \frac{1}{N}\sum_{1}^{n}(x_i - A)^2 \tag{10}$$

3.4.2.6 Skewness

Skewness measures the asymmetry of a data distribution. It can reveal whether the data is skewed to the left (negatively skewed), right (positively skewed), or symmetrically distributed. Skewness information is essential for understanding the shape of data. Skewness can be calculated as equation (11).

$$skewness = \frac{1}{N}\sum_{1}^{n}\left(\frac{x_i - A}{\sigma}\right)^3 \tag{11}$$

3.4.2.7 Kurtosis

Kurtosis helps in understanding the shape of a data distribution. It can indicate whether data is heavily tailed (leptokurtic) or flat-tailed (platykurtic) compared to a normal distribution. High kurtosis may suggest the presence of outliers or extreme values. The equation for sample kurtosis is given in (12).

$$kurtosis = \frac{1}{N}\sum_{1}^{n}\left(\frac{x_i - A}{\sigma}\right)^4 - 3 \qquad (12)$$

3.5 Selected Models

This section includes the selected models which are CNNs and LSTMs and described in detail their working flow.

3.5.1 Convolutional Neural Networks (CNNs)

Artificial Neural Networks (ANNs) have been around for long time until Convolutional Neural Network (CNN) was proposed. CNN is a very important technique primarily used for image recognition, machine vision, and other natural signals. CNN is breakthrough in deep learning held which achieved human level accuracy (Dong, 2018). This section gives a short overview of CNN before going into details of the proposed idea. Unlike ANN, CNN is divided into two parts. The rest of the part performs feature extraction from the input image while the second part acts as classifier. Inside feature extraction, input image passes through alternating combinations of convolution and pooling layers.

The core building block of CNN is convolutional layer which does most of the heavy computations. It maintains pixel relationship by learning features using learnable weight matrices called kernels or filters. The input image is convolved with the kernel and the resulting signal is called feature map. The convolutional layer in CNN learns features which are important for classification. It is important to note that correlation is different than convolution. The kernel needs to be flipped vertically and horizontally before computing the dot product. Figure 3.13 shows a simple convolutional layer of CNN. 5 x 5 image is convolved with 3 x 3 kernel. The resulting signal, feature map, size is 3 x 3. Different kernels extract different features for the input image. For example, one kernel will be used to detect the vertical edges of input image, another feature may detect horizontal, depending on the training data. There are some definitions related to convolutional layer.

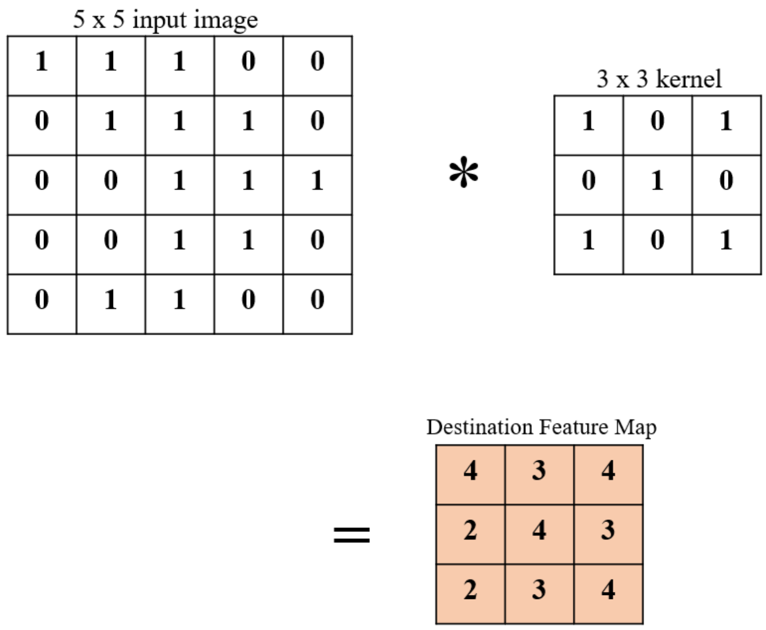

Figure 3.13. Simple Convolutional Operation of CNN

3.5.1.1 Stride

Stride is the number of pixels the kernel is slid over the input image. It is the shift in bits over the input image data. For example, if the stride is set to 1, we shift the kernel by 1 pixel over the input image matrix. When its value is set to 2, then we move the kernel by 2 bits over the source signal. It is important to note that the size of the destination feature map is dependent on the stride size.

3.5.1.2 Padding

As all images are not squared which causes fitting problem of kernel over input image. We can do add extra zero-valued pixels to the boundary of the image. This is called padding. If convolution is computed in the absence of padding, then the resulting feature map loses k/2 pixels on all the boundaries.

3.5.2 Pooling

Pooling layer is introduced after convolution layer in CNN. This layer is used for the spatial invariance to rotation, scaling, and translation. Pooling is done spatially and is also known as down-sampling. This decreases the size of each feature map without any significant loss. There are three types of spatial pooling.

a) Max Pooling
b) Average Pooling

c) Stochastic Pooling

Max Pooling is calculated by computing the largest element in the pooling region (Boureau et al., 2010). An example of Max Pooling is shown in Figure 3.14. Taking the average of all values under the pooling region is known as average pooling (Lin et al., 2013). For stochastic pooling, the selection of elements under pooling region is based on activation functions which calculates the probability of pooling region (Zeiler & Fergus, 2013).

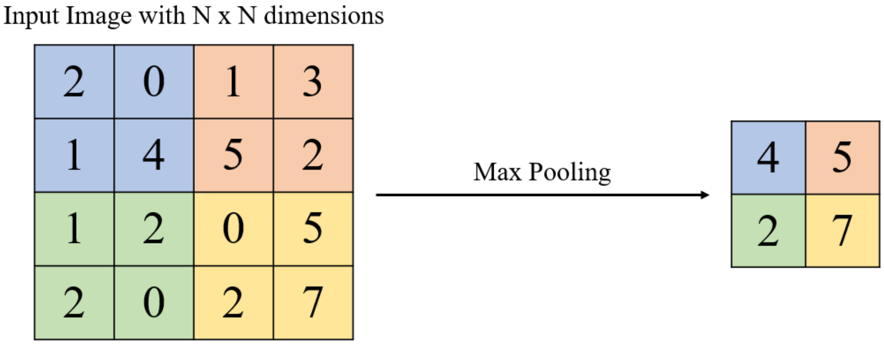

Figure 3.14. Sample Pooling Operation (Max Pooling)

3.5.3 Fully Connected Layer

FCL is just like simple ANN, which takes the output of convolutional layer or pooling layer. After processing an input image through different combination of convolutional layers and pooling layers. This resulted output from the two layers is flattened as a vector. A simple fully connected layer is shown in Figure 3.15. An ANN is inspired from the biological neural system which tries to mimic human cognition with ANNs as computational models. The brain is highly accurate on tangible and non-tangible concepts, modeling it with a machine is very difficult. ANN has many artificial neurons which are interconnected without neurons in a meshed topology. ANN can be trained in regression and classification problems. During classification, the main goal of ANN is to transform the input data into class labels.

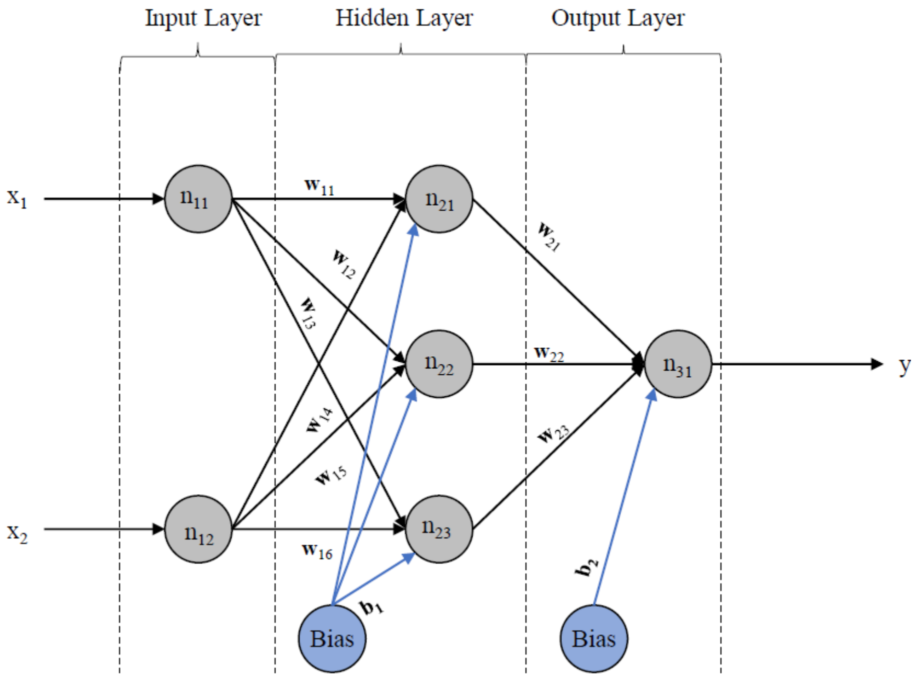

Figure 3.15. Simple FCL or ANN

The only difference between fully connected layer and convolutional layer is that convolutional layer neurons are connected to the local region at the input and many of them share parameters. However, the dot product is computed in both layers. It is important to note that both layers are interconvertible. Figure 3.16 shows a CNN having two convolutional layers, two pooling layers and a fully connected layer with one hidden layer. As depicted in the figure, Conv 1 and Conv 2 constitute the convolution layers, whereas Pool 1 and Pool 2 constitute the two pooling layers. FC 1 and FC 2 represent FCL. Output of FC 1 is input to FC 2 which is working as hidden layer. The number of output neurons of FC 1 is the number of neurons in the hidden layer.

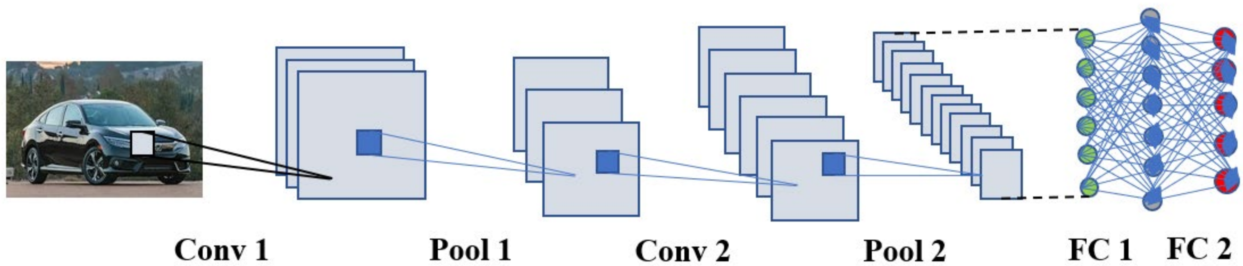

Figure 3.16. A complete CNN model.

3.5.4 Long Short-Term Memory (LSTM)

As the name implies, LSTM is designed to handle both long-term and short-term dependencies in sequential data. This is crucial for understanding and predicting patterns in signals data, where past and recent information often play significant roles. It is a type of recurrent neural network (RNN) architecture that has been particularly effective in handling sequential data, making it well-suited for analyzing and predicting signals data (Hochreiter & Schmidhuber, 1997). This research used LSTM as a deep sequential model. The model takes care of Long-Term Memory (LTM) and Short-Term Memory (STM). Four different gates are used in an LSTM unit, as depicted in Figure 3.17. These gates include:

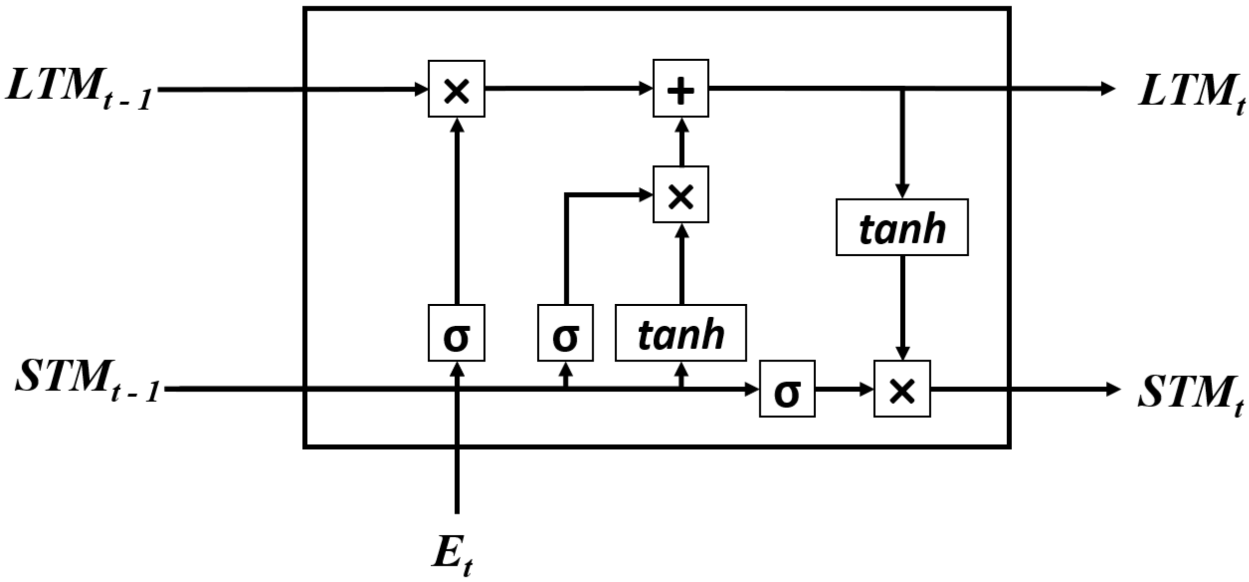

Figure 3.17. A Simple LSTM Model

a. **Forget Gate:**

The forget gate determines what information from the LTM should be retained and what should be discarded. It does this by considering the current input and the previous STM. If certain information is deemed less valid or irrelevant to the current task, the forget gate helps erase or "forget" it from long-term memory.

b. **Learn Gate:**

The learn gate combines the current input (an event) and the STM to remember the most recent information. This merging process helps the model adapt to the current event by incorporating relevant knowledge from short-term memory.

c. **Remember Gate:**

The main role of the remember gate is to preserve important information from the LTM that should not be forgotten. It works by merging the current event and the STM, effectively creating an updated version of the long-term memory that retains valuable historical context.

d. Use Gate:

The use gate is responsible for predicting the current event's output. It takes into account the STM, LTM, and the recent events. This prediction is often used to forecast future values in signal data.

3.6 ConvLSTM Model

A ConvLSTM (Convolutional Long Short-Term Memory) model is a recurrent neural network (RNN) architecture incorporating convolutional layers within the LSTM structure. It is designed to capture spatial and temporal dependencies in sequence data, making it particularly useful for tasks involving images, videos, and spatiotemporal sequences.

3.6.1 Components of a ConvLSTM Model

A detailed explanation of the components and workings of a ConvLSTM model is given in the following subsections.

3.6.1.1 Convolutional LSTM Cell

At the heart of the ConvLSTM model is the Convolutional LSTM cell. This cell combines the properties of convolutional layers and LSTM units, enabling the model to learn hierarchical representations across both spatial and temporal dimensions simultaneously. A standard LSTM cell has three gates: the input gate, the forget gate, and the output gate. In a ConvLSTM cell, these gates operate on convolutional operations.

3.6.1.2 Convolutional Operations

Convolutional operations are applied to the input data and hidden states within the LSTM cell. These operations help capture spatial patterns and relationships in the input sequences. Convolutional operations involve filters (kernels) that slide over the input, learning spatial features.

3.6.1.3 Memory Cell

Like traditional LSTM models, ConvLSTM cells have memory cells that retain information over long sequences. This enables the model to capture long-term dependencies in the input data.

3.6.1.4 Hidden State

The hidden state in a ConvLSTM cell is updated based on the input data, the previous hidden state, and the memory cell. The hidden state retains information about the current state of the model.

3.6.1.5 Return Sequences

ConvLSTM layers can be configured to return sequences, meaning the output is generated for each time step in the input sequence. This is useful for tasks involving sequence-to-sequence predictions.

3.6.2 Advantages of ConvLSTM Models

The reasons behind selecting the ConvLSTM model are discussed in the following subsections.

Spatial-Temporal Representations

ConvLSTM models effectively capture spatial and temporal dependencies in sequential data. This is crucial for tasks such as video analysis, where patterns evolve over time and space.

Reduced Information Loss

Combining convolutional operations and LSTM memory cells helps reduce information loss over long sequences, making ConvLSTM models well-suited for tasks requiring memory of past events.

Parameter Sharing

Parameter sharing through convolutional operations allows ConvLSTM models to learn spatial hierarchies efficiently, reducing the number of parameters compared to fully connected architectures.

In this model, we have merged initial layers from CNN followed by LSTM units to capture the spatial and temporal information from the pre-processed CSI signals.

Chapter 4

Results and Discussion

This chapter introduces the experimental setup and performance metrics used by this thesis before discussing the results in depth. The details are given in each respective section.

4.1 Environmental Setup

The proposed methodology was implemented on a Windows 10 operating system. The experiments were conducted on a core i7 desktop system with 16 GB of RAM, which provided sufficient computational resources for efficient model execution and testing. The models, including CNN, LSTM, and ConvLSTM, were developed and executed in MATLAB. MATLAB's robust suite of built-in functions and compatibility with deep learning frameworks were leveraged for data processing, model training, testing, and performance evaluation.

MATLAB's Deep Learning Toolbox was used to define the architectures, optimize hyperparameters, and perform iterative training of the models. The training was carried out on pre-processed datasets with multiple iterations to improve model performance and reduce overfitting. Standard libraries and visualization tools within MATLAB helped assess the training progress, generate diagnostic plots, and analyze results.

4.2 Performance Measures

The effectiveness of the models was evaluated using four key performance metrics. The performance measures used for evaluation and testing purposes include accuracy, f1-score, precision, and recall presented in 1, 2, 3, and 4, respectively, in terms of True Positive (TP), False Positive (FP), False Negative (FN), and True Negative (TN).

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \tag{1}$$

$$F1 - Score = \frac{2 * Precision * Recall}{Precision + Recall} \tag{2}$$

$$Precision = \frac{TP}{TP + FP} \tag{3}$$

$$Recall = \frac{TP}{TP + FN} \tag{4}$$

In this thesis, two different types of experiments are conducted: activity detection with data labeling and without data labeling. The results are shown and discussed in the following subsections of this chapter.

4.3 Results without data labeling

Following are the results achieved without performing the data labeling task. In the next phase, I will implement the data labeling algorithm, share the data labeling results, and compare the results with and without data labeling.

4.3.1 Training/Validation of CNN

Figure 4.1 shows the CNN model's training and validation graphs trained on the pre-processed CSI data. This shows that the model was not stable during training. The CNN model's best validation accuracy is 85.33%. The Confusion Matrix (CM) 4.1 shows 123 correctly identified falls and 129 correctly identified non-falls, with 27 falls misclassified as non-falls and 21 non-falls misclassified as falls. The model achieved an accuracy of 84.00%, precision of 82.00%, and recall of 85.42%, resulting in an F1 score of 83.67%. This indicates that while the CNN performs reasonably well, it struggles with misclassifying some fall events, potentially due to its limited temporal feature extraction capabilities.

CM 4.1. Confusion Matrix for CNN before data labeling.

	Falling	**Non-Falling**
Falling	123	27
Non-Falling	21	129

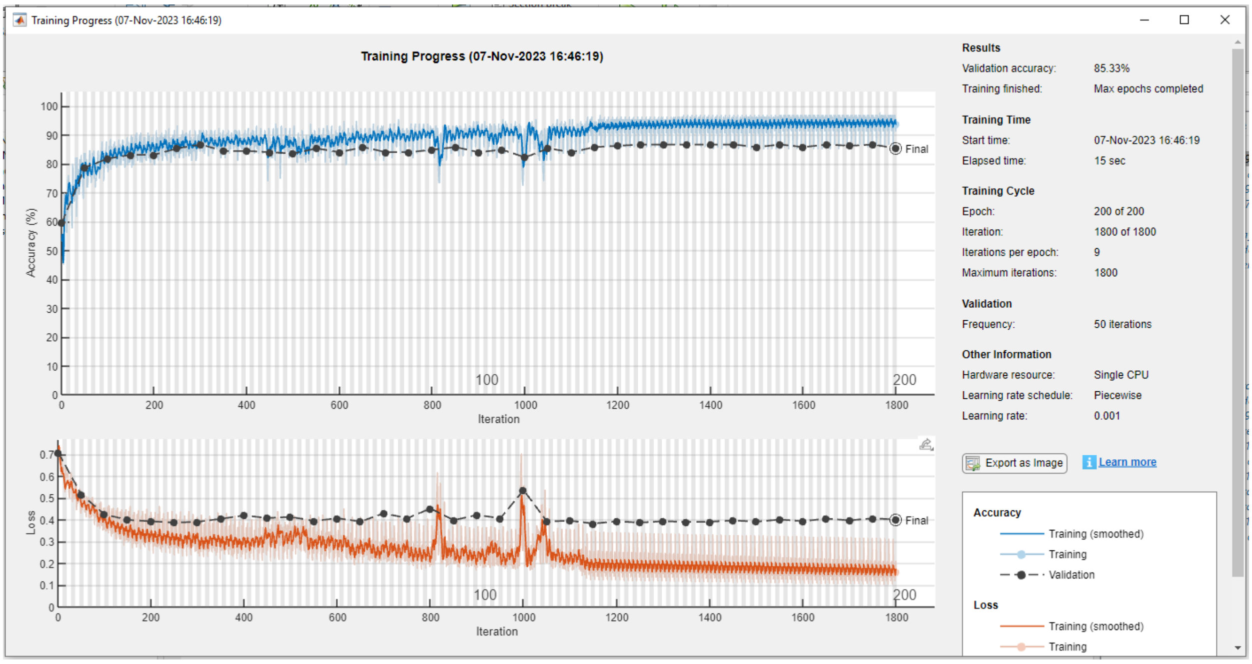

Figure 4.1. Training Validation graph of CNN model

4.3.2 Training/Validation of LSTM

Similarly, Figure 4.2 shows the training and validation graphs of the LSTM model trained on the pre-processed CSI data. It shows that the model is not stable while training. The best validation accuracy achieved by the LSTM model is 87.12%. The LSTM model correctly identified 124 falls and 135 non-falls, with 26 false positives and 15 false negatives, as shown in CM 4. It achieved an accuracy of 86.33%, precision of 82.67%, and recall of 89.21%, leading to an F1 score of 85.81%. The model demonstrates better recall than CNN, likely due to its superior handling of time-series data, although it still misclassifies a moderate number of falls as non-falls. The LSTM model is better than the CNN model because LSTM is more suitable for time series or signal data.

CM 4.2. Confusion Matrix for LSTM before data labeling.

	Falling	**Non-Falling**
Falling	124	26
Non-Falling	15	135

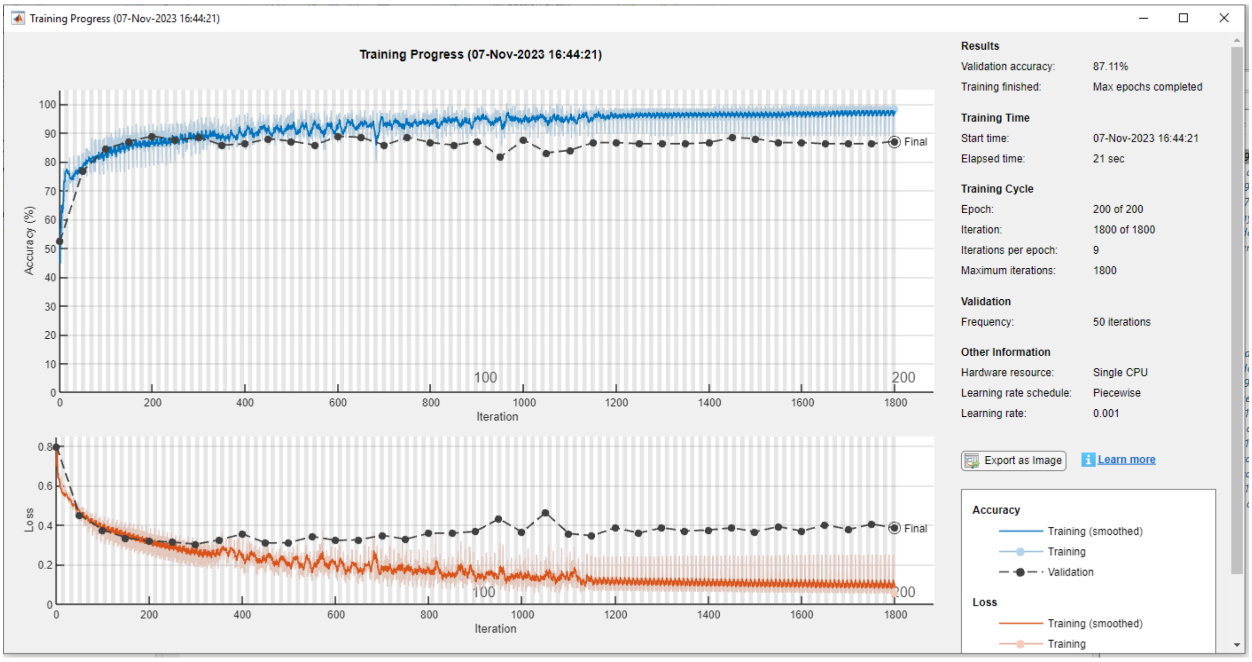

Figure 4.2. Training Validation graph of the LSTM model

4.3.3 Training/Validation of Conv-LSTM

The Conv-LSTM model combines CNN layers with LSTM models. Such models can detect spatial and temporal information from input data. Figure 4.3 shows the training and validation graphs of the Conv-LSTM model trained on the preprocessed CSI data. It shows that the model is more stable than CNN and LSTM. The best validation accuracy achieved by the LSTM model is 89.34%. The Conv-LSTM model is a better option than both the CNN and LSTM models as it fetches spatial and temporal information at the same time.

CM 4.3. Confusion Matrix for Conv-LSTM before data labeling.

	Falling	**Non-Falling**
Falling	125	25
Non-Falling	8	142

As shown in CM 4.3, the Conv-LSTM model correctly identified 125 falls and 142 non-falls, with only 25 false positives and 8 false negatives. It achieved an accuracy of 89.00%, precision of 83.33%, and recall of 93.98%, yielding an F1 score of 88.34%. This model outperforms both CNN and LSTM models by capturing spatial and temporal features, resulting in fewer misclassifications and a strong balance between precision and recall.

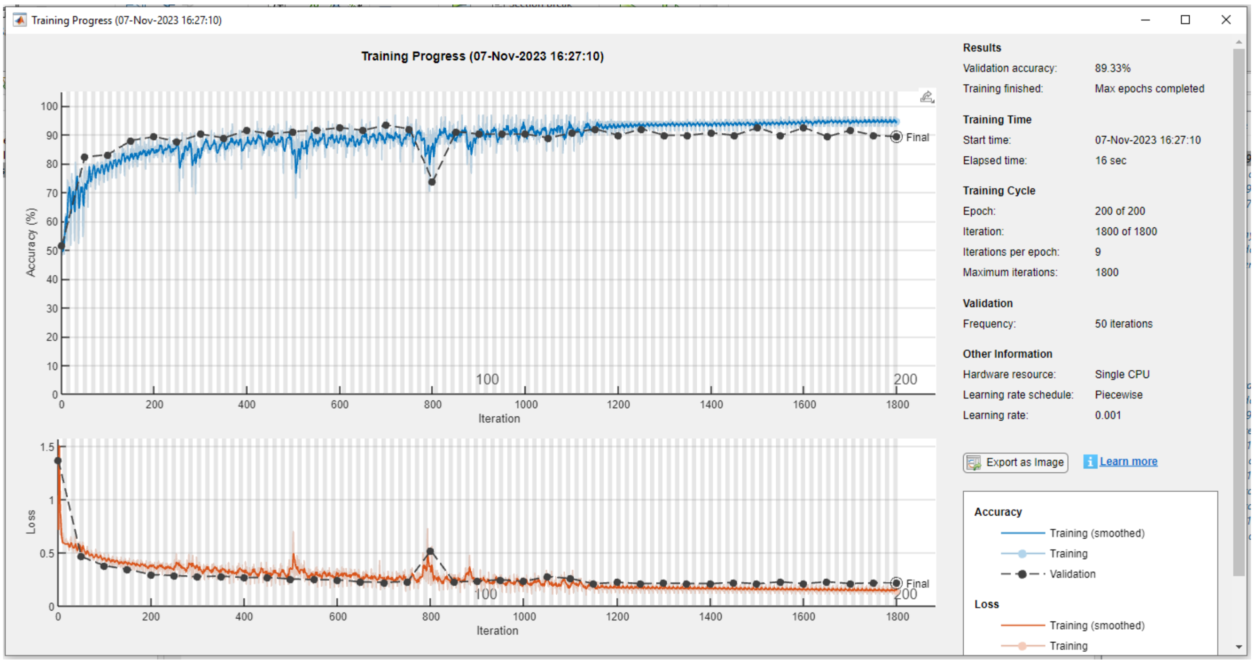

Figure 4.3. Training Validation graph of Conv-LSTM model

4.4 Results with Proposed Automatic Data Labelling

Following are the results achieved by performing the data labeling task. In the next phase, I will implement the data labeling algorithm, share the data labeling results, and compare the results with and without data labeling.

4.4.1 Training/Validation of CNN

The CNN model was trained solely on pre-processed CSI data without data labeling. Figure 4.4 illustrates the CNN model's training and validation graphs trained on the proposed data labeling data. The model exhibits noteworthy performance, with a validation accuracy of ~93%.

CM 4.4. Confusion Matrix for CNN with labeling

	Falling	**Non-Falling**
Falling	442	58
Non-Falling	22	478

Moreover, CM 4.4 shows that the CNN model correctly identified 442 falls and 478 non-falls, with 58 falls misclassified as non-falls and 22 non-falls misclassified as falls. The model achieved an accuracy of 92.00%, a precision of 88.40%, a recall of 95.26%, and an F1 score of 91.70%. This indicates that the data labeling method improved performance significantly, though a small number of fall events are still being misclassified.

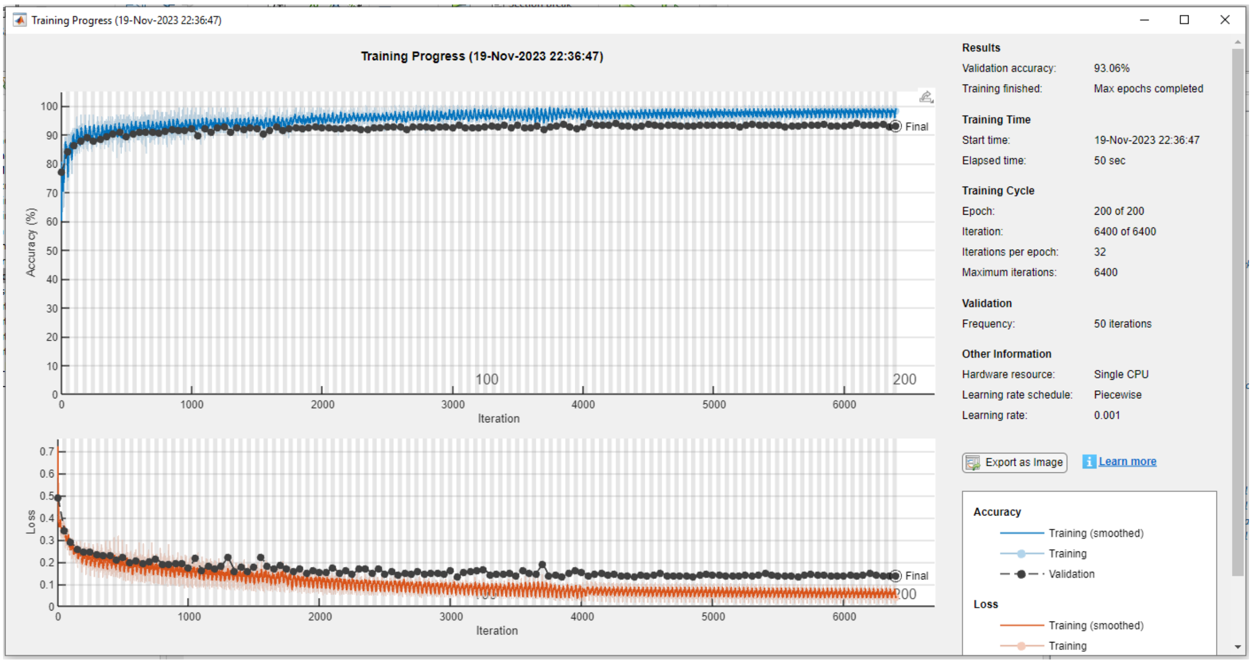

Figure 4.4. Training Validation graph of CNN model on a proposed labelled dataset

4.4.2 Training/Validation of LSTM

Similarly, without labeled data, the LSTM model underwent training on pre-processed CSI data. Figure 4.5 shows the training and validation graphs of the LSTM model with data labeling. Even without explicit labels, the model demonstrates robust training behaviour. The LSTM model achieves a remarkable validation accuracy of ~94%, outperforming the CNN model.

CM 4.5. Confusion Matrix for LSTM with data labeling.

	Falling	**Non-Falling**
Falling	461	39
Non-Falling	20	480

As depicted in CM 4.5, the LSTM model correctly identified 461 falls and 480 non-falls, with 39 falls misclassified as non-falls and 20 non-falls misclassified as falls. It achieved an accuracy of 94.10%, precision of 92.20%, recall of 95.84%, and an F1 score of 93.99%. These results demonstrate that the LSTM model benefited from data labeling, with both high recall and precision, making it more reliable in detecting falls than the CNN model.

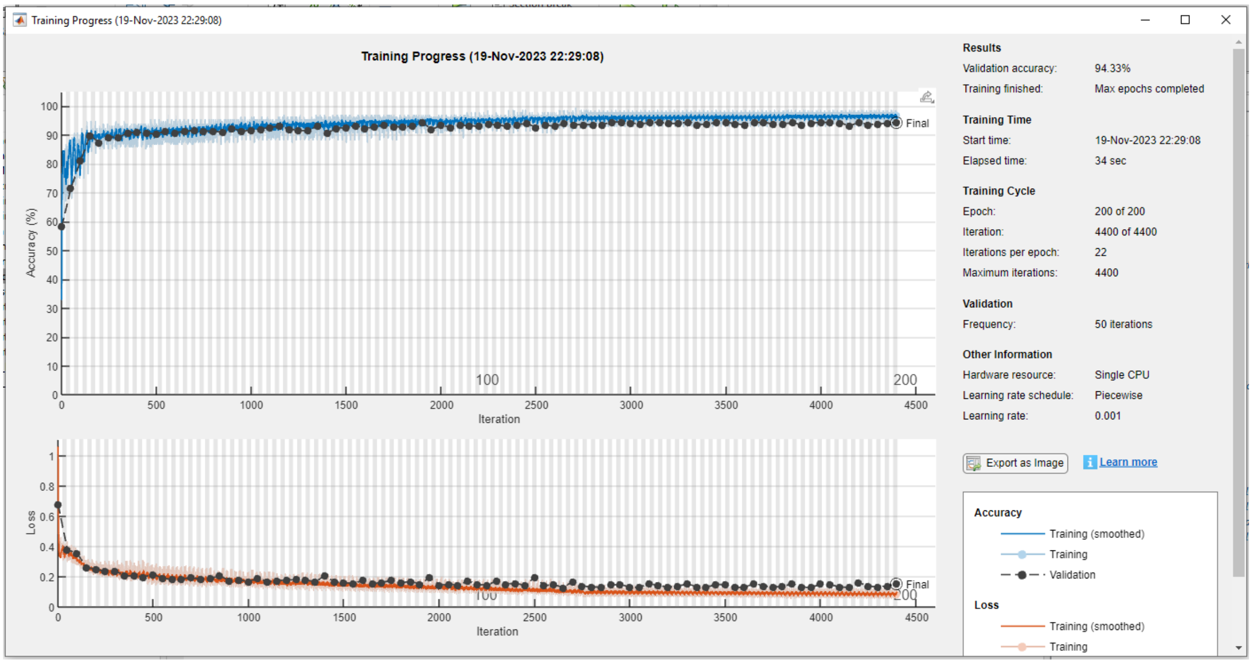

Figure 4.5. Training Validation graph of the LSTM model on a proposed labeled dataset

4.4.3 Training/Validation of Conv-LSTM

The ConvLSTM model, a fusion of CNN layers and LSTM models was trained without labelled data. This model, capable of capturing spatial and temporal information from input data, exhibited superior stability during training. Figure 4.6 highlights the training and validation graphs of the ConvLSTM model with the proposed data labeling. The model achieved exceptional results with a validation accuracy of ~97%.

The Conv-LSTM model accurately classified 487 falls and 487 non-falls, with only 13 falls and 13 non-falls misclassified, as shown in CM 4.6. It achieved an accuracy, precision, recall, and an F1 score of 97.40%. The model shows near-perfect performance, with minimal misclassifications, demonstrating its strong capability to capture both spatial and temporal patterns effectively after data labeling. The ConvLSTM model proves to be a robust choice, surpassing both CNN and LSTM models even without explicit data labels. Its ability to extract spatial and temporal information simultaneously makes it a compelling option.

CM 4.6. Confusion Matrix for Conv-LSTM with data labeling.

	Falling	Non-Falling
Falling	487	13
Non-Falling	13	487

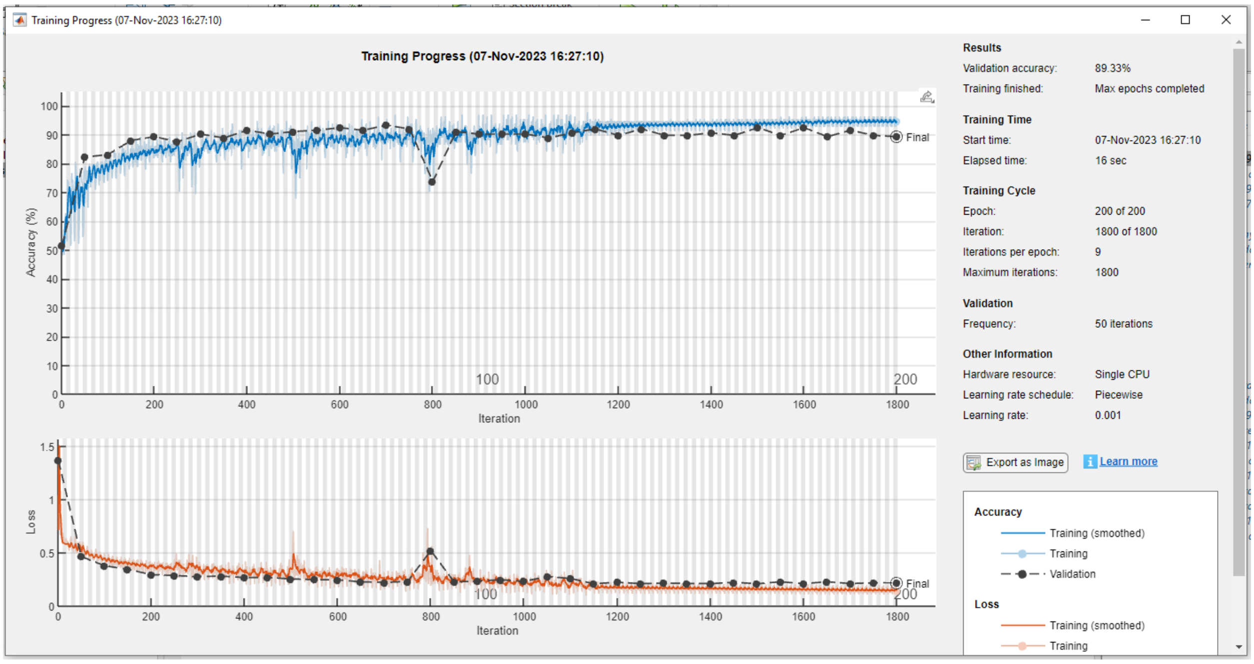

Figure 4.6. Training Validation graph of Conv-LSTM model on a proposed labelled dataset

In the absence of labelled data, our experiments with three distinct models—CNN, LSTM, and ConvLSTM—on two different datasets provide valuable insights into the performance and adaptability of these architectures. For the dataset without data labeling, we observed that the ConvLSTM model outshines both the CNN and LSTM models, demonstrating enhanced stability and achieving the highest testing accuracy of 90.16%. The LSTM model, tailored for time-series data, outperformed the CNN model. However, when the proposed data labeling was performed, the models exhibited remarkable resilience. The CNN model achieved a notable testing accuracy of ~92%, while the LSTM and ConvLSTM models demonstrated even more remarkable performances with ~94% and ~98%, respectively. These results suggest that our proposed data labeling method is very effective because we have removed the extra frames from raw data. Moreover, we have labelled the data using some heuristics.

4.5 Comparative Analysis

Table 4.1 gives the complete comparative analysis of the proposed data labeling with the data without performing data labeling.

Table 4.1. Comparative analysis between data with and without labeling.

Method	Model	Accuracy	Precision	Recall	F1 Score
		In Percent (%)			
Without Data Labeling	CNN	84.52	82	85.42	83.67
	LSTM	86.87	82.67	89.21	85.81
	Conv-LSTM	89.34	83.33	93.98	88.34
With Data Labeling	CNN	92.17	88.4	9.26	91.7
	LSTM	94.21	92.2	95.84	93.99
	Conv-LSTM	97.4	97.4	97.4	97.4

Without data labeling, the CNN model achieved an accuracy of 84.52%, a precision of 82.00%, a recall of 85.42%, and an F1 score of 83.67%. After applying the proposed data labeling method, the performance improved significantly, with an accuracy of 92.17%, precision of 88.40%, recall of 95.26%, and an F1 score of 91.70%. This indicates that the data labeling method helped the CNN model reduce misclassifications, particularly for falls.

Before data labeling, the LSTM model achieved an accuracy of 86.87%, precision of 82.67%, recall of 89.21%, and an F1 score of 85.81%. After data labeling, the performance metrics rose to an accuracy of 94.21%, precision of 92.20%, recall of 95.84%, and an F1 score of 93.99%. The improvements highlight that LSTM is well-suited for time-series data and can benefit greatly from better-structured labeled data.

Initially, the Conv-LSTM model achieved an accuracy of 89.34%, precision of 83.33%, recall of 93.98%, and an F1 score of 88.34%. Data labeling improved its performance to an impressive accuracy of 97.59%, precision of 97.40%, recall of 97.40%, and an F1 score of 97.40%. The model's near-perfect performance suggests that its ability to handle both spatial and temporal information is enhanced when data is pre-processed and labeled correctly.

To the best of our knowledge, no related literature can be found that can be used for a fair comparison. That is why only comparisons are made with and without data labelling.

4.6 Discussion

The results presented in this thesis underscore the significant impact of data preprocessing and labeling on the performance of machine learning models, specifically in the context of activity detection using Channel State Information (CSI) data. The exploration of three distinct models—Convolutional Neural Networks (CNN), Long Short-Term Memory (LSTM) networks, and a hybrid Convolutional LSTM (Conv-LSTM) model—provides a comprehensive understanding of how each model responds to the challenges of unlabeled and labelled data sets.

The CNN model, traditionally known for its prowess in handling spatial data, exhibited improved performance when trained on data enhanced with the proposed automatic data labeling technique, achieving testing accuracies of approximately 92%. This improvement underscores the value of precise, contextually relevant labeling in enhancing the model's ability to discern and classify complex patterns in CSI data.

The LSTM model, with its inherent ability to process time-series data, demonstrated a notable advantage in handling sequential information inherent in CSI data. This was evidenced by its superior performance over the CNN model in the unlabeled data scenario and its remarkable testing accuracy of approximately 94% with labelled data. The LSTM model's success highlights the critical role of temporal information in accurately detecting activities from CSI signals.

The Conv-LSTM model, a sophisticated amalgamation of CNN and LSTM architectures, was designed to leverage both spatial and temporal data characteristics. This model outperformed the standalone CNN and LSTM models in all test scenarios, reaching an impressive testing accuracy of approximately 98% with labelled data. The Conv-LSTM model's superior performance can be attributed to its ability to concurrently process spatial and temporal dimensions of the CSI data, providing a more nuanced understanding of the underlying activities.

The disparity in performance between models trained on automatically labelled datasets versus the original datasets reinforces the hypothesis that the quality of data preprocessing and labelling directly influences the efficacy of deep learning models. The proposed data labeling method, which meticulously removes extraneous frames and applies heuristic-based labels, has proven particularly effective. This also helps enhance the models' ability to interpret the data and significantly reduces the complexity and noise within the data set, leading to more accurate and reliable activity detection.

4.7 Implications

The findings of this study have several broader implications for the development of real-world, device-free sensing systems in healthcare:

4.7.1 Real-World Applications

The proposed system holds significant potential for real-world applications, particularly in elderly care facilities. Real-time fall detection can provide early intervention opportunities, reducing the severity of fall-related injuries and improving the overall quality of life for older adults. Additionally, the system can be extended to other applications, such as monitoring mobility, daily activities, and rehabilitation progress.

4.7.2 Cost-Effectiveness

The system's use of existing wireless infrastructure, such as Wi-Fi networks, makes it highly cost-effective. Unlike vision-based solutions that require expensive hardware, this device-free approach leverages CSI data, minimizing deployment costs while providing robust performance in challenging environments like low-light or smoke-filled areas.

4.7.3 Ease of Deployment

The architecture is designed to be hardware-agnostic, enabling deployment on various platforms, including edge devices, servers, and embedded systems. Since the system is device-free, users do not need to wear or carry devices, simplifying the deployment process in clinical and home settings.

4.7.4 Integration with Healthcare Systems

The system can seamlessly integrate with electronic health records (EHRs) and hospital monitoring systems, enhancing patient safety through continuous monitoring and data-driven decision-making. The collected data can also support long-term trend analysis, helping healthcare professionals make informed decisions regarding patient care and fall prevention strategies.

These broader implications highlight the system's value as a reliable, privacy-preserving, and scalable fall detection and activity monitoring solution in various healthcare environments.

Chapter 5
Conclusions and Future Directions

5.1 Conclusions

This research illustrates the profound influence of data quality, manifested through preprocessing and labeling, on the performance of machine learning models in activity detection applications using Channel State Information (CSI) data. The findings reveal several critical insights:

5.1.1 Model Suitability

Different models exhibit varying degrees of effectiveness in deciphering the intricacies of CSI data. The Conv-LSTM model demonstrated the highest overall performance due to its dual capability to capture spatial and temporal information. This model outperformed both the CNN and LSTM models in all test scenarios, reaching a testing accuracy of approximately 98% with labelled data.

5.1.2 Importance of Data Labeling

The implementation of a sophisticated data labeling process markedly enhances model performance. This research showed that accurate and contextually relevant data labeling significantly improves model accuracy. For instance, the CNN model's performance increased from 84.52% testing accuracy without labeling to around 92% with labeling, highlighting the critical role of precise data labeling in machine learning.

5.1.3 Model Adaptability

The significant improvement in model accuracy with labelled data underscores the adaptability of these models to more refined datasets. The LSTM model, particularly suited for time-series data, demonstrated robust training behavior and improved testing accuracy from 86.87% without labeling to approximately 94% with labeling. This adaptability showcases the potential of these models in real-world applications where the precision of activity detection is paramount.

5.1.4 Impact of Combined Architectures

The Conv-LSTM model, which integrates the strengths of both CNN and LSTM, proved the most effective. Its ability to concurrently process spatial and temporal dimensions of CSI data provided a more nuanced understanding of the underlying activities. This finding highlights the potential of hybrid architectures in enhancing the performance of machine learning models for complex data types.

Overall, this study provides a foundational understanding of the impact of data quality on machine learning models for activity detection using CSI data. The results emphasize the need for sophisticated data preprocessing and labeling techniques to maximize model performance.

5.2 Future Directions

Given the promising results of this study, future research could explore several avenues to further enhance the effectiveness and applicability of these models:

5.2.1 Advanced Labeling Techniques

Investigating more sophisticated data labeling techniques, such as semi-supervised or unsupervised learning methods, to further automate the labeling process. These approaches could leverage the vast amount of unlabelled data available, reducing the need for manual intervention and potentially improving model accuracy.

5.2.2 Real-time Processing

Developing methods for real-time data preprocessing and labeling to enable live activity detection. Real-time capabilities would expand the applicability of these models in dynamic environments, such as smart homes and healthcare facilities, where timely detection of activities is crucial.

5.2.3 Model Optimization

Exploring deeper and more complex model architectures or novel combinations of existing models to further improve accuracy and efficiency. Techniques such as transfer learning, ensemble learning, and reinforcement learning could be investigated to enhance model performance in processing CSI data for activity detection.

5.2.4 Cross-domain Applicability

Assessing the effectiveness of the proposed models and labeling techniques across different domains, such as healthcare monitoring, security surveillance, and industrial automation. Evaluating the versatility and adaptability of these models in varied applications would provide insights into their broader utility.

5.2.5 Energy Efficiency

Considering the energy efficiency of these models, especially in battery-powered or resource-constrained devices, to ensure their practical deployment in real-world scenarios. Research could focus on optimizing model architectures to reduce computational complexity and power consumption without compromising accuracy.

5.2.6 Integration with IoT Systems

Investigating the integration of these models with Internet of Things (IoT) systems to create comprehensive monitoring solutions. This could involve developing lightweight models that run on edge devices, facilitating real-time data processing and activity detection directly at the source.

5.2.7 Enhanced Data Preprocessing

Exploring advanced data preprocessing techniques, such as denoising and feature extraction, to further improve the quality of input data. High-quality preprocessing can enhance the ability of models to detect subtle and complex patterns in CSI data, leading to more accurate activity detection.

This thesis lays a foundational understanding of the impact of data quality on machine learning models for activity detection using CSI data. The insights gained from this research offer a pathway for future studies to build upon these findings and explore new frontiers in this rapidly evolving field, ultimately contributing to developing more effective and efficient activity detection systems.

References

Abbate, S., Avvenuti, M., Bonatesta, F., Cola, G., Corsini, P., & Vecchio, A. (2012). A smartphone-based fall detection system. *Pervasive and Mobile Computing*, *8*(6), 883-899.

Abuhoureyah, F., Wong, Y. C., Isira, A. S. B. M., & Al-Andoli, M. N. (2023). CSI-Based Location Independent Human Activity Recognition Using Deep Learning. *Human-Centric Intelligent Systems*, *3*(4), 537-557.

Agjei, E., & Orangeh, G. (2023). *E-Health Services for Elderly People: accessibility and usability* https://www.theseus.fi/handle/10024/807573

Alam, F., Faulkner, N., & Parr, B. (2020). Device-free localization: A review of non-RF techniques for unobtrusive indoor positioning. *IEEE Internet of Things Journal*, *8*(6), 4228-4249.

Alazeb, A., Azmat, U., Al Mudawi, N., Alshahrani, A., Alotaibi, S. S., Almujally, N. A., & Jalal, A. (2023). Intelligent Localization and Deep Human Activity Recognition through IoT Devices. *Sensors*, *23*(17), 7363. https://www.mdpi.com/1424-8220/23/17/7363

Alazrai, R., Hababeh, M., Baha'A, A., Ali, M. Z., & Daoud, M. I. (2020). An end-to-end deep learning framework for recognizing human-to-human interactions using Wi-Fi signals. *IEEE Access*, *8*, 197695-197710.

Bagala, F., Becker, C., Cappello, A., Chiari, L., Aminian, K., Hausdorff, J. M., Zijlstra, W., & Klenk, J. (2012). Evaluation of accelerometer-based fall detection algorithms on real-world falls. *PloS one*, *7*(5), e37062.

Beswick, A. D., Gooberman-Hill, R., Smith, A., Wylde, V., & Ebrahim, S. (2010). Maintaining independence in older people. *Reviews in Clinical Gerontology*, *20*(2), 128-153.

Bibbò, L., Carotenuto, R., & Della Corte, F. (2022). An overview of indoor localization system for human activity recognition (HAR) in healthcare. *Sensors*, *22*(21), 8119.

Boureau, Y.-L., Ponce, J., & LeCun, Y. (2010). A theoretical analysis of feature pooling in visual recognition. Proceedings of the 27th international conference on machine learning (ICML-10),

Bourke, A. K., O'Brien, J. V., & Lyons, G. M. (2007). Evaluation of a threshold-based tri-axial accelerometer fall detection algorithm. *Gait Posture*, *26*(2), 194-199. https://doi.org/10.1016/j.gaitpost.2006.09.012

Brinke, J. K., & Meratnia, N. (2019). Dataset: Channel state information for different activities, participants and days. Proceedings of the 2nd workshop on data acquisition to analysis,

Burns, E. R., Stevens, J. A., & Lee, R. (2016). The direct costs of fatal and non-fatal falls among older adults—United States. *Journal of safety research*, *58*, 99-103.

Carretero, S. (2015). Mapping of effective technology-based services for independent living for older people at home. *Seville: Joint Research Center, Institute for Prospective Technological Studies, JRC Scientific and Technical Reports Series. Zugriff am*, *15*, 2021.

Casilari, E., Álvarez-Marco, M., & García-Lagos, F. (2020). A study of the use of gyroscope measurements in wearable fall detection systems. *Symmetry*, *12*(4), 649.

Chen, F., Li, S., Han, J., Ren, F., & Yang, Z. (2024). Review of lightweight deep convolutional neural networks. *Archives of Computational Methods in Engineering*, *31*(4), 1915-1937.

Chen, S., Yang, W., Xu, Y., Geng, Y., Xin, B., & Huang, L. (2022). AFall: Wi-Fi-based device-free fall detection system using spatial angle of arrival. *IEEE Transactions on Mobile Computing*, *22*(8), 4471-4484.

Chen, W., Yu, C., Tu, C., Lyu, Z., Tang, J., Ou, S., Fu, Y., & Xue, Z. (2020). A survey on hand pose estimation with wearable sensors and computer-vision-based methods. *Sensors*, *20*(4), 1074.

Chen, Y., Zheng, B., Zhang, Z., Wang, Q., Shen, C., & Zhang, Q. (2020). Deep learning on mobile and embedded devices: State-of-the-art, challenges, and future directions. *ACM Computing Surveys (CSUR)*, *53*(4), 1-37.

Cheng, H., Zhang, J., Gao, Y., & Hei, X. (2019). Deep learning Wi-fi channel state information for fall detection. 2019 IEEE International Conference on Consumer Electronics-Taiwan (ICCE-TW),

Chowdhury, T. Z. (2018). *Using Wi-Fi channel state information (CSI) for human activity recognition and fall detection* University of British Columbia].

Chu, Y., Cumanan, K., Sankarpandi, S. K., Smith, S., & Dobre, O. A. (2023). Deep learning-based fall detection using WiFi channel state information. *IEEE Access*, *11*, 83763-83780.

Damodaran, N., Haruni, E., Kokhkharova, M., & Schäfer, J. (2020). Device free human activity and fall recognition using WiFi channel state information (CSI). *CCF Transactions on Pervasive Computing and Interaction*, *2*, 1-17.

Dang, L. M., Min, K., Wang, H., Piran, M. J., Lee, C. H., & Moon, H. (2020). Sensor-based and vision-based human activity recognition: A comprehensive survey. *Pattern Recognition*, *108*, 107561.

Delahoz, Y. S., & Labrador, M. A. (2014). Survey on fall detection and fall prevention using wearable and external sensors. *Sensors*, *14*(10), 19806-19842.

Ding, J., & Wang, Y. (2020). A WiFi-based smart home fall detection system using recurrent neural network. *IEEE Transactions on Consumer Electronics*, *66*(4), 308-317.

Dong, M. (2018). Convolutional neural network achieves human-level accuracy in music genre classification. *arXiv preprint arXiv:1802.09697*.

Falls, WHO - Fact sheet. (2021). https://www.who.int/news-room/fact-sheets/detail/falls

Fleming, J., & Brayne, C. (2008). Inability to get up after falling, subsequent time on floor, and summoning help: prospective cohort study in people over 90. *Bmj*, *337*, a2227. https://doi.org/10.1136/bmj.a2227

Guo, W., Yamagishi, S., & Jing, L. (2024). Human Activity Recognition via Wi-Fi and Inertial Sensors with Machine Learning. *IEEE Access*.

Gutiérrez, J., Rodríguez, V., & Martin, S. (2021). Comprehensive review of vision-based fall detection systems. *Sensors*, *21*(3), 947.

Habib, M. A., Mohktar, M. S., Kamaruzzaman, S. B., Lim, K. S., Pin, T. M., & Ibrahim, F. (2014). Smartphone-based solutions for fall detection and prevention: challenges and open issues. *Sensors*, *14*(4), 7181-7208.

Halperin, D., Hu, W., Sheth, A., & Wetherall, D. (2011). Tool release: Gathering 802.11 n traces with channel state information. *ACM SIGCOMM computer communication review*, *41*(1), 53-53.

He, W., Wu, K., Zou, Y., & Ming, Z. (2015). WiG: WiFi-based gesture recognition system. 2015 24th International Conference on Computer Communication and Networks (ICCCN),

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, *9*(8), 1735-1780.

Hu, Y., Zhang, F., Wu, C., Wang, B., & Liu, K. R. (2021). DeFall: Environment-independent passive fall detection using WiFi. *IEEE Internet of Things Journal*, *9*(11), 8515-8530.

Huang, T., Wang, S., Li, S., Liu, G., & Liu, J. (2021). Wiargan: A wifi-based action recognition method with gans. 2021 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/IOP/SCI),

Hur, T., Bang, J., Huynh-The, T., Lee, J., Kim, J.-I., & Lee, S. (2018). Iss2Image: A novel signal-encoding technique for CNN-based human activity recognition. *Sensors*, *18*(11), 3910.

Hussain, F., Hussain, F., Ehatisham-ul-Haq, M., & Azam, M. A. (2019). Activity-aware fall detection and recognition based on wearable sensors. *IEEE Sensors Journal*, *19*(12), 4528-4536.

Igual, R., Medrano, C., & Plaza, I. (2013). Challenges, issues and trends in fall detection systems. *Biomedical engineering online*, *12*(1), 66.

Islam, M. M., Tayan, O., Islam, M. R., Islam, M. S., Nooruddin, S., Kabir, M. N., & Islam, M. R. (2020). Deep learning based systems developed for fall detection: a review. *IEEE Access*, *8*, 166117-166137.

Janidarmian, M., Roshan Fekr, A., Radecka, K., & Zilic, Z. (2017). A comprehensive analysis on wearable acceleration sensors in human activity recognition. *Sensors*, *17*(3), 529.

Karlsson, M. K., Vonschewelov, T., Karlsson, C., Cöster, M., & Rosengen, B. E. (2013). Prevention of falls in the elderly: a review. *Scand J Public Health*, *41*(5), 442-454. https://doi.org/10.1177/1403494813483215

Keskes, O., & Noumeir, R. (2021). Vision-based fall detection using st-gcn. *IEEE Access*, *9*, 28224-28236.

Khan, M. A. A. H., Roy, N., & Misra, A. (2018). Scaling human activity recognition via deep learning-based domain adaptation. 2018 IEEE international conference on pervasive computing and communications (PerCom),

Kianoush, S., Savazzi, S., Vicentini, F., Rampa, V., & Giussani, M. (2016). Device-free RF human body fall detection and localization in industrial workplaces. *IEEE Internet of Things Journal*, *4*(2), 351-362.

Kim, Y.-S., Lee, J., Moon, Y., Kim, H. J., Shin, J., Park, J.-M., Uhm, K. E., Kim, K. J., Yoo, J. A., & Oh, Y. K. (2020). Development of a senior-specific, citizen-oriented healthcare service system in South Korea based on the Canadian 48/6 model of care. *BMC geriatrics*, *20*, 1-15.

Klein Brinke, J. (2018). *Device-free sensing and deep learning: analyzing human behavior through CSI using neural networks* University of Twente].

Kong, J., Wang, H., Wang, X., Jin, X., Fang, X., & Lin, S. (2021). Multi-stream hybrid architecture based on cross-level fusion strategy for fine-grained crop species recognition in precision agriculture. *Computers and Electronics in Agriculture*, *185*, 106134.

Koshmak, G., Loutfi, A., & Linden, M. (2016). Challenges and issues in multisensor fusion approach for fall detection. *Journal of Sensors*, *2016*(1), 6931789.

Kraft, D., Srinivasan, K., & Bieber, G. (2020). Deep learning based fall detection algorithms for embedded systems, smartwatches, and IoT devices using accelerometers. *Technologies*, *8*(4), 72.

Kummitha, R. K. R., & Crutzen, N. (2019). Smart cities and the citizen-driven internet of things: A qualitative inquiry into an emerging smart city. *Technological Forecasting and Social Change*, *140*, 44-53.

Lin, M., Chen, Q., & Yan, S. (2013). Network in network. *arXiv preprint arXiv:1312.4400*.

Liu, J., Liu, H., Chen, Y., Wang, Y., & Wang, C. (2019). Wireless sensing for human activity: A survey. *IEEE Communications Surveys & Tutorials*, *22*(3), 1629-1645.

Mahdi, S. Q., Gharghan, S. K., & Hasan, M. A. (2021). FPGA-Based neural network for accurate distance estimation of elderly falls using WSN in an indoor environment. *Measurement*, *167*, 108276.

Maia, J. C., Coutinho, J. F. V., Sousa, C. R. d., Barbosa, R. G. B., Mota, F. R. d. N., Marques, M. B., Silva, R. d. R. L., & Lima, R. B. d. S. (2018). Assistive technologies for demented elderly: a systematic review. *Acta Paulista de Enfermagem*, *31*, 651-658.

Mall, P. K., Singh, P. K., Srivastav, S., Narayan, V., Paprzycki, M., Jaworska, T., & Ganzha, M. (2023). A comprehensive review of deep neural networks for medical image processing: Recent developments and future opportunities. *Healthcare Analytics*, *4*, 100216. https://doi.org/https://doi.org/10.1016/j.health.2023.100216

Mansoor, M., Amin, R., Mustafa, Z., Sengan, S., Aldabbas, H., & Alharbi, M. T. (2022). A machine learning approach for non-invasive fall detection using Kinect. *Multimedia Tools and Applications*, *81*(11), 15491-15519.

Mattela, G., Tripathi, M., & Pal, C. (2022). A novel approach in WiFi CSI-based fall detection. *SN Computer Science*, *3*(3), 214.

Mekruksavanich, S., & Jitpattanakul, A. (2021). Deep convolutional neural network with rnns for complex activity recognition using wrist-worn wearable sensor data. *Electronics*, *10*(14), 1685.

Min, C., Alloulah, M., & Kawsar, F. (2018). An early resource characterisation of wi-fi sensing on residential gateways. Proceedings of the 5th Conference on Systems for Built Environments,

Moshiri, P. F., Shahbazian, R., Nabati, M., & Ghorashi, S. A. (2021). A CSI-based human activity recognition using deep learning. *Sensors*, *21*(21), 7225.

Nahian, M. J. A., Raju, M. H., Tasnim, Z., Mahmud, M., Ahad, M. A. R., & Kaiser, M. S. (2021). Contactless fall detection for the elderly. *Contactless Human Activity Analysis*, 203-235.

Nakamura, T., Bouazizi, M., Yamamoto, K., & Ohtsuki, T. (2020). Wi-fi-CSI-based fall detection by spectrogram analysis with CNN. GLOBECOM 2020-2020 IEEE Global Communications Conference,

Ngamakeur, K., Yongchareon, S., Yu, J., & Rehman, S. U. (2020). A survey on device-free indoor localization and tracking in the multi-resident environment. *ACM Computing Surveys (CSUR)*, *53*(4), 1-29.

Nguyen, T.-D. H., & Nguyen, H.-N. H. (2020). Towards a robust WiFi-based fall detection with adversarial data augmentation. 2020 54th Annual Conference on Information Sciences and Systems (CISS),

Noury, N., Fleury, A., Rumeau, P., Bourke, A. K., Laighin, G., Rialle, V., & Lundy, J.-E. (2007). Fall detection-principles and methods. 2007 29th annual international conference of the IEEE engineering in medicine and biology society,

Overview of the UK population: January 2021. (2021). *Office for National Statistics*.

Palipana, S., Rojas, D., Agrawal, P., & Pesch, D. (2018). FallDeFi: Ubiquitous fall detection using commodity Wi-Fi devices. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, *1*(4), 1-25.

Palmerini, L., Klenk, J., Becker, C., & Chiari, L. (2020). Accelerometer-based fall detection using machine learning: Training and testing on real-world falls. *Sensors*, *20*(22), 6479.

Pan, S. J., Zheng, V. W., Yang, Q., & Hu, D. H. (2008). Transfer learning for wifi-based indoor localization. Association for the advancement of artificial intelligence (AAAI) workshop,

Patil, R. (2015). Exercise in Older Women: Effects on Falls, Function, Fear of Falling and Finances.

Qian, X., Chen, H., Jiang, H., Green, J., Cheng, H., & Huang, M.-C. (2020). Wearable computing with distributed deep learning hierarchy: a study of fall detection. *IEEE Sensors Journal*, *20*(16), 9408-9416.

Ramachandran, A., & Karuppiah, A. (2020). A survey on recent advances in wearable fall detection systems. *BioMed research international*, *2020*(1), 2167160.

Rathor, R. G., & Joshi, R. D. (2021). Performance analysis of ieee802. 11ax (wi-fi 6) technology using multi-user mimo and up-link ofdma for dense environment. 2021 IEEE 2nd International Conference on Applied Electromagnetics, Signal Processing, & Communication (AESPC),

Rezaei, A. (2023). *Human Activity Recognition and Fall Detection Using Unobtrusive Technologies* UNSW Sydney].

Rubenstein, L. Z. (2006). Falls in older people: epidemiology, risk factors and strategies for prevention. *Age and ageing*, *35*(suppl_2), ii37-ii41.

Salau, B. A., Rawal, A., & Rawat, D. B. (2022). Recent advances in artificial intelligence for wireless internet of things and cyber–physical systems: A comprehensive survey. *IEEE Internet of Things Journal*, *9*(15), 12916-12930.

Şengül, G., Karakaya, M., Misra, S., Abayomi-Alli, O. O., & Damaševičius, R. (2022). Deep learning based fall detection using smartwatches for healthcare applications. *Biomedical Signal Processing and Control*, *71*, 103242.

Shalaby, E., ElShennawy, N., & Sarhan, A. (2022). Utilizing deep learning models in CSI-based human activity recognition. *Neural Computing and Applications*, 1-18.

Shang, S., Luo, Q., Zhao, J., Xue, R., Sun, W., & Bao, N. (2021). LSTM-CNN network for human activity recognition using WiFi CSI data. Journal of Physics: Conference Series,

Shen, Y., Bi, C.-X., Zhang, X.-Z., Zhang, Y.-B., & Zhou, R. (2024). Designing a sparse sensor array for sound field reconstruction using compressive-equivalent source method. *The Journal of the Acoustical Society of America*, *156*(2), 800-811.

Shi, Z., Cheng, Q., Zhang, J. A., & Da Xu, R. Y. (2022). Environment-robust WiFi-based human activity recognition using enhanced CSI and deep learning. *IEEE Internet of Things Journal*, *9*(24), 24643-24654.

Shiri, A., Prakash, B., Mazumder, A. N., Waytowich, N. R., Oates, T., & Mohsenin, T. (2021). An energy-efficient hardware accelerator for hierarchical deep reinforcement learning. 2021 IEEE 3rd International Conference on Artificial Intelligence Circuits and Systems (AICAS),

Shu, F., & Shu, J. (2021). An eight-camera fall detection system using human fall pattern recognition via machine learning by a low-cost android box. *Scientific reports*, *11*(1), 2471.

Singh, K., Rajput, A., & Sharma, S. (2020). Human Fall Detection Using Machine Learning Methods: A Survey. *International Journal of Mathematical, Engineering and Management Sciences*.

Smith, L. I. (2002). *A tutorial on principal components analysis*.

Sposaro, F., & Tyson, G. (2009). iFall: an Android application for fall monitoring and response. 2009 Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Surasakhon, W., Taparugssanagorn, A., Lerspalungsanti, S., & Thipprachak, K. (2022). Early detection of a fall using Wi-Fi and deep learning. *International Journal of Sensor Networks*, *40*(3), 190-202.

Tahir, A., Taylor, W., Taha, A., Usman, M., Shah, S. A., Imran, M. A., & Abbasi, Q. H. (2022). IoT based fall detection system for elderly healthcare. In *Internet of Things for Human-Centered Design: Application to Elderly Healthcare* (pp. 209-232). Springer.

Vedula, N. (2021). Running deep learning applications on resource constrained devices.

Wang, D., Yang, J., Cui, W., Xie, L., & Sun, S. (2021). Multimodal CSI-based human activity recognition using GANs. *IEEE Internet of Things Journal*, *8*(24), 17345-17355.

Wang, H., Zhang, D., Wang, Y., Ma, J., Wang, Y., & Li, S. (2016). RT-Fall: A real-time and contactless fall detection system with commodity WiFi devices. *IEEE Transactions on Mobile Computing*, *16*(2), 511-526.

Wang, L., Zhou, Y., Li, R., & Ding, L. (2022). A fusion of a deep neural network and a hidden Markov model to recognize the multiclass abnormal behavior of elderly people. *Knowledge-Based Systems*, *252*, 109351.

Wang, R., Niu, G., Cao, Q., Chen, C. S., & Ho, S.-W. (2024). A Survey of Visible-Light-Communication-Based Indoor Positioning Systems. *Sensors*, *24*(16), 5197.

Wang, X., Ellul, J., & Azzopardi, G. (2020). Elderly Fall Detection Systems: A Literature Survey. *Front Robot AI*, *7*, 71. https://doi.org/10.3389/frobt.2020.00071

Wang, X., Girshick, R., Gupta, A., & He, K. (2018). Non-local neural networks. Proceedings of the IEEE conference on computer vision and pattern recognition,

Wang, Y., Wu, K., & Ni, L. M. (2016). Wifall: Device-free fall detection by wireless networks. *IEEE Transactions on Mobile Computing*, *16*(2), 581-594.

Wei, B., Hu, W., Yang, M., & Chou, C. T. (2015). Radio-based device-free activity recognition with radio frequency interference. Proceedings of the 14th International Conference on Information Processing in Sensor Networks,

Wu, K., Huang, Y., Qiu, M., Peng, Z., & Wang, L. (2022). Toward Device-free and User-independent Fall Detection Using Floor Vibration. *ACM Transactions on Sensor Networks*, *19*, 1 - 20.

Wu, X., Zheng, Y., Chu, C.-H., Cheng, L., & Kim, J. (2022). Applying deep learning technology for automatic fall detection using mobile sensors. *Biomedical Signal Processing and Control*, *72*, 103355.

Xiao, J., Zhou, Z., Yi, Y., & Ni, L. M. (2016). A survey on wireless indoor localization from the device perspective. *ACM Computing Surveys (CSUR)*, *49*(2), 1-31.

Xu, T., Zhou, Y., & Zhu, J. (2018). New advances and challenges of fall detection systems: A survey. *Applied Sciences*, *8*(3), 418.

Xue, H., Jiang, W., Miao, C., Ma, F., Wang, S., Yuan, Y., Yao, S., Zhang, A., & Su, L. (2020). DeepMV: Multi-view deep learning for device-free human activity recognition. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, *4*(1), 1-26.

Yacchirema, D., de Puga, J. S., Palau, C., & Esteve, M. (2019). Fall detection system for elderly people using IoT and ensemble machine learning algorithm. *Personal and Ubiquitous Computing*, *23*(5), 801-817.

Yadav, S., & Gurjar, D. S. (2021). Secrecy performance of SIMO underlay cognitive radio networks over α–μ fading channels. *IEEE Access*, *9*, 62616-62629.

Yang, Z., Zhang, Y., & Zhang, Q. (2022). Rethinking fall detection with Wi-Fi. *IEEE Transactions on Mobile Computing*, *22*(10), 6126-6143.

Yao, S., Zhao, Y., Zhang, A., Hu, S., Shao, H., Zhang, C., Su, L., & Abdelzaher, T. (2018). Deep learning for the internet of things. *Computer*, *51*(5), 32-41.

Yousefi, S., Narui, H., Dayal, S., Ermon, S., & Valaee, S. (2017). A survey on behavior recognition using WiFi channel state information. *IEEE Communications Magazine*, *55*(10), 98-104.

Zafari, F., Gkelias, A., & Leung, K. K. (2019). A survey of indoor localization systems and technologies. *IEEE Communications Surveys & Tutorials*, *21*(3), 2568-2599.

Zeiler, M. D., & Fergus, R. (2013). Stochastic pooling for regularization of deep convolutional neural networks. *arXiv preprint arXiv:1301.3557*.

Zhang, R., Jing, X., Wu, S., Jiang, C., Mu, J., & Yu, F. R. (2020). Device-free wireless sensing for human detection: The deep learning perspective. *IEEE Internet of Things Journal*, *8*(4), 2517-2539.

Zou, H., Yang, J., Prasanna Das, H., Liu, H., Zhou, Y., & Spanos, C. J. (2019). WiFi and vision multimodal learning for accurate and robust device-free human activity recognition. Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops,

Appendix

Main File

Main file:

```
clear; close all; clc;

% Getting all days inside mat folder
path = './mat';
days = dir(path);
days(1:2) = [];

fall_csi_x = [];
fall_ctr = 1;
unfall_csi_x = [];
unfall_ctr = 1;

fall_csi_y = [];
unfall_csi_y = [];
sz = 97;
for day = string({days(:).name}) % For every day in all days
    files = dir(strcat(path, '/', day));
    files(1:2) = [];
    for file = string({files(:).name})
        if contains(file, 'falling')
            load(strcat(path, '/', day, '/', file));
            frames = [];
            for f = 1 : length(csi_trace)
                selected_csi = abs(find_stream(csi_trace{f, 1}.csi));
                frames(f, :) = selected_csi;
            end
            frame = mean(frames, 2);
            final_frame = self_padding(frame.', sz);
            fall_csi_x(fall_ctr, :) = final_frame;
            fall_ctr = fall_ctr + 1;
        elseif contains(file, 'nothing')
            load(strcat(path, '/', day, '/', file));
            frames = [];
            for f = 1 : length(csi_trace)
                selected_csi = abs(find_stream(csi_trace{f, 1}.csi));
                frames(f, :) = selected_csi;
            end
            frame = mean(frames, 2);
            final_frame = self_padding(frame.', sz);
            unfall_csi_x(unfall_ctr, :) = final_frame;
            unfall_ctr = unfall_ctr + 1;
        end
    end
end

%% Merging Labels and fall unfall to a single variable
fall_csi_y = ones(size(fall_csi_x, 1), 1);
unfall_csi_y = zeros(size(unfall_csi_x, 1), 1);

% Merging fall and unfall together
x_train = [fall_csi_x;unfall_csi_x];
```

```
y_train = [fall_csi_y;unfall_csi_y];
data = [x_train y_train];

% Randomly sequencing the samples
data = data(randperm(size(data, 1)), : );
x_train = data(:, 1:end - 1);
y_train = data(:, end);
```

Finding best stream

Finding best stream function:

```
function [selected_csi] = find_stream(comp_csi)
    minpower = inf;
    [ts, rs, ~] = size(comp_csi);
    for t = 1 : ts
        for r = 1 : rs
            selected_csi = comp_csi(t, r, :);
            selected_csi = selected_csi(:);
            phases = angle(selected_csi);
            sigpower = bandpower(phases);
            if sigpower < minpower
                minpower = sigpower;
                ht = t; rt = r;
            end
        end
    end
    selected_csi = comp_csi(1,1,:);
    selected_csi = selected_csi(:).';
end
```

Data Labeling Algorithm

Automatic Data Labeling Code

```
clear; close all; clc;
labeledData = readtable('preprocessed_data.csv');
labeledData = labeledData{:,:};
labeledData(1, :) = [];
labeledData(:, end) = [];

% Iterate through each signal
for signalIndex = 1:totalSignals
    signal = labeledData(signalIndex, :);

    % Iterate through different window sizes and offsets
    for windowIndex = 1:length(windowSizes)
        windowSize = windowSizes(windowIndex);
        slideSize = slideSizes(windowIndex);

        % Calculate variances for sliding windows
        for startIdx = 1:slideSize:signalLength-windowSize+1
            endIdx = startIdx + windowSize - 1;
            window = signal(startIdx:endIdx);
            variance = var(window);
```

```
            % Determine threshold based on the heuristic
            threshold = mean(variance) + thresholdMultiplier * std(variance);

            % Label the window based on the threshold
            if variance > threshold
                labeledData(signalIndex, startIdx:endIdx) = 1; % 1 for fallen
            else
                labeledData(signalIndex, startIdx:endIdx) = 0; % 0 for not
fallen
            end
        end
    end
end
```

Feature Selection

Features Selection function:

```
function [feats_vec] = feats_selection(csi_trace)
    feats_vec = zeros();
    for tctr = 1 : size(csi_trace, 1)
        comp_csi = csi_trace{tctr, 1}.csi;
        selected_csi = find_stream(comp_csi);
        amps = abs(selected_csi);
        phases = angle(selected_csi);
        mags = sqrt(amps.^2 + phases.^2);
        feats_vec(tctr, 1) = kurtosis(amps);
        feats_vec(tctr, 2) = max(amps);
        feats_vec(tctr, 3) = mean(amps);
        feats_vec(tctr, 4) = min(amps);
        feats_vec(tctr, 5) = std(amps);
        feats_vec(tctr, 6) = var(amps);
        feats_vec(tctr, 7) = skewness(amps);
        feats_vec(tctr, 8) = kurtosis(mags);
        feats_vec(tctr, 9) = max(mags);
        feats_vec(tctr, 10) = mean(mags);
        feats_vec(tctr, 11) = min(mags);
        feats_vec(tctr, 12) = std(mags);
        feats_vec(tctr, 13) = var(mags);
        feats_vec(tctr, 14) = skewness(mags);
        feats_vec(tctr, 15) = kurtosis(phases);
        feats_vec(tctr, 16) = max(phases);
        feats_vec(tctr, 17) = mean(phases);
        feats_vec(tctr, 18) = min(phases);
        feats_vec(tctr, 19) = std(phases);
        feats_vec(tctr, 20) = var(phases);
        feats_vec(tctr, 21) = skewness(phases);
    end
end
```

CNN Code

CNN Implementation Code:

```
clear; close all; clc;
data = readtable('preprocessed_data.csv');
data = data{:,:};
data(1, :) = [];
data = data(randperm(size(data, 1)), : );
x_train = data(:, 1:end - 1);
y_train = data(:, end);
%% Prediction using CNN model
numTrain = floor(0.8*size(x_train, 1));
numValid = floor(0.15*size(x_train, 1));
numTest = size(x_train, 1) - numTrain - numValid;
dataTrain = x_train(1:numTrain,:);
dataValid = x_train(numTrain+1:numTrain+numValid,:);
dataTest = x_train(numTrain+numValid+1:end,:);
YTrain = categorical(y_train(1:numTrain,:));
YValid = categorical(y_train(numTrain+1:numTrain+numValid,:));
YTest = categorical(y_train(numTrain+numValid+1:end,:));
mu = mean(dataTrain);
sig = std(dataTrain);
% Normalize the data
XTrain = num2cell(((dataTrain - mu) ./ sig), 2);
XTrain = cellfun(@transpose,XTrain,'UniformOutput',false);
XValid = num2cell(((dataValid - mu) ./ sig), 2);
XValid = cellfun(@transpose,XValid,'UniformOutput',false);
XTest = num2cell(((dataTest - mu) ./ sig), 2);
XTest = cellfun(@transpose,XTest,'UniformOutput',false);
% Set the parameters of CNN model
numFeatures = size(XTrain{1, 1}, 1);
numClasses = numel(unique(YTrain));
numHiddenUnits = 100;
filterSize = 3; numFilters = 64;
layers = [sequenceInputLayer(numFeatures)
    convolution1dLayer(filterSize,numFilters,Padding="causal")
    reluLayer
    globalAveragePooling1dLayer
    fullyConnectedLayer(numClasses)
    softmaxLayer
    classificationLayer];
options = trainingOptions('adam', ...
    'MaxEpochs',200, ...
    'GradientThreshold',1, ...
    'InitialLearnRate',0.005, ...
    'LearnRateSchedule','piecewise', ...
    'LearnRateDropPeriod',125, ...
    'LearnRateDropFactor',0.2, ...
    'Verbose',0, ...
    'ValidationData',{XValid,YValid},...
    'Plots','training-progress');
net = trainNetwork(XTrain,YTrain,layers,options);
YPred = classify(net, XTest);
acc = round((sum(YPred == YTest)./numel(YTest))*100);
```

LSTM Code

LSTM Implementation Code:

```
clear; close all; clc;
data = readtable('preprocessed_data.csv');
data = data{:,:};
data(1, :) = [];
data = data(randperm(size(data, 1)), : );
x_train = data(:, 1:end - 1);
y_train = data(:, end);
%% Prediction using LSTM model
numTrain = floor(0.8*size(x_train, 1));
numValid = floor(0.15*size(x_train, 1));
numTest = size(x_train, 1) - numTrain - numValid;
dataTrain = x_train(1:numTrain,:);
dataValid = x_train(numTrain+1:numTrain+numValid,:);
dataTest = x_train(numTrain+numValid+1:end,:);
YTrain = categorical(y_train(1:numTrain,:));
YValid = categorical(y_train(numTrain+1:numTrain+numValid,:));
YTest = categorical(y_train(numTrain+numValid+1:end,:));
mu = mean(dataTrain);
sig = std(dataTrain);
% Normalize the data
XTrain = num2cell(((dataTrain - mu) ./ sig), 2);
XTrain = cellfun(@transpose,XTrain,'UniformOutput',false);
XValid = num2cell(((dataValid - mu) ./ sig), 2);
XValid = cellfun(@transpose,XValid,'UniformOutput',false);
XTest = num2cell(((dataTest - mu) ./ sig), 2);
XTest = cellfun(@transpose,XTest,'UniformOutput',false);
% Set the parameters of LSTM model
numFeatures = size(XTrain{1, 1}, 1);
numClasses = numel(unique(YTrain));
numHiddenUnits = 100;
filterSize = 3; numFilters = 64;
layers = [sequenceInputLayer(numFeatures)
    lstmLayer(numHiddenUnits, 'OutputMode','last')
    fullyConnectedLayer(numClasses)
    softmaxLayer
    classificationLayer];
options = trainingOptions('adam', ...
    'MaxEpochs',200, ...
    'GradientThreshold',1, ...
    'InitialLearnRate',0.005, ...
    'LearnRateSchedule','piecewise', ...
    'LearnRateDropPeriod',125, ...
    'LearnRateDropFactor',0.2, ...
    'Verbose',0, ...
    'ValidationData',{XValid,YValid},...
    'Plots','training-progress');
net = trainNetwork(XTrain,YTrain,layers,options);
YPred = classify(net, XTest);
acc = round((sum(YPred == YTest)./numel(YTest))*100);
```

ConvLSTM Code

ConvLSTM Implementation Code:

```
clear; close all; clc;
data = readtable('preprocessed_data.csv');
data = data{:,:};
data(1, :) = [];
data = data(randperm(size(data, 1)), : );
x_train = data(:, 1:end - 1);
y_train = data(:, end);
%% Prediction using LSTM model
numTrain = floor(0.8*size(x_train, 1));
numValid = floor(0.15*size(x_train, 1));
numTest = size(x_train, 1) - numTrain - numValid;
dataTrain = x_train(1:numTrain,:);
dataValid = x_train(numTrain+1:numTrain+numValid,:);
dataTest = x_train(numTrain+numValid+1:end,:);
YTrain = categorical(y_train(1:numTrain,:));
YValid = categorical(y_train(numTrain+1:numTrain+numValid,:));
YTest = categorical(y_train(numTrain+numValid+1:end,:));
mu = mean(dataTrain);
sig = std(dataTrain);
% Normalize the data
XTrain = num2cell(((dataTrain - mu) ./ sig), 2);
XTrain = cellfun(@transpose,XTrain,'UniformOutput',false);
XValid = num2cell(((dataValid - mu) ./ sig), 2);
XValid = cellfun(@transpose,XValid,'UniformOutput',false);
XTest = num2cell(((dataTest - mu) ./ sig), 2);
XTest = cellfun(@transpose,XTest,'UniformOutput',false);
% Set the parameters of LSTM model
numFeatures = size(XTrain{1, 1}, 1);
numClasses = numel(unique(YTrain));
numHiddenUnits = 100;
filterSize = 3; numFilters = 64;
layers = [
    sequenceInputLayer(numFeatures)
%     lstmLayer(numHiddenUnits)
%     reluLayer
    convolution1dLayer(filterSize,numFilters,Padding="causal")
    reluLayer
%     layerNormalizationLayer
    globalAveragePooling1dLayer
    fullyConnectedLayer(200)
    reluLayer
    lstmLayer(200)
    fullyConnectedLayer(numClasses)
    softmaxLayer
    classificationLayer];
options = trainingOptions('adam', ...
    'MaxEpochs',200, ...
    'GradientThreshold',1, ...
    'InitialLearnRate',0.005, ...
    'LearnRateSchedule','piecewise', ...
    'LearnRateDropPeriod',125, ...
    'LearnRateDropFactor',0.2, ...
    'Verbose',0, ...
    'ValidationData',{XValid,YValid},...
    'Plots','training-progress');
```

```
net = trainNetwork(XTrain,YTrain,layers,options);
YPred = classify(net, XTest);
acc = round((sum(YPred == YTest)./numel(YTest))*100);
```