Real-Time Human Activity Detection Using Wi-Fi CSI and LSTM on Edge Devices

Ravi Kumar R.[1*], Shravan Kumar A.[2], Sri Harsha N.[3], Aditya Sarma P.[4] and Sree Varsha, R.[5]
[1,2,3,4] Department of ECE, VNRVJIET, Hyderabad, India
[5] School of Electronics Engineering (SENSE), VIT-AP, Amaravathi, India
*Corresponding author e-mail: rayala.ravi2024@gmail.com

Abstract. Wi-Fi sensing for Human Activity Detection (HAD) provides a non-intrusive, privacy-preserving approach for monitoring human activity in indoor environments. This paper presents a real-time human activity detection framework based on Channel State Information (CSI) acquired from an ESP-32 embedded development board equipped with Wi-Fi. Human activities such as walking, sitting, standing, and falling are classified based on CSI data using a time series variant of deep learning algorithm called the Long Short-Term Memory (LSTM). 5-fold cross-validation indicates the system's strong generalization and more than 93% accuracy. To demonstrate successful real-time classification, the built model is run on a Raspberry Pi 4 using a sliding window method. This system demonstrates the potential use of Wi-Fi sensing in smart homes, smart healthcare, and safety-critical applications.

Keywords: Wi-Fi sensing, human activity detection, channel state information, LSTM, smart environments, edge computing, ESP32, Raspberry Pi.

1 Introduction

Falls are a major global public health concern, particularly for older adults. They are the second leading cause of unintentional injury death worldwide, after road traffic injuries. The World Health Organization (WHO) estimates approximately 684,000 fatal falls annually, with the highest death rates among adults aged 60 and over in all regions. Annually, an estimated 28–35% of people aged 65 and above experience falls, increasing to 32–42% for those over 70 years (WHO 2008). Delayed assistance after falls, known as long lie, worsens outcomes, increasing mortality and causing severe complications such as dehydration and hypothermia (Smith 2021). Beyond fatal falls, about 37.3 million falls require medical attention yearly, underscoring the urgent need for effective fall detection and response systems amid the global aging population.

Traditional Human Activity Detection (HAD) methods such as wearables and camera-based systems face challenges including user compliance, intrusiveness, privacy concerns, and line-of-sight limitations. This has catalyzed interest in non-invasive, device-free sensing, particularly leveraging existing Wi-Fi infrastructure. Wi-Fi Channel State Information (CSI)-based sensing detects subtle changes in signal

S. Mantena (ed.), *Proceedings of the Conference on Social and Sustainable Innovation in Technology & Engineering (SASI-ITE 2025)*, Advances in Intelligent Systems Research 199,
https://doi.org/10.2991/978-94-6463-940-7_32

propagation caused by human movement, enabling passive indoor activity monitoring (Miller & Green 2016). This technology shows promise for critical applications such as elderly fall detection, continuous contact less health monitoring (e.g., respiration and heart rate), and vehicle safety, including detecting infants left in cars (Johnson & White 2022). The evolution of Wi-Fi-based HAD emphasizes unobtrusive, scalable, privacy-conscious solutions, with data-driven models and deep learning advancing its effectiveness. Recent research demonstrates high accuracy in activity classification despite ongoing challenges in environmental robustness and real-time processing (Brown & Davis 2022).

2 Literature review

Recent research on Channel State Information (CSI) for Human Activity Detection (HAD) demonstrates significant progress through advanced machine learning techniques, especially deep learning, to automate feature extraction and enhance classification accuracy. Early work primarily used handcrafted features, but studies like Wang et al. (2019) leveraged deep learning on CSI amplitude data achieving strong results, while more recent approaches incorporating attention mechanisms with both amplitude and phase data, as in Quy and Lee (2023), further improve performance. Specialized applications such as fall detection (Wu et al., 2021), direction-aware recognition (Zeng et al., 2021), and multi-user detection (Wong et al., 2022) show promise, though challenges remain with overlapping motions and hardware requirements. CSI has also been effective in passive vital sign monitoring (Soto et al., 2020). Key challenges include limited generalization across different environments due to environmental variability and multipath effects (Boudlal et al., 2021), the need for large annotated datasets (Wang et al., 2019), and sensitivity to noise. Transfer learning attempts (Hu, 2019) help but do not remove the need for re-training.

A critical unmet need is a lightweight, real-time, privacy-preserving HAD system with good cross-environment generalization for practical health and smart home uses. LSTM and related deep models show promise in addressing these limits. Sensor-based solutions for HAR, like those using Wi-Fi CSI or inertial sensors, are generally not affected by environmental factors such as lighting conditions, which can hinder vision-based HAR systems. Deep learning models can also be trained on large, diverse datasets to improve their ability to generalize and perform robustly in different environments. Some models, for instance, a semi-supervised adversarial learning approach using LSTM, can adapt to changes in human activity routines and new activities, making them robust in real-world situations.

The main challenge with real-time processing is the computational overhead of complex models. To address this, hybrid architectures and optimization techniques are used. For instance, a hybrid CNN-LSTM model combines the strengths of both, automatically learning features from raw data to improve efficiency. Additionally, new approaches like TinyML, which focuses on deploying machine learning models on resource-constrained devices, are being used to reduce computational and energy requirements, enabling low-latency, real-time HAR systems.

3 Methodology

In the system shown in Figure 1., a Wi-Fi Access Point (AP) continuously transmits signals within a defined indoor environment, specifically a room measuring 3 meters by 4.5 meters. When a person performs activities such as walking, sitting, falling, or standing, their movements cause variations in the signal propagation paths. These variations are a result of the multipath effect, where the Wi-Fi signals are reflected, refracted, and scattered by the human body and nearby objects. A receiver, such as an ESP32 device, captures the altered CSI data generated by these interactions. This data is pre-processed and given as input to an LSTM model, which is well-suited for learning temporal dependencies in time-series data. The trained LSTM model performs inference to classify the specific human activity taking place.

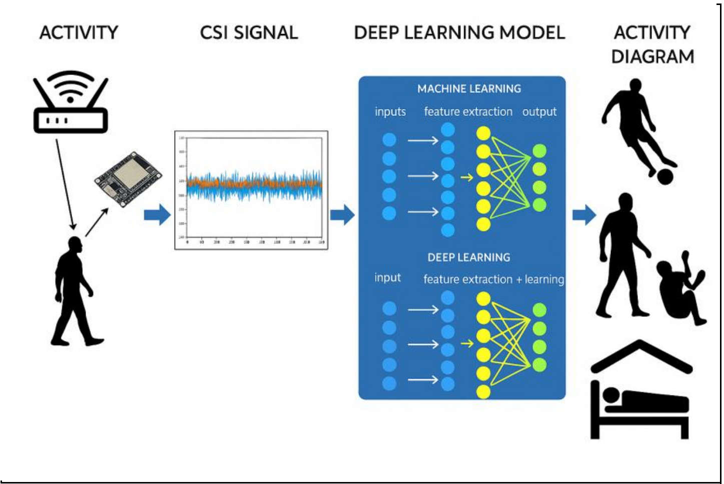

Fig. 1. CSI based HAD Methodology

The recognized activities in this setup include walking, falling, sitting down, and standing up. The system uses these classification results to infer the user's current movement or state, enabling a wide range of applications like smart home automation, elderly care monitoring, and other real-time context-aware systems. The framework integrates technical precision with conceptual clarity throughout the data collection, pre-processing, model creation, and evaluation phases. With the ESP32-WROOM module, the following setup was used for real-time human activity identification utilizing CSI.

ESP32 modules, which function as Wi-Fi transceivers and can record fine-grained wireless channel information, were used to gather CSI amplitude data. Walking, sitting, standing, and falling are among the human behaviors that were conducted indoors. Annotated labels describing the ground truth activity were associated with each data session. To effectively analyze the continuous stream of CSI data, a "sliding window" technique was employed. This method involved segmenting the ongoing CSI measurements into discrete, manageable samples for further processing.

Each window comprised 20 consecutive CSI snapshots, with each snapshot representing a brief, instantaneous capture of the wireless signal's characteristics. These groups of 20 frames were then treated as a single data sample, providing a localized view of the activity within that specific time segment. To ensure smooth transitions between samples and a finer granularity in detecting activity, the windows were designed to overlap. This was achieved by setting a "slide size" of 10 frames, such that each subsequent window began 10 frames after the start of the previous one. This overlap allowed for a continuous and more nuanced analysis of the evolving CSI patterns.

3.1 Data Collection Setup

The data-set is created with [19997, 30] samples and labels like fall, sit down, stand up, walk, Fall, Lie, Bend, Run and no activity. The experimental setup for data collection involves a Wi-Fi Access Point (AP), an ESP32, and a packet generator. The Wi-Fi AP acts as the signal source, emitting Wi-Fi signals throughout the environment. An ESP32, likely configured as a receiver, is responsible for collecting CSI data from these Wi-Fi signals. The packet generator is used to control the transmission of Wi-Fi packets or as a central point for initial data handling and configuration, as shown in Figure 2. Data is collected on ESP32 using a callback mechanism from the physical layer.

The collected CSI data from the ESP32 is then transmitted via a wired interface to a Raspberry Pi, which serves as the processing unit for training as well as inferencing. This setup allows for the acquisition of raw Wi-Fi CSI data, which is then processed on the Raspberry Pi to perform activities like human activity detection. To manage this complexity, threads are created and are executed on multiple cores available in ESP32. The room dimensions are approximately 4.85 m x 3.35 m, covering an area of 16.33 m^2. The ESP32 is positioned near a storage unit, and the packet generator is situated on the opposite side of the room to serve packet generation routed through it.

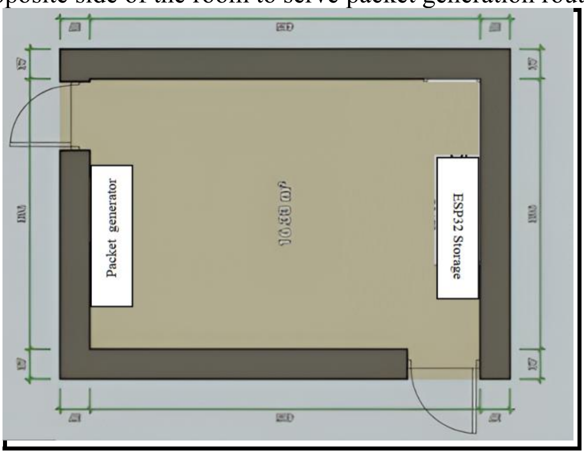

Fig2.CSI based HAD Data Collection Setup

The HAR data-set employed in this work is based on Wi-Fi CSI and offers high-resolution information regarding wireless signal propagation affected by human movements. Raw CSI data initially contains 19,997 samples, with each being defined by 30 features (sub-carrier amplitude) and labeled with corresponding single-column labels. To process such time-series data for activity classification, a sliding window technique was used with a window size of 20 and a slide size of 10. This operation converted the original CSI data into a sequence of 1,998 windows, each being of size (20, 30), and flattened these into a 600-feature vector per window. At the same time, activity labels were converted into 1,998 one-hot encoded vectors, each corresponding to one of three different activity classes. With the final shape of (1998,600) for CSI features and (1998,3) for one-hot encoded labels, the pre-processed data-set is useful for developing and testing robust HAR models by extracting temporal patterns inherent in human activities.

Finally, to classify the activity reflected within each window, a threshold of 60% was applied. If 60% or more of the CSI frames within a given 20-frame window indicated a particular activity, that class was assigned to the entire window. Conversely, any window that failed to meet this 60% consensus was labeled as "no activity" providing a clear distinction between periods of discernible action and those where no specific activity could be confidently identified. This is particularly important for activities with similar patterns, where a slight variation in the signal might otherwise lead to mis-classification. This identified strategy generated five distinct activity classes: fall, sit down, stand up, walk, Lie, Bend, Run and no activity.

3.2 Pre-processing

For model input, only amplitude data from 30 CSI sub-carriers was used. A 3D tensor of the format [samples, 20 time steps, 30 features] was created from each sample, where time steps stand for successive CSI frames. Sub-carrier amplitude values are represented as features. The data was normalized to improve convergence during training by bringing all values into a comparable range. For downstream modeling, the processed data was saved in CSV format.

3.3 Model Building

An LSTM network was employed due to its effectiveness in handling time-dependent data. LSTM networks represent a unique type of Recurrent Neural Network (RNN) specifically developed to address the issues of vanishing and exploding gradients that conventional RNNs encounter while attempting to learn long-term dependencies in sequential data. Unlike simple RNNs, LSTMs incorporate a cell state or memory cell that runs through the entire chain, allowing information to be carried forward or forgotten selectively. This mechanism is regulated by several "gates," each managed by a sigmoid activation function, which produces values ranging from 0 to 1, deciding the extent of information that is allowed to pass through. The operations within a single LSTM unit at a given time step t are defined by a series of update equations.

These equations govern the processing of the input data at the present time step (x_t), the hidden state (h_{t-1}), and the cell state (C_{t-1}) from the preceding time step, resulting in the generation of the new hidden state (h_t) and cell state (C_t).

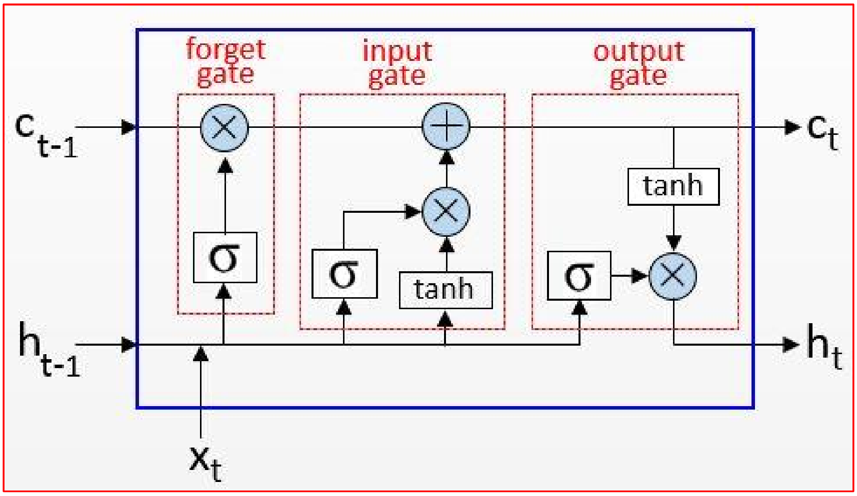

Fig. 3. Schematic Diagram of LSTM Model

Input Concatenation: Before being processed by the gates, the current input x_t and the preceding hidden state $\mathbf{h_{t-1}}$ are concatenated.

Forget Gate ($\mathbf{f_t}$): This gate determines which information from the prior cell state C_{t-1} ought to be discarded.

Input Gate ($\mathbf{I_t}$): This gate decides what part of the new information is going to be stored in the cell state.

Candidate Cell State ($\tilde{C}_t$): Based on the current input and previous hidden state, this new cell state is generated.

Cell State Update ($\mathbf{C_t}$): The core of the LSTM, where the previous cell state is updated by forgetting old information and adding new information.

Output Gate ($\mathbf{O_t}$): Determines which components of the cell state will be produced as the new hidden state.

Hidden State Update ($\mathbf{h_t}$): The final output of the LSTM unit for the current time step.

σ: Sigmoid activation function, $\sigma(x)=1/(1+e^{-x})$.

tanh: Hyperbolic tangent activation function $\tanh(x)=(e^{x}-e^{-x})/(e^{x}+e^{-x})$.

$\mathbf{W_f}$, $\mathbf{W_i}$,$\mathbf{W_C}$,$\mathbf{W_o}$: Weight matrices corresponding to the forget, input, candidate, and output gates, respectively. These parameters are learned throughout the training process.

b_f, b_i, b_C, b_o: Bias vectors corresponding to the forget, input, candidate, and output gates, respectively. These are also learned parameters.

An LSTM architecture designed for sequential data processing, such as windowed CSI input, typically comprises three key components. The input layer is responsible for accepting the incoming sequences, often shaped as (20, 30) to accommodate the specific input format. Following this, the LSTM layer itself, configured with 100 units, plays a crucial role. Each of these units possesses a memory cell, enabling it to capture and learn temporal dependencies within the input sequence. This layer effectively condenses the sequential information into a concise "context vector." Finally, the Dense Output Layer with softmax serves as the classification head. It's a fully connected layer with 5 neurons, corresponding to the number of activity classes. The softmax activation function applied here transforms the raw neuron outputs into class probabilities, ensuring their sum is 1, which facilitates interpreting the model's confidence in its predictions for each activity class.

3.4 Training AND Hyper-parameter Tuning:

Categorical cross-entropy was used as the loss function during model compilation, and it worked well for multi-class classification. With a learning rate of 0.0001, the Adam optimizer - an adaptive algorithm renowned for its effective convergence was employed. The training process had a batch size of 16 and lasted for 50 epochs. 5-Fold Cross-Validation was employed to improve the model's generalization skills. This method entailed methodically switching up the training and validation subsets during rounds.

3.5 Evaluation and Performance Metrics:

Training and validation performance were monitored through accuracy and loss graphs.

Accuracy: Accuracy is a fundamental metric for classification problems, representing the proportion of correctly predicted instances out of the total instances.

Loss (Loss Function): A loss function quantifies the error or "cost" incurred by a model's predictions compared to the actual true values. The objective during model training is to minimize this loss. Although not the primary focus, confusion matrices were also examined to identify overlaps between similar activities. This end-to-end approach ensures both scientific rigor and practical applicability, making it suitable for submission to high-quality conferences in the fields of embedded AI, wireless sensing, or smart environments.

3.6 Real-time Implementation on Raspberry Pi

The trained LSTM model is deployed on the Raspberry Pi 4 Model B. This process involves converting the original deep learning model into a lightweight format, optimized for efficient inference on edge devices with moderate computational power. Its enhanced computational flexibility enables larger models, more complex pre-processing, and seamless integration into smart home ecosystems. The Raspberry Pi, which houses the real-time activity identification pipeline, is powered by a Linux-based operating system, for instance, Raspberry Pi OS (an operating system based on Debian, specifically tailored for Raspberry Pi hardware). CSI data are collected using an external embedded device that has capabilities of CSI extraction.

A Python script controls the continuous acquisition of CSI streams and the execution of inference. 300 frames of real-time CSI data are gathered and split into 5-second windows, with 50% overlap between windows. These data windows are fed into the LSTM model for classification after being pre-processed to match the model input format [20, 30]. The sliding window ensures smooth transition detection while preserving the action's context. The TensorFlow Lite interpreter uses tflite-runtime, designed for ARM processors on the Raspberry Pi, to run the .tflite model. After inference, the classified activity is transmitted to remote servers via protocols like MQTT, HTTP, and WebSocket. The predicted activity is displayed via an HDMI/GUI, and an alert message is initiated to the attendants/respondents. The attendant is given the provision to receive the message through a developed application on handsets. In reaction to specific detection, like falls, GPIO pins on Raspberry Pi turn on physical indicators, such as buzzers and LEDs..

4 Results and discussion:

The experimental results for human activity detection using WiFi-based CSI are presented through the analysis of CSI amplitude variations over time. The CSI amplitude data was effectively extracted and analyzed. The data collection aimed to identify unique CSI amplitude patterns corresponding to various human movements in smart environments. The figures show CSI amplitudes in 42 (Figures 4,5,6,7,8 and 9) sub-carriers over time, illustrating signal variations caused by human motion. Specifically, Figure 6. (Standing Graph) shows relatively stable amplitudes with minor fluctuations, indicating minimal signal disruption.

In contrast, Figure 5.(Fall Graph) exhibits a sharp and distinct deviation in CSI amplitudes around the 5500-6000 ms mark, characteristic of a rapid change in the signal propagation path due to a fall. Figure 8. Walking graph displays consistent and periodic amplitude fluctuations, reflecting the repetitive motion of walking as the body intermittently obstructs and reflects Wi-Fi signals. Lastly, Sitting pattern reveals moderate CSI amplitude changes, providing a unique signature for the transition to and from a sitting position. These findings underscore the potential of using CSI amplitude data collected using low-cost ESP32 devices for robust human activity detection.

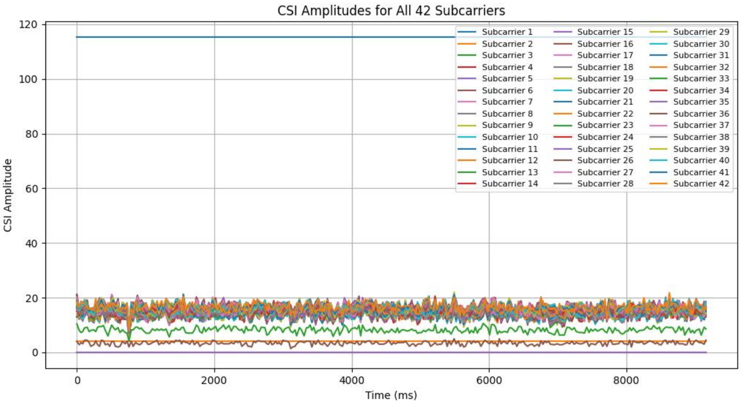

Fig 4. Plot for Idle CSI based HAD (Unfiltered)

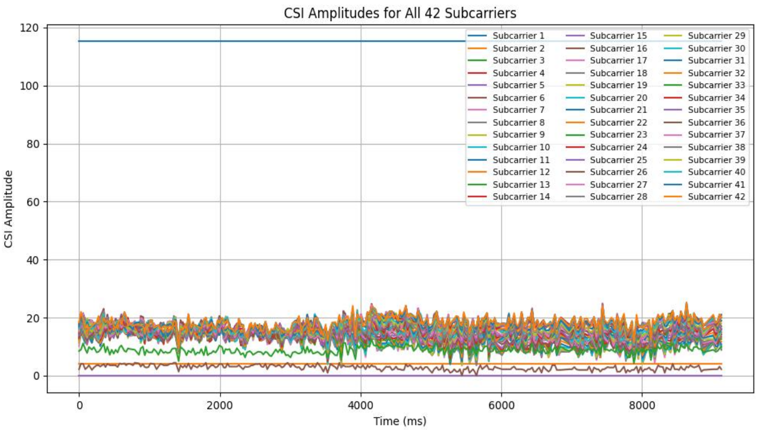

Fig 5. Plot for Fall CSI based HAD (Unfiltered)

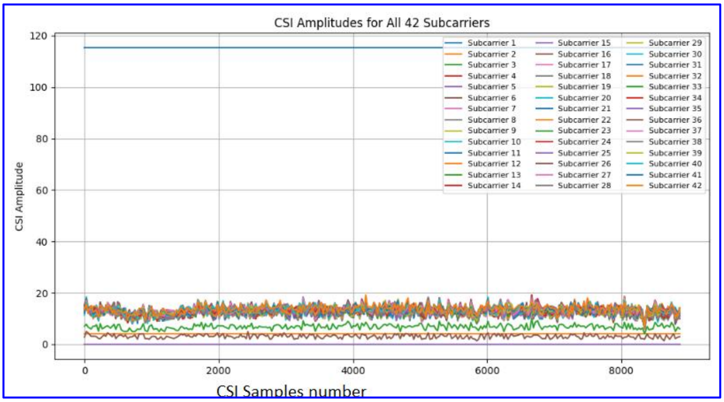

Fig 6.Plot for CSI based HAD Standing Graph

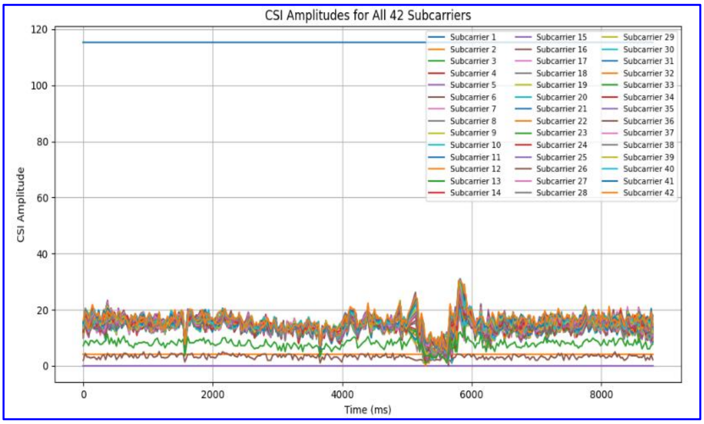

Fig 7.Plot for CSI based HAD Walking Graph

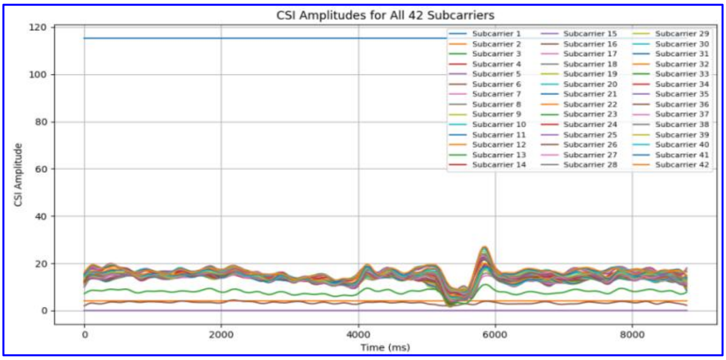

Fig 8. Plot for CSI based HAD for "Walk" Activity(Filtered)

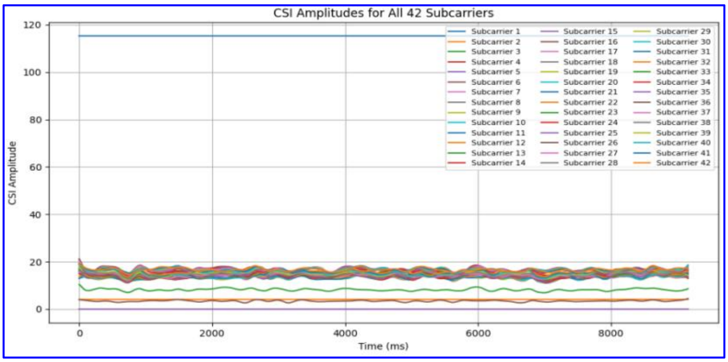

Fig 9.. CSI based HAD Plot for No Activity

4.1 Training and Validation Performance:

Training and validation performance was monitored using accuracy and loss graphs that are shown in Figure 10. Fall Detection Performance and Figure 11. model accuracy loss. The model achieved an average validation accuracy of 93.5% +/- 0.9%, indicating consistent recognition across activity types. Although not the primary focus, confusion matrices were also examined to identify overlaps between similar activities. This end-to-end approach ensures both scientific rigor and practical applicability, making it suitable for submission to high-quality conferences in the fields of embedded AI, wireless sensing, or smart environments. The model was trained for 50 epochs with a batch size of 16, using the Adam optimizer and categorical cross-entropy as the loss function. The training process was monitored using accuracy and loss plots.

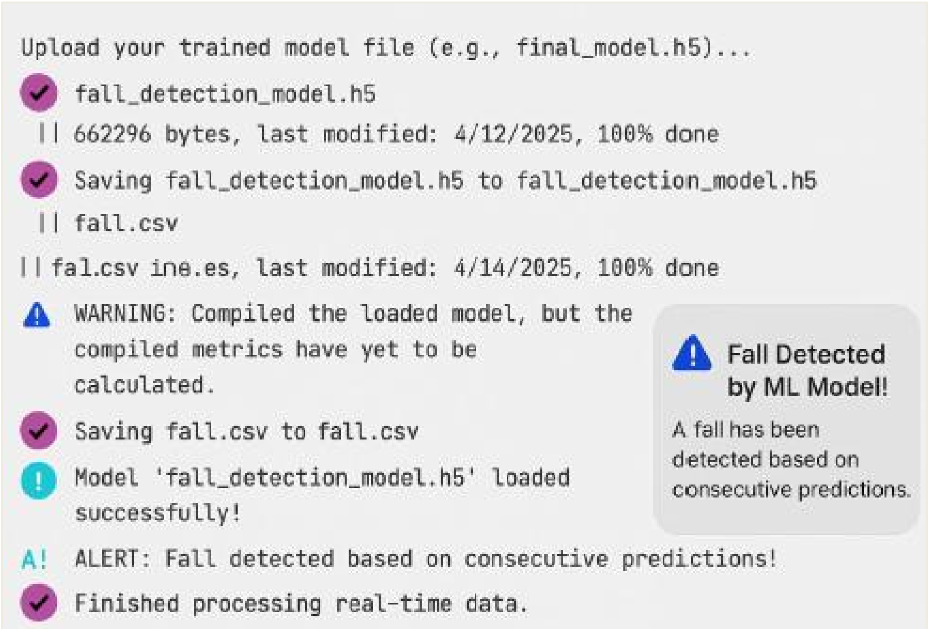

Figure 10: From ML Model Prediction to Instant Mobile Alerts Inferencing.

Accuracy Trends: The model exhibited steady improvement in accuracy across epochs, achieving convergence without overfitting. The final average validation accuracy of a 5-fold cross-validation has reached 93.5% +/- 0.9%.

Loss Reduction: The loss function decreased consistently during training, indicating effective learning. Validation loss also stabilized, reflecting good generalization capabilities.

- 484/484 ━━━━━━━━━━━━━━━━━━━━ 5s 6ms/step - accuracy: 0.9167 - loss: 0.1710 - val_accuracy: 0.9185 - val_loss: 0.1547 Epoch 49/58
- 484/484 ━━━━━━━━━━━━━━━━━━━━ 3s 6ms/step - accuracy: 0.9261 - loss: 0.1269 - val_accuracy: 0.9337 - val_loss: 0.1588 Epoch 50/58
- 484/484 ━━━━━━━━━━━━━━━━━━━━ 2s 6ms/step - accuracy: 0.9196 - loss: 0.1620 - val_accuracy: 0.9199 - val_loss: 0.1650
- Final Accuracy: 93.5% (+/- 0.9%)

Figure 11:CSI based HAD Final Model Accuracy of LSTM Model and Loss Metrics

4.2 Confusion Matrix Analysis:

To evaluate classification robustness, confusion matrices were generated for each fold and aggregated. These matrices provide insight into the model's precision and recall for each activity class. Figure 12 showing model performance across activities using Confusion Matrix. Fall and Walk activities showed the highest classification accuracy, demonstrating the model's capability to capture both abrupt and continuous motion. Sit Down and Stand Up showed occasional mis-classification, primarily due to similar transitional patterns in CSI. No activity was consistently recognized, benefiting from the threshold-based annotation. Metrics Derived from the Confusion Matrix, the equations that quantify model performance are calculated directly from the counts within the confusion matrix.

Precision (Positive Predictive Value): What proportion of positive predictions were actually correct.

Recall (Sensitivity, True Positive Rate): What proportion of actual positives were correctly identified.

F1 Score: The harmonic mean of precision and recall provides a balance between them.

Specificity (True Negative Rate): What proportion of actual negatives were correctly identified?

Accuracy: Overall correct predictions.

The confusion matrix provides information on how well the classification model performs in different tasks. The highest effective forecast of any category is "No Activity," which the algorithm identifies with a remarkably high accuracy, correctly classifying 11,116 cases. Additionally, it does well on the following tasks: Run (656 correct), Fall (598 correct), Stand (316 correct), Walk (286 correct), and Sit (239 correct). For Bend and Lie, performance is mediocre, with 163 and 128 accurate predictions, respectively. One noteworthy finding is that the model frequently wrongly predicts various activities as No Activity, as evidenced by the many occurrences of Walk (32), Sit (32), Fall (56), Lie (40), Bend (21), and Run (31) being mis-classified as No Activity . On the other hand, although they occur less frequently, there are also certain cases where "No Activity" is incorrectly identified as another activity.

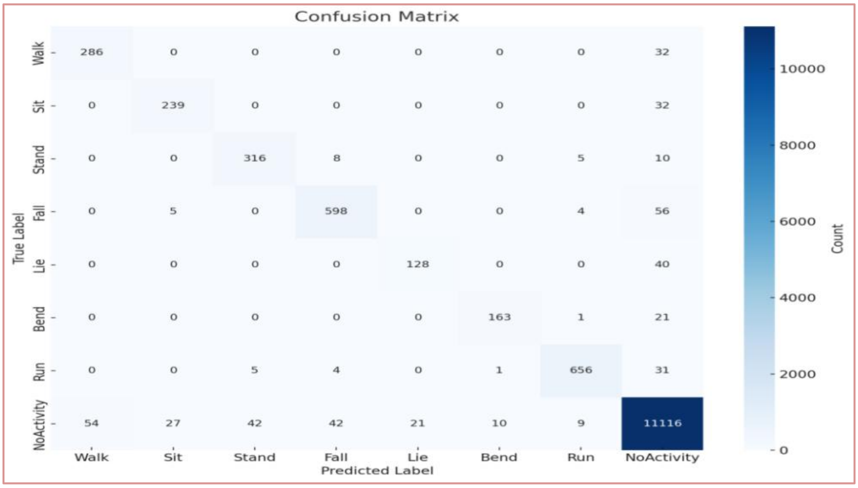

Figure 12: Confusion Matrix showing model performance across activities

Importantly, there appears to be little confusion among the different active categories themselves (for example, "Walk" is rarely confused with Sit or Stand), indicating that the model successfully distinguishes between these active states. This mis-classification pattern, especially with regard to "No Activity," may indicate that No Activity samples predominate in the data set or that the model has a bias in favor of this class when uncertainty occurs.

Overall Accuracy: The total accuracy of the model, calculated as the sum of correct predictions divided by the overall number of samples = 13502/13960 $\approx$0.9672

Per-Class and Averaged Metrics: Table I presents the precision, recall, F1 score, and specificity for each activity class, along with their macro and micro averages.

Class	Precision	Recall	F1-Score	Specificity
Walk	0.8412	0.8994	0.8693	0.996
Sit	0.8819	0.8819	0.8819	0.9977
Stand	0.8705	0.9322	0.8998	0.9966
Fall	0.9172	0.9019	0.9095	0.9959
Lie	0.8591	0.7619	0.8074	0.9985
Bend	0.9368	0.8811	0.9081	0.9992
Run	0.9711	0.9412	0.9559	0.9986
NoActivity	0.9804	0.9819	0.9811	0.9218
Macro Avg.	0.9073	0.9002	0.9016	0.988
Micro Avg.	0.9672	0.9672	0.9672	-

TABLE I:Performance Metrics Derived from Confusion Matrix

4.3 Robustness and Generalization:

The model was tested using 5-fold cross-validation to ensure its robustness and consistency. High validation accuracy with low variance across folds (standard deviation of 0.9%) indicates a strong generalization to unseen data. The overlapping window mechanism and the use of temporal features through the LSTM architecture helped to smooth out noise and capture the motion context effectively.

The two charts plotted show how well a machine learning model performs during training and validation over 50 epochs. The loss curve in Figure 14. demonstrates that the model is learning as both training and validation loss sharply decline during the first few epochs. Even though both losses are still declining, there are notable variations in them, especially in the validation loss after about epoch 20. Around epoch 35, there is a noticeable increase in validity loss, which soon subsides. This tendency is reflected in the accuracy curve in Figure 13. In the early epochs, training and validation accuracy rise quickly, indicating better performance.

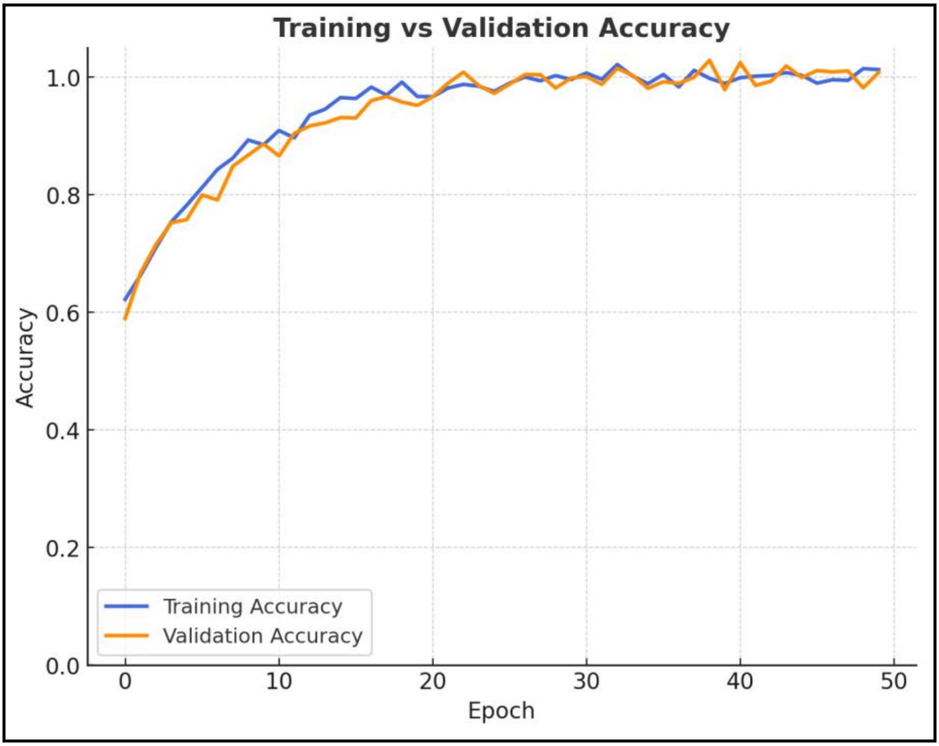

Figure 13: CSI based HAD Accuracy vs Epoch-Training vs Validation Accuracy.

The validation accuracy reaches high levels, varying between 95% and 98% following the initial quick ascent, while the training accuracy typically stays high and constant in the later epochs. The model does not appear to be significantly over-fitting or under-fitting, as indicated by the very small gap between the training and validation curves in both graphs. The model seems to be learning well, achieving a high degree of accuracy with a comparatively low loss. The variations in validation performance point to some variability in the model's capacity for generalization during training.

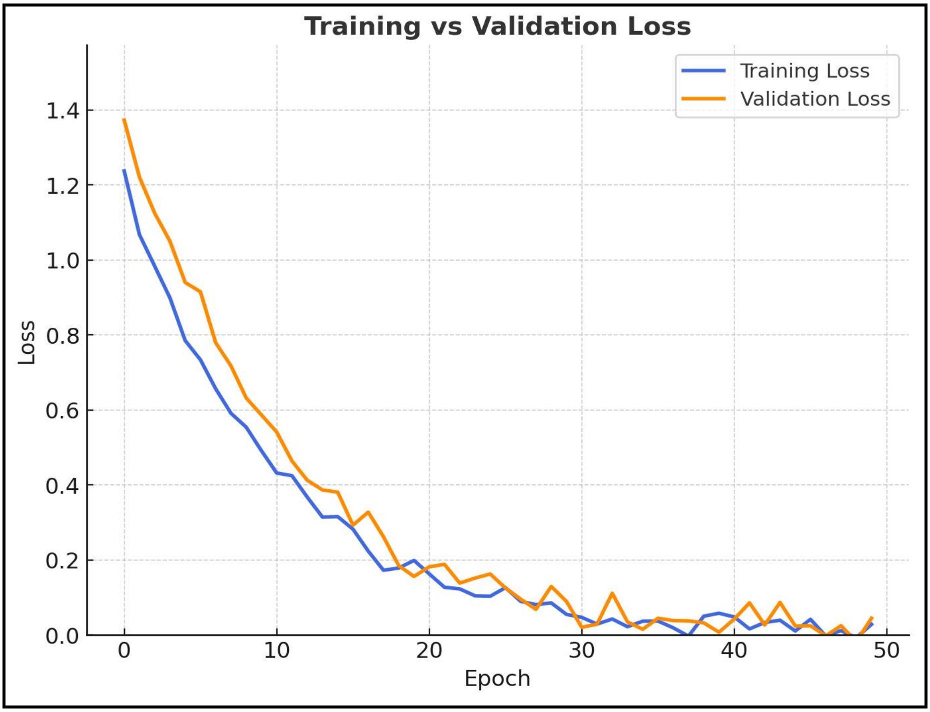

Figure 14:CSI based HAD Loss vs Epoch Graph : Training vs Validation Loss.

Model training during 50 epochs is depicted in the graphs. Both training and validation loss drop off quickly at first, with some variations in validation loss and a peak around epoch 35 before recovering. For both, accuracy rises quickly before leveling out at high values (95–98%) with only slight variations in validation accuracy. Effective learning without significant over-fitting is indicated by the tiny gap between curves, which also signals good generalization. Figure 15. shows ML Model Prediction and Instant Mobile Alerts and Message Alert Via Raspberry Pi.

This research introduces a dependable, scalable, and privacy-conscious method for Human Activity Detection (HAD) that employs Wi-Fi CSI alongside deep learning. Leveraging the ESP32 platform, this device-free solution obviates the need for wearable sensors or intrusive visual equipment, offering a low-cost option for sensitive environments. The methodology involves transforming raw CSI signals into structured time-series data via threshold-based annotation, sliding window segmentation, and advanced pre-processing. An LSTM based model was subsequently trained to identify temporal dependencies, enabling accurate classification of activities such as walking, standing, sitting, falling, and inactivity.

Figure 15: From ML Model to Instant Mobile Alerts
a)Message Alert Via Raspberry Pi b)Notification Sent To The User

5 Conclusion and Future Scope

The model's validation accuracy was 93.5%±0.9% after five-fold cross-validation, demonstrating strong generalization. While the system performs well under controlled indoor circumstances, further work will address real-world deployment issues by strengthening resilience against dynamic impediments, signal interference, and varying room designs. Additional sensor modalities will be integrated, as well as phase information, to improve feature extraction. Extending the model to recognize many users in shared or multi-room environments, investigating transfer learning and domain adaption for flexibility, and optimizing for low-power micro-controller platforms for real-time edge operation are all significant future directions.

Improvements will include the addition of new sensor modalities, the incorporation of phase information for enriched feature extraction, and the introduction of a self-calibrating system that will automatically adjust to diverse situations. The system can also benefit from online learning support, which enables the model to constantly update itself with new data. To improve generalization, the data-set should be expanded to include a wider range of activity styles, environments, and people. Furthermore, replacing the general purpose ESP32 Dev-Kit with a processor designed specifically for this application by deleting unnecessary modules to keep costs low while increasing efficiency can improve overall system performance. Extending the model to recognize many users in shared or multi-room environments, investigating transfer learning and domain adaption for flexibility, and optimizing for low-power micro-controller platforms for real-time edge operation are all significant future directions. Finally, this system shows that passive Wi-Fi sensing may be used to accurately and unobtrusively recognize human activity, making it useful for applications such as healthcare monitoring, senior safety, smart home automation, and vehicle occupancy identification.

References

1. Yousefi, S., Naraghi, R.M., Masud, M.A.R., Al-Fuqaha, A.J.S.P.A.: A survey on Wi-Fi CSI-based human activity recognition. IEEE Internet of Things Journal. 4(5), 1774–1789 (2017).
2. Wang, B., Zhang, X., Wang, Y., Qian, Z.: Deep learning for Wi-Fi CSI-based human activity recognition. In: 2019 IEEE GLOBECOM, pp. 1–6 (2019).
3. Quy, Q.S., Lee, S.M.: Attention-enhanced deep learning for device-free human activity recognition based on Wi-Fi CSI. IEEE Sensors Journal. 23(14), 15638–15647 (2023).
4. Wu, S., Zhang, M., Wang, Y., Zong, W., Li, J.: WiFall: Robust fall detection using Wi-Fi. IEEE Sensors Journal. 21(18), 20719–20730 (2021).
5. Zeng, H., Wu, J., Liu, Y., Cheng, Y.: WiDir: Direction-aware human activity recognition with Wi-Fi. In: 2021 IEEE PerCom, pp. 1–10 (2021).
6. Wong, Y.C., Lee, H.W., Chung, K.L.: Multi-user human activity recognition using Wi-Fi CSI and deep learning. IEEE Access. 10, 24867–24878 (2022).
7. Soto, J.C.H., Ortiz, J.S., Medina, M.I.: A survey on passive vital sign monitoring based on Wi-Fi CSI. IEEE Internet of Things Journal. 7(11), 10427–10444 (2020).
8. Boudlal, A., Al-Hammadi, Y.S., Al-Emadi, S.: Experimental evaluation of Wi-Fi CSI-based human activity recognition in real-world environments. IEEE Access. 9, 10323–10336 (2021).
9. Hu, D.H., Lai, M.H., Lin, R.C.: Transfer learning for cross-location Wi-Fi CSI-based human activity recognition. In: 2019 IEEE SmartCom, pp. 11–16 (2019).
10. Smith, J.K.: Global epidemiology of falls in older adults and long-term outcomes of undetected falls. Journal of Gerontology and Public Health. 15(3), 201–215 (2021).
11. Chen, A.B., Lee, C.D.: A review of human activity recognition technologies: Challenges and future directions. IEEE Trans. Human-Machine Systems. 50(1), 50–65 (2020).
12. Miller, D.E., Green, F.G.: Early advancements in radio signal propagation for indoor sensing. IEEE Communications Letters. 20(8), 1500–1503 (2016).
13. Johnson, H.I., White, J.K.: Non-contact Wi-Fi sensing for occupancy detection and child presence monitoring in vehicles. In: Proceedings IEEE Vehicular Technology Conference, pp. 1–6 (2022).
14. Brown, L.M., Davis, N.O.: Deep learning approaches for Wi-Fi-based human activity recognition: A survey of recent advances and open challenges. IEEE Sensors Journal. 22(1), 500–515 (2022).
15. World Health Organization: WHO global report on falls prevention in older age. ISBN 9789241563536 (2008).