Ensemble Learning based WiFi Sensing using Spatially Distributed TX-RX Links

Nafeez Fahad, Md Touhiduzzaman and Eyuphan Bulut
Department of Computer Science, Virginia Commonwealth University
401 West Main St. Richmond, VA 23284, USA
{fahadn, touhiduzzamm, ebulut}@vcu.edu

***Abstract*—Utilizing fine grained analysis of wireless signals for human activity recognition has gained a lot of traction recently. The unique changes made on the ambient wireless signals by different activities made it possible to recognize these fingerprints through deep learning classification methods. However, most existing approaches fail to fully leverage the rich information available from multiple transmitter-receiver (TX-RX) pairs in a given environment. This study proposes an aggregated weighted ensemble learning method that benefits from spatially distributed TX-RX links to enhance WiFi sensing based activity recognition performance. Our approach utilizes Channel State Information (CSI) data collected from multiple angles and viewpoints, made possible by strategically placed TX-RX pairs, and integrates them through the proposed method. This allows our system to benefit from the complementary strengths of each TX-RX pair, capturing a wider range of signal propagation patterns and environmental factors. We provide an experimental evaluation using datasets of human limb activities collected from six different TX-RX pairs. The results show that the proposed model can achieve a much higher accuracy (i.e., 90%) than the base models and the model that is trained on the combined dataset thanks to the integrated weighted ensemble learning technique and the data from multiple TX-RX pairs.**

***Index Terms*—WiFi sensing, channel state information, ensemble learning, human activity recognition.**

I. Introduction

WiFi signals are being used by nearly all establishments, including restaurants, houses, schools, colleges, institutions, and other significant locations, to provide Internet connectivity. This makes WiFi signals ubiquitously available all around us. Leveraging this availability of ambient WiFi signals together with machine learning models, WiFi sensing aims to recognize motion and activity recognition in the environment [1]–[3]. This technology has applications in different fields such as home security [4], occupancy detection [5]–[7], and health/well-being monitoring of individuals [8].

Most of the WiFi sensing studies depend on the low level Channel State Information (CSI) data of WiFi signals. In order to reach this information, however, a CSI extraction tool and a specialized Network Interface Card (NIC) (e.g., Intel 5300 NIC [9]) is needed at the receiver device (which is usually a computer thus costly). Thus, despite the growing number of studies on WiFi studies, most of them use only one or a few transmitter (TX) and receiver (RX) WiFi devices that collect the CSI data to be used in the corresponding WiFi sensing application. However, through a lightweight solution, a more scalable deployment of TX-RX devices in the environment can be achieved, and the performance of WiFi sensing system can be enhanced.

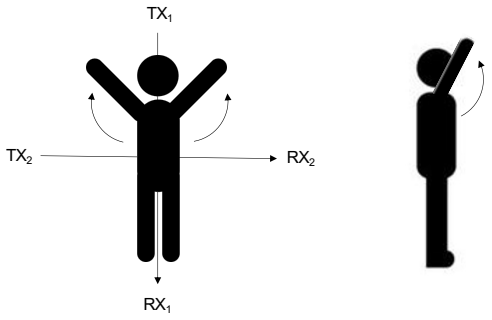

Fig. 1: Two different TX-RX pair deployments with the human performing an activity in the LoS. RX_1 sees the human as on the left, while RX_2 has the view on the right.

Consider the example scenario in Fig. 1, which shows two TX-RX pairs deployed on the left-right and front-back of the person monitored. Note that as signals transmitted from TX_1 propagate towards RX_1, they see the blockage of the human body with the angle seen on the left. For RX_2, however, this becomes like the body posture on the right, which generates a different interaction. With more number of such spatially distributed multiple TX-RX links, some viewing the person in line-of-sight (LoS) and some viewing in non-line-of-sight (NLoS), several angular viewpoints can be provided and the accuracy of the WiFi sensing application can be boosted [10].

While the deployment of multiple TX-RX links will increase the collected CSI data, they should not be just merged and used within a single model. Ensemble learning is a powerful machine learning technique that combines multiple models to improve predictive performance and robustness [11]. For the research works in WiFi sensing domain, deep learning architectures have been proving their worth for a long time, but the idea of combining ensemble learning and deep learning to create deep ensemble architectures has the potential to change the scenario in this context. Ensemble learning is specialized for the situations where we need to handle noisy and complex data [12]. This opens up the horizon for WiFi sensing as the complex and noisy CSI data can benefit from the ensemble model. In the case of multiple TX-RX pairs, each can be treated as an individual model within the ensemble itself. The ensemble model can leverage this

979-8-3315-2096-0/25/$31.00 ©2025 IEEE

diversity of information provided by different TX-RX pairs for more accurate predictions. This can be particularly useful in scenarios where the wireless environment is dynamic and the performance of individual TX-RX pairs changes with time.

To the best of our knowledge, there are a few studies [13]–[15] that study ensemble learning based WiFi sensing systems. However, these studies do not take the benefit of spatially distributed TX-RX pairs in the monitored area; instead, they create different models with the same data and use ensemble learning from their predictions, providing a limited benefit. In this study, we aim to benefit from the ensemble model architecture by constructing several base models trained by data from different TX-RX pairs of a multiple TX-RX-based setup. These help us get the CSI of different angles, heights, and distances and complement the features extracted from each for a better prediction.

The rest of the paper is organized as follows. We first review related work in WiFi sensing, particularly by focusing on the studies that consider ensemble learning in their prediction model in Section II. In Section III, we discuss the preliminaries, including the preprocessing steps applied to the raw collected CSI data. Then, in Section IV, we present the proposed solution elaborating on the developed ensemble learning model. Next, we provide the evaluation results in Section V. Finally, we provide our concluding remarks and discuss on future work in Section VI.

II. Related Work

Wireless sensing based human activity recognition has received growing attention recently and many studies in different application domains have been performed [1]–[3]. Several studies have also explored the recognition of limb and hand movements. For example, Tan et al. [16] investigate four types of limb movements, while Ren et al. in Winect [17], propose a method for tracking 3D human poses using a joint decomposition deep learning model. WiFit [18] system aims to identify and segment exercise types using Doppler displacement theory and SVM classification, and Wi-PT-Hand system [19] aims to recognize hand and finger movements based physical therapy exercises.

Ensemble learning approaches have also been applied in a few wireless sensing studies previously. Cui et al. [13] propose WiAReS, a WiFi sensing based activity recognition system utilizing deep ensemble methods. Liu et al. [14] introduce StackFi, which combines the strengths of CNN and RNN models for CSI data analysis. Similarly, Bernaola et al. [15] employ boosting ensemble methods for counting seated people using WiFi sensing. However, these studies do not consider data from multiple TX-RX links unlike our approach and instead use the same data to develop multiple different models to be used in an ensemble learning approach, offering a limited benefit.

Note that there are other studies exploring human activity recognition with an ensemble model approach by using other types of sensor data. For example, Imanzadeh et al. [20] propose an ensemble model using hybrid deep learning models trained on smartphone sensor data. Zehra et al. [21] present a CNN-based ensemble system for human activity recognition using accelerometer data.

Our approach differs from all these previous studies in several key aspects. First of all, we consider seven limb movements to ensure proper movement of both upper and lower limbs of human body. Besides that, we utilize datasets from multiple TX-RX pairs, ensuring usability across various configurations. Here, we employ a stacking ensemble followed by an aggregation strategy with logistic regression, allowing the optimal combination of base model predictions. Moreover, we propose a weighted ensemble approach based on F1-scores obtained from base models. The proposed approach leverages data from different TX-RX pairs to create diverse base models, capturing a more comprehensive range of signal propagation patterns and environmental factors, and aiming to enhance the robustness of the proposed WiFi sensing system.

III. Preliminaries

WiFi sensing relies on the utilization of Received Signal Strength Indicator (RSSI) or CSI metric from the ambient WiFi signals. Thanks to the rich information carried in CSI data compared to RSSI, most of the studies utilize CSI data.

A. Amplitude and Phase Extraction

CSI is a metric composed of signal amplitude and phase across N subcarrier frequencies used in the process of link adaption. In link adaption, each subcarrier is able to transmit symbols at adaptable data rates and power levels in parallel across each subcarrier [22] to allow for multiple-input multiple-output (MIMO). OFDM transmits shared pilot symbols interleaved within the data frame which can then be used to estimate a shared value for CSI H for the pair of devices as described in the equation:

$$y_i = H_i x_i + \eta_i, \tag{1}$$

where i indicates the subcarrier index, y_i indicates the signal characteristics received, x_i is the actual transmitted signal and η_i is the noise in the signal. The CSI vector H is a complex number with both real and imaginary parts representing the attributes of the received signal. With H_i^r as the real part of H at subcarrier i and H_i^m as the imaginary part, we then compute amplitude ($A_i = \sqrt{(H_i^m)^2 + (H_i^r)^2}$) and phase ($\phi_i = atan2(H_i^m, H_i^r)$) for each subcarrier i.

In this study, we collect the CSI frames using our ESP32 microcontrollers and our CSI extraction tool [23]. Out of the total 64 subcarriers, only 52 of them contain actual CSI data that is not static or zero, and thus, we use only the data of these subcarriers.

B. Preprocessing

Once the amplitude and phase values are extracted from the raw CSI values, next, we do some preprocessing on them. That is, we apply two main techniques that are also considered in previous works [24]–[28]: (i) Hampel filter and (ii) Moving average filter. When applied to CSI amplitude or phase, the

Hampel filter can help eliminate extreme values that may be caused by noise or interference, making the data more reliable for further analysis [1]. The moving average filter can also smooth out short-term fluctuations and highlight longer-term trends or cycles [1]. The sanitation of amplitude values is performed by applying the Hampel filter followed by the moving average filter.

For the phase values, before we apply these filters, we first perform a calibration to avoid the effects of several offsets on the observed phase value. The observed CSI phase of the i-th subcarrier at the receiver can be denoted by [29],

$$\theta_i = \phi_i + \frac{2\pi i}{D}T + \alpha + \epsilon_i, \tag{2}$$

where ϕ_i is true phase value of the i-th subcarrier, $i \in \mathcal{S} = \{-26, ..., -1, 1, ..., 26\}$ is subcarrier index, T is time delay caused by Sampling Time Offset (STO) and Sampling Frequency Offset (SFO), α is phase offset caused by Carrier Frequency Offset (CFO) and Phase-Lock-Loop (PLL), D is OFDM dimension, and ϵ_i is measurement error.

Firstly, we remove time delay and phase offset. STO and SFO are responsible for the time delay [29], [30]. We estimate the mean offset ($\hat{\alpha}$) and slope (Δn) by

$$\hat{\alpha} = \frac{1}{M}\sum_{i \in \mathcal{S}} \theta_i \tag{3}$$

$$\Delta n = \frac{\sum_{i \in \mathcal{S}} i \times (\theta_i - \hat{\alpha})}{2\pi \sum_{i \in \mathcal{S}} i^2}, \tag{4}$$

where $\mathcal{S}$ is the set of available subcarriers, and $M = 52$ is the total number of subcarriers. CFO and PLL are the reasons for phase offset. By eliminating those, we calibrate our phase value. The calibrated phase value ($\hat{\theta}_i$) is then given by,

$$\hat{\theta}_i = \theta_i - 2\pi i \Delta n - \hat{\alpha}. \tag{5}$$

After phase calibration, the Hampel filter and moving average filter are applied to the calibrated phase values, as they are applied to amplitude values.

At the end of these calibration and filtering steps, we obtain 52 amplitude and 52 phase values for each timestamp. We then split our dataset into train, test and validation set. Finally, we perform Principal Component Analysis (PCA) by fitting it to the training set and using them in validation and test sets. Next, we discuss how the proposed learning model is developed using these principal components.

IV. Proposed Method

We propose a weighted ensemble learning based neural network model for the CSI data collected from multiple spatially distributed TX-RX links. The goal of our ensemble learning based approach is to benefit from different datasets obtained from different pairs and combine the predictions of their models in the best way to achieve a better result. An overview of the proposed system and steps are illustrated in Fig. 2.

Let $\mathcal{N}$ denote the number of TX-RX pairs used in the setup and let $\mathcal{M}_i$ denote the neural network model developed for

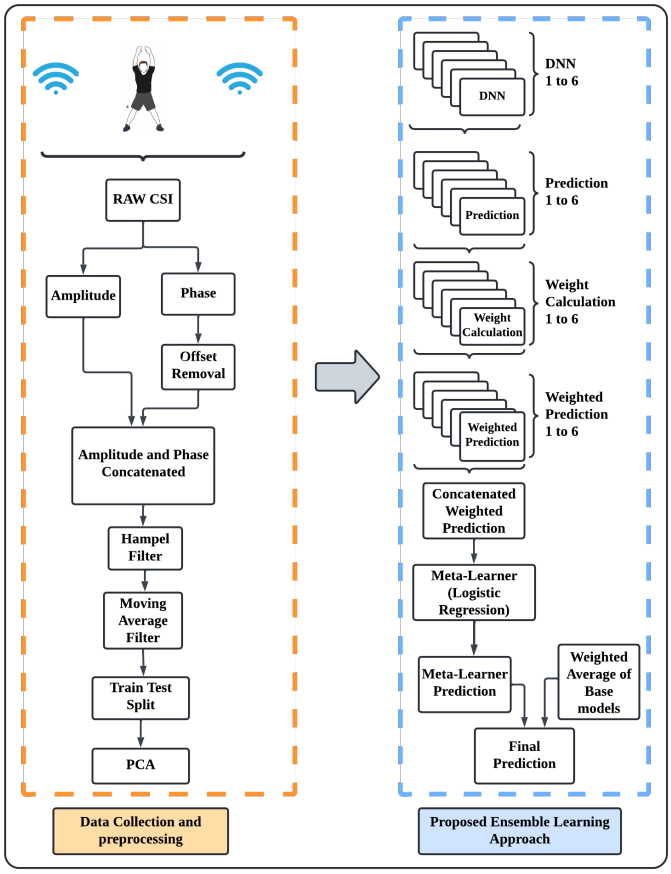

Fig. 2: System Design

the i-th pair. Note that we do not define the architecture of these models; depending on the application needs and the performance of different models, an optimized one can be selected.

Our ensemble model implementation relies on forming of a stacking ensemble and the usage of an aggregation method which takes the predictions of base models as input and learns to combine them optimally. This is usually achieved by using another model, such as logistic regression, to learn the relationship of the stacked list of probabilities of all individual models with the class labels. Since each model's performance on different class predictions can be different, this combining process should be performed carefully for maximum benefit. For each base model, we calculate a weight based on the F1 score. We use softmax to normalize the F1 score because softmax always produces positive values between 0 and 1, and they sum to 1. Additionally, softmax amplifies differences between the input values. This causes larger values to increase further and smaller values to decrease further. As a result, classes with higher F1 scores are assigned relatively larger weights, while classes with lower F1 scores receive smaller weights compared to simple normalization (i.e., dividing by their sum). Each of the $\mathcal{N}$ models may perform differently across various classes. By calculating class-specific weights based on F1 scores, we can harness each model's strengths for

Algorithm 1: Weighted Ensemble Model

Input:
Training data $D_{train_1}, D_{train_2}, ..., D_{train_\mathcal{N}}$,
Test data $D_{test_1}, D_{test_2}, ..., D_{test_\mathcal{N}}$,
Validation data $D_{val_1}, D_{val_2}, ..., D_{val_\mathcal{N}}$,
Number of base models $\mathcal{N}$,
Number of classes C
Output:
Class label for each test instance as determined by the ensemble model

1 **for** $i = 1$ *to* $\mathcal{N}$ **do**
2 Train base model $\mathcal{M}_i$ on D_{train_i};
3 **for** $i = 1$ *to* $\mathcal{N}$ **do**
4 $P_i \leftarrow$ Predictions of $\mathcal{M}_i$ on validation set;
5 $F_i \leftarrow F1$ scores for each class from P_i;
6 $W_i \leftarrow \text{softmax}(F_i)$ // Class-specific weights
7 $X_{\text{agg}} \leftarrow \emptyset$ // Aggregated training data
8 $Y_{\text{agg}} \leftarrow \emptyset$ // Aggregated training labels
9 **for** *each sample* x *in validation set* **do**
10 **for** $i = 1$ *to* $\mathcal{N}$ **do**
11 $p_i \leftarrow \mathcal{M}_i(x)$ // Prediction probabilities
12 $p_i^w \leftarrow p_i W_i$ // Apply weights
13 $X_{\text{agg}} \leftarrow X_{\text{agg}} \cup \{p_1^w, p_2^w, ..., p_M^w\}$;
14 $Y_{\text{agg}} \leftarrow Y_{\text{agg}} \cup \{\text{True label of } x\}$;
15 Train a logistic regression model LR on $(X_{\text{agg}}, Y_{\text{agg}})$;
16 **Function** `Predict_On_Test(`x`)`**:**
17 **for** $i = 1$ *to* $\mathcal{N}$ **do**
18 $p_i \leftarrow \mathcal{M}_i(x)$;
19 $p_i^w \leftarrow p_i W_i$;
20 $p_{\text{agg}} \leftarrow LR(p_1^w, p_2^w, ..., p_N^w)$;
21 $p_{\text{final}} \leftarrow \frac{1}{N+1}(\sum_{i=1}^{N} p_i^w + p_{\text{agg}})$;
22 **return** $\arg\max(p_{final})$;

different classes. Through the application of these weights to individual predictions and combining them, we can determine the final prediction.

Algorithm 1 presents the steps of our weighted ensemble model. The model consists of $\mathcal{N}$ base models with i-th of them trained on the training data from i-th TX-RX pair i.e., $D_{train_1}, D_{train_2}, ..., D_{train_N}$ (lines 1-2). For each base model $\mathcal{M}_i$, we calculate class-specific weights W_i using the softmax of F1 scores obtained from validation set predictions (lines 3-6). This results in a set of weights for each base model, where each weight corresponds to a class. These weights emphasize each model's strengths across different classes. The weights are then applied by multiplying the probability predictions of each base model by their respective weights to obtain the p_i^w (lines 9-12). We then create a training set by aggregating the weighted predictions p_i^w from all base models for each sample (lines 13-15). A logistic regression model LR is trained on this data to learn optimal combination strategies. The final prediction function combines weighted predictions from all base models and the LR prediction (lines 16-22). Here, p_i represents the probability predictions of the i-th base model, p_i^w is the predictions weighted by W_i, and p_{agg} is the LR model's prediction. The final prediction p_{final} is obtained by averaging all weighted base model predictions and the aggregated prediction (LR model's prediction), with the predicted class being the $\arg\max$ of this final probability distribution. In this way, we emphasize the predictions where each model is the most confident and accurate, which allows the ensemble model to leverage the strength of each individual model across different classes.

Note that the ensemble learning approach heavily depends on base models' performances. Thus, errors or biases in the base models could be propagated or even amplified through the ensemble approach. Weights introduced in this process can help mitigate this potential issue. However, it is still possible that if F1 scores used in weight decisions are very close across classes or models, softmax may still amplify minor but possibly insignificant differences.

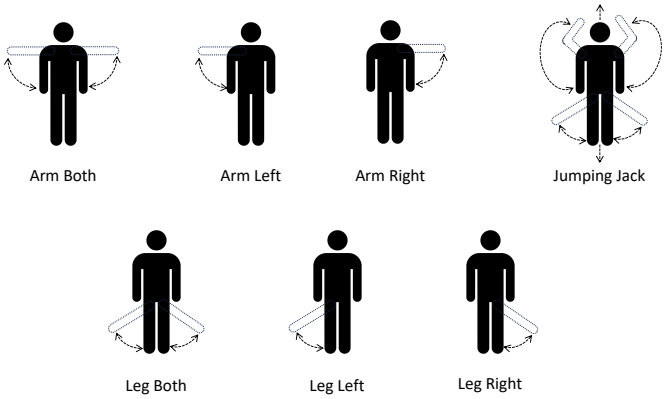

Fig. 3: Performed Limb Activities

TABLE I: TX-RX position and their respective channel

Pair No.	TX-RX Position		Channel
	Height	With respect to the test subject	
1	Eye Level	Front-Back	9
2	Eye Level	Sidewise	9
3	Diagonal	Front-Back	11
4	Diagonal	Sidewise	11
5	Knee Level	Front-Back	6
6	Knee Level	Sidewise	6

V. Evaluation

In this section, we present the evaluation results of the proposed approach with a dataset of human limb activities.

A. Experimental Set up and Data Collection

In order to test the proposed system, we collected WiFi CSI data for 7 different limb activities depicted in Fig. 3. Each activity was performed for 10 seconds, with a 10-second transition period between activities. We repeated the 7 activities for 10 times.

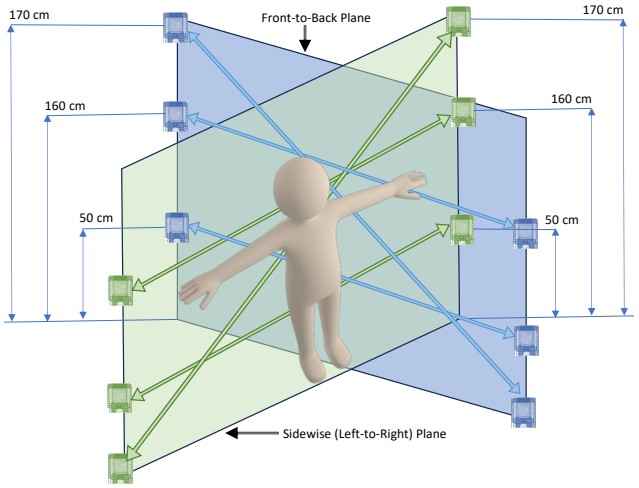

Fig. 4: Position of TX and RX devices

We used our ESP32-CSI-Toolkit [23], [31] to collect CSI, which uses two ESP32 WiFi-enabled microcontrollers for our transmitter and receiver, respectively. The ESP32 devices were set to send and receive CSI at a packet rate of 100Hz. However, due to the slight fluctuations in this rate, we ended up having data samples in the range of 9064 to 9627 for each class. We deployed six pairs of these ESP32 pairs. Among these, three pairs were positioned with TX and RX placed on the left and right side of the subject, while the other three pairs were positioned with TX and RX located in the front and at the back of the subject. Each of the two pairs are also located at the same level. We set up the TX and RX at the eye level, knee level, and diagonally through the test subject's body. The locations of these pairs and the channels they used for packet exchange are given in Table I and in Fig. 4. Note that, the subject is in Line-of-Sight (LOS) of all TX-RX pairs.

For training, we use 60% of the entire dataset collected. 20% of the dataset is considered as validation set and the remaining 20% is used as test set. We also consider sliding windows for the CSI data before feeding them into each model. The window size is set to 100 (i.e., 1 sec of data). We use a sequential Deep Neural Network (DNN) model as our base model for each TX-RX pair's data because of its simplicity. DNN models are known for being lightweight, making their onboard deployment on ESP32s possible for real-time inference [2]. Fundamentally, we create six identical neural network models, each trained on a different subset of the training set from each of the six datasets we have. Each model has three dense layers with ReLU activation, followed by batch normalization and dropout layers. The output layer uses softmax activation for multi-class classification. The aggregation of base model predictions is performed with a logistic regression model that learns from the stacked list of probabilities of all six models. We apply majority voting by combining the predicted probabilities from all base models and the logistic regression, as shown in line 21 of Algorithm 1.

TABLE II: Prediction Results

Model	Precision	Recall	F1 Score	Accuracy
DNN 1 (Pair 1)	0.84	0.84	0.83	0.84
DNN 2 (Pair 2)	0.61	0.53	0.47	0.53
DNN 3 (Pair 3)	0.39	0.32	0.28	0.32
DNN 4 (Pair 4)	0.37	0.30	0.22	0.31
DNN 5 (Pair 5)	0.49	0.49	0.47	0.49
DNN 6 (Pair 6)	0.33	0.37	0.31	0.38
Combined Dataset	0.72	0.68	0.66	0.70
CNN and RNN Ensemble to the Combined Dataset [14]	0.84	0.82	0.79	0.80
Proposed Weighted Ensemble Model	0.91	0.90	0.89	0.90

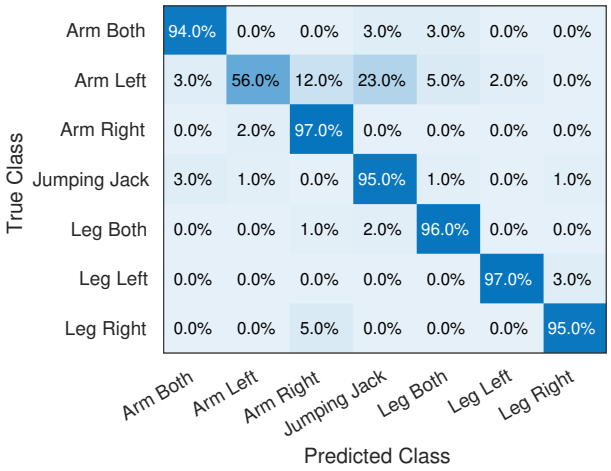

Fig. 5: *Confusion Matrix for Weighted Ensemble Model*

B. Results

Table II shows the detailed performance comparison of six base models and the proposed weighted ensemble model. The proposed model can achieve an accuracy of 90%, which is 42% better accuracy than the average of all base models and 6% better than the best base model. We also present the results from the combined dataset of all six pairs of TX and RX which was aligned based on their class at each repetition. A DNN model using the combined dataset (with aforementioned training/validation/test splits) can only produce 70% accuracy. When we use an ensemble model with CNN and RNN models developed separately on the combined dataset, as in [14], the accuracy increases to 80%, but it is still lower than the accuracy of the proposed approach. This is because it does not truly benefit from the diversity of different datasets as it is performed in our weighted ensemble approach, which achieves the highest performance among all.

Looking at the confusion matrix for the weighted ensemble model in Fig. 5, we notice that the *Arm-Left* class is the most confused class in our case. This could potentially be due to some issues while collecting data as base model predictions show high confusion for this class as well. From base model results shown in Table II, we also understand that the location of first TX-RX pair (i.e., eye level in front-back plane) that uses channel 9 for data transmission is better than other pairs' locations as it gives the highest among all base models.

VI. Conclusion

In this work, we have studied ensemble learning based WiFi sensing system that utilizes datasets from multiple TX-RX pairs spatially distributed in the monitored area. We propose a weighted approach based on F1-score, where we use ensemble learning to the data obtained from multiple TX-RX pairs. On top of that, we use a logistic regression based aggregation method, which uses the predictions of base models as input and learns to combine them optimally.

The experimental results obtained on a human limb activity recognition scenario show that the proposed weighted ensemble based approach achieves 90% accuracy, which is better than not only the performances of individual base models but also the performance of previous ensemble learning based approaches that use different models trained on the combined dataset.

In our future work, we plan to dynamically select the most informative TX-RX pairs for different activities or environments, potentially improving efficiency and accuracy. We will also explore the model's transferability across different environments or setups [32], potentially reducing the need for extensive data collection in new deployments.

References

[1] Y. Ma, G. Zhou, and S. Wang, "WiFi Sensing with Channel State Information: A Survey," *ACM Computing Surveys*, vol. 52, no. 3, pp. 46:1–46:36, 2019.

[2] S. M. Hernandez and E. Bulut, "WiFi Sensing on the Edge: Signal Processing Techniques and Challenges for Real-World Systems," *IEEE Commun. Surv. Tutorials*, vol. 25, no. 1, pp. 46–76, 2023.

[3] C. Chen, G. Zhou, and Y. Lin, "Cross-Domain WiFi Sensing with Channel State Information: A Survey," *ACM Computing Surveys*, vol. 55, no. 11, pp. 231:1–231:37, 2023.

[4] S. Zhang, R. H. Venkatnarayan, and M. Shahzad, "A WiFi-based Home Security System," in *17th IEEE Inter. Conf. on Mobile Ad Hoc and Sensor Systems (MASS), India, Dec. 10-13*, 2020, pp. 129–137.

[5] F. Chu, C. Chiu, A. Hsiao, K. Feng, and P. Tseng, "WiFi CSI-Based Device-free Multi-room Presence Detection using Conditional Recurrent Network," in *93rd IEEE Vehicular Technology Conference, VTC Spring 2021, Helsinki, Finland, April 25-28*, 2021, pp. 1–5.

[6] S. M. Hernandez and E. Bulut, "Performing WiFi Sensing with Off-the-shelf Smartphones," in *IEEE International Conference on Pervasive Computing and Communications Workshops, PerCom Workshops, Austin, TX, USA, March 23-27*, 2020, pp. 1–3.

[7] S. M. Hernandez and E. Bulut, "Adversarial Occupancy Monitoring using One-Sided Through-Wall WiFi Sensing," in *IEEE International Conference on Communications (ICC), Montreal, QC, Canada, June 14-23*, 2021, pp. 1–6.

[8] B. Yu, Y. Wang, K. Niu, Y. Zeng, T. Gu, L. Wang, C. Guan, and D. Zhang, "WiFi-Sleep: Sleep Stage Monitoring Using Commodity Wi-Fi Devices," *IEEE Internet Things Journal*, vol. 8, no. 18, pp. 13 900–13 913, 2021.

[9] D. Halperin, W. Hu, A. Sheth, and D. Wetherall, "Tool Release: Gathering 802.11n Traces with Channel State Information," *ACM SIGCOMM CCR*, vol. 41, no. 1, p. 53, Jan. 2011.

[10] "How Wi-Fi sensing became usable tech — MIT Technology Review," https://www.technologyreview.com/2024/02/27/1088154/wifi-sensing-tracking-movements/, (Accessed on 07/16/2024).

[11] M. Ganaie, M. Hu, A. Malik, M. Tanveer, and P. Suganthan, "Ensemble deep learning: A review," *Engineering Applications of Artificial Intelligence*, vol. 115, p. 105151, 2022.

[12] X. Dong, Z. Yu, W. Cao, Y. Shi, and Q. Ma, "A survey on ensemble learning," *Frontiers of Computer Science*, vol. 14, no. 2, p. 241–258, Aug. 2019.

[13] W. Cui, B. Li, L. Zhang, and Z. Chen, "Device-free single-user activity recognition using diversified deep ensemble learning," *Applied Soft Computing*, vol. 102, p. 107066, 2021.

[14] J. Liu, "StackFi: An Ensemble Learning-Based Model for WiFi Sensing Classification Tasks," in *IEEE 4th Inter. Conference on Consumer Electronics and Computer Engineering (ICCECE)*, 2024, pp. 265–269.

[15] J. R. M. Bernaola, I. Sobrón, J. D. Ser, I. Landa, I. Eizmendi, and M. Vélez, "Ensemble Learning for Seated People Counting using WiFi Signals: Performance Study and Transferability Assessment," in *IEEE Globecom Workshops, Madrid, Spain, December 7-11*, 2021, pp. 1–6.

[16] B. Tan, A. Burrows, R. J. Piechocki, I. Craddock, Q. Chen, K. Woodbridge, and K. Chetty, "Wi-Fi based passive human motion sensing for in-home healthcare applications," in *2nd IEEE World Forum on Internet of Things, WF-IoT 2015, Milan, Italy, Dec. 14-16*, 2015, pp. 609–614.

[17] Y. Ren, Z. Wang, S. Tan, Y. Chen, and J. Yang, "Winect: 3D Human Pose Tracking for Free-form Activity Using Commodity WiFi," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 5, no. 4, pp. 176:1–176:29, 2021.

[18] S. Li, X. Li, Q. Lv, G. Tian, and D. Zhang, "WiFit: Ubiquitous Bodyweight Exercise Monitoring with Commodity Wi-Fi Devices," in *IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation*, 2018, pp. 530–537.

[19] M. Touhiduzzaman, S. M. Hernandez, P. E. Pidcoe, and E. Bulut, "Wi-PT-Hand: Wireless Sensing based Low-cost Physical Rehabilitation Tracking for Hand Movements," *ACM Transactions on Computing for Healthcare*, 2024.

[20] S. Imanzadeh, J. Tanha, and M. Jalili, "Ensemble of deep learning techniques to human activity recognition using smart phone signals," *Multimedia Tools and Applications*, 2024.

[21] N. Zehra, S. H. Azeem, and M. Farhan, "Human activity recognition through ensemble learning of multiple convolutional neural networks," in *55th Annual Conference on Information Sciences and Systems (CISS)*, 2021, pp. 1–5.

[22] T. Hwang, C. Yang, G. Wu, S. Li, and G. Y. Li, "OFDM and Its Wireless Applications: A Survey," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 4, pp. 1673–1694, May 2009.

[23] S. M. Hernandez, "ESP32 CSI Tool," 2023. [Online]. Available: https://stevenmhernandez.github.io/ESP32-CSI-Tool/

[24] H. Zhu, F. Xiao, L. Sun, R. Wang, and P. Yang, "R-TTWD: Robust Device-Free Through-The-Wall Detection of Moving Human With WiFi," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 5, pp. 1090–1103, 2017.

[25] X. Zheng, J. Wang, L. Shangguan, Z. Zhou, and Y. Liu, "Design and Implementation of a CSI-Based Ubiquitous Smoking Detection System," *IEEE/ACM Trans. on Networking*, vol. 25, no. 6, pp. 3781–3793, 2017.

[26] X. Wang, C. Yang, and S. Mao, "PhaseBeat: Exploiting CSI Phase Data for Vital Sign Monitoring with Commodity WiFi Devices," in *IEEE 37th Inter. Conf. on Distributed Computing Systems (ICDCS)*, 2017, pp. 1230–1239.

[27] J. Liu, Y. Wang, Y. Chen, J. Yang, X. Chen, and J. Cheng, "Tracking Vital Signs During Sleep Leveraging Off-the-shelf WiFi," in *Proc. of the 16th ACM International Symposium on Mobile Ad Hoc Networking and Computing (Mobihoc)*, 2015, p. 267–276.

[28] N. Yu, W. Wang, A. X. Liu, and L. Kong, "QGesture: Quantifying Gesture Distance and Direction with WiFi Signals," *Proc. ACM Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 2, no. 1, March 2018.

[29] C.-L. Chen, C.-H. Ko, S.-H. Wu, H.-S. Tseng, and R. Y. Chang, "Device-Free Target Following with Deep Spatial and Temporal Structures of CSI," *Journal of Signal Processing Systems*, vol. 95, no. 11, pp. 1327–1340, 2023.

[30] X. Wang, L. Gao, and S. Mao, "CSI Phase Fingerprinting for Indoor Localization With a Deep Learning Approach," *IEEE Internet of Things Journal*, vol. 3, no. 6, pp. 1113–1123, 2016.

[31] S. M. Hernandez and E. Bulut, "Lightweight and Standalone IoT based WiFi Sensing for Active Repositioning and Mobility," in *21st International Symposium on "A World of Wireless, Mobile and Multimedia Networks" (WoWMoM)*, Cork, Ireland, Jun. 2020.

[32] S. M. Hernandez and E. Bulut, "WiFederated: Scalable WiFi Sensing Using Edge-Based Federated Learning," *IEEE Internet of Things Journal*, vol. 9, no. 14, pp. 12 628–12 640, 2022.