MilliFit: Millimeter-Wave Wireless Sensing Based At-Home Exercise Classification

Edward M Sitar IV; Sanjib Sur
Department of Computer Science and Engineering; University of South Carolina, Columbia, USA
esitar@email.sc.edu; sur@cse.sc.edu

***Abstract*—The proliferation of smart, ubiquitous devices has inspired many researchers to develop at-home personal documentation systems. One application of such systems is at-home exercise monitoring, which is important for remote healthcare and fitness regimens. This work explores a millimeter-wave (mmWave) wireless sensing based at-home exercise monitoring using commodity devices. We leverage the mmWave signals reflected off a person exercising and design a deep-learning network that uses a combination of CNN and LSTM to classify the activities. We evaluate the performance of our classifier extensively, using several input signal representations.**

***Index Terms*—Wireless sensing; Millimeter-wave; Exercise classification; CNN-LSTM.**

I. Introduction

With advances in deep learning frameworks, at-home personal documentation is an increasingly researched topic in the computer vision community. The main objectives of such documentation are fitness monitoring and rehabilitative care. Since the start of the COVID-19 pandemic, there has been a growing demand for telemedicine, which allows medical professionals to continuously monitor their patients and provide meaningful feedback remotely for physical therapy and injury recovery. Furthermore, commuting to in-person appointments is highly time consuming and can be cumbersome, especially for those with physical health conditions or injuries. This motivates the development of remote Human Activity Recognition (HAR) systems, which can help address the challenges and inconveniences with current healthcare solutions.

Vision-based solutions, using optical cameras or LiDAR, present a reliable opportunity for activity monitoring, detection, and classification. But LiDARs are expensive, and any true color based system is unlikely to be a popular choice for activity documentation since they require extensive processing to extract the human silhouette from every frame of video to interpret the activity, and they are heavily reliant on good lighting conditions and numerous viewing angles to overcome occlusions. Many other works have proposed the use of wearable sensors at specific body positions on the trainee or embedded in the workout equipment. Two major problems with this approach are cost and comfort, as this method would require a large number of sensors which becomes costly. Moreover, wearing sensors is uncomfortable for a person, especially during physical activity, and if the user forgets to wear the sensors, no data can be collected for analysis. So, wireless sensing is one of the useful mediums for HAR systems deployment at-home.

While low-frequency RF-based systems, such as Wi-Fi, can capture human motion, they are unable to detect small enough movements to provide detailed activity feedback. Millimeter-wave (mmWave) signals present a unique solution to this problem, as the small wavelength (1-10 mm) and high bandwidth make mmWave well-suited for the fine-grained tracking of human body motion. Exploiting the ubiquity of mmWave devices present a feasible opportunity for at-home personal documentation. Past research works have tried to produce human silhouettes or skeletons from mmWave signals, however, similar to vision-based approaches, these may raise privacy concerns for many users.

We propose *MilliFit* to enable the HAR systems at-home for a set of exercise classification. Instead of recording video, or estimating the silhouette of an individual performing activity, *MilliFit* classifies the activity using only mmWave reflected signals, which embeds useful information about the activity being performed. Moreover, by analyzing the successive frames, we can extract the temporal features distinctive to the routine to classify the activity. *MilliFit* first removes the effect of background clutter by utilizing the fact that the background is static while the trainee performing the exercise makes continuous movement. The sequence of frames with the trainee performing the activity is acquired, and the envelope on the received signal, which embeds the information of the activity, is extracted from the acquired frames. *Then*, the received signal is used for data augmentation, where Gaussian noise is added to the signal before it is further processed. *Finally*, *MilliFit* builds a deep learning framework to classify the activity type.

We implement *MilliFit* on a Commercial-Off-The-Shelf (COTS) device, which operates at 77–81 GHz. The device has 3 transmitting and 4 receiving antennas. During the data collection process, a co-located depth camera captures ground-truth depth images, which are used to isolate the range that the user is performing activity and label the exercise type. *MilliFit* is evaluated on an extensive dataset collected from one individual performing 18 distinct activities, with a combination of static poses and dynamic exercises, and the experimental results demonstrate classification accuracy upwards of 90% with high precision and recall.

II. Background

FMCW Primer: Traditionally, mmWave devices for sensing use Frequency Modulated Continuous Wave (FMCW) signals to extract meaningful information about the target

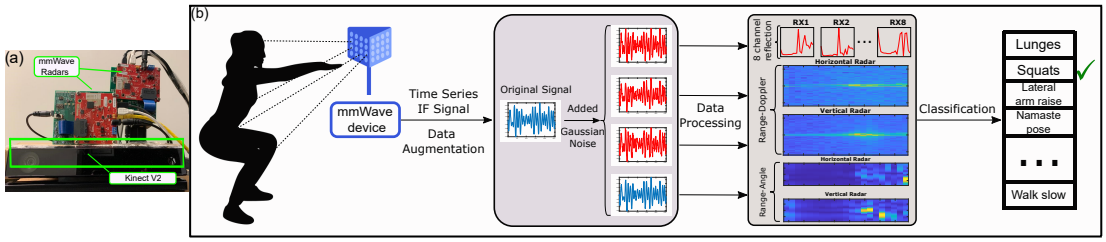

Figure 1: (a) Data collection setup with two mmWave transceivers and Kinect v2; (b) System pipeline for *MilliFit*.

scene. By employing multiple antennas in horizontal and vertical planes on the device, the information of targets can be obtained in the entire azimuth and elevation field-of-view of the transceiver. Since various parts of the human body exist at a different depth, azimuth, and elevation *w.r.t.* the device, information about the body parts and their motion can be extracted by capturing the reflected signals over time. *MilliFit* leverages these reflected signals from multiple antennas on the device to classify the exercise type.

III. *MilliFit* Design

Figure 1 shows *MilliFit*'s overall structure. At a high level, *MilliFit* takes the following steps: First, using our data collection setup described in section IV-A, we collect mmWave reflection data of a user performing activity. Then, the data is sent to a host PC, where it is processed, as described in section III-A. The processed data is then fed into a classification network, which outputs the exercise type with the highest probability as its prediction.

A. Data Processing

Each data sample involves users performing exercises or poses for approximately 11 seconds, giving us a total of 279 frames per sample. We ask the user to remain outside of the device's FoV during the first frames of data acquisition, so the environment only contains the static background, which helps us determine the max range of the user. The mmWave transceiver samples at a much higher frame rate than the Kinect's depth sensor, so we take the median over all chirps within a frame to keep the number of frames consistent. Because the exercises do not begin at the same time that data acquisition starts, we discard the first 69 frames of mmWave data for all exercises, leaving us with a total of 210 frames per sample. All our data have the same number of frames, which is a choice we made to simplify data collection and overcome memory shortcomings of the Kinect. However, our design allows for sequence input of variable length, making *MilliFit* adaptable to real-world scenarios where it may be required to monitor activities for varying duration.

To capture the motion involved with exercises, we use several data processing techniques. First, we apply a 256-point 1-D range FFT directly on the IF signal of each frame and receive antenna independently to acquire the reflection profile, which encodes the strength of reflectors at various ranges relative to the radar. Naively subtracting the signal amplitude for each range bin in consecutive frames intuitively will minimize reflections from static objects and capture frame-to-frame changes associated with exercise movement, however, such a representation has poor performance and velocity is better-suited for capturing movement. Thus, we apply a 2-D FFT to the IF signals to obtain a range-doppler response, which encodes range and velocity information of objects in the FoV of the radar. Specifically, we apply the 2-D FFT to each of the temporal frames and receive antennas independently and concatenate them together, resulting in a 256 x 128 x 2 x 210 range-doppler response for every data sample, where the dimensions represent range, velocity, the two transceivers, and time, respectively. Finally, we obtain a range-angle response by leveraging information about the distance between Rx antennas and the difference in phase of the received signal at different Rx antennas. To acquire the range-angle response, we apply a 256 point 1-D FFT on the IF signal on each frame and receive antenna independently to acquire the range information, and then apply another FFT along the set of receive antennas. The size of the resulting range-angle matrix is 256 x 256 x 2 x 210 for every data sample, where the first two dimensions represent range and angular information, and the third representing the horizontal and vertical transceivers. We reshape the 1-D FFT range data to have the same number of dimensions as the range-doppler and range-angle data, so we can use the same network architecture for evaluation. Specifically, we arrange the data into 256 x 4 x 2 x 210, where we consider the 4 receiving antennas from each mmWave devices as another spatial dimension, with the third dimension of size 2 representing the horizontal and vertical channels. For these three data representations (1-D FFT reflection, range-doppler, and range-angle), we apply a Hanning Window to the time series before any processing, which helps reduce the spectral leakage inherent to the FFT. We aim to provide an analysis of the performance when using each of these data representations for training a HAR classification network.

The mmWave transceiver captures reflections from a range greater than the fixed range of our user performing activity, so we can use the Kinect's depth data to remove reflections from ranges that are irrelevant to our task. We iterate over all Kinect frames in our dataset and do a pixel-wise subtraction of the depth values of each frame with respect to the first frame of that sample. Due to our assumption that the only dynamic object in the environment is an individual performing exercises, subtracting the first frame - which intentionally only contains the static background - will leave us with only the silhouette of our subject. Then, we iterate over every frame

of the processed Kinect data and determine the user's max distance from the Kinect while performing the exercises was 4 meters. All three data representations use the same range axis of length 256, so we apply this range pruning to all processed data by finding the frequency bin corresponding to a range of 4 meters. With a chirp slope of 29.9820 MHz/μs we calculate the IF frequency corresponding to a 4 meter range to be 800273.51 Hz, meaning index 21 of the range axis represents reflections from 4 meters. Therefore, we prune our data, only considering the first 21 indices along the range dimension for each of the three data representations.

Similar to range pruning, we prune angular information. The spacing of the Rx antennas of the radar is exactly λ / 2, meaning we have a max angular FoV of $\pm 90°$. All the data we have collected is performed directly in front of the radar, so much of the angular FoV does not contain human activity and is therefore not useful to the HAR task. We prune the angular FoV, disregarding 30° on both edges of the FoV. This gives us a total angular FoV of 120° in both the horizontal and vertical mmWave transceivers after pruning.

B. Data Augmentation

The data collection process for these experiments is time consuming, making it challenging to acquire a sufficient dataset for training deep learning classification models. Increasing the dataset used to train a deep learning model can help improve its performance. This has been shown time and time again, including in our recent poster [1], which contains preliminary results from this work showing that a similar model has improved classification performance after increasing the number of training samples. Thus, we have expanded our dataset using a basic data augmentation technique, adding random noise to the data. Noise in radar data is largely unavoidable in the real world, so adding additional noise to the training data will make our classification model more robust to noise that it may face in deployment. We take the following steps for data augmentation. *First*, for every data sample, we generate three matrices with random values derived from a complex normal distribution with a mean 0 and covariance 0.1, equal in size to the time series data acquired from the mmWave radars before any processing. The radar data is then normalized to have a mean of 0 and a variance of 1. *Next*, we apply an element-wise addition with the original time series and each of the noise matrices, leaving us with the original time series along with 3 new samples with added noise. *Then*, we apply the data processing techniques described in section III-A to acquire the range-FFT reflection, as well as the range-doppler and range-angle responses. Due to the limited availability of mmWave human activity datasets online, we plan to release our dataset to the to public.

C. Classification

Our system classifies unseen mmWave data as one of the 18 classes. We test the same architecture with the different input data representations: 1-D FFT range data, range-angle data, and range-doppler data. We take a data-driven supervised learning approach, where we train a customized Convolutional Neural Network (CNN) with Long Short-Term Memory (LSTM) network with labeled examples of each exercise. The objective of the CNN is to extract spatial features from the input data using several sliding convolutional filters along the spatial dimensions. Convolution layers have been shown to be very effective in extracting features from data, and are more efficient in terms of memory and time than the matrix multiplications associated with fully connected layers. Because our data consists of sequences of frames of mmWave data, our network needs to learn long-term dependencies across the duration of each exercise. Recurrent networks are designed for precisely this purpose. LSTM has been shown to be among the most successful of the recurrent network variants in learning long-term dependencies. Additionally, LSTMs accept input of variable size, making this network design more suitable for real-world scenarios where the length of the input data, *i.e.* the duration of exercises will not be fixed.

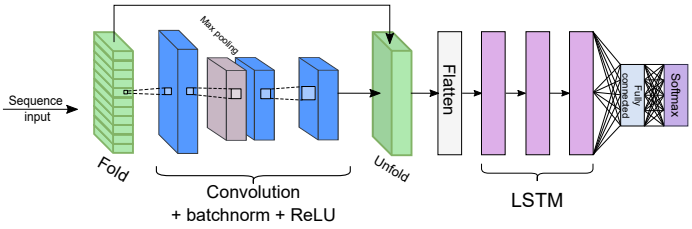

Figure 2: Classification network architecture for *MilliFit*.

At a high level, the classification network takes the following steps to predict the exercise. First, the network folds each input into a sequence structure, and the convolution layers extract spatial features resulting in a feature map for every frame. Then, the feature map is flattened to acquire a feature vector. The feature vectors are then passed to a series of LSTM layers, which serve the purpose of learning the temporal variations across the duration of the exercise. Following the LSTM layers, we use one fully connected layer followed by a softmax to assign a prediction probability for each of the 18 classes. The network finally outputs the class with the highest probability as its predicted class. Our network architecture for classification is shown in Figure 2. Table I summarizes the different network parameters.

We use cross-entropy for the loss function, which is simply the negative log probability the network predicts each data sample as its true class label summed over all samples. The equation for cross-entropy loss for multi-label classification is: $L_{CE} = -\frac{1}{N}\sum_{n=1}^{N}\sum_{c=1}^{C} t_{n,c} ln(\hat{y}_{n,c})$.

IV. Implementation

A. Data Collection

We collect mmWave data using two TI-IWR1443-BOOST transceivers. Each of the mmWave transceivers has 3 Tx and 4 Rx antennas, but we only use one transmitter for each, giving a total of 8 channels. We also use a co-located Microsoft Kinect v2 for ground-truth depth data collection. We refer readers to [2] for more details on the data collection setup. We collect data from one individual performing a total of

Table I: Classification network parameters for different inputs. C2D: Convolution 2D Layer; MP2D: Max Pooling 2D Layer; LSTM: Long Short-Term Memory Layer; FC: Fully Connected Layer.

	C2D1 (#Filters, Filter Size)	MP2D (Size)	C2D2 (#Filters, Filter Size)	Dropout	C2D3 (#Filters, Filter Size)	LSTM1 (#Hidden Units)	Dropout	LSTM2 (#Hidden Units)	Dropout	LSTM3 (#Hidden Units)	FC (#Units)
Range-FFT	(16, 3x3)	2x2	(16, 3x3)	0.35	(32, 3x3)	164	0.3	128	0.25	64	18
Range-Doppler	(16, 3x3)	2x2	(32, 3x3)	0.35	(32, 3x3)	128	0.35	128	0.2	64	18
Range-Angle	(16, 3x3)	2x2	(16, 3x3)	0.35	(32, 3x3)	128	0.3	128	0.25	64	18

18 distinct activities, with a combination of exercises with dynamic movement, and static poses (see [2] for more details on the dataset). We use approximately 30 samples per class, with a total number of samples for classification training of 517. Additionally, we augment our dataset by adding Gaussian noise to existing data, as described in section III-B, bringing the total number of samples to 2068. We assume the environment to be static, and the only moving object in the FoV is an individual doing exercises.

B. Classification Network Training

We test our classification network architectures with three different inputs: range-FFT, range-doppler, and range-angle. Additionally, we evaluate the impact of data augmentation, by examining the classification performance when training both on only the real dataset that we have collected, as well as on the augmented dataset. The hyperparameters, like number of epochs, are adjusted for networks with different inputs to ensure the network converges. We implement, train, and evaluate our classification networks in MATLAB. We found that the networks perform best when training using the ADAM optimizer with a constant learning rate of 0.001. An L2 regularization term is added to the loss function during training to help prevent overfitting. Given the vector of weights, w, the loss function with regularization is: $L_{CE}^{R} = L_{CE} + \lambda\Omega(w)$, where $\Omega(w) = ((1/2)w^{T}w)$ is the regularization function. We find that a λ of 2.5×10^{-5} has the best performance. To further prevent overfitting, we add dropout between some layers during network training, as specified in Table I.

V. System Evaluation

A. Evaluation of classification

To evaluate our classification network, we find the classification accuracy, specificity, precision, recall, F1-score, and Matthews Correlation Coefficient (MCC). These metrics are used to evaluate the network performance on all 18 classes, and on the dynamic and static activities independently, to give a better understanding of the performance in these different cases. Precision, recall, specificity, F1-score, and MCC are designed for binary classification scenarios, so they are calculated for each class individually by considering a binary scenario for each class (i.e. either a member or not a member of the class), and then we find their micro and macro averages across the different classes.

For training and evaluating our classification network, we partition our data into three disjoint subsets each time we train the network. Firstly, we take 20% of the data to use for post-training evaluation. Then, the remaining 80% of the data is divided with a ratio of 80/20 for training and validation data, respectively, giving us a total of 332 training samples for the unagumented case and 1324 training samples for the augmented case. All data partitions are stratified to ensure each of the 18 classes have approximately the same number of examples in each subset. We train each network multiple times and find the mean classification accuracy across the trials. Figure 3 shows the results for the classification accuracy for each of the networks with the augmented dataset. Table II shows the micro and macro average of the remaining metrics across the trials for each of the classification networks when trained on the augmented dataset.

The network with the range-doppler input outperforms the range-FFT and range-angle; this is the case when considering all 18 classes, and when considering dynamic and static classes individually. We find that the performance is consistently better among the dynamic exercises in comparison to the static poses. For the static poses, LSTM layers are ineffective, as there is no significant temporal change in the data to learn, and spatial features learned from the convolution layers are hampered by the specular reflectivity, making it difficult to differentiate between poses that are similar.

Impact of data augmentation: Training on samples with injected noise makes the model more robust and generalizable. We find significant increases in all classification metrics among both static and dynamic activities when training with the augmented compared to the unaugmented dataset. Specifically, among all 18 classes, the MCC has a 15.79% increase in the range-FFT network, a 19.03% increase for range-doppler, and a 34.42% increase in range-angle network, with similar improvements manifesting in the other metrics. This gives us confidence that the augmented dataset did indeed improve the classification performance.

VI. Related Works

Vision-based methods use optical cameras or technologies like LiDAR to capture videos of humans performing activities. Vision-based approaches require the human silhouette to be segmented from the image to be further interpreted. Works such as [3], [4] apply temporal optical flow and temporal smoothing techniques to extract meaningful features capturing the motion across frames. Such methods are only effective under good lighting conditions and without the presence of occlusions blocking the view of an individual performing activity. Additionally, having HAR-dedicated cameras in every room that you wish to detect activity - likely with a need for multiple viewing angles - is extremely costly. Vision-based methods also have privacy concerns, as many users will not want a camera recording them in their homes.

Wearable sensor-based approaches require users to wear one or more sensors on their body while performing ac-

Table II: Micro and macro averages of the classification metrics for each network trained on the augmented datasets.

			Specificity(%)		Recall(%)		Precision(%)		F1-Score(%)		Matthews Correlation Coefficient		Accuracy(%)
			Micro Avg.	Macro Avg.	Micro Avg.	Macro Avg.	Micro Avg.	Marco Avg.	Micro Avg.	Macro Avg.	Micro Avg.	Macro Avg.	Avg.
Augmented	Range-FFT	All	99.64	99.64	93.95	93.88	93.95	94.82	93.95	93.92	0.9359	0.9331	90.69
		Static	99.69	99.69	92.78	92.76	96.96	95.05	93.60	93.50	0.9324	0.9336	90.11
		Dynamic	99.61	99.61	94.85	94.78	93.58	94.64	94.20	94.26	0.9386	0.9416	91.14
	Range-Doppler	All	99.85	99.85	97.46	97.45	97.46	97.66	95.64	97.42	0.9731	0.9735	94.84
		Static	99.67	99.67	94.39	94.39	94.34	94.87	94.34	94.34	0.9401	0.9417	89.29
		Dynamic	99.99	99.99	99.89	99.90	99.89	99.89	97.45	97.42	0.9989	0.9989	99.16
	Range-Angle	All	99.78	99.78	96.29	96.29	96.29	96.41	96.29	96.24	0.9607	0.9609	96.13
		Static	99.63	99.63	94.31	94.40	93.62	93.91	93.96	93.94	0.9361	0.9371	93.66
		Dynamic	99.90	99.91	97.84	97.81	98.40	98.42	98.12	98.08	0.9801	0.9799	98.05

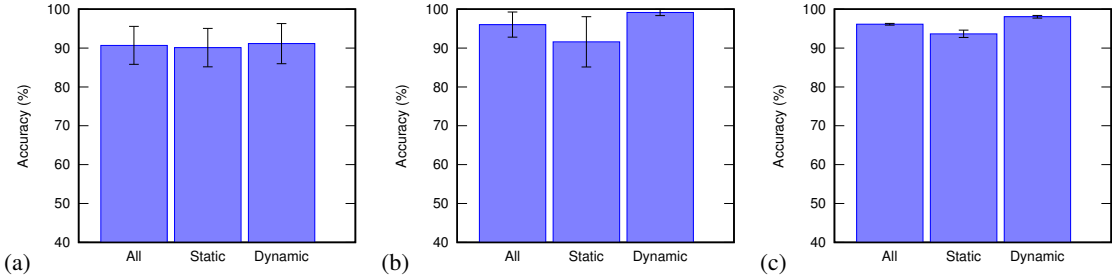

Figure 3: Classification accuracy when trained on the augmented dataset. (a) Range-FFT; (b) Range-Doppler; (c) Range-Angle.

tivities. These approaches leverage the inertial measurement units (IMUs) within the wearable devices to get position, orientation, and acceleration information which are then used to predict the human pose and activity [5]. However, due to the limited information provided by wearable sensors, many researchers have fused the wearables with other technologies, like optical cameras [6], [7], or other sensors embedded in workout equipment such as dumbbells [8]. Wearables are expensive and require users to constantly wear them, which may make users uncomfortable, and in the case where users forgets to wear the sensors, no data can be collected.

In comparison with the aforementioned approaches, Wi-Fi and radar-based approaches have the ability to detect movement in low-light environments and in the presence of occlusions, making them more robust to environmental conditions, and less costly than other solutions, as they do not require many viewing angles to overcome occlusions. Wi-Fi activity monitoring, in general, uses Channel State Information (CSI) to infer the dynamics of human activity. Some works, such as [9], [10], [11], have used deep learning to extract the activity information from CSI data. Other radar-based approaches use FMCW Radar to extract range and velocity information to reconstruct the human pose and skeleton [12]. Although Wi-Fi-based HAR systems have success in predicting human activity, radar-based approaches utilizing mmWave, offer higher bandwidth signals, meaning they can more accurately detect motion in an environment. Many existing HAR works using mmWave radar aim to reconstruct the human skeleton or silhouette [2], [13], offering visual representations of activity that are comparable to vision-based systems. Other approaches use point cloud data generated from mmWave data to recognize activity [14]. The point clouds, however, are coarse and do not offer sufficient information for detailed activity assessment.

VII. Conclusion

In this work, we design and evaluate *MilliFit*, an exercise classification system using commercial mmWave devices. This work leverages the mmWave signals reflected off a person exercising and designs a deep-learning network that uses a combination of CNN and LSTM to classify the activities. We demonstrate the performance of *MilliFit* using COTS mmWave devices, and in the future, we will extend this work to more individuals performing a wider set of activities.

VIII. Acknowledgement

We sincerely thank the reviewers for their comments. This work is partially supported by the NSF under grants CAREER-2144505, MRI-2018966, and CNS-1910853.

References

[1] Edward M Sitar, et al., "A Millimeter-Wave Wireless Sensing Approach for at-Home Exercise Recognition," in *ACM MobiSys*, 2022.
[2] Aakriti Adhikari, et al., "MiShape: Accurate Human Silhouettes and Body Joints from Commodity Millimeter-Wave Devices," *ACM IMWUT*, vol. 6, no. 3, 2022.
[3] Amin Ullah, et al., "Activity Recognition Using Temporal Optical Flow Convolutional Features and Multilayer LSTM," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 12, 2019.
[4] Natalia Neverova, et al., "Multi-scale Deep Learning for Gesture Detection and Localization," in *ECCV Workshops*, 2015.
[5] Xiaonan Guo, et al., "FitCoach: Virtual Fitness Coach Empowered by Wearable Mobile Devices," in *IEEE INFOCOM*, 2017.
[6] Fuyang Huang, et al., "DeepFuse: An IMU-Aware Network for Real-Time 3D Human Pose Estimation from Multi-View Image," in *IEEE WACV*, 2020.
[7] David Strömbäck, et al., "MM-Fit: Multimodal Deep Learning for Automatic Exercise Logging across Sensing Devices," *ACM IMWUT*, vol. 4, no. 4, 2020.
[8] Meera Radhakrishnan, et al., "ERICA: Enabling Real-Time Mistake Detection & Corrective Feedback for Free-Weights Exercises," in *ACM SenSys*, 2020.
[9] Wenjun Jiang, et al., "Towards 3D Human Pose Construction Using Wi-Fi," in *ACM MobiCom*, 2020.
[10] Xiaonan Guo, et al., "Device-Free Personalized Fitness Assistant Using WiFi," *ACM IMWUT*, vol. 2, no. 4, 2018.
[11] Yan Zhu, et al., "FitAssist: Virtual Fitness Assistant Based on WiFi," in *ACM MobiQuitous*, 2019.
[12] Fadel Adib, et al., "Capturing the human figure through a wall," *ACM Trans. Graph.*, vol. 34, no. 6, 2015.
[13] Hao Kong, et al., "M3Track: Mmwave-Based Multi-User 3D Posture Tracking," in *ACM MobiSys*, 2022.
[14] Peijun Zhao, et al., "mID: Tracking and Identifying People with Millimeter Wave Radar," in *IEEE DCOSS*, 2019.