Egocentric Human Pose Estimation using Head-mounted mmWave Radar

Wenwei Li*
wenweili@seu.edu.cn
Southeast University
Nanjing, Jiangsu, China

Ruofeng Liu*
liux4189@umn.edu
Robert Bosch LLC
Sunnyvale, California, United States

Shuai Wang†
shuaiwang@seu.edu.cn
Southeast University
Nanjing, Jiangsu, China

Dongjiang Cao
djcao@seu.edu.cn
Southeast University
Nanjing, Jiangsu, China

Wenchao Jiang
wenchao_jiang@sutd.edu.sg
Singapore University of Technology and Design
Singapore, Singapore

ABSTRACT

3D human pose plays a critical role in human behavior understanding and has many applications (e.g., VR/AR). Conventional pose estimations deploy sensors as fixed infrastructure, which significantly restrains the mobility of the user. Inspired by the emerging head-mounted devices (e.g., VR/AR glasses) and the recent advance in low-cost mmWave radar, we present mmEgo, the first egocentric human pose estimation design using a head-mounted mmWave radar, which offers ubiquitous pose tracking with high mobility, robustness to complex environments, and privacy preservation. To tackle the unique challenges of radar sensing from the egocentric perspective (e.g., random radar motion and the scarcity of information on the lower body), we propose several technical designs, including root-relative radar motion tracking for radar motion decoupling and a two-stage pose estimator that incorporates human kinematics priors. Extensive experiments and case studies show that our method can reduce the joint localization error by 44.2% and potentially enable a wide spectrum of applications.

CCS CONCEPTS

• **Human-centered computing → Ubiquitous and mobile computing**; • **Computer systems organization → Embedded and cyber-physical systems**.

KEYWORDS

Human Sensing, Millimeter Wave, Deep Learning, Virtual Reality

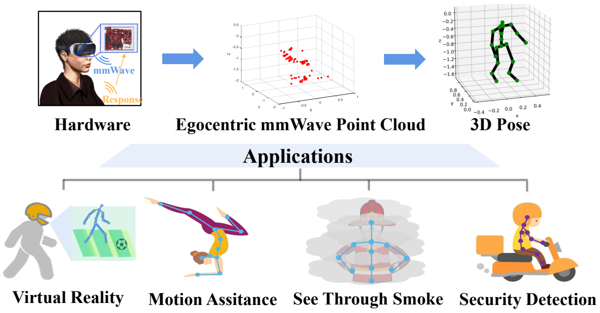


Fig. 1. Egocentric human pose estimation with head-mounted mmWave radar and applications.

ACM Reference Format:
Wenwei Li, Ruofeng Liu, Shuai Wang, Dongjiang Cao, and Wenchao Jiang. 2023. Egocentric Human Pose Estimation using Head-mounted mmWave Radar. In *ACM Conference on Embedded Networked Sensor Systems (SenSys '23), November 12–17, 2023, Istanbul, Turkiye.* ACM, New York, NY, USA, 14 pages. https://doi.org/10.1145/3625687.3625799

1 INTRODUCTION

Human pose estimation, aiming at reconstructing 3D body motions, plays an important role in many applications such as sports analysis, human-computer interaction, augmented reality (AR), virtual reality (VR), and rehabilitation. Conventionally, human poses are captured by MoCap devices (e.g., cameras) installed around the scene or a large number of IMUs on the body. However, the fixed MoCap infrastructure suffers from a limited recording volume, which constrains the range of spatial motions and thus cannot serve daily activities involving large-range mobility. On the other hand, body-worn IMUs require cumbersome setups and complicated calibration operations, causing inconveniences for users and hindering normal activities and social interactions.

The recent popularity of head-mounted devices (e.g., VR glasses and smart helmets) inspires a new direction named *egocentric pose estimation* [17, 20, 33, 40, 46, 51]. Specifically, it estimates the pose from a single head-mounted device worn by the user, which offers the user both mobility and convenience. For example, the Apple Vision Pro [1] is equipped with several cameras to replace the traditional handle. However, vision-based approaches are sensitive to lighting conditions, smoke, dust, and the appearance of humans and often cause privacy concerns.

[*] Both authors contributed equally to the paper.
[†] Corresponding author.

SenSys '23, November 12–17, 2023, Istanbul, Turkiye

ACM ISBN 979-8-4007-0414-7/23/11...$15.00
https://doi.org/10.1145/3625687.3625799

In this work, we propose mmEgo, the first egocentric pose estimation design using mmWave radar. The emerging integrated radar is low-cost and compact (approximately 10 centimeters), making it easy to embed it into head-mounted devices (e.g., AR headsets). Furthermore, the researchers [8, 26, 36, 52, 53, 65] recently demonstrated their potential to provide various human sensing capabilities while being robust against adverse lighting or weather, privacy-preserving, and non-intrusive to users. We envision that radar-based egocentric pose estimation enables a wide range of applications such as immersive VR, motion-assisted analysis, AR vision enhancement for first responders, and security behavior detection for drivers (depicted in Fig. 1).

Despite the recent success of radar in various human sensing tasks, the egocentric perspective imposes two unique challenges for pose estimation. First, head-mounted devices are non-stationary, and therefore the radar signal contains not only the change of pose but also random device movement caused by head motion. Our experiment shows that the radar movement significantly corrupts the spatial-temporal features of the pose obtained by the state-of-the-art method, leading to significant joint localization errors (a maximum of 16cm). Secondly, due to the top-down view angle, the radar signal on the lower body suffers from severe specular reflection and self-occlusion by the upper limbs. As a result, an extremely limited amount of lower-limb motion is perceived by the radar, causing challenging accuracy in lower-body estimation.

To tackle the aforementioned challenges, we introduce several novel approaches for accurate egocentric 3D pose estimation with a single head-mounted mmWave radar. Firstly, to mitigate the impact of radar movement, we fuse the radar point cloud with IMU measurements commonly available in head wearables. We design a multi-scale LSTM network to accurately track the root-relative position of radar while carefully avoiding the curse of large drift of IMU. Using accurate radar position tracking, we manage to decouple the radar self-motion from the radar point cloud and restore the spatial-temporal signature of the human pose. Secondly, to overcome the scarcity of lower-body information, we exploit the inherent correlation between upper and lower-body movements. We design a two-stage pose estimation network that explicitly incorporates the sparse lower-limb point cloud with the available upper-body clues to infer the missing lower-body posture. The proposed approach effectively reduces the upper and lower joint error by 35.4% and 42.7% respectively.

To summarize, our contributions are as follows:

- To the best of our knowledge, we present mmEgo, the first egocentric human pose estimation design using commercial-off-the-shelf mmWave radar, providing ubiquitous pose estimation with high mobility, environment robustness, and privacy preservation simultaneously.
- We develop a head-mounted testbed with a commodity radar device and collect a real-world egocentric radar dataset of various daily activities. Through benchmark experiments, we identify the fundamental challenges of egocentric human pose estimation using radar.
- We propose a novel multi-stage pose estimation network that is resilient to random radar motion incurred by head motion and tackles the scarcity of lower-limb information.
- Extensive evaluations and case studies are conducted, which demonstrate that mmEgo achieves an average joint localization error of 4.3cm and an average rotation error of 4.9°.

2 MOTIVATION AND CHALLENGES

2.1 Motivation

The feasibility of human pose estimation using mmWave radar has recently been studied in [26, 36, 52, 53]. However, these existing works deploy the radar as a fixed infrastructure (e.g., mounted on the wall or the ceiling) and perceive the user from the peripheral view, which has fundamental limitations. A single radar deployment suffers from a limited sensing range (10m), which significantly restrains the user's mobility while installing multiple radars incur high cost and complexity. To address the limitations, we propose *egocentric* radar pose estimation by integrating radar into the emerging head-mounted wearables (e.g., Apple Vision Pro [1] and Microsoft Hololens [3]) and monitoring the user's pose from the top-down perspective. The benefits are obvious. Embedding radar into wearable naturally enables ubiquitous pose estimation for a wide range of applications with high mobility. Furthermore, the user only needs to wear a single device, which is both more cost-effective and convenient.

Potential Applications. Fig. 1 illustrates the potential use cases of the proposed system. In immersive applications, VR/AR glasses can use the embedded radar to continuously track users' poses and dynamically produce content. For athletes, the sports helmet equipped with radar offers real-time pose analysis and provides feedback for them to optimize their performance, prevent injuries, and enhance their overall fitness levels. In addition, the pose estimated by radar-equipped safety helmets monitors the safety status of firefighters or miners in complex environments and facilitates effective coordination among team members. Finally, the radar-equipped vehicle helmet worn by the driver can monitor the user's safety behavior and prevent traffic accidents.

2.2 Challenges

Technically, estimating pose with radar from the egocentric perspective is non-trivial and incurs two major challenges.

2.2.1 Random radar motion. Unlike the fixed radar infrastructure, the head-mounted radar device is non-stationary as the head might move voluntarily (e.g., the user looks left and right) or involuntarily (e.g., vibrates on walking). This imposes challenges for pose estimation since the radar data contains both motions caused by pose changes as well as the random device motion caused by the head. The pose estimation needs to distinguish human pose changes from the motion of the device, which is non-trivial for mmWave radar.

More specifically, limited by the angular resolution and aperture size, low-cost radar (e.g., TI IWR6843) produces 3D point clouds that are sparse and noisy (an example is shown in Fig. 1). A single frame of point cloud often lacks sufficient information for pose estimation. Recent pose estimation designs [9, 47, 53] commonly employ the neural network that can extract spatial-temporal features from multiple consecutive radar frames. The neighboring frames supplement each other and provide additional details of various

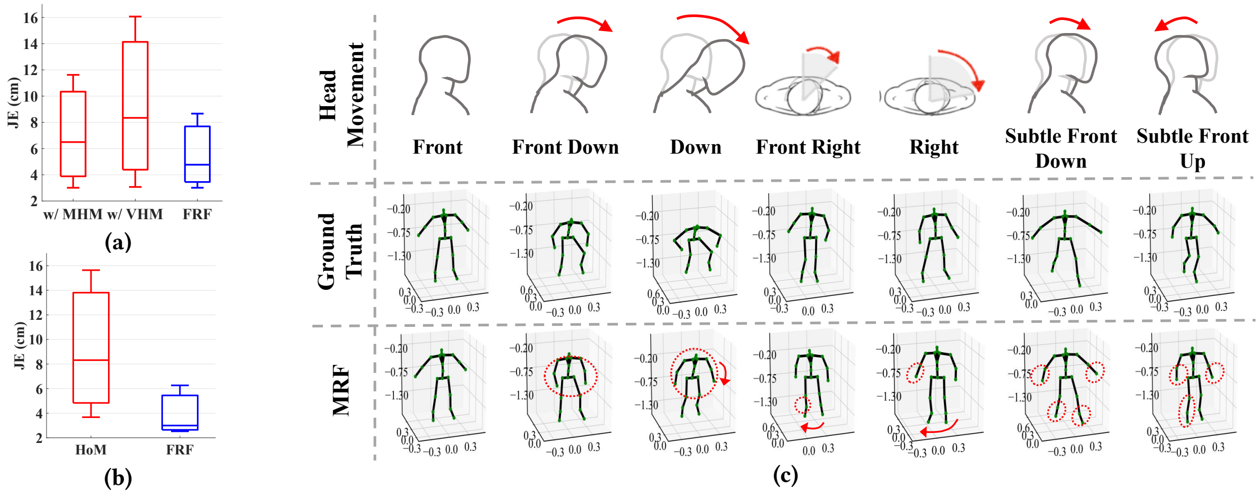


Fig. 2. Impact of random radar motion: (a) joint localization error (JE) with vigorous and minor head motion (VHM and MHM); (b) JE with head motion only (HoM); (c) visualization of pose estimation error for SOTA [53].

body parts. However, the presence of random radar motion renders the multi-frame feature extraction and fusion difficult because spatial-temporal features of a specific body motion might vary significantly under different radar motions. Moreover, voluntary head motions cause dramatic changes in the reference coordinate system between frames, making it impossible to associate point clouds of the same body part across frames and track their trajectory.

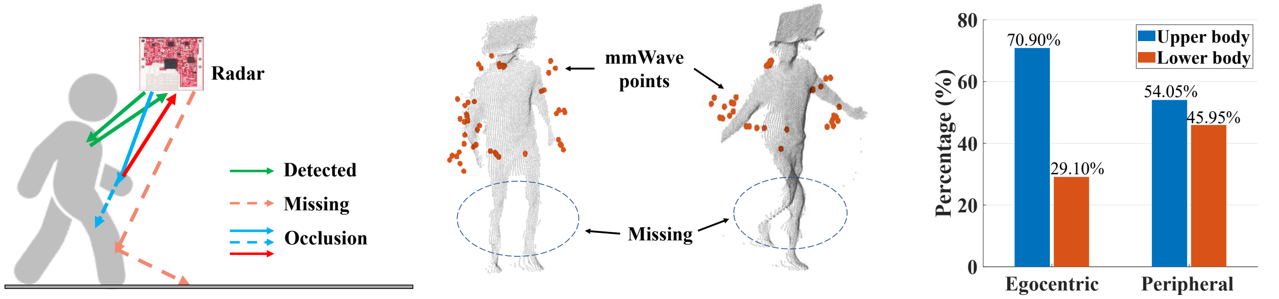
Benchmark Experiment. We investigate the impact of radar motions on egocentric pose estimation. The subjects wear the experimental helmet (detailed in Section 4.1) and perform the actions of three categories: body pose change accompanied by minor (i.e., involuntary) head motion (e.g., walking, raising hands), body pose change accompanied by vigorous (i.e., voluntary) head motion (e.g., looking up and down while raising hands, looking left and right while walking), and pose with only head motion (e.g., nodding, shaking head). We collect raw point clouds observed from the moving reference frame of radar (MRF) as well as the absolute position of the radar in the world coordinate system using an Azure Kinect. We implement the state-of-the-art design for stationary radar infrastructure (mmMesh [53]) to predict joint positions directly from raw point clouds in MRF. In addition, to measure the impact of head motion, we manually transform raw point cloud data into the fixed reference frame (FRF) using ground truth radar positions in the world coordinate system (from Kinect) and repeat the predictions with mmMesh.

Fig. 2 compares the distribution of joint localization errors between MRF (with radar motion) and FRF (radar motion is canceled with ground truth). As depicted in Fig. 2(a), voluntary head motions (VHM) of the human body can disrupt pose estimation, incurring a 72.9% overall error increments and causing up to 16.07 cm in leaf joints (e.g., wrist and ankles). Even involuntary head motion (MHM) results in a non-trivial error increase (47.9%). Interestingly, Fig. 2(b) shows the 177% error increments when there are only head motions (HoM), implying that the machine learning model cannot distinguish pose change and radar motion. Fig. 2(c) depicts examples of the results where dotted circles and arrows indicate areas of significant error and overall bias compared to ground truth. While the result closely resembles the true skeleton without head motion (1^{st} frame), voluntary head motions ($2^{nd}-5^{th}$ frames) introduce severe estimation errors, (e.g., whole body rotation) and minor head motions ($6^{th}-7^{th}$ frames) also result in deviations of body parts.

The results demonstrate that radar motions severely affect the accuracy of egocentric radar pose estimation. To overcome the challenges, a straightforward approach is to leverage IMU to track radar at each timestep and align the radar frames to the world coordinate system. However, IMU tracking is non-trivial, and the estimation error of the absolute position using conventional algorithms (e.g., double integration [10]) drifts over time, which remains an ill-posed problem to address. We will propose our method to address this challenge in Section 3.3.

2.2.2 Scarcity of lower-body information. The second critical challenge for egocentric pose estimation is the scarcity of radar point clouds on the lower body. Specifically, mmWave signal undergoes specular reflection (mirror-like reflection) on the human skin. As Fig. 3(a) shows, due to the top-down perspective of the radar, a significant amount of signal impinging on the lower body is reflected towards the ground. Moreover, radar signals could be occluded by the upper limbs, further limiting the received information from the lower body. Fig. 3(b) showcases an extreme example of point clouds captured during walking and lunging, where the lower body is completely missing in the point cloud.

To quantify the scarcity of information regarding the lower body, we gather radar point clouds corresponding to 13 representative activities (elaborated in Section 4.2) from both egocentric and peripheral perspectives. These point clouds are segmented into upper and lower body portions, with the pelvis joint serving as the boundary. In Fig. 3(c), we present a statistical comparison of the proportion of point clouds detected on the upper and lower body from both traditional peripheral and our innovative egocentric perspectives, revealing a notable shift from a nearly 1:1 ratio to an approximate 3:1 ratio. Consequently, addressing lower-body pose estimation with limited information becomes a non-trivial challenge, which we will delve into further in Section 3.4.



(a) Specular reflection and self-occlusion.

(b) Loss of lower body point cloud.

(c) Distributions of point clouds: egocentric vs. peripheral.

Fig. 3. Scarcity of lower-body information in egocentric perspective.

3 DESIGN

3.1 Overview

mmEgo consists of three major components depicted in Fig. 4.

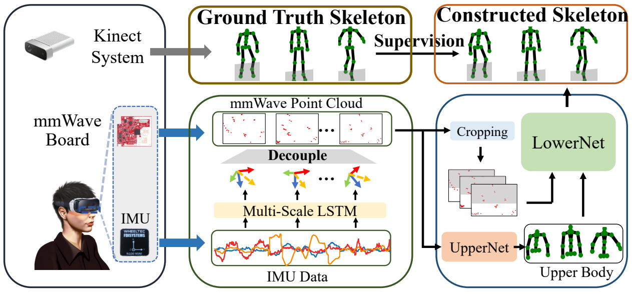
Data Acquisition. The head-mounted device is equipped with a mmWave radar and an IMU sensor that simultaneously collects the radar point cloud and IMU measurements. Specifically, the radar emits FMCW (Frequency Modulated Continuous Wave) signals, captures reflections from the user, and generates 3D radar points. The collocated IMU sensor measures the acceleration and angular velocity of the device using its built-in gyroscope and accelerometer. Additionally, in the training stage, we deploy an Azure Kinect v2 [4] to obtain the ground truth labels of the user's pose and the radar motion from the peripheral view.

Radar Motion Tracking. To mitigate the impact of radar motion, we introduce a multi-scale LSTM network that estimates the radar position and orientation from IMU data. The estimation serves to decouple the information of human pose change from the raw point cloud and restore the spatial-temporal features for human pose estimation. The details are described in Section 3.3.

Human Pose Estimation. The 3D human skeleton is reconstructed with the decoupled point cloud. A two-stage deep neural network is presented to estimate upper and lower body poses separately (i.e., UpperNet and LowerNet). The details are given in Section 3.4.

3.2 System Input

The input of our system consists of the point clouds obtained from mmWave radar and IMU measurements. The sequence of mmWave point clouds can be denoted as $p_{mm} = \{p_{i,t}\}$, $t \in [0, T_1)$ and $i \in [1, N]$, where T_1 represents the number of frames and each frame contains N points. Each point $p_{i,t} \in \mathbb{R}^6$ includes 3D coordinates $(x_{i,t}, y_{i,t}, z_{i,t})$ in the radar coordinate system, range $r_{i,t}$, intensity $s_{i,t}$ and velocity $v_{i,t}$. The measurements of IMU is represented as $x_{imu} = \{a_t, \omega_t\}$, $t \in [0, T_2)$, where T_2 is the number of IMU samples, $a_t \in \mathbb{R}^3$ is the acceleration, and $\omega_t \in \mathbb{R}^3$ is the angular velocity.



Data Acquisition Radar Motion Tracking Human Pose Estimation

Fig. 4. System Overview.

3.3 Radar Motion Tracking

This section introduces multi-scale LSTM to track the motion of the radar from IMU, which addresses pose estimation error incurred by radar motion (described in Section 2.2.1).

3.3.1 Design Methodology. Due to the noise and bias of low-cost IMU measurements, the estimation of the *absolute* radar motion in the global reference system suffers from the large drift [10]. Recent studies [12, 15, 28, 38] attempt to address this issue using learning approaches but they commonly make assumptions about the user's motion (e.g., only walking) and therefore cannot be applied to pose estimation. Our key insight is that to mitigate the impact of radar motion on egocentric pose estimation, we only need to estimate the *relative* radar position and orientation rather than the absolute one. Specifically, egocentric pose estimation can be considered as tracking the relative position of various joints to the root of the skeleton. Therefore, we can define and address the radar motion tracking problem in the root-relative coordinate system as well.

As Fig. 5 shows, we choose the neck to represent the root of the skeleton and estimate the radar position and orientation in the neck-relative system. We find that benefiting from the inherent structure of head motion, it is feasible to predict the relative rotation of the radar to this point based on IMU acceleration and angular velocity. Furthermore, due to the geometric constraints of the head, the relative position of radar has a definite one-to-one mapping with its relative rotation and thus is also predictable from IMU measurements.

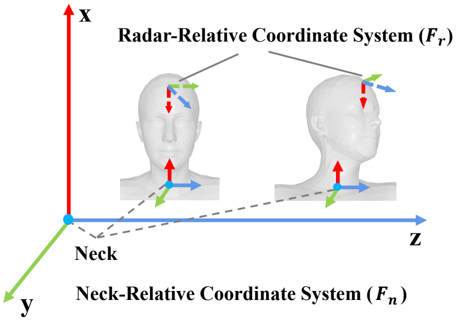

x
Radar-Relative Coordinate System (F_r)
z
Neck
y
Neck-Relative Coordinate System (F_n)

Fig. 5. Estimating radar position and orientation in the neck-relative coordination system (F_n).

3.3.2 Multi-scale LSTM. We use a data-driven approach to predict the relative radar motion. To capture both involuntary (subtle) head motion at short time scales and intentional (violent) motion at long time scales, we design a multi-scale LSTM network that models the radar motion and predicts the root-relative radar position and orientation. As illustrated in Fig. 6, we first segment the IMU input sequence $x_{imu}^{0:T_2}$ into sub-sequences $[x_{imu}^{0:\sigma}, x_{imu}^{\sigma:2\sigma}, \cdots, x_{imu}^{T_2-\sigma:T_2}]$ where the length $\sigma = T_2/T_1$ is the number of IMU samples during a radar frame interval. We use fully connected layers to generate feature embeddings for the sub-sequences, which are then fed into a bidirectional two-layer LSTM [34] to extract sequential features at a short time scale (i.e., 100 milliseconds). Then, a self-attention layer [41] aggregates the extracted features in each sub-sequence:

$$f_i = \sum_{t=0}^{\sigma} L(c_i^t) \cdot c_i^t \tag{1}$$

where i is the index of sub-sequence, t is the sample index in each sub-sequence, c_i^t is the representation of each time step in the ith sub-sequence, L is a learnable linear mapping function and f_i is the aggregated representation.

To extract sequence features at a long time scale, the aggregation representations undergo another bidirectional two-layer LSTM. Finally, a fully connected layer is used to map the extracted features to the output including root-relative radar orientation $R_r \in \mathbb{R}^{3\times3}$, and the root-relative position $p_r \in \mathbb{R}^3$. In the network, we use the 6D orientation as the intermediate representation of R_r, which is a smoother and more continuous rotation representation mentioned in Zhou et al. [67], and finally convert it to the rotation matrix in SO(3). The loss function used to train this radar motion tracking module is defined as:

$$\mathcal{L}_R = \alpha \sum_{t=0}^{T_1} \cos^{-1}\left(\frac{tr(\hat{\mathbf{R}_r^t}\mathbf{R}_r^{t\,T}) - 1}{2}\right) + \beta \sum_{t=0}^{T_1} \left\|\hat{\mathbf{p}_r^t} - \mathbf{p}_r^t\right\|_2 \tag{2}$$

where α and β represent hyperparameters, $\hat{\mathbf{R}_r^t}$ and $\hat{\mathbf{p}_r^t}$ denotes the predicted orientation and position of the radar at the t^{th} frame, while $\mathbf{R}_r^t$ and $\mathbf{p}_r^t$ correspond to the ground truth values.

3.3.3 Decouple procedure. To mitigate the impact of radar motion, we use the estimated radar position and orientation to unify the sequence of radar point clouds to an identical coordinate system defined by the root of the skeleton. In this way, the transformed point cloud can be considered as being observed from a virtual radar on a fixed spot in the root-relative system, and the radar motion is decoupled. More specifically, the relative radar orientation ($\mathbf{R}_r$) and position ($\mathbf{p}_r$) predicted by the multi-scale LSTM are used to transform the coordinate system of the point cloud p_{mm} from the radar coordinate system (denoted as F_r) to the neck coordinate system (denoted as F_n) as shown in Fig. 5. As a result, the estimated human pose from the point cloud is also transformed into F_n. Note that in order to eliminate the influence of radar motion errors on the human pose, we perform a reverse transformation of the human pose back to F_r in the end. In the following sections, unless specified otherwise, the input point clouds and body joints are represented in F_n and the output body joints are represented in F_r.

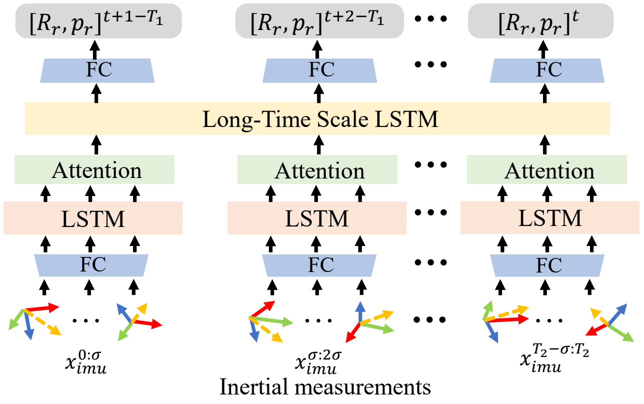


Fig. 6. Radar motion tracking module.

3.4 Two-stage Human Pose Estimation

This section introduces our two-stage method to estimate the position of joints from sparse mmWave point clouds.

3.4.1 ***Design Methodology.*** Egocentric pose estimation is nontrivial for radar because the lower-body information is extremely scarce due to specular reflection and self-occlusion. As discussed in Section 2.2.2, we observe a significant imbalance of point cloud distribution on the upper and lower body. Motivated by this observation, we design a two-stage pose estimation network, in which we break down the task into upper body estimation with UpperNet (Section 3.4.2) and lower body estimation with LowerNet (Section 3.4.3). UpperNet directly predicts the upper-body pose from radar point clouds whereas the LowerNet combines the UpperNet prediction with the sparse point cloud on the lower body.

Two-stage pose prediction brings about two major benefits. First, it can explicitly exploit the prior knowledge of kinematics to enhance the lower-body pose estimation. In specific, there is a close relationship between the upper and lower body in many daily actions. For example, the motion of the arms and legs is highly correlated when walking. The arms tend to retract when squatting, and the arms extend when lunging. Recent works in computer vision [13, 21, 31, 56] demonstrate the feasibility of leveraging the priors to recover whole-body postures from partial observations. Technically, we employ a graph convolutional network (GCN) to extract the features from upper-body predictions and incorporate them as clues to enhance the lower-body predictions. In addition, the two-stage design addresses the full-body estimation problem with the "divide and conquer" paradigm, decomposing the task into the upper-body estimation which is relatively easy and the more challenging lower-limber problem. Compared to a one-stage approach (i.e., predicting the full body simultaneously), the two-stage model converges faster during training and suffers less from overfitting.

3.4.2 ***Stage I: upper body estimation.*** To estimate the upper body from the mmWave point cloud p_{mm}, we propose UpperNet as demonstrated in Fig. 7 to learn the mapping from input to the 3D coordinate of each upper-body joint $J_i \in \mathbb{R}^3, i \in [1, M_{upper}]$, where $M_{upper} = 15$ denotes the number of upper body joints to be estimated in this work.

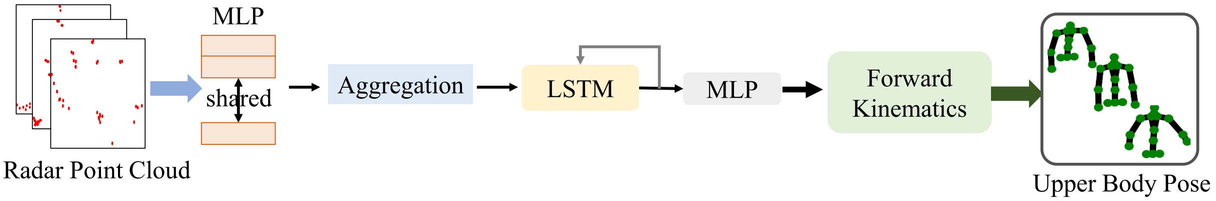


Fig. 7. Stage I: Upper body pose estimation module (UpperNet).

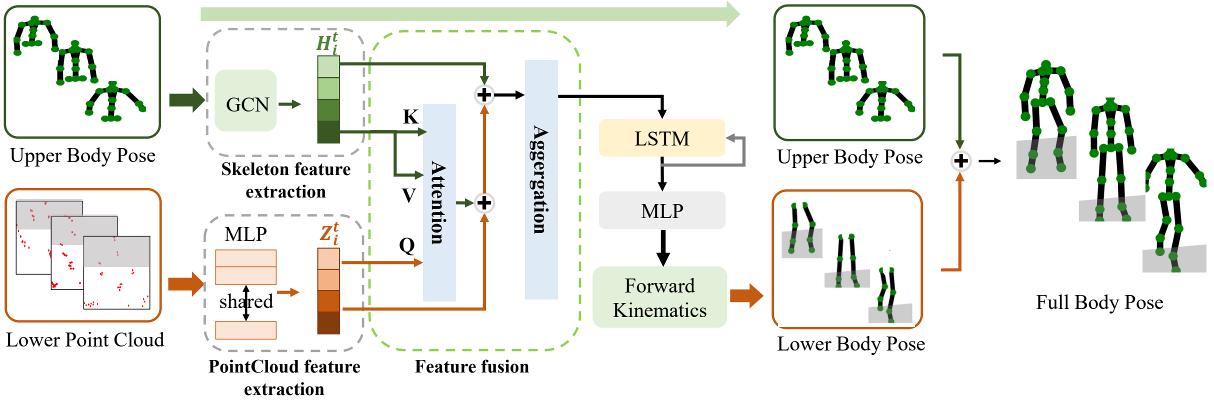


Fig. 8. Stage II: Lower body pose estimation module (LowerNet).

Specifically, we use a shared-weighted MLP [30] (Multi-layer Perception) to extract high-level features for each point $p_{i,t}$ in the point cloud p_{mm}. Next, we use the self-attention [41] mechanism to aggregate features of all points in each frame to obtain a feature representation for each frame, which is demonstrated the effectiveness in mmMesh [53]. The feature representation is then fed into a bidirectional LSTM to capture the temporal relationship between consecutive frames, which aggregates the supplementary information of neighboring frames to features of the current frame. Finally, the feature is mapped to the upper body pose $\theta = [RJ_i^{6D} | i = 1, 2, \cdots, M_{upper} - 1]$ by MLP, where RJ_i^{6D} is the rotation of each joint relative to its parent joint represented as 6D vector [67]. Following the radar motion tracking in Section 3.3, the neck is chosen as the root joint. To obtain the final joint positions, 6D rotations are converted to the rotation matrix and fed into a forward kinematics module with an initial upper body skeleton. The joint positions are calculated as:

$$\mathbf{J}_i = J_{parent(i)} + \mathbf{RJ}_i(\overline{J}_i - \overline{J}_{parent(i)}) \tag{3}$$

where $J_{parent(i)} \in \mathbb{R}^3$ is the parent joint of J_i on the upper body skeleton tree, $\mathbf{RJ} \in R^{3\times3}$ is the rotation of the joint J_i with respect to its parent, and $\overline{J}_i, \overline{J}_{parent(i)}$ are the initial position of $\mathbf{J}_i, J_{parent(i)}$ respectively. Note that initial skeletons are unnecessary during operation because the human pose can be represented and understood via joint rotations.

The loss function used to train this UpperNet is defined as the L2 norm of the difference between predicted joint positions $\hat{\mathbf{J}}_i^t$ and ground truth joint positions $\mathbf{J}_\mathbf{i}^\mathbf{t}$:

$$\mathcal{L}_U = \sum_{t=0}^{T_1} \sum_{i=1}^{M_{upper}} \left\| \hat{\mathbf{J}}_i^t - \mathbf{J}_i^t \right\|_2 \tag{4}$$

3.4.3 *Stage II: lower body estimation.*

To estimate the lower body pose with extremely sparse point clouds, we propose LowerNet which explicitly learns kinematics priors. As shown in Fig. 8, the input consists of the upper body joint positions $\mathbf{J_i}, i \in [1, M_{upper}]$ estimated by UpperNet and cropped lower-body point cloud p_{mm}^l which is segmented based on the height of the pelvis joint. We employ two different feature extraction networks for each modality and then perform multi-modal feature fusion and temporal feature aggregation to estimate the pose of the lower body.

Skeleton feature extraction. As the human body skeleton is a natural graph structure, it is suitable to use graph convolutional networks (GCN) [25] to extract the upper body skeleton feature. GCN represents a specific variation of Convolutional Graph Neural Networks. Its effectiveness lies in its capability to learn node representations by simultaneously incorporating graph structures and node features, extending the concept of convolution from grid data to graph data. In this module, the upper body skeleton is represented by graph $G(V, E)$ where V is the set of joints and E is the connections between joints. For each joint v_i, a connection pointing from v_j to v_i is denoted as $e_{ij} = (v_i, v_j) \in E$, and its neighborhood is defined as $N(v_i) = \{u \in V | (v_i, u) \in E\}$. The adjacency matrix is a $M_{upper} \times M_{upper}$ matrix with $A_{ij} = 1$ if $e_{ij} \in E$ and $A_{ij} = 0$ if $e_{ij} \notin E$. The joint attributes of the skeleton graph are denoted as X, where $X \in \mathbb{R}^{M_{upper}\times3}$ is a joint feature matrix with $x_v \in \mathbb{R}^3$ representing the 3D coordinates of a joint v. The simplified graph convolution operation is defined as:

$$H^t = f(\overline{A}X^t\Theta) \tag{5}$$

where $\overline{A} = \widetilde{D}^{-1/2}\widetilde{A}\widetilde{D}^{-1/2}$. $\widetilde{A} = A + I$ denotes the adjacency matrix with inserted self-loops and $\widetilde{D}$ is the diagonal degree matrix of $\widetilde{A}$. Θ is a learnable parameter and f is an activation function. We apply three consecutive convolution operations to the upper body skeleton graph to obtain the spatial feature H of the skeleton.

Pointcloud feature extraction. We use a shared MLP (Similar to UpperNet) to extract high-level features Z_i^t for each radar point on the lower body.

Feature fusion. To enrich the lower-body features with the upper-body cues, we perform the mutual attention operation to dynamically learn the semantic features of each point in the context of the upper-body skeleton. Formally, at frame t, we first obtain the query, key, and value matrices:

$$Q^t = Z^t W_Q, K^t = H^t W_K, V^t = H^t W_V \tag{6}$$

where $W_Q, W_K, W_V \in \mathbb{R}^{d \times d'}$ are learnable parameters, d is the dimension of Z_i^t and H_i^t, and d' is the dimension of the query, key, and value matrix. Then we apply attention operations in the spatial dimension to model the interactions between radar points and upper joints. The skeleton-aware point feature is obtained by multiplying the attention scores with the value matrix and residual connection as:

$$A^t = Z^t + softmax(\frac{Q^t (K^t)^T}{\sqrt{d'}}) V^t \tag{7}$$

The obtained feature A^t is concatenated with the skeleton feature H^t, and the resulting feature is fed into a self-attention module for aggregation. This aggregated representation is then fed into a three-layer bidirectional LSTM to learn the temporal dependencies and patterns of human motion. Finally, an MLP predicts the rotational angles of lower body joints, followed by a forward kinematics module (similar to UppeNet) to obtain the lower-body joint positions $L_i \in \mathbb{R}^3, i \in [1, M_{lower}]$, where $M_{lower} = 8$ denotes the number of lower body joints to be estimated in this work. The loss function for training this LowerNet is defined as:

$$\mathcal{L}_L = \sum_{t=0}^{T_1} \sum_{i=1}^{M_{lower}} \left\| \hat{\mathbf{L}_i^t} - \mathbf{L}_i^t \right\|_2 \tag{8}$$

The output lower body skeleton L of this module is concatenated with the upper body skeleton J from the input to obtain the final output of the full body skeleton J_{all}.

4 IMPLEMENTATION

This section describes the implementation of mmEgo, including the setup for data collection, preprocessing of radar data, and the neural network's details.

4.1 Experiment Platform

mmWave Radar. The IWR6843ISK-ODS [2] mmWave radar is used, operating on the frequency range of 60GHz to 64GHz, with a wavelength of approximately 4mm. It consists of three transmitting antennas and four receiving antennas, which provide a 120° azimuth FoV and a 120° elevation FoV, with an angle resolution of around 15°. We utilize the FMCW processing chain provided by TI to produce 3D point clouds.

IMU Platform. We utilize an off-the-shelf inertial navigation device, Wheeltec N100. It integrates a gyroscope, accelerometer, and magnetometer. We obtain the raw data output from the device and perform calibration prior to its usage. The sampling frequency of the raw data is set to 200Hz.

Helmet Platform. As depicted in Fig. 9(c), the mmWave radar and IMU devices are securely mounted on a 3D-printed helmet plate using screws. The radar is oriented downward, and the distance between the radar and the calibration board is set at 19.8cm, while the IMU is situated directly above the radar antenna. The central part of the helmet plate features a vertical checkerboard for calibration with a size of 6×9, where each square has a side length of 30mm.

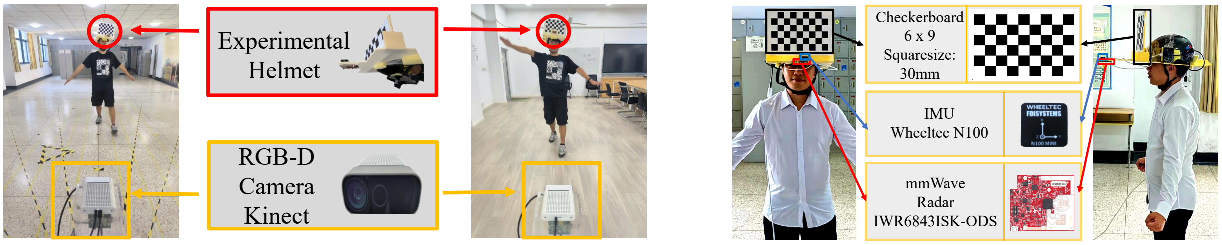
Kinect Platform. To collect the ground truth of pose and radar motion, we use an RGB-D camera (Azure Kinect v2) that can collect fine-grained 3D mesh of the subjects and produce accurate 3D human pose, as well capture high-resolution RGB images (for chessboard calibration). Because of its much better resolution (typical systematic error < 11mm + 0.1% of distance) than mmWave radar (~ 4cm), Kinect is commonly used to collect ground truth in RF sensing tasks [43, 47, 55].

4.2 Data Acquisition and Preprocessing

4.2.1 Data Acquisition. We conduct data collection at two distinct locations: a dimly lit office building hallway (Fig. 9(a)) and a well-lit classroom (Fig. 9(b)). These environments have unique characteristics: the hallway is relatively open, with walls approximately 4 meters from the subject, while the classroom is a confined space with tables, chairs, and electronic devices situated at around 2 meters from the subject. Three adult male volunteers, varying in height from 1.75 to 1.83 meters and weighing between 70 to 83 kilograms, are recruited to wear the helmet depicted in Fig. 9(c) and execute a predefined set of activities. Notably, the use of a calibration board is unnecessary in real-world applications.

For our experimental activities, we meticulously selected 13 actions to encompass a wide range of head and upper/lower limb movements. These actions included activities with subtle head motions, such as (1) walking in place; (2) walking; (3) horizontal abduction and retraction of the arms; activities primarily involving head motions, including (4) shaking head; (5) nodding head; (6) turning head; actions with voluntary head motions, namely (7) looking left and right while walking in place; (8) looking up and down while walking in place; (9) looking up and down while swinging arms; and activities featuring leg movements, such as (11) lunge; (12) high leg raise; (13) squat. Each action was repeated for one minute, resulting in the acquisition of more than 5400 radar frames and 100K IMU measurements for each activity. To establish the ground truth of the pose, volunteers performed all actions facing the Kinect camera. Our data collection procedure received approval from the Institutional Review Board (IRB) of authors' institution.

4.2.2 Preprocessing. To mitigate the clutter in point clouds and separate human subjects, we apply a cylindrical crop to the point cloud using the radar as the upper circle center with a diameter of 1.5m and a height equal to the distance from the radar to the ground. For accurate ground truth, we transform them from the Kinect coordinate system to the radar coordinate system utilizing chessboard calibration [60]. Finally, we use the Robot Operating System (ROS) for data collection, label them with corresponding timestamps, and synchronize heterogeneous data by adjusting the timestamps with internal time delays of each sensor.


Experimental Helmet
RGB-D Camera Kinect
Checkerboard 6 x 9 Squaresize: 30mm
IMU Wheeltec N100
mmWave Radar IWR6843ISK-ODS

(a) Hallway (b) Room (c) Helmet setup

Fig. 9. Head-mounted radar testbed and experimental scenarios.

4.3 Model Setting and Training

This section describes the details of the mmEgo model, training, and testing procedure for each stage. In **Radar motion tracking module,** an FC with size (6, 512) is used for feature embedding. Both bidirectional LSTMs for short and long timescales are composed of two layers with dimensions being 512. The Attention layer is realized by a (1024, 1) fully connected network. Finally, an FC layer with a size of (1024, 9) outputs the radar position and orientation. In **UpperNet,** the feature extraction uses a shared MLP consisting of three layers, with the dimensions (6, 16, 32, 64). We adopt batch normalization followed by ReLU activation functions after all layers. The aggregation module is implemented by an FC of (64, 1). The subsequent LSTM adopts a bidirectional three-layer architecture with a size of 64. Finally, an MLP consisting of two layers with sizes (128, 64, 42) obtains joint rotation angles. The point feature extraction of **LowerNet** is the same as UpperNet. The GCN used for extracting upper body skeleton features consists of three layers with channel sizes of (32, 64, 128). The dimension of the Attention module used for fusion is (128, 128). The size of the aggregation module is (128, 1). A bidirectional LTSM and MLP (similar to UpperNet) produce the angle estimations.

For every activity, 80% of the data collected from each subject are used for model training, and the remaining 20% serve as the testing set. The training records are segmented into 10-second fragments, containing a sequence of 100 radar frames. During the training, we set the batch size to 20 and the learning rate to 0.0003. The hyper-parameters α and β in the loss functions (Equation 2) are set as follows: $\alpha = 3$ and $\beta = 5$. Our model is implemented with Python 3.9 and PyTorch 1.11.0, and trained with NVIDIA RTX 3090.

5 EVALUATION

The section starts with the evaluation metrics (Section 5.1) and competing approaches (Section 5.2). The overall performance is first reported in Section 5.3, followed by evaluations of key design components (Section 5.4). The system complexity and latency are reported in Section 5.4.6. Section 5.5 demonstrates case studies.

5.1 Evaluation Metrics

To quantify the performance of our proposed approach, we adopt the following metrics to evaluate the accuracy of radar motion tracking and body pose estimation, respectively.

Average Joint Localization Error (S). The metric is defined as the average Euclidean distance between the predicted skeleton key points (i.e., joint locations) and their ground truths[23, 39, 53].

Average Joint Rotation Error (Q). Joint rotation angles reflect the accuracy of the pose more directly than joint positions. The metric is defined as the average joint angle (represented in axis-angles) differences between predicted joint rotations and the ground truth rotations [53].

Average Radar Rotation Error (R). Similar to Average Joint Rotation Error, this metric measures the average angle differences between predicted radar rotations and the ground truth rotations.

Average Radar Translation Error (T). This metric is defined as the average Euclidean distance between the predicted radar location and the ground truth locations.

5.2 Baseline

We compare our approach with the following radar motion tracking and human pose estimation baselines. These approaches reportedly outperform conventional approaches [15, 38]. Note that radar motion cancellation for other sensing modalities (e.g., UWB) and sensing tasks (e.g., vital signs detection [59]) are considered out of scope due to the distinct radar point cloud patterns.

5.2.1 Radar motion tracking. As mmEgo is the first to estimate 6DoF (degree-of-freedom) root-relative radar motion from IMU, we develop three baseline approaches based on the popular neural networks suitable for IMU data.

RNN + Attention. A single RNN is applied on the entire IMU measurement sequence, representing the most frequently used approach [12, 38, 58]. Attention layer is added to make the output sample frequency compatible.

BiRNN [34] + Attention. Bidirectional RNN is adopted which considers the information both before the current time and after the current time. BiRNN effectively captures more comprehensive sequence information than RNN.

Temporal self-attention [41]. Temporal self-attention can model the temporal relationships between time steps in a time series. We use it as a baseline to capture the time dependencies of the IMU measurement data.

5.2.2 Human pose estimation. To evaluate our advantages in the egocentric scenario, we reproduce the recent designs for fixed infrastructure as baselines.

mm-Pose [36]. mm-Pose is an early work of pose estimation using mmWave radar. A low-size high-resolution radar-to-image representation is presented and a forked CNN architecture was used to predict the real-world position of the skeletal joints in 3D space.

P4Transformer [14]. P4Transformer is an advanced model designed for point clouds used by recent works (e.g., mmBody [11]). It improves the efficiency and accuracy of point cloud processing through 4D convolution operations and Transformer structures.

Table 1: Overall performance of mmEgo.

Model	Whole body		Upper body		Lower body	
	Q (°)	S (cm)	Q (°)	S (cm)	Q (°)	S (cm)
mm-Pose[36]	10.885	10.734	10.439	8.964	11.923	13.682
P4Transformer[14]	11.804	10.873	11.326	9.052	12.921	13.832
PCB[66]	11.278	11.031	10.796	9.070	12.404	14.214
mmPose-NLP[35]	10.640	10.406	10.453	8.492	11.077	13.493
mmMesh[53]	7.215	7.783	7.127	6.639	7.423	9.529
mmEgo	**4.914**	**4.346**	**5.217**	**4.287**	**4.207**	**5.460**

Point-convolution-based (PCB) [66]. PCB is also a convolution method designed for point cloud [50] and pose estimation.

mmPose-NLP [35]. mmPose-NLP adopts natural language processing ideas to represent mmWave point clouds as a collection of word vectors by voxelization and employs a seq2seq model to estimate human body pose.

mmMesh [53]. mmMesh is a state-of-the-art method that estimates human pose with radar point clouds. It proposes optimized pointNet and anchor module design.

5.3 Overall Performance

5.3.1 Average accuracy. The average accuracy of mmEgo and baselines are reported in Table 1, including the whole body, upper body, and lower body accuracy. mmEgo achieves an average joint localization (rotation) error of 4.3cm (4.9°) in the whole body estimation, 4.2cm (5.2°) in the upper body, and 5.4cm (4.2°) in the lower body. Our method outperforms the first 4 baselines by at least 6cm. Compared to state-of-the-art (i.e., mmMesh), mmEgo also improves the overall accuracy by 3.4 cm (44.2%). The results show the overall performance gain of mmEgo in egocentric settings.

5.3.2 Per-joint accuracy. We further break down the localization error of each joint in Fig. 12, where the corresponding locations of 21 joints on a skeleton are labeled in Fig. 11. Per-joint accuracy highlights the effectiveness of mmEgo. Specifically, our method significantly outperforms the state-of-the-art (e.g., mmMesh) at wrists (#7 and #11) and ankles (#14 and #18). The prior method suffers from 17cm errors, whereas mmEgo manages to control the error of all joints within 7.03 cm, improving the accuracy by up to 10.59 cm. Note that these leaf joints have the largest angle limits among all joints and therefore represent the ones that are most challenging to predict. Suffering from radar motion, the real-world experience of baselines dramatically deteriorates and even becomes difficult to interpret human pose, while our design which considers radar motion and kinematics priors can produce much more robust predictions.

5.3.3 Qualitative results. Fig. 10 shows 4 qualitative examples of the reconstructed 3D human skeletons. Fig. 10(a) is the result of walking in place. Our method is slightly better than mmMesh for the activity with subtle head motion. In Fig. 10(b), a lunge action is captured. It is clear that the baseline has difficulty estimating the lower-body pose, while our method that integrates upper-body cues is almost identical to the ground truth. Finally, Fig. 10(c) and (d) demonstrate left and right arm horizontal abduction and retraction with voluntary head motions. The baseline is greatly affected by head motion, making the posture difficult to estimate, while our method, which restores the spatial-temporal features of pose changes, provides more reliable results.

5.4 Effectiveness of Designs

5.4.1 Accuracy of radar motion tracking. Table 2 compares the radar motion tracking of our method with baselines in Section 5.2.1. Our approach reduces rotation errors by 26.0% and the translation error by 46.6%. These results come from our multi-scale LSTM design that predicts head motions at both small and large time scales.

5.4.2 Impact of radar motion decoupling. We investigate the effectiveness of the radar motion decoupling on pose estimation accuracy. Table 3 shows that decoupling radar motion using proposed neck-relative radar position and orientation ($Pred_r$) can clearly outperform the results by directly estimating the radar's absolute position and orientation (IMU_r). This observation highlights the superiority of our proposed approach in overcoming the large drift issue of IMU. Moreover, we repeat the experiment with ground truth radar position and orientation (GT_r). Our joint rotation approaches approach ground truth, while the ground truth further improves joint position accuracy by 1.2 cm. To further demonstrate our method is generic, we apply the radar motion decoupling to enhance all baseline methods. Comparing the performance in Table 5 with the original results in Table 1, the accuracy of all existing baselines shows improvement with the proposed radar motion decoupling. However, our method still has the best performance, outperforming the baseline by at least 23%, which is benefited from our two-stage network design.

Our design benefits from accurate radar motion tracking to decouple radar point cloud. We further investigate the impact of the error from radar motion tracking on the downstream tasks of pose estimation by iterating through rotation errors (R) from 0 to 6 degrees, as well as translation errors (T) from 0 to 6 cm. As shown in Fig. 13, the average joint location error in pose estimation steadily

Table 2: Accuracy of radar motion tracking.

Model	R (°)	T (cm)
RNN + Attention	2.872	3.831
biRNN + Attention	2.776	3.428
Temporal self attention	3.209	4.811
Ours	**2.375**	**2.567**

Table 3: Effects of radar motion decoupling.

	Q(°)	S(cm)
w/ GT_r	**4.822**	**3.173**
w/ $Pred_r$	4.914	4.346
w/ IMU_r	6.506	6.031
w/o r	6.939	6.095

Table 4: Effects of upper-body clues to lower-body estimation.

	Q(°)	S(cm)
w/ GT_u	**2.925**	**3.526**
w/ $Pred_u$	4.207	5.460
w/o u	5.737	6.952

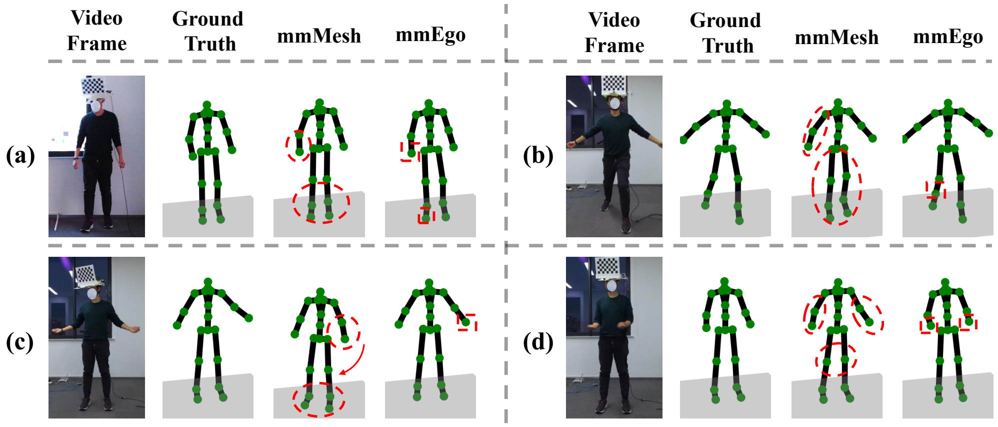


Fig. 10. Qualitative results of mmEgo vs. baseline.

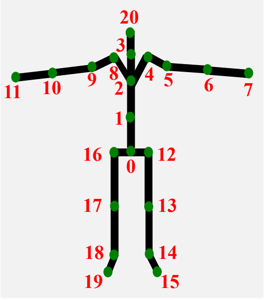


Fig. 11. Joint index map.

increases when rotation and translation errors of the radar tracking escalate. The contour line (denoted as AJE=7.7cm) represents the error produced by the state-of-the-art method (mmMesh [53]). When the trajectory error of radar tracking surpasses this threshold, it adversely impacts the pose estimation task. The pentagram ★ denotes the error position estimated by our design. The result indicates our radar motion decoupling can still improve the pose estimation accuracy when the tracking error increases within 100%.

Table 5: Generic benefits of radar motion decoupling.

Model	Q (°)	S (cm)
mm-Pose[36]	9.767	8.098
P4Transformer[14]	8.962	6.943
PCB[66]	8.906	7.701
mmPose-NLP[35]	6.522	6.988
mmMesh[53]	5.467	5.650
mmEgo	**4.914**	**4.346**

5.4.3 Impact of upper body cues. We study the effectiveness of using upper-body cues to enhance lower-body predictions. Table 4 shows that the predicted upper body (w/ $Pred_u$) improves the average rotation accuracy by 26.7% compared to directly estimating the lower body from scarce point cloud (w/o u). The error is further reduced if the ground truth upper-body pose (w/ GT_u) is available. The results prove that kinematics priors provide important hints for inferring the body parts that are not directly observed.

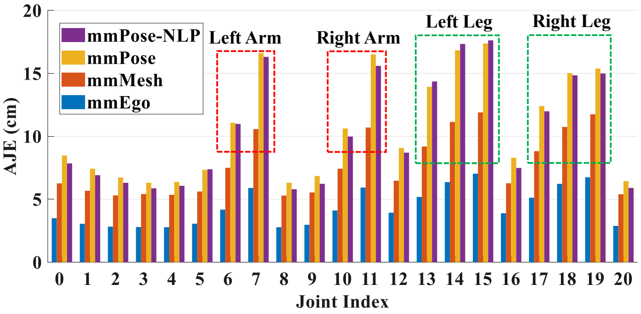


Fig. 12. Per-joint localization errors.

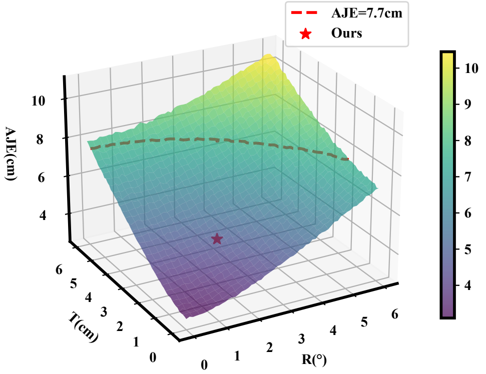


Fig. 13. Impact of radar tracking errors on pose estimation.

5.4.4 Impact of unseen scenes. mmWave signals traverse multiple paths and carry environment-specific information. We investigate the robustness of our design in unseen environments. Specifically, we use data collected in the hallway (depicted in Fig. 9(a)) for training and test the performance in a different scene: the room in Fig. 9(b). As shown in Table 6, the model demonstrates robustness in the new environment. In our future extension, we might collect more data from diverse environments and explore domain adaption techniques [22] to further enhance the robustness.

Table 6: Performance in unseen scenes.

	Q (°)	S (cm)
Hallway	5.474	4.456
Room	5.835	4.597

5.4.5 Impact of unseen subjects. We assess the generalization ability of mmEgo by the evaluating performance of new subjects who are not included in the training stage. Specifically, we train the model using data from one subject and reserve the data from two other subjects for testing. The results in Table 7 clearly show our superiority to the baseline. For example, the mmMesh's rotation error increases by 3.7° (51%) whereas our method only suffers from 0.1° (2%) more errors. This demonstrates the multi-stage design breaks down a complex problem into simpler stages, making the model much more generalizable.

5.4.6 Complexity and latency. We measure the model complexity and the prediction latency of mmEgo. The system comprises a total of 25.13M trainable parameters, with the radar motion tracking

Table 7: Performance for unseen subjects.

Model	Q (°)	S (cm)
mm-Pose[36]	11.597	11.349
P4Transformer[14]	11.961	11.169
PCB[66]	12.741	12.856
mmPose-NLP[35]	13.052	11.009
mmMesh[53]	10.938	8.517
mmEgo	**5.025**	**4.574**

module accounting for 92% of the parameters (23.11M). The UpperNet consists of 0.77M parameters, while the LowerNet has 1.25M parameters. Furthermore, we stream the data to a desktop with NVIDIA GeForce RTX 2060 GPU and Intel Core i7-9700 CPU and to perform inference. This setup is commonly adopted in VR headsets [5, 6]. The average latency of the entire system is measured at 19.8ms, where the radar motion tracking module accounted for 6.9ms of the latency, while UpperNet and LowerNet have about 6.9ms and 6.0ms latency respectively. These results demonstrate the feasibility of real-time operations. With the increasing availability of 5G networks and mobile edge computing, we envision that resource-constrained devices can offload the computation to the edge or cloud, enabling real-time egocentric pose estimation with large mobility.

5.5 Case Study

mmEgo can be widely utilized for various downstream tasks in emerging scenarios (e.g., VR and AR). We develop and evaluate two representative tasks: action recognition and air painting.

Case1: Action Recognition. The results of egocentric pose estimation can be applied to recognize the action, which plays a crucial role in many applications (e.g., safety monitoring, and motion assistance). We implement the state-of-the-art action recognition network (e.g., STGCN [54]) which takes a skeleton sequence as the input. The training and testing are performed with our dataset introduced in Section 4, which consists of 13 different activities denoted as (1) ~ (13). The ground truth is labeled during data collection. To highlight the accuracy of egocentric pose estimation, we compare the recognition accuracy between two settings: using the predicted skeleton by mmEgo and using the ground truth skeleton collected by Kinect. Remarkably, both of them achieve an average accuracy and F1 Score exceeding 99% (Table 8), suggesting that the impact of pose estimation errors of mmEgo can be considered negligible for this task. Fig. 14 further reports the recognition accuracy of each action via a confusion matrix. All actions are identified with an accuracy of 96% or higher, with a significant majority being correctly identified with a precision of 100%.

Case2: Air Painting. We develop air painting using mmEgo which can understand the fine-grained hand motions of the user. We emulate a VR/AR gaming scenario, where the user draws various shapes with a hand in the air including (1) "O", (2) "X", (3) "✓", (4) "□", (5) "△", and (6) "-" as the commands to control objects in the virtual environment. Fig. 15 compares the predicted hand trajectories by mmEgo and the ground truth. The estimated hand trajectories reproduce the true semantics well. Furthermore, we adopt an ST-GCN network (similar to action recognition) for air painting classification, with the only difference being that the arm poses rather than the whole-body pose is used as the input. Table 9 compares recognition accuracy with predicted arm pose with ground truth arm pose. The predicted arm pose achieves an F1 score and accuracy of 91%, which is 3% less than the ground truth. This suggests the potential to use egocentric arm pose estimation for human and VR/AR interactions.

To sum up, the case studies show the success of mmEgo in representative downstream tasks. Therefore, we envision that it has the potential to be a generic enabler for a wide range of applications.

6 DISCUSSION AND FUTURE WORK

This work presents the first proof-of-principle egocentric human pose estimation using radar. We notice that there are limitations to be further investigated in future extensions.

Global human pose estimation. mmEgo predicts the pose in the root-relative system. There might be applications that might require the global posture of the user (i.e., pose in the global coordinate system). mmEgo can be incorporated with the existing radar SLAM techniques [29] to localize the user and convert local pose to global pose, allowing for a more comprehensive understanding and interaction within the VR environment.

More complicated situations. We evaluate mmEgo with the collected dataset consisting of 13 representative activities. In the future, we plan to investigate more complex scenarios. We admit that the inherent randomness and diversity of human actions could impose new challenges, calling for more sophisticated designs. There are two challenging situations we will consider in our future work. (i) There are activities where the upper and lower body are loosely correlated. For example, people could wave their hands in either stand-up or sitting poses. When the correlation between upper and lower body movements is disrupted, our current model relies on the points on the lower limbs to predict the lower-body pose. We envision that our future work can exploit the context information to improve the results. For example, we could use previous sitting-down motions to predict the user is in the sitting pose. (ii) the user might carry items that could lead to a more significant occlusion of the lower limbers. In such cases, the material of the items will play an important role in the results. For the materials that mmWave can penetrate through (plastic, wood, cloth), our radar-based design could effectively obtain low-limb pose information even under significant occlusion, which is a unique advantage over the camera-based approaches. However, metallic items might be more challenging to handle for an RF-based approach and we require advanced signal processing and machine learning methods to mitigate their impacts.

Alternatives for IMU. The value of the work is more on the impact of egocentric radar on human pose sensing than radar motion tracking itself. IMU is used due to its low cost, robustness, and ubiquity. Alternative solutions such as cameras or LIDAR devices also achieve radar motion tracking in other suitable scenarios. In addition, an interesting topic in the future might be how to estimate radar motion using IMU data collected *offline* and radar data without external devices such as IMUs.

Implicit IMU fusion. In our design, we explicitly measure the radar motion and mitigate its impact on radar via coordinate alignments. Alternatively, we could also implicitly combine IMU measurement with radar using a data-driven approach, i.e., training a model that learns to perform radar motion mitigation. In our

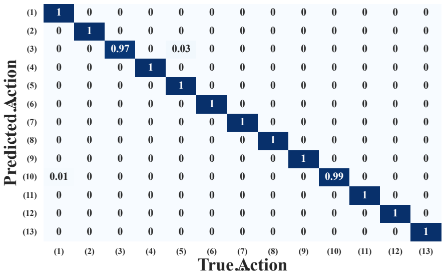


Fig. 14. Accuracy of action recognition.

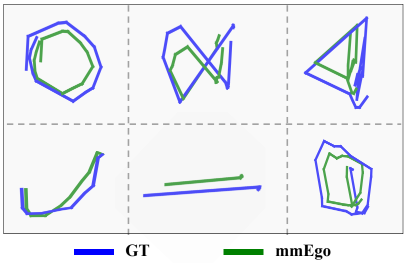


Fig. 15. Air painting.

Models	F1 (%)	Accuracy (%)
GT	**99.62**	**99.64**
mmEgo	99.61	99.63

Table 8. Action recognition.

Models	F1 (%)	Accuracy (%)
GT	**94.02**	**94.83**
mmEgo	91.81	91.11

Table 9. Air painting recognition.

future work, we will exploit various sensor fusion models (e.g., self/cross-attention) and customize them to accommodate the significant modal difference between radar and IMU.

Upper-body pose refinement. In general, the radar point clouds are more sparse on the lower body than they are on the upper body. This motivates our design that integrates upper-body pose and kinematics priors to predict lower-body pose. We notice that there could be specific activities (e.g., squat) in which the lower limbs are prominent in the radar point cloud, making lower-body pose easier to estimate. Therefore, our future work could use the lower-limb prediction to further refine the upper-body pose in such activities. We believe that dynamically adopting the optimization strategies for activities with different characteristics of point cloud distributions have the potential to further improve accuracy.

7 RELATED WORK

Egocentric Human Pose Estimation. Existing egocentric pose estimation designs primarily rely on camera [17, 19, 20, 33, 51, 61]. Earlier studies [33, 40, 51] employ head-mounted fisheye cameras and gradually reduce the distance between the camera and the user. Recent inside-in work, SelfPose [39], has achieved an AJE of 4.23 cm on the Human3.6M dataset [18]. With the advancement of SLAM and image processing technologies, there is a growing interest in outward-looking research, which shifting the focus to regular RGB cameras placed inside-out [17, 19, 20, 61] to capture self-pose, continuously lowering the lower bound of information required for reconstructing the human body. However, camera-based methods are susceptible to the impact of lighting and adverse weather conditions, as well as the risk of privacy breaches. Our work, on the other hand, utilizes radio frequency (RF) signals to perceive the human body, thus avoiding privacy concerns and exhibiting better robustness to environmental factors.

RF-based Human Pose Estimation. RF sensing has been extensively studied for various applications including pose estimation [32, 37, 43–45, 48, 55, 62–64]. Among them, recent works using mmWave radar achieve impressive performance. mmPose [36] converts the radar point cloud into an image and uses CNN to perform human pose estimation. Li et al. [27] utilizes a forked CNN to process the radar range-angle heatmaps for pose estimation. MARS [8] also uses a CNN for pose estimation after sorting the point cloud by coordinates. mmPose-NLP [35] analogizes the pose estimation task to natural language processing and extracts the skeleton from voxelized point clouds using a seq2seq structure. mmMesh [53] uses optimized PointNet as the backbone network to extract both global and local features for a finer-grained human mesh reconstruction. The authors' follow-up work M^{4}esh [52] extends the scene to multiple people by detecting individuals' bounding boxes on the radar heatmaps and addresses occlusion among subjects. m^{3}Track [26] also accomplishes multi-person pose tracking by preprocessing the range-angle heatmap with MVDR and using a conv-LSTM network for prediction. However, these aforementioned studies observe the subject from the outside-in view and thus do not suffer from random radar motion and scarcity of lower-body point clouds. In contrast, our work is the first to estimate the human skeleton from the egocentric view and overcome these challenges.

Pose Estimation from Sparse Wearable Sensors. Full-body Pose estimation using sparse wearables is also studied. SIP [42] was the first to use six IMUs placed on the head, arms, pelvis, and legs to estimate full-body pose with optimizations. DIP [16], TransPose [58], PIP [57] and TIP [24] further improve the accuracy, smoothness, and real-time performance of this setup. LoBSTr [56] considers fewer sensors, i.e., tracker information from 4 joints (head, two hands, torso), they used a GRU network to infer lower body motions from these joint signals. Recent advances [7, 13, 21, 49] further relax the input constraints to head and hand signals only. In contrast, our design only requires the user to wear a single head-mounted device. We leverage the remote sensing capability of radar for full-body pose estimation. Additionally, we designed the IMU tracking algorithm specifically for the compensation of the random motion noise in the radar point cloud.

8 CONCLUSION

This paper presents mmEgo, the first proof-of-concept egocentric human pose estimation design using mmWave radar. Novel designs are proposed to address the challenges of radar sensing from the egocentric perspective. We implemented mmEgo on the commodity mmWave radar and evaluated it on the representative activities. mmEgo achieves an average joint localization error of 4.3cm and an average rotation error of 4.9°. We envision that our approach holds significant potential in applications (e.g., VR/AR).

9 ACKNOWLEDGEMENTS

We thank the anonymous reviewers for their insightful feedback. This work was supported in part by Science and Technology Innovation 2030 - Major Project 2021ZD0114202, National Natural Science Foundation of China under Grant No. 62272098 and the Ministry of Education, Singapore, under its Academic Research Fund Tier 2 (MOE-T2EP20221-0017), National Research Foundation, Singapore and Infocomm Media Development Authority under its Future Communications Research and Development Programme.

REFERENCES

[1] 2023. Apple Vision Pro. https://www.apple.com/apple-vision-pro/.

[2] 2023. IWR6843ISK-ODS. https://www.ti.com.cn/tool/cn/IWR6843ISK-ODS.

[3] 2023. Microsoft HoloLens 2. https://www.microsoft.com/en-us/hololens.

[4] 2023. Sensor Capture + Azure Kinect + Refinement Workflow. https://www.depthkit.tv/tutorials/azure-kinect-microsoft-volumetric-capture-depth-workflow-depthkit.

[5] 2023. VALVE INDEX. https://store.steampowered.com/valveindex.

[6] 2023. VIVE Pro 2 Headset. https://www.vive.com/us/product/vive-pro2/specs/.

[7] Sadegh Aliakbarian, Pashmina Cameron, Federica Bogo, Andrew Fitzgibbon, and Thomas J Cashman. 2022. Flag: Flow-based 3d avatar generation from sparse observations. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 13253–13262.

[8] Sizhe An and Umit Y Ogras. 2021. Mars: mmwave-based assistive rehabilitation system for smart healthcare. ACM Transactions on Embedded Computing Systems (TECS) 20, 5s (2021), 1–22.

[9] Sizhe An and Umit Y. Ogras. 2022. Fast and Scalable Human Pose Estimation using mmWave Point Cloud.

[10] Yifeng Cao, Ashutosh Dhekne, and Mostafa H. Ammar. 2021. ITrackU: tracking a pen-like instrument via UWB-IMU fusion. In MobiSys. ACM, 453–466.

[11] Anjun Chen, Xiangyu Wang, Shaohao Zhu, Yanxu Li, Jiming Chen, and Qi Ye. 2022. mmBody Benchmark: 3D Body Reconstruction Dataset and Analysis for Millimeter Wave Radar. Proceedings of the 30th ACM International Conference on Multimedia (2022).

[12] Changhao Chen, Chris Xiaoxuan Lu, A. Markham, and Agathoniki Trigoni. 2018. IONet: Learning to Cure the Curse of Drift in Inertial Odometry. In AAAI Conference on Artificial Intelligence.

[13] Andrea Dittadi, Sebastian Dziadzio, Darren Cosker, Ben Lundell, Thomas J Cashman, and Jamie Shotton. 2021. Full-body motion from a single head-mounted device: generating SMPL poses from partial observations. In Proceedings of the IEEE/CVF International Conference on Computer Vision. 11687–11697.

[14] Hehe Fan, Yi Yang, and Mohan S. Kankanhalli. 2021. Point 4D Transformer Networks for Spatio-Temporal Modeling in Point Cloud Videos. 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2021), 14199–14208.

[15] Sachini Herath, Hang Yan, and Yasutaka Furukawa. 2020. RoNIN: Robust Neural Inertial Navigation in the Wild: Benchmark, Evaluations, & New Methods. In ICRA. IEEE, 3146–3152.

[16] Yinghao Huang, Manuel Kaufmann, Emre Aksan, Michael J Black, Otmar Hilliges, and Gerard Pons-Moll. 2018. Deep inertial poser: Learning to reconstruct human pose from sparse inertial measurements in real time. ACM Transactions on Graphics (TOG) 37, 6 (2018), 1–15.

[17] Dong-Hyun Hwang, Kohei Aso, Ye Yuan, Kris Kitani, and Hideki Koike. 2020. Monoeye: Multimodal human motion capture system using a single ultra-wide fisheye camera. In Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology. 98–111.

[18] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. 2014. Human3.6M: Large Scale Datasets and Predictive Methods for 3D Human Sensing in Natural Environments. IEEE Trans. Pattern Anal. Mach. Intell. 36, 7 (2014), 1325–1339.

[19] Hao Jiang and Kristen Grauman. 2017. Seeing invisible poses: Estimating 3d body pose from egocentric video. In 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 3501–3509.

[20] Hao Jiang and Vamsi Krishna Ithapu. 2021. Egocentric pose estimation from human vision span. In 2021 IEEE/CVF International Conference on Computer Vision (ICCV). IEEE, 10986–10994.

[21] Jiaxi Jiang, Paul Streli, Huajian Qiu, Andreas Fender, Larissa Laich, Patrick Snape, and Christian Holz. 2022. Avatarposer: Articulated full-body pose tracking from sparse motion sensing. In Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part V. Springer, 443–460.

[22] Wenjun Jiang, Chenglin Miao, Fenglong Ma, Shuochao Yao, Yaqing Wang, Ye Yuan, Hongfei Xue, Chen Song, Xin Ma, Dimitrios Koutsonikolas, Wenyao Xu, and Lu Su. 2018. Towards Environment Independent Device Free Human Activity Recognition. In MobiCom. ACM, 289–304.

[23] Wenjun Jiang, Hongfei Xue, Chenglin Miao, Shiyang Wang, Sen Lin, Chong Tian, Srinivasan Murali, Haochen Hu, Zhi Sun, and Lu Su. 2020. Towards 3D human pose construction using WiFi. In Proceedings of the 26th Annual International Conference on Mobile Computing and Networking. 1–14.

[24] Yifeng Jiang, Yuting Ye, Deepak Gopinath, Jungdam Won, Alexander W Winkler, and C Karen Liu. 2022. Transformer Inertial Poser: Attention-based Real-time Human Motion Reconstruction from Sparse IMUs. arXiv preprint arXiv:2203.15720 (2022).

[25] Thomas Kipf and Max Welling. 2016. Semi-Supervised Classification with Graph Convolutional Networks. ArXiv abs/1609.02907 (2016).

[26] Hao Kong, Xiangyu Xu, Jiadi Yu, Qilin Chen, Chenguang Ma, Yingying Chen, Yi-Chao Chen, and Linghe Kong. 2022. m3track: mmwave-based multi-user 3d posture tracking. In Proceedings of the 20th Annual International Conference on Mobile Systems, Applications and Services. 491–503.

[27] Guangzheng Li, Ze Zhang, Hanmei Yang, Jin Pan, Dayin Chen, and Jin Zhang. 2020. Capturing human pose using mmWave radar. In 2020 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops). IEEE, 1–6.

[28] Wenxin Liu, David Caruso, Eddy Ilg, Jing Dong, Anastasios I. Mourikis, Kostas Daniilidis, Vijay Kumar, and Jakob Engel. 2020. TLIO: Tight Learned Inertial Odometry. IEEE Robotics Autom. Lett. 5, 4 (2020), 5653–5660.

[29] Chris Xiaoxuan Lu, Muhamad Risqi U. Saputra, Peijun Zhao, Yasin Almalioglu, Pedro P. B. de Gusmao, Changhao Chen, Ke Sun, Niki Trigoni, and Andrew Markham. 2020. milliEgo: Single-chip mmWave Radar Aided Egomotion Estimation via Deep Sensor Fusion. international conference on embedded networked sensor systems (2020).

[30] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. 2017. Pointnet: Deep learning on point sets for 3d classification and segmentation. In Proceedings of the IEEE conference on computer vision and pattern recognition. 652–660.

[31] Davis Rempe, Tolga Birdal, Aaron Hertzmann, Jimei Yang, Srinath Sridhar, and Leonidas J. Guibas. 2021. HuMoR: 3D Human Motion Model for Robust Pose Estimation. In ICCV. IEEE, 11468–11479.

[32] Yili Ren, Zi Wang, Sheng Tan, Yingying Chen, and Jie Yang. 2021. Winect: 3D human pose tracking for free-form activity using commodity WiFi. Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies 5, 4 (2021), 1–29.

[33] Helge Rhodin, Christian Richardt, Dan Casas, Eldar Insafutdinov, Mohammad Shafiei, Hans-Peter Seidel, Bernt Schiele, and Christian Theobalt. 2016. Egocap: egocentric marker-less motion capture with two fisheye cameras. ACM Transactions on Graphics (TOG) 35, 6 (2016), 1–11.

[34] Mike Schuster and Kuldip K Paliwal. 1997. Bidirectional recurrent neural networks. IEEE transactions on Signal Processing 45, 11 (1997), 2673–2681.

[35] Arindam Sengupta and Siyang Cao. 2021. mmPose-NLP: A Natural Language Processing Approach to Precise Skeletal Pose Estimation using mmWave Radars. IEEE transactions on neural networks and learning systems PP (2021).

[36] Arindam Sengupta, Feng Jin, Renyuan Zhang, and Siyang Cao. 2020. mm-Pose: Real-time human skeletal posture estimation using mmWave radars and CNNs. IEEE Sensors Journal 20, 17 (2020), 10032–10044.

[37] Cong Shi, Li Lu, Jian Liu, Yan Wang, Yingying Chen, and Jiadi Yu. 2022. mPose: Environment-and subject-agnostic 3D skeleton posture reconstruction leveraging a single mmWave device. Smart Health 23 (2022), 100228.

[38] Scott Sun, Dennis Melamed, and Kris Kitani. 2021. IDOL: Inertial Deep Orientation-Estimation and Localization. In AAAI. AAAI Press, 6128–6137.

[39] Denis Tomè, Thiemo Alldieck, Patrick Peluse, Gerard Pons-Moll, Lourdes de Agapito, Hernán Badino, and Fernando De la Torre. 2020. SelfPose: 3D Egocentric Pose Estimation from a Headset Mounted Camera. IEEE transactions on pattern analysis and machine intelligence PP (2020).

[40] Denis Tome, Patrick Peluse, Lourdes Agapito, and Hernan Badino. 2019. xr-egopose: Egocentric 3d human pose from an hmd camera. In Proceedings of the IEEE/CVF International Conference on Computer Vision. 7728–7738.

[41] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. Advances in neural information processing systems 30 (2017).

[42] Timo Von Marcard, Bodo Rosenhahn, Michael J Black, and Gerard Pons-Moll. 2017. Sparse inertial poser: Automatic 3d human pose estimation from sparse imus. In Computer graphics forum, Vol. 36. Wiley Online Library, 349–360.

[43] Chuyu Wang, Jian Liu, Yingying Chen, Lei Xie, Hong Bo Liu, and Sanclu Lu. 2018. RF-kinect: A wearable RFID-based approach towards 3D body movement tracking. Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies 2, 1 (2018), 1–28.

[44] Fei Wang, Stanislav Panev, Ziyi Dai, Jinsong Han, and Dong Huang. 2019. Can WiFi estimate person pose? arXiv preprint arXiv:1904.00277 (2019).

[45] Fei Wang, Sanping Zhou, Stanislav Panev, Jinsong Han, and Dong Huang. 2019. Person-in-WiFi: Fine-grained person perception using WiFi. In Proceedings of the IEEE/CVF International Conference on Computer Vision. 5452–5461.

[46] Jian Wang, Lingjie Liu, Weipeng Xu, Kripasindhu Sarkar, and Christian Theobalt. 2021. Estimating egocentric 3d human pose in global space. In Proceedings of the IEEE/CVF International Conference on Computer Vision. 11500–11509.

[47] Shuai Wang, Dongjiang Cao, Ruofeng Liu, Wenchao Jiang, Tianshun Yao, and Chris Xiaoxuan Lu. 2023. Human Parsing with Joint Learning for Dynamic mmWave Radar Point Cloud. Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies 7, 1 (2023), 1–22.

[48] Yichao Wang, Yili Ren, Yingying Chen, and Jie Yang. 2022. Wi-Mesh: A WiFi Vision-Based Approach for 3D Human Mesh Construction. In SenSys. ACM, 362–376.

[49] Alexander Winkler, Jungdam Won, and Yuting Ye. 2022. QuestSim: Human Motion Tracking from Sparse Sensors with Simulated Avatars. In SIGGRAPH Asia 2022 Conference Papers. 1–8.

[50] Wenxuan Wu, Zhongang Qi, and Li Fuxin. 2019. Pointconv: Deep convolutional networks on 3d point clouds. In Proceedings of the IEEE/CVF Conference on

computer vision and pattern recognition. 9621–9630.

[51] Weipeng Xu, Avishek Chatterjee, Michael Zollhoefer, Helge Rhodin, Pascal Fua, Hans-Peter Seidel, and Christian Theobalt. 2019. Mo 2 cap 2: Real-time mobile 3d motion capture with a cap-mounted fisheye camera. IEEE transactions on visualization and computer graphics 25, 5 (2019), 2093–2101.

[52] Hongfei Xue, Qiming Cao, Yan Ju, Haochen Hu, Haoyu Wang, Aidong Zhang, and Lu Su. 2022. M^4esh: mmWave-Based 3D Human Mesh Construction for Multiple Subjects. In SenSys. ACM, 391–406.

[53] Hongfei Xue, Yan Ju, Chenglin Miao, Yijiang Wang, Shiyang Wang, Aidong Zhang, and Lu Su. 2021. mmMesh: towards 3D real-time dynamic human mesh construction using millimeter-wave. In MobiSys. ACM, 269–282.

[54] Sijie Yan, Yuanjun Xiong, and Dahua Lin. 2018. Spatial temporal graph convolutional networks for skeleton-based action recognition. In Proceedings of the AAAI conference on artificial intelligence, Vol. 32.

[55] Chao Yang, Xuyu Wang, and Shiwen Mao. 2020. RFID-pose: Vision-aided three-dimensional human pose estimation with radio-frequency identification. IEEE transactions on reliability 70, 3 (2020), 1218–1231.

[56] Dongseok Yang, Doyeon Kim, and Sung-Hee Lee. 2021. Lobstr: Real-time lower-body pose prediction from sparse upper-body tracking signals. In Computer Graphics Forum, Vol. 40. Wiley Online Library, 265–275.

[57] Xinyu Yi, Yuxiao Zhou, Marc Habermann, Soshi Shimada, Vladislav Golyanik, Christian Theobalt, and Feng Xu. 2022. Physical inertial poser (pip): Physics-aware real-time human motion tracking from sparse inertial sensors. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 13167–13178.

[58] Xinyu Yi, Yuxiao Zhou, and Feng Xu. 2021. TransPose: real-time 3D human translation and pose estimation with six inertial sensors. ACM Transactions on Graphics (Jul 2021), 1–13. https://doi.org/10.1145/3450626.3459786

[59] Fusang Zhang, Jie Xiong, Zhaoxin Chang, Junqi Ma, and Daqing Zhang. 2022. $Mobi^2$Sense: empowering wireless sensing with mobility. In MobiCom. ACM, 268–281.

[60] Zhengyou Zhang. 2000. A flexible new technique for camera calibration. IEEE Transactions on pattern analysis and machine intelligence 22, 11 (2000), 1330–1334.

[61] Dongxu Zhao, Zhen Wei, Jisan Mahmud, and Jan-Michael Frahm. 2021. EgoGlass: Egocentric-View Human Pose Estimation From an Eyeglass Frame. In 3DV. IEEE, 32–41.

[62] Mingmin Zhao, Tianhong Li, Mohammad Abu Alsheikh, Yonglong Tian, Hang Zhao, Antonio Torralba, and Dina Katabi. 2018. Through-wall human pose estimation using radio signals. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 7356–7365.

[63] Mingmin Zhao, Tianhong Li, Mohammad Abu Alsheikh, Yonglong Tian, Hang Zhao, Antonio Torralba, and Dina Katabi. 2018. Through-Wall Human Pose Estimation Using Radio Signals. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (2018), 7356–7365.

[64] Mingmin Zhao, Yonglong Tian, Hang Zhao, Mohammad Abu Alsheikh, Tianhong Li, Rumen Hristov, Zachary Kabelac, Dina Katabi, and Antonio Torralba. 2018. RF-based 3D skeletons. In Proceedings of the 2018 Conference of the ACM Special Interest Group on Data Communication. 267–281.

[65] Peijun Zhao, Chris Xiaoxuan Lu, Bing Wang, Niki Trigoni, and Andrew Markham. 2023. CubeLearn: End-to-End Learning for Human Motion Recognition From Raw mmWave Radar Signals. IEEE Internet Things J. 10, 12 (2023), 10236–10249.

[66] Jinxiao Zhong, Liangnian Jin, and Ran Wang. 2022. Point-convolution-based human skeletal pose estimation on millimetre wave frequency modulated continuous wave multiple-input multiple-output radar. IET Biometrics 11, 4 (2022), 333–342.

[67] Yi Zhou, Connelly Barnes, Jingwan Lu, Jimei Yang, and Hao Li. 2019. On the continuity of rotation representations in neural networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 5745–5753.