Multi-modal 3D Human Pose Estimation using mmWave, RGB-D, and Inertial Sensors

Anonymous Author(s)
Affiliation
Address
email

Abstract

The ability to estimate 3D human body pose and movement, also known as human pose estimation (HPE), enables many applications for home-based health monitoring, such as remote rehabilitation training. Several possible solutions have emerged using sensors ranging from RGB cameras, depth sensors, millimeter-Wave (mmWave) radars, and wearable inertial sensors. Despite previous efforts on datasets and benchmarks for HPE, few dataset exploits multiple modalities and focuses on home-based health monitoring. To bridge this gap, we present human pose estimation using multiple modalities with an in-house dataset[1]. We perform extensive experiments and delineate the strength of each modality.

1 Introduction

3D Human pose estimation (HPE) refers to detecting and tracking human body parts or key joints (e.g., wrists, shoulders, and knees) in the 3D space. It is a fundamental and crucial task in human activity understanding and movement analysis with numerous application areas, including rehabilitation [28, 20, 5, 4], professional sports [22], augmented/virtual reality, and autonomous driving [15]. In particular, human pose estimation plays an increasingly important role in healthcare applications, such as remote rehabilitation training [24, 11]. The current mainstream rehabilitation treatment involves a physical therapist supervising the patients in person. In contrast, HPE-based health monitoring systems can help clinicians correct patients' movements or instruct them remotely. To this end, multiple datasets have studied HPE with health-related physical movements [4, 28, 20, 5].

Many existing studies rely heavily on processing RGB frames from color cameras for human pose estimation [12, 3, 9, 21, 10, 13]. RGB image and video frames are the most common input types since they offer an non-invasive approach for HPE. However, the image quality depends heavily on the environmental setting, such as light conditions and visibility [1]. Moreover, using image and video data poses significant privacy concerns, especially in a household environment. Finally, the data-intensive nature of real-time video processing requires computationally powerful equipment with high cost and energy consumption.

Frame quality, privacy, and computational power drawbacks of video processing can be addressed by emerging *complementary sensor modalities*, such as lidar, millimeter wave (mmWave) radar [1, 34, 37], and wearable inertial sensors [31, 29, 30, 32, 17, 35], For instance, wearable inertial sensors provide accurate rotation and acceleration information regarding joints with low cost and computational power requirements [29, 30, 32], yet at a price of body worn sensors. mmWave radar can generate high-resolution 3D point clouds of objects while maintaining low cost, privacy, and computational power advantages.

[1]Dataset is accepted to NeurIPS 2022 Datasets&Benchmark track.

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

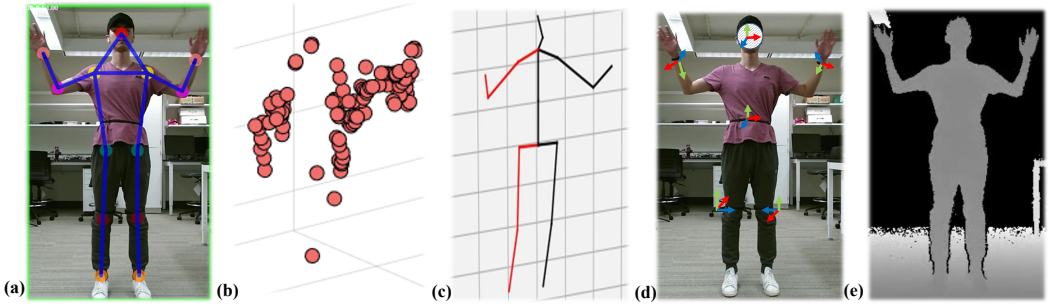

Figure 1: Overview of all modalities and human pose representations. All sub-figures uses the same sample frame during 'both upper limb extension'. (a) 2D human keypoints with bounding box on RGB image, (b) 3D mmWave point cloud, (c) 3D human skeletons, (d) IMU rotations, (e) depth image. Based on our work, researchers from the fields of machine learning, computer vision, wearable computing can exploit the complementary advantages of **multi-modality**, while clinical and rehabilitation experts can focus on its **healthcare** movements.

This work presents human pose estimation using multiple sensing modalities and exploit their complementary advantages. We also open-source a comprehensive 3D human pose estimation dataset performed by 20 human subjects, consisting of more than 160k synchronized frames from three sensing modalities. We hope that our work can contribute to the multi-modal machine learning community, and facilitate applications of HPE for rehabilitation and other healthcare problems.

2 Dataset

Overview. Our dataset includes 3D point cloud from mmWave, RGB frames and depth maps from RGB-D cameras, joints rotations and accelerations from wearable IMU sensors, as well as annotations of 2D keypoints, 3D joints, and action labels of 12 clinically relevant movements. Table 1 shows detailed specifications and properties of our sensors. More details can be found in the supplementary A.3.

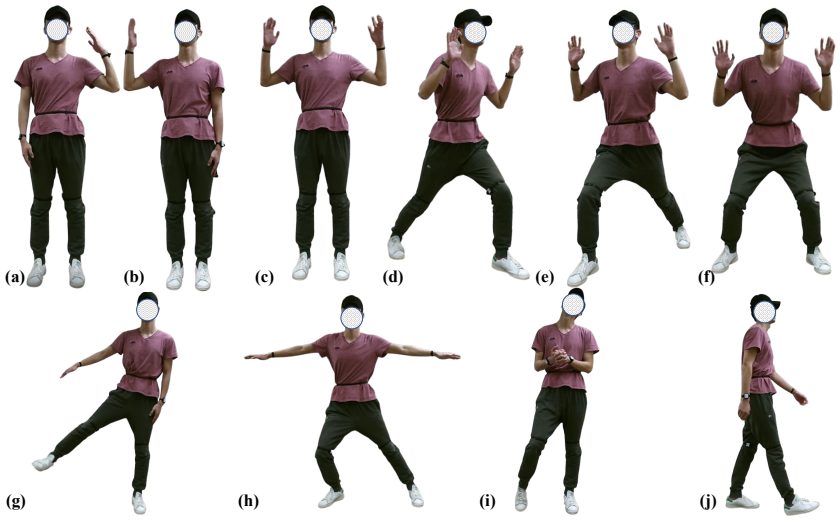

Figure 2: Overview of all rehab movements. The mirror movements of (g) and (h) are not shown due to limited space.

Rehabilitation exercises. We consider 12 movements related to rehabilitation exercises covering the entire human body. The first 10 rehabilitation movements are modified from [28, 1]. Figure 2 shows all movements: (a) left upper limb extension, (b) right upper limb extension, (c) both upper

	#	Freq.	Con.	Power	Privacy↑	Anti-inter.↑	Intrusive	Output
mmWave [25]	1	10 Hz	Wired	2.1 W	⋆⋆⋆	⋆⋆⋆	No	Point cloud
RGB [14]	2	30 Hz	Wired	16 W	⋆	⋆	No	RGB frame
Depth [14]	2	30 Hz	Wired	16 W	⋆⋆	⋆⋆	No	Depth and infra-red frame
IMU [33]	6	50 Hz	BLE	120 mW	⋆⋆⋆	⋆⋆⋆	Yes	Accelerations and quaternions

Table 1: **Comparison across sensors**. #: Number of sensors. Freq.: Sampling frequency. Con.: Type of connection to the host PC. Privacy indicates privacy-preserving ability. Anti-interference represents how much it is affected by environmental factors like non-ideal lighting.

limb extension, (d) left front lunge, (e) right front lunge, (f) squat, (g) left side lunge, right side lunge, (h) left limb extension, and right limb extension. The 11th and 12th movement are stretching and relaxing in free forms (11), and walking in a straight line (12), respectively. These two movements are meant to increase the diversity of the dataset, as the movement in 11 is determined by each subject and the movement in 12 features a global displacement. The duration of each type of movement is around 1 minute per subject.

3 3D Human Pose Estimation Evaluation

We introduce a standardized evaluation pipeline for 3D human pose estimation, use latest models to benchmark the performance of each modality, and discuss their results.

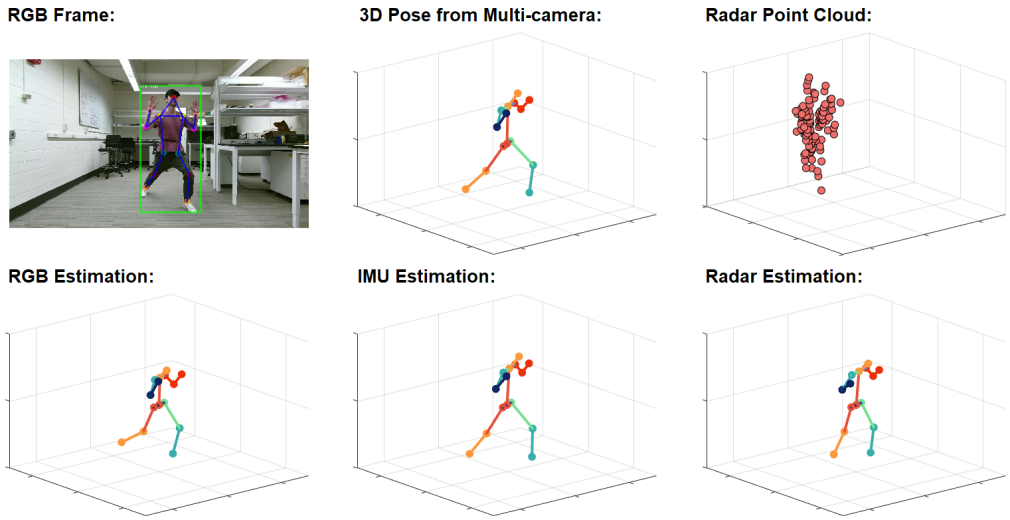

Figure 3: Visualization of sample pose data and results during left front lunge. Top row (from left to right): an RGB frame with detected human bounding box and 2D keypoints, the refined 3D pose derived from two cameras, and the 3D point cloud from mmWave radar. Bottom row (from left to right) shows the estimated 3D pose from a single RGB camera, IMU signals, and mmWave radar.

Experiment protocol. We consider two settings of data splits. **Setting 1 (S1 Random Split)**: A random split of 80% and 20% of all data is used as the training and testing set, respectively. **Setting 2 (S2 Split by Subjects)**: A randomly selected subset (80%, i.e., 16 out of 20) of the subjects is used for training, while the rest are for testing. S1 mimics a case where personalized HPE model is possible, while S2 evaluates across-subject generalization. For each setting, we randomly sample three splits and report the averaged results. More details are provided in the supplement A.3.

Further, we also define two evaluation protocols based on the design of our movements. **Protocol 1 (P1)** consists of all 12 movements, including stretching and relaxing in free forms and walking. While **Protocol 2 (P2)** only considers the first ten rehabilitation movements. Such protocols help us investigate the robustness of the model in terms of fixed/free form movement.

Evaluation metrics. We adopt Mean Per Joint Position Error (MPJPE) and Procrustes Analysis MPJPE (PA-MPJPE), widely used in human body pose estimation [9], as the main metrics. MPJPE

		Protocol 1		Protocol 2	
Modality	Setting	MPJPE (mm)↓	PA-MPJPE (mm)↓	MPJPE (mm)↓	PA-MPJPE (mm)↓
mmWave	S1	163.3±9.1	94.1±3.6	125.1±2.4	74.1±1.0
	S2	186.6±23.8	97.3±7.8	126.6±11.3	75.0±7.1
RGB	S1	116.9±0.1	66.8±0.2	115.0±0.1	64.4±0.1
	S2	120.1±3.7	67.5±1.9	118.4±3.8	64.7±1.4
IMUs	S1	**80.2**±12.6	**51.9**±1.9	**40.9**±1.0	**28.4**±0.9
	S2	147.4±18.4	74.5±5.9	94.3±13.8	54.0±4.9

Table 2: 3D human pose estimation results for mmWave, RGB, and IMUs. We report the mean and standard deviation of joint errors averaged across multiple splits under both our settings (**S1** & **S2**).

represents the mean Euclidean distance between ground truth and prediction for all joints. MPJPE is calculated after aligning the root joints (the pelvis) of the estimated and ground truth 3D pose. PA-MPJPE is MPJPE after being aligned to the ground truth by the Procrustes method [6], a similarity transformation including rotation, translation, and scaling.

Methods. We conduct 3D human pose estimation using mmWave, RGB, and IMUs separately using latest methods. Here we briefly introduce the methods considered in our evaluation and refer to our supplement for more implementation details.

- **mmWave**: We use the model from [1] that learns a convolutional neural network on the 5D point cloud to regress the 3D joints. The model is trained from scratch on our dataset, and outputs the 3D joints in the global coordinates system.
- **RGB**: We adopt the model from [16], where 2D keypoints from a sequence of frames are "lifted" into 3D joints (in the camera coordinate system) using a convolutional neural network. We use the pre-trained model from [16]. As the pre-trained model outputs a different set of joint, we only evaluate on a subset that intersects with our set of joints.
- **IMUs**: We employ the feature processing method from [35], with a convolutional neural network trained to regress rotations relative to a root joint (e.g., pelvis) using data from IMUs. The model is trained from scratch on our dataset.

Results and discussion. Table 2 shows the 3D HPE results for mmWave, RGB, and IMUs. Under **S1** and **P1**, mmWave-based HPE achieves 163 and 94 mm for MPJPE and PA-MPJPE, respectively. The metrics are further reduced to 125 and 74 mm for **P2**. IMU-based HPE obtains MPJPE and PA-MPJPE of 87 and 60 mm, respectively, under **S1** and **P1**. Figure 3 shows visualization comparison of estimation across different modalities.

Under **S2**, mmWave-based HPE performs similarly to **S1**, while IMU-based HPE obtains worse results than **S1**. This is because the sensing data from IMU is more fine-grained on the joints while mmWave grasps more information about body trunk, which is not too subject-specific. As a result, the IMU-based model is more sensitive to different subjects. We can observe that for all modalities **S2** yields higher standard deviations than **S1** since the difference between subjects is much more significant than random split, between train and test set. Similarly, **P1** yields higher standard deviations than **P2** since all movements in **P2** are fixed positions, which makes the model learning the keypoints distribution easier.

RGB-based HPE achieve 116 and 66 mm MPJPE and PA-MPJPE for **P1** under **S1**. Both data-split yield similar results. To compare, the same model achieves 36 mm PA-MPJPE on Human3.6M dataset. However, the model is trained and evaluated on Human3.6M while it is only evaluated on our dataset. In summary, all modalities perform reasonably well on our dataset.

4 Conclusion and future work

In this paper, we proposed health-related human pose estimation using multiple sensing modalities. Our results help to understand the advantages of individual sensing modalities in the context of home-based health monitoring. We hope that our work can catalyze the research including but not limited to pose estimation, multi-modal learning, and action understanding, thus facilitating critical applications in healthcare.

References

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

[2] S. An and U. Y. Ogras. Fast and scalable human pose estimation using mmwave point cloud. In *Proceedings of the 59th ACM/IEEE Design Automation Conference*, page 889–894, 2022.

[3] M. Andriluka, L. Pishchulin, P. Gehler, and B. Schiele. 2d human pose estimation: New benchmark and state of the art analysis. In *Proceedings of the IEEE Conference on computer Vision and Pattern Recognition*, pages 3686–3693, 2014.

[4] J. Antunes, A. Bernardino, A. Smailagic, and D. P. Siewiorek. Aha-3d: A labelled dataset for senior fitness exercise recognition and segmentation from 3d skeletal data. In *Prof. of The British Machine Vision Conference (BMVC)*, page 332, 2018.

[5] I. Ar and Y. S. Akgul. A computerized recognition system for the home-based physiotherapy exercises using an rgbd camera. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 22(6):1160–1171, 2014.

[6] K. Daniilidis. Pose from 3d point correspondences: The procrustes problem - pose estimation, 2022.

[7] V. Dham. Programming chirp parameters in ti radar devices. *Application Report SWRA553, Texas Instruments*, 2017.

[8] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask R-CNN. In *in Proc. of IEEE Intl. Conf. on Computer Vision*, pages 2961–2969, 2017.

[9] C. Ionescu, D. Papava, V. Olaru, and C. Sminchisescu. Human3. 6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *IEEE transactions on pattern analysis and machine intelligence*, 36(7):1325–1339, 2013.

[10] H. Joo, H. Liu, L. Tan, L. Gui, B. Nabbe, I. Matthews, T. Kanade, S. Nobuhara, and Y. Sheikh. Panoptic studio: A massively multiview system for social motion capture. In *The IEEE International Conference on Computer Vision (ICCV)*, 2015.

[11] Y. Li, C. Wang, Y. Cao, B. Liu, J. Tan, and Y. Luo. Human pose estimation based in-home lower body rehabilitation system. In *2020 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2020.

[12] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[13] D. Mehta, H. Rhodin, D. Casas, P. Fua, O. Sotnychenko, W. Xu, and C. Theobalt. Monocular 3d human pose estimation in the wild using improved cnn supervision. In *3D Vision (3DV), 2017 Fifth International Conference on*. IEEE, 2017.

[14] Microsoft. Kinect sensor. https://developer.microsoft.com/en-us/windows/kinect/ accessed 29 Sep. 2020, 2014.

[15] E. Odemakinde. Human pose estimation with deep learning - ultimate overview in 2021, Sep 2021.

[16] D. Pavllo, C. Feichtenhofer, D. Grangier, and M. Auli. 3d human pose estimation in video with temporal convolutions and semi-supervised training. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[17] G. Pons-Moll, A. Baak, T. Helten, M. Müller, H.-P. Seidel, and B. Rosenhahn. Multisensor-fusion for 3d full-body human motion capture. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2010.

[18] Pytorch. Pytorch Mobile. https://pytorch.org/mobile/home/ accessed 8 Jul. 2021, 2022.

[19] S. Rao. Introduction to mmwave sensing: Fmcw radars. *Texas Instruments (TI) mmWave Training Series*, 2017.

[20] A. Reiss and D. Stricker. Creating and benchmarking a new dataset for physical activity monitoring. In *Proceedings of the 5th International Conference on PErvasive Technologies Related to Assistive Environments*, pages 1–8, 2012.

[21] A. Shahroudy, J. Liu, T.-T. Ng, and G. Wang. Ntu rgb+d: A large scale dataset for 3d human activity analysis. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1010–1019, 2016.

[22] C. SIMON-AL-ARAJI. Bringing ai to the nba, 2019.

[23] K. Sun, B. Xiao, D. Liu, and J. Wang. Deep high-resolution representation learning for human pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5693–5703, 2019.

[24] T. Tao, X. Yang, J. Xu, W. Wang, S. Zhang, M. Li, and G. Xu. Trajectory planning of upper limb rehabilitation robot based on human pose estimation. In *2020 17th International Conference on Ubiquitous Robots (UR)*, pages 333–338. IEEE, 2020.

[25] Texas Instruments. IWR1443BOOST. https://www.ti.com/tool/IWR1443BOOST accessed 29 Sep. 2020, 2014.

[26] Texas Instruments. mmWavetutorial. https://www.ti.com/lit/pdf/swra553 accessed 29 Sep. 2020, 2014.

[27] Texas Instruments. mmWavefundamentals. https://www.ti.com/lit/spyy005 accessed 8 Apr. 2021, 2020.

[28] A. Vakanski, H.-p. Jun, D. Paul, and R. Baker. A data set of human body movements for physical rehabilitation exercises. *Data*, 3(1):2, 2018.

[29] T. von Marcard, R. Henschel, M. Black, B. Rosenhahn, and G. Pons-Moll. Recovering accurate 3d human pose in the wild using imus and a moving camera. In *European Conference on Computer Vision (ECCV)*, sep 2018.

[30] T. von Marcard, R. Henschel, M. J. Black, B. Rosenhahn, and G. Pons-Moll. Recovering accurate 3d human pose in the wild using imus and a moving camera. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 601–617, 2018.

[31] T. von Marcard, G. Pons-Moll, and B. Rosenhahn. Human pose estimation from video and imus. *Transactions on Pattern Analysis and Machine Intelligence*, 38(8):1533–1547, Jan. 2016.

[32] T. Von Marcard, B. Rosenhahn, M. J. Black, and G. Pons-Moll. Sparse inertial poser: Automatic 3d human pose estimation from sparse IMUs. In *Computer Graphics Forum*, volume 36-2, pages 349–360. Wiley Online Library, 2017.

[33] Wit-motions. BWT901CL. https://www.wit-motion.com/9-axis/witmotion-bluetooth-2-0-mult.html accessed 8 Apr. 2021, 2021.

[34] H. Xue, Y. Ju, C. Miao, Y. Wang, S. Wang, A. Zhang, and L. Su. mmmesh: Towards 3d real-time dynamic human mesh construction using millimeter-wave. In *Proceedings of the 19th Annual International Conference on Mobile Systems, Applications, and Services*, pages 269–282, 2021.

[35] X. Yi, Y. Zhou, and F. Xu. Transpose: Real-time 3d human translation and pose estimation with six inertial sensors. *ACM Transactions on Graphics*, 40(4), 08 2021.

[36] C. Zhang, J. Wu, and Y. Li. Actionformer: Localizing moments of actions with transformers. In *European Conference on Computer Vision*, 2022.

[37] M. Zhao et al. Rf-based 3d skeletons. In *Proc. of Conf. of the ACM Special Interest Group on Data Communication*, pages 267–281, 2018.

A Supplementary Materials

This document complements the main paper by describing: (1) results of pose estimation using additional metrics related to rehabilitation (A.1); (2) an analysis of our 3D pose refinement used to obtain ground-truth pose for our dataset (A.2); (3) details of our implementation and benchmark (A.3); (4) details of mmWave imaging (A.4); and (5) visualization of our pose estimation results (A.5).

For sections, figures, tables, and equations, we use numbers (e.g., Table 1) to refer to the main paper and capital letters (e.g., Table A) to refer to this supplement.

A.1 Further Analysis of Pose Estimation Results

We report additional evaluation metric, the mean average error (MAE) of joints angle to supplement our main results in the paper (using MPJPE and PA-MPJPE). The metric is widely considered to evaluate rehabilitation-specific movements — a main focus of our dataset. We only consider **Protocol 2** here since it has all rehabilitation-related movements.

Joints angle. We use the joint coordinates estimated by our models to find the angles between critical joints. We focus on the four commonly used joint angles: left & right elbow angles and left & right knee angles. The elbow angle is found using the shoulder, elbow, and wrist positions. First, we obtain the bone length between the shoulder and elbow and the length between the elbow and wrist using joint coordinates. Then, the angle is calculated using triangulation from the law of cosines. Similarly, the knee angles are obtained using the hip, knee, and ankle positions. The ground truth angle is computed using the refined ground truth 3D coordinates, and we calculated MAE between the ground truth and each modality. Table A shows detailed results of joins angle MAE. We observe that under **S1**, RGB modality yields below 10° for all joints, while mmWave and IMUs lead to larger errors regarding the elbow angles (>10°). This behavior is observed since the movement of the upper limbs is larger than that of the lower limbs for most movements. The setting of **S2** yields higher errors than under **S1** since **S2** requires the model to generalize to unseen subjects, which is arguably more challenging.

Modality	Setting	Left elbow	Left knee	Right elbow	Right knee
mmWave	S1	18.7±0.2	2.9±0.1	16.0±0.2	3.2±0.1
	S2	24.5±2.3	10.4±1.3	22.9±2.9	11.6±1.3
RGB	S1	9.0±0.1	8.3±0.1	**9.3**±0.1	7.7±0.1
	S2	11.5±0.6	14.8±1.6	11.1±0.7	14.0±1.5
IMUs	S1	**7.9**±0.1	**2.6**±0.1	11.3±0.2	**2.4**±0.1
	S2	8.4±1.0	5.6±0.2	9.7±0.8	5.7±0.2

Table A: MAE of joints angle (°) for mmWave, RGB, and IMUs. We report the mean and standard deviation of MAE averaged across multiple splits under both our settings (**S1** & **S2**).

A.2 Analysis of 3D Joints Quality

To validate the reliability of the obtained 3D joints, we further annotate a subset of the whole dataset and calculate the error. Specifically, we manually annotate 2D keypoints of the images, randomly sampled from subjects and movements. Then, we obtain the re-project 2D keypoints using refined 3D joints and camera parameters via camera calibration. Finally, we calculate the mean absolute percentage error (MAPE), and the percentage of correct keypoints threshold at 50% of the head segment length (PCKh) between the 2D keypoints from the model and the re-projection. The MAPE is 1.5%, and PCKh is 98.92. These quantitative results show that the proposed method of obtaining 3D joints is reasonably accurate. Figure A compares manual annotated 2D keypoints and re-projected 2D keypoints from refined 3D joints. Blue dots represent manual annotations, and red dots show the re-projection keypoints. We can observe that keypoints from the two methods almost overlap.

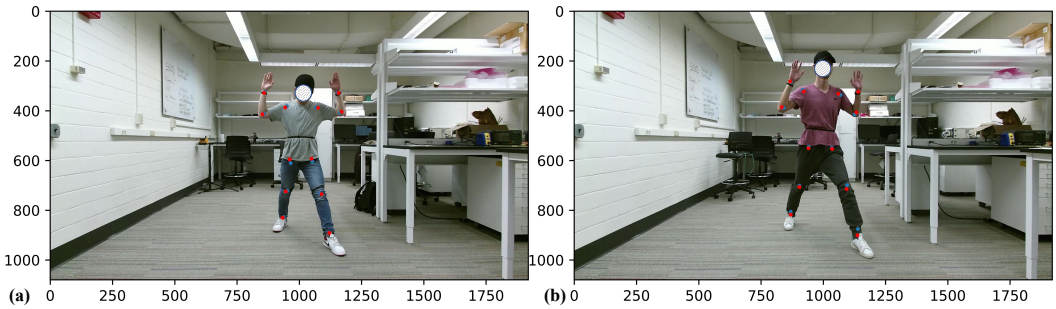

Figure A: A comparison of manual annotated 2D keypoints and re-projected 2D keypoints from refined 3D joints. Blue dots represent manual annotations and red dots show the re-projection keypoints.

A.3 Benchmark and Implementation Details

We now describe implementation details of methods considered in our benchmark. We use PyTorch [18] to implement all our models. Intel Xeon Gold 6242R @ 80x 4.1GHz and NVIDIA GeForce RTX 3090 are used to train these models. The code and pre-trained models will be open-sourced to facilitate the research area. Both raw data and synchronized data are released to the public as well. All material published is made available under the following Creative Commons license: Attribution-NonCommercial 4.0 International (CC BY-NC 4.0).

Data-split. For **Setting 1 (S1 Random Split)**, we set three different random seeds to split the data to 80% and 20% as training and testing set, respectively. For **Setting 2 (S2 Split by Subjects)**, we selected subset of the subjects to split the data, generated by three random seeds as well. Three different splits we used in the paper are shown as follows: (1) $[17, 13, 11, 15]$, (2) $[9, 7, 20, 8]$, and (3) $[3, 16, 7, 2]$. For example, $[17, 13, 11, 15]$ means that subject 17, 13, 11, 15 are used for testing and the rest for training.

mmWave-HPE. We follow [1] for the implementation for mmWave-HPE model. The input layer of the CNN takes the stacked 5-channel feature tensors. Two consecutive convolution layers follow the input layer with 16 and 32 channels, respectively. After the convolutions, the output is fed to the first fully-connected (FC) layer with 512 neurons. The final output of CNN contains 51 neurons, representing 3D coordinates for the 17 joints. All activation functions are Relu except for the final FC layer, where we use linear activation. Dropout layers are used after the convolution and fully connected layers to avoid excessive dependency on specific neurons. The model converges within around 50 epochs with early stopping settings.

RGB-HPE. We employ HRNet-W32 [23] (with bounding boxes from Mask RCNN [8]) to detect 2D keypoints of human body parts in all RGB frames from both cameras. W32 in HRNet represents the width of the high-resolution nets in the last three stages. The pre-trained model from [16] is utilized for 3D joints estimation. It "lift" 2D keypoints from a sequence of frames into 3D joints.

IMU-HPE. We follow [35] for IMU calibration, normalization, and features generation. Each IMU has its own coordinate system. As a result, two steps are needed to make the output compatible with neural network models. First, *calibration:* transforming the raw inertial measurements into the same reference frame. Second, *normalization:* transforming the leaf joint inertia into the root's space and scaling it to a suitable size for the network input. This method calculates the transition matrices for each sensor before capturing the movements, and it requires subjects to perform a 'T pose' before the experiments. The feature tensors extracted and transformed by this method capture the joint rotation and acceleration effectively such that multilayer perceptron (MLP) or CNN can regress the 3D joints with these features. We use a similar model as mmWave-HPE, except the input tensors are only 1-channel feature tensors for IMUs. The model converges within around 30 epochs with early stopping settings.

Skeleton-based action detection. We re-purpose an existing model [36] for the skeleton-based action detection. Specifically, the model takes a sequence of estimated 3D poses from individual modality as inputs. These poses are further encoded into a feature pyramid using a multi-scale

Symbol	Description	Values	Symbol	Description	Values
f_c	Starting frequency	77 GHz	θ_{res}	Angle resolution	9.55°
T_c	Chirp signal duration	32 μs	N_{RX}	No. of RX antennas	4
B	Bandwidth	3.20 GHz	N_P	Maximum points detectable per frame	64
S	Slope of chirp signal	100 MHz/μs	N_{TX}	No. of TX antennas	3
N	No. of chirps per frame	96	v_{res}	Velocity resolution	0.35 m/s
d_{res}	Range resolution	4.69 cm	v_{max}	Maximum Velocity	5.69 m/s

Table B: List of major parameters and variables related to mmWave and their values for mmWave point cloud generation.

transformer. Shared classification and regression heads check the feature pyramid, thus producing an action candidate at every timestamp.

A.4 mmWave Imaging

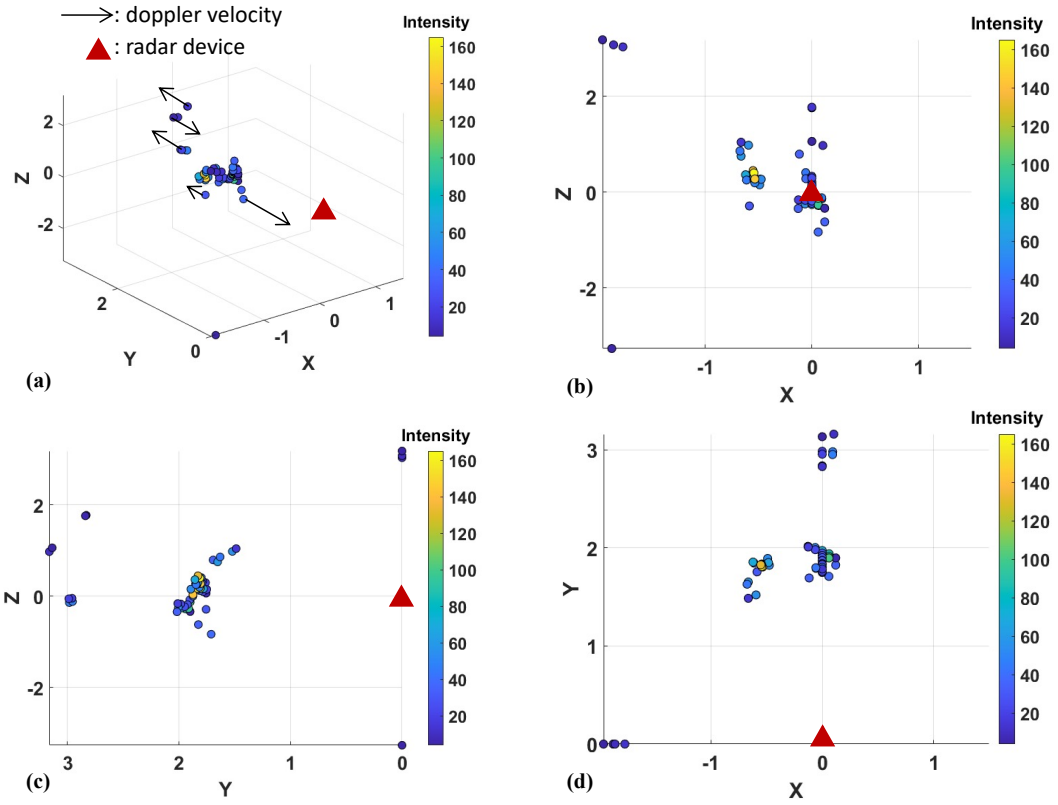

Figure B: mmWave point cloud representation for one frame. (a), (b), (c), and (d) shows the 3D view, front view, side view, and top view, respectively.

We follow [1] for the mmWave point cloud generation including software and hardware setup, data pre-process, and follow [2] for fusing the continuous frames point cloud to reduce the effect of sparsity. For the comprehensive details and math derivation of mmWave imaging background, please refer to [19, 26, 27, 7]. Figure B shows a sample input frame from different views. The red marker represents the radar location. Figure B(a) shows that point positions in 3D view, while the other plots show the front view, side view, and top view. Specifically, Figure B(a) illustrates the Doppler velocity, which indicates the relative velocity from the detected point to the radar. The colors in the figures represent the energy intensity of the reflected signals. Table B lists the key parameters we used to generate the mmWave point cloud.

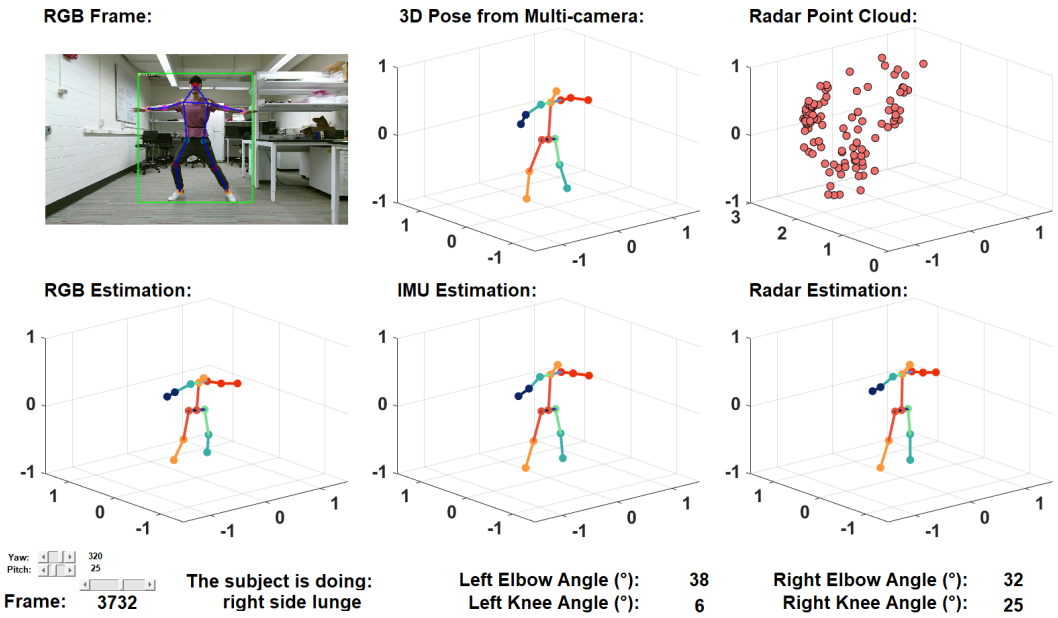

Figure C: Visualization of body poses from one subject performing right side lunge. The units in axes are meter.

A.5 Additional Visualization

Figure C shows one subject performing right side lunge. Figure D and E demonstrate pose estimation results from different camera pose. The results are displayed with the RGB frame from the camera, the refined 3D pose, and the 3D point cloud from mmWave radar. The first row, from left to right: RGB frame with detected human bounding box and 2D keypoints, the refined 3D pose from multiple cameras, and mmWave radar point cloud signal. The second row, from left to right: estimated 3D pose from a single RGB camera, IMU signals, and mmWave radar point cloud. The captions include the action label and four commonly used joint angles: left & right elbow angles and left & right knee angles.

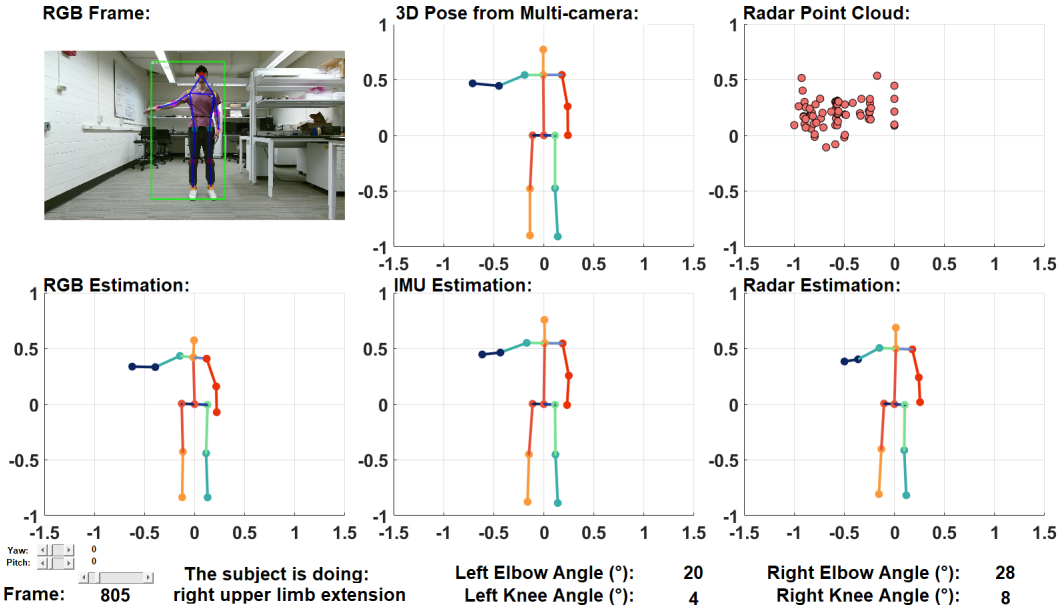

Figure D: Dataset visualization when $yaw = 0°, pitch = 0°$. The units in axes are meter.

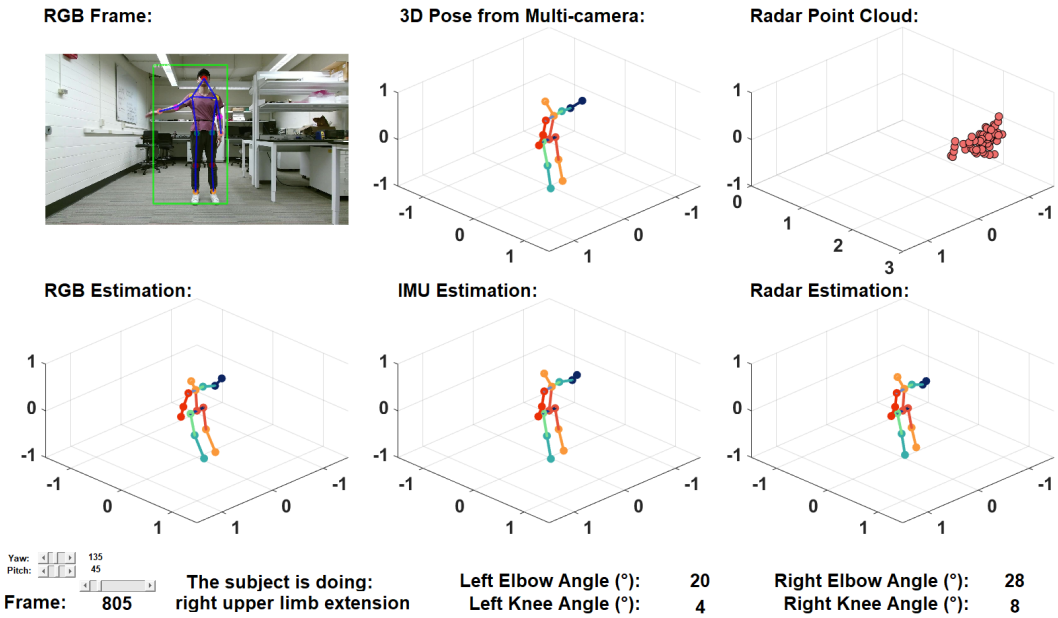

Figure E: Dataset visualization when $yaw = 135°, pitch = 45°$. The units in axes are meter.