TransHuPR: Cross-View Fusion Transformer for Human Pose Estimation Using mmWave Radar

Niraj Prakash Kini[1]
nirajnycu.ee06@nycu.edu.tw
Ruey-Horng Shiue[1]
raymondshiue.cs11@nycu.edu.tw
Ryan Chandra[1]
ryanchandra2015.ee11@nycu.edu.tw
Wen-Hsiao Peng[1]
wpeng@cs.nycu.edu.tw
Ching-Wen Ma[1]
machingwen@nycu.edu.tw
Jenq-Neng Hwang[2]
hwang@uw.edu

[1] National Yang Ming Chiao Tung University
Hsinchu, Taiwan
[2] University of Washington
Seattle, Washington, USA

Abstract

We present a novel Cross-View Fusion Transformer for Human Pose Estimation task based on mmWave Radar (TransHuPR). It is an mmWave Radar-based 2D Human Pose Estimation (HPE). Our work incorporates a 2D front projection view of the 3D pointcloud representation of the radar data as an input modality. The fusion transformer effectively fuses features derived from 2D front projection views of 2 independent radars and delivers high-quality predictions of human pose keypoints. We also introduce a new dataset consisting of fast actions with high frame rates as continuous radar sequences. Unlike other publicly available datasets, our dataset stands out because of its size, which ensures good generalization. We also incorporate single-action and mixed-action sequences, making the dataset more challenging. We use a non-expensive multi-radar system, which can be easily replicated. Our proposed method demonstrates significant improvements over existing methods in terms of both average precision scores and qualitative analysis. The dataset and code are available at https://github.com/nirajpkini/TransHuPR

1 Introduction

One of the most explored applications of computer vision is Human Pose Estimation (HPE). The aim is to predict the coordinates of body keypoints (joints) in a 2D or 3D coordinate system. HPE is very well explored, but the focus has always been on the RGB-based HPE [10, 14]. In the RGB-based HPE, the input modality is an RGB image or a video.

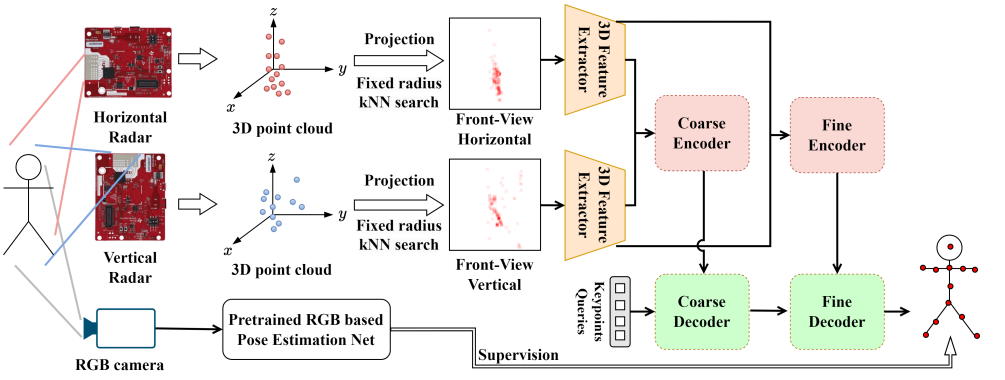

Figure 1: Illustration of our overall system setup.

Although RGB data are relatively easy to obtain, they come with limitations. The quality of the RGB data is highly dependent on external lighting conditions. Bright daylight versus low light conditions at night can make a huge difference in RGB data. Weather conditions such as rain, fog, snow, and dust can severely affect it too. Another important issue with RGB data is the inability to retain the subject's privacy in the scene, making RGB-based HPE unsuitable for environments where privacy retention has no alternative.

To overcome the limitations of RGB-based HPE, a new data modality, Radio Frequency (RF) signals, has been introduced in recent years. The application is RF-based HPE. Initially, WiFi signals were used to achieve HPE [20] and WiFi-based HPE has produced satisfactory results. However, there are two main drawbacks associated with WiFi-based HPE. First, there is no off-the-shelf WiFi hardware specifically designed for HPE purposes. All the WiFi-based HPE has to design custom hardware. This includes signal and data processing schemes, making it difficult to reproduce WiFi-based HPE settings. Second, WiFi frequency can penetrate walls [9] and monitor human activity in other rooms, making them unfavorable for privacy retention.

Even though mmWave-based 2D HPE has been explored in recent years by quite a few researchers, there is a scarcity of publicly available datasets targeting mmWave radar signals as the input modality. There are two recently published datasets [2, 5, 19], but they have their shortcomings. The mmBody dataset [2] adopts a high resolution radar module, which is as expensive as a medium resolution LiDAR sensor. In addition, it focuses on the mesh reconstruction task, and does not provide any benchmark results on the HPE task. On the other hand, the mm-Fi dataset [19] uses an inexpensive radar module. It features multi-modality sensor data (Infrared, LiDAR, mmWave Radar, WiFi). However, the actions provided in the dataset are simple actions with low frame rates and less than 300 frames per action, making it not so challenging on the HPE task. Furthermore, the pointclouds reconstructed from the sensor data are extremely sparse. Both [2, 19] use a single radar sensor. HuPR dataset [5] focuses primarily on the mmWave radar modality and 2D HPE task. It however has low frame rates, short sequences, only 4 subjects, and few fast moving objects.

This work introduces a custom dataset that overcomes these shortcomings. Notably, it includes all the raw radar data and detailed radar parameters used for dataset collection. More importantly, we generate medium resolution pointclouds from raw radar data. These pointclouds are useful for exploring both 2D and 3D HPE. Our dataset includes approximately 800k frames, with simple to complex mixed actions in single sequence, generated

Table 1: Radar settings

Parameters	Values	Parameters	Values
Start Frequency	77 GHz	Ramp End Time	65 μs
ADC Start Time	6 μs	Chirp Loops	256
Frequency Slope	60.012	Periodicity	33.33 ms
ADC Samples	256	Transmitters	3
Sample Rate	4400 ksps	Receivers	4
Rx Gain	30 dB	Total Frames	1800
Idle Time	7 μs	FPS	30

with 20 subjects. We use two inexpensive off-the-shelf radar sensors (one for horizontal scanning and the other for vertical scanning) and showcase in the ablation study that two radars system is significantly better than one radar system.

In addition, this work provides a benchmark method, known as TransHuPR, that aims at answer the question: "instead of directly using 3D pointcloud as an input modality, can 2D projections of 3D pointcloud help?". Currently, there are 2 modalities in mmWave-based HPE: 2D heatmaps and 3D pointclouds. The 2D heatmap representations are Range-Azimuth (RA map) and Range-Elevation (RE map). These modalities have been used and shown preliminary results on 2D/3D HPE. 3D pointcloud representations use Azimuth-Range-Elevation (XYZ) and is passed through CFAR algorithm [4] for peak detection. As compared to the heatmap representation, the pointcloud representation is less noisy. [11, 12] showcased accurate HPE results on pointclouds but failed to predict difficult keypoints (e.g. elbow and wrist) accurately. Our TransHuPR is designed to be easily extended to accommodate pointclouds, output by the radar sensors. Our overall system setup is depicted in Fig. 1.

Specifically, our contributions include: (1) a high-quality mixed action dataset, focusing primarily on mmWave radar-based 2D HPE task, (2) using temporally consistent density-based 2D projections of 3D pointcloud representations of radar data for 2D HPE, (3) proposing cross-view fusion transformer (CVF) that fuses 2D projections of horizontal and vertical sensors.

2 Related Work

2.1 Transformer-based RGB HPE

There are many studies on Transformer-based HPE with RGB images as input. TFPose [8] directly predicts keypoint coordinates using a CNN backbone and a Transformer. Other previous works [7, 16] employ Vision Transformer as a part of their architectures. PETR [13] introduces multi-scale feature maps into a Transformer-based architecture to conduct multi-person HPE. PRTR [6] proposes HPE based on DETR [1], which includes the Hungarian algorithm to match object queries and ground truths. All these previous works deal with only RGB images. In this paper, we propose cross-view fusion with Transformer-based HPE on pointcloud FVs as inputs.

Table 2: Comparison of the different mmWave datasets.

Dataset	Actions	Sub	Frames/seq	Total Frames	FPS	Radars	Task
mmBody [2]	100	20	-	200k	10	1	3D Mesh
mmFi [19]	27	40	297	320k	14	1	3D HPE
HuPR [5]	-	4	600	141k	10	2	2D HPE
TransHuPR (Ours)	22	20	1800	792k	30	2	2D HPE

2.2 CNN-based mmWave HPE

Most works on mmWave-based HPE use 2D or 3D CNN as basic building blocks [12]. HuPR [5] uses heatmaps as the input modality and 3D CNN to learn temporal consistency and spatial features. The 3D pointcloud modality [11] is also proven useful, but provides mediocre quality predictions. mmMesh [17] is a mesh reconstruction method, but it has skeletal loss associated with HPE, and the combination of MLP and 3D CNN was shown effective in mesh reconstruction. Even though the CNN-based mmWave HPE architectures have been explored significantly, the overall accuracy remains substandard, and the highly accurate predictions of hard keypoints are still an issue to solve.

2.3 Transformer-based mmWave HPE

Recently, [18, 22] use Transformer to conduct HPE on radar-based input data. RadarFormer [21] directly processes radar echoes and conducts self-attention, yielding comparable results to CNN-based schemes that process radar images, such as pointclouds. However, radar echoes suffer from noise, which affects the quality of input data. MPTFormer [3] adopts a dual-view mmWave radar setup, which has a side-view and a front-view radar. Although these works show interesting results on HPE, they only work on single-scale features. Our work explores a cross-view, multi-scale Transformer architecture for HPE.

3 Dataset

Hardware Settings: We design a radar hardware board to collect a custom dataset for the 2D HPE task. We use two radars IWR1843BOOST and an on-board smartphone camera. One radar (horizontal) is setup to scan the horizontal view of the scene, and the other (vertical) for the vertical view. The vertical radar is oriented 90 degrees clockwise with respect to the horizontal radar. The on-board camera captures the RGB videos of the subject and their actions/postures. These RGB videos are then used to generate the ground-truth 2D keypoints, with the help of the state-of-the-art 2D pose estimation network [14]. The radar settings are described in Table 1.

Dataset Processing: The raw data collected by two radar sensors are processed to get pointclouds. We incorporate the density-based 2D front-view projection of the 3D pointclouds. For each pixel in the 2D projection, the nearest neighbors (NN) are counted within a fixed radius r. The number of kNN is assigned to that pixel. The Aziumth-Elevation (X-Z) is termed Front View (FV) and the Azimuth-Range (X-Y) is termed Bird's Eye View. The 2D projections of 3D pointclouds allow us to use CNN to extract features.

There are 20 subjects and 22 sequences per subject in our custom dataset. Each action is one sequence consisting of 1800 frames, which are captured at 30 frames per second for

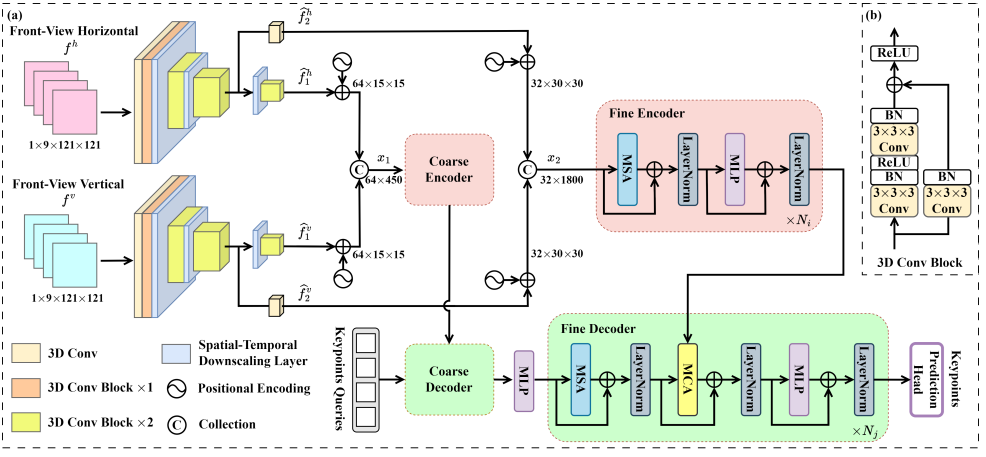

Figure 2: Illustration of our proposed TransHuPR: (a) the overall architecture of TransHuPR, where the two encoder-decoder pairs share the same network architecture, and (b) the 3D Conv Block.

1 minute. All the actions are fast actions, meaning that either the subject or a body part of the subject is always in motion. The 22 sequences are divided into 3 categories. The first category has 12 sequences of single actions, the second category has 6 sequences of 2 mixed actions, and the third has 4 sequences of 3 mixed actions. The mixed actions are combinations of single actions. The 12 single actions are walking in place, waving left hand, walking back and forth, waving right hand, lunges, making front-up-side-down movement of both hands, walking clockwise, walking counterclockwise, standing torso twist, one leg balance, waving both hands, and flapping both hands. The dataset contains a total of 792,000 frames. The comparison of publicly available mmWave-based datasets is described in Table 2.

4 Proposed Method

4.1 System Overview

Fig. 2(a) illustrates our TransHuPR. It has two major parts: 3D CNN feature extractors and cross-view, multi-scale fusion Transformer (CVF). As shown, $f^h, f^v \in R^{1\times9\times121\times121}$ denote the pointcloud FVs from the horizontal and vertical radars, respectively. In our design, f^h, f^v each contains nine 1-channel FV frames of size 121×121. Temporal information from these FV frames is fused and transformed into multi-scale features by two 3D CNNs, one for each radar. The features at each scale and from each radar are converted into discrete tokens, which are then fused and processed by CVF in a coarse-to-fine manner through encoder-decoder pairs. Inspired by [15], which is for low-level line detection, our coarse-to-fine encoder-decoder architecture successively updates the initial keypoint queries to arrive at the final keypoint predictions.

4.2 Multi-scale Spatio-temporal Feature Extraction

Our 3D CNN is used as a feature extractor to generate multi-scale features. It fuses information from both spatial and temporal dimensions of the pointcloud FVs f^h, f^v. It consists of multiple 3D Conv Blocks (Fig. 2(b)) and downscaling layers. The downscaling is performed in 3D and has a scale of 0.5. Since the horizontal branch is a copy of the vertical branch, we take only the vertical branch for exposition.

In the vertical branch, the 3D CNN generates two outputs $\hat{f}_1^v \in R^{64\times15\times15}$ and $\hat{f}_2^v \in R^{32\times30\times30}$, where an additional 3D CNN with the kernel size $2\times1\times1$ is added to aggregate the remaining temporal information to get $\hat{f}_2^v$. The 3D CNN in the horizontal branch functions similarly and generates $\hat{f}_1^h \in R^{64\times15\times15}$ and $\hat{f}_2^h \in R^{32\times30\times30}$. In our design, a separate 3D CNN is used for each radar sensor. The two 3D CNNs do not share weights, as notable differences exist between their inputs.

Tokens are formed and input to the next module. We flatten the features, $\hat{f}_1^h, \hat{f}_1^v, \hat{f}_2^h, \hat{f}_2^v$, and merge those of the same scale to form $x_1 \in R^{64\times450}$ and $x_2 \in R^{32\times1800}$.

4.3 Cross-View Fusion Transformer (CVF)

We now explain our Cross-View Fusion Transformer (CVF) in detail. Inspired by DETR [1], a Transformer-based object detection scheme, our CVF follows a similar encoder-decoder architecture yet with several notable differences. These include: 1) fixing the number of object queries at 14, which is the number of keypoints in our task; 2) removing the bipartite matching loss during training since the Hungarian algorithm is not used; 3) modifying the positional embeddings for cross-view fusion; and (4) introducing multi-scale encoder-decoder pairs.

Keypoint Queries: The initial keypoint queries in our CVF are learnable parameters. They are similar to the object queries in Transformer-based object detection schemes, such as DETR [1]. However, our task is keypoint detection. The number of keypoints is always fixed at 14. In comparison, with DETR [1], a maximum number of object queries is pre-determined and the Hungarian algorithm is used during training to match the object predictions and the ground-truths. This is not required in our scheme.

Fixed Positional Embeddings: Similar to DETR [1], our TransHuPR adopts fixed instead of learnable positional embeddings. The sine and cosine positional embeddings allow our model to differentiate positions through varying frequencies and offsets without additional parameters. The inputs of TransHuPR come from horizontal and vertical radars. These inputs are processed into multi-scale feature maps. Separate positional embeddings are applied to these two distinct inputs at different scales.

Transformer Encoder and Decoder: Using the Transformer-based encoder-decoder architecture, the keypoint queries learn relevant information from the input feature maps. The encoder and decoder are stacked with multiple encoder and decoder layers, respectively.

The encoder learns the pairwise relation of the input tokens through self-attention. Suppose x_0 is a group of input tokens to the encoder. Each encoder layer takes as input the tokens x_{l-1} from the previous layer. The output tokens x_l are obtained by

$$\begin{aligned} x'_l &= LN(MHSA(x'_{l-1}) + x'_{l-1})), \\ x_l &= LN(MLP(x'_l) + x'_l), \end{aligned} \tag{1}$$

where $l = 1, 2, ..., L$ is the layer index, MHSA is the multi-head self-attention, and LN is the layer norm. The output tokens have the same dimensions as the input tokens.

Table 3: Comparison results of the baseline methods and TransHuPR.

Model	Param.	AP								AP	AP50	AP75
		Head	Neck	Shoulder	Elbow	Wrist	Hip	Knee	Ankle	Total		
mmPose [17] (B)	14M	50.2	55.6	44.2	30.2	14.1	76.3	64.5	53.7	44.5	84.9	41.2
HRNet [24] (B)	18M	59.9	65.4	58.2	48.9	31.7	82.0	75.3	68.8	60.5	89.0	68.3
3D CNN D (Ours) (B)	137M	**65.7**	**70.1**	58.8	46.8	30.2	83.5	74.0	66.7	60.1	**93.0**	66.3
3D CNN S (Ours) (B)	69M	**65.7**	69.6	58.0	48.4	28.7	**84.1**	**77.4**	68.2	60.8	**93.0**	66.4
TransHuPR (Ours)	11M	60.1	66.8	**59.4**	**49.5**	**32.1**	81.6	75.8	**69.4**	**61.3**	89.2	**69.2**

The decoder follows the standard Transformer decoder architecture. Each decoder layer takes as input the tokens in x_L from the encoder along with the keypoint queries. The keypoint queries first go through self-attention, followed by cross-attention. Each query attends to different tokens in x_L.

Coarse-to-Fine Decoding: We adopt a coarse-to-fine strategy to update keypoint queries in a progressive manner. In the coarse decoding stage, we proceed with our initial keypoint queries (learnable parameters) and lower-resolution feature maps that are input to the encoder-decoder architecture. The coarse encoder performs self-attention among input tokens from low-resolution feature maps. The coarse decoder then updates the initial keypoint queries by having them interact with the tokens from the coarse encoder. The fine decoder acts similarly by having the updated keypoint queries from the coarse decoder attend to the tokens output by the fine encoder. In the process, a fully-connected layer matches the token size between the keypoint queries and fine-encoded tokens. The resulting keypoint queries are input to the prediction head to get the final keypoint predictions.

5 Experimental Results

Training Setup and Evaluation Metrics: The common train, validation, and test split protocol used in mmWave radar-based HPE [12, 17] is to divide each action sequence into three sets such that both subjects and action sequences of validation and test set are not completely unknown in the training set. Our proposed protocol makes the training process more challenging by ensuring that no part of any (subject, action sequence) pair is common in the training, validation, and test set. In other words training, validation, and test sets are completely independent of each other. Also, the validation and test set includes action sequences of the subjects completely unseen in the training set. Our custom, fast-actions dataset contains 440 sequences, each one minute long at 30 frames per second. The total number of pointcloud FVs per sequence is 1800. Following the common test protocol [17], the training, validation, and test sets have 633600, 79200, and 79200 pointcloud FVs. For evaluation, we report the Average Precision (AP) over Object Keypoint Similarity (OKS) of 2D keypoints. We have 3 variants, named AP, AP50, and AP75 respectively. AP is the average precision over 10 OKS thresholds (0.5, 0.55, 0.6, 0.65, 0.7, 0.75, 0.8, 0.85, 0.9, 0.95). AP50 and AP75 are strict and loose constraints, respectively. We adopt a regression-based training loss that aims at minimizing the mean-squared error between the keypoint predictions and their ground-truths.

Table 4: Comparison results of 1 radar (H) vs. 2 radars (H and V).

Radars	AP	AP50	AP75
1	51.9	85.4	54.9
2	61.3	89.2	69.2

Table 5: Comparison results of TransHuPR with and without multi-scale.

Scales	AP	AP50	AP75
1	60.2	89.1	66.6
2	61.3	89.2	69.2

5.1 Performance Comparison

Table 3 provides accuracy results for our TransHuPR and the baseline methods, which include mmPose [12], HRNet [14], shallow 3D CNN, and deep 3D CNN. mmPose uses a forked CNN architecture that takes radar reflection data projected onto $16 \times 16 \times 3$ XY and YZ planes as inputs. Each input is processed through a 3-layer CNN and a 3-layer MLP to produce the prediction results. Similarly, HRNet [14] processes radar reflection data through parallel multi-scale convolution layers, gradually adding high-to-low resolution feature maps. Before the input goes to HRNet branches, we fuse information along temporal dimensions by using the 3D CNN part of the TranHuPR. 2 branches of HRNet are utilized to process horizontal and vertical data separately and concatenate the outputs to generate heatmap prediction of each keypoints. For Shallow 3D CNN, we use only the 3D CNN part of the TransHuPR with a fully-connected decoder layer to predict the keypoints. In contrast, deep 3D CNN has two additional convolutional layers added to the shallow 3D CNN.

As shown, our TransHuPR outperforms mmPose [12] and shows comparable results with 3D CNN methods and HRNet [14]. Compared to mmPose, TransHuPR attains 16.8% and 7.2% in AP and AP50. The significant gains are resulted from three different aspects: the number of frames being processed, the multi-scale architecture, and the Transformer model. On the other hand, TransHuPR achieves very similar performance with HRNet, deep 3D CNN, and shallow 3D CNN in terms of AP scores. Note that TransHuPR has the smallest model size, with only 11M learnable parameters. To put things in perspective, the 3D CNN models have significantly more learnable parameters with shallow 3D CNN and deep 3D CNN being more than 6 times larger. Although the 3D CNN model and HRNet use the same method as TransHuPR in processing 9 frames simultaneously, their architectures only consist of convolution layers and MLP to generate keypoints prediction. In contrast, TransHuPR utilizes a multi-scale Transformer architecture to capture the global correlation between features in different scales. Interestingly, the elbow, wrist, and ankle keypoints are better predicted with TransHuPR than the 3D CNN models. The common thing among these keypoints is that they are far from the torso, and they move a lot. Therefore, TransHuPR strikes a better balance between detection accuracy and complexity among the competing methods.

Fig. 3 presents a qualitative comparison between mmPose [12], HRNet [14], the 3D CNN models, and our TransHuPR. More details of our proposed method and qualitative results are included in the supplementary material.

Figure 3: Subjective comparison between mmPose [12], HRNet [14], 3DCNN-S (Shallow), 3DCNN-D (Deep) and TransHuPR (ours).

5.2 Ablation Experiments

This section presents several ablation studies to show the importance of several settings contributing to the final performance.

Number of Radars: Table 4 presents ablation results showing how the number of radar sensors affects the detection accuracy. With only the horizontal sensor, TransHuPR has an AP score close to 52%. When incorporating both horizontal and vertical sensors, an improvement of 9.4% AP score was achieved. These results confirm that horizontal and vertical sensors capture distinct information that is useful for keypoint detection.

Number of Scales: Table 5 reports detection accuracy by varying the number of encoder-decoder scales. In this experiment, we remove the coarse encoder-decoder from the design to show how the multi-scale design contributes to the performance. We observe that without this multi-scale feature, the AP of TransHuPR is around 60.2%. When multi-scale is included, a higher AP score of 61.3% is reached. This suggests that even though the input tokens to the coarse encoder-decoder are from feature maps of a lower resolution, the information they carry differs from the input tokens of the higher resolution.

Number of Frames: Table 6 ablates different design choices on input frames. In our design, the number of input frames is chosen to be 9, which is 0.3 seconds since our dataset

Table 6: Comparison results of different number of frames used.

Frames	AP	AP50	AP75
5	59.1	89.0	65.9
7	59.5	89.2	67.3
9	61.3	89.2	69.2

Table 7: Comparison results of different cross-view fusion methods.

Fusion methods	Param.	AP	AP50	AP75
Early fusion (Ours)	11M	61.3	89.2	69.2
Late fusion	19M	61.5	89.2	69.3

has a framerate of 30FPS. The result shows that the best performance is achieved with 9 input frames. This also demonstrates that each neighboring frame in our dataset carries different information. Therefore, making use of the temporal information is quite important. Using more than 9 frames will probably bring more gain to the performance, but it also requires more future frames, making it less practical.

Fusion Techniques: Table 7 compares fusion methods to justify our design. In our proposed method, early fusion is used, in which the tokens from horizontal and vertical sensors are combined before being passed to the Transformer encoder. With late fusion, the tokens from each sensor are passed to a separate set of encoders and decoders. There are also two sets of keypoint queries, which are concatenated before the prediction head to obtain the final prediction. The result shows no clear difference in AP scores between the two methods. However, when late fusion is used, the number of learnable parameters is increased greatly. This is because two sets of encoders and decoders are used separately for the two sensors.

6 Conclusion

We introduce a custom fast action dataset. It consists of 792k frames, where 22 types of action sequences were performed by 20 subjects. It is collected from two radars of different orientations, and we have shown that this setup brings significant gain to keypoint detection accuracy. We also introduced the collection process. We propose TransHuPR, a Transformer-based HPE model with mmWave radar input. Experimental results show that TransHuPR outperforms the baseline methods, e.g. 3D CNN, HRNet [14], and mmPose [17], when using pointcloud FV as input. With TransHuPR, keypoints on the limbs can be better predicted while reducing the number of learnable parameters. We plan to release the dataset and code and extend TransHuPR to 3D HPE.

7 Acknowledgment

This work is supported by National Center for High-performance Computing, Taipei, Taiwan.

References

[1] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, May 2020.

[2] Anjun Chen, Xiangyu Wang, Shaohao Zhu, Yanxu Li, Jiming Chen, and Qi Ye. mmbody benchmark: 3d body reconstruction dataset and analysis for millimeter wave radar. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 3501–3510, 2022.

[3] Lin Chen, Xuemei Guo, and Guoli Wang. Mptformer: Towards robust arm gesture pose tracking using dual-view radar system. *IEEE Sensors Journal*, 23, Nov 2023.

[4] HM Finn. Adaptive detection mode with threshold control as a function of spatially sampled clutter-level estimates. *RAC Rev 414-465*, 1968.

[5] Shih-Po Lee, Niraj Prakash Kini, Wen-Hsiao Peng, Ching-Wen Ma, and Jenq-Neng Hwang. Hupr: A benchmark for human pose estimation using millimeter wave radar. In *IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, Jan 2023.

[6] Ke Li, Shijie Wang, Xiang Zhang, Yifan Xu, Weijian Xu, and Zhuowen Tu. Pose recognition with cascade transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2021.

[7] Y. Li, S. Zhang, Z. Wang, S. Yang, W. Yang, S. Xia, and E. Zhou. Tokenpose: Learning keypoint tokens for human pose estimation. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, Oct 2021.

[8] Weian Mao, Yongtao Ge, Chunhua Shen, Zhi Tian, Xinlong Wang, and Zhibin Wang. Tfpose: Direct human pose estimation with transformers. *arXiv preprint arXiv:2103.15320*, March 2021.

[9] Hang Zhao Tianhong Li Mohammad Abu Alsheikh Rumen Hristov Zachary Kabelac Dina Katabi Antonio Torralba Mingmin Zhao, Yonglong Tian. Rf-based 3d skeletons. *ACM Special Interest Group on Data Communication (SIGCOMM)*, Aug 2018.

[10] Dario Pavllo, Christoph Feichtenhofer, David Grangier, and Michael Auli. 3d human pose estimation in video with temporal convolutions and semi-supervised training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[11] Arindam Sengupta and Siyang Cao. mmpose-nlp: A natural language processing approach to precise skeletal pose estimation using mmwave radars. *IEEE Transactions on Neural Networks and Learning Systems*, page 267–281, Mar 2022.

[12] Arindam Sengupta, Feng Jin, Renyuan Zhang, and Siyang Cao. mm-pose: Real-time human skeletal posture estimation using mmwave radars and cnns. *IEEE Sensors Journal*, 20(17):10032–10044, May 2020.

[13] Dahu Shi, Xing Wei, Liangqi Li, Ye Ren, and Wenming Tan. End-to-end multi-person pose estimation with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2022.

[14] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for human pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[15] Yifan Xu, Weijian Xu, David Cheung, and Zhuowen Tu. Line segment detection using transformers without edges. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, April 2021.

[16] Yufei Xu, Jing Zhang, Qiming ZHANG, and Dacheng Tao. Vitpose: Simple vision transformer baselines for human pose estimation. In *Advances in Neural Information Processing Systems*, volume 35, Oct 2022.

[17] Hongfei Xue, Yan Ju, Chenglin Miao, Yijiang Wang, Shiyang Wang, Aidong Zhang, and Lu 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 (MobiSys)*, June 2021.

[18] Jie Yan, Xianlin Zeng, Anfu Zhou, and Huadong Ma. Mm-hat: Transformer for millimeter-wave sensing based human activity recognition. In *GLOBECOM 2022 - 2022 IEEE Global Communications Conference*, Dec 2022.

[19] Jianfei Yang, He Huang, Yunjiao Zhou, Xinyan Chen, Yuecong Xu, Shenghai Yuan, Han Zou, Chris Xiaoxuan Lu, and Lihua Xie. Mm-fi: Multi-modal non-intrusive 4d human dataset for versatile wireless sensing. *Advances in Neural Information Processing Systems*, 36, 2024.

[20] Mingmin Zhao, Tianhong Li, Mohammad Abu Alsheikh, Yonglong Tian, Hang Zhao, Antonio Torralba, and Dina Katabi. Through-wall human pose estimation using radio signals. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[21] Zhijie Zheng, Diankun Zhang, Xiao Liang, Xiaojun Liu, and Guangyou Fang. Radarformer: End-to-end human perception with through-wall radar and transformers. *IEEE Transactions on Neural Networks and Learning Systems*, pages 1–15, Sep 2023.

[22] Yunjiao Zhou, He Huang, Shenghai Yuan, Han Zou, Lihua Xie, and Jianfei Yang. Metafi++: Wifi-enabled transformer-based human pose estimation for metaverse avatar simulation. *IEEE Internet of Things Journal*, 10(16):14128–14136, Mar 2023.