RAPTR: Radar-based 3D Pose Estimation using Transformer

Sorachi Kato[1,2]*, **Ryoma Yataka**[1,3], **Pu (Perry) Wang**[1]†, **Pedro Miraldo**[1], **Takuya Fujihashi**[2], **Petros Boufounos**[1]
[1]Mitsubishi Electric Research Laboratories (MERL), USA
[2]The University of Osaka, Japan
[3]Information Technology R&D Center (ITC), Mitsubishi Electric Corporation, Japan

Abstract

Radar-based indoor 3D human pose estimation typically relied on fine-grained 3D keypoint labels, which are costly to obtain especially in complex indoor settings involving clutter, occlusions, or multiple people. In this paper, we propose **RAPTR** (RAdar Pose esTimation using tRansformer) under weak supervision, using only 3D BBox and 2D keypoint labels which are considerably easier and more scalable to collect. Our RAPTR is characterized by a two-stage pose decoder architecture with a pseudo-3D deformable attention to enhance (pose/joint) queries with multi-view radar features: a pose decoder estimates initial 3D poses with a 3D template loss designed to utilize the 3D BBox labels and mitigate depth ambiguities; and a joint decoder refines the initial poses with 2D keypoint labels and a 3D gravity loss. Evaluated on two indoor radar datasets, RAPTR outperforms existing methods, reducing joint position error by 34.3% on HIBER and 76.9% on MMVR. Our implementation is available at https://github.com/merlresearch/radar-pose-transformer.

1 Introduction

Accurate human perception is essential for indoor applications, including elderly monitoring, smart building management, and robotic navigation. Although vision sensors offer high spatial resolution, they raise privacy concerns and perform poorly under low light, occlusions, and hazardous conditions (fire or smoke). In contrast, radar provides penetration capability, robustness to adverse conditions, and low deployment cost, ideal for privacy-preserving indoor sensing [44, 18, 23, 30, 26].

By processing 4D radar tensors, RF-Pose 3D [48] demonstrated through-the-wall 3D pose estimation with a convolutional neural network (CNN), while HRRadarPose [11] employed an hourglass neural network HRNet [34]. mRI [1] is a multi-modal 3D human pose estimation dataset that integrates mmWave radar, RGB-D cameras, and inertial sensors to facilitate research in human pose estimation and action detection. QRFPose [33] is a novel approach that adopts a DETR [3]-style query mechanism for end-to-end 3D regression using multi-view radar perceptions. Existing pipelines often rely on expensive fine-grained 3D keypoint labels [35], typically collected using non-portable 3D motion capture systems such as VICON, or using LiDAR, which can still suffer from occlusions and incomplete observations.

Collecting cheaper, lower-cost labels, such as fine-grained 2D keypoints in the image plane and/or coarse-grained 3D bounding boxes (BBoxes), is considerably easier and more scalable particularly in complex indoor settings (e.g., cluttered, occlusion, multi-person), compared with acquiring dense

*The work was initiated during his internship at MERL.
†Project Lead.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

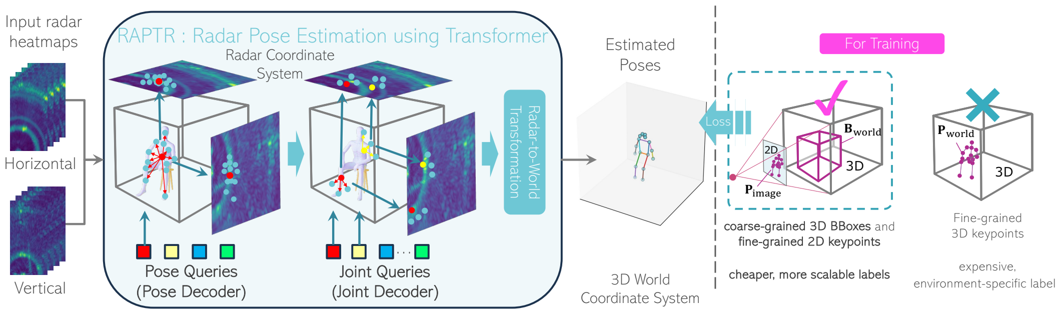


Figure 1: RAPTR takes multi-view radar heatmaps as inputs and performs a novel Pseudo-3D deformable attention between (pose and joint) queries and multi-view radar features in a two-stage decoder to estimate 3D human poses in a 3D coordinate system. Rather than relying on expensive, environment-specific fine-grained 3D keypoint labels, RAPTR makes use of cheaper, more scalable labels such as coarse-grained 3D BBoxes and fine-grained 2D keypoints to train the model.

3D keypoints labels. Examples include RF-Pose [47], HuPR [15], and, more recently, MMVR [24] datasets. To the best of our knowledge, the use of 2D keypoints and 3D BBoxes, as a substitute for costly 3D keypoints, for radar-based 3D human pose estimation has not been systematically investigated in the literature before.

To address this gap, we propose **RAPTR** (RAdar Pose esTimation using tRansformer) in Fig. 1, a radar-based pipeline designed to take multi-view radar heatmaps as inputs and estimate 3D human poses under weak supervision using only 3D BBox and 2D keypoint labels. RAPTR builds on the two-stage (pose and joint) decoder architecture of the state-of-the-art RGB-based 2D pose estimation PETR framework [28] and introduces a structural loss function that is designed to utilize weak supervision labels to mitigate the depth ambiguity. RAPTR also lifts the 2D deformable attention in PETR to a pseudo-3D deformable attention, wherein reference points (dots in Fig. 1) and offsets (arrows in Fig. 1) are proposed in the 3D radar coordinate system and projected onto multiple radar views (dots on the radar heatmaps in Fig. 1) to eliminate redundant per-view offset estimation and offer better scalability as the number of radar views increases. Our model outperforms a list of radar-based 3D pose baselines over two indoor radar datasets: HIBER [35] and MMVR [24]. The main contributions of this work are:

- To the best of our knowledge, RAPTR is the first radar-based 3D human pose estimation framework to explicitly utilize low-cost weak supervision in the form of 3D BBoxes and 2D keypoints, rather than relying on fine-grained 3D keypoint labels.
- We introduce a structured loss function that tightly couples the two-stage decoder architecture to enable 3D pose estimation under weak supervision. Specifically, we design a 3D Template Loss, which utilizes the 3D BBox labels at the pose decoder, and a combined 3D Gravity and 2D Keypoint Loss at the pose decoder, allowing RAPTR to effectively learn geometrically consistent 3D poses from weak supervision.
- We further introduce a pseudo-3D deformable attention mechanism to bridge the 3D spatial domain and 2D radar views, enabling scalable view association while preserving pose estimation performance.

2 Related Work

Human Pose Estimation with RGB Image: Human pose estimation from images involves localizing body joints for multiple subjects and associating them for each subject. Existing architectures fall into two main paradigms: top-down and bottom-up. The top-down methods first detect each person using detectors such as Faster R-CNN [25] or Mask R-CNN [10], then applying a single-person pose estimator to each cropped region. These approaches achieve state-of-the-art accuracy with models like Stacked-Hourglass [22], HRNet [31], and DarkPose [46]. In contrast, bottom-up methods such as OpenPose [2], HigherHRNet [6], and SAHR [19] bypass the detection step by predicting all joint candidates across the entire image and grouping them into individuals. PETR [28] introduces an end-to-end pose estimation framework using a query-based, two-stage transformer decoder architecture. Beyond 2D, recent methods addresses 3D pose from RGB or RGB-D inputs, either by directly regressing 3D joints [20] or by lifting 2D predictions into the 3D space through geometric reasoning or weak supervision [4, 5, 21, 32].

Human Pose Estimation with radar or radio frequency signals: Recent studies have shown that information extracted from commercial radars is sufficiently informative to perform fine-grained human pose estimation, both for 2D and 3D. Despite the coarse-grained nature of the radar point clouds (PCs), deep neural pipelines have achieved a multitude of performance gains [29, 27, 38, 1, 41, 36, 12, 14, 39, 42, 8, 7, 43]. On the other hand, methods using raw radar measurements and radar heatmaps have been widely explored [47, 49, 15, 35, 11, 24, 33, 45, 37]. RF-Pose [47] pioneered multi-view 3D CNNs for through-wall 2D estimation. HuPR [15] refines such heatmaps via a graph convolutional network (GCN). HRRadarPose [11] adopts an HRNet-style [34] single-stage head for 3D output. QRFPose [33], based on a DETR-style Transformer [3] for end-to-end query-based 3D pose estimation, is the closest baseline to ours. It differs by applying per-view 2D deformable attention and using a single decoder for all keypoints, followed by grouping. In contrast, our method employs pseudo-3D deformable attention and a two-stage decoder.

3 Preliminary

Multi-View Radar Heatmaps: As shown in Fig. 2, two synchronized radar arrays (horizontal and vertical) collect reflected pulses that form a 3D data cube per array (ADC samples $\times$ pulses $\times$ elements). A 3D FFT converts each cube into a range–Doppler–angle spectrum, whose angle dimension is azimuth for the horizontal array and elevation for the vertical array. After Doppler-axis integration to boost SNR (signal-to-noise ratio), we obtain two polar 2D heatmaps (range-azimuth and range-elevation). These are mapped to the Cartesian space: $\mathbf{Y}_{\texttt{hor}}(t) \in \mathbb{R}^{W\times D}$ for horizontal-depth and $\mathbf{Y}_{\texttt{ver}}(t) \in \mathbb{R}^{H\times D}$ for vertical-depth at frame t. The temporal context is captured by stacking T consecutive frames, giving $\mathbf{Y}_{\texttt{hor}} \in \mathbb{R}^{T\times W\times D}$ and $\mathbf{Y}_{\texttt{ver}} \in \mathbb{R}^{T\times H\times D}$.

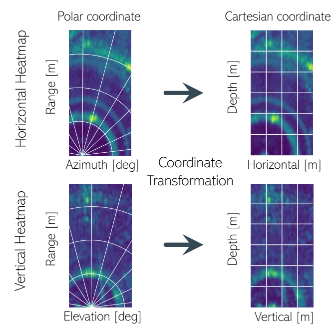


Figure 2: Multi-view radar heatmaps.

Problem Formulation: The 3D pose estimation task takes T consecutive radar frames, $\mathbf{Y}_{\texttt{hor}}$ and $\mathbf{Y}_{\texttt{ver}}$, as input and estimates poses $\hat{\mathbf{P}}_{\texttt{world}}$ in the 3D world coordinate system,

$$\hat{\mathbf{P}}_{\texttt{world}} = \mathcal{T}_{\texttt{r2w}}(\hat{\mathbf{P}}_{\texttt{radar}}) = \mathcal{T}_{\texttt{r2w}}(f(\mathbf{Y}_{\texttt{hor}}, \mathbf{Y}_{\texttt{ver}})), \tag{1}$$

where f represents the 3D pose estimation pipeline in the 3D radar coordinate system, and $\mathcal{T}_{\texttt{r2w}}$ is a known radar-to-world coordinate transformation that converts the estimated 3D poses into the 3D world coordinate system. Rather than relying on costly, non-scalable fine-grained 3D keypoint labels $\mathbf{P}_{\texttt{world}}$, we consider cheaper, more scalable labels such as coarse-grained 3D BBoxes $\mathbf{B}_{\texttt{world}}$ and fine-grained 2D keypoints $\mathbf{P}_{\texttt{image}}$ for supervision, as shown in Fig. 1.

4 RAPTR: Radar-based 3D Pose Estimation using Transformer

We present the RAPTR architecture in Fig. 3, following a left-to-right order, and highlight radar-specific modifications. Refer to Appendix A for detailed architecture and computational complexity.

4.1 Architecture

Backbone: Given $\mathbf{Y}_{\texttt{hor}} \in \mathbb{R}^{T\times W\times D}$ and $\mathbf{Y}_{\texttt{ver}} \in \mathbb{R}^{T\times H\times D}$, a shared backbone network (e.g., ResNet [9]) generates separate multi-scale horizontal-view and vertical-view radar feature maps: $\mathbf{Z}_{\texttt{hor}} = \{\mathbf{Z}_{\texttt{hor},i}\}_{i=1}^{S} = \texttt{backbone}\,(\mathbf{Y}_{\texttt{hor}})$ and $\mathbf{Z}_{\texttt{ver}} = \{\mathbf{Z}_{\texttt{ver},i}\}_{i=1}^{S} = \texttt{backbone}\,(\mathbf{Y}_{\texttt{ver}})$, where the i-th scale feature maps $\mathbf{Z}_{\texttt{hor},i} \in \mathbb{R}^{W_i\times D_i\times d}$ and $\mathbf{Z}_{\texttt{ver},i} \in \mathbb{R}^{H_i\times D_i\times d}$ have a spatial dimension of $W_i \times D_i$ or $H_i \times D_i$ and a feature dimension of d, and S is the number of scales.

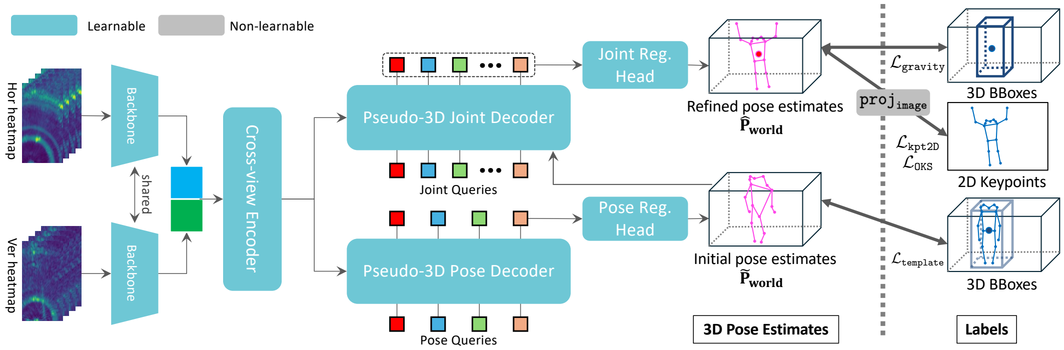


Figure 3: The RAPTR architecture consists of: 1) **Cross-view Encoder** that extracts multi-scale radar features; 2) **Pseudo-3D Pose Decoder** that enhances pose queries via a pseudo-3D deformable attention and predicts initial 3D poses; and 3) **Pseudo-3D Joint Decoder** that further refines joint queries and outputs final 3D poses. In terms of **loss function**, RAPTR leverages 3D BBox and 2D keypoint labels through coarse-grained 3D loss (gravity and template) and 2D keypoint loss.

Cross-View Encoder is a Transformer encoder with $L_{\texttt{enc}}$ layers that fuses the horizontal- and vertical-view radar features. Each layer runs a shared cross-attention twice: first with $\mathbf{Z}_{\texttt{hor}}$ as key/value and $\mathbf{Z}_{\texttt{ver}}$ as query, then vice versa. This bidirectional exchange embeds complementary cues, while residual connections keep view-specific details, producing refined features $\mathbf{F}_{\texttt{enc}}^{(i)}$, $i = 1, \cdots, L_{\texttt{enc}}$,

$$\mathbf{F}_a^{(i)} = \mathbf{F}_a^{(i-1)} + \texttt{CrossAttn}(\mathbf{F}_a^{(i-1)}, \mathbf{F}_b^{(i-1)}), \quad (a, b) \in \{(\texttt{hor}, \texttt{ver}), (\texttt{ver}, \texttt{hor})\}, \tag{2}$$

where $\mathbf{F}_{\texttt{hor}}^{(0)} = \mathbf{Z}_{\texttt{hor}}$, $\mathbf{F}_{\texttt{ver}}^{(0)} = \mathbf{Z}_{\texttt{ver}}$, and $\texttt{CrossAttn}(\cdot, \cdot)$ denotes the deformable cross-attention [50] following [28] with fixed positional embeddings added beforehand for efficiency. After $L_{\texttt{enc}}$ iterations, the encoded features $\mathbf{F}_{\texttt{hor}}$ and $\mathbf{F}_{\texttt{ver}}$ are obtained at the output of the cross-view encoder.

Pseudo-3D Pose Decoder associates N pose queries $\mathbf{Q}_{\texttt{pose}} \in \mathbb{R}^{N \times d}$ (embedding dimension d) with encoded radar features $(\mathbf{F}_{\texttt{hor}}, \mathbf{F}_{\texttt{ver}})$, where each query corresponds to a reference pose refined through pseudo-3D deformable attention over $L_{\texttt{pose}}$ layers. We define the l-th decoder layer as a function $\mathcal{D}_{\texttt{pose}}^{(l)}$ that updates both the pose queries and reference poses in the 3D radar space:

$$(\mathbf{Q}_{\texttt{pose}}^{(l)}, \tilde{\mathbf{P}}_{\texttt{radar}}^{(l)}) = \mathcal{D}_{\texttt{pose}}^{(l)}(\mathbf{Q}_{\texttt{pose}}^{(l-1)}, \mathbf{F}_{\texttt{hor}}, \mathbf{F}_{\texttt{ver}}, \tilde{\mathbf{P}}_{\texttt{radar}}^{(l-1)}), \tag{3}$$

where $\tilde{\mathbf{P}}_{\texttt{radar}}^{(0)}$ is initialized by passing $\mathbf{Q}_{\texttt{pose}}$ through an MLP. Reference poses are iteratively refined by applying predicted coordinate offsets $\Delta\tilde{\mathbf{P}}_{\texttt{radar}}^{(l)}$ in the normalized scale:

$$\tilde{\mathbf{P}}_{\texttt{radar}}^{(l)} \in \mathbb{R}^{N \times 3K} = \sigma(\sigma^{-1}(\tilde{\mathbf{P}}_{\texttt{radar}}^{(l-1)}) + \Delta\tilde{\mathbf{P}}_{\texttt{radar}}^{(l-1)}), \quad l = 1, \ldots, L_{\texttt{pose}}, \tag{4}$$

where σ and σ^{-1} denote the Sigmoid function and its inverse. The predicted offsets $\Delta\tilde{\mathbf{P}}_{\texttt{radar}}^{(l)} = H_{\texttt{pose}}(\mathbf{Q}_{\texttt{pose}}^{(l)})$ are obtained by passing pose queries at each layer to a shared regression head $H_{\texttt{pose}}$.

We convert the initial pose estimates $\tilde{\mathbf{P}}_{\texttt{radar}} = \tilde{\mathbf{P}}_{\texttt{radar}}^{(L_{\texttt{pose}})}$ from the radar coordinate system to the world coordinate system via $\tilde{\mathbf{P}}_{\texttt{world}} = \mathcal{T}_{\texttt{r2w}}(\tilde{\mathbf{P}}_{\texttt{radar}})$, along with the corresponding confidence scores $\tilde{\mathbf{c}}$. We defer the pseudo-3D deformable attention to Section 4.2.

Pseudo-3D Joint Decoder associates K joint queries $\mathbf{Q}_{\texttt{joint}} \in \mathbb{R}^{K \times d}$ with encoded radar features $(\mathbf{F}_{\texttt{hor}}, \mathbf{F}_{\texttt{ver}})$, where each query corresponds to a single joint refined by pseudo-3D deformable attention over $L_{\texttt{joint}}$ layers. Here, K joint queries correspond to the same subject. We define the l-th decoder layer as a function $\mathcal{D}_{\texttt{joint}}^{(l)}$ that updates both the joint queries and corresponding joints:

$$(\mathbf{Q}_{\texttt{joint}}^{(l)}, \tilde{\mathbf{p}}_{i,\texttt{radar}}^{(l)}) = \mathcal{D}_{\texttt{joint}}^{(l)}(\mathbf{Q}_{\texttt{joint}}^{(l-1)}, \mathbf{F}_{\texttt{hor}}, \mathbf{F}_{\texttt{ver}}, \tilde{\mathbf{p}}_{i,\texttt{radar}}^{(l-1)}), \tag{5}$$

where $\tilde{\mathbf{p}}_{i,\texttt{radar}}^{(l)} \in \mathbb{R}^{K \times 3}, i = 1, \cdots, N$ is one specific pose in the N poses, and $\tilde{\mathbf{p}}_{i,\texttt{radar}}^{(0)}$ is i-th pose prediction from the pose decoder. Joints in the reference pose are iteratively refined by applying predicted coordinate offsets $\Delta\tilde{\mathbf{p}}_{i,\texttt{radar}}^{(l)}$:

$$\tilde{\mathbf{p}}_{i,\texttt{radar}}^{(l)} \in \mathbb{R}^{K \times 3} = \sigma(\sigma^{-1}(\tilde{\mathbf{p}}_{i,\texttt{radar}}^{(l-1)}) + \Delta\tilde{\mathbf{p}}_{i,\texttt{radar}}^{(l-1)}), \quad l = 1, \cdots, L_{\texttt{joint}}, \tag{6}$$

where the predicted offsets are given as $\Delta\tilde{\mathbf{p}}_{i,\mathtt{radar}}^{(l)} = H_{\mathtt{joint}}(\mathbf{Q}_{\mathtt{joint}}^{(l)})$ with a shared regression head.

We collect refined reference poses from the joint decoder as $\hat{\mathbf{P}}_{\mathtt{radar}} = \{\tilde{\mathbf{p}}_{i,\mathtt{radar}}^{(L_{\mathtt{joint}})}\}_{i=1}^{N}$ and convert them into the 3D world coordinate system as $\hat{\mathbf{P}}_{\mathtt{world}} = \mathcal{T}_{\mathtt{r2w}}(\hat{\mathbf{P}}_{\mathtt{radar}})$.

4.2 Pseudo-3D Deformable Attention

Our two-stage decoder incorporates a pseudo-3D deformable attention module, where "pseudo" highlights that reference points and sampling offsets are defined in 3D space, while feature sampling occurs on the 2D radar views, as illustrated in Fig. 4.

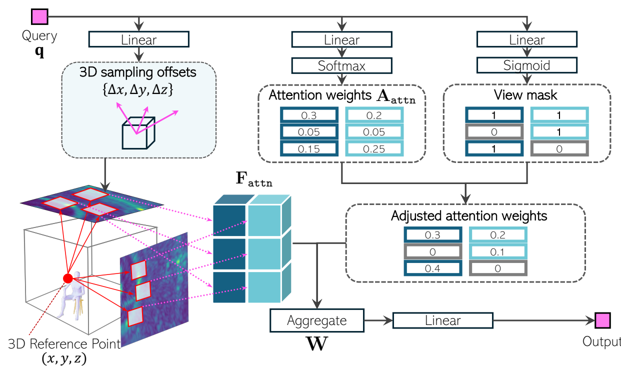


Figure 4: The pseudo-3D deformable attention operates on a 3D reference point and 3D sampling offsets that are projected to different radar views for pseudo-3D attention between multi-view radar features and the query.

Consider a 3D reference point (x, y, z) in the 3D radar space with a corresponding query $\mathbf{q} \in \mathbb{R}^d$ (from either pose queries in the pose decoder or joint queries in the joint decoder). We first feed $\mathbf{q}$ into a linear projection layer to predict a set of 3D sampling offsets $\{\Delta x_i, \Delta y_i, \Delta z_i\}_{i=1}^{N_{\mathtt{offset}}}$. Given the 3D reference point and sampling offsets, we can locate the 3D sampling coordinates and project them onto the two radar views, extracting deformable multi-view radar features:

$$\mathbf{f}_{\mathtt{hor}}^{(i)} = \mathbf{F}_{\mathtt{hor}}(x + \Delta x_i, z + \Delta z_i), \quad \mathbf{f}_{\mathtt{ver}}^{(i)} = \mathbf{F}_{\mathtt{ver}}(y + \Delta y_i, z + \Delta z_i) \quad i = 1, \cdots, N_{\mathtt{offset}}. \tag{7}$$

We group deformable multi-view radar features as $\mathbf{F}_{\mathtt{attn}} = \{\mathbf{f}_{\mathtt{hor}}^{(1)}, \mathbf{f}_{\mathtt{ver}}^{(1)}, \cdots, \mathbf{f}_{\mathtt{hor}}^{(N_{\mathtt{offset}})}, \mathbf{f}_{\mathtt{ver}}^{(N_{\mathtt{offset}})}\}$.

Meanwhile, multi-view attention weights $\mathbf{A}_{\mathtt{attn}} \in \mathbb{R}^{N_{\mathtt{offset}} \times 2}$ (where 2 corresponds to the two radar views) are proposed by linearly projecting the query and applying a softmax normalization. These weights capture the relative importance of radar features across the two views in a unified manner.

Given the deformable multi-view radar features $\mathbf{F}_{\mathtt{attn}}$ and the multi-view attention weights $\mathbf{A}_{\mathtt{attn}}$, deformable multi-view attention features can be calculated as

$$\bar{\mathbf{F}}_{\mathtt{attn}} = \sum_{i=1}^{N_{\mathtt{offset}}} (A_{i,0}\mathbf{W}\mathbf{F}_{\mathtt{attn}}^{(2i-1)} + A_{i,1}\mathbf{W}\mathbf{F}_{\mathtt{attn}}^{(2i)}), \tag{8}$$

where $A_{i,0}$ and $A_{i,1}$ are the attention weights in $\mathbf{A}_{\mathtt{attn}}$ for the i-th deformable radar feature in the horizontal and, respectively, vertical radar views, and $\mathbf{W} \in \mathbb{R}^{C \times C}$ is a learnable weight matrix. We denote the overall pseudo-3D deformable attention as $\bar{\mathbf{F}}_{\mathtt{attn}} = \mathtt{DeformableAttn}(\mathbf{F}_{\mathtt{ver}}, \mathbf{F}_{\mathtt{hor}}, (x, y, z), \mathbf{q})$.

Appendix B provides implementation details of $\mathtt{DeformableAttn}(\cdot, \cdot, \cdot, \cdot)$ and a computational complexity comparison with the decoupled 2D deformable attention used in QRFPose [33], demonstrating better scalability of the proposed pseudo-3D attention as the number of radar views increases. Appendix C describes an optional view mask module (top right of Fig. 4) that adds flexibility in selecting multi-view radar features per query. For example, an all-zero mask can be applied to exclude features from a specific radar view.

4.3 Structural Loss Function

As illustrated in Fig. 3, RAPTR utilizes weak supervision labels: coarse-grained BBox labels $\mathbf{B}_{\mathtt{world}}$ in the 3D world coordinate system and fine-grained 2D keypoint labels $\mathbf{P}_{\mathtt{image}}$ in the image plane. The loss function is calculated between these labels $\{\mathbf{B}_{\mathtt{world}}, \mathbf{P}_{\mathtt{image}}\}$ and the initial and refined 3D pose estimates $\{\tilde{\mathbf{P}}_{\mathtt{world}}, \hat{\mathbf{P}}_{\mathtt{world}}\}$ with details included in Appendix D.

3D Template (T3D) Loss at Pose Decoder utilizes coarse-grained 3D BBox labels $\mathbf{B}_{\mathtt{world}}$. For each $\mathbf{B}_{\mathtt{world}}$, we construct a 3D keypoint template by computing the centroid of the corresponding 3D BBox, which serves as the 3D gravity center label $\mathbf{g}_{\mathtt{world}} \in \mathbb{R}^{1 \times 3}$.

Then, given a keypoint template defined at the coordinate origin $\mathbf{K}_{\texttt{world}} \in \mathbb{R}^{K\times 3}$, the corresponding template pose $\mathbf{T}_{\texttt{world}}$ is computed as $\mathbf{T}_{\texttt{world}} = \mathbf{K}_{\texttt{world}} + \mathbf{1}^{\top}\mathbf{g}_{\texttt{world}}$. As illustrated in the lower right of Fig. 3, the T3D loss $\mathcal{L}_{\texttt{template}}$ is defined as the Euclidean distance between the template poses $\mathbf{T}_{\texttt{world}}$ and the initial 3D pose estimates $\tilde{\mathbf{P}}_{\texttt{world}}$ at the pose decoder.

Combined 3D Gravity (G3D) Loss and 2D Keypoint (K2D) Loss at Joint Decoder utilizes both the coarse-grained 3D BBox labels $\mathbf{B}_{\texttt{world}}$ and the fine-grained 2D keypoint labels $\mathbf{P}_{\texttt{image}}$ in the image plane, as illustrated in the upper right of Fig. 3

For the G3D loss, the refined 3D pose estimate $\hat{\mathbf{P}}_{\texttt{world}}$ is collapsed into its centroid as $\hat{\mathbf{g}}_{\texttt{world}} \in \mathbb{R}^{1\times 3}$ by averaging the keypoint coordinates along each spatial axis. The resulting G3D loss $\mathcal{L}_{\texttt{gravity}}$ is then defined as the Euclidean distance between the predicted and ground-truth 3D gravity centers, $\hat{\mathbf{g}}_{\texttt{world}}$ and $\mathbf{g}_{\texttt{world}}$.

For the K2D loss, the refined 3D pose estimate $\hat{\mathbf{P}}_{\texttt{radar}}$ in the radar coordinate system are first transformed into the 3D camera coordinate system via a calibrated coordinate transformation: $\hat{\mathbf{P}}_{\texttt{camera}} = \mathbf{R}\hat{\mathbf{P}}_{\texttt{radar}} + \mathbf{1}^{\top}\mathbf{t}$ where $\mathbf{R}$ and $\mathbf{t}$ denote the calibrated 3D rotation matrix and the translation vector, respectively. The resulting 3D camera-space pose estimates are then projected onto the 2D image plane via a known 3D-to-2D projection: $\hat{\mathbf{P}}_{\texttt{image}} = \texttt{proj}_{\texttt{image}}(\hat{\mathbf{P}}_{\texttt{camera}})$. Finally, the fine-grained 2D loss combines the image-plane Euclidean error $\mathcal{L}_{\texttt{kpt2D}}$ and the object keypoint similarity (OKS) loss $\mathcal{L}_{\texttt{OKS}}$ [28] between $\mathbf{P}_{\texttt{image}}$ and $\hat{\mathbf{P}}_{\texttt{image}}$.

Structural Loss Function: Following the set-based loss in [3], we employ bipartite matching to associate predictions $\{\hat{\mathbf{g}}_{\texttt{world}}, \tilde{\mathbf{P}}_{\texttt{world}}, \hat{\mathbf{P}}_{\texttt{world}}\}$ with their ground-truth labels $\{\mathbf{g}_{\texttt{world}}, \mathbf{T}_{\texttt{world}}, \mathbf{P}_{\texttt{world}}\}$. Based on these associations, we define the structural loss function as

$$\mathcal{L} = \frac{1}{N'}\sum_{i=1}^{N'}(\lambda_1\mathcal{L}_{\texttt{template}} + \lambda_2\mathcal{L}_{\texttt{gravity}} + \lambda_3\mathcal{L}_{\texttt{kpt2D}} + \lambda_4\mathcal{L}_{\texttt{OKS}}) + \lambda_5\mathcal{L}_{\texttt{cls}}, \tag{9}$$

where N' is the number of matched pairs, λ_i is the corresponding weighting factor for each loss term, and $\mathcal{L}_{\texttt{cls}}$ is the classification loss of the focal loss [17] with the confidence scores of the matched estimates.

5 Evaluation

5.1 Settings

Datasets: We assess the performance of RAPTR and baseline models on the HIBER dataset[3] [35] and the MMVR dataset[4] [24], both of which are publicly available multi-view mmWave radar datasets designed for indoor human perception tasks. The HIBER dataset includes two-view radar heatmaps from 10 different viewpoints, the corresponding 3D keypoint labels, and the 3D BBox labels. We use data protocols "MULTI" and "WALK", and use views 2 through 10 for training, validation, and testing. The MMVR dataset includes two-view radar heatmaps in various indoor scenarios, the corresponding 2D keypoint labels, and the 3D BBoxes. We use a data split "P1S1", a single-person case in an open space. A detailed description of the datasets is provided in Appendix E.

Parameter Settings for RAPTR: We use $T = 4$ consecutive frames as input to our RAPTR network. For the point decoder, the number of pose queries N is 10. For the joint decoder, the number of joint queries K depends on the dataset to be evaluated: $K = 14$ for HIBER and $K = 17$ for MMVR. The parameters relating to model training are summarized in Appendix F.

Baselines: We consider the following competitive radar/RF-based 3D pose estimation baselines: **Person-in-WiFi 3D** [40], **HRRadarPose** [11], and **QRFPose** [33]. We evaluate Person-in-WiFi 3D and HRRadarPose using their open-source implementations. As QRFPose has no public code, we reimplement it from scratches and verify similar performance to the original report [33] using 3D keypoint labels. For fair comparison, we adopt a loss function, similar to RAPTR, combining 2D keypoint loss and 3D gravity loss. Baseline implementation details are provided in Appendix G.

[3] https://github.com/Intelligent-Perception-Lab/HIBER

[4] https://zenodo.org/records/12611978

Table 1: 3D pose estimation performance on HIBER (MPJPE: cm).

Env	Method	Head	Neck	Shoulder	Elbow	Wrist	Hip	Knee	Ankle	Overall	(h)	(v)	(d)
WALK	Person-in-WiFi 3D	54.28	57.01	54.18	54.81	59.98	53.98	60.32	68.84	58.25	25.60	23.94	36.20
	QRFPose	42.23	34.21	37.37	38.05	41.25	31.24	34.39	46.87	38.20	14.78	13.40	26.76
	HRRadarPose	30.23	25.44	33.70	34.15	42.33	27.71	31.55	40.46	33.96	15.14	13.13	19.85
	RAPTR (ours)	21.75	17.41	20.72	23.23	26.55	18.97	21.06	26.10	**22.32**	**8.41**	**4.85**	**17.73**
MULTI	Person-in-WiFi 3D	88.48	85.14	89.44	84.33	84.29	88.69	81.70	81.53	85.25	34.06	28.57	58.93
	QRFPose	49.49	44.48	45.54	46.77	49.06	40.99	41.87	51.57	46.11	18.20	14.13	34.39
	HRRadarPose	30.24	24.24	30.14	35.17	44.34	28.76	31.38	35.31	33.19	16.77	10.75	21.84
	RAPTR (ours)	18.39	13.13	16.44	20.12	24.62	15.01	17.76	23.22	**18.99**	**7.80**	**4.38**	**14.54**

Metrics: We employ **Mean Per Joint Pose Error (MPJPE)** with the unit of centimeters in the world coordinate. In addition, we evaluate this MPJPE for each body joint and along each 3D axis, horizontal (h), vertical (v), and depth (d), independently. For MMVR, since 3D keypoint labels are not available, we construct a 3D bounding box (BBox) that encloses the estimated 3D keypoints and then use **the distance between the center points** of this BBox and the 3D BBox labels, as well as **the absolute error in the lengths of the edges** along each axis of the box, as metrics to approximate the 3D pose estimation performance. Detailed evaluation metrics are described in Appendix H.

5.2 Main results

HIBER: Table 1 shows the performance of 3D pose estimation for HIBER, using 2D and coarse 3D labels for baselines and our RAPTR. The qualitative results are provided in Fig. 5a.

For WALK, RAPTR achieves a significantly lower overall MPJPE of $22.32\,\mathrm{cm}$ and outperforms all other baselines in the metric. More specifically, RAPTR reduces the overall error by 61.7%, 41.6%, and 34.3% compared to Person-in-WiFi 3D, QRFPose, and HRRadarPose, respectively. The per-joint breakdown demonstrates that RAPTR maintains its performance on relatively challenging joints, such as the wrist and ankle, where other baselines exhibit significant degradation. For example, HRRadarPose reports a wrist error of $42.33\,\mathrm{cm}$, whereas RAPTR reports an error of $26.55\,\mathrm{cm}$. Moreover, RAPTR maintains the error gap between the best- and worst-estimated joints within $10\,\mathrm{cm}$, showing a consistent level of accuracy throughout the body. In terms of directional components, RAPTR shows much lower errors in the horizontal and vertical dimensions than baselines, indicating that RAPTR estimates well-proportioned 3D poses across all axes.

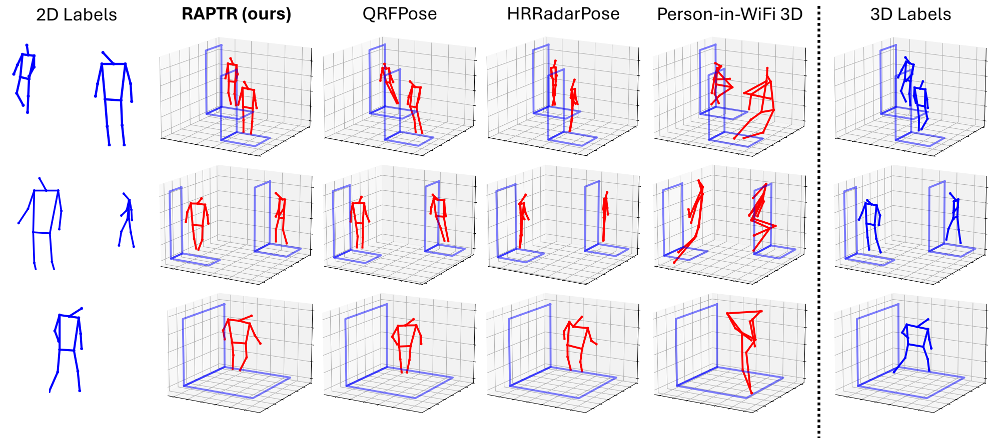
For MULTI, the more challenging multi-person scenario, RAPTR continues to outperform with an overall MPJPE of $18.99\,\mathrm{cm}$ and shows a substantial margin compared to the second-best HRRadar-Pose at $33.19\,\mathrm{cm}$. RAPTR reduces the overall error by 77.7%, 58.8%, and 42.7% compared to Person-in-WiFi 3D, QRFPose, and HRRadarPose, respectively. Although the overall accuracy of Person-in-WiFi 3D and QRFPose, noticeably degrades on the MULTI split compared to WALK, likely due to the increased complexity of handling multiple objects, RAPTR maintains a nearly consistent level of performance.

Referring to the qualitative results provided in Fig. 5a, RAPTR estimates structurally consistent 3D poses that match the 3D labels in both position and orientation, while baselines often suffer from misaligned limbs and implausible joint configurations. While baselines often fail to maintain human-like pose structure in the MULTI setting despite performing well in WALK, RAPTR consistently produces plausible estimates in both scenarios, indicating its robustness to multi-person scenes.

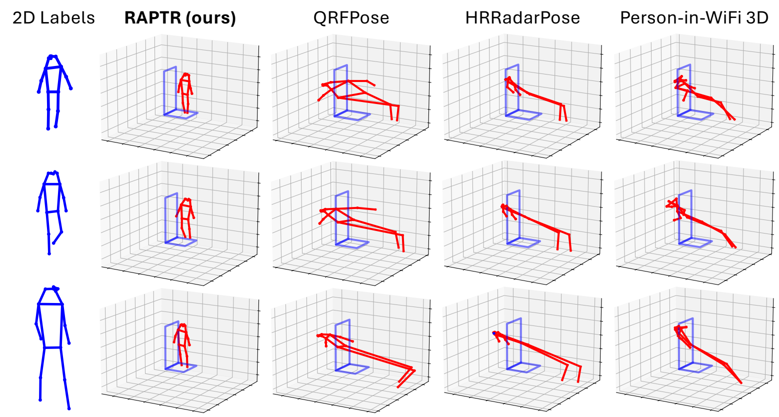
Table 2: Pose estimation performance on MMVR (P1S1).

Method	Center distance (cm)	Edge length error (cm) (h)	(v)	(d)
Person-in-WiFi 3D	136.14	33.18	95.43	242.86
QRFPose	210.75	38.12	73.69	409.38
HRRadarPose	164.46	37.84	74.00	313.81
RAPTR (ours)	**31.41**	**22.90**	**10.66**	**50.56**

MMVR: Table 2 shows the performance comparison for baselines and RAPTR with MMVR, and Fig. 5b provides qualitative results. Although we cannot directly evaluate the precise 3D pose estimation performance for MMVR due to the absence of 3D pose labels, the results demonstrate that RAPTR effectively preserves reasonable human pose and location accuracy in the 3D space. Specifically, RAPTR shows improvements in center distance by 76.9%, 85.1%, and 80.9% compared to Person-in-WiFi 3D, QRFPose, and HRRadarPose, respectively. As shown in Fig. 5b, other



(a) Visualization of 3D pose estimation by RAPTR and baseline methods on the HIBER dataset.



(b) Visualization of 3D pose estimation by RAPTR and baseline methods on the MMVR dataset.

Figure 5: Qualitative results. Blue lines indicate the keypoint labels, blue rectangles indicate the 3D BBox labels, and red lines indicate the predictions.

baselines exhibit degraded performance due to structural collapse in 3D space, caused by overfitting to 2D alignment when projected onto the image plane. We assume that RAPTR effectively avoids this issue by not directly predicting the keypoints, but instead refining the final output through 2D keypoint supervision applied to each joint of a template pose that is placed in the 3D space.

6 Ablation Study

In this section, we present ablation studies of our RAPTR on the HIBER dataset. Unless otherwise stated, all reported evaluation results are reported as the mean $\pm$ standard deviation, computed over three random seeds. Additional ablation results and visualizations are provided in Appendix I.

Visualization of Pose Refinement Process: Fig. 6 illustrates the refinement process of a 3D prediction through the two-stage decoder architecture. The pose decoder first establishes coarse 3D structures of the human body under the constraint of the 3D template loss (1st row). Subsequently, the joint decoder fine-tunes the keypoints to better capture the subject orientation and limb configuration (2nd row), while preserving the structure consistency provided by the pose decoder.

Effect of Loss Terms: Table 3 provides an ablation study on the effect of different combinations of loss terms in the two-stage decoder. When only the K2D loss is applied at the joint decoder (row 1), the 3D pose estimation suffers from depth ambiguity due to the absence of any 3D constraint, resulting in a substantial increase in MPJPE to $381.18\,\mathrm{cm}$ and $375.73\,\mathrm{cm}$ on the WALK and MULTI splits, respectively. From rows 2 to 4 in Table 3, we remove or modify one loss term at a time from

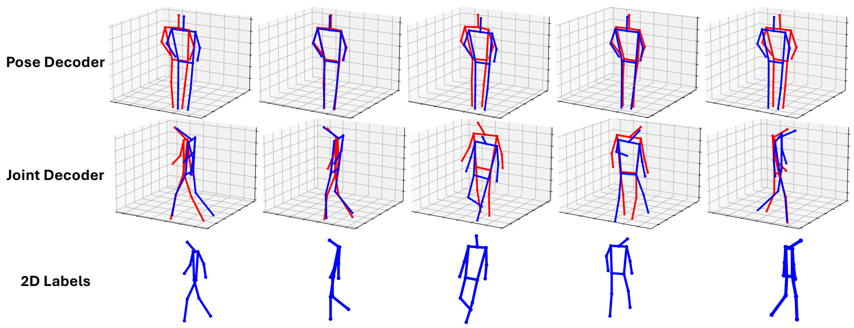


Figure 6: Pose Refinement Process: the pose prediction is first constrained by the 3D template at the pose decoder, and subsequently refined at the joint decoder.

the proposed structural loss. Removing the T3D loss at the pose decoder (row 2), replacing the T3D with the K2D+G3D loss at the pose decoder (row 3), or removing the G3D loss at the joint decoder (row 4) leads to a noticeable degradation in 3D pose estimation.

Table 3: Effect of loss terms for RAPTR (MPJPE: cm).

Loss		**WALK**	**MULTI**	**Notes**
Pose Dec.	Joint Dec.			
–	`K2D`	381.18 ± 0.28	375.73 ± 6.31	2D keypoint loss only at joint decoder
–	`K2D+G3D`	28.54 ± 4.57	57.90 ± 9.81	without 3D template loss at pose decoder
`K2D+G3D`	`K2D+G3D`	27.49 ± 3.40	23.43 ± 3.44	with 2D keypoint + 3D gravity loss at both decoders
`T3D`	`K2D`	25.96 ± 4.95	25.83 ± 3.87	without 3D gravity loss at joint decoder
`T3D`	`K2D+G3D`	$\mathbf{22.32 \pm 0.06}$	$\mathbf{18.99 \pm 0.16}$	proposed structural loss

`K2D` = 2D Keypoint loss, `T3D` = 3D Template loss, `G3D` = 3D Gravity loss

Effect of Deformable Attention Mechanisms: Table 4 presents an ablation study on the effect of the deformable attention mechanism for RAPTR. In this study, the pseudo-3D deformable attention is replaced with the decoupled 2D deformable attention used in QRFPose [33], while keeping the cross-view encoder, two-stage decoder architecture, and the proposed structural loss unchanged. The results show that the pseudo-3D deformable attention yields marginal performance improvements, approximately 4% and 2.5% on the WALK and MULTI splits, respectively.

Table 4: Effect of deformable attention mechanisms for RAPTR (MPJPE: cm).

Attn.	**WALK**	**MULTI**	**Notes**
`2D`	23.25 ± 1.38	19.47 ± 0.95	RAPTR with decoupled 2D deformable attention
`3D`	$\mathbf{22.32 \pm 0.06}$	$\mathbf{18.99 \pm 0.16}$	RAPTR with pseudo-3D deformable attention

Comparison with a 2D-to-3D Pose Uplifting Model: We further compare RAPTR with a baseline that first estimates 2D keypoints in the image plane and subsequently lifts them to 3D space using a pre-trained 2D-to-3D pose uplifting model [21] trained on vision-based datasets such as Human3.6M [13]. To ensure a fair comparison, this baseline adopts the same network architecture as RAPTR, but both the pose and joint decoders are supervised only by the 2D keypoint loss. Because the 3D poses predicted by the uplifting model are defined in a pelvis-centered coordinate system, we additionally estimate a translation offset to align the estimated poses with their correct position in the world coordinate system. As shown in Table 5, the pose uplifting baseline performs significantly worse than RAPTR, with MPJPEs of $43.43\,\mathrm{cm}$ and $41.76\,\mathrm{cm}$ on the WALK and MULTI splits, respectively.

Limitation: Given that the process of refining the template to the actual pose in the joint decoder is supervised by the 2D keypoint labels, the accuracy of the 3D pose estimation is highly dependent on the precision of the labels in the image plane. In this context, since the 2D keypoint label lacks the

Table 5: Comparison with a 2D-to-3D pose uplifting model (MPJPE: cm).

	Loss		WALK	MULTI
	Pose Dec.	Joint Dec.		
Pose Lifting [21]	K2D	K2D	43.43 ± 2.66	41.76 ± 6.85
RAPTR (ours)	T3D	K2D+G3D	$\mathbf{22.32 \pm 0.06}$	$\mathbf{18.99 \pm 0.16}$

ability to discern whether the person is facing forward or backward to the camera, the estimated 3D poses may have joints that are bent in the opposite direction in depth from the actual pose. In addition, real-world conditions such as occlusion and human-to-human interference can further degrade the pose estimation performance. These effects become more pronounced in crowded or interactive environments.

7 Conclusion

We introduced RAPTR, a radar-based 3D human pose estimation system using reliable 2D keypoint labels and 3D BBoxes as the coarse-grained 3D information. We designed the network architecture and the loss function to integrate multi-view radar features and consistently represent human poses in the 3D space, whose effectiveness was demonstrated through experimental results.

Broader Impacts: Indoor radar perception technologies, such as RAPTR, provide diverse indoor applications. These technologies may improve the safety and energy efficiency of indoor systems while preserving privacy. However, it is paramount that the perception results remain secure and private to prevent misuse.

References

[1] Sizhe An, Yin Li, and Umit Ogras. mRI: Multi-modal 3D human pose estimation dataset using mmWave, RGB-D, and inertial sensors. In *Advances in Neural Information Processing Systems*, pages 27414–27426, 2022.

[2] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime multi-person 2D pose estimation using part affinity fields. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7291–7299, 2017.

[3] 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*, page 213–229, 2020.

[4] Ching-Hang Chen and Deva Ramanan. 3D human pose estimation = 2D pose estimation + matching. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5759–5767, 2017.

[5] Ching-Hang Chen, Ambrish Tyagi, Amit Agrawal, Dylan Drover, Rohith Mv, Stefan Stojanov, and James M Rehg. Unsupervised 3D pose estimation with geometric self-supervision. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5714–5724, 2019.

[6] Bowen Cheng, Bin Xiao, Jingdong Wang, Honghui Shi, Thomas S Huang, and Lei Zhang. HigherHRNet: Scale-aware representation learning for bottom-up human pose estimation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[7] Fangqiang Ding, Zhen Luo, Peijun Zhao, and Chris Xiaoxuan Lu. milliFlow: Scene flow estimation on mmWave radar point cloud for human motion sensing. In *European Conference on Computer Vision (ECCV)*, pages 202–221, 2024.

[8] Junqiao Fan, Jianfei Yang, Yuecong Xu, and Lihua Xie. Diffusion model is a good pose estimator from 3D RF-vision. In *European Conference on Computer Vision*, page 1–18, 2024.

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[10] Kaiming He, Georgia Gkioxari, Piotr Dollar, and Ross Girshick. Mask R-CNN. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(2):386–397, 2020.

[11] Yuan-Hao Ho, Jen-Hao Cheng, Sheng Yao Kuan, Zhongyu Jiang, Wenhao Chai, Hsiang-Wei Huang, Chih-Lung Lin, and Jenq-Neng Hwang. RT-Pose: A 4D radar tensor-based 3D human pose estimation and localization benchmark. In *European Conference on Computer Vision*, page 107–125, 2024.

[12] Shuting Hu, Siyang Cao, Nima Toosizadeh, Jennifer Barton, Melvin G. Hector, and Mindy J. Fain. mmPose-FK: A forward kinematics approach to dynamic skeletal pose estimation using mmWave radars. *IEEE Sensors Journal*, 24(5):6469–6481, 2024.

[13] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian 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, 2014.

[14] Niraj Prakash Kini, Ruey-Horng Shiue, ryan chandra, Wen-Hsiao Peng, Ching-Wen Ma, and Jenq-Neng Hwang. TransHuPR: Cross-view fusion transformer for human pose estimation using mmWave radar. In *British Machine Vision Conference*, 2024.

[15] 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 *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 5715–5724, 2023.

[16] Jiefeng Li, Siyuan Bian, Ailing Zeng, Can Wang, Bo Pang, Wentao Liu, and Cewu Lu. Human pose regression with residual log-likelihood estimation. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

[17] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollar. Focal loss for dense object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(2):318–327, 2020.

[18] Chris Xiaoxuan Lu, Stefano Rosa, Peijun Zhao, Bing Wang, Changhao Chen, John A. Stankovic, Niki Trigoni, and Andrew Markham. See through smoke: robust indoor mapping with low-cost mmWave radar. In *International Conference on Mobile Systems, Applications, and Services (MobiSys)*, page 14–27, 2020.

[19] Zhengxiong Luo, Zhicheng Wang, Yan Huang, Liang Wang, Tieniu Tan, and Erjin Zhou. Rethinking the heatmap regression for bottom-up human pose estimation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13259–13268, 2021.

[20] Sebastian Lutz, Richard Blythman, Koustav Ghosal, Matthew Moynihan, Ciaran Simms, and Aljosa Smolic. Jointformer: Single-frame lifting transformer with error prediction and refinement for 3D human pose estimation. In *International Conference on Pattern Recognition (ICPR)*, pages 1156–1163, 2022.

[21] Julieta Martinez, Rayat Hossain, Javier Romero, and James J Little. A simple yet effective baseline for 3D human pose estimation. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2659–2668, 2017.

[22] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. *arXiv [cs.CV]*, 2016.

[23] Ashish Pandharipande, Chih-Hong Cheng, Justin Dauwels, Sevgi Z. Gurbuz, Javier Ibanez-Guzman, Guofa Li, Andrea Piazzoni, Pu Wang, and Avik Santra. Sensing and machine learning for automotive perception: A review. *IEEE Sensors Journal*, 23(11):11097–11115, 2023.

[24] M. Mahbubur Rahman, Ryoma Yataka, Sorachi Kato, Pu (Perry) Wang, Peizhao Li, Adriano Cardace, and Petros Boufounos. MMVR: Millimeter-wave multi-view radar dataset and benchmark for indoor perception. In *European Conference on Computer Vision (ECCV)*, pages 306–322, 2024.

[25] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6):1137–1149, 2017.

[26] Avik Santra, Pu Wang, George Shaker, Bhavani Shankar Mysore, Guido Dolmans, Yan Chen, Negin Shariati, and Ashish Pandharipande. Machine learning-powered radio frequency sensing: A review. *IEEE Sensors Journal*, 25(13):23164–23183, 2025.

[27] 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, 2020.

[28] Dahu Shi, Xing Wei, Liangqi Li, Ye Ren, and Wenming Tan. End-to-end multi-person pose estimation with transformers. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11059–11068, 2022.

[29] Akash Deep Singh, Sandeep Singh Sandha, Luis Garcia, and Mani Srivastava. RadHAR: Human activity recognition from point clouds generated through a millimeter-wave radar. In *ACM Workshop on Millimeter-Wave Networks and Sensing Systems*, pages 51–56, 2019.

[30] Mikael Skog, Oleksandr Kotlyar, Vladimír Kubelka, and Martin Magnusson. Human detection from 4D radar data in low-visibility field conditions. *arXiv:2404.05307*, 2024.

[31] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for human pose estimation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5686–5696, 2019.

[32] Denis Tome, Chris Russell, and Lourdes Agapito. Lifting from the deep: Convolutional 3D pose estimation from a single image. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[33] Hong Wan, Ruiyuan Song, Chunyang Xie, Zhi Lu, Qi Chen, Zhi Wu, Dongheng Zhang, Yang Hu, and Yan Chen. QRFPose: Query-based 3D pose estimation using radio signals. In *International Conference on Wireless Communications and Signal Processing (WCSP)*, pages 1271–1277, 2024.

[34] Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, Wenyu Liu, and Bin Xiao. Deep high-resolution representation learning for visual recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(10):3349–3364, 2021.

[35] Zhi Wu, Dongheng Zhang, Chunyang Xie, Cong Yu, Jinbo Chen, Yang Hu, and Yan Chen. RFMask: A simple baseline for human silhouette segmentation with radio signals. *IEEE Transactions on Multimedia*, 25:4730–4741, 2023.

[36] Qian Xie, Qianyi Deng, Ta Ying Cheng, Peijun Zhao, Amir Patel, Niki Trigoni, and Andrew Markham. mmPoint: Dense human point cloud generation from mmWave. In *British Machine Vision Conference (BMVC)*, pages 194–196, 2023.

[37] Qian Xie, Xinyu Hou, Qianyi Deng, Amir Patel, Niki Trigoni, and Andrew Markham. mmDiffusion: mmWave diffusion for sequential 3d human dense point cloud generation. In *2025 International Conference on 3D Vision (3DV)*, pages 781–790, 2025.

[38] 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 *International Conference on Mobile Systems, Applications, and Services (MobiSys)*, page 269–282, 2021.

[39] Hongfei Xue, Qiming Cao, Yan Ju, Haochen Hu, Haoyu Wang, Aidong Zhang, and Lu Su. M^{4}esh: mmWave-based 3D human mesh construction for multiple subjects. In *Proceedings of the 20th ACM Conference on Embedded Networked Sensor Systems*, pages 391–406, 2022.

[40] Kangwei Yan, Fei Wang, Bo Qian, Han Ding, Jinsong Han, and Xing Wei. Person-in-WiFi 3D: End-to-end multi-person 3D pose estimation with Wi-Fi. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 969–978, 2024.

[41] 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. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 18756–18768, 2023.

[42] Jiarui Yang, Songpengcheng Xia, Yifan Song, Qi Wu, and Ling Pei. mmBaT: A multi-task framework for mmWave-based human body reconstruction and translation prediction. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 8446–8450, 2024.

[43] Jiarui Yang, Songpengcheng Xia, Zengyuan Lai, Lan Sun, Qi Wu, Wenxian Yu, and Ling Pei. mmDEAR: mmWave point cloud density enhancement for accurate human body reconstruction. In *IEEE International Conference on Robotics and Automation*, pages 11227–11233, 2025.

[44] Shanliang Yao, Runwei Guan, Zitian Peng, Chenhang Xu, Yilu Shi, Weiping Ding, Eng Gee Lim, Yong Yue, Hyungjoon Seo, Ka Lok Man, Jieming Ma, Xiaohui Zhu, and Yutao Yue. Exploring radar data representations in autonomous driving: A comprehensive review. *arXiv:2312.04861*, 2024.

[45] Ryoma Yataka, Adriano Cardace, Pu (Perry) Wang, Petros Boufounos, and Ryuhei Takahashi. RETR: Multi-view radar detection transformer for indoor perception. *Neural Information Processing Systems*, 37: 19839–19869, 2024.

[46] Feng Zhang, Xiatian Zhu, Hanbin Dai, Mao Ye, and Ce Zhu. Distribution-aware coordinate representation for human pose estimation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7091–7100, 2020.

[47] 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 *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7356–7365, 2018.

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

[49] Peijun Zhao, Chris Xiaoxuan Lu, Bing Wang, Niki Trigoni, and Andrew Markham. CubeLearn: End-to-end learning for human motion recognition from raw mmWave radar signals. *IEEE Internet of Things Journal*, 10(12):10236–10249, 2023.

[50] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable DETR: Deformable transformers for end-to-end object detection. In *International Conference on Learning Representations*, 2021.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction clearly outline the major challenge and our contributions on that in 3D human pose estimation achieved by the proposed architecture using bullet points. Specifically, following each bullet point, Section 4 and Section 5 provide detailed descriptions of the proposed architecture and its experimental results, demonstrating the effectiveness of the proposed method. Therefore, the overall content is consistent.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We included the discussion about the limitation in Section 6. Besides, we analyzed the limitations by visualizing the failure cases in Appendix I.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: We provide the theoretical computational complexity of our proposed system in Appendix A. Although we mainly demonstrate empirical results in this paper, we show that our proposed method is broadly applicable by performing evaluations with two different datasets.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: We provide the main parameters regarding the dataset, model architecture and model training in Table 7 in Appendix F. In addition, we provide a detailed description of the network architecture in Appendix A. This allows us to reproduce the main experimental results in the paper.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: We plan to release our source code online. In addition, the link to the datasets we used in the evaluations are provided in Section 5.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: We included the specifications of data and hyper parameters. Refer Table 7 in Appendix F.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: Due to the high computational cost, statistical results are not included. We instead report more comprehensive experimental results by conducting experiments with multiple datasets. Please refer to Section 5 and Appendix I.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Refer Table 7 in Appendix F including descriptions for computer resources we used.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We carefully confirmed it.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: We disucuss the social impact of our proposal in Section 7.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: The datasets we use in our evaluation are distributed through regular procedures. Besides, we do not use any pre-trained models, and thus there is no such a risk.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: The papers for HIBER and MMVR dataset are cited; see Section 5 that the papers are cited in the proper context.

Guidelines:

- The answer NA means that the paper does not use existing assets.

- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, paperswithcode.com/datasets has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: We plan to release the source code required to reproduce the evaluation results in the main content.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification:

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification:

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Justification:

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (https://neurips.cc/Conferences/2025/LLM) for what should or should not be described.