AG-EgoPose: Leveraging Action-Guided Motion and Kinematic Joint Encoding for Egocentric 3D Pose Estimation

Md Mushfiqur Azam⬤, John Quarles⬤, and Kevin Desai⬤

The University of Texas at San Antonio, San Antonio, TX 78249, USA
{mdmushfiqur.azam, john.quarles, kevin.desai}@utsa.edu

Abstract. Egocentric 3D human pose estimation remains challenging due to severe perspective distortion, limited body visibility, and complex camera motion inherent in first-person viewpoints. Existing methods typically rely on single-frame analysis or limited temporal fusion, which fails to effectively leverage the rich motion context available in egocentric videos. We introduce AG-EgoPose, a novel dual-stream framework that integrates short- and long-range motion context with fine-grained spatial cues for robust pose estimation from fisheye camera input. Our framework features two parallel streams: A spatial stream uses a weight-sharing ResNet-18 encoder-decoder to generate 2D joint heatmaps and corresponding joint-specific spatial feature tokens. Simultaneously, a temporal stream uses a ResNet-50 backbone to extract visual features, which are then processed by an action recognition backbone to capture the motion dynamics. These complementary representations are fused and refined in a transformer decoder with learnable joint tokens, which allows for the joint-level integration of spatial and temporal evidence while maintaining anatomical constraints. Experiments on real-world datasets demonstrate that AG-EgoPose achieves state-of-the-art performance in both quantitative and qualitative metrics. Code is available at: https://github.com/Mushfiq5647/AG-EgoPose.

Keywords: Egocentric 3D pose estimation · Transformer-based fusion · Spatiotemporal modeling

1 Introduction

Egocentric 3D human pose estimation is increasingly important for augmented reality (AR), virtual reality (VR), and human–computer interaction (HCI). As wearable and head-mounted cameras become more common, AR/VR systems need an accurate full-body pose of the camera wearer to enable immersive interaction. However, first-person views are particularly challenging: the field of view is narrow, perspective distortion is severe, and large parts of the body are often self-occluded or completely outside the image.

Most classical pose estimators [5,8,23], designed for third-person cameras, do not handle these effects well. Recent methods tailored to first-person views,

such as xR-EgoPose [25], Scene-Ego [28], Ego-VisionSpan [10], EgoPW [26], and EgoGlobalMocap [27], have improved robustness to perspective distortion, self-occlusion, and out-of-view joints. However, they still tend to struggle in dynamic, interactive scenes where several people move and interact in close proximity. A few approaches [28,1,19] start to use scene context to recover occluded body parts, such as depth cues or inter-person relations, but the potential of short- and long range motion patterns as a contextual signal remains unexplored.

Our work targets this gap by leveraging action-driven context together with the structure of the human skeleton. We hypothesize that motion cues can help reconstruct body parts that are intermittently invisible to the camera. By capturing these temporal dynamics and explicitly modeling joint dependencies, our method improves the fidelity of reconstructed motion, and remains robust in the dynamic environments where existing approaches often fail.

To this end, we propose *AG-EgoPose*, a dual-stream, end-to-end framework. The spatial stream uses a weight-shared ResNet-18 encoder–decoder to generate 2D joint heatmaps from RGB frames and converts them into compact, joint-specific tokens. In parallel, a temporal stream employs an ActionFormer[32]-based motion encoder that processes a sequence of RGB frames using multi-scale temporal attention to capture both short- and long-range motion dynamics. We concatenate heatmap and motion features into a unified memory, then use a Transformer decoder with learnable joint tokens—self-attending across joints and cross-attending to the memory to predict 3D joint coordinates. We evaluate AG-EgoPose on real-world egocentric datasets and show that it achieves state-of-the-art performance in both quantitative metrics and qualitative visualizations. Our primary contributions are:

- We introduce action-informed 3D pose estimation that injects short- and long-term temporal context into the pose regressor, significantly improving pose disambiguation in the presence of occlusion and motion blur.
- We propose an adaptive spatio-temporal fusion strategy that converts 2D joint heatmaps into compact per-joint tokens and uses a Transformer decoder with learnable joint tokens to selectively attend to spatial evidence and temporal motion features, enabling accurate 3D joint regression.
- We provide a comprehensive evaluation on real-world datasets, demonstrating that our method achieves state-of-the-art performance on the EgoPW [26] and SceneEgo [28] datasets.

2   Related Work

Egocentric 3D human pose estimation is critical for AR/VR, telepresence, and embodied AI, yet remains challenging due to severe perspective distortion, frequent self-occlusions, and limited field-of-view. A comprehensive survey [4] reviews datasets, sensing setups, and model families in this space. Early head-mounted systems like EgoCap [21] demonstrated feasibility but lacked real-time performance. Subsequent work improved robustness by wide-view, fisheye, and stereo configurations [28,3,1], while recent designs extend HMDs by adding rear cameras beyond frontal views to improve full-body tracking under occlusion [2].

Heatmap-guided 2D-to-3D pose refinement. Many recent methods first estimate 2D joint heatmaps and then lift or refine them using additional cues. Mo^2Cap^2 [30] and EgoGlass [34] focus on improving 2D joint evidence under occlusion, while xR-EgoPose [25] and SelfPose [24] employ encoder–decoder architectures to refine pose from monocular VR imagery. EgoTAP [12] summarizes heatmaps into compact spatial descriptors to better handle noisy joints.

Scene-aware and geometry-assisted approaches. To address out-of-view body parts and self-occlusions, several methods incorporate depth or global scene constraints. SceneEgo [28] predicts wide-view depth and uses depth inpainting to mitigate occlusions. InvisiblePose [9] combines motion signatures with scene structure via a classifier-based design. Ego-VisionSpan [10] leverages SLAM and geometric consistency for stable estimates. These approaches are effective but rely on additional scene reconstruction signals.

 Temporal context modeling. A growing body of work emphasizes the importance of temporal context for egocentric motion understanding. EgoFormer [16] and related transformer-based designs aim to capture longer-range dynamics in AR/VR settings. EgoSTAN [20] employs a spatiotemporal transformer to handle distortion and improve occluded joint estimation. Social and interaction context has also been shown to contribute to improved estimation, as demonstrated by Ego+X [17] and You2Me [19].

Fisheye-aware and mesh/motion-focused methods. Perspective distortion from head-mounted fisheye cameras has motivated specialized architectures [11,20,33]. Beyond skeleton estimation, notable works extend egocentric understanding to richer body representations and motion. Fish2Mesh [22] introduces a fisheye-aware transformer for egocentric 3D human mesh recovery. Recent work has also explored diffusion-based formulations for egocentric whole-body motion recovery. EgoEgo [15] and EgoAllo [31] estimate body motion in the scene frame by leveraging egocentric SLAM and diffusion-based modeling, while REWIND [14] introduces a real-time diffusion framework for whole-body motion estimation from head-mounted camera inputs.

Unlike existing methods that rely on stereo, SLAM, or depth, we instead exploit the underused potential of motion dynamics as an action-guided prior for resolving egocentric pose ambiguity. Our monocular framework fuses these action-informed features with spatial heatmap evidence in a joint-level Transformer decoder with learnable joint queries, enabling precise 3D pose estimation without additional sensors or costly scene reconstruction.

3   Method

Our goal is to estimate the 3D joint positions of the camera wearer from a sequence of frames captured by a head-mounted fisheye camera. As shown in Fig. 1, *AG-EgoPose* takes a video segment of T consecutive frames, $I_{\mathrm{seq}} = \{I_1, \ldots, I_T\}$, and predicts the corresponding 3D joint positions $P_{\mathrm{seq}}^g = \{P_1^g, \ldots, P_T^g\}$ using a two-stage end-to-end framework. First, we train a heatmap prediction network to estimate 2D joint heatmaps from fisheye frames, which are embedded into compact joint-specific tokens. In parallel, we extract static visual features using
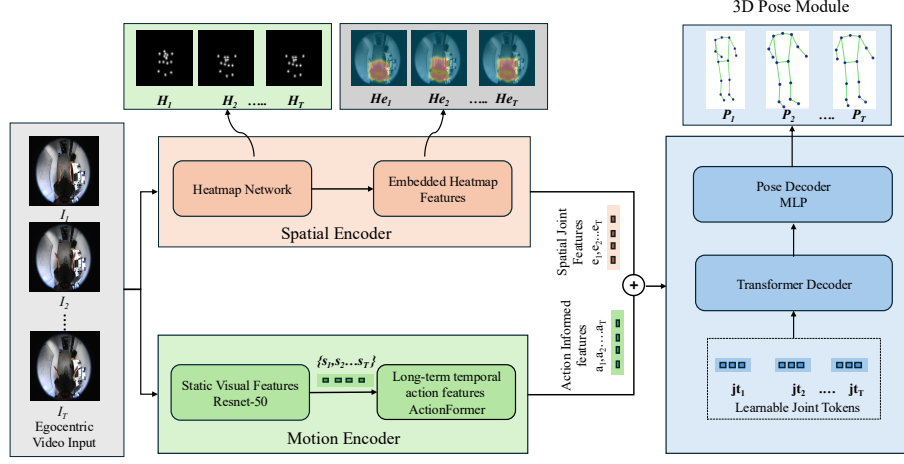
Fig. 1: Overview of our egocentric 3D pose estimation model. Egocentric fisheye video frames are processed by two parallel streams: (1) **Spatial Encoder:** generates 2D joint heatmaps using a weight-sharing ResNet-18 encoder–decoder with unified skip connections and encodes spatial joint features. (2) **Motion Encoder:** an ActionFormer [32] based temporal encoder operates on visual features extracted using ResNet-50 to capture short- and long-term motion dynamics. Spatial and temporal features are concatenated per joint to form a joint-level memory. A transformer decoder with learnable joint queries attends to this memory, enabling joint-specific integration of spatial and temporal evidence. The decoder output is then passed through a pose head to regress 3D joint coordinates.

a ResNet-50 pretrained on ImageNet [6], keeping only its final layers trainable, and encode them with an ActionFormer [32] backbone to capture short- and long-term motion context (Sec. 3.2). A transformer decoder with self-attention and cross-attention mechanisms refines concatenated spatial joint features and action-guided motion features using learnable joint tokens to regress 3D joint positions across the sequence window (Sec. 3.3).

3.1   Spatial Feature Extraction Module

2D Heatmap Estimation. This module provides explicit spatial supervision that bridges RGB input and 3D pose estimation. We take a sequence of RGB frames $\{I_t\}_{t=1}^T$, where $I_t \in \mathbb{R}^{256 \times 256 \times 3}$, and predict corresponding 2D joint heatmaps $\{\mathbf{H}_t\}_{t=1}^T$ with $\mathbf{H}_t \in \mathbb{R}^{64 \times 64 \times 15}$ as depicted
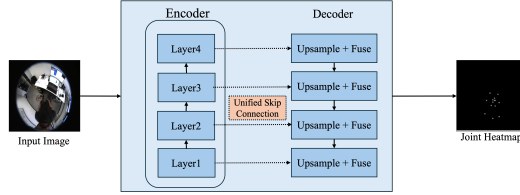

Fig. 2: 2D heatmap prediction network with ResNet-18 encoder and FPN decoder using unified skip connections.

in Fig. 2. Following [34], we use a shared ResNet-18 backbone with a Feature

Pyramid Network (FPN) decoder and unified skip connections [29] to fuse multi-scale features while preserving fine-grained spatial detail at 64×64 resolution. Weight sharing across joints allows geometrically similar joints (e.g., left/right wrists) to share representations, improving data efficiency and generalization. We train the heatmap network using `BCEWithLogitsLoss`, which enforces sharp, localized confidence maps via pixel-wise supervision:

$$\mathcal{L}_{2D} = \texttt{BCEWithLogitsLoss}(\hat{H}, H), \tag{1}$$

where H and $\hat{H}$ denote the ground-truth and predicted heatmaps. These heatmaps provide reliable joint localization cues for downstream 3D pose regression.

Joint Heatmap Embedding. The joint heatmaps $\mathbf{H} \in \mathbb{R}^{64 \times 64 \times 15}$ predicted using the heatmap network are converted into joint-specific feature vectors using a convolutional embedding module. A 3-layer CNN with downsampling and adaptive pooling transforms each heatmap $\mathbf{H}_j \in \mathbb{R}^{64 \times 64}$ into a 128-dimensional embedding.

$$\mathbf{e}_j = f_{\text{CNN}}(\mathbf{H}_j) \in \mathbb{R}^{128}, \quad j = 1, \ldots, 15 \tag{2}$$

where f_{CNN} denotes the convolutional embedding function. The final embedding set is represented as

$$\mathbf{H_e} = [\mathbf{e}_1, \ldots, \mathbf{e}_{15}] \in \mathbb{R}^{15 \times 128}$$

This embedding preserves spatial semantics while reducing dimensionality for efficient transformer processing, allowing the model to distinguish joint configurations and reason about occlusions through learned spatial patterns.

3.2 Action-Guided Motion Feature Module

Our motion-aware encoder complements the 2D heatmap stream with appearance and scene context specific to egocentric video. We use a ResNet-50 backbone pre-trained on ImageNet [6] and update only the last block (layer4), while keeping earlier layers frozen. In practice, this lets us reuse strong generic features and adapt high-level filters to typical egocentric artifacts such as tilted viewpoints, motion blur, and indoor lighting. The resulting features not only describe what is visible in the frame, but also capture scene elements (e.g., floors, countertops, furniture) linked to certain poses, which helps when parts of the body are outside the field of view.

Static Visual Feature Extraction. Given a sequence of T frames $\{f_t\}_{t=1}^{T}$, each frame is passed through ResNet-50 to obtain a global feature $\mathbf{v}_t \in \mathbb{R}^{2048}$. A linear layer followed by batch normalization projects this to $\mathbf{s}_t \in \mathbb{R}^{384}$. Stacking these embeddings over batch and time gives

$$F_s = [\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_T] \in \mathbb{R}^{B \times T \times 384}, \tag{3}$$

which we use as the static appearance input to the temporal motion encoder.

Temporal motion encoder. To model motion semantics, we feed F_s into the *branch* layer of ActionFormer [32], used here purely as a temporal encoder. We modify its original stack of eight Transformer blocks into a *local-to-global* configuration: the first four blocks apply LocalMaskedMHCA over a window of $w=8$ frames to focus on short-range motion (small limb shifts, hand swings), while the remaining four use MaskedMHCA over the full sequence to aggregate long-range dependencies and overall body dynamics. Formally, with $\mathbf{z}_t^{(0)} = \mathbf{s}_t$,

$$
\mathbf{z}_t^{(i)} = \begin{cases} \text{LocalMaskedMHCA}(\mathbf{z}_{t-w:t+w}^{(i-1)}) + \text{FFN}(\mathbf{z}_t^{(i-1)}), & i \leq 4, \\ \text{MaskedMHCA}(\mathbf{z}_{1:T}^{(i-1)}) + \text{FFN}(\mathbf{z}_t^{(i-1)}), & i > 4. \end{cases}
$$

We discard ActionFormer's detection head and initialize the branch from an Ego4D-pretrained checkpoint [7], keeping all its parameters frozen. This preserves strong priors on egocentric motion while avoiding additional trainable temporal parameters. After the 8 temporal blocks, we take the time-aligned output sequence as the motion embedding, $F_m \in \mathbb{R}^{B \times T \times 384}$, which provides motion-aware features that explicitly combine short- and long-range temporal structure to guide pose inference.

3.3   3D Pose Decoder Module

We regress 3D joint coordinates using a Transformer decoder that integrates learnable joint tokens with a fused multi-modal memory. This design explicitly couples spatial evidence from heatmaps with temporal motion cues, enabling robust estimation even under occlusion or out-of-view joints.

Multi-modal memory. Spatial features are extracted from heatmap-based joint embeddings $\mathbf{H}_e \in \mathbb{R}^{B \times T \times J \times 128}$, while temporal context is encoded in action-informed features $\mathbf{F}_m \in \mathbb{R}^{B \times T \times 384}$. We project $\mathbf{F}_m$ into $d = 128$ dimensions, repeat it across joints, and concatenate with $\mathbf{H}_e$:

$$
\mathbf{M} = \phi\Big([\mathbf{H}_e \,\|\, \tilde{\mathbf{F}}_m]\mathbf{W}_c\Big) \in \mathbb{R}^{B \times T \times J \times d} \tag{4}
$$

where $\tilde{\mathbf{F}}_m$ is the joint-tiled motion embedding and $\mathbf{W}_c$ is a learned linear projection. This produces joint-conditioned memory that preserves spatial precision while enriching it with temporal semantics.

Learnable joint tokens and decoding. We introduce $J = 15$ learnable joint token vectors $\mathbf{Q} \in \mathbb{R}^{J \times d}$, one for each joint. These queries are tiled across time to shape $(B \times T, J, d)$ for batch processing. A stack of 3 Transformer decoder layers, each with 4 attention heads and feedforward dimension $4d$, refines these tokens through self-attention among joint queries and cross-attention over the multi-modal memory $\mathbf{M}$:

$$
\mathbf{F_d} = \text{Decoder}(\mathbf{Q}, \mathbf{M}) \in \mathbb{R}^{B \times T \times J \times d} \tag{5}
$$

Since the memory $\mathbf{M}$ already encodes temporal information via the motion features $\mathbf{F}_m$, we do not add temporal positional encodings to the queries; the decoder learns to attend to temporally-appropriate memory through cross-attention. By allocating a query to each joint, the decoder learns kinematics-aware retrieval patterns, allowing joint estimates to remain consistent under occlusion or motion blur.

The decoded features are mapped to 3D joint coordinates using a lightweight MLP with two hidden layers followed by LeakyReLU activations and dropout:

$$\mathbf{P}_{3\mathrm{D}} = \mathrm{MLP}(\mathbf{F_d}) \in \mathbb{R}^{B \times T \times J \times 3} \tag{6}$$

3.4   Loss Functions

We train the network using a composite loss that enforces accurate joint localization and kinematic consistency. Let $\hat{\mathbf{X}}, \mathbf{X} \in \mathbb{R}^{J \times 3}$ denote predicted and ground-truth joint coordinates with joint set $\mathcal{J} = \{1, \ldots, J\}$, and let $\mathcal{B}$ be the set of bones defined by directed edges $(p \to c)$ in the kinematic tree. We define bone vectors as

$$\hat{\mathbf{b}}_{pc} = \hat{\mathbf{X}}_c - \hat{\mathbf{X}}_p, \qquad \mathbf{b}_{pc} = \mathbf{X}_c - \mathbf{X}_p.$$

Joint position loss. We penalize per-joint 3D errors using the mean joint position error:

$$\mathcal{L}_{\mathrm{pos}} = \frac{1}{J} \sum_{j \in \mathcal{J}} \|\hat{\mathbf{X}}_j - \mathbf{X}_j\|_2.$$

Bone consistency losses. To enforce anatomically plausible poses, we regularize both bone length and orientation. The bone-length loss preserves limb proportions:

$$\mathcal{L}_{\mathrm{bone}} = \frac{1}{|\mathcal{B}|} \sum_{(p,c) \in \mathcal{B}} \left(\|\hat{\mathbf{b}}_{pc}\|_2 - \|\mathbf{b}_{pc}\|_2 \right)^2.$$

To align bone orientations, we minimize the negative cosine similarity between predicted and ground-truth bone vectors:

$$\mathcal{L}_{\mathrm{cos}} = -\frac{1}{|\mathcal{B}|} \sum_{(p,c) \in \mathcal{B}} \frac{\hat{\mathbf{b}}_{pc} \cdot \mathbf{b}_{pc}}{\|\hat{\mathbf{b}}_{pc}\|_2 \, \|\mathbf{b}_{pc}\|_2},$$

The cosine term is larger for better-aligned bones, so the minus sign simply turns this similarity into a loss that decreases as alignment improves—minimizing $\mathcal{L}_{\mathrm{cos}}$ thus maximizes cosine similarity.

Overall objective. We combine these terms as

$$\mathcal{L} = \lambda_{\mathrm{pos}} \, \mathcal{L}_{\mathrm{pos}} + \lambda_{\mathrm{bone}} \, \mathcal{L}_{\mathrm{bone}} + \lambda_{\mathrm{cos}} \, \mathcal{L}_{\mathrm{cos}},$$

with $\lambda_{\mathrm{pos}} = 1.0$, $\lambda_{\mathrm{bone}} = 0.1$, and $\lambda_{\mathrm{cos}} = 0.01$, balancing localization accuracy and kinematic validity.

4   Experimental Setup

4.1   Datasets

We train and evaluate our model on the EgoPW [26] and SceneEgo [28] datasets to balance realism and diversity in egocentric 3D pose estimation.

EgoPW Dataset. EgoPW [26] is a large-scale in-the-wild egocentric 3D pose dataset captured with a head-mounted fisheye camera and a synchronized external camera. It contains over 318K frames from 10 actors performing 20 everyday activities. Because obtaining ground-truth 3D annotations is infeasible in these settings, EgoPW provides pseudo labels generated by a spatio-temporal optimization framework that fuses egocentric and external observations.

SceneEgo Dataset. SceneEgo [28] is a real-human egocentric dataset with 28K images of two actors performing diverse daily activities. We use it to evaluate cross-dataset generalization.

4.2   Implementation Details

2D heatmap training. We first train our heatmap network on the official training splits of EgoPW [26] and SceneEgo [28] to predict 2D joint heatmaps (J=15). We resize frames to 256×256 and train for 20 epochs with batch size 8 using `BCEWithLogitsLoss`. We generate ground-truth heatmaps by placing Gaussian kernels (σ=2) at 2D joint locations:

$$H_j(u,v) = \exp\left(-\frac{(u-u_j)^2 + (v-v_j)^2}{2\sigma^2}\right),\tag{7}$$

where (u_j, v_j) is the annotated 2D position of joint j. This produces smooth confidence maps that provide stable spatial supervision. Since SceneEgo does not include 2D annotations, we obtain them by projecting its 3D joints into the image plane using the provided fisheye calibration. We use mixed-precision training with gradient clipping and cosine-annealing learning-rate scheduling for stable convergence. After training, we freeze the heatmap network and use it to extract per-joint spatial cues for downstream 3D estimation.

3D pose estimation training. We train our 3D pose model on EgoPW [26] (official split) and report results on its test set. To evaluate generalization, we fine-tune the EgoPW-pretrained model on the SceneEgo [28] training split and test on its test split. We use a sliding-window setup with sequence length T=64 and stride 32. We resize and normalize all frames to 256×256. We optimize with Adam for 30 epochs using a learning rate decayed from 10^{-3} to 10^{-4} via cosine annealing and batch size 8. We keep the heatmap network frozen during this stage.

Table 1: Comparison of PA-MPJPE (mm) with prior egocentric 3D pose estimation methods on the EgoPW [26] dataset.

Method	PA-MPJPE↓
SelfPose [24]	112.0
Mo^{2}Cap^2 [30]	102.3
Kolotouros et al. [13]	88.6
EgoHMR [18]	85.8
EgoPW [26]	84.2
Ours	**76.7**

Table 2: Comparison of MPJPE (mm) and PA-MPJPE (mm) with prior egocentric 3D pose estimation methods in SceneEgo [28] dataset.

Method	MPJPE↓	PA-MPJPE↓
Mo^{2}Cap^2 [30]	200.3	121.2
xR-EgoPose [25]	241.3	133.9
EgoPW [26]	189.6	105.3
SceneEgo [28]	118.5	92.7
Ours	**104.0**	**76.2**

4.3 Evaluation Metrics

We evaluate using two standard 3D pose metrics. Mean Per Joint Error (**MPJPE**) computes the mean Euclidean distance between predicted and ground-truth 3D joints reflecting absolute accuracy. **PA-MPJPE** first applies Procrustes alignment (rotation, translation, and scale) and then measures MPJPE, reflecting reconstruction quality independent of global pose and scale.

5 Results

In this section, we compare our *AG-EgoPose* method with state-of-the-art models on the EgoPW [26] and SceneEgo [28] test datasets to assess its effectiveness in estimating 3D body pose.

5.1 Quantitative Results

Table 1 shows the results on the EgoPW dataset [26]. Since EgoPW provides pseudo-labeled 3D poses from external-view supervision, PA-MPJPE is the standard metric here. Our model achieves **76.7** mm PA-MPJPE, improving over the previous best of 84.2 mm [26] by 7.5 mm reduction (9% relative). This gain indicates lower joint localization error while preserving structural consistency.

For cross-dataset evaluation, we transfer the EgoPW-pretrained model to SceneEgo. Table 2 reports **104.0** mm MPJPE and **76.2** mm PA-MPJPE, improving over the prior best (118.5/92.75 mm) by **14.5**/**16.5** mm, respectively

Table 3: Comparison of accuracy and computational efficiency with prior egocentric 3D pose estimation methods on the SceneEgo dataset [28].

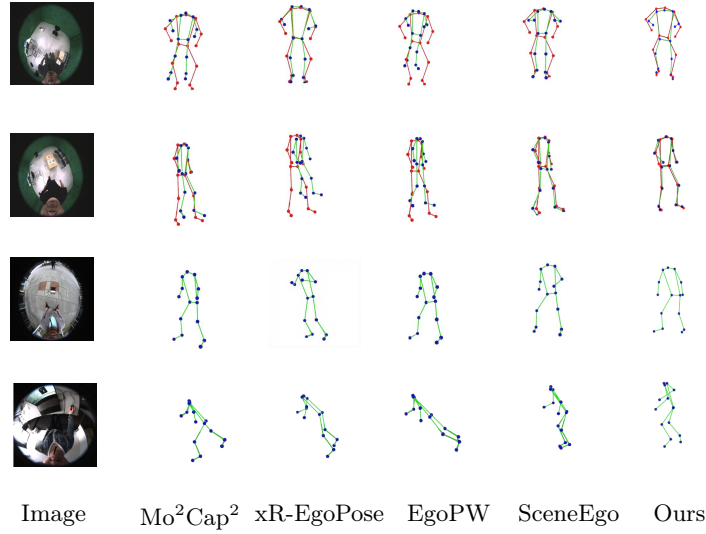| Method	MPJPE↓	PA-MPJPE↓	Params	FLOPs
xR-EgoPose [25]	241.3	133.9	14.99M	0.83G
SceneEgo [28]	118.5	92.7	11.96M	8.4G
Ours	**104.0**	**76.2**	11.4M	8.0G

Fig. 3: Qualitative comparison between our method and state-of-the-art egocentric 3D pose estimation methods. From left to right, we show the input image followed by the results of Mo^{2}Cap^2, xR-EgoPose, EgoPW, SceneEgo, and our method. The top two rows are from the SceneEgo [28] dataset, where ground-truth poses are shown in red. The bottom two rows are from the EgoPW [26] dataset (without ground-truth poses).

corresponding to a relative improvement of nearly 18% in PA-MPJPE. Unlike the previous state of the art [28], which uses auxiliary signals such as ground-truth depth and semantic masks, we achieve higher accuracy with a simple transfer setup, demonstrating strong generalization across egocentric benchmarks.

Computational Efficiency. Table 3 shows that our method improves accuracy while remaining computationally efficient. Although xR-EgoPose [25] uses fewer FLOPs (0.83G), it incurs much higher error (241.3 mm MPJPE vs. 104.0 mm). Its low cost comes from a convolution-heavy design with linear layers applied only to compressed CNN features, which limits capacity. In contrast, we achieve a better accuracy–efficiency trade-off, reducing error substantially while using 8.0G FLOPs and fewer parameters (11.4M vs. 14.99M).

5.2   Qualitative Results

Figure 3 shows qualitative results on both the studio dataset and in-the-wild sequences. Compared with Mo^{2}Cap^2 [30] and xR-EgoPose [25], our method produces more stable and anatomically consistent poses under egocentric distortion and partial visibility. While EgoPW [26] and SceneEgo [28] predictors improve

Table 4: Ablation study on EgoPW dataset: impact of spatial heatmap features (H_e), motion features (F_m), and cross-attention fusion (F_d) on PA-MPJPE (mm).

Spatial (H_e)	Motion (F_m)	Cross Attn (F_d)	PA-MPJPE↓
✓	✓		83.1
✓		✓	90.8
	✓	✓	78.8
✓	✓	✓	76.7

Table 5: Ablation study on SceneEgo dataset: impact of spatial heatmap features (H_e), motion features (F_m), and cross-attention fusion (F_d) on MPJPE/PA-MPJPE (mm).

Spatial (H_e)	Motion (F_m)	Cross Attn (F_d)	MPJPE↓	PA-MPJPE↓
✓	✓		114.3	81.6
✓		✓	113.2	80.8
	✓	✓	108.1	79.6
✓	✓	✓	104.0	76.2

over earlier baselines, they still miss or distort body parts in challenging frames. Our predictions better preserve upper-body structure and leg articulation, particularly under occlusion or when joints move out of the field of view.

5.3   Ablation Studies

We ablate our model on EgoPW and SceneEgo datasets to show the effectiveness of our model and understand the individual contributions of spatial heatmap features, temporal motion features, and cross-attention fusion mechanisms.

Spatial heatmap features (H_e). As reported in Table 4 and Table 5, using only the spatial stream (embedded heatmap features) with cross-attention mechanism gives 90.8 mm PA-MPJPE on EgoPW [26] and 113.2 mm MPJPE / 80.8 mm PA-MPJPE on SceneEgo [28]. Adding motion features improves EgoPW to 76.7 mm PA-MPJPE (14.1 mm reduction), showing spatial evidence benefits from temporal context in egocentric sequences.

Temporal motion features (F_m). Using only the motion encoder with cross-attention yields 78.8 mm PA-MPJPE on EgoPW [26] and 108.1 mm MPJPE / 79.6 mm PA-MPJPE on SceneEgo [28] (Table 4, Table 5). This indicates strong temporal cues, but the remaining gap to the full model suggests complementary spatial evidence is still crucial, especially under rapid motion and occlusion.

Significance of cross-attention fusion (F_d). Our transformer decoder with learnable joint tokens enables adaptive refinement of fused spatial and temporal cues; removing cross-attention degrades performance. Without cross-attention, combining spatial and motion features yields 83.1 mm PA-MPJPE on EgoPW [26] and 114.3 mm MPJPE / 81.6 mm PA-MPJPE on SceneEgo [28] (Table 4, Table 5). This confirms cross-attention is critical for joint-specific fusion and resolving conflicts between spatial and temporal representations.

Impact of Heatmap Loss Function. In Table 6, we compare heatmap pre-training losses (MSE vs. BCEWithLogitsLoss) and their impact on downstream

Table 6: Comparison of MSE and BCEWithLogitsLoss for heatmap pretraining on SceneEgo [28] and EgoPW [26]. Results are in mm (lower is better).

Dataset	Loss Function	MPJPE ↓	PA-MPJPE ↓
SceneEgo	MSE	108.2	80.3
	BCEWithLogitsLoss	**104.0**	**76.2**
EgoPW	MSE	–	83.1
	BCEWithLogitsLoss	–	**76.7**

Table 7: Comparison of temporal window sizes T on EgoPW [26] and SceneEgo [28] datasets.

T (frames)	EgoPW	SceneEgo	
	PA-MPJPE ↓	MPJPE ↓	PA-MPJPE ↓
32	77.0	106.6	76.9
64	76.7	104.0	76.2
128	77.7	107.5	77.0

3D pose estimation on EgoPW and SceneEgo. On EgoPW [26], switching from MSE to BCEWithLogitsLoss reduces PA-MPJPE from 83.1 to 76.7 mm (7.7% improvement). On SceneEgo [28], BCEWithLogitsLoss achieves 104.0 mm MPJPE and 76.2 mm PA-MPJPE versus 108.2 mm and 80.3 mm with MSE (5.1% gain in PA-MPJPE). We attribute the gains to BCEWithLogitsLoss, which models joint presence as confidence distributions, yielding sharper peaks, fewer background activations, and more accurate localization for improved 3D estimation.

Effect of sequence length. Table 7 studies the temporal window size with $T \in \{32, 64, 128\}$. We find that T=64 yields the best overall performance, improving EgoPW (PA-MPJPE 76.7) and SceneEgo (MPJPE 104.0 / PA-MPJPE 76.2) compared to T=32. This indicates that a 64-frame temporal window better balances *short-range* motion cues (local limb shifts and rapid corrections) with *long-range* temporal context (motion consistency and occlusion recovery) that are critical for egocentric pose estimation. With shorter windows, the model may miss longer-term dependencies needed to stabilize ambiguous frames. Increasing the window further to T=128 hurts performance, likely because very long clips mix in viewpoint changes and irrelevant motion that dilute pose-relevant cues.

Joint-wise error analysis. In Fig. 4, we compare joint-wise errors with and without temporal motion encoder for both EgoPW and SceneEgo dataset. Using only spatial joint features leads to higher errors, especially in the lower body, where joints frequently move out of view and appearance cues alone are ambiguous. Adding temporal motion features consistently reduces error across almost all joints. This confirms that short- and long-range motion context provides strong

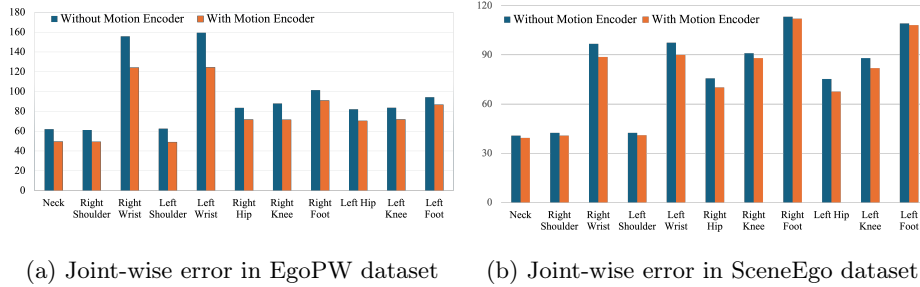(a) Joint-wise error in EgoPW dataset   (b) Joint-wise error in SceneEgo dataset

Fig. 4: Joint-wise error analysis for EgoPW and SceneEgo datasets.

temporal priors that stabilize egocentric pose estimation where visual evidence is weaker.

6   Conclusion

We present a dual-stream architecture for egocentric 3D pose estimation that combines explicit spatial evidence with short- and long-term temporal context. We use a pre-trained heatmap network to extract 2D joint cues, embed them into compact per-joint tokens, and fuse them with action-guided motion features. A transformer decoder with learnable joint tokens performs cross-attention to adaptively integrate the spatial and motion streams, yielding robust predictions.

Our current framework focuses on single-person estimation within fixed temporal windows for stable sequence-level inference. In real-world egocentric settings, multiple people may enter/leave the field of view and action durations vary beyond fixed windows. We plan to extend our approach to multi-person estimation and adaptive temporal windows, and explore multi-scale heatmap representations with adaptive spatio–temporal fusion.

References

1. Akada, H., Wang, J., Golyanik, V., Theobalt, C.: 3d human pose perception from egocentric stereo videos. In: Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition. pp. 767–776 (2024)
2. Akada, H., Wang, J., Golyanik, V., Theobalt, C.: Bring your rear cameras for egocentric 3d human pose estimation. arXiv preprint arXiv:2503.11652 (2025)
3. Akada, H., Wang, J., Shimada, S., Takahashi, M., Theobalt, C., Golyanik, V.: Unrealego: A new dataset for robust egocentric 3d human motion capture. In: European Conference on Computer Vision. pp. 1–17. Springer (2022)

4. Azam, M.M., Desai, K.: A survey on 3d egocentric human pose estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1643–1654 (2024)
5. Cao, Z., Simon, T., Wei, S.E., Sheikh, Y.: Realtime multi-person 2d pose estimation using part affinity fields. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 7291–7299 (2017)
6. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition. pp. 248–255. Ieee (2009)
7. Grauman, K., Westbury, A., Byrne, E., Chavis, Z., Furnari, A., Girdhar, R., Hamburger, J., Jiang, H., Liu, M., Liu, X., et al.: Ego4d: Around the world in 3,000 hours of egocentric video. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 18995–19012 (2022)
8. Güler, R.A., Neverova, N., Kokkinos, I.: Densepose: Dense human pose estimation in the wild. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 7297–7306 (2018)
9. Jiang, H., Grauman, K.: Seeing invisible poses: Estimating 3d body pose from egocentric video. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 3501–3509. IEEE (2017)
10. Jiang, H., Ithapu, V.K.: Egocentric pose estimation from human vision span. In: 2021 IEEE/CVF International Conference on Computer Vision (ICCV). pp. 10986–10994. IEEE (2021)
11. Kang, T., Lee, K., Zhang, J., Lee, Y.: Ego3dpose: Capturing 3d cues from binocular egocentric views. In: SIGGRAPH Asia 2023 Conference Papers. pp. 1–10 (2023)
12. Kang, T., Lee, Y.: Attention-propagation network for egocentric heatmap to 3d pose lifting. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 842–851 (2024)
13. Kolotouros, N., Pavlakos, G., Jayaraman, D., Daniilidis, K.: Probabilistic modeling for human mesh recovery. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 11605–11614 (2021)
14. Lee, J., Xu, W., Richard, A., Wei, S.E., Saito, S., Bai, S., Wang, T.L., Sung, M., Kim, T.K., Saragih, J.: Rewind: Real-time egocentric whole-body motion diffusion with exemplar-based identity conditioning. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 7095–7104 (2025)
15. Li, J., Liu, K., Wu, J.: Ego-body pose estimation via ego-head pose estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 17142–17151 (2023)
16. Li, T., Zhang, C., Su, W., Liu, Y.: Egoformer: Transformer-based motion context learning for ego-pose estimation. In: 2023 IEEE International Conference on Systems, Man, and Cybernetics (SMC). pp. 4052–4057. IEEE (2023)
17. Liu, Y., Yang, J., Gu, X., Guo, Y., Yang, G.Z.: Ego+ x: An egocentric vision system for global 3d human pose estimation and social interaction characterization. In: 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 5271–5277. IEEE (2022)
18. Liu, Y., Yang, J., Gu, X., Guo, Y., Yang, G.Z.: Egohmr: Egocentric human mesh recovery via hierarchical latent diffusion model. In: 2023 IEEE International Conference on Robotics and Automation (ICRA). pp. 9807–9813. IEEE (2023)
19. Ng, E., Xiang, D., Joo, H., Grauman, K.: You2me: Inferring body pose in egocentric video via first and second person interactions. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9890–9900 (2020)

20. Park, J., Kaai, K., Hossain, S., Sumi, N., Rambhatla, S., Fieguth, P.: Domain-guided spatio-temporal self-attention for egocentric 3d pose estimation. In: Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. pp. 1837–1849 (2023)
21. Rhodin, H., Richardt, C., Casas, D., Insafutdinov, E., Shafiei, M., Seidel, H.P., Schiele, B., Theobalt, C.: Egocap: egocentric marker-less motion capture with two fisheye cameras. ACM Transactions on Graphics (TOG) **35**(6), 1–11 (2016)
22. Shen, T., Puranik, A., Vong, J., Deogirikar, V., Fell, R., Dietrich, J., Kyrarini, M., Kitts, C., Jeong, D.C.: Fish2mesh transformer: 3d human mesh recovery from egocentric vision. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 6498–6507 (October 2025)
23. Sun, K., Xiao, B., Liu, D., Wang, J.: Deep high-resolution representation learning for human pose estimation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 5693–5703 (2019)
24. Tome, D., Alldieck, T., Peluse, P., Pons-Moll, G., Agapito, L., Badino, H., de la Torre, F.: Selfpose: 3d egocentric pose estimation from a headset mounted camera. IEEE Transactions on Pattern Analysis and Machine Intelligence **45**(6), 6794–6806 (2023). https://doi.org/10.1109/TPAMI.2020.3029700
25. Tome, D., Peluse, P., Agapito, L., Badino, H.: xr-egopose: Egocentric 3d human pose from an hmd camera (October 2019)
26. Wang, J., Liu, L., Xu, W., Sarkar, K., Luvizon, D., Theobalt, C.: Estimating egocentric 3d human pose in the wild with external weak supervision. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13157–13166 (2022)
27. Wang, J., Liu, L., Xu, W., Sarkar, K., Theobalt, C.: Estimating egocentric 3d human pose in global space. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 11500–11509 (October 2021)
28. Wang, J., Luvizon, D., Xu, W., Liu, L., Sarkar, K., Theobalt, C.: Scene-aware egocentric 3d human pose estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 13031–13040 (June 2023)
29. Weng, W., Zhu, X.: Inet: convolutional networks for biomedical image segmentation. Ieee Access **9**, 16591–16603 (2021)
30. Xu, W., Chatterjee, A., Zollhoefer, M., Rhodin, H., Fua, P., Seidel, H.P., Theobalt, C.: Mo 2 cap 2: Real-time mobile 3d motion capture with a cap-mounted fisheye camera. IEEE transactions on visualization and computer graphics **25**(5), 2093–2101 (2019)
31. Yi, B., Ye, V., Zheng, M., Li, Y., Müller, L., Pavlakos, G., Ma, Y., Malik, J., Kanazawa, A.: Estimating body and hand motion in an ego-sensed world. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 7072–7084 (2025)
32. Zhang, C.L., Wu, J., Li, Y.: Actionformer: Localizing moments of actions with transformers. In: European Conference on Computer Vision. pp. 492–510. Springer (2022)
33. Zhang, Y., You, S., Gevers, T.: Automatic calibration of the fisheye camera for egocentric 3d human pose estimation from a single image. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 1772–1781 (2021)
34. Zhao, D., Wei, Z., Mahmud, J., Frahm, J.M.: Egoglass: Egocentric-view human pose estimation from an eyeglass frame. In: 2021 International Conference on 3D Vision (3DV). pp. 32–41. IEEE (2021)