EgoPoseFormer: A Simple Baseline for Stereo Egocentric 3D Human Pose Estimation

Chenhongyi Yang[1,2⋆], Anastasia Tkach[2], Shreyas Hampali[2], Linguang Zhang[2], Elliot J. Crowley[1], and Cem Keskin[2]

[1] University of Edinburgh
[2] Meta Reality Labs

Abstract. We present EgoPoseFormer, a simple yet effective transformer-based model for stereo egocentric human pose estimation. The main challenge in egocentric pose estimation is overcoming joint invisibility, which is caused by self-occlusion or a limited field of view (FOV) of head-mounted cameras. Our approach overcomes this challenge by incorporating a two-stage pose estimation paradigm: in the first stage, our model leverages the global information to estimate each joint's coarse location, then in the second stage, it employs a DETR style transformer to refine the coarse locations by exploiting fine-grained stereo visual features. In addition, we present a Deformable Stereo Attention operation to enable our transformer to effectively process multi-view features, which enables it to accurately localize each joint in the 3D world. We evaluate our method on the stereo UnrealEgo dataset and show it significantly outperforms previous approaches while being computationally efficient: it improves MPJPE by 27.4mm (45% improvement) with only 7.9% model parameters and 13.1% FLOPs compared to the state-of-the-art. Surprisingly, with proper training settings, we find that even our first-stage pose proposal network can achieve superior performance compared to previous arts. We also show that our method can be seamlessly extended to monocular settings, which achieves state-of-the-art performance on the SceneEgo dataset, improving MPJPE by 25.5mm (21% improvement) compared to the best existing method with only 60.7% model parameters and 36.4% FLOPs. Code is available at https://github.com/ChenhongyiYang/egoposeformer.

1 Introduction

In the rapid expansion of Virtual Reality (VR) and Augmented Reality (AR) technologies [1, 2], the capability to accurately interpret and emulate human actions becomes increasingly crucial. Central to this pursuit is the egocentric pose estimation task [3, 11, 12, 14, 22, 26, 31, 32, 35–37, 41, 45], which aims to estimate the 3D body pose from a vantage point inherent to the user, predominantly from head-mounted cameras. Its precision is pivotal in a wide range of applications, such as gaming and virtual meetings, making it essential for crafting an immersive user experience for the next generation of VR/AR systems.

⋆ Work done when working as a research scientist intern at Meta Reality Labs

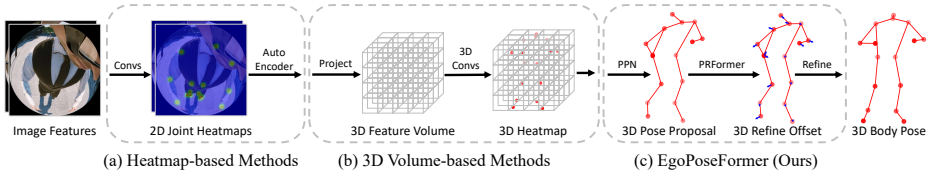


Fig. 1: Illustration of different egocentric pose estimation methods. While previous approaches predict joints' locations via 2D heatmaps or 3D feature voxels, EgoPoseFormer first estimates the coarse locations of each joint using a Pose Proposal Network (PPN) and uses a transformer to refine the estimated pose.

Different from outside-in pose estimation, where the human bodies are well covered by the images, a key challenge of egocentric pose estimation is the joint invisibility problem, which usually results from two causes. First, the limited field of view (FOV) of head-mounted cameras cannot fully capture the human body [3, 32], especially when hands and legs are stretched out. Another cause arises from the self-occlusion of different body parts [32], especially the lower body, which is very prone to be occluded by the main trunk. To overcome this limitation, some recent works [3, 32, 45] directly regress the 3D joint locations using the 2D heatmaps by employing an auto-encoder style architecture, which allows the locations of invisible joints to be inferred from the global information and other visible joints' locations. These approaches, however, take 2D heatmaps as input and cannot leverage the rich appearance information of the input image, which limits its 3D regression capability. This also causes a poor scaling-up ability to the developed model, i.e., even if the model is equipped with a larger backbone network, its pose estimation accuracy cannot be improved [3]. In another recent work, SceneEgo [37], a 3D feature voxel grid was first built using fish-eye projection [28] with the help of depth and semantic maps, and 3D convolution was employed to operate on the feature grid to estimate each joint's location in a monocular setting. Despite the fact that 3D joints outside the FOV can be estimated by leveraging a sufficiently large voxel grid, 3D convolutions are computationally expensive and the cost increases with the size of the voxel grid.

In this work, we propose *EgoPoseFormer*, a simple transformer-based model for multi-view egocentric pose estimation. As illustrated in Fig. 1, our method uses a two-stage framework [25] to overcome the joint invisibility challenge, and the body pose is predicted in a coarse-to-fine manner [24, 38, 39]. Specifically, the first stage of our model is the Pose Proposal Network (PPN), which is a simple 2-layer MLP that leverages the global information of the input multi-view feature maps to predict a joint's coarse location. Similar to the recent heatmap-based methods [3, 32], the usage of global features allows our method to reason about the locations of all joints, including the invisible ones. Surprisingly, we found, with proper training settings, this simple MLP can already outperform previous state-of-the-art methods in stereo inputs are available. Then, in the second stage, we employed a DETR-style [4] transformer, Pose Refinement Transformer (PRFormer), to predict 3D refinement offsets related to the first stage

estimations by exploiting the multi-view stereo features and human kinematic information. Specifically, we embed each joint's location and identity information into a Joint Query Token (JQT). Each JQT interacts with the multi-view image features and other JQTs through attention operations in each layer of PRFormer; subsequently, the refinement offsets are predicted from updated JQTs. Furthermore, we design a new Deformable Stereo Attention [48] to effectively process the fine-grained multi-view stereo features, which allows us to accurately estimate a joint's 3D location. In summary, we make three contributions:

- We propose EgoPoseFormer, a simple transformer-based model for stereo egocentric pose estimation. Our model composes an MLP-based pose proposal network for computing coarse joint locations, which already demonstrates a strong accuracy, and a transformer-based pose refinement network to further improve the localization accuracy.
- Our method achieves state-of-the-art on the stereo UnrealEgo dataset [3] by a huge advantage over previous arts with much lower computation costs.
- We demonstrate that our method can be easily extended to the monocular egocentric pose estimation problem and achieve state-of-the-art performance on the SceneEgo dataset [37].

2 Related Work

Egocentric Pose Estimation. Prior to our work, there have been several works on egocentric pose estimation, which cover both monocular and stereo settings. Most previous approaches are based on predicting 3D joints locations from 2D heatmaps. For example, Mo^{2}Cap^2 [41] first predicts the 2D joint heatmaps and their corresponding depth, and the 3D coordinates are computed with fish-eye unprojection. In xR-EgoPose [32], the 3D joint coordinates are directly estimated with an auto-encoder, whose input is the predicted joint heatmap. This allows it to tackle the joint invisibility difficulty. SelfPose [31] improves xR-EgoPose by introducing joint rotation loss and UNet [27] backbone. EgoSTAN [22] improves the quality of visual features by introducing temporal modeling. EgoGlass [45] extends the monocular heatmap-based methods to multi-view settings, where the joints' locations are estimated from the multi-view joint heatmaps. It also adds an auxiliary segmentation loss to improve accuracy further. EgoPW [36] explores extending existing pose estimation methods to estimate body poses in a global space. UnrealEgo [3] further improves EgoGlass by introducing cross-view information exchange in the UNet decoder. The recently proposed Ego3DPose [12] improves UnrealEgo by explicitly modeling limb heatmaps and orientations. Apart from heatmap-based approaches, SceneEgo [37] was recently introduced to directly predict joint locations by running 3D convolution on 3D feature voxels with the help of scene depth and segmentation. There are also works [14, 21, 43] about egocentric pose hallucination, where the headset wearer's body pose is estimated with front-facing cameras, in which the body is rarely observed. Different from ours, the focus of those works is generating body poses that are harmonious with

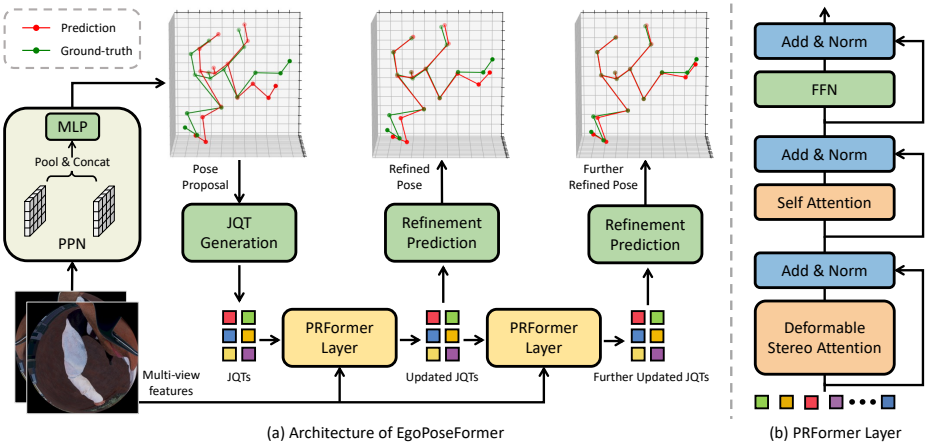


Fig. 2: (a) An overview of the proposed EgoPoseFormer. The input of EgoPoseFormer is the multi-view image features. In the first stage: Pose Proposal Network (PPN), the multi-view features are globally pooled and concatenated, from which an MLP is used to estimate the coarse location of each joint (pose proposal). Then the joints' identity and location information are embedded into Joint Query Tokens (JQTs) to serve as the queries in the second stage Pose Refinement Transformer (PRFormer). In PRFormer, a JQT iteratively interacts with the stereo features and other JQTs to update itself through the attention mechanism. The updated JQTs are used to predict refinement offsets related to the pose proposal, yielding more accurate pose estimations. **(b)** The architecture of PRFormer layer is similar to the transformer decoder layer, which includes a cross-attention block, a self-attention block, and a feed-forward network (FFN). However, in PRFormer, the cross-attention is replaced by the proposed Deformable Stereo Attention to better exploit stereo visual features.

the background scene.

Transformer for Outside-in Pose Estimation. There have been lots of successful attempts to apply transformers for the outside-in body pose estimation. One line of work [8, 16, 29, 44, 46, 47] aims to develop high-performing transformer-based backbone networks for outside-in pose estimation. For example, PoseFormer [47] introduces spatial and temporal attention mechanisms to generate high-quality visual features for 3D pose estimation, which is improved by the followed-up PoseFormer v2 [46] by introducing frequency modelling. Furthermore, improvements in model efficiency are also elaborated in [8, 15]. The other line of work, similar to ours, focuses on developing DETR-style [6, 18, 23, 30, 40, 42] sparse transformers for human body pose estimation. For example, PETR [30] introduces inter-instance and intra-instance attention for accurate 2D multi-person pose estimation. PSVT [23] introduces spatial-temporal encoder and decoder for 3D pose and body shape estimation. On the other hand, transformer architecture is also used for 2D human pose estimation [19, 40]. For example, GroupPose [19] uses keypoint and instance queries to directly estimate the 2D human poses in

a multi-person setting. Despite the huge success of transformer architectures in outside-in pose estimation, applying it to egocentric settings requires non-trivial adaptation due of the intrinsic difference between the two problems. For instance, in outside-in pose estimation, the human body usually lies within the camera's FOV, while the out-of-FOV problems usually happen in egocentric pose estimation. Another difference is the input of those two tasks: most outside-in pose estimation models take regular images captured by pin-hole cameras as input [10, 17], while in egocentric settings the images are usually captured by fish-eye cameras [3, 37, 45] to expand FOV, causing image distortions and posing further difficulties to the task.

3 Method

In this section, we introduce the proposed EgoPoseFormer in detail. We first discuss the motivation of our two-stage framework in Sec. 3.1. We present the Pose Proposal Network inSec. 3.2 and the Pose Refinement Transformer in Sec. 3.3. Later, we introduce the loss function in Sec. 3.4 and the feature extractor in Sec. 3.5. An overview of our model is illustrated in Fig. 2 (a).

3.1 Two-stage Pose Estimator

Fig. 2 (a) shows the proposed two-stage framework, which is designed to overcome the joint invisibility challenge caused by self-occlusion or the limited FOV of head-mounted cameras. In the first stage, we estimate the coarse location of each joint, which we call *pose proposal* [25], by utilizing the global feature pooled from the stereo features. This design enables the network to roughly localize all joints, including the invisible ones, by jointly reasoning visual clues from visible joints and background scenes. The global feature, however, does not preserve fine-grained local details that enable more accurate joint localization. Motivated by this observation, we added a second stage to refine the pose proposal, where we use a transformer that exploits fine-grained stereo features and the body kinematic information through the attention mechanism. Finally, the refined poses are output as the final pose estimation results.

3.2 Pose Proposal Network (PPN)

Given the multi-view image features $\mathbf{F} \in \mathbb{R}^{V \times H \times W \times C}$, the PPN computes the pose proposal $P^0 \in \mathbb{R}^{N_j \times 3}$ using global information. Here V denotes the number of views; H, W and C denotes the height, width and channel number of the image feature maps; N_j denotes the number of body joints. Specifically, PPN begins by applying a global average pooling to the feature maps of each view, then the resultant averaged features are concatenated to form a unified feature representation, which captures the salient global features and stereo information

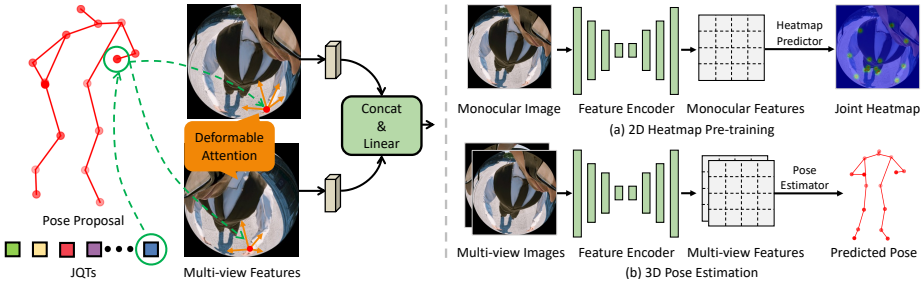


Fig. 3: Left: An illustration of our Deformable Stereo Attention. The 3D joints are first projected to each view plane using camera parameters. Within each view, we compute 2D deformable attention by querying the image features with the JQTs with the projected points serving as reference points. Finally, the attention results for each view are concatenated and fed into a linear layer to be projected into the original dimension. **Right**: (a) The feature extractor is first pre-trained to predict 2D joint heatmaps using monocular images. (b) The multi-view feature maps are computed using the pre-trained feature extractor.

across views. We then predict each joint's initial 3D location with a 2-layer MLP with GELU activation:

$$P^0 = \text{MLP}_{\text{ppn}}\Big(\text{Concat}_{\{k\}}\big(\text{AvgPool}(F_k)\big)\Big) \tag{1}$$

where $k \in \{1, ..., V\}$. As we will show in Sec. 4.3, this simple design can already provide reasonable location estimation for each joint.

3.3 Pose Refinement Transformer (PRFormer)

The PRFormer takes the multi-view image features $\mathbf{F} \in \mathbb{R}^{V \times H \times W \times C}$ and the pose proposal $P^0 \in \mathbb{R}^{N_j \times 3}$ as inputs. Structurally, the transformer comprises S layers. At each layer s, a refinement offset $\Delta P^s \in \mathbb{R}^{N_j \times 3}$ relative to P^0 is predicted, and the pose estimation is computed by adding the offset to the pose proposals $P^s = \Delta P^s + P^0$ where $s = \{1, ..., S\}$. During the inference phase, the final layer's output, P^S, is used as the model's final prediction.

Joint Query Tokens. Inspired by DETR [4], in PRFormer every joint is characterized by a unique Joint Query Token (JQT). The JQTs will serve as the queries in our transformer to interact with each other and the multi-view image features through attention mechanisms. We compute the JQTs by embedding each joint's identity and initial location information with a Query Generation MLP. Specifically, for the j-th joint, its JQT $Q_j \in \mathbb{R}^C$ is computed by feeding its initial location $P_j^0 = (x_j^0, y_j^0, z_j^0)$ and a scalar identifier σ_j into the MLP:

$$Q_j = \text{MLP}_{\text{JQT}}\big(\sigma_j, x_j^0, y_j^0, z_j^0\big) \tag{2}$$

In practice, we simply use the joint index j to serve as the scalar identifier σ_j.

PRFormer Layer. As shown in Fig. 2 (b), each layer of the PRFormer is a transformer decoder layer [34]. Each input JQT undergoes a cross-attention operation to interact with the fine-grained stereo image features, and a subsequent self-attention operation to extract spatial and human kinematic information from other JQTs. Note that here we follow [5] to put the cross-attention operation before the self-attention operation. Finally, we use a Feed-forward Network (FFN) [34] to non-linearly transform the JQTs. A distinct characteristic of the PRFormer layer vis-à-vis traditional transformer decoder layers is our replacement of the conventional cross-attention with Deformable Stereo Attention, which enables our transformer to effectively reason about multi-view stereo and thus can accurately locate the joints in the 3D world.

Deformable Stereo Attention. As shown in Fig. 3 Left, the proposed Deformable Stereo Attention has three steps. Firstly, we project each body joint's initial 3D location onto each view plane by leveraging the camera parameters [28]. Specifically, for joint j and its initial 3D location P_j^0, we compute its 2D location on each view $\{\tilde{P}_j^k\} = \{(\tilde{x}_j^k, \tilde{y}_j^k)\}$ where $k = \{1, ..., V\}$. Secondly, with the joints' projected 2D locations as reference points, we independently apply deformable attention [48] in each view to let the JQTs extract useful information from image features:

$$Z_j^k = \text{DeformAttn}\big(Q_j, \tilde{P}_j^k, F_k\big) \tag{3}$$

where Q_j is the j-th JQT and F_k is the image feature of the k-th view. Notably, if a joint is out of the view's FOV, we simply fill the computed result Z_j^k with zeros to make this information explicit to the model. Finally, the computed results $\{Z_j^k\}$ from each view are concatenated and fed into a linear projection layer to fuse multi-view stereo information and transform the result to the original dimension:

$$Z_j = \text{Linear}\big(\text{Concat}_{\{k\}}(Z_j^k)\big) \tag{4}$$

In this way, the Deformable Stereo Attention becomes an *atomic* attention operation and can replace the normal cross-attention operation in a plug-and-play manner.

Predicting Refinement Offsets. Each PRFormer layer empowers a JQT to engage with the multi-view image features and other JQTs to exploit stereo and human kinematic information, imbuing it with joint-specific knowledge. Based on this enriched representation, we use a shallow MLP to predict the refinement offset relative to the pose proposal. Specifically, the refinement offset for joint j at the s-th PRFormer layer is computed as:

$$\Delta P_j^s = \text{MLP}_{\text{offset}}^s(Q_j^s) \tag{5}$$

3.4 Loss Function

To keep our method concise, we employ a simple per-joint error loss [3] to train our model. Specifically, the loss function is formulated as the following:

$$\mathcal{L} = \sum_{s=0}^{S} \sum_{j=1}^{N_j} ||P_j^s - \hat{P}_j||_2 \tag{6}$$

Here, different loss stages are indexed with s. Specifically, $s = 0$ corresponds to the pose proposals generated by PPN, and $s > 0$ represents the subsequent refinement predictions derived from the transformer layers. N_j is the total number of joints; P_j^* and $\hat{P}_j$ are the predicted and ground-truth 3D coordinates of the j-th joint.

3.5 Feature Extractor

We follow UnrealEgo [3] to adopt a UNet [27] architecture as our visual feature extractor to compute the multi-view image features. The key difference in our method is the exclusion of multi-view concatenation in the decoder part [3]. As the PRFormer can effectively process multi-view stereo features, we can extract features from each view independently instead of aggregating multi-view features within the feature extractor. As we show in Sec. 4.2, this modification significantly reduces the model size and improves its computational efficiency.

Inspired by the heatmap-based methods [3,32], we further enhance our model's efficacy by pre-training its feature extractor to predict 2D joint heatmaps. As shown in Fig. 3 right, during the pre-training phase, the model predicts 2D joint heatmaps for each view using a light-weighted fully convolutional head [3], which is removed after pre-training.

4 Experiments

4.1 Experiment Settings

Dataset Settings. We use the multi-view UnrealEgo [3] dataset to benchmark our proposed method. The UnrealEgo dataset has 451k synthetic stereo views collected using 30 different actions, which are captured by two head-mounted cameras placed 1cm from the head. We follow the official dataset splits: the model is trained using the training set (357k views) and evaluated using the test set (48k views); The validation set (46k views) is used for tuning the hyperparameters. We also test our method using the monocular SceneEgo [37] dataset to test its generalization ability to monocular settings. SceneEgo is a real-human dataset recorded by two actors with different daily actions. It has a total 28k images. For both datsets, we report the Mean Per Joint Position Error (MPJPE) and Procrustes Analysis MPJPE (PA-MPJPE) in millimeters as evaluation metrics following their official papers [3, 37].

Table 1: Comparison of MPJPE (PA-MPJPE) between our method and previous state-of-the-art approaches on the UnrealEgo dataset.

Method	Jump	Fall	Exercise	Pull	Sing	Roll	Crawl	Lay	Sitting	Crouch-N	
xr-EgoPose [32]	106.3 (-)	167.2 (-)	133.2 (-)	119.5 (-)	99.6 (-)	116.1 (-)	223.5 (-)	146.7 (-)	274.9 (-)	172.2 (-)	
EgoGlass [45]	81.3(63.4)	131.6(94.7)	100.2(74.1)	81.9(62.0)	70.6(52.6)	103.3(92.7)	182.4(113.6)	109.6(81.7)	207.3(114.7)	132.7(110.8)	
UnrealEgo [3]	76.6(61.1)	126.8(95.9)	90.3(68.4)	78.2(61.4)	67.3(49.8)	86.2(71.6)	181.5(116.4)	97.8(76.6)	194.3(150.3)	116.8(97.9)	
Ego3DPose [12]	60.8(49.4)	95.8(79.1)	76.1(63.3)	58.1(45.7)	51.7(41.0)	81.9(71.1)	148.7(105.1)	83.4(69.8)	153.8(133.3)	93.6(79.6)	
Ours	**34.9(35.2)**	**71.0(63.9)**	**40.0(39.6)**	**24.9(24.1)**	**27.2(25.8)**	**42.4(45.3)**	**99.5(77.4)**	**61.7(59.7)**	**98.4(99.3)**	**54.1(52.9)**	
Method	Crouch-T	Crouch-TS	Crouch-F	Crouch-B	Crouch-S	Stand-WB	Stand-UB	Stand-T	Stand-TC	Stand-F	
xr-EgoPose [32]	173.8 (-)	108.9 (-)	119.9 (-)	136.5 (-)	145.8 (-)	94.3 (-)	93.3 (-)	103.3 (-)	101.6 (-)	99.7 (-)	
EgoGlass [45]	128.5(110.8)	82.3(60.5)	79.2(65.4)	94.8(77.9)	93.9(77.9)	70.2(51.5)	70.2(46.3)	77.7(59.3)	77.8(70.0)	77.3(62.3)	
UnrealEgo [3]	128.2(104.3)	76.8(55.2)	73.6(61.9)	78.2(62.0)	84.9(71.4)	68.5(50.1)	66.5(45.4)	74.3(57.9)	74.1(61.2)	70.8(56.6)	
Ego3DPose [12]	109.0(89.9)	65.3(48.7)	53.9(45.9)	58.1(46.9)	67.7(56.7)	50.4(39.9)	50.5(37.2)	58.5(48.1)	59.6(52.8)	55.7(48.7)	
Ours	**78.4(69.3)**	**27.9(26.1)**	**23.5(25.3)**	**28.9(29.6)**	**33.1(36.5)**	**26.2(25.3)**	**25.6(24.3)**	**30.4(32.8)**	**22.2(23.2)**	**33.8(37.2)**	
Method	Stand-B	Stand-S	Dance	Boxing	Wrestling	Soccer	Baseball	Basketball	Gridiron	Golf	All
xr-EgoPose [32]	105.8 (-)	114.3 (-)	116.7 (-)	97.3 (-)	116.6 (-)	104.6 (-)	103.7 (-)	98.6 (-)	149.7 (-)	117.5 (-)	112.6 (-)
EgoGlass [45]	75.1(58.5)	85.2(67.8)	83.7(65.5)	71.3(54.2)	86.1(65.3)	52.3(58.2)	77.7(61.0)	59.7(47.3)	102.1(80.5)	69.4(48.2)	83.3(61.6)
UnrealEgo [3]	68.1(55.1)	78.9(62.3)	79.9(63.2)	68.7(51.4)	83.7(64.5)	78.3(56.2)	72.8(56.7)	60.5(45.4)	99.5(81.8)	73.8(49.0)	79.1(59.2)
Ego3DPose [12]	51.9(42.6)	60.9(51.2)	61.3(50.8)	49.9(40.2)	65.5(51.7)	54.4(43.7)	69.2(55.6)	48.4(35.8)	83.3(72.9)	54.5(38.1)	60.8(48.5)
Ours	**26.8(28.5)**	**32.2(34.5)**	**35.0(36.2)**	**22.9(22.6)**	**37.0(36.6)**	**27.0(27.4)**	**41.6(42.9)**	**16.5(16.1)**	**58.4(48.7)**	**26.9(21.8)**	**33.4(32.7)**

Table 2: Comparison of MPJPE and PA-MPJPE with other methods on the monocular SceneEgo dataset.

Method	MPJPE	PA-MPJPE
Mo^2Cap^2 [41]	200.3	121.2
xR-egopose [32]	241.3	133.9
EgoPW [36]	189.6	105.3
SceneEgo [37]	118.5	92.7
Ours	**93.0**	**74.3**

Table 3: Comparison of model efficiency between EgoPoseFormer and other methods on UnrealEgo and SceneEgo datasets.

Dataset	Methods	MPJPE	PA-MPJPE	Params	FLOPs
UnrealEgo	EgoGlass	83.3	61.6	107.3 M	16.1 G
	UnrealEgo	79.1	59.2	106.8 M	27.1 G
	Ego3DPose	60.8	48.5	178.4 M	55.6 G
	Ours	**33.4**	**32.7**	**14.1** M	**7.3** G
SceneEgo	SceneEgo	118.5	92.7	45.9 M	157.3 G
	Ours	**93.0**	**74.3**	**27.9** M	**50.4** G

Model and Training Settings. Following their original papers [3, 37], we use ResNet-18 [9] as the backbone network for UnrealEgo and ResNet-50 for SceneEgo. Following [3], we set the visual feature's down-sampling stride to 4. We employ three layers of transformer decoder in our PRFormer. We use very similar training recipes for both datasets without careful tuning. Specifically, for both 2D joint heatmap pre-training and 3D pose estimation, we set the batch size to 16 and trained the model for 12 epochs. AdamW [20] is used as the default optimizer. The initial learning rate is set to 0.001 and weight decay is set to 0.005. We set the gradient clip to 5.0. The learning rates are decayed by a factor of 10 after the 8th and 11th epochs.

4.2 Main Results

Comparison with the state-of-the-art. In Tab. 1, we compare our EgoPoseFormer with previous state-of-the-art methods on the UnrealEgo [3] dataset. We report both average and action-specific metrics. The results of competing methods are taken from their corresponding papers [3, 12]. As shown by the results, our approach achieves a significant advantage over all competing methods across all actions. Specifically, our method is 27.4mm better in the averaged MPJPE than the previous state-of-the-art Ego3DPose [12], pushing the best performance on

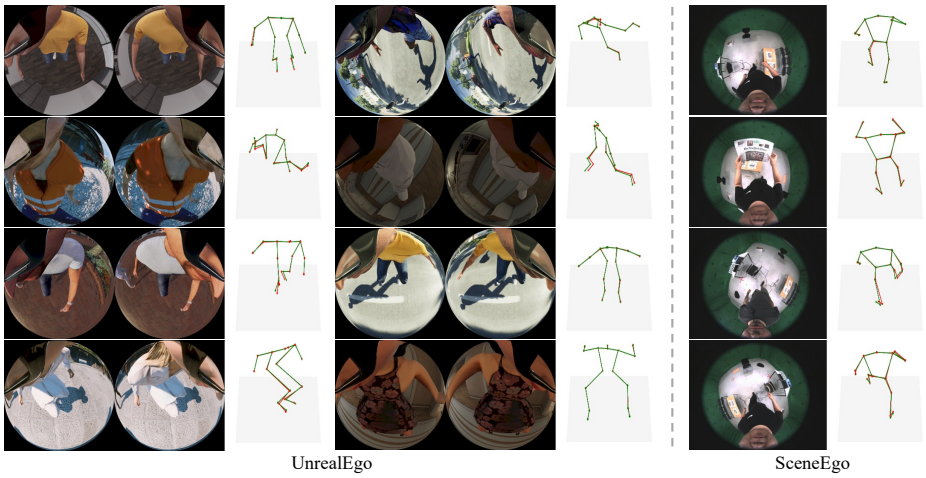


Fig. 4: Qualitative visualization on UnrealEgo and SceneEgo. Ground-truths are colored in green and predictions are colored in red.

this benchmark by 45% lower error. In Tab. 2, we compare our method with previous egocentric body pose estimation approaches on the monocular SceneEgo [37] dataset. Our method achieved an MPJPE that is 25.5mm lower than the previous best method. Note that SceneEgo [37] requires ground-truth depth and semantic masks to train their model, but our method does not require them. Also, the previous best model was pre-trained on a large pose estimation dataset and fine-tuned on the SceneEgo dataset [37], but we only used the SceneEgo dataset itself to train our model. In Fig. 4, we show qualitative results produce by our method on both datasets, which validates that EgoPoseFormer can generalize well to challenging poses. In summary, these results validate the effectiveness of our method on both multi-view and monocular egocentric pose estimation.

Computation efficiency. In Tab. 3, we compare the computational efficiency of the proposed EgoPoseFormer with other competing models on the UnrealEgo and SceneEgo datasets, where we report the model parameters together with the number of FLOPs. The results show that our method is highly effective yet efficient. Specifically, on the UnrealEgo dataset, our method decreases the error over the previous best method by 45% with only 7.9% of the parameters and 13.1% of the FLOPs. Similarly, on the SceneEgo dataset, our method reduces the MPJPE by 25.5mm compared to the previous best method, but requires only 60.7% of the parameters and 36.4% of the FLOPs.

4.3 Ablation Studies and Discussions

PRFormer. We first conduct experiments on the UnrealEgo dataset to study the two attention modules in our PRFormer. We present the results in Tab. 4.

Table 4: Ablation Study of attention operations on the UnrealEgo dataset. (DS-Attn denotes our Deformable Stereo Attention.)

DS Attn	Self Attn	First Stage MPJPE	First Stage PA-MPJPE	Second Stage MPJPE	Second Stage PA-MPJPE
		46.7	39.0	-	-
	✓	46.9	38.6	45.9	38.0
✓		46.5	39.0	38.1	36.2
✓	✓	45.5	38.3	33.4	32.7

Table 5: Ablation study of the stereo and monocular settings on the UnrealEgo dataset. Note here the feature extractors are also pre-trained using the corresponding settings.

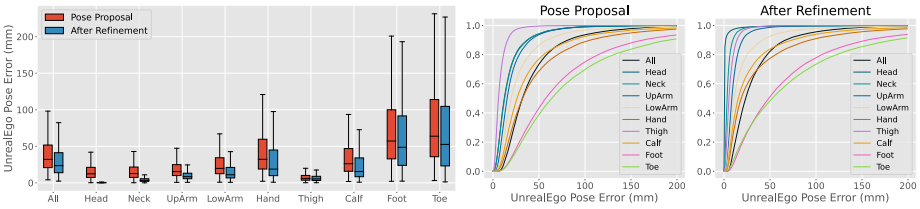
Model Input	First Stage MPJPE	First Stage PA-MPJPE	Second Stage MPJPE	Second Stage PA-MPJPE
Monocular Left	56.0	49.1	48.3	47.2
Monocular Right	56.3	49.0	48.7	47.3
Monocular Left + Right	55.6	48.8	48.2	47.2
Stereo	45.5	38.3	33.4	32.7

Table 6: Error of different body parts when they are captured by different numbers of views on UnrealEgo. We report our reproduced result for the UnrealEgo model.

Method	Views	Head	Neck	Upper Arm	Lower Arm	Hand	Thigh	Calf	Foot	Toe	MPJPE
UnrealEgo	0	-	-	-	114.7	139.2	21.5	236.2	282.3	319.1	80.1
	1	-	-	-	104.8	119.3	39.5	241.9	234.1	292.7	
	2	49.1	45.4	50.0	74.3	98.3	16.3	85.7	122.1	139.6	
Ours	0	-	-	-	55.4	111.8	19.1	147.2	178.4	179.3	33.4
	1	-	-	-	47.4	83.8	32.6	74.3	119.9	170.3	
	2	1.73	5.3	11.4	18.6	31.7	7.2	31.0	71.8	81.5	

We start with a baseline that only contains the PPN with the PRFormer part being removed. Surprisingly, this simple baseline achieves a 14.1mm MPJPE advantage over the previous state-of-the-art. We speculate such an advantage is achieved by the usage of visual features. We then added PRFormer but with only the self-attention operation, from which the refined pose estimation is only marginally better than the pose proposal. Subsequently, we add another version of PRFormer that only contains Deformable Stereo Attention, this time the refined pose estimation improves MPJPE by 8.4mm over the initial pose proposal, validating its effectiveness in exploiting multi-view stereo features. Lastly, we add the full version of PRFormer with both types of attention, from which the refined pose estimation is 12.1mm more accurate than the initial pose proposal. We therefore conclude that it is essential to combine the Deformable Stereo Attention with self-attention, in which the JQTs interact with each other to propagate human kinematic information. To further understand how PRFormer refines the pose proposal, in Fig. 5 we report the MPJPE improvement w.r.t. different body parts. Finally, we provide qualitative examples to better understand how the attention mechanisms work: In Fig. 6, we visualize the averaged attention weights between different joints in the self-attention operation on both datasets, from which we can see even if we do not explicitly introduce any kinematic-related losses in our model, the attention can implicitly infer the kinematic structures of a human body.

Monocular v.s. Multi-view. We design Deformable Stereo Attention to effectively infer the multi-view stereo information, which is crucial to accurate



(a) Error Analysis by Different Body Parts (b) Error CDFs of Different Body Parts

Fig. 5: Error analysis of the pose proposal and the refined body pose estimation.

3D localization. Here, we examine if the performance improvement is brought by the transformer model itself, or is a result of the effective stereo reasoning. To this end, we test our model on the UnrealEgo dataset in both monocular and stereo settings. The results are reported in Tab. 5. For the monocular setting, we treat every image as an independent sample. We tried three monocular settings: training and testing the model with the left view only, the right view only, and a mixing of both views. We got similar numbers for all three settings. The results show using monocular images decreased the performance of both PPN and PRFormer. Specifically, the monocular PPNs increases MPJPE by 10mm compared to the stereo variant. In addition, the monocular PRFormers can only improve the MPJPE of pose proposal by 7.5mm, but in the multi-view setting PRFormer can improve the MPJPE of the pose proposal by 12mm. From the results, we conclude: *1) Multi-view input is important to the performance of both PPN and PRFormer. 2) While PRFormer can refine the pose proposal with monocular inputs, it becomes more effective with multi-view inputs.* To further validate the conclusions, in Tab. 6 we presents error metrics for different body parts when they are captured by varying numbers of views. We also include results for the heatmap-based UnrealEgo [3] for comparasion. From the findings, a clear correlation emerges between the localization accuracy of a joint and the number of views in which it is visible. When a joint is concurrently visible in both views, our method consistently achieves highly accurate localization.

Error Analysis. We first study the error distributions of our method. We report the error distribution of different body parts in Fig. 5 (a), encompassing both pre-refinement and post-refinement results. Our observations reveal that, for the majority of joints, our PRFormer exhibits a substantial capacity to reduce the errors in the pose proposals. Notably, except for the challenging *Foot* and *Toe*, the median error in refined poses even falls below the quartile error of the pose proposals, validating the effectiveness of our proposed PRFormer in enhancing the overall accuracy of the pose estimation. On the other hand, it is worth noting that the *Foot* and *Toe* joints, although improved in the post-refinement phase, still exhibit considerable inaccuracy, limiting the overall accuracy of the pose estimation system. Furthermore, following [12], we visualize the error distributions in the form of Cumulative Distribution Function (CDF) for different body parts. The results are shown in Fig. 5 (b), including both the pre-refinement and

Table 7: Ablation study about how the number of refinement layers in PRFormer affects the model performance on the UnrealEgo dataset.

Layers	0	1	2	3	4	5
MPJPE	46.7	37.9	35.1	33.4	33.0	32.8

Table 8: Comparison of MPJPE (PA-MPJPE) between EgoPoseFormer and the heatmap-based UnrealEgo using different backbones on the UnrealEgo dataset.

Method	ResNet-18	ResNet-34	ResNet-50	ResNet-101
UnrealEgo [3]	79.1 (59.2)	80.5 (60.1)	80.1 (60.1)	80.1 (60.5)
Ours	33.4 (32.7)	29.5 (29.0)	28.5 (28.1)	27.9 (27.4)

Table 9: Comparison between our approach and the baseline UnrealEgo method on the UnrealEgo dataset using different training strategies. † denotes our reproduced result.

Method	Training Strategy	MPJPE	PA-MPJPE
UnrealEgo	UnrealEgo	79.1	59.2
	UnrealEgo†	80.1	57.9
	Ours	70.5	55.3
Ours	Ours	33.4	32.7

Table 10: Detailed ablation study on training strategies of PPN on the UnrealEgo dataset. Note the PPN performance is a bit better than the result in Tab. 4 because we used a newly pre-trained backbone.

Feature Extractor	OPT	ACT	# Layer	Heatmap Pretrain	MPJPE	PA-MPJPE
UnrealEgo	AdamW	GELU	2	✗	48.3	40.4
Ours	Adam	Leaky	3	✗	168.9	125.5
	AdamW	Leaky	3	✗	59.6	48.5
	AdamW	Leaky	3	✓	43.5	36.5
	AdamW	GELU	3	✓	42.6	36.1
	AdamW	GELU	2	✓	43.1	36.3

post-refinement CDFs. Similar to the earlier observations, the accuracy of pose estimation is substantially improved after refinement. For example, for the *Hand* joint, 68% of the proposals have an error less than 50mm, and this number rises to 77% for refined estimations under the same threshold.

PRFormer layers. In Tab. 7, we report EgoPoseFormer's performance with different numbers of layers of PRFormer, in which the JQTs iteratively interact with each other and the image features to refine the pose proposal. The results show that more layers of PRFormer can lead to better accuracy of pose estimation. However, when the number of layers is greater than 3, the additional layers can only bring marginal improvement. For example, increasing layers from three to four only brings 0.4mm MPJPE improvement. Therefore, we adopt a 3-layer PRFormer in our model.

Scaling up experiments. In Tab. 8, we show our approach has a good scaling-up ability when using larger backbones. Specifically, we test our methods using different ResNet [9] variants on the UnrealEgo dataset. We keep the training recipe for all backbones the same, as described in Sec. 4.1. We observe that the performance of our method gets consistently improves when we use larger backbones, which shows that our method scales well. In addition, we also report the performance of the previous heatmap-based UnrealEgo [3] method, which exhibits poor scaling-up ability when equipped with larger backbones. As we explained in Sec. 1, this may be caused by the idling of visual features.

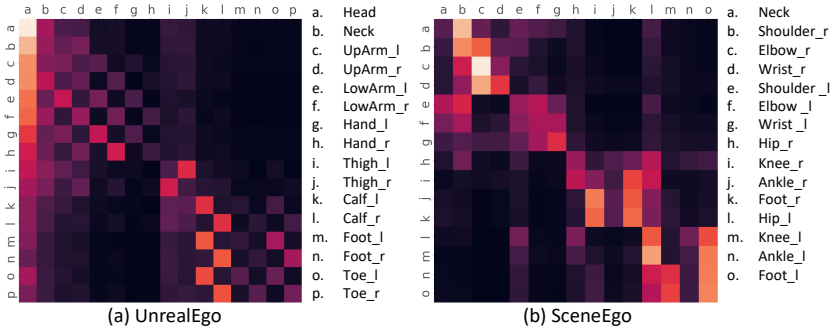


Fig. 6: Heatmaps of averaged self-attention weights between different joints on the UnrealEgo and SceneEgo datasets.

Training Strategy. One surprising result of our paper is that even our Pose Proposal Network excels the prior state-of-the-art method. Therefore, we raise a pertinent question: *Is the superiority of EgoPoseFormer a consequence of a better training strategy instead of the model itself?* Notably, the most distinct difference in our training strategy, as compared to previous approaches [3, 12], is that we followed the common practices [4, 7, 33] of transformer training to use the advanced AdamW [20] optimizer. In contrast, prior works employed the Adam optimizer [13]. To investigate this, we present the results of the UnrealEgo [3] baseline method using different training strategies in Tab. 9. The outcomes clearly demonstrate that switching to our training strategy can indeed improve the MPJPE of the baseline method by around 9mm. Nevertheless, it is noteworthy that even with this improved baseline, our EgoPoseFormer still significantly outperforms it, validating the effectiveness of our method. In Tab. 10, we report extensive ablation studies about how the training techniques affect the performance PPN when the PRFormer part is removed. Here we investigate several factors: 1) the feature extractor architecture, 2) optimizer, 3) activation function, 4) number of MLP layers, 5) whether the feature extractors are pre-trained. The result shows the AdamW optimizer and the heatmap pre-training are the two most important reasons for PPN's good performance.

5 Conclusion

We introduce EgoPoseFormer, a new transformer-based egocentric pose estimation method. Our two-stage method first infers each joint's coarse from the global features, then uses a DETR-style transformer Pose Refinement Transformer to refine the coarse locations by exploiting fine-grained stereo features and human kinematic information. Furthermore, we design Deformable Stereo Attention to better exploit the multi-view stereo information. Our method achieves state-of-the-art with significant advantages over previous arts on two pose estimation datasets, including stereo and monocular settings. We hope our model can serve as a strong baseline approach for future research in this field.

References

1. Apple Vision Pro. https://www.apple.com/apple-vision-pro/
2. Meta Quest 3. https://www.meta.com/quest/quest-3/
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 (ECCV) (2022)
4. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-End Object Detection with Transformers. ECCV (2020)
5. Cheng, B., Misra, I., Schwing, A.G., Kirillov, A., Girdhar, R.: Masked-attention mask transformer for universal image segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 1290–1299 (June 2022)
6. Choi, H., Moon, G., Park, J., Lee, K.M.: Learning to estimate robust 3d human mesh from in-the-wild crowded scenes. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1475–1484 (2022)
7. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. In: ICLR (2021)
8. Einfalt, M., Ludwig, K., Lienhart, R.: Uplift and upsample: Efficient 3d human pose estimation with uplifting transformers. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 2903–2913 (2023)
9. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR (2016)
10. Ionescu, C., Papava, D., Olaru, V., Sminchisescu, C.: Human3. 6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. IEEE transactions on pattern analysis and machine intelligence **36**(7), 1325–1339 (2013)
11. 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)
12. Kang, T., Lee, K., Zhang, J., Lee, Y.: Ego3dpose: Capturing 3d cues from binocular egocentric views. In: SIGGRAPH Asia 2023 Conference Papers (2023)
13. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
14. 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)
15. Li, W., Liu, H., Ding, R., Liu, M., Wang, P., Yang, W.: Exploiting temporal contexts with strided transformer for 3d human pose estimation. IEEE Transactions on Multimedia **25**, 1282–1293 (2022)
16. Li, W., Liu, H., Tang, H., Wang, P., Van Gool, L.: Mhformer: Multi-hypothesis transformer for 3d human pose estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13147–13156 (2022)
17. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: ECCV (2014)
18. Liu, H., Chen, Q., Tan, Z., Liu, J.J., Wang, J., Su, X., Li, X., Yao, K., Han, J., Ding, E., et al.: Group pose: A simple baseline for end-to-end multi-person pose estimation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 15029–15038 (2023)

19. Liu, H., Chen, Q., Tan, Z., Liu, J., Wang, J., Su, X., Li, X., Yao, K., Han, J., Ding, E., Zhao, Y., Wang, J.: Group pose: A simple baseline for end-to-end multi-person pose estimation. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV) (2023)
20. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. In: International Conference on Learning Representations (2019), https://openreview.net/forum?id=Bkg6RiCqY7
21. Luo, Z., Hachiuma, R., Yuan, Y., Kitani, K.: Dynamics-regulated kinematic policy for egocentric pose estimation. Advances in Neural Information Processing Systems **34**, 25019–25032 (2021)
22. Park, J., Kaai, K., Hossain, S., Sumi, N., Rambhatla, S., Fieguth, P.: Building spatio-temporal transformers for egocentric 3d pose estimation. arXiv preprint arXiv:2206.04785 (2022)
23. Qiu, Z., Yang, Q., Wang, J., Feng, H., Han, J., Ding, E., Xu, C., Fu, D., Wang, J.: Psvt: End-to-end multi-person 3d pose and shape estimation with progressive video transformers. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 21254–21263 (2023)
24. Ren, P., Wen, C., Zheng, X., Xue, Z., Sun, H., Qi, Q., Wang, J., Liao, J.: Decoupled iterative refinement framework for interacting hands reconstruction from a single rgb image. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 8014–8025 (2023)
25. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. In: NeurIPS (2015)
26. 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)
27. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18. pp. 234–241. Springer (2015)
28. Scaramuzza, D., Martinelli, A., Siegwart, R.: A flexible technique for accurate omnidirectional camera calibration and structure from motion. In: Fourth IEEE International Conference on Computer Vision Systems (ICVS'06). pp. 45–45. IEEE (2006)
29. Shan, W., Liu, Z., Zhang, X., Wang, S., Ma, S., Gao, W.: P-stmo: Pre-trained spatial temporal many-to-one model for 3d human pose estimation. In: European Conference on Computer Vision. pp. 461–478. Springer (2022)
30. Shi, D., Wei, X., Li, L., Ren, Y., Tan, W.: End-to-end multi-person pose estimation with transformers. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11069–11078 (2022)
31. 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 (2020)
32. Tome, D., Peluse, P., Agapito, L., Badino, H.: xr-egopose: Egocentric 3d human pose from an hmd camera. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 7728–7738 (2019)
33. Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., Jégou, H.: Training data-efficient image transformers & distillation through attention. In: International conference on machine learning. pp. 10347–10357. PMLR (2021)

34. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems **30** (2017)
35. 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)
36. 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. pp. 11500–11509 (2021)
37. 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. pp. 13031–13040 (2023)
38. Wang, L., Yan, S., Zhen, J., Liu, Y., Zhang, M., Zhang, G., Zhou, X.: Deep active contours for real-time 6-dof object tracking. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 14034–14044 (2023)
39. Wang, Y., Daniilidis, K.: Refit: Recurrent fitting network for 3d human recovery. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 14644–14654 (October 2023)
40. Xiao, Y., Su, K., Wang, X., Yu, D., Jin, L., He, M., Yuan, Z.: Querypose: Sparse multi-person pose regression via spatial-aware part-level query. Advances in Neural Information Processing Systems **35**, 12464–12477 (2022)
41. 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)
42. Yang, J., Zeng, A., Liu, S., Li, F., Zhang, R., Zhang, L.: Explicit box detection unifies end-to-end multi-person pose estimation. arXiv preprint arXiv:2302.01593 (2023)
43. Yuan, Y., Kitani, K.: 3d ego-pose estimation via imitation learning. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 735–750 (2018)
44. Zhang, J., Tu, Z., Yang, J., Chen, Y., Yuan, J.: Mixste: Seq2seq mixed spatio-temporal encoder for 3d human pose estimation in video. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 13232–13242 (2022)
45. 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)
46. Zhao, Q., Zheng, C., Liu, M., Wang, P., Chen, C.: Poseformerv2: Exploring frequency domain for efficient and robust 3d human pose estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8877–8886 (2023)
47. Zheng, C., Zhu, S., Mendieta, M., Yang, T., Chen, C., Ding, Z.: 3d human pose estimation with spatial and temporal transformers. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 11656–11665 (2021)
48. Zhu, X., Su, W., Lu, L., Li, B., Wang, X., Dai, J.: Deformable {detr}: Deformable transformers for end-to-end object detection. In: International Conference on Learning Representations (2021), https://openreview.net/forum?id=gZ9hCDWe6ke