Egocentric Whole-Body Motion Capture with FisheyeViT and Diffusion-Based Motion Refinement

Supplementary Material

7. Full Comparison with Existing Egocentric Pose Estimation Methods

The comparison results between our method and all previous methods [30, 36, 47, 50–52, 54] are shown in Tab. 5 and **??**. "*" indicates that the methods are re-trained with our EgoWholeBody training dataset. In this experiment, since the GlobalEgoMocap [50] can be applied to refine the egocentric human body motion predicted from any egocentric pose estimation method, we base the method on Mo^2Cap^2 [54] following the original setting in GlobalEgoMocap [50]. We also do not show the GlobalEgoMocap results in Mo^2Cap^2 test dataset [54] since it does not provide egocentric camera poses for all of the sequences. Note that our EgoWholeBody dataset does not contain ground truth scene geometry annotations, therefore we freeze the weights of the depth estimation module in SceneEgo [52] and only train the human pose estimation part.

From the results in Tab. 5, we can show our single-frame method and our refinement method consistently outperforms all of the previous methods, even if they are trained on our new dataset, which further strengthens the claim in our experiment section (Sec. 5.2).

Method	MPJPE	PA-MPJPE
SceneEgo test dataset [52]		
Mo^2Cap^2 [54]	200.3	121.2
GlobalEgoMocap† [50]	183.0	106.2
xR-egopose [47]	241.3	133.9
EgoPW [51]	189.6	105.3
SceneEgo [52]	118.5	92.75
Mo^2Cap^2* [54]	92.20	66.01
GlobalEgoMocap*† [50]	89.35	63.03
xR-egopose* [47]	121.5	98.84
EgoPW* [51]	90.96	64.33
SceneEgo* [52]	89.06	70.10
Ours-Single	<u>64.19</u>	<u>50.06</u>
Ours-Refined†	**57.59**	**46.55**
Method	PA-MPJPE	BA-MPJPE
GlobalEgoMocap test dataset [50]		
Mo^2Cap^2 [54]	102.3	74.46
xR-egopose [47]	112.0	87.20
GlobalEgoMocap†[50]	82.06	62.07
EgoPW [51]	81.71	64.87
EgoHMR [30]	85.80	–
SceneEgo [52]	76.50	61.92
Mo^2Cap^2* [54]	78.39	63.48
GlobalEgoMocap*† [50]	75.62	61.06
xR-egopose* [47]	106.3	79.56
EgoPW* [51]	77.95	62.36
SceneEgo* [52]	76.51	61.74
Ours-Single	<u>68.59</u>	<u>55.92</u>
Ours-Refined†	**65.83**	**53.47**
Mo^2Cap^2 test dataset [54]		
Mo^2Cap^2 [54]	91.16	70.75
xR-egopose [47]	86.85	66.54
EgoPW [51]	83.17	64.33
Ego-STAN† [36]	102.4	–
SceneEgo [52]	79.65	62.82
Mo^2Cap^2* [54]	79.76	63.53
xR-egopose* [47]	84.92	65.39
EgoPW* [51]	78.01	62.37
SceneEgo* [52]	79.32	62.77
Ours-Single	<u>74.66</u>	<u>59.26</u>
Ours-Refined†	**72.63**	**57.12**

Table 4. Performance of our method on three different test datasets. Our method outperforms all previous state-of-the-art methods. $*$ denotes the method trained with the datasets in Sec. 5.1. † denotes the temporal-based methods.

8. Fisheye Camera Model

In this section, we describe the projection and re-projection function of Scaramuzza's fisheye camera model [41] as follows:

The projection function $\mathcal{P}(x, y, z)$ of a 3D point $[x, y, z]^T$ in the fisheye camera space into a 2D point $[u, v]^T$ on the fisheye image space can be written as:

$$[u, v]^T = f(\rho)\frac{[x, y]^T}{\sqrt{x^2 + y^2}} \tag{7}$$

where $\rho = \arctan(z/\sqrt{x^2 + y^2})$ and $f(\rho) = k_0 + k_1\rho + k_2\rho^2 + k_3\rho^3 + \dots$ is a polynomial obtained from camera calibration.

Given a 2D point $[u, v]^T$ on the fisheye images and the distance d between the 3D point $[x, y, z]^T$ and the camera, the position of the 3D point can be obtained with the fisheye reprojection function $\mathcal{P}^{-1}(u, v, d)$:

$$[x, y, z]^T = d\frac{[u, v, f'(\rho')]^T}{\sqrt{u^2 + v^2 + (f'(\rho'))^2}} \tag{8}$$

where $\rho' = \sqrt{u^2 + v^2}$ and $f'(\rho) = k'_0 + k'_1\rho + k'_2\rho^2 + k'_3\rho^3 + \dots$ is another polynomial obtained from camera calibration.

The calibration of the fisheye camera and more details about the fisheye camera model can be found in Scaramuzza *et al*. [41].

Method	MPJPE	PA-MPJPE
Mo^2Cap^2* [54]	89.75	74.32
GlobalEgoMocap*† [50]	86.44	66.76
xR-egopose* [47]	118.2	94.33
EgoPW* [51]	84.21	63.02
SceneEgo* [52]	87.57	69.46
Ours-Single	<u>66.28</u>	<u>43.14</u>
Ours-Refined	**60.32**	**40.35**

Table 5. Performance of our method on our EgoWholeBody test datasets. Our method outperforms all previous state-of-the-art methods. $*$ denotes the method trained with the datasets in Sec. 5.1. † denotes the temporal-based methods.

Note that a number of different fisheye camera models exist and our method does not depend on one specific fisheye camera model.

9. Implementation Details

In this section, we describe the implementation details of our methods. We use NVIDIA RTX8000 GPUs for all experiments.

9.1. FisheyeViT and Pose Regressor with Pixel-Aligned 3D Heatmap

9.1.1 Network Structure

FisheyeViT In FisheyeViT, we first undistort the image patches with the method described in Sec. 3.1.1, then put the patches into a ViT transformer. In the ViT transformer, the embedding dimension is 768, the network depth is 12, the attention head number is 12, the expansion ratio of the MLP module is 4, and the drop path rate is 0.3. The output sequence from the transformer (with a length equal to 256) is reshaped to a 2D feature map with size 16×16.

Pose Regressor with Pixel-Aligned 3D Heatmap In the pixel-aligned heatmap, we first use two deconvolutional modules to up-sample the feature map from the FisheyeViT. The first deconv module contains one deconv layer with 768 input channels and 1024 output channels, one batch normalization layer, and one ReLU activation function. The deconv layer's kernel size is 4, the stride is 2, the padding is 1, and the output padding is 0. The second deconv module contains one deconv layer with 1024 input channels and 15×64 output channels, one batch normalization layer, and one ReLU activation function. The hyper-parameters of the deconv layer in the second module are the same as that in the first one.

These deconvolutional modules converts the features from shape $(C \times N \times N) = (768 \times 16 \times 16)$ to shape $(J \times D_h \times H_h \times W_h) = (15 \times 64 \times 64 \times 64)$. Then the soft-argmax function and fisheye reprojection function are applied to get the final body pose prediction.

9.1.2 Training Details

In this section, we introduce the training of our single-frame human body pose estimation network, *i.e.* the FisheyeViT and pose regressor with pixel-aligned 3D heatmap. The ViT network in FisheyeViT is initialized with the training weight from ViTPose [55] and the pose regressor is initialized with normal distribution, whose mean is 0 and standard deviation is 1. The network is trained on the combination dataset of EgoWholeBody and EgoPW. The ratio between the EgoWholeBody and EgoPW datasets is 9:1. The network is trained for 10 epochs with a batch size of 128, a learning rate of $1e^{-4}$ with the Adam optimizer.

9.2. Hand Detection Network

As described in Sec. 3.1.3, we use our EgoWholeBody dataset for training the ViTPose network to regress the heatmap of 2D hand joints. Based on the 2D hand joint predictions, we get the center $\mathbf{C}_{lh}$, $\mathbf{C}_{rh}$, and the size d_{lh}, d_{rh} of the square hand bounding boxes. We use the ViTPose network for the simplicity of implementation. Other detection methods can also be used for training the hand detection network. Taking the left hand as an example, we use the bounding center $\mathbf{C}_{lh}$ as the image patch center in **Step 1** of FisheyeViT (Sec. 3.1.1) and use the half of the bounding box size $d_{lh}/2$ as the offset d in **Step 2**. After obtaining the projected points of bounding box center $\mathbf{P}^c_{lh}$ and the bounding box edge $\mathbf{P}^x_{lh}$ on the tangent plane $\mathbf{T}_{lh}$, we set the l in **Step 3** as two times of the Euclidean distance between $\mathbf{P}^x_{lh}$ and $\mathbf{P}^c_{lh}$. Following **Step 4**, we get the undistorted hand image crop of the left hand $\mathbf{I}_{lh}$.

The hand detection network is trained for ten epochs with a batch size of 128 and a learning rate of $1e^{-4}$ with the Adam optimizer.

9.3. Hand Pose Estimation Network

As described in Sec. 3.1.3, we train the hand-only Pose2Pose network in Hand4Whole method [34] with EgoWholeBody training dataset to regress the 3D hand pose from hand image crops. During training, we only use the ground truth 3D hand joint positions as supervision to fine-tune the Pose2Pose network that has been pretrained on the FreiHAND dataset [67]. The hand pose estimation network is fine-tuned for ten epochs with a batch size of 128 and an initial learning rate of $1e^{-5}$ with the Adam optimizer.

9.4. Diffusion-Based Motion Refinement

In Sec. 3.2, we use the transformer decoder in EDGE [48] as our diffusion denoising network. We disable the music condition in EDGE [48] by replacing the music features with a learnable feature vector that is agnostic to input. Here we describe the training details and the refinement details of our diffusion model.

9.4.1 Training Details

In this section, we describe the details of training the DDPM model [18] for learning the whole-body motion prior. Given a whole-body motion sequence with 196 frames from training datasets (Sec. 5.1) represented with joint locations of the human body (with shape 15×3) and hands (with shape 21×3), we transform all poses to the pelvis-related coordinate system and align them to make the human body poses facing forward, obtaining the aligned whole-body motion sequence $\mathbf{x}$. The motion sequence $\mathbf{x}$ is normalized and sent to the DDPM model for training. During training, we randomly sample a diffusion step $t \in \{0, 1, ..., T-1\}$, and use the diffusion forward process to generate the noisy motion $\mathbf{x}_t$. Here the T is the maximal diffusion step and we set T as 1000. We finally run the denoising network to get the original motion $\hat{\mathbf{x}}$ and compare the reconstructed human motion $\hat{\mathbf{x}}$ and the original human motion $\mathbf{x}_t$ with Eq. (4). The network is trained for thirty epochs with a batch size of 256 and an initial learning rate of $2e^{-4}$ with the Adam optimizer.

9.4.2 Refinement Details

After obtaining the trained diffusion model, we follow Sec. 3.2.2 to refine the input whole-body motion. Here we describe how to obtain the uncertainty values for each joint in the human body and hands. We smooth the 3D heatmap predictions with Gaussian smoothness. The standard deviation of the Gaussian kernel is 1. Then we get the 3D heatmap values $\mathbf{HM}$ at the predicted joint locations with the bilinear interpolation. The heatmap values $\mathbf{HM}$ are firstly normalized to range $[0, 1]$ by making the maximal value of $\mathbf{HM}$ equal to 1. The uncertainty values $\mathbf{u}$ is obtained with:

$$\mathbf{u} = 0.05 \times (1 - \mathbf{HM}) \tag{9}$$

In this case, the maximal uncertainty value is 0.05. This value is empirically defined to limit the effect of the stochastic diffusion process in motion refinement.

10. Synthetic Dataset Comparisons

Compared to other egocentric motion capture training datasets, the EgoWholeBody dataset offers several notable advantages (also see Table 6):

Larger Amount of Frames: EgoWholeBody contains a substantially larger quantity of frames, providing an extensive and diverse dataset for training.

Inclusion of Hand Poses: Unlike other datasets, EgoWholeBody includes hand motion data, making it suitable for egocentric whole-body motion capture.

High Diversity in Motions and Backgrounds: The dataset captures a wide range of human motions and diverse background settings, reflecting real-world scenarios.

Figure 6. Examples of our synthetic dataset EgoWholeMocap. The upper row shows the data rendered with Renderpeople models [3], the lower row shows the data rendered with SMPL-X models [37].

Publicly Available Models, Motions, and Backgrounds: The models, motions, and backgrounds are all publicly available. Additionally, the data generation pipeline will be made public, enabling researchers to reproduce or modify the dataset for various different tasks.

These advantages position EgoWholeBody as a valuable resource for advancing research in egocentric whole-body motion capture.

To show the quality of our synthetic dataset, we also visualize some examples of our synthetic EgoWholeMocap dataset in Fig. 6.

11. Details of Evaluation Metrics

In this section, we give a detailed explanation of the evaluation metrics used in our method. Mean Per Joint Position Error (MPJPE) is the mean of Euclidean distances for each joint in the predicted and ground truth poses.

For the Mean Per Joint Position Error with Procrustes Analysis (PA-MPJPE), we rigidly align the estimated pose to the ground truth pose with Procrustes analysis [24] and then calculate MPJPE.

We also evaluate the BA-MPJPE, i.e. the MPJPE with aligned bone length. For BA-MPJPE, we first resize the bone length of predicted poses and ground truth poses to the bone length of a standard human skeleton. Then, we calculate the PA-MPJPE between the two resulting poses.

12. Details of Evaluation Datasets

In our experiment in Sec. 5.2, we use three evaluation datasets including SceneEgo test dataset [52], GlobalEgoMocap test dataset [50] and Mo^{2}Cap^2 test dataset [54].

The SceneEgo test dataset contains around 28K frames of 2 persons performing various motions such as sitting, walking, exercising, reading a newspaper, and using a computer. This dataset provides ground truth egocentric camera pose so that we can evaluate MPJPE on it. This dataset is

Training Dataset	Motion Diversity	Frame Numbers	Motion Type	Image Quality	Annotation Type
EgoPW [51]	low	318 k	body motion	real-world	pseudo ground truth
ECHP [29]	low	75 k	body motion	real-world	pseudo ground truth
Mo^2Cap^2 [54]	middle	530 k	body motion	low	ground truth
xR-EgoPose [47]	middle	380 k	body motion	**realistic**	ground truth
EgoGTA [52]	low	320 k	body motion	low	ground truth
EgoWholeBody	**high**	**870 k**	**body + hands motion**	**realistic**	ground truth

Table 6. Comparison between different training datasets for egocentric body pose estimation.

evenly split into training and testing splits. We finetuned our method on the training split before the evaluation.

The GlobalEgoMocap test dataset [50] contains 12K frames of two people captured in the studio. The Mo^2Cap^2 test dataset [54] contains 2.7K frames of two people captured in indoor and outdoor scenes. These two datasets do not provide ground truth egocentric camera poses, thus we first rigidly align the predicted body poses and ground truth body poses and then evaluate PA-MPJPE and BA-MPJPE.

13. The Standard Deviation of Refinement Method

As described in Sec. 5.2, we generate five samples and calculate the mean and standard deviations of the MPJPE values. The results are shown in Tab. 7. From the results, we can see the standard deviations of our results are all around 0.003 mm, which is quite small. We suppose that the standard deviations of our results are small for two reasons:

First, our diffusion process is guided by the low-uncertainty joints. The low-uncertainty joints are more likely to follow the initial motion estimations $\mathbf{x}_e$ and guide the diffusion denoising process of other joints to obtain similar values.

Second, according to Eq. (9), the maximal uncertainty value is 0.05 (the actual uncertainty value can be even smaller), which means that when $k = 0.1$ in Eq. (6), the $\mathbf{w} \sim 1$ when $t = 100$ for all joints:

$$\mathbf{w} = 1/\left(1 + e^{-0.1(100-1000\times 0.05)}\right) = 0.9933 \quad (10)$$

This shows that when t is large enough, the denoising process is always initialized by the estimated motion $\mathbf{x}_e$ and the refinement starts when $t < 100$. When $t < 100$, the Gaussian noise added in Eq. (5) is relatively small. This also means that we can start from diffusion step $t = 200$ for accelerating the diffusion refinement steps.

14. Different Parameters in Weight Function

In this section, we analyze the effectiveness of parameter k in the weight function Eq. (6). We suppose that the uncertainty value of one specific joint is 0.02, then we draw the $\mathbf{w}$-t figure in Fig. 7. We can observe that when $t \to 0$, the weight $\mathbf{w}$ is still large when $k = 0.01$. In this case, the initial pose predictions $\mathbf{x}_e$ will significantly affect the final refinement result. When the $k = 1$, the weight $\mathbf{w} \sim 0$ when $t < 15$, which makes the diffusion model generate freely without any guidance of the initial joint estimations. This will make the refined motion largely deviate from the initial joint estimations. In our method, we choose a moderate $k = 0.1$, such that the diffusion refinement process can be initially guided by the whole-body pose estimations $\mathbf{x}_e$ and finally refined through the generation of diffusion denoising process.

We also show the results under different k values in Tab. 8. The results show that the accuracy of human body poses is the best when $k = 0.1$. We also observe that the standard deviations become larger when k is larger. This also demonstrates the above analysis.

Dataset	MPJPE	PA-MPJPE
SceneEgo-Body	57.59±0.003	46.55±0.003
SceneEgo-Hands	19.37±0.002	9.05±0.002
Dataset	PA-MPJPE	BA-MPJPE
GlobalEgoMocap	65.83±0.003	53.47±0.002
Mo^2Cap^2	72.63±0.003	57.12±0.003

Table 7. The mean and standard deviations of our refinement method. “SceneEgo-Body” and “SceneEgo-Hands” show the body and hand results on the SceneEgo dataset. “GlobalEgoMocap” and “Mo^2Cap^2” shows the human body results on the GlobalEgoMocap and Mo^2Cap^2 datasets.

Method	MPJPE	PA-MPJPE
k=0.01	58.41±0.001	46.92±0.001
k=0.1	**57.59±0.003**	**46.55±0.003**
k=1	59.90±0.006	48.57±0.006

Table 8. Comparison with Spherenet and Panoformer.

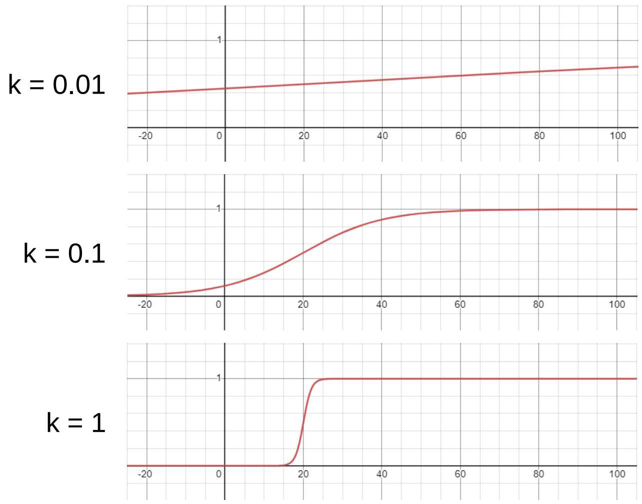

Figure 7. The weight function with different hyper-parameters k. The x-axis is the diffusion time step t and the y-axis is the weight $\mathbf{w}$.

15. Comparision with networks for panorama images

Recent studies [11, 26, 56–58] have adopted various approaches to address fisheye image distortion within deep learning frameworks. Yet, these strategies are tailored to tasks distinctly different from 3D human pose estimation, such as object detection [11] and depth estimation [26].

Nevertheless, we compare our FisheyeViT network with two other methods dealing with camera distortions, the SphereNet [11] and the OmniFusion [26]. In this experiment, we replace our FisheyeViT with the SphereNet and OmniFusion networks. In SphereNet, we limit the sampling range to the semi-sphere. In OmniFusion, we use the output of the transformer network as the image features and put the image features into our pose regressor. We evaluate the accuracy of the estimated human body pose on the SceneEgo dataset. The results are shown in Table 9, which demonstrates that our FisheyeViT performs better than the previous methods for the distorted images. This might caused by the different patch sampling strategy: our method samples the image patches on the fisheye image uv space, while previous methods samples the patches on the $r\theta\phi$ sphere coordinate system. Our method can generate patches that align well with the layout of egocentric fisheye images and match the design of our pixel-aligned 3D heatmap as mentioned in the introduction: "the voxels in the 3D heatmap directly correspond to pixels in 2D features, subsequently linking to image patches in FisheyeViT". However, sampling in the $r\theta\phi$ sphere coordinate system will cause discontinuity due to the *coordinate singularity* of the sphere coordinate system. For example, the neighboring pixels on the fisheye image can be assigned to two patches far away from each other.

Method	MPJPE	PA-MPJPE
SphereNet [11]	90.72	75.07
OmniFusion [26]	86.58	70.69
Ours-Single	**64.19**	**50.06**

Table 9. Comparison with Spherenet and Panoformer.

16. Replacing the Pixel-Aligned 3D Heatmap to MLP

In this section, we replace our pose regressor with the pixel-aligned 3D heatmap with a simple MLP network. The features extracted with FisheyeViT, with shape $(768 \times 16 \times 16)$ are firstly flattened and we further use two MLP layers to regress the 3D human body poses. The first layer contains one fully connected layer with an output dimension of 1024, one batch normalization layer, and one ReLU activation layer. The second layer contains one fully connected layer with an output dimension of 15×3. The MPJPE and the PA-MPJPE on the SceneEgo dataset are 130.7 mm and 73.91 mm respectively. This demonstrates the effectiveness of our egocentric pose regressor with pixel-aligned 3D heatmap.

17. Compare with Gaussian Smooth

In this section, we compare our diffusion-based motion refinement method with the simple Gaussian smoothness. The MPJPE and the PA-MPJPE on the SceneEgo dataset are 62.68 mm and 48.87 mm respectively. This demonstrates that our refinement method performs better than the Gaussian smooth approach. This shows that our method relies on motion priors to guide the refinement of human motion, making it more effective than the simple smoothing techniques.

18. Egocentric Camera Setup

We use the same egocentric camera setup as previous methods [50–52, 54]. In this setup, one down-facing PointGrey fisheye camera is mounted in front of the head. The illustration is shown in Fig. 8.

19. Limitations

Due to serious self-occlusion issues, our method may still predict poses suffering from physical implausibility. This can be solved by introducing the physics-aware motion diffusion models or motion refinement models, such as PhysDiff [61] and PhysCap [43].

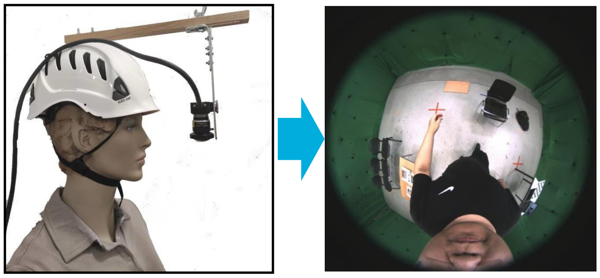

Figure 8. The setup of the egocentric fisheye camera and one example of the egocentric image.

20. More Visualization Results

Here we show more results of our methods in Fig. 9 and Fig. 10.

Acknowledgments This project was supported by the Saarbrücken Research Center for Visual Computing, Interaction and AI. Christian Theobalt was supported by ERC Consolidator Grant 4DReply (770784).

References

[1] Blender. https://www.blender.org. 6

[2] Mixamo. https://www.mixamo.com. 6

[3] Renderpeople. https://renderpeople.com. 6, 3

[4] Hiroyasu Akada, Jian Wang, Soshi Shimada, Masaki Takahashi, Christian Theobalt, and Vladislav Golyanik. Unrealego: A new dataset for robust egocentric 3d human motion capture. In *European Conference on Computer Vision*, pages 1–17. Springer, 2022. 1, 2

[5] Hiroyasu Akada, Jian Wang, Vladislav Golyanik, and Christian Theobalt. 3d human pose perception from egocentric stereo videos. In *37th IEEE/CVF Conference on Computer Vision and Pattern Recognition*. IEEE, 2024. 2

[6] Zhongang Cai, Wanqi Yin, Ailing Zeng, Chen Wei, Qingping Sun, Yanjun Wang, Hui En Pang, Haiyi Mei, Mingyuan Zhang, Lei Zhang, et al. Smpler-x: Scaling up expressive human pose and shape estimation. *arXiv preprint arXiv:2309.17448*, 2023. 3

[7] Young-Woon Cha, True Price, Zhen Wei, Xinran Lu, Nicholas Rewkowski, Rohan Chabra, Zihe Qin, Hyounghun Kim, Zhaoqi Su, Yebin Liu, et al. Towards fully mobile 3d face, body, and environment capture using only head-worn cameras. *IEEE transactions on visualization and computer graphics*, 24(11):2993–3004, 2018. 2

[8] Jeongjun Choi, Dongseok Shim, and H Jin Kim. Diffupose: Monocular 3d human pose estimation via denoising diffusion probabilistic model. *arXiv preprint arXiv:2212.02796*, 2022. 3

[9] Vasileios Choutas, Georgios Pavlakos, Timo Bolkart, Dimitrios Tzionas, and Michael J Black. Monocular expressive body regression through body-driven attention. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part X 16*, pages 20–40. Springer, 2020. 3

[10] Hai Ci, Mingdong Wu, Wentao Zhu, Xiaoxuan Ma, Hao Dong, Fangwei Zhong, and Yizhou Wang. Gfpose: Learning 3d human pose prior with gradient fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4800–4810, 2023. 3

[11] Benjamin Coors, Alexandru Paul Condurache, and Andreas Geiger. Spherenet: Learning spherical representations for detection and classification in omnidirectional images. In *Proceedings of the European conference on computer vision (ECCV)*, pages 518–533, 2018. 5

[12] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021. 6

[13] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 4, 8

[14] Yuming Du, Robin Kips, Albert Pumarola, Sebastian Starke, Ali Thabet, and Artsiom Sanakoyeu. Avatars grow legs: Generating smooth human motion from sparse tracking inputs with diffusion model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 481–490, 2023. 3

[15] Yao Feng, Vasileios Choutas, Timo Bolkart, Dimitrios Tzionas, and Michael J Black. Collaborative regression of expressive bodies using moderation. In *2021 International Conference on 3D Vision (3DV)*, pages 792–804. IEEE, 2021. 3

[16] Lin Geng Foo, Jia Gong, Hossein Rahmani, and Jun Liu. Distribution-aligned diffusion for human mesh recovery. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9221–9232, 2023. 3

[17] Jia Gong, Lin Geng Foo, Zhipeng Fan, Qiuhong Ke, Hossein Rahmani, and Jun Liu. Diffpose: Toward more reliable 3d pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13041–13051, 2023. 3

[18] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 2, 3, 5, 6

[19] Karl Holmquist and Bastian Wandt. Diffpose: Multi-hypothesis human pose estimation using diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15977–15987, 2023. 3

[20] Ludovic Hoyet, Kenneth Ryall, Rachel McDonnell, and Carol O'Sullivan. Sleight of hand: perception of finger motion from reduced marker sets. In *Proceedings of the ACM SIGGRAPH symposium on interactive 3D graphics and games*, pages 79–86, 2012. 6

[21] Hao Jiang and Kristen Grauman. Seeing invisible poses: Estimating 3d body pose from egocentric video. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3501–3509. IEEE, 2017. 2

[22] Zhongyu Jiang, Zhuoran Zhou, Lei Li, Wenhao Chai, Cheng-Yen Yang, and Jenq-Neng Hwang. Back to optimization:

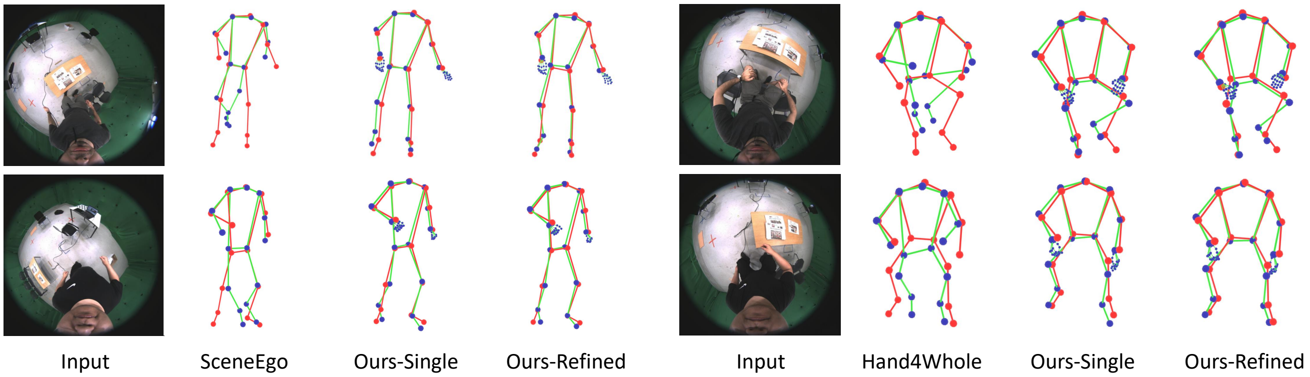

Figure 9. Qualitative comparison on human body pose estimations between our methods and the state-of-the-art SceneEgo [52] method. The red skeleton is the ground truth while the green skeleton is the predicted pose. Our methods predict more accurate body poses.

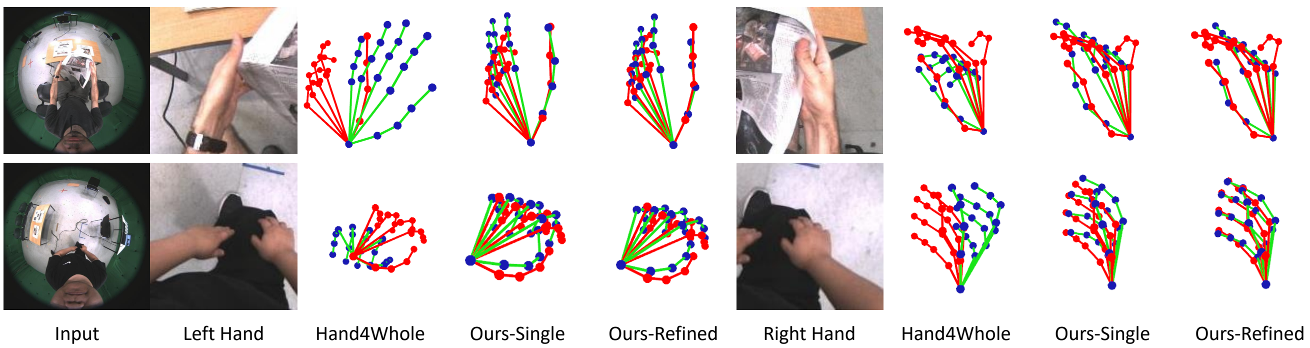

Figure 10. Qualitative comparison on hand pose estimation results. Our single-view and refined hand poses are more accurate than the poses from Hand4Whole [34] method. The red skeleton is the ground truth while the green skeleton is the predicted pose.

Diffusion-based zero-shot 3d human pose estimation. *arXiv preprint arXiv:2307.03833*, 2023. 3

[23] Taeho Kang, Kyungjin Lee, Jinrui Zhang, and Youngki Lee. Ego3dpose: Capturing 3d cues from binocular egocentric views. *arXiv preprint arXiv:2309.11962*, 2023. 2

[24] David G Kendall. A survey of the statistical theory of shape. *Statistical Science*, 4(2):87–99, 1989. 3

[25] Jiaman Li, Karen Liu, and Jiajun Wu. Ego-body pose estimation via ego-head pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17142–17151, 2023. 2, 3

[26] Yuyan Li, Yuliang Guo, Zhixin Yan, Xinyu Huang, Ye Duan, and Liu Ren. Omnifusion: 360 monocular depth estimation via geometry-aware fusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2801–2810, 2022. 5

[27] Jing Lin, Ailing Zeng, Haoqian Wang, Lei Zhang, and Yu Li. One-stage 3d whole-body mesh recovery with component aware transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21159–21168, 2023. 3

[28] Yuxuan Liu, Jianxin Yang, Xiao Gu, Yao Guo, and Guang-Zhong Yang. 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)*, pages 5271–5277. IEEE, 2022. 2

[29] Yuxuan Liu, Jianxin Yang, Xiao Gu, Yijun Chen, Yao Guo, and Guang-Zhong Yang. Egofish3d: Egocentric 3d pose estimation from a fisheye camera via self-supervised learning. *IEEE Transactions on Multimedia*, 2023. 1, 2, 4

[30] Yuxuan Liu, Jianxin Yang, Xiao Gu, Yao Guo, and Guang-Zhong Yang. Egohmr: Egocentric human mesh recovery via hierarchical latent diffusion model. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 9807–9813. IEEE, 2023. 1, 2, 3, 5

[31] Zhengyi Luo, Ryo Hachiuma, Ye Yuan, and Kris Kitani. Dynamics-regulated kinematic policy for egocentric pose estimation. *Advances in Neural Information Processing Systems*, 34:25019–25032, 2021. 2

[32] Christen Millerdurai, Hiroyasu Akada, Jian Wang, Diogo Luvizon, Christian Theobalt, and Vladislav Golyanik. Eventego3d: 3d human motion capture from egocentric event streams. In *37th IEEE/CVF Conference on Computer Vision and Pattern Recognition*. IEEE, 2024. 2

[33] Gyeongsik Moon, Ju Yong Chang, and Kyoung Mu Lee. V2v-posenet: Voxel-to-voxel prediction network for accurate 3d hand and human pose estimation from a single depth map. In *Proceedings of the IEEE conference on computer vision and pattern Recognition*, pages 5079–5088, 2018. 2

[34] Gyeongsik Moon, Hongsuk Choi, and Kyoung Mu Lee. Accurate 3d hand pose estimation for whole-body 3d human mesh estimation. In *Computer Vision and Pattern Recognition Workshop (CVPRW)*, 2022. 4, 5, 7, 8, 2

[35] Evonne Ng, Donglai Xiang, Hanbyul Joo, and Kristen Grauman. 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*, pages 9890–9900, 2020. 2

[36] Jinman Park, Kimathi Kaai, Saad Hossain, Norikatsu Sumi, Sirisha Rambhatla, and Paul Fieguth. 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*, pages 1837–1849, 2023. 2, 1

[37] Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed AA Osman, Dimitrios Tzionas, and Michael J Black. Expressive body capture: 3d hands, face, and body from a single image. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10975–10985, 2019. 3

[38] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1 (2):3, 2022. 5

[39] Helge Rhodin, Christian Richardt, Dan Casas, Eldar Insafutdinov, Mohammad Shafiei, Hans-Peter Seidel, Bernt Schiele, and Christian Theobalt. Egocap: egocentric marker-less motion capture with two fisheye cameras. *ACM Transactions on Graphics (TOG)*, 35(6):1–11, 2016. 2

[40] Yu Rong, Takaaki Shiratori, and Hanbyul Joo. Frankmocap: A monocular 3d whole-body pose estimation system via regression and integration. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1749–1759, 2021. 3

[41] Davide Scaramuzza, Agostino Martinelli, and Roland Siegwart. A toolbox for easily calibrating omnidirectional cameras. In *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5695–5701. IEEE, 2006. 1

[42] Wenkang Shan, Zhenhua Liu, Xinfeng Zhang, Zhao Wang, Kai Han, Shanshe Wang, Siwei Ma, and Wen Gao. Diffusion-based 3d human pose estimation with multi-hypothesis aggregation. *arXiv preprint arXiv:2303.11579*, 2023. 3

[43] Soshi Shimada, Vladislav Golyanik, Weipeng Xu, and Christian Theobalt. Physcap: Physically plausible monocular 3d motion capture in real time. *ACM Transactions on Graphics (ToG)*, 39(6):1–16, 2020. 5

[44] Xiao Sun, Bin Xiao, Fangyin Wei, Shuang Liang, and Yichen Wei. Integral human pose regression. In *Proceedings of the European conference on computer vision (ECCV)*, pages 529–545, 2018. 4, 5

[45] Yu Sun, Tianyu Huang, Qian Bao, Wu Liu, Wenpeng Gao, and Yili Fu. Learning monocular mesh recovery of multiple body parts via synthesis. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2669–2673. IEEE, 2022. 3

[46] Omid Taheri, Nima Ghorbani, Michael J Black, and Dimitrios Tzionas. Grab: A dataset of whole-body human grasping of objects. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IV 16*, pages 581–600. Springer, 2020. 6

[47] Denis Tomè, Patrick Peluse, Lourdes Agapito, and Hernán Badino. xr-egopose: Egocentric 3d human pose from an HMD camera. In *ICCV*, pages 7727–7737, 2019. 1, 2, 5, 6, 7, 8, 4

[48] Jonathan Tseng, Rodrigo Castellon, and Karen Liu. Edge: Editable dance generation from music. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 448–458, 2023. 5, 2

[49] Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, et al. Deep high-resolution representation learning for visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, 43(10):3349–3364, 2020. 5

[50] Jian Wang, Lingjie Liu, Weipeng Xu, Kripasindhu Sarkar, and Christian Theobalt. Estimating egocentric 3d human pose in global space. *ICCV*, 2021. 1, 2, 6, 7, 8, 3, 4, 5

[51] Jian Wang, Lingjie Liu, Weipeng Xu, Kripasindhu Sarkar, Diogo Luvizon, and Christian Theobalt. Estimating egocentric 3d human pose in the wild with external weak supervision. In *CVPR*, pages 13157–13166, 2022. 1, 2, 6, 7, 8, 4

[52] Jian Wang, Diogo Luvizon, Weipeng Xu, Lingjie Liu, Kripasindhu Sarkar, and Christian Theobalt. Scene-aware egocentric 3d human pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13031–13040, 2023. 1, 2, 5, 6, 7, 8, 3, 4

[53] Donglai Xiang, Hanbyul Joo, and Yaser Sheikh. Monocular total capture: Posing face, body, and hands in the wild. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10965–10974, 2019. 3

[54] Weipeng Xu, Avishek Chatterjee, Michael Zollhöfer, Helge Rhodin, Pascal Fua, Hans-Peter Seidel, and Christian Theobalt. Mo^2cap^2: Real-time mobile 3d motion capture with a cap-mounted fisheye camera. *IEEE Trans. Vis. Comput. Graph.*, 25(5):2093–2101, 2019. 1, 2, 5, 6, 7, 8, 3, 4

[55] Yufei Xu, Jing Zhang, Qiming Zhang, and Dacheng Tao. Vitpose: Simple vision transformer baselines for human pose estimation. *Advances in Neural Information Processing Systems*, 35:38571–38584, 2022. 2

[56] Dianyi Yang, Jiadong Tang, Yu Gao, Yi Yang, and Mengyin Fu. Sector patch embedding: An embedding module conforming to the distortion pattern of fisheye image. *arXiv preprint arXiv:2303.14645*, 2023. 5

[57] Shangrong Yang, Chunyu Lin, Kang Liao, and Yao Zhao. Dual diffusion architecture for fisheye image rectification: Synthetic-to-real generalization. *arXiv preprint arXiv:2301.11785*, 2023.

[58] Fanghua Yu, Xintao Wang, Mingdeng Cao, Gen Li, Ying Shan, and Chao Dong. Osrt: Omnidirectional image super-resolution with distortion-aware transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13283–13292, 2023. 5

[59] Ye Yuan and Kris Kitani. 3d ego-pose estimation via imitation learning. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 735–750, 2018. 2

[60] Ye Yuan and Kris Kitani. Ego-pose estimation and forecasting as real-time pd control. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10082–10092, 2019. 2

[61] Ye Yuan, Jiaming Song, Umar Iqbal, Arash Vahdat, and Jan Kautz. Physdiff: Physics-guided human motion diffusion model. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16010–16021, 2023. 5

[62] Siwei Zhang, Qianli Ma, Yan Zhang, Zhiyin Qian, Taein Kwon, Marc Pollefeys, Federica Bogo, and Siyu Tang. Egobody: Human body shape and motion of interacting people from head-mounted devices. In *European Conference on Computer Vision*, pages 180–200. Springer, 2022. 6

[63] Siwei Zhang, Qianli Ma, Yan Zhang, Sadegh Aliakbarian, Darren Cosker, and Siyu Tang. Probabilistic human mesh recovery in 3d scenes from egocentric views. *arXiv preprint arXiv:2304.06024*, 2023. 3

[64] Yahui Zhang, Shaodi You, and Theo Gevers. 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*, pages 1772–1781, 2021. 2

[65] Dongxu Zhao, Zhen Wei, Jisan Mahmud, and Jan-Michael Frahm. Egoglass: Egocentric-view human pose estimation from an eyeglass frame. In *2021 International Conference on 3D Vision (3DV)*, pages 32–41. IEEE, 2021. 2

[66] Yuxiao Zhou, Marc Habermann, Ikhsanul Habibie, Ayush Tewari, Christian Theobalt, and Feng Xu. Monocular real-time full body capture with inter-part correlations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4811–4822, 2021. 3

[67] Christian Zimmermann, Duygu Ceylan, Jimei Yang, Bryan Russell, Max Argus, and Thomas Brox. Freihand: A dataset for markerless capture of hand pose and shape from single rgb images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 813–822, 2019. 2