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

Jian Wang[1,4] Zhe Cao[2] Diogo Luvizon[1,4] Lingjie Liu[3]
Kripasindhu Sarkar[2] Danhang Tang[2] Thabo Beeler[2] Christian Theobalt[1,4]
[1]MPI Informatics & Saarland Informatics Campus [2]Google [3]University of Pennsylvania
[4]Saarbrücken Research Center for Visual Computing, Interaction and Artificial Intelligence

Project Page: https://jianwang-mpi.github.io/egowholemocap

Figure 1. From an image sequence captured by a single head-mounted fisheye camera, our method can predict accurate and temporally coherent whole-body motion, including human body and hand poses. The SMPL-X parameters are obtained using inverse kinematics.

Abstract

In this work, we explore egocentric whole-body motion capture using a single fisheye camera, which simultaneously estimates human body and hand motion. This task presents significant challenges due to three factors: the lack of high-quality datasets, fisheye camera distortion, and human body self-occlusion. To address these challenges, we propose a novel approach that leverages FisheyeViT to extract fisheye image features, which are subsequently converted into pixel-aligned 3D heatmap representations for 3D human body pose prediction. For hand tracking, we incorporate dedicated hand detection and hand pose estimation networks for regressing 3D hand poses. Finally, we develop a diffusion-based whole-body motion prior model to refine the estimated whole-body motion while accounting for joint uncertainties. To train these networks, we collect a large synthetic dataset, EgoWholeBody, comprising 840,000 high-quality egocentric images captured across a diverse range of whole-body motion sequences. Quantitative and qualitative evaluations demonstrate the effectiveness of our method in producing high-quality whole-body motion estimates from a single egocentric camera.

1. Introduction

Egocentric 3D human motion estimation using head-mounted devices [47, 54] has garnered significant traction in recent years, driven by its diverse applications in VR/AR. Immersed in a virtual world, we can traverse virtual environments, interact with virtual objects, and even simulate real-world interactions. To fully capture the intricacies of human motion during such interaction, understanding both body and hand movements is essential. While existing egocentric motion capture methods [30, 47, 50–52, 54] focus solely on body motion, neglecting the hands, this work proposes the task of egocentric *whole-body* motion capture, *i.e.* simultaneous estimation of the body motion and hand motion from a single head-mounted fisheye camera (shown in Fig. 1). This task is extremely challenging due to three factors: First, the fisheye image introduces significant distortion, making it difficult for existing networks, which are designed for non-distorted images, to extract features. Second, the egocentric camera perspective frequently leads to the occlusion of body parts, such as the feet and hands, further complicating the task of whole-body motion capture. Lastly, large-scale training data with ground truth annotations for both body and hand poses is absent in existing datasets [4, 29, 47, 51, 54].

In this work, we propose a novel egocentric whole-body motion capture method to address the aforementioned challenges. To effectively address fisheye distortion, we propose *FisheyeViT* for extracting image features, along with a joint regressor employing *pixel-aligned 3D heatmap* for predicting 3D body poses. Instead of attempting to undis-

tort the entire fisheye image, which is impractical due to the fisheye lens's large field of view (FOV), we opt to partition the image into smaller patches aligned with a specific FOV range. This approach enables individual patch-level undistortion and seamlessly aligns with the vision transformer architecture that is employed for extracting the complete image feature map. We further propose an egocentric 3D pose regressor utilizing 3D heatmap representations. Unlike the existing approach [52] that projects image features into 3D space through fisheye reprojection functions and regresses 3D heatmaps with V2V networks [33]–leading to intricate network learning and high computational complexity–our proposed egocentric pose regressor adopts a simpler approach. It employs deconvolutional layers to obtain pixel-aligned 3D heatmaps. Notably, the voxels in the 3D heatmap directly correspond to pixels in 2D features, subsequently linking to image patches in FisheyeViT. This streamlined approach significantly simplifies network training. Joint locations from the pixel-aligned 3D heatmap are finally transformed with the fisheye camera model to obtain the 3D human body poses. Due to the large size difference between body and hands, we train a hand detection network and a hand pose estimation network to accurately regress 3D hand poses.

To overcome the challenges posed by self-occlusion and improve the accuracy of pose estimation, we propose a novel method for refining the whole-body motion predictions by incorporating temporal context and a motion prior. Our method learns a whole-body motion prior with the diffusion model [18] from a collection of diverse human motion sequences, capturing intrinsic correlations between hand and body movements. Following this, we extract the joint uncertainties from the pixel-aligned 3D heatmap and utilize them to guide the refinement of the whole-body motion. The joint uncertainties act as indicators of the trustworthiness of the pose regressor's predictions. By conditioning on joints with low uncertainty, our whole-body motion diffusion model selectively refines joints with high uncertainty. This strategy substantially improves the quality of whole-body pose estimations and effectively mitigates the effects of self-occlusion.

In response to the absence of the egocentric whole-body motion capture datasets, we present *EgoWholeBody*, a new large-scale high-quality synthetic dataset. This dataset encompasses a wide range of whole-body motions, comprising over 870k frames, which significantly surpasses the size of previous egocentric training datasets. EgoWholeBody could serve as a valuable resource for advancing research in egocentric whole-body motion capture.

A thorough evaluation across a range of datasets, including SceneEgo [52], GlobalEgoMocap [50] and Mo^2Cap^2 [54], has demonstrated the remarkable improvements of our method in estimating egocentric whole-body motion compared to previous approaches. This substantiates the effectiveness of our approach in addressing the special challenges encountered in egocentric views, including the fisheye distortion and self-occlusion.

In summary, our key contributions are the following:

- The first egocentric whole-body motion capture method that predicts accurate and temporarily coherent egocentric body and hand motion;
- FisheyeViT for alleviating fisheye camera distortion and pose regressor using pixel-aligned 3D heatmaps for accurate egocentric body pose estimation from a single image;
- Uncertainty-aware refinement method based on motion diffusion models for correcting initial pose estimations and predicting plausible motions even under occlusion;
- *EgoWholeBody*, a new high-quality synthetic dataset for egocentric whole-body motion capture.

2. Related Work

Egocentric 3D Human Body Pose Estimation. Recently, there has been growing interest in estimating egocentric 3D poses from body-worn cameras. Some methods [21, 25, 31, 35, 59, 60] use front-facing cameras and infer the human body motion from the camera view. However, since the user's body is often unobserved by the camera, these methods fail when the human body is not roaming around. Millerdurai *et al*. [32] leverage event cameras for estimating egocentric body pose. Other methods [4, 5, 7, 23, 39, 65] use head-mounted down-facing stereo cameras to estimate body poses. However, stereo camera setups introduce extra burdens of weight and energy consumption.

Xu *et al*. [54] and Tome *et al*. [47] introduce the single head-mounted down-facing fisheye camera setup for the egocentric 3D human pose estimation task. Zhang *et al*. [64] regressed fisheye camera parameters and 3D human pose simultaneously. To address the self-occlusion issue, Park *et al*. [36] leveraged the temporal information with the spatio-temporal self-attention network, and Liu *et al*. [30] introduced diffusion model to generate 3D human pose conditioned on egocentric image features. Wang *et al*. [50] and Liu *et al*. [28] combined the SLAM and egocentric pose estimation methods to estimate human body poses in the world coordinate. Wang *et al*. [51] and Liu *et al*. [29] leverage the synchronized egocentric camera and external cameras to collect large-scale egocentric pose estimation datasets with pseudo-ground truth. Considering the human-scene interaction, Wang *et al*. [52] estimated the scene geometry from the egocentric camera and constrained the 3D human pose with it.

These methods only focus on estimating human body poses while omitting the hand motion, and they still suffer from fisheye camera distortion since they directly put the highly distorted fisheye images into the neural network. Our proposed method can capture whole-body motion and

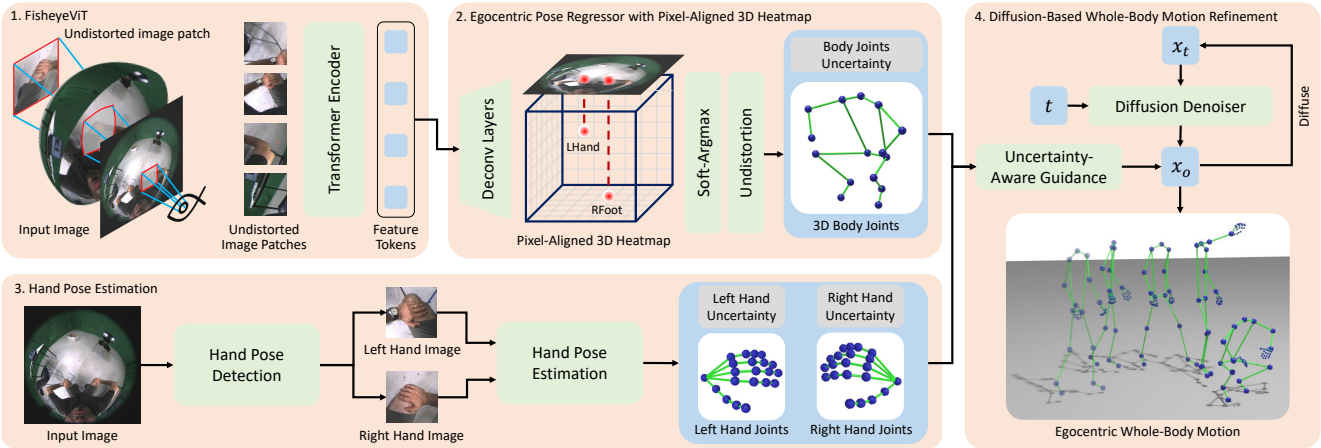

Figure 2. Overview of our whole-body motion capture pipeline. We first use FisheyeViT to undistort the input image and generate image feature tokens (3.1.1). Next, we use a 1D convolutional network to convert the image features to a pixel-aligned 3D heatmap and use soft-argmax and fisheye camera undistortion function to obtain the 3D body joins positions and uncertainty (3.1.2). We further detect the hand location and regress the 3D hand poses from the input image (3.1.3). Finally, the estimated hand motion and human body motion are combined and the uncertainty-aware diffusion model is applied to refine the estimated whole-body motion (3.2).

resolve the fisheye camera distortion issue with the FisheyeViT and pixel-aligned 3D heatmap.

Whole-Body 3D Pose Estimation. Whole-body 3D pose estimation aims to estimate the 3D human body, face, and hands parameters from input images. This task is crucial for many applications, e.g., modeling human activities and human-scene interactions. Some methods [37, 53] fit the 2D body joints estimated from images with optimization algorithms, while these methods suffer from high computation overhead and can fall into local optima. Some other learning-based methods [6, 9, 15, 27, 40, 45, 66] use the neural network to regress the SMPL-X [37] parameters from input images. For example, ExPose [9] introduced body-driven attention to extract face and hand crops and used a refinement module to regress whole-body pose. OSX [27] proposed a one-stage pipeline for whole-body mesh recovery without separate networks for each part. SMPLer-X [6] propose a foundation model for whole-body pose estimation trained with the large model and big data.

Though much progress has been made on whole-body pose estimation from an external view, the task from an egocentric view is still unexplored. In this paper, we introduce the first whole-body 3D pose estimation method from a single egocentric image and also incorporate temporal information with diffusion-based motion refinement.

Diffusion Models for Pose Estimation. Recently, some methods [8, 10, 16, 17, 19, 42] have effectively applied Denoising Diffusion Probabilistic Models (DDPM) [18] to human pose estimation tasks. Building on the success of motion diffusion models in human pose estimation, many methods have extended this approach to egocentric pose estimation, where the human body is only partially visible from RGB cameras or VR sensors. Zhang *et al.*'s work [63] uses a diffusion model to generate realistic human poses considering scene geometry. AGROL [14] generates body motion based on head and hand 6D pose estimates from a VR headset. EgoEgo [25] estimates head poses from a head-mounted front-facing camera and uses them to generate body poses. EgoHMR [30] extracts image features and uses them as a condition for the diffusion denoising process.

However, the aforementioned pose estimation methods train the *conditioned* diffusion model with image features or IMU signals. This cannot be generalized since the trained network only accepts one specific condition format and is inclined to learn domain-specific distributions of condition features. ZeDO [22] tackles this issue with a zero-shot diffusion-based optimization approach that doesn't require training with 2D-3D or image-3D pairs. Our method leverages the uncertainty value given by the single-frame pose estimation network and refines the initial motion estimation with the uncertainty of each joint. Moreover, different from previous methods that only focus on human body motion, we train a whole-body motion diffusion model to construct the relationship between hand and body motion.

3. Method

In this section, we propose a new method for predicting accurate egocentric whole-body poses from egocentric image sequences. An overview of our approach is shown in Fig. 2.

3.1. Single Image Based Egocentric Pose Estimation

3.1.1 FisheyeViT

In this section, we introduce FisheyeViT, which is specially designed to alleviate the fisheye distortion issue. Instead of undistorting the entire fisheye image, we extract undistorted

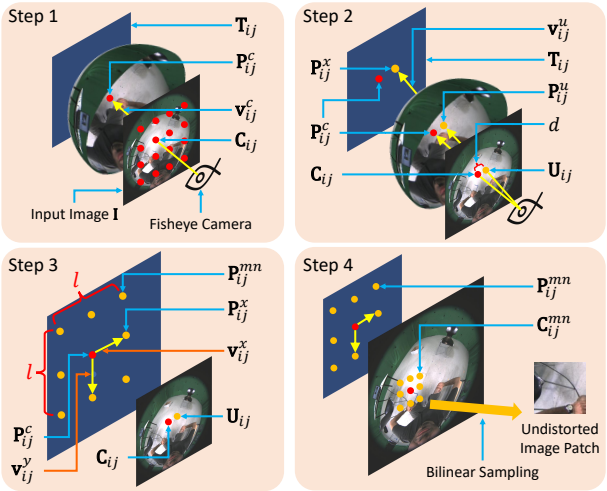

Figure 3. The detailed illustration of FisheyeViT (Sec. 3.1.1).

image patches from the fisheye image and then fit these patches as tokens into the transformer network [13]. To get the undistorted patches, we first warp the fisheye image to a unit semi-sphere, then get the patches with the gnomonic projection (see Fig. 2). The FisheyeViT can be split into five steps, the first four of which are illustrated in Fig. 3.

Step 1. Given an input image $\mathbf{I}$ with size $H \times W$, we first evenly sample $N \times N$ patch center points: $\{\mathbf{C}_{ij} = (u_i, v_j) = \left(\frac{H}{N}(i+\frac{1}{2}), \frac{W}{N}(j+\frac{1}{2})\right) | i,j \in 0,...,N-1\}$. Then, the patch center points $\mathbf{C}_{ij}$ are projected onto a unit sphere with the fisheye reprojection function: $\mathbf{P}^c_{ij} = (x^c_{ij}, y^c_{ij}, z^c_{ij}) = \mathcal{P}^{-1}(u_i, v_j, 1)$. The fisheye camera model is described in Sec. 8 of the supplementary material. Given a point $\mathbf{P}^c_{ij}$ on the unit sphere, the tangent plane $\mathbf{T}_{ij}$ that passes through the point is defined by the normal vector $\mathbf{v}^c_{ij} = (x^c_{ij}, y^c_{ij}, z^c_{ij})$. In the following steps, we implement the gnomonic projection by sampling grid points in the plane and projecting them back onto the fisheye image.

Step 2. In this step, we determine the orientation of the grid points in the tangent plane, ensuring that the grid points from different tangent planes $\mathbf{T}_{ij}$ have the same orientation when projected back onto the fisheye image. To achieve this, we select a 2D point $\mathbf{U}_{ij} = (u_i + d, v_j)$ in the fisheye image space that is d pixels to the right of the patch center point and project it to the unit sphere using the fisheye reprojection function: $\mathbf{P}^u_{ij} = (x^u_{ij}, y^u_{ij}, z^u_{ij}) = \mathcal{P}^{-1}(u_i + d, v_j, 1)$. We then calculate the intersection point $\mathbf{P}^x_{ij}$ between the vector $\mathbf{v}^u_{ij} = (x^u_{ij}, y^u_{ij}, z^u_{ij})$ that is passing the origin and the tangent plane $\mathbf{T}_{ij}$:

$$\mathbf{P}^x_{ij} = \frac{\langle \mathbf{P}^c_{ij}, \mathbf{v}^c_{ij} \rangle}{\langle \mathbf{v}^u_{ij}, \mathbf{v}^c_{ij} \rangle} \mathbf{v}^u_{ij} = \frac{1}{\langle \mathbf{v}^u_{ij}, \mathbf{v}^c_{ij} \rangle} \mathbf{v}^u_{ij}, \tag{1}$$

where $\langle \cdot, \cdot \rangle$ denotes the inner product.

Step 3. Based on the center point $\mathbf{P}^c_{ij}$ and intersection point $\mathbf{P}^x_{ij}$ on the tangent plane $\mathbf{T}_{ij}$, we build a coordinate system with the x axis: $\mathbf{v}^x_{ij} = \text{Norm}(\mathbf{P}^x_{ij} - \mathbf{P}^c_{ij})$, the z axis: $\mathbf{v}^z_{ij} = \text{Norm}(\mathbf{v}^c_{ij})$ and the y axis: $\mathbf{v}^y_{ij} = \mathbf{v}^z_{ij} \times \mathbf{v}^x_{ij}$, where Norm denotes the normalize operation. We grid-sample $M \times M$ points in a $l \times l$ square on the x-y plane:

$$\{\mathbf{P}^{mn}_{ij} = \mathbf{P}^c_{ij} + (l\frac{m}{M}\mathbf{v}^x_{ij}, l\frac{n}{M}\mathbf{v}^y_{ij})\} \tag{2}$$

where $m, n \in -\frac{1}{2}(M-1), ..., -\frac{3}{2}, -\frac{1}{2}, \frac{1}{2}, \frac{3}{2}, ...\frac{1}{2}(M-1)$.

Step 4. The points $\mathbf{P}^{mn}_{ij}$ are projected back to the fisheye image with the fisheye projection function: $\mathbf{C}^{mn}_{ij} = \mathcal{P}(\mathbf{P}^{mn}_{ij})$. We then apply bilinear sampling to obtain the colors at points $\mathbf{C}^{mn}_{ij}$ of the input image $\mathbf{I}$, yielding the undistorted image patch $\mathbf{I}^{\text{undis}}_{ij}$. Please also see the supplementary video for a visual demonstration of undistorted image patches and their movement on the fisheye image.

Step 5. The image patches $\{\mathbf{I}^{\text{undis}}_{ij}\}$ are sent to a ViT transformer network [13] to obtain the feature tokens $\{\mathbf{F}_{ij}\}$. The feature token is further reshaped in $i \times j$ matrix and obtain the image feature $\mathbf{F}$. In the FisheyeViT, we empirically chose $N = 16; M = 16; d = 8; l = 0.2m$ given the image size $H = W = 256$.

Note that $\mathbf{C}^{mn}_{ij}$ is independent of the image $\mathbf{I}$. This means that, given a fixed fisheye camera model, we can precompute $\mathbf{C}^{mn}_{ij}$ for all combinations of m, n and i, j in advance. This significantly speeds up both the training and evaluation processes. Furthermore, the number and dimensions of image patches $\{\mathbf{I}^{\text{undis}}_{ij}\}$ match exactly with those in the traditional ViT network. This compatibility allows us to finetune existing ViT networks on our egocentric datasets. Our sampling strategy ensures that each image patch $\mathbf{I}^{\text{undis}}_{ij}$ corresponds to the same FOV range in the fisheye camera. In our ablation study in Sec. 5.3, we show that FisheyeViT enhances the performance of the pose estimation network when applied to egocentric fisheye images.

3.1.2 Pose Regressor with Pixel-Aligned 3D Heatmap

After collecting image features with FisheyeViT, we utilize a 3D heatmap-based network to estimate the body poses. The existing 3D heatmap-based pose regressors [34, 44] are designed for the weak-perspective cameras and predict the 3D heatmap in xyz space. Directly applying these regressors will result in misalignment between 3D heatmap features in xyz space and 2D image features in the fisheye image space. Therefore, we introduce a novel egocentric pose regressor that relies on the pixel-aligned 3D heatmap, tailored to address the needs of fisheye cameras. The idea is to regress the 3D heatmap in uvd space rather than traditional xyz space, where uv corresponds to the fisheye image uv space. Specifically, given a feature map $\mathbf{F} \in \mathbb{R}^{C \times N \times N}$, where C is the channel number, N is feature map height and width, we firstly use two deconvolutional layers to convert the feature map $\mathbf{F}$ into shape

$(D_h \times J, H_h, W_h)$, and further reshape it to pixel-aligned 3D heatmap $\mathbf{H} \in \mathbb{R}^{J \times D_h \times H_h \times W_h}$, where J is the joint number and D_h, H_h, W_h is the 3D heatmap depth, height and width. The illustration of pixel-aligned 3D heatmap is shown in Fig. 2. Next, we obtain the max-value positions $\tilde{\mathbf{J}}_b = \{(u_i, v_i, d_i) \mid i \in 0, 1, 2, ..., J\}$ from $\mathbf{H}$ by the differentiable soft-argmax operation [44]. Here, we note that u_i and v_i correspond to the uv-coordinate of the 3D body joint projected in the fisheye image space, and d_i denotes the distance of the joint to the fisheye camera. Finally, the 3D body joints $\hat{\mathbf{J}}_b = \{(x_i, y_i, z_i) \mid i \in 0, 1, 2, ..., J\}$ are recovered with the fisheye reprojection function: $(x_i, y_i, z_i) = \mathcal{P}^{-1}(u_i, v_i, d_i)$. The predicted body pose $\hat{\mathbf{J}}_b$ is finally compared with the ground truth body pose $\mathbf{J}_b$ with the MSE loss. By first regressing 3D body poses in uvd space and then reprojecting it, we ensure that the 3D heatmap is pixel-aligned with the end-to-end training.

With the pixel-aligned heatmap, our proposed 3D pose regressor solves problems in all three types of previous egocentric joint regressors. First, Mo^{2}Cap^2 [54] employs separate networks to predict 2D joint positions and joint distances. However, this method can yield unrealistic joint estimations because small errors in 2D joints can result in large errors in 3D joints due to the projection effect. Second, xR-egopose [47] and EgoHMR [30] directly regress the 3D joint positions. However, this method is agnostic to the fisheye camera parameters, making it suitable only for a specific camera configuration (*e.g.*, camera parameters, head-mounted position, *etc.*). Third, SceneEgo [52] projects 2D features into 3D voxel space and uses a V2V network to regress 3D poses. Because of these, the SceneEgo method suffers from low accuracy and large computation overhead. Different from previous methods, our pose regressor with pixel-aligned 3D heatmap is versatile and efficient since it directly estimates 3D joints while also incorporating an explicitly parametrized fisheye camera model. Moreover, it can preserve the uncertainty of the estimated joints, which will be used in our uncertainty-aware motion refinement method (Sec. 3.2.2). Detailed comparison with other pose prediction heads is shown in Table 3.

3.1.3 Egocentric Hand Pose Estimation

In this section, we first train a network to detect hand pose locations and then train a 3D hand pose estimation network to regress 3D hand poses. Then, we describe how to integrate the estimated hand and body poses.

Hand Detection. Given an input image $\mathbf{I}$, we finetune the HRNet [49] network to regress the 2D hand poses of left hand $\mathbf{J}^{2d}_{lh}$ and right hand $\mathbf{J}^{2d}_{rh}$. From the hand poses, we obtain the center point of left hand $\mathbf{C}_{lh}$ and right hand $\mathbf{C}_{rh}$, along with the bounding box sizes, d_{lh} and d_{rh}. We use our approach described in Sec. 3.1.1 to compute undistorted image patches of left $\mathbf{I}_{lh}$ and right hands $\mathbf{I}_{rh}$.

Hand Pose Estimation. Given the cropped image $\mathbf{I}_{lh}$ or $\mathbf{I}_{rh}$, we regress the 3D hand poses $\hat{\mathbf{J}}^{loc}_{lh}$ and $\hat{\mathbf{J}}^{loc}_{rh}$ with the Hand4Whole [34] network, which is fine-tuned on our Ego-FullBody dataset.

Integration of Body and Hand Poses. It is not straightforward to integrate the hand poses with the body pose in the egocentric camera view primarily due to the fisheye camera's perspective effects. Take the left hand as an example. Following Step 3 in Sec. 3.1.1, we establish a local coordinate system on the tangent plane of the left-hand image with XYZ axes as follows: $x : \mathbf{v}^x_{lh}$; $y : \mathbf{v}^y_{lh}$; $z : \mathbf{v}^z_{lh}$. We define a rotation matrix, denoted as $\mathbf{R}$, that represents the transformation between the root coordinate system and the local coordinate system on the tangent plane. The estimated hand pose is first rotated with the rotation matrix $\hat{\mathbf{J}}_{lh} = \mathbf{R}\hat{\mathbf{J}}^{loc}_{lh}$ and then translated to align the wrist location of the human body. This same process is also applied to the right hand to get the right hand pose $\hat{\mathbf{J}}_{rh}$. The whole-body joints $\hat{\mathbf{J}}$ are obtained by combining $\hat{\mathbf{J}}_b$, $\hat{\mathbf{J}}_{lh}$, and $\hat{\mathbf{J}}_{rh}$. The uncertainty of whole-body joints $\hat{\mathbf{U}}$ is also obtained from the maximal value of the 3D heatmap in pose estimation modules.

3.2. Diffusion-Based Motion Refinement

We notice that the single-frame estimations in Sec. 3.1 suffer from inaccuracies and temporal instabilities. In this section, we propose a diffusion-based motion refinement method to tackle this problem. We first learn the whole-body motion prior with the motion diffusion model in Sec. 3.2.1 and then introduce an uncertainty-aware zero-shot motion refinement method in Sec. 3.2.2.

3.2.1 Whole-Body Motion Diffusion Model

We follow DDPM [18] as our diffusion approach to capture the whole-body motion prior $q(\mathbf{x})$. DDPM learns a distribution of whole-body motion $\mathbf{x}$ through a forward diffusion process and an inverse denoising process. The forward diffusion process is a Markov process of adding Gaussian noise over $t \in \{0, 1, ..., T-1\}$ steps:

$$q(\mathbf{x}_t|\mathbf{x}_{t-1}) = \mathcal{N}(\sqrt{\alpha_t}\mathbf{x}_{t-1}, (1-\alpha_t)I) \tag{3}$$

where $\mathbf{x}_t$ denotes the whole-body motion sequence at step t, the variance $(1-\alpha_t) \in (0, 1]$ denotes a constant hyper-parameter increases with t.

The inverse process uses a denoising network $D(\cdot)$ to remove the added Gaussian noise at each time step t. Here we use the transformer-based framework in EDGE [48] as the motion-denoising network $D(\cdot)$. We follow Ramesh *et al.*'s work [38] to make the network predict the original signal itself, *i.e.* $\hat{\mathbf{x}}_0 = D(\mathbf{x}_t, t)$ and train it with the simple objective [18]:

$$\mathcal{L}_{\text{simple}} = E_{\mathbf{x}_0 \sim q(\mathbf{x}_0), t \sim [1,T]} \left[||\mathbf{x}_0 - D(\mathbf{x}_t, t)||^2_2 \right] \tag{4}$$

3.2.2 Uncertainty-Aware Motion Refinement

Given the learned whole-body motion prior, we leverage the uncertainty value for each pose to guide the diffusion denoising process with the classifier-guided diffusion sampling [12]. Given an initial sequence of whole-body pose estimation $\mathbf{x}_e = \{\hat{\mathbf{J}}_i\}$ and the uncertainty value for each pose $\mathbf{u} = \{\hat{\mathbf{U}}_i\}$, where i denotes the ith pose in the sequence, we keep the joints with low uncertainty but use the diffusion model to generate joints with high uncertainty conditioned on the low-uncertainty joints. Specifically, in the tth sampling step of the diffusion process, the denoising network predicts $\hat{\mathbf{x}}_0 = D(\mathbf{x}_t, t)$, which is noised back to $\mathbf{x}_{t-1}$ by sampling from the Gaussian distribution:

$$\mathbf{x}_{t-1} \sim \mathcal{N}(\hat{\mathbf{x}}_0 + \mathbf{w}(\mathbf{x}_e - \hat{\mathbf{x}}_0), \Sigma_t) \tag{5}$$

where Σ_t is a scheduled Gaussian distribution in DDPM [18] and $\mathbf{w}$ controls the weight of a specific joint between the predicted motion $\hat{\mathbf{x}}_0$ and the estimated motion $\mathbf{x}_e$. Generally, we expect $\mathbf{w} \to \overrightarrow{0}$ when $t \to 0$ such that the temporal stability is guaranteed through the generation of the denoising process, and $\mathbf{w} \to \overrightarrow{1}$ when $t \to T$ such that the denoising process is initialized by the estimated motion $\mathbf{x}_e$. We also expect that $w_{ij} = \mathbf{w}[i][j]$, which is the weight of jth joints in the ith pose, is smaller when the uncertainty value $u_{ij} = \mathbf{u}[i][j]$ of the jth joints in the ith pose is large. Based on this requirement, we design $\mathbf{w}$ as:

$$\mathbf{w} = 1/\left(1 + e^{-k(t - T\mathbf{u})}\right) \tag{6}$$

where T is the overall diffusion steps, k is a hyperparameter which is empirically set to 0.1. From the experimental results in Sec. 5, we demonstrate the effectiveness of uncertain-aware motion refinement and our uncertainty-guided diffusion sampling strategy.

4. EgoWholeBody Dataset

In this section, we introduce EgoWholeBody, a large-scale high-quality synthetic dataset built for the task of egocentric whole-body motion capture. The EgoWholeBody dataset is organized into two sections. The first part, containing over 700k frames, is rendered with 14 different rigged Renderpeople [3] models driven by 2367 Mixamo [2] motion sequences. The second part focuses on hand motions and contains 170k frames with the SMPL-X model. This data is constructed from 24 different shapes and textures, driven by 262 motion sequences selected from the GRAB [46] and TCDHandMocap dataset [20]. We also created synthetic test sequences, which include 133k images rendered with 3 Renderpeople models and Mixamo motions.

During the rendering process, we first attach a virtual fisheye camera to the forehead of human body models and render the images, semantic labels, and depth map with Blender [1]. Our dataset is larger and more diverse than previous egocentric training datasets–see Sec. 10 in the supplementary material for a detailed comparison.

5. Experiments

5.1. Datasets and Evaluation Metrics

Training Datasets. To train our body pose estimation module (Sec. 3.1.1 and Sec. 3.1.2), we use our EgoWholeBody dataset and the EgoPW dataset [50]. Additionally, the EgoWholeBody dataset is used to train the hand pose estimation module in Sec. 3.1.3. For training the whole-body diffusion model (Sec. 3.2), we utilize a combined motion capture dataset that includes EgoBody [62], Mixamo [2], TCDHandMocap dataset [20] and GRAB dataset [46].

Evaluation Datasets. In our experiment, we evaluate our methods on four datasets: the GlobalEgoMocap test datasets [50], the Mo^{2}Cap^2 test dataset [54], the SceneEgo test dataset [52] and out EgoWholeBody test dataset. The details of the datasets are shown in Sec. 12 of supplementary materials. Note that evaluating whole-body poses requires accurate annotations for human hands, which is absent in real-world datasets. To resolve the issue, we request the multi-view videos of the SceneEgo test dataset [52] from the authors and use a multi-view motion capture system to obtain the hand motion. The hand pose annotations will be made publicly available.

Evaluation Metrics. To evaluate the precision of human body poses on the SceneEgo test dataset [52], we use MPJPE and PA-MPJPE. For the GlobalEgoMocap test dataset [50] and Mo^{2}Cap^2 test dataset [54], where egocentric camera poses are unavailable, we evaluate PA-MPJPE and BA-MPJPE. For hand pose accuracy, we align the predicted and ground truth hand poses at the root position, followed by computing MPJPE and PA-MPJPE. Detailed explanations of these metrics are in Sec. 11 of the supplementary materials. All reported metrics are in millimeters.

5.2. Comparisons on Whole-Body Pose Estimation

For a fair comparison with existing methods focusing solely on body or hand pose, we split our evaluation into two parts, reporting results of body poses in Table 1 and hand pose in Table 2. We first compare the accuracy of the human body poses with state-of-the-art methods, including EgoPW [51] and SceneEgo [52], on EgoWholeBody and SceneEgo [52] test datasets. The comparison with more previous methods [47, 50, 54] and on more evaluation datasets [50, 54] are shown in Sec. 7 of the supplementary materials. Since our motion refinement method incorporates random Gaussian noise, we generate five samples and calculate the average MPJPE values. The standard deviation is low (< 0.01mm) and is discussed in Sec. 13 of supplementary materials. Results are presented in Table 1, where our single-frame re-

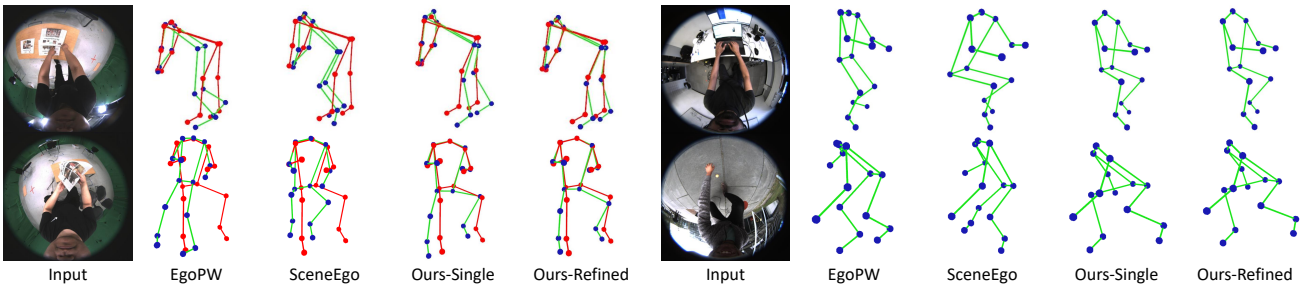

Figure 4. Qualitative comparison on human body pose estimations between our methods and the state-of-the-art egocentric pose estimation methods on in-the-studio (left column) and in-the-wild scenes (right column). The red skeleton is the ground truth while the green skeleton is the predicted pose. Our methods predict more accurate body poses compared with EgoPW [51] and SceneEgo [52].

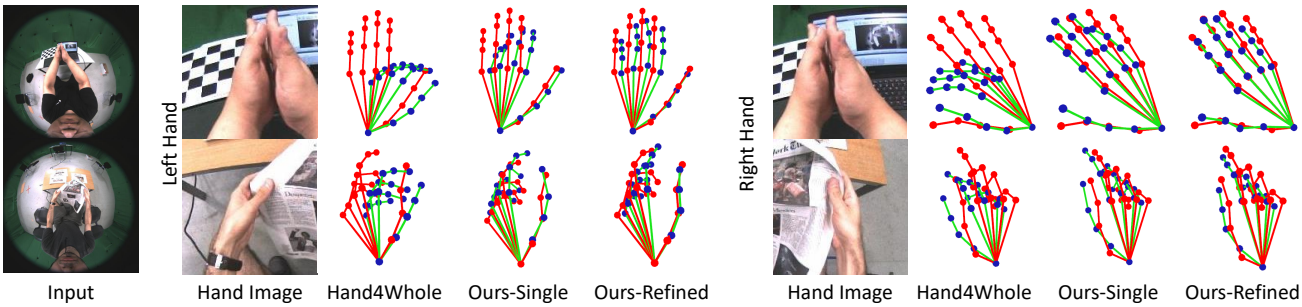

Figure 5. Qualitative comparison on human hand pose estimations between our methods and the state-of-the-art third-view pose estimation methods. Our single-view and refined hand poses are more accurate than the poses from Hand4Whole [34] method. The red skeleton is the ground truth while the green skeleton is the predicted pose.

sults are labeled as "Ours-Single" and our refinement results are labeled as "Ours-Refined". Our single-frame body pose estimation method outperforms all previous methods by a large margin. Our diffusion-based motion refinement method can further improve the accuracy of body poses estimated by the single-frame methods.

Note that previous methods [47, 50–52, 54] use training datasets different from each other. For a fair comparison, we re-train previous methods with our training datasets in Sec. 5.1 and show the results with "*" in Table 1. This retraining led to significant improvements across all previous methods, demonstrating our dataset's broad applicability. However, these methods still underperformed compared to ours, highlighting our approach's superiority.

To evaluate the accuracy of our hand pose estimation method, we first crop the hand images with the hand detection method in Sec. 3.1.3. Then we show the results of our single-frame hand pose estimation (labeled as "Ours-Single") and whole-body motion refinement methods (labeled as "Ours-Refined") in Table 2. Our single-frame hand pose estimation method outperforms the state-of-the-art method Hands4Whole [34], demonstrating the effectiveness of training the network on our EgoWholeBody dataset. Our whole-body motion refinement method can also enhance the accuracy of hand motion.

For a qualitative comparison, we compare the body and hand poses of our method with existing methods on the SceneEgo dataset and the in-the-wild EgoPW [51] evaluation sequences. The results are shown in Fig. 4 and Fig. 5, showing that our method can predict high-quality whole-body poses from an egocentric camera. Please refer to our supplementary video for more qualitative evaluation results.

Method	MPJPE	PA-MPJPE
SceneEgo test dataset [52]		
EgoPW [51]	189.6	105.3
SceneEgo [52]	118.5	92.75
EgoPW* [51]	90.96	64.33
SceneEgo* [52]	89.06	70.10
Ours-Single	64.19	50.06
Ours-Refined	**57.59**	**46.55**
EgoWholeBody test dataset		
EgoPW* [51]	84.21	63.02
SceneEgo* [52]	87.57	69.46
Ours-Single	66.28	43.14
Ours-Refined	**60.32**	**40.35**

Table 1. Egocentric human body pose accuracy of our method on SceneEgo test datasets and EgoWholeBody test dataset. Our method outperforms all previous state-of-the-art methods. * denotes the method trained with the datasets in Sec. 5.1.

5.3. Ablation Study

EgoWholeBody Dataset. Compared to existing egocentric datasets, our EgoWholeBody dataset contains diverse body

Method	MPJPE	PA-MPJPE
SceneEgo test dataset [52]		
Hand4Whole [34]	49.66	13.85
Ours-Single	23.63	9.59
Ours-Refined	**19.37**	**9.05**
EgoWholeBody test dataset		
Hand4Whole [34]	52.85	35.04
Ours-Single	33.10	19.68
Ours-Refined	**28.29**	**14.51**

Table 2. Egocentric hand pose accuracy of our method. Our method outperforms the Hand4Whole [34] on both datasets.

Method	MPJPE	PA-MPJPE
Body Pose Results		
w/o EgoWholeBody	75.10	58.62
w/o FisheyeViT	67.36	53.44
w/ Mo^2Cap^2 [54] head	87.47	65.10
w/ xR-egopose [47] head	116.5	95.78
w/ SceneEgo [52] head	77.73	62.69
Ours-Single	**64.19**	**50.06**
w/ GlobalEgoMocap†	69.83	56.73
w/o uncert. guidance†	62.16	48.40
Only body diffusion	58.95	47.03
Ours-Refined†	**57.59**	**46.55**
Hand Pose Results		
Only hand diffusion	21.69	9.24
Ours-Refined	**19.37**	**9.05**

Table 3. Ablation Study on SceneEgo test dataset [52]. † denotes the temporal-based method.

and hand motions, larger quantity of images, and higher image quality. We show this by training our body pose estimation network without our dataset, using the Mo^2Cap^2 [54] and EgoPW [51] training dataset. The results, labeled as "w/o EgoWholeBody" in Table 3, show that performance without the EgoWholeBody dataset is inferior to our proposed method. This highlights that training with our EgoWholeBody dataset enhances the performance of the pose estimation method. We also compare this result with existing methods on the SceneEgo test set (Table 1). Trained without EgoWholeBody, our approach still outperforms previous methods, showing the effectiveness of our method.

FisheyeViT and Pose Regressor with Pixel-Aligned 3D Heatmap. To assess the individual contributions of FisheyeViT and the pixel-aligned 3D heatmap in our single-frame pose estimation pipeline, we perform experiments to measure their impact on the overall performance. First, we substitute the FisheyeViT module in our single-frame pose estimation method to ViT [13]. The result is shown in "w/o FisheyeViT" in Table 3 and it is worse than our full method. This demonstrates the effectiveness of FisheyeViT in addressing fisheye distortion and feature extraction.

Next, we analyze the performance of the single-frame pose estimation network when substituting our pose regressor based on pixel-aligned 3D heatmap with the pose estimation heads of previous works [47, 52, 54]. The results of the three experiments, labeled as "w/ Mo^2Cap^2 head", "w/ xR-egopose head" and "w/ SceneEgo head", show a performance drop compared to our full method. This emphasizes the crucial role of the pixel-aligned 3D heatmap in accurately estimating egocentric 3D body joint positions.

Diffusion-based Motion Refinement. We assess the effectiveness of our diffusion-based motion refinement with the following experiments: First, we compare the performance of our diffusion-based motion refinement with GlobalEgoMocap [50] by applying the GlobalEgoMocap optimizer on the single-frame body pose estimation results. The result, labeled as "w GlobalEgoMocap" in Table 3, indicates that our refinement method outperforms GlobalEgoMocap.

Second, we remove the uncertainty-aware guidance in the motion refinement. Instead, we use fixed Gaussian denoising steps to refine the motion. The result "w/o uncert. guidance" in Table 3, shows that our uncertainty-aware refinement method performs better. Our approach relies on the uncertainty values for each joint, using low-uncertainty joints to guide the generation of high-uncertainty joints. This helps reduce errors in joint predictions caused by egocentric self-occlusion, leading to improved results.

Third, we replace our whole-body motion diffusion model with the separate human body and left/right-hand diffusion models and show the accuracy of refined body and hand motion in "Only body diffusion" and "Only hand diffusion" in Table 3. From the results, we observe improvements in the accuracy of motion refined by our whole-body diffusion method, proving that learning the whole-body motion prior can help both the refinement of the body and hand motion by learning the correlation between them.

6. Conclusion

In this work, we have introduced an innovative approach to capture egocentric whole-body human motion. Our method comprises a single-frame-based whole-body pose estimation process, which includes FisheyeViT and pixel-aligned 3D heatmap representations. To enhance the initial whole-body pose estimates, we have integrated an uncertainty-aware diffusion-based motion refinement technique. Our experimental results demonstrate that both our single-frame method and the temporal-based method surpass all existing state-of-the-art techniques in terms of both quality and accuracy. Looking ahead, we see the potential for extending the applications of FisheyeViT to other vision tasks involving fisheye cameras. Future work could also involve incorporating facial expressions in whole-body motion capture.

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: 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

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. 4 and Tab. 5. "*" 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. 4, 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	64.19	50.06
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	68.59	55.92
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	74.66	59.26
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	66.28	43.14
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^2Cap^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:

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

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

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.

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.

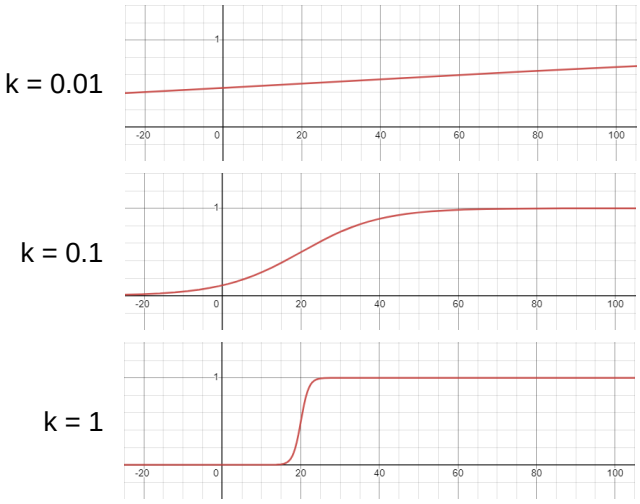

Figure 7. The weight function with different hyper-parameters k. The x-axis is the diffusion time step t and the y-axis is the weight $\mathbf{w}$.

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].

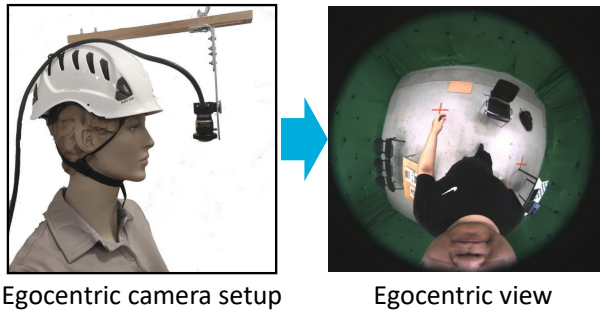

Figure 8. The setup of the egocentric fisheye camera and one example of the egocentric image.

20. More Visualization Results

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

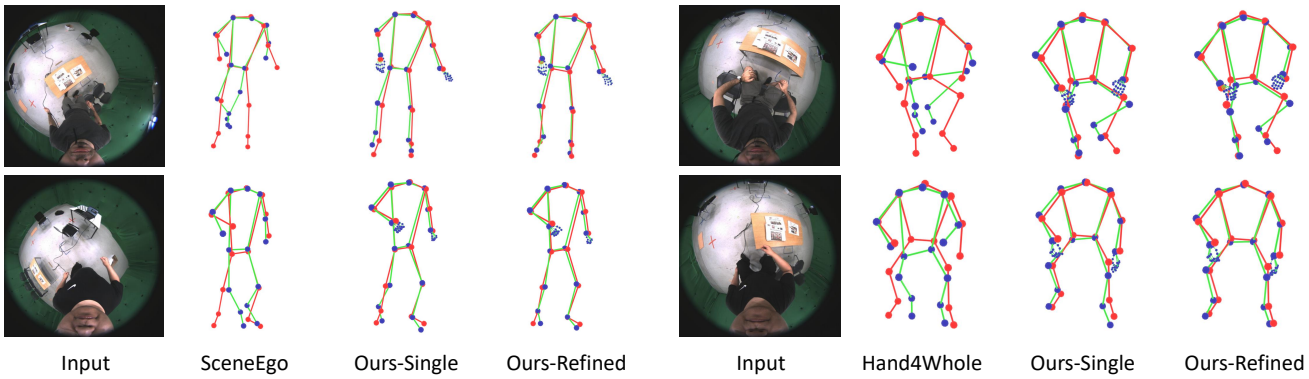

Figure 9. Qualitative comparison on human body pose estimations between our methods and the state-of-the-art SceneEgo [52] method. The red skeleton is the ground truth while the green skeleton is the predicted pose. Our methods predict more accurate body poses.

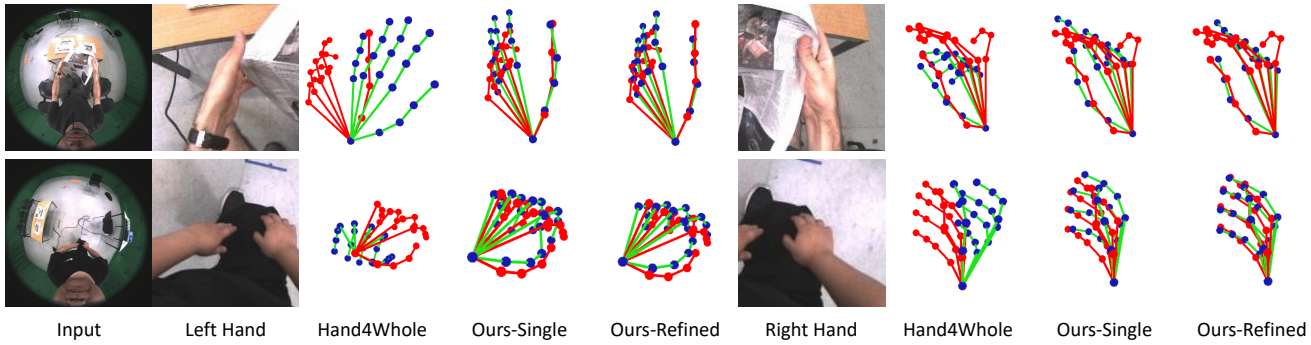

Figure 10. Qualitative comparison on hand pose estimation results. Our single-view and refined hand poses are more accurate than the poses from Hand4Whole [34] method. The red skeleton is the ground truth while the green skeleton is the predicted pose.