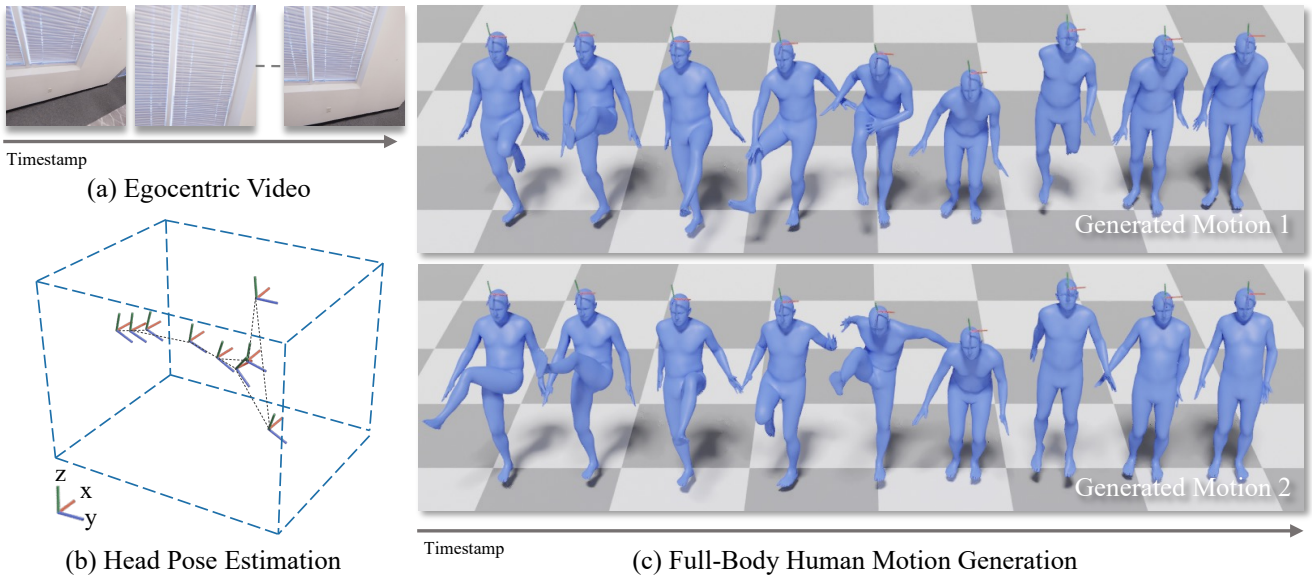
Ego-Body Pose Estimation via Ego-Head Pose Estimation

Jiaman Li C. Karen Liu† Jiajun Wu†
Stanford University
{jiamanli,karenliu,jiajunwu}@cs.stanford.edu



(a) Egocentric Video (b) Head Pose Estimation (c) Full-Body Human Motion Generation

Figure 1. Taking egocentric video (a) as input, our approach first predicts head pose (b); it then estimates multiple plausible full-body human motions (c) from the predicted head pose. This motion sequence shows a human kicking and jumping in place. Please refer to the supplementary video on our project page for a complete motion sequence visualization.

Abstract

Estimating 3D human motion from an egocentric video sequence plays a critical role in human behavior understanding and has various applications in VR/AR. However, naively learning a mapping between egocentric videos and human motions is challenging, because the user's body is often unobserved by the front-facing camera placed on the head of the user. In addition, collecting large-scale, high-quality datasets with paired egocentric videos and 3D human motions requires accurate motion capture devices, which often limit the variety of scenes in the videos to lab-like environments. To eliminate the need for paired egocentric video and human motions, we propose a new method, Ego-Body Pose Estimation via Ego-Head Pose Estimation (EgoEgo), which decomposes the problem into two stages, connected by the head motion as an intermediate representation. EgoEgo first integrates SLAM and a learning approach to estimate accurate head motion. Subsequently, leveraging the estimated head pose as input, EgoEgo utilizes conditional diffusion to generate multiple plausible full-body motions. This disentanglement of head and body pose eliminates the need for training datasets with paired egocentric videos and 3D human motion, enabling us to leverage large-scale egocentric video datasets and motion capture datasets separately. Moreover, for systematic benchmarking, we develop a synthetic dataset, AMASS-Replica-Ego-Syn (ARES), with paired egocentric videos and human motion. On both ARES and real data, our EgoEgo model performs significantly better than the current state-of-the-art methods.

1. Introduction

Estimating 3D human motion from an egocentric video, which records the environment viewed from the first-person perspective with a front-facing monocular camera, is critical to applications in VR/AR. However, naively learning a mapping between egocentric videos and full-body human motions is challenging for two reasons. First, modeling this complex relationship is difficult; unlike reconstruction motion from third-person videos, the human body is often out of view of an egocentric video. Second, learning this

† indicates equal contribution.

mapping requires a large-scale, diverse dataset containing paired egocentric videos and the corresponding 3D human poses. Creating such a dataset requires meticulous instrumentation for data acquisition, and unfortunately, such a dataset does not currently exist. As such, existing works have only worked on small-scale datasets with limited motion and scene diversity [22, 47, 48].

We introduce a generalized and robust method, **EgoEgo**, to estimate full-body human motions from only egocentric video for diverse scenarios. Our key idea is to use head motion as an intermediate representation to decompose the problem into two stages: head motion estimation from the input egocentric video and full-body motion estimation from the estimated head motion. For most day-to-day activities, humans have an extraordinary ability to stabilize the head such that it aligns with the center of mass of the body [13], which makes head motion an excellent feature for full-body motion estimation. More importantly, the decomposition of our method removes the need to learn from paired egocentric videos and human poses, enabling learning from a combination of large-scale, single-modality datasets (e.g., datasets with egocentric videos or 3D human poses only), which are commonly and readily available.

The first stage, estimating the head pose from an egocentric video, resembles the localization problem. However, directly applying the state-of-the-art monocular SLAM methods [33] yields unsatisfactory results, due to the unknown gravity direction and the scaling difference between the estimated space and the real 3D world. We propose a hybrid solution that leverages SLAM and learned transformer-based models to achieve significantly more accurate head motion estimation from egocentric video. In the second stage, we generate the full-body motion based on a diffusion model conditioned on the predicted head pose. Finally, to evaluate our method and train other baselines, we build a large-scale synthetic dataset with paired egocentric videos and 3D human motions, which can also be useful for future work on visuomotor skill learning and sim-to-real transfer.

Our work makes four main contributions. First, we propose a decomposition paradigm, **EgoEgo**, to decouple the problem of motion estimation from egocentric video into two stages: ego-head pose estimation, and ego-body pose estimation conditioned on the head pose. The decomposition lets us learn each component separately, eliminating the need for a large-scale dataset with two paired modalities. Second, we develop a hybrid approach for ego-head pose estimation, integrating the results of monocular SLAM and learning. Third, we propose a conditional diffusion model to generate full-body poses conditioned on the head pose. Finally, we contribute a large-scale synthetic dataset with both egocentric videos and 3D human motions as a test bed to benchmark different approaches and showcase that our method outperforms the baselines by a large margin.

2. Related Work

Motion Estimation from Third-person Video. 3D pose estimation from images and videos in third-person view has been extensively studied in recent years. There are mainly two typical categories in this direction. One is to regress joint positions directly from images and videos [25, 28, 36, 52]. The other adopts parametric human body model [20] to estimate body model parameters from images or videos [3, 12, 15–18, 21]. And recently, learned motion prior is applied to address the issues of jitters, lack of global trajectory, and missing joints or frames [19, 29, 46]. Moreover, physical constraints are enforced in motion estimation from videos [41, 49]. In contrast to third-person videos, where the full body can be seen, body joints are mostly not visible in an egocentric video, which poses a significant challenge for the problem. Although the body joints are unobserved from egocentric views, the visual information of how the environment changes provides a strong signal to infer how the head moves. In this work, we propose to use the head pose as an intermediate representation to bridge the egocentric video and full-body motions.

Motion Estimation from Egocentric Video. Growing attention is received in pose estimation from egocentric videos. Special hardware like the fisheye camera is deployed to predict full body pose from captured images [9, 35, 38, 42]. While body joints are usually visible in images captured with a fisheye camera, the distortion of images poses a significant challenge. Jiang et al. [8] deploy a standard camera to a human chest and propose an implicit motion graph matching approach to predict full body motions from the input video. You2Me [26] predicts full body motions by observing the interaction pose of the second-person in the camera view. Towards a goal of estimating and forecasting physically plausible motions from a head-mounted camera, EgoPose [47, 48] develop a Deep-RL framework to learn a control policy to estimate current poses and forecast future poses. Follow-up work Kinpoly [22] integrates kinematics and dynamics to predict physically plausible motions interacting with known objects. While their method achieves impressive results in their collected dataset, it cannot handle scenes and motions out of their data distribution. This work aims to establish a more generalized and robust framework to infer full-body motions from egocentric video only. To validate the effectiveness in more generalized scenes and motions, we also introduce an approach of synthesizing egocentric video corresponding to mocap data in diverse 3D scenes for quantitative evaluation.

Motion Estimation from Sparse Sensors. Instead of estimating motion from videos, some work explored reconstructing human motions from sparse sensor input. TransPose [45] proposes a real-time pipeline to predict full body motions from 6 IMU sensors, including head, torso, left/right arms,

and left/right lower legs. Follow-up work PIP [44] further includes a PD controller on top of the kinematic estimator to introduce physics constraints during reconstruction. TIP [11] follows the same sensor setting and deploys a transformer-based model to leverage IMU sequential information effectively. Fewer sensors are investigated in LoBSTr [43]. Given tracker information from 4 joints (head, left/right hands, torso), they present an RNN-based model to infer lower-body motions from past upper-body joint signals. Recent advances [1, 4, 10, 40] further relax the input constraints to head and hand signals only. In this work, we do not rely on any observations from inertial sensors. Instead, we aim to develop a solution with egocentric video input only.

3. Method

Our method, EgoEgo, estimates 3D human motion from a monocular egocentric video sequence. As shown in Figure 2, our key idea is to leverage *head motion*: first estimating head motion from egocentric video, and then estimating full body motion from head motion. We show that head motion is an excellent feature for full-body motion estimation and a compact, intermediate representation that reduces the challenge into two much simpler sub-problems. Such a disentanglement also allows us to leverage a large-scale egocentric video dataset with head motion (but no full body motion) in stage one, and a separate 3D human motion dataset (but no egocentric videos) in stage two.

Notations. We denote full body motion as $\boldsymbol{X} \in \mathbb{R}^{T\times D}$ and egocentric images captured from a front-facing, head-mounted camera as $\boldsymbol{I} \in \mathbb{R}^{T\times h\times w\times 3}$, where T is the sequence length, D is the dimension of the pose state, and $h \times w$ is the size of an image. We introduce head motion $\boldsymbol{H} \in \mathbb{R}^{T\times D'}$ as an intermediate representation to bridge the input egocentric video and the output human motions, where D' is the dimension of the head pose.

3.1. Head Pose Estimation from Egocentric Video

Estimating the head motion from an egocentric video can be viewed as a camera localization problem. However, we observed three issues that prevent us from directly applying the state-of-the-art monocular SLAM method [33] to our problem. First, the gravity direction of the estimated head pose is unknown. Thus, the results cannot be directly fed to the full-body motion estimator, since it expects the head pose expressed in a coordinate frame where the gravity direction is $[0, 0, -1]^T$. Second, the estimated translation by monocular SLAM is not to scale when compared with the distance in the real world. Third, monocular SLAM tends to be less accurate in estimating relative head rotation than translation.

Based on these observations, we propose a hybrid method that leverages SLAM and learned models to achieve more accurate head pose estimation than the state-of-the-art SLAM alone. First, we develop a transformer-based model GravityNet to estimate the gravity direction from the rotation and the translation trajectories computed by SLAM. We rotate the SLAM translation by aligning the estimated gravity direction with the real gravity direction $[0, 0, -1]^T$ in the 3D world. Moreover, from the optical flow features extracted from the egocentric video, our method learns a model, HeadNet, to estimate head rotations and translation distance. The predicted translation distance of HeadNet is used to re-scale the translation estimated by SLAM. The predicted head rotation by HeadNet is directly used to replace the rotation estimated by SLAM. Figure 2 summarizes our process to generate head poses.

Monocular SLAM. We adopt DROID-SLAM [33] to estimate camera trajectory from egocentric videos. DROID-SLAM [33] is a learning-based method to estimate camera pose trajectory and reconstruct the 3D map of the environment simultaneously. By a design of recurrent iterative updates to camera pose and depth, it showcases superior and more robust results compared to prior SLAM systems [2]. For more details, please refer to [33].

Gravity Direction Estimation. We introduce GravityNet to predict gravity direction $\boldsymbol{g} \in \mathbb{R}^3$ from a sequence of head poses $\hat{\boldsymbol{h}}_1, \hat{\boldsymbol{h}}_2, ..., \hat{\boldsymbol{h}}_T$. The gravity direction $\boldsymbol{g}$ is represented by a unit vector. The head poses input $\hat{\boldsymbol{h}}_t$ consists of a 3D head translation, a head rotation represented by a continuous 6D rotation vector [53], head translation difference, and head rotation difference computed by $\boldsymbol{O}_{t-1}^{-1}\boldsymbol{O}_t$ where $\boldsymbol{O}_t$ denotes the head rotation matrix at time step t. We adopt a transformer-based architecture [37] consisting of two self-attention blocks, each of which has a multi-head attention layer followed by a position-wise feed-forward layer. We take the first output of the transformer and feed it to an MLP to predict the gravity direction $\boldsymbol{g}$. We train our GravityNet on the large-scale motion capture dataset AMASS [23]. However, the motion sequences in AMASS have the correct gravity direction $\boldsymbol{g}_c = [0, 0, -1]^T$. To emulate the distribution of the predicted head poses from monocular SLAM, we apply a random scale and a random rotation to the head poses in each AMASS sequence to generate our training data for gravity estimation. L_1 loss for the gravity vector is used during training. Based on the prediction of GravityNet, we compute the rotation matrix $\boldsymbol{R}_g$ to align the prediction $\boldsymbol{g}$ and $\boldsymbol{g}_c$. Then we apply $\boldsymbol{R}_g$ to the SLAM translation denoted as $\hat{\boldsymbol{P}}_1, \hat{\boldsymbol{P}}_2, ..., \hat{\boldsymbol{P}}_T$ and get $\boldsymbol{P}_1, \boldsymbol{P}_2, ..., \boldsymbol{P}_T$, where $\boldsymbol{P}_t = \boldsymbol{R}_g\hat{\boldsymbol{P}}_t$.

Head Pose Estimation. We propose HeadNet to predict a sequence of distance $d_1, d_2, ..., d_T$ and head rotations $\boldsymbol{R}_1, ..., \boldsymbol{R}_T$ from a sequence of optical flow features $\boldsymbol{o}_1, ..., \boldsymbol{o}_T$. The optical flow features are extracted by a pre-trained ResNet-18 [6]. We deploy the same model architecture as GravityNet. Since the scale from the monocular SLAM system may not be consistent with the real

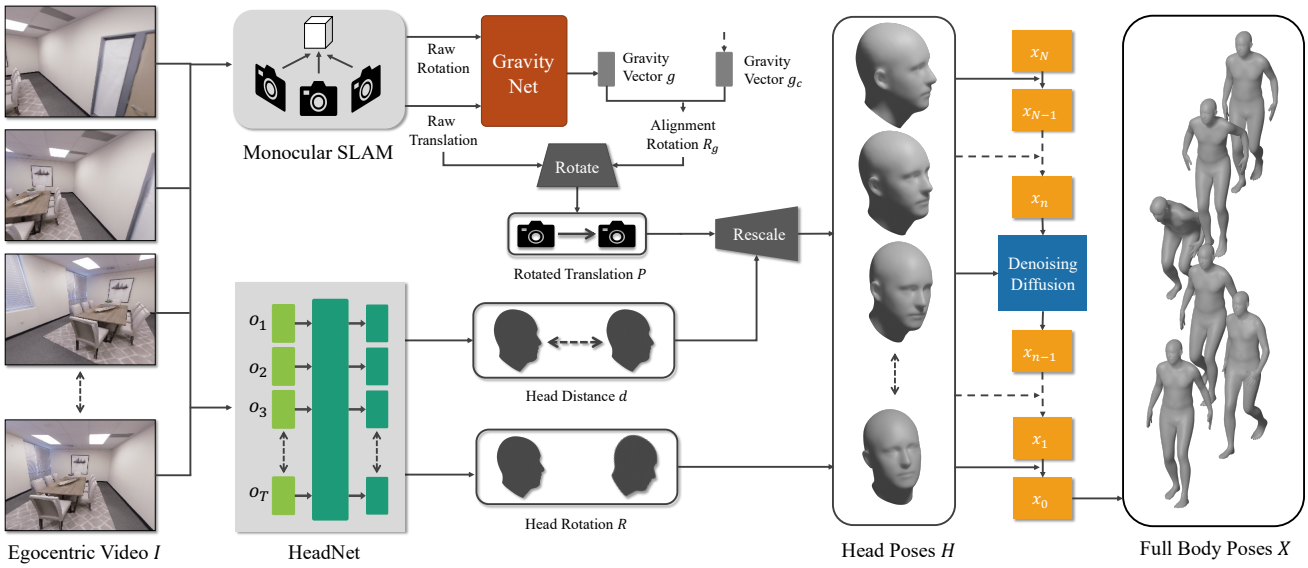


Figure 2. Overview of EgoEgo. The model first takes an egocentric video as input and predicts the head pose with a hybrid approach that combines monocular SLAM and the learned GravityNet and HeadNet. The predicted head pose is then fed to a conditional diffusion model to generate the full-body pose.

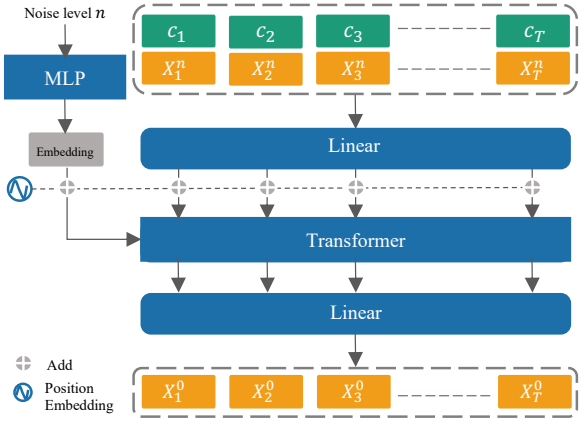


Figure 3. Model architecture of the denoising network in a single step of the reverse diffusion process.

3D world where human moves, we use HeadNet to predict the vector norm of the translation difference between consecutive time steps denoted as $d_1, ..., d_T$, where d_t represents a scalar value. For each camera translation sequence produced by monocular SLAM and rotated by aligning gravity direction, given the camera translation trajectory $\boldsymbol{P} \in \mathbb{R}^{T\times 3}$, we calculate the distance d_t^s between $\boldsymbol{P}_t$ and $\boldsymbol{P}_{t+1}$ as $d_t^s = ||\boldsymbol{P}_{t+1} - \boldsymbol{P}_t||_2$. We take the mean value for the sequence of distances as $d^s = \frac{1}{T}\sum_{t=1}^{T} d_t^s$. Similarly, we compute the mean of the predicted distance sequence $d = \frac{1}{T}\sum_{t=1}^{T} d_t$. The scale is calculated as $s = \frac{d}{d^s}$. We multiply scale s to the predicted translation $\boldsymbol{P}$ and use $s\boldsymbol{P}$ as our global head translation results.

The network also predicts the head angular velocity, $\boldsymbol{\omega}_1, ..., \boldsymbol{\omega}_T$, in the head frame. We integrate predicted angular velocity to generate corresponding rotations $\boldsymbol{R}_1, ..., \boldsymbol{R}_T$. During inference, we assume that the first head orientation is given and integrate the predicted head angular velocity to estimate the subsequent head orientations.

The training loss of the HeadNet is defined as: $\mathcal{L} = \mathcal{L}_{dist} + \mathcal{L}_{vel} + \mathcal{L}_{rot}$. $\mathcal{L}_{dist}$ represents the $L1$ loss for translation distance. $\mathcal{L}_{vel}$ represents the $L1$ loss for angular velocity. $\mathcal{L}_{rot}$ denotes the rotation loss $\mathcal{L}_{rot} = ||\boldsymbol{R}_{pred}\boldsymbol{R}_{gt}^T - \boldsymbol{I}||_1$ where $\boldsymbol{R}_{pred}$ represents the integrated rotation using predicted angular velocity, $\boldsymbol{R}_{gt}$ represents the ground truth rotation matrix and $\boldsymbol{I}$ represents the identity matrix.

3.2. Full-Body Pose Estimation from Head Pose

Predicting full-body pose from head pose is not a one-to-one mapping problem as different full-body motions may have the same head pose. Thus, we formulate the task using a conditional generative model. Inspired by the recent success of the diffusion model in image generation [30], we deploy a diffusion model to generate full-body poses conditioned on head poses. We use the formulation proposed in the denoising diffusion probabilistic model (DDPM) [7], which has also been applied in some concurrent work [14, 34, 50] for motion generation and motion interpolation tasks. We will first introduce our data representation and then detail the conditional diffusion model formulation.

A body pose $\boldsymbol{X}_t \in \mathbb{R}^D$ at time t consists of the global joint position ($\mathbb{R}^{J\times 3}$) and global joint rotations ($\mathbb{R}^{J\times 6}$). We adopt the widely used SMPL model [20] as our skeleton, and the number of joints J is 22. For the convenience of notation in the diffusion model, we use $\boldsymbol{x}_n$ to denote a sequence of body poses $\boldsymbol{X}_1^n, \boldsymbol{X}_2^n, ..., \boldsymbol{X}_T^n$ at noise level n.

The high-level idea of the diffusion model is to design a forward diffusion process to add Gaussian noises to the

original data with a known variance schedule and learn a denoising model to gradually denoise N steps given a sampled $\boldsymbol{x}_N$ from a normal distribution to generate $\boldsymbol{x}_0$.

Specifically, diffusion models consist of a forward diffusion process and a reverse diffusion process. The forward diffusion process gradually adds Gaussian noise to the original data $\boldsymbol{x}_0$. And it is formulated using a Markov chain of N steps as shown in Equation 1:

$$q(\boldsymbol{x}_{1:N}|\boldsymbol{x}_0) := \prod_{n=1}^{N} q(\boldsymbol{x}_n|\boldsymbol{x}_{n-1}). \tag{1}$$

Each step is decided by a variance schedule using β_n and is defined as

$$q(\boldsymbol{x}_n|\boldsymbol{x}_{n-1}) := \mathcal{N}(\boldsymbol{x}_n; \sqrt{1-\beta_n}\boldsymbol{x}_{n-1}, \beta_n \boldsymbol{I}). \tag{2}$$

To generate full-body motion conditioned on the head pose, we need to reverse the diffusion process. The reverse process can be approximated as a Markov chain with a learned mean and fixed variance:

$$p_\theta(\boldsymbol{x}_{n-1}|\boldsymbol{x}_n, c) := \mathcal{N}(\boldsymbol{x}_{n-1}; \boldsymbol{\mu}_\theta(\boldsymbol{x}_n, n, c), \sigma_n^2 \boldsymbol{I}). \tag{3}$$

where $\boldsymbol{\theta}$ represents the parameters of a neural network, c is the head conditions. The learned mean $\boldsymbol{\mu}_\theta(\boldsymbol{x}_n, n, c)$ (we use $\boldsymbol{\mu}_\theta$ in the equation for brevity) can be represented as follows where α_n and $\bar{\alpha}_n$ are fixed parameters, $\hat{\boldsymbol{x}}_\theta(\boldsymbol{x}_n, n, c)$ is the prediction of $\boldsymbol{x}_0$:

$$\boldsymbol{\mu}_\theta = \frac{\sqrt{\alpha_n}(1-\bar{\alpha}_{n-1})\boldsymbol{x}_n + \sqrt{\bar{\alpha}_{n-1}}(1-\alpha_n)\hat{\boldsymbol{x}}_\theta(\boldsymbol{x}_n, n, c)}{1-\bar{\alpha}_n} \tag{4}$$

Learning the mean can be reparameterized as learning to predict the original data $\boldsymbol{x}_0$. The training loss is defined as a reconstruction loss of $\boldsymbol{x}_0$:

$$\mathcal{L} = \mathbb{E}_{\boldsymbol{x}_0, n}||\hat{\boldsymbol{x}}_\theta(\boldsymbol{x}_n, n, c) - \boldsymbol{x}_0||_1 \tag{5}$$

As shown in Figure 3, in denoising step n, we concatenate head pose condition $\boldsymbol{c}_1, ..., \boldsymbol{c}_T$ with body pose representation $\boldsymbol{X}_1^n, ..., \boldsymbol{X}_T^n$ at noise level n, combined with noise embedding as input to a transformer model, and estimate $\boldsymbol{x}_0$.

3.3. Synthetic Data Generation

Our method does not need paired training data. Still, for benchmarking purposes, we develop a way to automatically synthesize a large-scale dataset with various paired egocentric videos and human motions.

Generate Motions in 3D Scenes. To generate a dataset with both egocentric video and ground truth human motions, we use a large-scale motion capture dataset AMASS [23] and a 3D scene dataset Replica [31]. We convert the scene mesh from Replica to the signed distance field (SDF) for the penetration calculation. We divide each sequence of AMASS [23] into sub-sequences with 150 frames. For each sub-sequence, based on the semantic annotation provided by Replica [31], we place the first pose in a random location with the feet in contact with the floor. Then we calculate penetration loss following Wang et al. [39] for each pose in this sequence. We empirically set the threshold to 2 and only keep the poses with penetration loss less than the threshold. Specifically, for human mesh $\boldsymbol{M}_t$ at time t represented by a parameterized human model [20, 27], we denote d_i as the signed distance of vertex i. The penetration loss is then defined as $L_{pen}^t = \sum_{d_i<0} ||d_i||$.

Synthesize Realistic Egocentric Images. The motion sequences produced by detecting penetration with 3D scenes provide the camera pose trajectories to render synthetic egocentric videos. AI Habitat [24,32] is a platform for embodied agent research that supports fast rendering given a camera trajectory and a 3D scene. We feed the head pose trajectories to the platform and synthesize realistic images in the egocentric view. We generate 1,664,616 frames with 30 fps, approximately 15 hours of motion in 18 scenes. We name the synthetic dataset AMASS-Replica-Ego-Syn (ARES) and show some examples from our synthetic dataset in Figure 4.

4. Experiments

We evaluate and compare our method to baselines on five commonly used metrics for human motion reconstruction, in addition to the human perception studies. We also conduct ablation studies to analyze the performance of each stage of our method, as well as the design choices in our model.

4.1. Datasets and Evaluation Metrics

AMASS-Replica-Ego-Syn (ARES) is our synthetic dataset which contains synthetic egocentric videos and ground truth motions. ARES contains about 15 hours of motion across 18 scenes. We remove 5 scenes from training as unseen scenes. The training dataset consists of about 1.2M frames in 13 different scenes. The testing dataset contains 34, 850 frames from 5 unseen scenes.

AMASS [23] is a large-scale motion capture dataset with about 45 hours of diverse motions. We split training and testing data following HuMoR [29].

Kinpoly-MoCap [22] consists of egocentric videos captured using a head-mounted camera and corresponding 3D motions captured with motion capture devices. The total motion is about 80 minutes long. Since it uses motion capture devices, the egocentric video is constrained to a single lab scene.

Kinpoly-RealWorld [22] contains paired egocentric videos and head poses captured using iPhone ARKit. Unlike Kinpoly-MoCap which is captured in a lab scene, Kinpoly-RealWorld provides in-the-wild egocentric videos.

GIMO [51] consists of egocentric video, eye gaze, 3D motions, and scanned 3D scenes. This dataset is collected using

Figure 4. Illustration of our ARES Dataset. "Ego-View" represents the synthetic egocentric images using our proposed data generation pipeline. We also provide a third-person view reference in the second row. The left and right show two sequences from different scenes.

	ARES					Kinpoly-MoCap [22]					GIMO [51]				
Method	$\mathbf{O}_{head}$	$\mathbf{T}_{head}$	MPJPE	Accel	FS	$\mathbf{O}_{head}$	$\mathbf{T}_{head}$	MPJPE	Accel	FS	$\mathbf{O}_{head}$	$\mathbf{T}_{head}$	MPJPE	Accel	FS
PoseReg [48]	0.77	354.7	147.7	127.6	87.1	1.05	1943.9	160.4	61.8	10.8	1.51	1528.6	189.3	71.5	14.2
Kinpoly-OF [22]	0.62	323.4	141.6	7.3	4.2	1.33	2475.5	230.5	16.4	15.8	1.52	1739.3	404.2	21.9	14.4
EgoEgo (ours)	**0.20**	**148.0**	**121.1**	**6.2**	**2.7**	**0.58**	**505.1**	**125.9**	**8.0**	**1.6**	**0.67**	**356.8**	**152.1**	**10.4**	**1.9**

Table 1. Full-body motion estimation from egocentric video on ARES, Kinpoly-MoCap [22], and GIMO [51].

Hololens, iPhone 12, and IMU-based motion capture suits to study motion prediction tasks guided by eye gaze. We use 15 scenes for training and 4 scenes for testing.

Evaluation Metrics.

- **Head Orientation Error** $(\mathbf{O}_{head})$ computes the Frobenius norm of the difference between the 3×3 rotation matrix $||\boldsymbol{R}_{pred}\boldsymbol{R}_{gt}^{-1} - \boldsymbol{I}||_2$, where $\boldsymbol{R}_{pred}$ is the predicted head rotation matrix and $\boldsymbol{R}_{gt}$ is the ground truth head rotation matrix.
- **Head Translation Error** $(\mathbf{T}_{head})$ is computed by taking the mean Euclidean distance of two trajectories. We use this metric to measure the head joint translation errors in millimeters (mm).
- **MPJPE** represents mean per-joint position errors in millimeters (mm).
- **Accel** represents the difference of acceleration between predicted joint positions and ground truth joint positions measured in $\left(mm/s^2\right)$.
- **FS** represents foot skating metric and is computed following NeMF [5]. Specifically, we first project the toe and ankle joints' velocity to the xy plane and compute the $L1$ norm of the projected velocities in each step denoted as v_t. We only accumulate the horizontal translation for those steps that have a height h_t lower than a specified threshold H. And the metric is calculated as a mean of weighted values $v_t(2 - 2^{\frac{h_t}{H}})$ across the sequence and is measured in (mm).

4.2. Body Pose Estimation from Egocentric Video

Training Data. We train our HeadNet using paired egocentric videos and head poses provided by ARES, Kinply-RealWorld, and GIMO. Note that the body motion in these datasets is not used for training HeadNet. Our GravityNet and the conditional diffusion model are both trained on AMASS. For the baselines below, we train them using paired egocentric videos and ground truth motions in ARES.

Baselines. We compare our approach with two baselines PoseReg [48] and Kinpoly [22]. **PoseReg** [48] takes a sequence of optical flow features as input and uses an LSTM model to predict the pose state at each time step. The pose state consists of root translation, root orientation, joint rotations, and corresponding velocities including root linear velocity and angular velocities of all the joints. **Kinpoly-OF** [22] proposes a per-step regression model to estimate full body motion from optical flow features. Because our problem only allows for egocentric video as input, we choose the option of Kinpoly which only has optical flow features as input, without relying on ground truth head poses and action labels that depend on additional knowledge.

Results. We compare the complete pipeline of EgoEgo with baseline methods PoseReg [48] and Kinpoly-OF [22] on ARES, Kinpoly-MoCap [22] and GIMO [51], as shown in Table 1. We show that our EgoEgo outperforms all the baselines by a large margin on all three datasets. We show qualitative results in Figure 5. Our generated motions better preserve the root trajectories. And our approach can also generate more dynamic and realistic motions compared to the baselines.

4.3. Head Pose Estimation from Egocentric Video

Baselines. We compare our hybrid approach with the prediction results of DROID-SLAM [33]. For a fair comparison, we apply a rotation to the SLAM trajectory by aligning the first predicted head pose of SLAM with the ground truth head pose. We train our GravityNet on AMASS training split. As for HeadNet, we train on ARES, Kinpoly-RealWorld, and GIMO separately for the evaluation of different datasets.

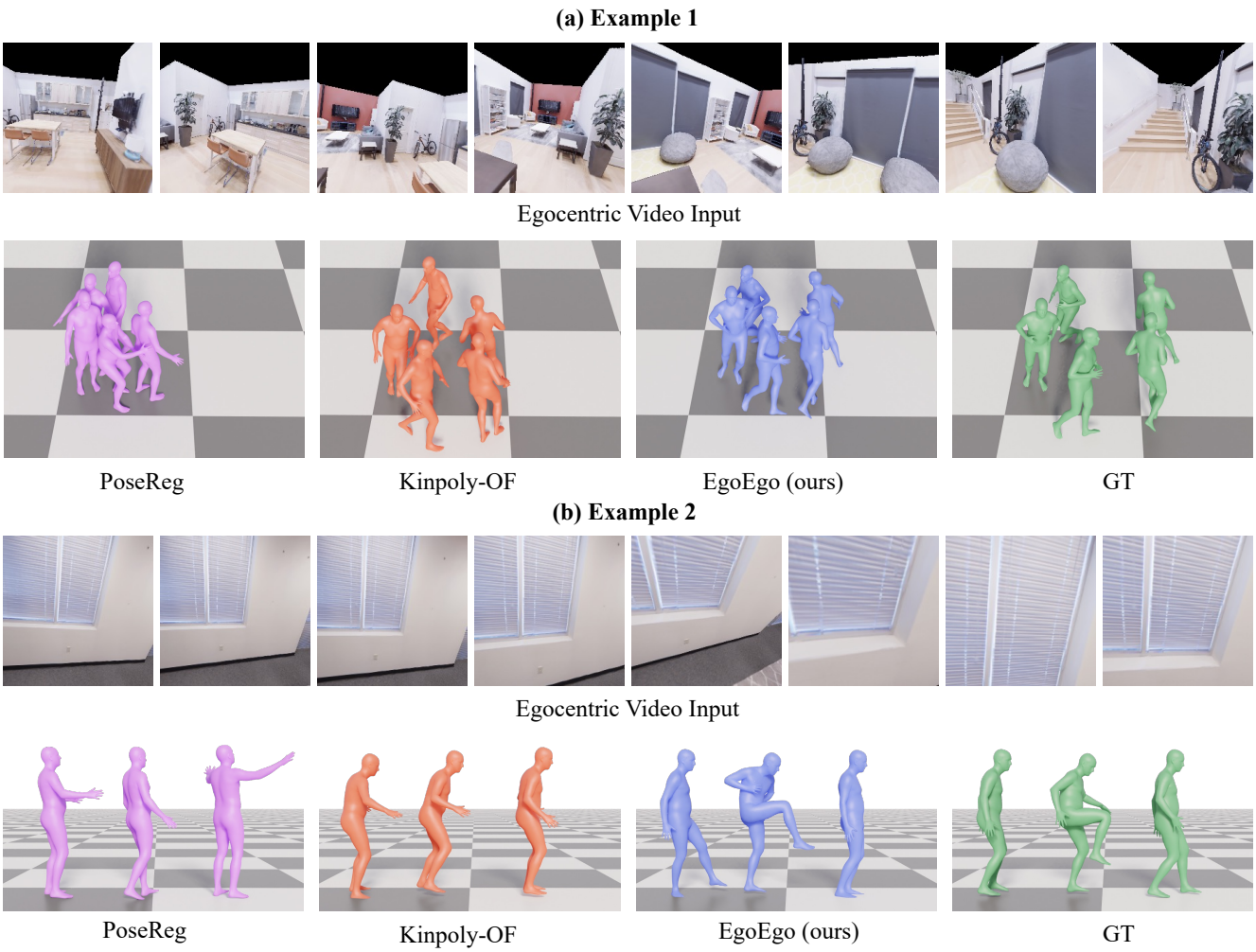


Figure 5. Qualitative results comparisons. We show the results of two egocentric videos from different scenes.

Results. We evaluate the head pose estimation on the three datasets as shown in Table 2. We show more accurate head rotation prediction results on ARES and comparable results on real-captured data. As the real-captured data is limited in scale (Kinpoly-RealWorld contains 20 minutes of training videos and GIMO contains 30 minutes of training videos), we believe the head rotation prediction can be further improved by future developments of large-scale real-captured datasets with head poses. Overall, our hybrid approach combines the accurate rotation prediction from HeadNet and re-scaled translation of gravity-aligned SLAM results, and produces more accurate head pose estimation results as input to the second stage.

4.4. Body Pose Estimation from the Head Pose

Baselines. We compare our conditional diffusion model for full-body pose estimation with two baselines AvatarPoser [10] and Kinpoly-Head [22]. **AvatarPoser** [10] takes both head and hand pose as input to predict full body motion.

	DROID-SLAM [33]		Ours	
	$\mathbf{O}_{head}$	$\mathbf{T}_{head}$	$\mathbf{O}_{head}$	$\mathbf{T}_{head}$
ARES	0.62	411.3	**0.23**	**176.5**
Kinpoly-MoCap	**0.55**	1290.8	0.58	**487.8**
GIMO	**0.67**	865.4	0.68	**304.7**

Table 2. Head pose estimation on test sets.

We remove the hand poses from the input and modified it to a setting with head pose input only. **Kinpoly-Head** [22] is our modified variant of the Kinpoly model that only takes the head pose as input. Both the baselines and our method are trained on the training split of AMASS with high-quality motion capture data.

Results. We evaluate the baselines and our method on the AMASS test set, as shown in Table 3. Since our model is generative, there are multiple plausible predictions from the same head pose input. For a quantitative comparison, we generate 200 samples for each head pose input and use the

Method	$\mathbf{O}_{head}$	$\mathbf{T}_{head}$	MPJPE	Accel	FS
AvatarPoser [10]	0.19	**28.6**	124.7	16.1	18.8
Kinpoly-Head [22]	0.19	87.8	110.9	11.4	11.2
Diffusion Model (ours)	**0.04**	36.7	**109.0**	**10.5**	**4.4**

Table 3. Full-body motion estimation from GT head pose on AMASS testing dataset [22].

	ARES		Kinpoly-MoCap		GIMO	
	$\mathbf{O}_{head}$	$\mathbf{T}_{head}$	$\mathbf{O}_{head}$	$\mathbf{T}_{head}$	$\mathbf{O}_{head}$	$\mathbf{T}_{head}$
SLAM	0.62	411.33	**0.55**	1290.82	**0.67**	865.41
SLAM+S	0.62	325.95	0.55	643.45	0.67	569.48
SLAM+S+G	0.62	176.54	0.55	487.77	0.67	304.74
Full model	**0.23**	**176.54**	0.58	**487.77**	0.68	**304.74**

Table 4. Ablation study for the components in head pose estimation. S represents the scale predicted by HeadNet, and G represents GravityNet.

one with the smallest MPJPE as our result.

4.5. Ablation Studies

Effects of Components in Head Pose Estimation. We study the effects of each component in head pose estimation in Table 4. We showcase that the rotation for aligning gravity direction and the learned scale are both effective to improve head translation results.

Effects of Head Pose in Full-Body Pose Estimation. We compare the full-body pose estimation results that take our predicted head poses and the ground truth head poses as input. Table 5 shows that the ground truth head poses significantly improve the full body pose estimation, indicating that by developing methods that predict more accurate head pose, the full body pose estimation can be further improved.

4.6. Human Perceptual Study

We also conduct two human perceptual studies as part of the evaluation. The first is to evaluate the quality of predicted full-body motion from egocentric video, the second is to evaluate the quality of predicted full-body motion from ground truth head poses. In both studies, we compare four types of motion: results from our EgoEgo and from two baselines, as well as the ground truth. For the first human study, each time, users are presented with two motions and an egocentric video, and asked to select which one is more plausible. For the second human study, users are presented with two motions and asked to select which one looks more natural and realistic. Because there are 10 examples and the two motions can be from four sources, we have 60 questions for each study. Each question was answered by 20 Amazon Mechanical Turk workers.

As shown in Figure 6(a), for full-body estimation from egocentric video, our results are preferred by 98% and 69% of workers when compared to the baselines. Also, when compared with the ground truth, 34% of the responses prefer our results (note that a perfect output would achieve 50%), suggesting that people cannot easily distinguish our results from ground truth motions. As shown in Figure 6(b), for full-body estimation from head poses, our results are preferred by 88% and 79% of workers when compared to the baselines.

	ARES		Kinpoly-MoCap		GIMO	
	EE	EE w/ GT	EE	EE w/ GT	EE	EE w/ GT
$\mathbf{O}_{head}$	0.20	**0.04**	0.58	**0.03**	0.67	**0.06**
$\mathbf{T}_{head}$	148.0	**29.1**	505.1	**60.7**	356.8	**66.0**
MPJPE	121.1	**105.6**	125.9	**76.0**	152.1	**125.7**
Accel	**6.2**	6.3	8.0	**7.2**	10.4	**10.2**
FS	**2.7**	3.0	**1.6**	2.5	1.9	**1.7**

Table 5. Ablation study for the effects of head pose to full-body pose estimation. EE represents EgoEgo. GT represents ground truth head poses.

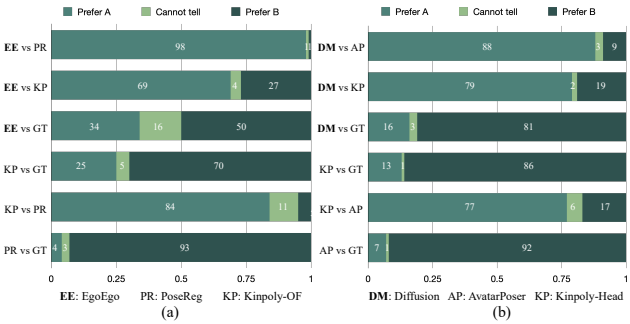


Figure 6. Results of human perceptual studies. The numbers shown in the chart represent %.

5. Conclusion

We presented a generalized framework to estimate full-body motions from egocentric video. The key is to decompose the problem into two stages. We predicted the head pose from an egocentric video and fed the output from the first stage to estimate full-body motions in the second stage. In addition, we developed a hybrid solution to produce more accurate head poses on top of monocular SLAM. We also proposed a conditional diffusion model to generate diverse high-quality full-body motions from predicted head poses. To benchmark different methods in a large-scale dataset, we proposed a data generation pipeline to synthesize a large-scale dataset with paired egocentric videos and 3D human motions. We showcased superior results on both the synthetic and the real-captured dataset compared to prior work.

Acknowledgement. We thank Zhengfei Kuang for the help with visualizations and Jonathan Tseng for discussions about the diffusion model. This work is in part supported by ONR MURI N00014-22-1-2740, NSF CCRI 2120095, the Toyota Research Institute (TRI), the Stanford Institute for Human-Centered AI (HAI), Innopeak, Meta, and Samsung.

References

[1] Sadegh Aliakbarian, Pashmina Cameron, Federica Bogo, Andrew Fitzgibbon, and Thomas J Cashman. FLAG: Flow-based 3D avatar generation from sparse observations. In *CVPR*, 2022. 3

[2] Carlos Campos, Richard Elvira, Juan J Gómez Rodríguez, José MM Montiel, and Juan D Tardós. ORB-SLAM3: An accurate open-source library for visual, visual–inertial, and multimap SLAM. *IEEE Transactions on Robotics*, 37(6):1874–1890, 2021. 3

[3] Hongsuk Choi, Gyeongsik Moon, Ju Yong Chang, and Kyoung Mu Lee. Beyond static features for temporally consistent 3D human pose and shape from a video. In *CVPR*, 2021. 2

[4] Andrea Dittadi, Sebastian Dziadzio, Darren Cosker, Ben Lundell, Thomas J Cashman, and Jamie Shotton. Full-body motion from a single head-mounted device: Generating SMPL poses from partial observations. In *ICCV*, 2021. 3

[5] Chengan He, Jun Saito, James Zachary, Holly Rushmeier, and Yi Zhou. NeMF: Neural motion fields for kinematic animation. In *NeurIPS*, 2022. 6

[6] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 3

[7] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020. 4

[8] Hao Jiang and Kristen Grauman. Seeing invisible poses: Estimating 3D body pose from egocentric video. In *CVPR*, 2017. 2

[9] Hao Jiang and Vamsi Krishna Ithapu. Egocentric pose estimation from human vision span. In *ICCV*, 2021. 2

[10] Jiaxi Jiang, Paul Streli, Huajian Qiu, Andreas Fender, Larissa Laich, Patrick Snape, and Christian Holz. AvatarPoser: Articulated full-body pose tracking from sparse motion sensing. In *ECCV*, 2022. 3, 7, 8

[11] Yifeng Jiang, Yuting Ye, Deepak Gopinath, Jungdam Won, Alexander W Winkler, and C Karen Liu. Transformer Inertial Poser: Attention-based real-time human motion reconstruction from sparse imus. In *SIGGRAPH ASIA*, 2022. 3

[12] Angjoo Kanazawa, Michael J Black, David W Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *CVPR*, 2018. 2

[13] EA Keshner and BW Peterson. Motor control strategies underlying head stabilization and voluntary head movements in humans and cats. *Progress in Brain Research*, 76:329–339, 1988. 2

[14] Jihoon Kim, Jiseob Kim, and Sungjoon Choi. FLAME: Free-form language-based motion synthesis & editing. In *AAAI*, 2023. 4

[15] Muhammed Kocabas, Nikos Athanasiou, and Michael J Black. VIBE: Video inference for human body pose and shape estimation. In *CVPR*, 2020. 2

[16] Muhammed Kocabas, Chun-Hao P Huang, Otmar Hilliges, and Michael J Black. PARE: Part attention regressor for 3D human body estimation. In *ICCV*, 2021. 2

[17] Muhammed Kocabas, Chun-Hao P Huang, Joachim Tesch, Lea Müller, Otmar Hilliges, and Michael J Black. SPEC: Seeing people in the wild with an estimated camera. In *ICCV*, 2021. 2

[18] Nikos Kolotouros, Georgios Pavlakos, Michael J Black, and Kostas Daniilidis. Learning to reconstruct 3D human pose and shape via model-fitting in the loop. In *ICCV*, 2019. 2

[19] Jiaman Li, Ruben Villegas, Duygu Ceylan, Jimei Yang, Zhengfei Kuang, Hao Li, and Yajie Zhao. Task-generic hierarchical human motion prior using vaes. In *3DV*, 2021. 2

[20] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: A skinned multi-person linear model. *ACM Transactions on Graphics (TOG)*, 34(6):1–16, 2015. 2, 4, 5

[21] Zhengyi Luo, S Alireza Golestaneh, and Kris M Kitani. 3D human motion estimation via motion compression and refinement. In *ACCV*, 2020. 2

[22] Zhengyi Luo, Ryo Hachiuma, Ye Yuan, and Kris Kitani. Dynamics-regulated kinematic policy for egocentric pose estimation. In *NeurIPS*, 2021. 2, 5, 6, 7, 8

[23] Naureen Mahmood, Nima Ghorbani, Nikolaus F Troje, Gerard Pons-Moll, and Michael J Black. AMASS: Archive of motion capture as surface shapes. In *ICCV*, 2019. 3, 5

[24] Manolis Savva*, Abhishek Kadian*, Oleksandr Maksymets*, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, Devi Parikh, and Dhruv Batra. Habitat: A platform for embodied AI research. In *ICCV*, 2019. 5

[25] Dushyant Mehta, Srinath Sridhar, Oleksandr Sotnychenko, Helge Rhodin, Mohammad Shafiei, Hans-Peter Seidel, Weipeng Xu, Dan Casas, and Christian Theobalt. VNect: Real-time 3D human pose estimation with a single RGB camera. *ACM Transactions on Graphics*, 36(4):1–14, 2017. 2

[26] Evonne Ng, Donglai Xiang, Hanbyul Joo, and Kristen Grauman. You2me: Inferring body pose in egocentric video via first and second person interactions. In *CVPR*, 2020. 2

[27] Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed A. A. Osman, Dimitrios Tzionas, and Michael J. Black. Expressive body capture: 3D hands, face, and body from a single image. In *CVPR*, 2019. 5

[28] Georgios Pavlakos, Xiaowei Zhou, Konstantinos G Derpanis, and Kostas Daniilidis. Coarse-to-fine volumetric prediction for single-image 3D human pose. In *CVPR*, 2017. 2

[29] Davis Rempe, Tolga Birdal, Aaron Hertzmann, Jimei Yang, Srinath Sridhar, and Leonidas J Guibas. HuMoR: 3D human motion model for robust pose estimation. In *ICCV*, 2021. 2, 5

[30] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022. 4

[31] Julian Straub, Thomas Whelan, Lingni Ma, Yufan Chen, Erik Wijmans, Simon Green, Jakob J Engel, Raul Mur-Artal, Carl Ren, Shobhit Verma, et al. The Replica dataset: A digital replica of indoor spaces. *arXiv preprint arXiv:1906.05797*, 2019. 5

[32] Andrew Szot, Alex Clegg, Eric Undersander, Erik Wijmans, Yili Zhao, John Turner, Noah Maestre, Mustafa Mukadam, Devendra Chaplot, Oleksandr Maksymets, Aaron Gokaslan, Vladimir Vondrus, Sameer Dharur, Franziska Meier, Wojciech

Galuba, Angel Chang, Zsolt Kira, Vladlen Koltun, Jitendra Malik, Manolis Savva, and Dhruv Batra. Habitat 2.0: Training home assistants to rearrange their habitat. In *NeurIPS*, 2021. 5

[33] Zachary Teed and Jia Deng. DROID-SLAM: Deep visual slam for monocular, stereo, and RGB-D cameras. In *NeurIPS*, 2021. 2, 3, 6, 7

[34] Guy Tevet, Sigal Raab, Brian Gordon, Yonatan Shafir, Daniel Cohen-Or, and Amit H Bermano. Human motion diffusion model. In *ICLR*, 2023. 4

[35] Denis Tome, Patrick Peluse, Lourdes Agapito, and Hernan Badino. xR-EgoPose: Egocentric 3D human pose from an HMD camera. In *ICCV*, 2019. 2

[36] Hsiao-Yu Tung, Hsiao-Wei Tung, Ersin Yumer, and Katerina Fragkiadaki. Self-supervised learning of motion capture. In *NeurIPS*, 2017. 2

[37] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017. 3

[38] Jian Wang, Lingjie Liu, Weipeng Xu, Kripasindhu Sarkar, and Christian Theobalt. Estimating egocentric 3D human pose in global space. In *ICCV*, 2021. 2

[39] Jiashun Wang, Huazhe Xu, Jingwei Xu, Sifei Liu, and Xiaolong Wang. Synthesizing long-term 3D human motion and interaction in 3D scenes. In *CVPR*, 2021. 5

[40] Alexander Winkler, Jungdam Won, and Yuting Ye. QuestSim: Human motion tracking from sparse sensors with simulated avatars. In *SIGGRAPH ASIA*, 2022. 3

[41] Kevin Xie, Tingwu Wang, Umar Iqbal, Yunrong Guo, Sanja Fidler, and Florian Shkurti. Physics-based human motion estimation and synthesis from videos. In *ICCV*, 2021. 2

[42] Weipeng Xu, Avishek Chatterjee, Michael Zollhofer, Helge Rhodin, Pascal Fua, Hans-Peter Seidel, and Christian Theobalt. Mo^{2}cap^2: Real-time mobile 3D motion capture with a cap-mounted fisheye camera. *IEEE Transactions on Visualization & Computer Graphics (TVCG)*, 25(05):2093–2101, 2019. 2

[43] Dongseok Yang, Doyeon Kim, and Sung-Hee Lee. Lobstr: Real-time lower-body pose prediction from sparse upper-body tracking signals. *Computer Graphics Forum*, 40(2):265–275, 2021. 3

[44] Xinyu Yi, Yuxiao Zhou, Marc Habermann, Soshi Shimada, Vladislav Golyanik, Christian Theobalt, and Feng Xu. Physical inertial poser (PIP): Physics-aware real-time human motion tracking from sparse inertial sensors. In *CVPR*, 2022. 3

[45] Xinyu Yi, Yuxiao Zhou, and Feng Xu. TransPose: real-time 3D human translation and pose estimation with six inertial sensors. *ACM Transactions on Graphics (TOG)*, 40(4):1–13, 2021. 2

[46] Ye Yuan, Umar Iqbal, Pavlo Molchanov, Kris Kitani, and Jan Kautz. GLAMR: Global occlusion-aware human mesh recovery with dynamic cameras. In *CVPR*, 2021. 2

[47] Ye Yuan and Kris Kitani. 3D ego-pose estimation via imitation learning. In *ECCV*, 2018. 2

[48] Ye Yuan and Kris Kitani. Ego-pose estimation and forecasting as real-time PD control. In *ICCV*, 2019. 2, 6

[49] Ye Yuan, Shih-En Wei, Tomas Simon, Kris Kitani, and Jason Saragih. SimPoE: Simulated character control for 3D human pose estimation. In *CVPR*, 2021. 2

[50] Mingyuan Zhang, Zhongang Cai, Liang Pan, Fangzhou Hong, Xinying Guo, Lei Yang, and Ziwei Liu. MotionDiffuse: Text-driven human motion generation with diffusion model. *arXiv preprint arXiv:2208.15001*, 2022. 4

[51] Yang Zheng, Yanchao Yang, Kaichun Mo, Jiaman Li, Tao Yu, Yebin Liu, C. Karen Liu, and Leonidas J Guibas. GIMO: Gaze-informed human motion prediction in context. In *ECCV*, 2022. 5, 6

[52] Xiaowei Zhou, Menglong Zhu, Spyridon Leonardos, Konstantinos G Derpanis, and Kostas Daniilidis. Sparseness meets deepness: 3D human pose estimation from monocular video. In *CVPR*, 2016. 2

[53] Yi Zhou, Connelly Barnes, Jingwan Lu, Jimei Yang, and Hao Li. On the continuity of rotation representations in neural networks. In *CVPR*, 2019. 3