EgoBody: Human Body Shape and Motion of Interacting People from Head-Mounted Devices

Supplementary Material

A Details of Dataset Building

A.1 Calibration

To spatially calibrate the Kinects-Kinects and Kinects-HoloLens2, we employ a checkerboard to obtain an initial calibration; we then refine the result by running ICP [11] on the scene point clouds reconstructed from the depth sensor of the devices. Additionally, the Kinects-HoloLens2 calibration is refined by the keypoint-based optimization scheme as described in the main paper Sec. 3.3. To register camera data into the 3D scene, we first manually annotate a set of correspondence points between the scene mesh and scene point clouds given by the Kinect depth frames to obtain an initial rigid transformation, which is again refined via ICP.

A.2 SMPL-X Body Model

We use SMPL-X [73], which represents the body as a function $\mathcal{M}_b(\boldsymbol{\gamma}, \boldsymbol{\beta}, \boldsymbol{\theta}, \boldsymbol{\phi})$. It maps global translation $\boldsymbol{\gamma} \in \mathbb{R}^3$, body shape $\boldsymbol{\beta} \in \mathbb{R}^{10}$, pose $\boldsymbol{\theta}$ and facial expression $\boldsymbol{\phi} \in \mathbb{R}^{10}$ to a triangle body mesh with 10,475 body vertices. $\mathcal{M}_b = (V_b, F_b)$ with body vertices $V_b \in \mathbb{R}^{10475 \times 3}$ and faces F_b. The pose parameters include body, facial and hand poses. $J(\boldsymbol{\beta})$ denotes the 3D body joints in the neutral pose, which can then be posed according to a given $\boldsymbol{\theta}$.

A.3 Data Processing

Body point cloud extraction. We use Mask-RCNN [31] to get coarse human instance segmentation masks in Kinect RGB frames and refine the masks with DeepLabv3 [31]. The obtained human instance segmentation masks are mapped to Kinect depth frames to segment the human body point clouds from point clouds extracted from depth.
Subject index reordering. Note that OpenPose [13], DeepLabv3 and Mask-RCNN all work per-frame, without temporal tracking. For each sequence, we therefore reorder subject indices from OpenPose detections and human masks by their relative position to each other in 2D, such that each subject has a consistent index across all frames and all Kinect views.
Data cleaning. 2D joint detection and human instance segmentation can fail in the presence of body-body or body-scene occlusions (Fig. S1 left/middle). Thus we manually clean the failed detections and exclude them from the reconstruction

pipeline. We also manually clean inaccurate 2D joint detections due to self-occlusions (Fig. S1 right). We also leverage the depth information from Kinect cameras, to filter out 2D joints with a large difference between its depth value and the median depth value of all 2D joints of the target person.

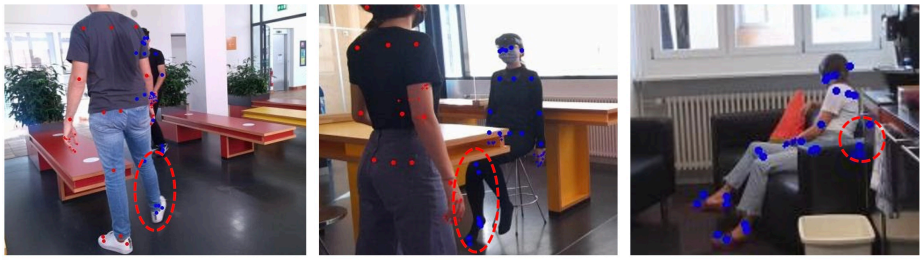


Fig. S1: Limitations for OpenPose 2D joint detection when body-body occlusion (left), body-scene occlusion (middle), or self-occlusion (right) happens. Different keypoint colors denote different detected body indices.

EgoSet-interactee subset selecting. Egocentric image frames with extreme human body truncations are excluded from EgoSet-interactee subset by the following filtering procedure. We run OpenPose 2D joint detection on all egocentric image frames, and manually exclude spurious detections (irrelevant people in the background or false positives on scene objects) for each frame. As OpenPose may split the joints from the same body into several detections, we merge them into one body in such case, where the joint conflict is resolved by the confidence score. We include the frames in EgoSet-interactee subset when at least six valid joints (OpenPose BODY_25 format) of the interactee are detected. We threshold the joint confidence score by 0.2 as in [48]. Note that five joints concentrated on the head are considered as one joint due to the close distances, as well as the four joints on each foot. The bounding box is computed with the processed results.

B Ground-truth Annotation Quality

B.1 Reconstruction Accuracy

To evaluate the 2D accuracy of the reconstructed body in the egocentric view, we randomly select 1,517 frames and manually annotate their 2D joints via Amazon Mechanical Turk (AMT), using the SMPL-X skeleton definition. By projecting 3D joints estimated by our pipeline on the egocentric view, the mean 2D joint error (2D Euclidean distance between the projections and AMT annotations) is 31.08 pixels (image in 1920×1080 resolution). We also evaluate the 3D accuracy of our ground truth on two metrics: **Scan-to-body Chamfer Distance** (*1.65cm*), and **3D per joint error (PJE)** (*4.32cm*). The 3D PJE is computed as follows: we leverage AMT to annotate 2D body joints in all Kinect views

Table S1: Motion smoothness evaluation of HPS and our dataset. PSKL(X, A) denotes PSKL(HPS/ours, AMASS), and PSKL(A, HPS/ours) the reverse direction. Better result in boldface.

	PSKL(X,A) ↓	PSKL(A,X) ↓
HPS [29]	0.924	1.044
EgoBody	**0.312**	**0.262**

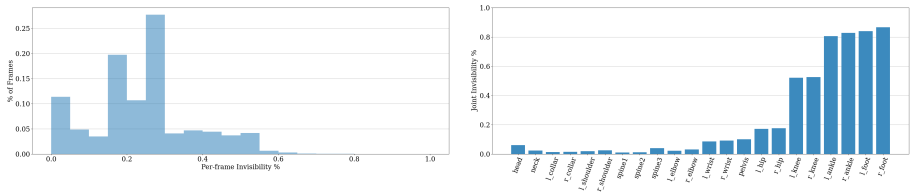
for 100 frames, from which the annotated 3D joints are obtained via multi-view triangulation. The 3D PJE is then measured between our ground truth SMPL-X body joints and the annotated 3D joints.

B.2 Motion Smoothness

The motion smoothness of our dataset is evaluated with the Power Spectrum KL divergence score (PSKL) [32] as in [102]: we measure the distance between the distribution of joint accelerations in our dataset and that in the high-quality mocap dataset AMASS [62]. The lower the score is, the more the motions resemble the natural motions in AMASS. Besides, we compare with HPS [29], a recent egocentric view dataset (Tab. S1). The significantly lower PSKL score of EgoBody reflects the high quality of our ground truth motions.

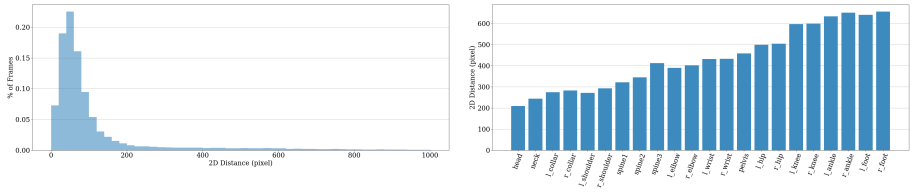
C More Statistics

Joint invisibility. We consider 2 types of "invisibility" measurements: frame-wise invisibility and joint-wise invisibility ratio. For each frame, the frame-wise invisibility ratio calculates the percentage of invisible joints among all body joints. As shown in Fig. S2(a), partial body invisibility occurs in most frames, with even over 60% body joints invisible in extreme cases. For each body joint, the joint-wise invisibility ratio calculates the ratio of frames when the joint is invisible among all frames (See Fig. S2(b)). The lower part of the body exhibits higher chances of invisibility (knees around 50% and feet around 80%). The upper body parts are more visible: neck, shoulder, spine, and elbow joints above all, while wrist and head joints have slightly higher invisibility (around 10%).
Eye gaze and attention. The HoloLens2 eye tracking provides the eye gaze 3D ray's starting point and orientation, which we can intersect with our 3D reconstructions to calculate the location the user looks at. By projecting the 3D eye gaze point onto the egocentric images, we perform analysis on the distances between this projected 2D eye gaze point (attention area) and the body joints of the interactee on the egocentric view images. For all frames where the 2D gaze point lies within the image, Fig. S3(a) plots the distribution of the Euclidean distance between the 2D gaze point and its nearest body joint. For more than 75% of the frames, the distance between the 2D gaze point and its nearest joint lies within 120 (pixels), indicating that the camera wearer's attention is highly



(a) **(b)**

Fig. S2: (a) Distribution of frame-wise invisibility ratio (% of invisible joints among all body joints for each frame). (b) Joint-wise invisibility ratio (% of occurrences when the corresponding joint is invisible among all frames.): 'l_' denotes 'left_' and 'r_' denotes 'right_'.



(a) **(b)**

Fig. S3: (a) Distribution of the 2D distance between the 2D gaze point and its nearest body joint. (b) Mean 2D distance between the 2D gaze point and each body joint. 'l_' denotes 'left_' and 'r_' denotes 'right_'.

focused on the interactee during interactions. The mean distance between each joint and the 2D gaze point over all frames (Fig. S3(b)) reveals that the subjects' attention tends to be closer to the upper body joints during interactions.

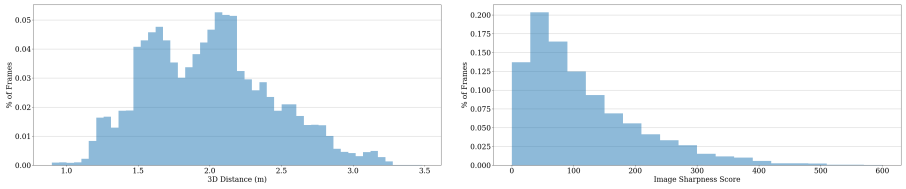
Interaction distance. EgoBody covers a large range of indoor interaction distances: the 3D Euclidean distance between the pelvis joints of the interacting subjects ranges from $0.90m$ to $3.48m$. Fig. S4(a) shows the distribution of the interaction distances between 2 interacting subjects.

Motion blur. The image sharpness score (variance of the Laplacian of the image) [74] quantifies the motion blur in our dataset (the distribution is shown in Fig. S4(b)). A higher score indicates sharper images.

D Experiments: Details and Discussions

D.1 Discussion on Evaluation Metrics

While MPJPE is a commonly used measurement for pose estimation accuracy, it is very sparse and does not penalize the error caused by wrong joint twisting. This motivates us to also include the V2V error as a metric in our benchmark: it is not only a straightforward measure for the shape estimation, but also measures the pose error in a denser and stricter way than MPJPE as it also penalizes erroneous longitudinal joint rotations.



(a) **(b)**

Fig. S4: (a) Distribution of interaction distances between two interacting subjects. (b) Distribution of the image sharpness score on EgoSet-interactee test set.

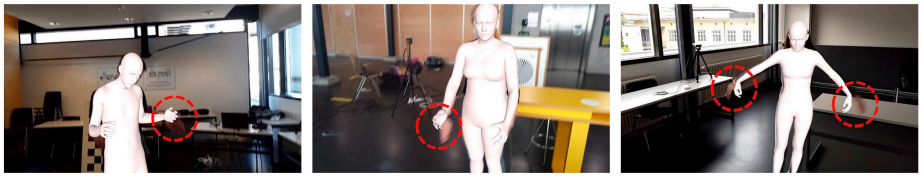


Fig. S5: Misalignments between the body mesh and HoloLens2 images during fast motions (for example, fast hand movements).

By default we consider the MPJPE and V2V errors without the Procrustes Alignment (PA), as PA eliminates the discrepancy in the global orientation, a major source of errors for most methods. Deprecating PA-based metrics is becoming a recent trend [46, 72].

D.2 Baseline Improvement on EgoBody

Implementation details. We fine-tune SPIN [48], METRO [57] and EFT [36] using the official codes, but with slight customization in the training as follows. For SPIN, we disable the SMPLify-in-the-loop during training since the EgoBody training set already provides direct 3D supervision from the pseudo ground truth. Note that this is in fact the default setting in SPIN when the 3D ground truth is available.

Extended results and discussions. As shown in Fig. S6, while SPIN works well on images when the full body is visible, it fails on images where the subjects are truncated. EFT, in contrast, is more robust against such truncation as the model is trained on aggressively cropped images as data augmentation during training. The effectiveness of EFT's data augmentation is further supported by our fine-tuning experiments. After fine tuning SPIN and EFT on our training set, both models show greater robustness against motion blur and image truncation, and quantitatively achieve lower errors than the original models on all metrics, as shown in the main paper Tab. 3. Together with the experiments on the You2Me dataset (see main paper Sec. 5.4), this shows that our training set can help adapt existing 3DHPS models to egocentric view data.

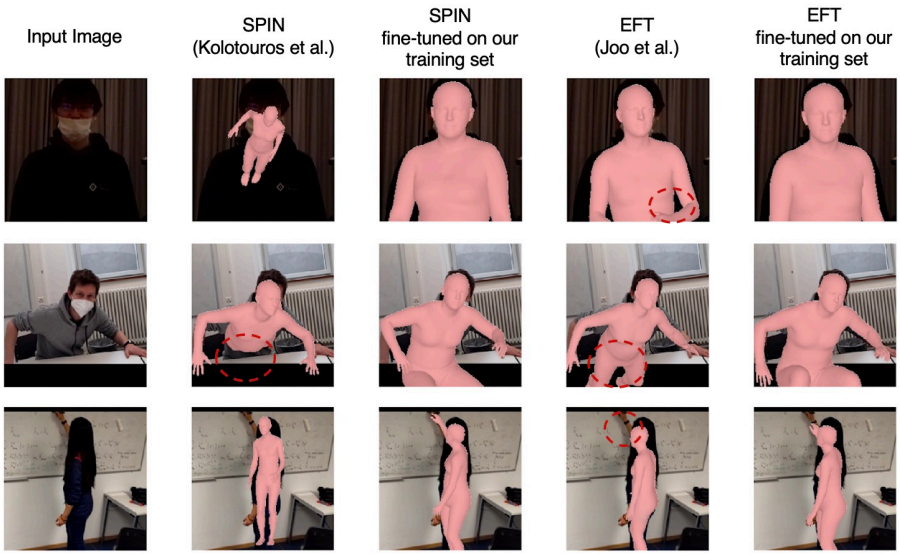


Fig. S6: Qualitative results from the baseline evaluation.

D.3 Details of the Cross-dataset Evaluation on You2Me

Here we report the experimental setup of the You2Me dataset experiment. The You2Me dataset [68] provides egocentric view images (taken with a chest-mounted GoPro camera), and the ground truth 3D joint locations of both the camera wearer and the interactee. The ground truth 3D joints are however in a world coordinate system, making it infeasible to compute the translation-only MPJPE (see main paper Sec. 5.1): since the camera calibration between the GoPro and the world coordinate is unknown, even a perfect prediction (in the camera coordinate system) may differ from the ground truth up to a rigid transformation. To account for this problem, we first perform the Procrustes Alignment, which solves for the scale, translation and global rotation, to align the predicted 3D body joints with the ground truth, and then compute the MPJPE of the aligned bodies, resulting in the PA-MPJPE errors reported in the paper.

D.4 Experiment with Motion Blur Augmentation

Data augmentation could potentially simulate blurring and truncation. Can the performance of existing methods be enhanced on EgoBody by simply fine-tuning them on the original dataset that they are trained on with extra data augmentation? EFT is trained with aggressive *image cropping* which to an extent simulates body truncation in our dataset. Indeed its superior performance has proven the effect of data augmentation, but such augmentation does not fully address the challenges in EgoBody: a clear performance gap can be seen between the original EFT and all models fine-tuned on our dataset (SPIN-ft, METRO-ft, EFT-ft,

see Tab. 4 in main paper). Likewise, here we additionally analyze motion blur augmentation. We fine-tune the pre-trained SPIN model with additional motion blur augmentation on the datasets it's originally trained on. For the motion blur we randomly set blur direction, angle, and kernel size to blur the training images with a probability of 0.5 during fine-tuning, and we experiment with multiple settings with different kernel sizes. No improvements are observed compared with the original SPIN model when evaluated on our egocentric test set (Tab. S2). Both observations indicate that existing data augmentation techniques cannot fully resolve the challenges in the egocentric setup, and EgoBody fills this gap.

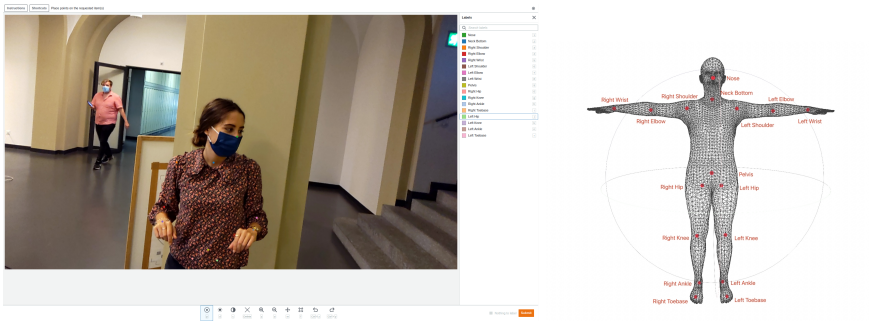
Table S2: Evaluation of motion blur augmentation on our egocentric test set. 'SPIN-blur' denotes the result of fine-tuning SPIN on its original training set with additional motion blur augmentation.

	MPJPE ↓	PA-MPJPE ↓	V2V ↓	PA-V2V ↓
SPIN [48]	182.8	116.6	187.3	123.7
SPIN-blur	184.9	128.2	188.5	129.2

D.5 Details for Baseline Methods

CMR [49] firstly regresses 3D locations of SMPL body vertices via a graph convolutional network. An image-based CNN encodes the input image into a feature vector, which is attached to the graph network defined by a mesh template. A Multi-Layer Perceptron (MLP) predicts the SMPL parameters based on regressed body vertices.
METRO [57] adopts the model-free formulation to estimate body vertices and 3D body joints directly. A transformer encoder models interactions for vertex-vertex and vertex-joint via self-attention mechanism.
SPIN [48] integrates iterative optimization loops into the neural network training to combine advantages of both regression-based and optimization-based methods. The optimization fits the SMPL body model to 2D joints on the image to enable more robust supervision for the regressor.
EFT [36] augments existing large-scale 2D datasets with 3D annotations by Exemplar Fine-Tuning. Starting from a pre-trained 3D pose regressor, the model weights are fine-tuned by fitting 2D joints to images. Supervised by the obtained 3D annotations, a model with the same architecture as SPIN [48] is trained with extreme crop augmentation and auxiliary input representations.
LGD [81] proposes to use neural networks to predict the parameter update rules in the optimization framework. A Gradient Updating Network regresses the update step for SMPL parameters in each optimization iteration.
PARE [46] leverages the visibility information of each body part, and predicts body-part-guided attention masks to achieve robust prediction for SMPL parameters with body occlusions.

E AMT Annotation Details



(a) **(b)**

Fig. S7: (a) The user interface, and the (b) definition of the 17 joints, for AMT manual annotation.

To evaluate the body shape and pose annotation accuracy, we collect manual annotations of 2D locations of 17 body joints on the EgoSet-interactee frames via Amazon Mechanical Turk (AMT). The user interface is illustrated in Fig. S7(a). We exclude body joints that are ambiguous for manual annotating (head, spine1, spine2, spine3, left_collar, right_collar) from the first 22 SMPL-X body joints, and add the nose joint which is easy to define for users. The user is provided with the definition of body joints (Fig. S7(b)), and an image of the target person to annotate (in case of irrelevant people in the background). Self-occluded keypoints need to be inferred, while keypoints occluded by scene objects are not required to be annotated. We downsample with a rate of 50 on the EgoSet-interactee data, which yields a total number of 2,286 frames.

For better annotation quality, each frame is annotated by five users, and joints annotated by less than three users are ignored. A small part of the users flip the left and right side, inducing non-negligible noise for the ground-truth evaluation. To address this issue, we filter out the outliers and correct the flipped annotations by the following procedure. For each annotated joint, the 2D distance from each annotation $\mathbf{x}_i$ $(i = 0, 1, ..., 4)$ to the mean location $\bar{\mathbf{x}}$ is calculated as $d_i = ||\mathbf{x}_i - \bar{\mathbf{x}}||_2$. An annotation is considered as the outlier if $\frac{d_i - \bar{d}}{\sigma} > 1.5$, where $\bar{d}$ is the average distance, and $\sigma = \sqrt{\frac{1}{n}(d_i - \bar{d})^2}$ is the standard deviation of the distance. For joints that have the counterpart on the other side of body we flip the left/right side of the annotations and perform the same outlier detection, to fix cases when the users flip the left and right side.

F Limitations

As there exists no solution to synchronize HoloLens2 and Kinect via hardware, we align their clocks via software, using a flashlight which is visible to all devices as signal for the first frame. Although HoloLens2 exhibits frame drops occasionally, the corresponding frames of all devices can be aligned according to the timestamps provided by HoloLens2 Research Mode API [88]. Besides, we empirically observe a small temporal misalignment. In our case, the misalignment can be observed for fast motions (for example, for hand movements, as shown in Fig. S5). However, this issue is inevitable for the synchronization between third-person view cameras and HMDs [68]. Despite the small misalignment, our reconstruction reaches a high accuracy as proved by the reconstruction accuracy in Sec. 4.2.