Supplementary Materials: Fish2Mesh Transformer

Tianma Shen[1] Aditya Puranik[1] James Vong[1] Vrushabh Deogirikar[1] Ryan Fell[1]
Julianna Dietrich[1] Maria Kyrarini[1] Christopher Kitts[1]
David C. Jeong[1]
[1]Santa Clara University, 500 El Camino Real, Santa Clara, CA 95053
{tshen2, apuranik, jvong, vdeogirikar, rfell, jdietrich, mkyrarini, ckitts, dcjeong}@scu.edu

S.1 Equipment and Setup

S.1.1 Recording Equipment

Each recording session utilized two iPhone 14s (requiring at least 1.3 GB of storage per participant) and two tripods to secure these iPhones at specified positions (See Fig. 1. Additionally, an Insta360 ONE X2 head-mounted camera with 5.7K 360-degree video resolution was employed, requiring a minimum storage of 2 GB per participant.

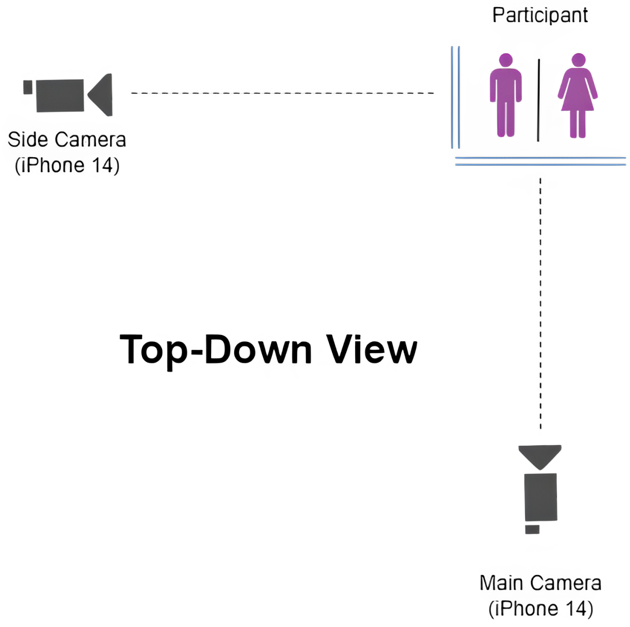


Figure 1. Top-Down View of the Experimental Setup

S.1.2 Camera Positioning and Setup

For each session, the main camera and the side camera were placed in predetermined locations on the floor, marked with blue tape to ensure consistent positioning as shown in Fig. 1. The tripods were adjusted to capture the participant's full body, allowing for actions involving movement beyond the immediate standing area, such as lunges. Additionally, as shown in Fig. 2, both cameras were set to 1x zoom to limit distortions caused by camera settings. The head-mounted camera was configured to Hi-Resolution Video Recording Mode, with the screen facing downward to facilitate easy referencing by the participant (See Fig. 3).

S.2 Participant Instructions and Recording Procedure

S.2.1 Initial Setup and Synchronization

Before recording, all three cameras (the Main, Side, and Head cameras) were started simultaneously to ensure synchronization. Each participant performed a single loud clap at the beginning of each session, which served as a synchronization cue across all cameras.

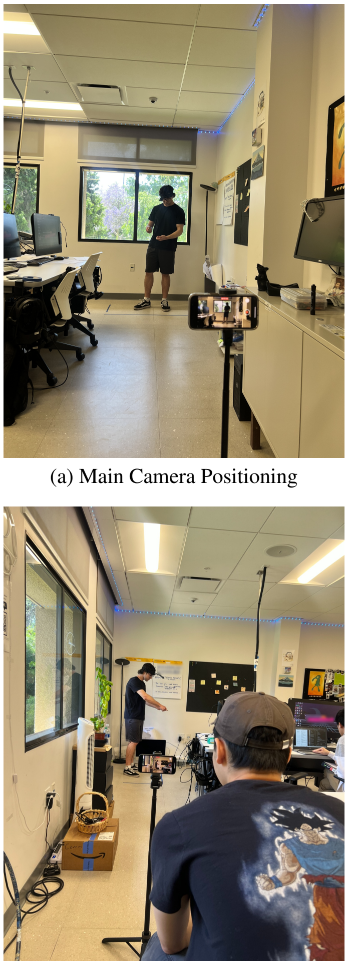
S.2.2 Action Instructions

Participants were given a prompt (an action to be performed), which they sequentially presented to the Head Camera, Main Camera, and Side Camera as shown in 3. They were instructed to perform the action naturally for 10 to 20 seconds, with a maximum duration of 30 seconds. Participants could step outside the marked area as required for specific actions. The session concluded after the participant completed the final prompt, at which point all cameras were stopped.

S.3 Post-Session Procedures and Data Management

S.3.1 Equipment Collection and Storage

Following each recording session, the Main, Side, and Head Cameras were collected. Video files from the Main and Side Cameras were uploaded to Google Drive using specific naming conventions.


(a) Main Camera Positioning

(b) Side Camera Positioning

Figure 2. Both labeled (Main and Side) cameras were set up to capture body movement with some flexibility beyond the restricted area.

S.3.2 Video Uploading and Naming Conventions

Videos were uploaded to designated folders in Google Drive, categorized by camera type:

- Main Camera videos were stored in the `MainCamera/` folder and named using the format `Main_#` (where # denotes the participant ID).
- Side Camera videos were stored in the `SideCamera/` folder, with the naming format `Side_#`.
- Head Camera videos were stored in the `HeadsetCamera/` folder, named `Headset_#`.

We will be releasing the dataset, along with the code

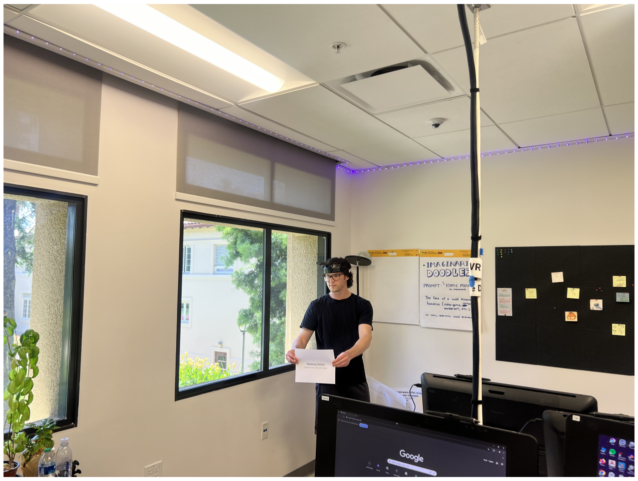


Figure 3. Participant Presenting the Prompt to the Side Camera

S.4 Data Processing and Annotation

S.4.1 Frame Extraction and Timestamping

All recorded videos were segmented into individual frames using the `video_split.py`[1] script, with outputs saved to a temporary folder for subsequent processing. Key frames, marking the beginning and end of each activity, were identified: the starting frame was noted as the moment the participant set the prompt paper aside, while the end frame was marked just before the participant reached for the next prompt. The initial clap served as a synchronization point across all three cameras, with the frame of the clap identified in each recording.

S.4.2 Camera Synchronization

Frames from the main and head cameras were aligned using the `move_frames.py`[1] script, ensuring that each frame sub-folder corresponded accurately to a specific recording session. Frames were automatically renamed with a numerical suffix, starting from 1, to facilitate consistency across datasets.

S.4.3 Mesh Labeling and Annotation

Human mesh labeling was performed using the HMR model, which was run on an NVIDIA 4090 GPU to maximize efficiency. Labels were synchronized with the Head Camera frames, with further renaming handled by the `rename_main_label.py`[1] script, ensuring alignment between datasets for analysis.

[1] This script is located in the dataset folder within our code's files.

S.5 Final Dataset Preparation and Cleanup

S.5.1 Frame Deletion and Cleanup

To optimize storage and maintain data quality, redundant or unaligned frames were removed using the `deleteUnlabel.py`[1] script. Specifically, frames from the Head Camera without corresponding labels were deleted, as were Main Camera labels without matching frames in the Head Camera data.

S.5.2 Dataset Finalization

The cleaned and processed frames were organized into a final dataset folder, ensuring alignment between the Main, Side, and Head Camera recordings. This dataset was prepared for subsequent analysis.

S.6 Action Prompts for Participants

Participants performed a set of loosely-defined actions in front of the cameras, facing the Main Camera while the Side and Head Cameras recorded from alternative angles. The prompts included are listed in Table 1.

S.6.1 Prompt Selection Reasoning

We selected these prompts for our fisheye view-to-mesh generation experiment because they emphasize activities involving a wide range of arm movements and dynamic body postures, providing diverse and challenging data for accurate 3D reconstruction. Activities such as sweeping the floor, raking leaves, and shoveling the ground involve broad arm sweeps and bending motions, capturing extended arm positions and varied postural changes. Similarly, tasks like stirring a big cauldron and flipping a pancake highlight rotational arm movements and interactions with imaginary objects, adding complexity to the dataset. This focus on arm-centric activities ensures that the dataset covers a wide range of human motion that reflects everyday physical activities.

To enhance the variability and richness of the dataset, we also designed the experiment to encourage participants to introduce natural variation in how they performed each activity. For example, stirring a big cauldron could be done with different arm trajectories, speeds, and stances, while activities like throwing a frisbee or hitting a tennis ball allowed for differences in angle, intensity, and follow-through motion. Even simpler tasks like clapping or stretching arms offer opportunities to capture subtle differences in execution, such as variations in arm height or motion path. These variations help to ensure the dataset captures a broad spectrum of movement styles and realistic body configurations.

By focusing on arm movements and dynamic postures, we aimed to create a dataset that challenges mesh generation algorithms to handle large-scale limb movement and body articulation effectively. The inclusion of a fisheye camera mounted on the head complements the front and side views by capturing a broader field of view, ensuring that even when parts of the body are occluded in one perspective, they are likely visible in another. This comprehensive approach allows us to build a robust foundation for advancing 3D human pose and shape reconstruction.

Table 1. Grouped activities under Personal Care, Household, and Outdoor/Leisure.

Category	**Prompt**
Personal Care	Brushing Teeth Combing Hair Reading a Book Stretching Arms Lunges Arm Circles Drinking a Cup of Water Fanning Yourself Taking Bites out of a Burger
Household Activities	Sweeping Floor Ironing Clothes Washing Dishes Stirring a Big Cauldron Vacuuming the Floor Stuffing a Bag Flipping a Pancake Swatting a Fly with a Flyswatter Pouring Many Glasses of Water Pulling out a Drawer and Closing It
*Outdoor/Leisure Activities	Clapping Raking Leaves Shoveling the Ground Throwing a Frisbee Flying a Kite Hitting a Tennis Ball Fishing (Casting a Line) Skipping Rocks Walking a Dog

S.7 Data Processing and Evaluation Metrics

S.7.1 Performance Evaluation Metrics

For assessing human mesh accuracy, the Mean Per Joint Position Error (MPJPE) and Procrustes-aligned MPJPE (PA-MPJPE) metrics were used. Both metrics were derived from code embedded within the EgoHMR model scripts, providing a basis for evaluation.

S.7.2 Model Efficiency and Runtime Analysis

To support real-time applications in AR/VR and robotics, Fish2Mesh is intentionally designed to be lightweight while maintaining strong reconstruction performance. As shown in Table 2, our model uses only 7.5 M

Table 2. Comparative results of model complexity.

	GFLOPs	Parameters	Model Size	Inference
Fish2Mesh (Ours)	**4.74**	**7.5 M**	**48.19 MB**	**4.47 ms**
4DHumans	125.64	730.3 M	2583.98 MB	14.21 ms
EgoHMR	12.47	46.0 M	1798.96 MB	5.61 ms
FisheyeViT	85.36	167.5 M	650.12 MB	9.88 ms

parameters, requires 4.74 GFLOPs, and has a model size of 48 MB, which is significantly smaller than EgoHMR (46 M, 12.47 GFLOPs, 1.8 GB) and FisheyeViT (167.5 M, 85.36 GFLOPs, 650 MB). This enables a fast inference time of 4.47 ms, outperforming EgoHMR (5.61 ms) and FisheyeViT (9.88 ms) in runtime speed.

To achieve this, we deliberately avoid large vision backbones and egocentric-specific architectures like EgoFormer and EgoViT, which either introduce incompatible attention mechanisms or lack architectural flexibility. Instead, we leverage the Swin Transformer due to its hierarchical design and seamless integration with our proposed Egocentric Positional Embedding (EPE). This pairing enables both geometric adaptability and runtime efficiency within a compact design footprint.

S.8 Related Datasets and Literature

S.8.1 Datasets Considered

In this section, we summarize the various datasets explored throughout our research on egocentric 3D human pose estimation and motion capture. Each dataset is described by its primary purpose, content, and notable details. Additionally, we provide a section for datasets ultimately selected for model training and evaluation.

Table 3 summarizes the distribution of various camera types and mounting setups to evaluate generalization across our explored datasets. These include first-person footage captured with different device specifications and custom rigs with certain field of view (FOV) and sensor characteristics. Our model demonstrates consistent performance across variations, suggesting strong generalizability across multiple optical distortions and viewpoints.

Table 3. Comparison of camera specifications across datasets.

Dataset	Camera(s)	Focal Length[1]	FOV	Mount Setup
ECHP	GoPro HERO9	16.4 mm	155°	Helmet-mounted[2]
Ego4D	Varies[3]	Varies	Varies	Varies
Mo2cap2	N/A	N/A	182°	Hat-mounted
EgoPW	N/A	N/A	N/A	Helmet-mounted, downward facing
Fish2Mesh (Ours)	Insta360 ONE X2	7.2 mm	180° / 360°	Head-mounted

S.8.1.1 Reviewed Datasets

- **Mo2Cap2**: A mobile 3D motion capture dataset collected using a cap-mounted fisheye camera, designed for real-time 3D motion estimation in egocentric settings. This dataset includes both synthetic fisheye training data and references the MPII Human Pose and Leeds Sports Pose (LSP) datasets for broader applicability [13].
- **xR-EgoPose**: Created for 3D egocentric pose estimation from a headset-mounted camera, this dataset is widely used in the development of VR and AR applications. It also includes references to datasets like Mo2Cap2, Human3.6M, COCO, and MPII to enhance training diversity [10].
- **Ego4D**: This large-scale egocentric video dataset spans over 3,000 hours of global footage, capturing diverse everyday activities. It is supplemented by several other datasets, including HowTo100M and AVA Speech, making it valuable for multimodal AI research [4].
- **EgoPW**: A dataset specifically designed for egocentric 3D human pose estimation in varied environments, integrating weak supervision to aid model training. The dataset's scene-aware version, EgoPW-Scene, includes additional real-world variability for robust model performance [12].
- **EgoWholeBody**: Contains over 700,000 frames with detailed annotations, created to support whole-body motion capture in egocentric views using a fisheye camera. The dataset includes SMPL-X model frames, with augmentation from synthetic sources like Mixamo for enhanced body and hand realism [11].
- **EgoBody**: A synthetic dataset designed for egocentric whole-body pose estimation with significant variability in body shapes, actions, and environmental scenes. EgoBody is commonly used with the EgoWholeBody dataset to train advanced motion capture models [14].
- **UnrealEgo**: A dataset aimed at robust egocentric 3D human motion capture, featuring synthetic environments rendered in Unreal Engine. It is used primarily for evaluating model performance in diverse virtual scenarios [1].
- **EgoCap**: Designed for egocentric marker-less motion capture, this dataset uses two fisheye cameras to capture complex motion data. It includes both synthetic and real-world data for enhanced model accuracy [9].
- **EgoFish3D**: A self-supervised learning dataset captured using a fisheye camera, focused on egocentric 3D pose estimation. This dataset also includes synthetic training data to facilitate model learning [8].
- **Human3.6M**: A large-scale 3D human sensing dataset

used as a benchmark for human pose estimation. It provides high-resolution images with 3D ground truth annotations [3, 5].

- **COCO**: A widely used dataset for object detection and segmentation tasks, offering diverse annotations for over 80 object categories. It provides a foundational resource for training egocentric pose models [7].
- **MPII**: Created for 2D human pose estimation, the MPII dataset serves as a standard benchmark for evaluating human pose models. It covers a wide range of everyday activities in real-world settings [2].
- **LSP**: The Leeds Sports Pose (LSP) dataset is specifically designed for human pose estimation in sports contexts. It includes challenging poses and occlusions, making it valuable for training robust pose estimation models [6].

S.8.1.2 Final Datasets Used in Analysis

The final list of datasets employed as shown in Table 3 our research model's training and testing phases.

- **ECHP** [8]: Dataset of daily human actions used in EgoFish3D with GoPro cameras with fish-eye lenses as a basis for egocentric 3D pose estimation, with emphasis on hand pose data.
- **Ego4D** [4]: A large-scale egocentric video dataset comprising over 3,000 hours of video from around the world. Additional datasets mentioned within Ego4D include HowTo100M, VoxCeleb, AVA Speech, AVA Active Speaker, AVDIAR, and EasyCom.

References

[1] Hiroyasu Akada, Jian Wang, Soshi Shimada, Masaki Takahashi, Christian Theobalt, and Vladislav Golyanik. Unrealego: A new dataset for robust egocentric 3d human motion capture. 2022. 4

[2] Mykhaylo Andriluka, Leonid Pishchulin, Peter Gehler, and Bernt Schiele. 2d human pose estimation: New benchmark and state of the art analysis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014. 5

[3] Cristian Sminchisescu Catalin Ionescu, Fuxin Li. Latent structured models for human pose estimation. In *International Conference on Computer Vision*, 2011. 5

[4] Kristen Grauman, Andrew Westbury, and Eugene Byrne et al. Ego4d: Around the world in 3,000 hours of egocentric video, 2021. 4, 5

[5] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(7):1325–1339, jul 2014. 5

[6] Sam Johnson and Mark Everingham. Clustered pose and nonlinear appearance models for human pose estimation. pages 1–11, 01 2010. 5

[7] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft coco: Common objects in context, 2014. 5

[8] 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*, 25:8880–8891, 2023. 4, 5

[9] 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, 2016. 4

[10] Denis Tome, Thiemo Alldieck, Patrick Peluse, Gerard Pons-Moll, Lourdes Agapito, Hernan Badino, and Fernando De la Torre. Selfpose: 3d egocentric pose estimation from a headset mounted camera. 2020. 4

[11] Jian Wang, Zhe Cao, Diogo Luvizon, Lingjie Liu, Kripasindhu Sarkar, Danhang Tang, Thabo Beeler, and Christian Theobalt. Egocentric whole-body motion capture with fisheyevit and diffusion-based motion refinement, 2023. 4

[12] 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, 2022. 4

[13] Weipeng Xu, Avishek Chatterjee, Michael Zollhoefer, Helge Rhodin, Pascal Fua, Hans-Peter Seidel, and Christian Theobalt. Mo2cap2: Real-time mobile 3d motion capture with a cap-mounted fisheye camera, 2018. 4

[14] 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, 2021. 4

[1] 35mm equivalent.

[2] Forward 13–18 cm, tilted 15–25° downward.

[3] GoPro, Vuzix Blade, Pupil Labs, ZShades, ORDRO EP6, iVue Rincon 1080, Weeview7.