EgoFish3D: Egocentric 3D Pose Estimation from a Fisheye Camera via Self-Supervised Learning

Yuxuan Liu [1,1], Jianxin Yang [2], Xiao Gu [2], Yijun Chen [2], Yao Guo [2], and Guang-Zhong Yang [2]

[1]Institute of Medical Robotics
[2]Affiliation not available

October 31, 2023

Abstract

Egocentric vision has a wide range of applications for human-centric activity recognition. However, the use of the egocentric fisheye camera allows wide angle coverage but image distortion is introduced along with strong human body self-occlusion, which can impose significant challenges in data processing and model reconstruction. Unlike previous work only leveraging synthetic data for model training, this paper first presents a new real-world EgoCentric Human Action (ECHA) dataset. By using the self-supervised learning under multi-view constraints, we propose a simple yet effective framework, namely EgoFish3D, for egocentric 3D pose estimation from a single image in different real-world scenarios.

EgoFish3D: Egocentric 3D Pose Estimation from a Fisheye Camera via Self-Supervised Learning

Yuxuan Liu, Jianxin Yang, Xiao Gu, *Student Member, IEEE*, Yijun Chen, Yao Guo, *Member, IEEE*, and Guang-Zhong Yang, *Fellow, IEEE*

***Abstract*—Egocentric vision has a wide range of applications for human-centric activity recognition. However, the use of the egocentric fisheye camera allows wide angle coverage but image distortion is introduced along with strong human body self-occlusion, which can impose significant challenges in data processing and model reconstruction. Unlike previous work only leveraging synthetic data for model training, this paper presents a new real-world EgoCentric Human Action (ECHA) dataset. To tackle the difficulty of collecting 3D ground truth using motion capture systems, we simultaneously collect the images from a head-mounted egocentric fisheye camera as well as from two third-person-view cameras, circumventing the environmental restrictions. By using the self-supervised learning under multi-view constraints, we propose a simple yet effective framework, namely EgoFish3D, for egocentric 3D pose estimation from a single image in different real-world scenarios. EgoFish3D incorporates three main modules. 1) *The third-person-view module* takes two exocentric images as input and estimates the 3D pose represented in the third-person camera frame; 2) *the egocentric module* predicts the 3D pose in the egocentric camera frame; and 3) *the interactive module* estimates the rotation matrix between the third-person and the egocentric views. Experimental results on our proposed ECHA dataset and existing benchmark datasets demonstrate the effectiveness of the proposed EgoFish3D, which can achieve superior performance to existing methods.**

***Index Terms*—Egocentric vision, 3D human pose estimation, Self-supervised learning, Multi-view constraints**

I. Introduction

EGOCENTRIC vision is an emerging field in computer vision, involving the analysis of data captured from a head-mounted or chest-mounted wearable camera [1]–[3]. When the egocentric camera is directed downwards, especially by incorporating a fisheye lens, the human body and the surrounding environment can be captured with an enlarged field-of-view, offering expanded visual cues for processing [3]–[6]. Compared to the third-person view, egocentric vision is advantageous for long-term human-centric perception in a free-living environment, offering new opportunities for understanding human behavior and social activities [5]–[8].

One of the prerequisites of egocentric vision for downstream applications is accurate pose estimation from egocentric views. Although extensive progress in monocular 2D/3D human pose estimation have been achieved in recent years [9]–[12], these conventional third-person-view pose estimation methods are prone to inaccurate when directly used to predict 2D/3D poses from the novel first-person viewpoint, as shown in Fig. 1(b), which shows typical results that can be achieved. Hitherto, there are few dedicated egocentric human pose estimation methods available due to several inherent challenges in egocentric vision as shown in Fig. 1(c). First, there exist significant human body self-occlusions from the first-person view, especially for the head and lower limb, making the estimation of occluded joints difficult. Second, although an egocentric camera with a fisheye lens enlarges the field of view and captures more details of the human body, the recorded images are severely distorted, increasing the difficulty of accurate annotation in practice. Furthermore, there is a lack of real-world datasets with accurate ground truth data as the collection using a motion capture (MoCap) system is labor-intensive and limited to small laboratory settings. To partially address these issues, recent effort within the vision community has been directed to building public datasets using synthetic human models [13] [14]. However, training with synthetic datasets can affect the generalization capability of the model when subsequently applied to real-world scenarios. To circumvent the above problems, it is necessary to develop

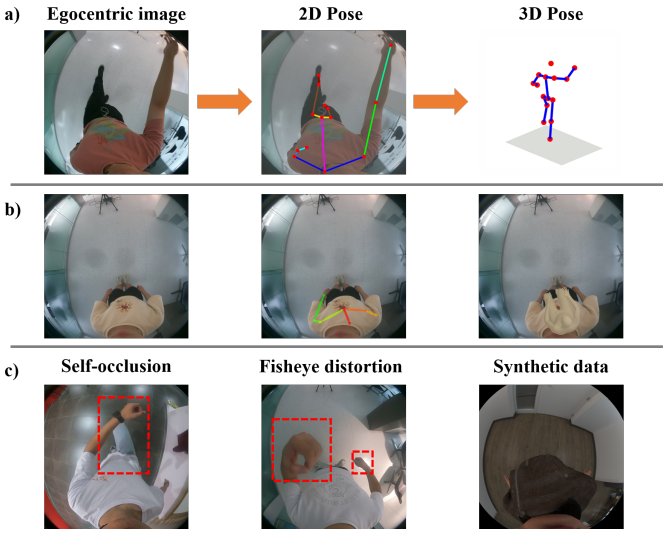

Fig. 1. Egocentric 3D pose estimation from a single fisheye camera. a) Our proposed EgoFish3D can achieve accurate 2D and 3D pose estimation from the distorted image captured by a single fisheye camera. b) Existing third-person-view 2D/3D pose estimation methods [9], [10] fail on this challenging task from images captured by the fisheye camera. c) The strong self-occlusion for lower limbs, the severe distortion of the fisheye camera, and the lack of real-world datasets are the inherent challenges in egocentric vision.

Y. Liu, J. Yang, Y. Guo, G.-Z. Yang are with Institute of Medical Robotics, Shanghai Jiao Tong University, Shanghai, China. ({200009051yx, jianxinyang, yao.guo, gzyang}@sjtu.edu.cn). X. Gu is with the Hamlyn Centre for Robotic Surgery, Imperial College London, London, UK. (xiao.gu17@imperial.ac.uk). Y. Chen is with School of Information Science and Technology, Shanghai Tech University, China (chengyj2@shanghaitech.edu.cn).

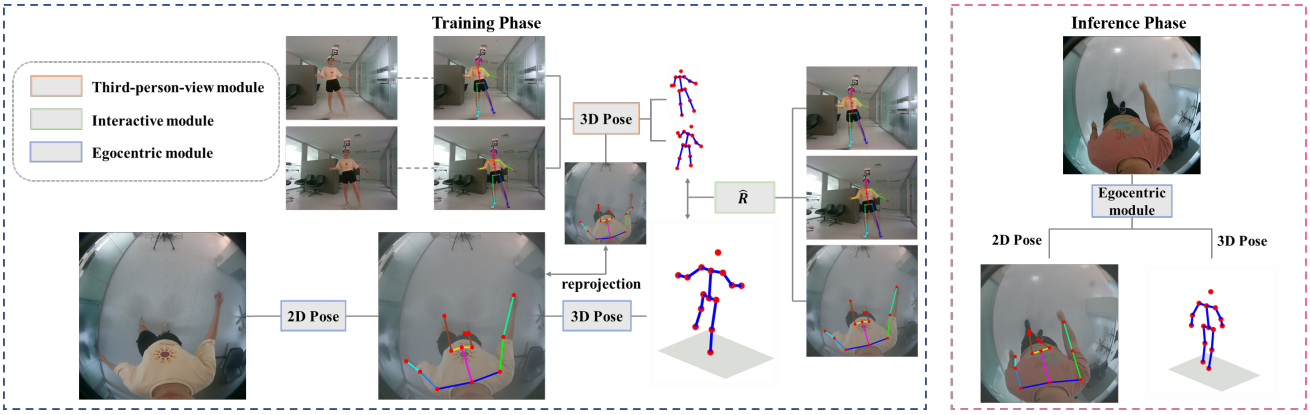

Fig. 2. Illustration of the training and inference phases of our proposed EgoFish3D. During the training phase, the third-person-view module takes two images from the third-person view as input and generates a relatively accurate 3D pose represented in the coordinate system of the external cameras, the interactive module predicts the rotation difference between the third-person-view and the egocentric coordinate systems, and the egocentric module estimates both 2D and 3D poses from an egocentric distorted image. During the inference phase, only the egocentric module can directly predict 2D and 3D poses from an egocentric image captured by a fisheye camera.

a dedicated method for egocentric 3D human pose estimation. Furthermore, it is advantageous to incorporate self-supervised learning. Some recent work has already leveraged the intrinsic constraints across multiple third-person views, such as multi-view geometry and view consistency [15]–[17], to enable the 3D human pose estimation without ground truth. These self-supervised methods have demonstrated comparative pose estimation performance against those fully-supervised counterparts. In practice, however, there is still the challenge of directly transferring the multi-view self-supervised mechanism for egocentric 3D pose estimation. First, the intrinsic parameters of the third-person-view and egocentric cameras are totally different. Second, there is often limited overlap between the third-person-view and first-person-view images, and the transformation between two views is hard to acquire. More importantly, conventional 2D/3D pose estimation methods can hardly work on egocentric images, thus limiting the direct use of self-supervised learning with multi-view constraints.

To address the aforementioned challenges, we first constructed the EgoCentric Human Action (ECHA) dataset using a head-mounted GoPro camera with a fisheye lens, and two RGB cameras were used to simultaneously capture images from a third-person view. The training and validation datasets of ECHA consist of 30 video sequences (∼75k frames) recorded in 8 different real-world indoor/outdoor scenes, in which 10 different daily actions performed by 9 subjects with 20 different body textures were recorded. The test dataset consisting of 10 actions performed by 4 subjects with new body textures were simultaneously captured by a multi-camera motion capture system with ground truth annotations. This dataset can not only contribute as a new benchmark for egocentric 3D pose estimation, but also help enhance the generalization capability of the proposed method to real-world scenarios. Central to this paper, we propose a novel self-supervised method, namely EgoFish3D, for egocentric 3D human pose estimation from a single head-mounted fisheye camera. An overview of our proposed method is shown in Fig. 2. The EgoFish3D consists of three modules: 1) the third-person-view module; 2) the egocentric module; and 3) the interactive module. The third-person-view module first generates the 3D pose from two third-person-view cameras, providing a relatively accurate 3D pose represented in a third-person-view coordinate system. The egocentric module then takes the distorted first-person-view image as input, performs 2D pose estimation, and predicts the 3D pose in the egocentric coordinate system as the final output. Within this module, the latent features, 2D heatmaps and human masks are incorporated in order to improve the accuracy. Another interactive module is introduced to estimate the 3D rotation of two 3D poses, adding supervision rules for training the other two modules. During the training phase, we train these three modules in a self-supervised manner. During the inference stage, only the egocentric module is used to predict the 3D pose from a single egocentric fisheye image. The proposed work represents the first attempt in achieving egocentric 3D pose estimation from a fisheye camera looking downwards in a self-supervised manner without 3D pose ground truth as prior. Experimental results on our ECHA and the public synthetic datasets [13], [14] demonstrate that our method can achieve good accuracy compared to existing supervised approaches.

In summary, the main contributions of this paper include:

- A self-supervised method is proposed to achieve egocentric 3D pose estimation from a single image without the need for 3D ground truth annotations.
- A real-world dataset ECHA is constructed, which contains the synchronized images from two third-person-view cameras and an egocentric fisheye camera.
- An interactive module is introduced to learn the relationship between the third-person and egocentric views.

II. Related Work

A. Human Pose Estimation from Egocentric Camera

With the increasing popularity of egocentric vision [7], human pose estimation from a single egocentric camera has received significant attention in recent years. Unlike the image/video captured by a camera from the third-person view, egocentric vision still faces inherent challenges originates from the visual data captured from a novel first-person viewpoint. In

previous works, the egocentric cameras are typically placed on the chest [4] or head [13], [14], [18]–[21], looking outwards [4], [18], [19] or looking downwards [13], [14], [20].

To mimic the visual perception of a human, one line of research is based on the egocentric camera looking outwards, but suffering from difficulties for recovering the 3D human pose only with limited observed human body. Jiang *et al.* [4] took advantage of the dynamic motion signatures of the surroundings to infer the invisible pose from a chest-mounted camera. However, the estimated poses are inaccurate and easily affected by the changing of environment. Another group of research [18], [19] modeled the egocentric 3D pose estimation and foresting as a Markov decision processing when the pose estimation is limited to a single mode of action, such as running or walking.

Compared to the camera looking outwards, some recent works change the egocentric camera to a downward-looking setting, which can consistently capture the human body in the field of view. Moreover, they all integrate the camera with a fisheye lens to enlarge the perception area of human body, which can boost the performance of egocentric 3D human pose estimation. By using a head-mounted stereo fisheye camera, Rhodin *et al.* [20] proposed a markerless egocentric full-body motion capture method, but the stereo camera is inconvenience in practical applications. Xu *et al.* [14] presented a real-time egocentric 3D pose estimation method with a single cap-mounted fisheye camera, which takes both original and zoom-in images as input to deal with the strong occlusion of the lower limbs. However, this method does not directly regress the 3D pose in the inference phase but predict the absolute depths of the joints instead. Most recently, Tome *et al.* [13] introduced a large corpus of synthetic dataset from a head-mounted fisheye camera, and proposed a three-branch network that achieves the state-of-the-art 3D pose estimation performance on this synthetic dataset. To deal with the problem of the severe distortion, Zhang *et al.* [21] proposed an automatic calibration module to estimate the fisheye camera parameters, thus mitigating the effect of image distortions for robust egocentric 3D pose estimation. It should be pointed out that the models as proposed in [13], [14], [21] are trained on synthetic data under fully-supervision, leading to a degraded generalization capability of the model in real-world applications. Motivated by this, in this paper we first establish a real-world egocentric human action dataset. In addition, we take full advantage of the multi-view constraints between the third-person-view and egocentric cameras to achieve egocentric 3D human pose estimation in a self-supervised manner.

B. 3D Human Pose Estimation via Self-Supervised Learning under Multi-View Constraints

Recently, self-supervised learning has attracted increasing attention in estimating 3D human pose, in which the multi-view information is utilized to mitigate the ambiguity of learning 3D human poses from synchronized 2D images captured from third-person-view cameras [22]–[25]. However, the fusion of multiple views and the annotation of 3D poses in different camera views are challenging. Employing some typical fusing methods, such as multi-view consistency of the same pose [15], [22], [25] and triangulation [24], [26], can tremendously reduce the labor cost of 3D human pose labelling and make the network learning in a self-supervised manner.

Rhodin *et al.* [22] employed the multi-view consistency to constrain the system to predict the same pose in all views with the help of only a few annotated data. CanonPose [15] disentangled the observed 2D pose into a canonical 3D pose and a camera orientation. In specific, it contains several sub-networks inferring the same 3D pose from different views, and the predictions from all views are aggregated to produce the final 3D pose inference. However, only applying multi-view consistency constraint is not sufficient enough because the model may trap in a trivial solution with different inputs [22], [27]. Iqbal *et al.* [27] introduced a novel objective function based on normalized 3D bone lengths that computed from Human3.6M dataset. Differently, Rhodin *et al.* [28] took advantages of temporal consistency prior to first learn a geometry-aware body representation from sequential unlabelled multi-view images, and then map the novel geometry representation to actual 3D poses. In this paper, we aim to address the multi-view self-supervised learning from the combination of third-person and egocentric views.

The other branch of study in fusing multi-view data employed conventional triangulation method, given the intrinsic and extrinsic parameters of the cameras [24], [26]. Iskakov *et al.* [26] firstly proposed a baseline method that computes the 3D human pose from multi-view 2D poses algebraically. Kocabas *et al.* [24] utilized the epipolar geometry theorem to generate 3D pose annotations from multiple 2D poses. Our method also incorporates the triangulated 3D pose as the prior for faster convergence and better performance.

III. EgoCentric Human Action (ECHA) Dataset

There is a pressing need for available real-world data to enable the development of egocentric pose estimation algorithms due to the unique camera placement and the images captured from first-person view. However, the existing training methods for egocentric pose estimation are almost based on synthetic datasets, and some real-world data is only used for the test. Due to the domain gap between the synthetic environment and the real-world environment, the generalization capability of the model in real applications is limited. More important, it is faced with difficulties for simultaneously capture ground truth data using MoCap system for supervised learning. This is because the multi-camera MoCap system is limited in a certain environment and the annotation is labour-intensive.

In this paper, we first construct an EgoCentric Human Action (ECHA) dataset to enable the egocentric 3D human pose estimation in a self-supervised learning manner. Unlike previous datasets mainly consist of synthetic images [13], [14], our ECHA is composed of real-world images collected by an egocentric camera with a fisheye lens in different scenarios.

A. Data Collection

To overcome the difficulty in acquiring ground truth data by MoCap system, two RGB cameras are used to capture images from the third-person view and a head-mounted GoPro

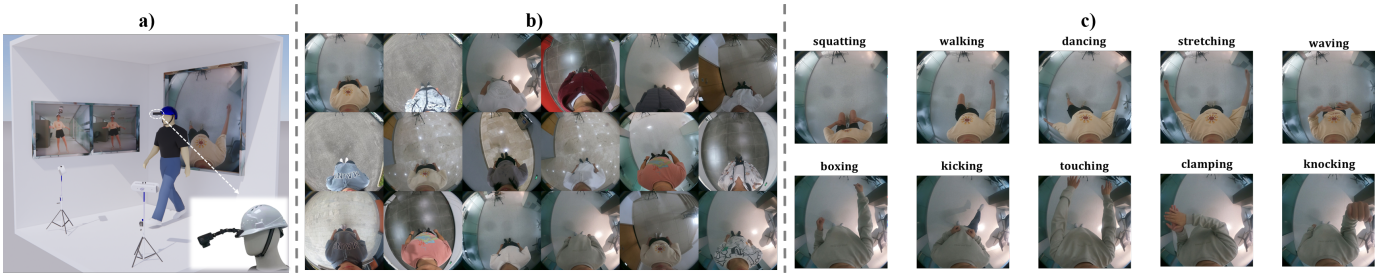

Fig. 3. The details of our proposed ECHA dataset. a) Placement of two third-person-view cameras (rgb camera of Realsense D455) and a head-mounted egocentric camera (GoPro) with a fisheye lens looking downwards; b) Selected examples of the different scenes, subjects, and body textures in our ECHA dataset. In total, we capture the data of 9 subjects in 8 different scenes with 20 different body textures; c) 10 different daily actions are recorded in our dataset.

camera with a fisheye lens captures the images from the egocentric view. In this manner, the ECHA dataset consists of well-synchronized images captured from two different views, i.e., the third-person view and egocentric view. In practice, the egocentric camera is fixed on the head through a helmet and extends forward about 13-18cm, and tilts downwards about 15-25 degrees to reduce the self-occlusion and ensure that the lower limbs can be seen as much as possible. We also use Aruco [29] marker to obtain the 6D pose of the egocentric camera represented in the third-person-view frames. All three cameras are well calibrated with a 25mm chessboard to determine the intrinsic parameters [30].

In the training and validation sets of our ECHA dataset, there are 9 different subjects with 20 different body textures performing 10 daily actions (i.e., *squatting*, *walking*, *dancing*, *stretching*, *waving*, *boxing*, *kicking*, *touching*, *clamping*, *knocking*). To improve the diversity of the dataset, real-world data are captured in 8 different scenes, both indoor and outdoor. In total, there are 30 video sequences about 75k frames in the ECHA dataset. To fully evaluate the performance as well as the generalization capability of different egocentric pose estimation algorithms, in the test set we simultaneously use the VICON MoCap system with a full-body gait model to collect these 10 daily actions performed by 4 subjects with new body textures, i.e., $\sim$17K frames with 3D ground truth of anatomical joint positions. It should be emphasized that the test data are captured in a same indoor scene due to the use of MoCap system, and more importantly 2 of 4 subjects are unseen in the training set.

In summary, the ECHA dataset can be divided into three different parts: about 65k images with both third-person-view images and egocentric images are for training, the remaining 10k images are for validation, and the egocentric data along with 3D ground truth captured by the VICON Mocap system form the test dataset. Further details of our dataset can be found in our *supplementary material*.

B. Data Preparation

It is noteworthy that our main focus is on egocentric pose estimation, so we exploit OpenPose [9] to offline extract 2D joints of the target human given the rgb images captured from third-person-view cameras. These two 2D poses are served as the input to the third-person-view module of EgoFish3D. Besides, a pretrained human instance segmentation model [31] is used to offline extract the human mask as the input to our egocentric module. The extrinsic parameters between the two third-person-view cameras are pre-measured and fixed during the data collection. To fit in our proposed network, we resize the images from the third-person view to 640×480 and the images from the egocentric view to 384×384.

IV. EgoFish3D for Egocentric 3D Pose Estimation

A. Problem Statement

In this paper, our aim is to perform egocentric 3D pose estimation from a single head-mounted fisheye camera by leveraging the self-supervision provided by two third-person-view cameras. Let denote $\{\mathrm{C}_1\},\{\mathrm{C}_2\}$ as the coordinate systems of two third-person-view cameras c_1, c_2 and $\{\mathrm{C}_{ego}\}$ indicates the local coordinate system of the egocentric fisheye camera c_{ego}. For a camera c, the captured image at each frame can be defined as $\mathbf{I}_c$, the corresponding 2D joints of the target human is $J_c = \{j_{c,1}, j_{c,2}, \cdots, j_{c,N}\} \in \mathbb{R}^{N\times 2}$, where $j_{c,i} = [u_{c,i}, v_{c,i}]$ indicates the pixel coordinates of a joint i and N is the number of key body joints. Accordingly, the estimated 3D joints represented in the camera c is $P_c = \{p_{c,1}, p_{c,2}, \cdots, p_{c,N}\} \in \mathbb{R}^{N\times 3}$, where $p_{c,i} = [x_{c,i}, y_{c,i}, z_{c,i}]$. Note that we first exploit OpenPose [9] to extract the 2D pose J_{c_1}, J_{c_2} from two third-person-view images $\mathbf{I_{c_1}}, \mathbf{I_{c_2}}$, and the transformation matrix $\{^{c_2}R_{c_1}, ^{c_2}T_{c_1}\}$ between two third-person-view cameras is given.

Given an image $\mathbf{I}_{c_{ego}}$ captured by the egocentric fisheye camera, a deep neural network $f_\theta(\cdot)$ with parameters θ is designed to first predict the 2D joint positions $J_{c_{ego}}$, and then estimate the 3D human pose $P_{c_{ego}}$ represented in the egocentric camera coordinate system. In this paper, we propose EgoFish3D to perform this task via self-supervised learning.

B. Overview of EgoFish3D

The architecture of our proposed EgoFish3D is shown in Fig. 4. In specific, our network contains three modules, i.e., third-person-view module f_θ^{trd}, egocentric module f_θ^{ego}, and the interactive module f_θ^{itr}.

The third-person-view module f_θ^{trd} achieves 3D pose estimation from two external cameras, which aims to offer a relatively accurate 3D pose as the supervision represented in the third-person-view camera coordinate system. The input of this module is the two 2D poses J_{c_1}, J_{c_2} estimated from two third-person-view images and the output is the 3D poses $\hat{P}_{c_1}, \hat{P}_{c_2}$ under the third-person-view coordinate system. The network is composed of several MLPs, where each MLP is

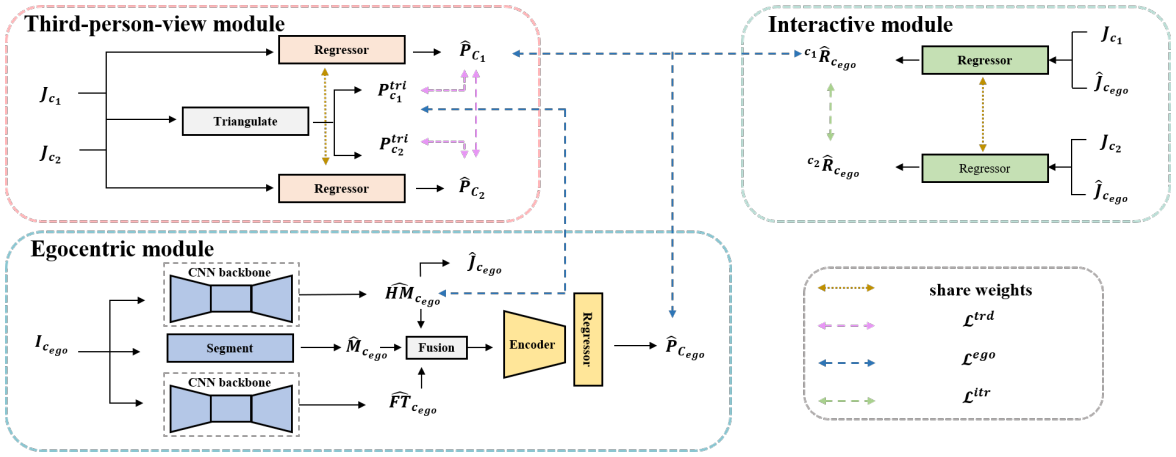

Fig. 4. Overview of our proposed EgoFish3D. The figure shows the network architecture, which contains three modules: third-person-view module, egocentric module and interactive module. The black arrows indicate direction of information flow. The colorized lines with arrows indicate different loss functions. The yellow dotted line indicates that two sub-networks share the same weights.

the combination of a linear layer, a batchnorm layer and an activation function.

For egocentric module f_θ^{ego}, the input is a single egocentric image $\mathbf{I}_{c_{ego}}$. We first perform 2D pose estimation to obtain the heatmaps of the body joints, and then predict the 3D pose $\hat{P}_{c_{ego}}$ under the egocentric coordinate system. To tackle the inherent challenges of pose estimation in distorted egocentric images, a feature fusion mechanism is proposed by combining the high-level features $\hat{FT}_{c_{ego}}$ of the input image, 2D heatmaps of body joints $\hat{HM}_{c_{ego}}$ and human masks $\hat{M}_{c_{ego}}$ together. The module is first built upon the CNN backbone to achieve feature fusion. Then the fused features are fed into an encoder-regressor, a combination of several CNN layers and MLPs, to generate the final 3D pose estimation.

Noted that the 3D poses estimated by the above two modules are represented in different coordinate systems, hence, it is necessary to perform the rotation alignment between third-person-view and the egocentric coordinate systems across frames. To this end, we introduce an interactive module in this paper, which takes the paired 2D poses $\{J_{c_1}, \hat{J}_{c_{ego}}\}, \{J_{c_2}, \hat{J}_{c_{ego}}\}$ from different coordinate systems as input. The network structure is similar to the third-person-view module and is composed of several MLPs, predicting the rotation matrices ${}^{c_1}\hat{R}_{c_{ego}}, {}^{c_2}\hat{R}_{c_{ego}}$, repesctively.

C. Third-Person-View Module

This module aims to predict 3D poses under the third-person-view camera coordinate system, by giving two third-person-view images and the transformation matrix between these two cameras. In order to generate accurate pose estimation, we combine the conventional triangulation via multi-view constraints and a learning-based depth estimation model.

1) 3D pose triangulation: Given the intrinsic parameters and transformation matrix of two cameras, a point in 3D space can be determined by triangulation with respect to its projections on two images. Among different triangulation algorithms, we use the depth estimation related one. After obtaining 2D poses J_{c_1}, J_{c_2} via OpenPose [9], the analytical solution of 3D positions $P_{c_1}^{tri}, P_{c_2,i}^{tri}$ can be calculated.

2) 3D pose estimation via depth prediction: To avoid the inaccurate 3D pose estimation by triangulation, we also introduce a network f_θ^{trd} to predict the depth values of 2D joints, i.e., $d_c = f_\theta^{trd}(J_c)$. Thus, the 3D joints $\hat{P}_c$ represented in each camera coordinate system can be calculated by leveraging the intrinsic parameters of this camera.

3) Loss functions: In this module, two constraints are involved to facilitate the training of the network. First, the constraint $\mathcal{L}_{pose}^{trd}$ between the depth-based prediction and the triangulation result.

$$\mathcal{L}_{pose}^{trd} = \sum_{c=c_1,c_2} \sum_{i=1}^{N} ||\hat{p}_{c,i} - p_{c,i}^{tri}||_1 \tag{1}$$

Besides, two estimated 3D poses represented in $\{C_1\}$ and $\{C_2\}$ are expected to the same after transformation.

$$\mathcal{L}_{tran}^{trd} = \sum_{i=1}^{N} ||\hat{p}_{c_1,i} - ({}^{c_2}R_{c_1}\hat{p}_{c_2,i} + {}^{c_2}T_{c_1})||_1 \tag{2}$$

During the training, the total loss $\mathcal{L}^{trd}$ can be formulated as the weighted sum $\mathcal{L}^{trd} = \omega_1\mathcal{L}_{pose}^{trd} + \omega_2\mathcal{L}_{tran}^{trd}$.

D. Egocentric Module

This module aims to predict both 2D and 3D poses of the target human under the egocentric camera coordinate system given a distorted fisheye image, where the 3D pose is the final output. To achieve better performance on 3D pose estimation, we propose a feature fusion method to combine the high-level features of the input image, the 2D heatmap of body joints and the mask of human body together. Moreover, the 2D heatmap as well as the 2D pose branch is incorporated with the reprojection constraints by leveraging the 3D pose estimated from the third-person-view module.

1) 2D pose and heatmap prediction by reprojection: After obtaining the 3D joints from the third-person-view module, an intuitive way is to project the 3D data onto the distorted egocentric image plane, thus providing a supervised information for training the egocentric 2D pose detector. The intrinsic parameters of the fisheye camera is extracted by the calibration

method [30]. As is well known, it is extremely challenging to directly predict the transformation between two cameras with less overlap and different intrinsic parameters. For simplicity, in our ECHA dataset, we make use of the Aruco marker to determine the 6D pose $\{^{c_1}\tilde{R}_{c_{ego}}, ^{c_1}\tilde{T}_{c_{ego}}\}$ of egocentric camera in the third-person-view coordinate system. However, the detection is seriously affected by the illumination and distance from the marker to camera. Thus, we only use the images with stable detection of the Aruco marker to train the egocentric 2D pose detector. Given the $P^{tri}_{c_1}$ estimated from the third-person-view camera c_1, it can be reprojected on the egocentric image. Accordingly, we can get the egocentric 2D pose $J^{rep}_{c_{ego}}$ as well as the heatmap $HM^{rep}_{c_{ego}}$. Next, the egocentric 2D pose detector is trained to predict the 2D heatmaps $\hat{HM}_{c_{ego}}$ under the supervision of $HM^{rep}_{c_{ego}}$, where MSE loss $\mathcal{L}^{ego}_{rep}$ is used as the constraints. The reprojected 2D pose is relatively low-accuracy due to the influence of the triangulated 3D pose and extrinsic parameters. It is noteworthy that the 2D pose detector can finally predict more accurate heatmaps compared to the reprojected ones.

$$\mathcal{L}^{ego}_{rep} = \sum_{i=1}^{N} ||\hat{HM}_{c_{ego},i} - HM^{rep}_{c_{ego},i}||^2_2 \tag{3}$$

2) Information fusion for egocentric 3D pose estimation: The main objective of this paper is to train the network f^{ego}_θ that can predict the 3D pose $\hat{P}_{c_{ego}}$ given a single fisheye image. To achieve this, we propose a feature fusion mechanism in the egocentric module to boost the pose estimation performance, which consists of three branches. The first branch uses the pretrained 2D pose detector to obtain the heatmaps $\hat{HM}_{c_{ego}}$, the second branch extracts the latent features $\hat{FT}_{c_{ego}}$ of the input image in feature space, and the third branch uses a pretrained human instance segmentation network [31] to extract the human masks $\hat{M}_{c_{ego}}$.The final fusion is built upon an attention-aware way, i.e, $\hat{P}_{c_{ego}} = f^{ego}_\theta(\hat{HM}_{c_{ego}} \odot \hat{FT}_{c_{ego}} \odot \hat{M}_{c_{ego}})$. Then the fused feature is fed into an Encoder and Regressor to predict the final 3D poses. During the training, two self-supervised constraints are used. One is the reprojection loss $\mathcal{L}^{ego}_{rep}$ mentioned above to force the egocentric 2D detector to generate reasonable heatmaps. The other one is the transformation constraint $\mathcal{L}^{ego}_{pose}$ of 3D poses between the third-person view and the egocentric view, which is supervised by the rotation matrix $^{c_1}\hat{R}_{c_{ego}}$ predicted by the interactive module. Note that we choose $\{C_1\}$ as the reference frame. Besides, an additional loss $\mathcal{L}^{ego}_{bone}$ on bone length $\hat{b}$ is utilized to force the length of L left and right links to be the same.

$$\mathcal{L}^{ego}_{pose} = \sum_{i=1}^{N} ||\hat{p}_{c_{ego},i} - ^{c_1}\hat{R}_{c_{ego}}\hat{p}_{c_1,i}||_2 \tag{4}$$

$$\mathcal{L}^{ego}_{bone} = \sum_{i=1}^{L} ||\hat{b}^{left}_{c_{ego},i} - \hat{b}^{right}_{c_{ego},i}||_2 \tag{5}$$

3) Loss functions: During the training of the egocentric module, the total loss $\mathcal{L}^{ego}$ is the weighted sum $\mathcal{L}^{ego} = \alpha_1\mathcal{L}^{ego}_{rep} + \alpha_2\mathcal{L}^{ego}_{pose} + \alpha_3\mathcal{L}^{ego}_{bone}$.

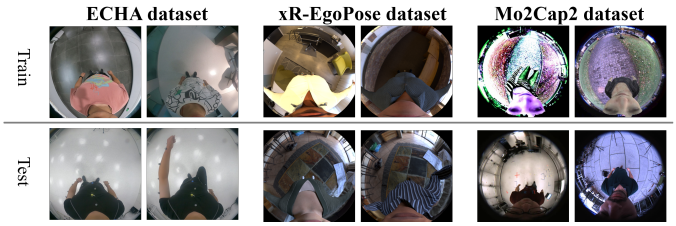

Fig. 5. Examples from our proposed ECHA dataset and the public datasets, i.e., xR-EgoPose [13] and Mo^{2}Cap^2 [14]. Compared to xR-EgoPose and Mo2^{2}Cap^2, both training and test data in the ECHA dataset are real-world images and suffer from less occlusions of lower limbs, improving the generalization capability of the learned models in practical applications.

E. Interactive Module

As the detection of Aruco markers is greatly affected by the environment and easily fails in some circumstances, we propose the interactive module to automatically predict the rotation matrix between the third-person-view and egocentric coordinate systems, offering the transformation constraints for training the egocentric module. Given the pair of 2D poses $\{J_c, \hat{J}_{c_{ego}}\}$, where $c \in \{c_1, c_2\}$, the interactive module f^{itr}_θ predicts the Euler angles $[\hat{\theta}, \hat{\phi}, \hat{\psi}]$ through a simple MLPs. Accordingly, the rotation matrix $\hat{R}$ can be determined. By leveraging the extrinsic parameters of two third-person-view cameras, the transformation constraint $\mathcal{L}^{itr}_{tran}$ can be defined as follow.

$$\mathcal{L}^{itr}_{tran} = ||^{c_1}\hat{R}_{c_{ego}} - ^{c_2}R_{c_1}\,^{c_2}\hat{R}_{c_{ego}}||_1 \tag{6}$$

To facilitate a fast convergence of the interactive module, we use the extrinsic parameters $^{c}\tilde{R}_{ego}$ predicted by Aruco markers to form the constraint $\mathcal{L}^{itr}_{mat}$ at the beginning of the training.

$$\mathcal{L}^{itr}_{mat} = ||^{c}\hat{R}_{c_{ego}} - ^{c}\tilde{R}_{c_{ego}}||_1, c = c_1, c_2 \tag{7}$$

During the training stage, the total loss $\mathcal{L}^{itr}$ can be formulated as the weighted sum $\mathcal{L}^{itr} = \beta_1\mathcal{L}^{itr}_{tran} + \beta_2\mathcal{L}^{itr}_{mat}$. Noted that $\mathcal{L}^{itr}_{mat}$ is only used in the first 5 epochs.

In summary, the loss function for training the whole network of our EgoFish3D is the sum of losses for the aforementioned modules, i.e., $\mathcal{L} = \mathcal{L}^{trd} + \mathcal{L}^{ego} + \mathcal{L}^{itr}$.

V. Experimental Settings

A. Dataset

For the **ECHA** dataset as mentioned in Sec. III, we train the model based on 65k frames from the sequences {*seq*1-*seq*2, *seq*4-*seq*11, *seq*13-*seq*17, *seq*21-*seq*30}, and validate our model on the rest of images from {*seq*3, *seq*12, *seq*18-*seq*20} to demonstrate qualitative results. To demonstrate the effectiveness of our proposed self-supervised method, we also capture seven test videos named {*test*1-*test*7} of 4 subjects (2 of them are novel subjects and all body textures are unseen in the training set) with 3D pose ground truth provided by the VICON system to offer quantitative results. Note that we do not capture the joint position of nose in VICON system, so we only report the results of 14 body joints.

To evaluate the effectiveness of our method, we also conduct the comparison experiments on the other two existing synthetic datasets. 1) **Mo^{2}Cap^2** [14] is a large-scale synthetic dataset

that simulates the images captured by a single cap-mounted fisheye camera, which contains 530k rendered images including about 3000 different actions and 700 different body textures. Besides, there is a real-world test set consisting of ∼5.5k frames for quantitatively evaluation. 2) **xR-EgoPose** [13] is also a synthetic dataset captured by a head-mounted fisheye camera. It has 383k frames with 23 male and 23 female characters performing 9 different actions. Both Mo^2Cap^2 and xR-EgoPose datasets contain the ground truth annotations of 2D and 3D joint positions. Fig. 5 demonstrates several examples of three datasets. Although our ECHA contains less images than Mo^2Cap^2 and xR-EgoPose, we believe that the real-world egocentric images in our dataset can somehow contribute to the community for the development of egocentric pose estimation algorithms.

B. Implementation Details

The detailed architecture of the proposed EgoFish3D is presented in the *supplementary material*. Since our method is based on self-supervised learning method, the network can hardly converge with an end-to-end training strategy. To make our model converge fast and reduce overfitting, we adopt a multi-stage training strategy instead. First, we train the third-person-view module with 3D pose triangulation and depth estimation methods to estimate the 3D pose under the third-person-view coordinate system with $\omega_1 = 1.0, \omega_2 = 0.5$. Then the egocentric 2D pose detector is trained to estimate the heatmaps with $\alpha_1 = 1.0$. Note that the input egocentric image is of size 384×384 and the dimension of heatmap is 48×48. To improve the generalization of the egocentric 2D pose estimation model in different real-world scenarios, we train the model on a combination of ECHA (∼40k with available 2D reprojection from triangulated 3D pose) and Mo^2Cap^2 (∼40k synthetic data with annotated 2D pose). Next, we train the interactive module to estimate the rotation matrix between the third-person-view coordinate system and the egocentric one. In the first 5 epochs, we use extrinsic parameters estimated from Aruco as the supervision to speed up the convergence of the network with $\beta_1 = 1.0, \beta_2 = 1.0$. Finally, we train the egocentric module and finetune the whole network for better performance with $\alpha_2 = 1.0, \alpha_3 = 0.05$. The proposed method and comparison methods are implemented by PyTorch, and we apply Adam for optimization with a learning rate of 0.001.

C. Evaluation Metrics

Two evaluation protocols are used in this paper. One refers to Mean Per Joint Position Error (MPJPE), which calculates the average distance between the ground truth and the predicted 3D joints as in Eq. (8).

$$E(P, \hat{P}) = \frac{1}{K}\frac{1}{N}\sum_{k=1}^{K}\sum_{i=1}^{N}||p_i^k - \hat{p}_i^k||_2 \tag{8}$$

The other refers to PA-MPJPE, which indicates the MPJPE after applying alignment by Procrustes Analysis to remove the global translation, rotation and scale of the two 3D poses.

D. Comparison Methods

Since we are the first to propose a self-supervised method for egocentric 3D pose estimation from a looking downwards fisheye camera by leveraging multi-view constraints from the third-person view, we compare our proposed EgoFish3D with three existing supervised methods [13], [14], [32] that use egocentric images with ground truth annotations. First, the comparison experiments are conducted on our ECHA dataset. For a fair comparison, we replace our egocentric module with the other comparison methods and keep the third-person-view and interactive modules. For the network proposed by Martinez [32], we implement the network after extracting the 2D pose from the heatmaps we predicted. Due to the code of [14] [13] are not publicly available, we implement these two models by ourselves instead. As it is difficult to determine the rotations between body parts in real-world data, only the first and the third branches of the network proposed by Tome [13] is implemented for the comparison. For the network proposed by Xu [14], we implement the heatmap-zoom and joint depth estimation modules, and generate the 3D pose by the reprojection formula of the egocentric fisheye camera. Second, we compare our proposed EgoFish3D with other methods on the Mo^2Cap^2 [13] and xR-EgoPose [14] datasets. To illustrate the generalization ability of our EgoFish3D, we direct apply our trained model to the real-world test data in Mo^2Cap^2 without finetuning and demonstrate the qualitative results. To compare the performance of 3D pose estimation, we retrain our egocentric view module on xR-EgoPose dataset and report both quantitative and qualitative results. The comparison methods on our dataset are noted as follows.

- Martinez [32], a baseline method with several MLPs for 3D pose estimation, where the input is 2D pose.
- Tome [13], a state-of-the-art supervised method with encoder-decoder network for egocentric 3D pose estimation.
- Xu [14], a two-branch network that takes both original and zoom-in images as input for supervised pose estimation.
- EgoFish3D, the full network of our proposed method.

E. Ablation Study

We also conduct ablation studies of the proposed EgoFish3D to demonstrate the effectiveness of different sub-networks and loss functions. We remove or change the following parts of our network one by one.

- EgoFish3D, the full network of our proposed method.
- w/o $\hat{M}$, an ablated model by removing the human instance segmentation $\hat{M}_{c_{ego}}$.
- w/o $\hat{FT}$, an ablated model by removing the branch of feature extraction $\hat{FT}_{c_{ego}}$.
- w/o $\hat{HM}$, an ablation study that removes the heatmap prediction branch $\hat{HM}_{c_{ego}}$ in the egocentric view module.
- w/o f_θ^{trd}, an ablated study removing the third-person-view module f_θ^{trd}. Only triangulated 3D pose is for supervision.
- w/o $\mathcal{L}_{pose}^{trd}$, the loss constraining the depth-based and triangulated poses in the third-person-view module is removed.
- w/o $\mathcal{L}_{mat}^{itr}$, an ablated model without $\mathcal{L}_{mat}^{itr}$ as in Eq. (7).

TABLE I
COMPARISON MPJPE(PA-MPJPE) RESULTS OF EGOCENTRIC 3D POSE ESTIMATION IN MILLIMETERS (mm) ON ECHA DATASET

Approach	All	squatting	Walking	Dancing	Stretching	Waving	Boxing	Kicking	Touching	Clamping	Knocking
Martinez [32]	118.3(80.0)	135.5(75.9)	114.8(73.9)	122.9(84.3)	133.2(85.0)	107.9(76.1)	116.7(82.2)	102.6(71.8)	117.9(79.0)	115.9(93.3)	116.4(84.6)
Tome [13]	112.4(73.9)	136.3(78.4)	113.9(72.6)	119.5(79.8)	123.9(78.2)	99.0(63.9)	110.3(71.9)	100.2(73.1)	110.7(67.7)	104.3(81.1)	103.4(72.0)
Xu† [14]	110.9(71.2)	**112.4**(70.7)	114.3(76.8)	**108.5**(70.1)	**114.7**(69.5)	105.9(65.5)	**109.5**(68.1)	106.7(72.0)	**102.2**(65.8)	128.0(85.9)	106.6(66.0)
EgoFish3D	**107.9**(73.1)	123.8(71.2)	**106.8**(68.9)	110.4(80.1)	121.4(81.3)	**95.6**(66.4)	111.2(75.4)	**94.6**(69.3)	110.5(70.0)	**101.6**(80.8)	**102.7**(71.4)
Ablated models	All	squatting	Walking	Dancing	Stretching	Waving	Boxing	Kicking	Touching	Clamping	Knocking
A: EgoFish3D	**107.9**(73.1)	**123.8**(71.2)	**106.8**(68.9)	**110.4**(80.1)	121.4(81.3)	95.6(66.4)	**111.2**(75.4)	**94.6**(69.3)	110.5(70.0)	101.6(80.8)	102.7(71.4)
B: w/o $\hat{M}$	114.1(82.5)	133.7(80.3)	110.4(71.3)	113.0(82.6)	127.8(92.0)	108.3(85.3)	116.9(88.0)	101.7(75.5)	109.0(83.3)	106.0(87.0)	112.3(87.7)
C: w/o $\hat{FT}$	114.5(73.5)	135.8(77.5)	116.6(70.2)	120.2(79.9)	126.8(80.5)	102.0(64.5)	113.4(73.5)	101.2(69.3)	110.9(68.9)	107.3(81.9)	107.6(71.1)
D: w/o $\hat{HM}$	123.9(82.3)	150.0(91.2)	119.9(81.6)	131.2(91.9)	128.7(85.6)	120.5(73.6)	115.7(77.3)	120.2(86.3)	117.8(74.8)	128.4(87.2)	111.8(74.0)
E: w/o f_{θ}^{trd}	114.4(78.3)	127.0(80.0)	112.0(76.6)	111.0(78.2)	126.7(79.3)	102.6(69.1)	114.0(75.3)	104.0(78.7)	117.0(72.8)	117.4(94.4)	113.0(79.1)
F: w/o $\mathcal{L}_{pose}^{trd}$	111.1(79.5)	148.1(86.1)	114.1(87.8)	115.9(88.6)	**117.1**(82.0)	**92.5**(64.3)	111.3(77.4)	98.5(78.5)	**106.1**(68.5)	**99.9**(76.6)	**101.0**(74.9)
G: w/o $\mathcal{L}_{mat}^{itr}$	521.1(78.7)	481.2(75.4)	510.7(76.7)	478.2(83.7)	572.7(83.0)	532.9(70.5)	563.0(76.4)	466.7(80.7)	519.7(76.3)	525.4(88.5)	558.1(77.2)

† Require additional information (i.e., the intrinsic parameters of the fisheye camera) during the inference phase.

TABLE II
COMPARISON MPJPE RESULTS OF EGOCENTRIC 3D POSE ESTIMATION IN MILLIMETERS (mm) ON XR-EGOPOSE DATASET

Approach	All	Gaming	Gesticulating	Greeting	Lower Stretching	Patting	Reacting	Talking	Upper Stretching	Walking
Martinez [32]	122.1	109.6	105.4	119.3	125.8	93.0	119.7	111.1	124.5	130.5
Tome(p3d) [13]	130.4	138.3	108.5	100.3	133.3	117.8	175.6	93.5	129.0	131.9
Tome(p3d+hm) [13]	58.2	56.0	50.2	**44.6**	**51.1**	59.4	60.8	43.9	53.9	57.7
Tome(p3d+hm+rot) [13]	**54.7**	60.4	54.6	44.7	56.5	57.7	52.7	56.4	**53.6**	**55.4**
EgoFish3D	57.8	**47.3**	**44.7**	47.7	58.3	**53.9**	**51.0**	**40.6**	61.7	56.7

VI. EXPERIMENTAL RESULTS

A. Quantitative Results

Without further clarify, the **bold** and underline values in the table indicate the best and the second best results in each column, respectively. All the elements indicate the result in millimeters (mm). By leveraging a full-body gait model provided by VICON MoCap system, the ground truth annotations are the 3D joint positions from the anatomical level.

1) ECHA dataset: The comparison 3D pose estimation results with other state-of-the-art methods on our ECHA dataset are reported in the upper part in Table I. The second column lists the average MPJPE(PA-MPJPE) results of all test data, and the remaining shows the results for each action.

It can be seen, our proposed EgoFish3D achieve the best average 3D pose estimation results (MPJPE=107.9 and PA-MPJPE=73.1) against other comparison methods [13], [14], [32]. This is because that our egocentric feature fusion method can leverage more useful information implied in the input image. For instance, the human mask is beneficial for removing the heatmaps with low confidence or beyond the scope of human body. It should be pointed out that the method proposed by Xu [14] is sensitive to the distorted egocentric images captured from a fisheye lens. Compared to our EgoFish3D, their method requires the intrinsic parameters of the fisheye camera to compute the 3D pose during the inference phase. Hence, the performance of this method will be significant degraded without the known intrinsic parameters of the camera. For action-level comparison, our EgoFish3D can still achieve the best or second best pose estimation for nine actions, except for *Boxing* that with upper limb outside the field of view, which proves that our method is more stable to different actions than the comparison methods.

2) xR-EgoPose dataset: For xR-EgoPose dataset, the comparison MPJPE results are presented in Table II. Compared to [13], our method (57.8) performs better than the two-branch network (58.2) and is on par with the three-branch network (54.7). However, the three-branch network of [13] contains the supervised information of the relative rotations between body joints, which is not easily acquired in real-world data. For different actions, our EgoFish3D achieves the best performance on five out of eight actions.

3) Ablation Study: The lower part of Table I lists the results of ablation studies on ECHA dataset. We report the average MPJPE(PA-MPJPE) results of all test data in the second column, and the rest shows the results for each action. It can be found that, compared to the other six ablated models *B*-*G*, our full network (model *A*) achieves the best on both MPJPE and PA-MPJPE. For action-level comparison, our full network can achieve the best or second best pose estimation for nine actions, which significantly outperforms other models.

Does the feature fusion method work? To evaluate the effectiveness of the proposed three-branch feature fusion mechanism in the egocentric view module, we remove each branch one by one corresponding to the models *B*-*D*. The results show that all three branches contribute to improve the pose estimation performance. The heatmap $\hat{HM}_{c_{ego}}$ introduces the

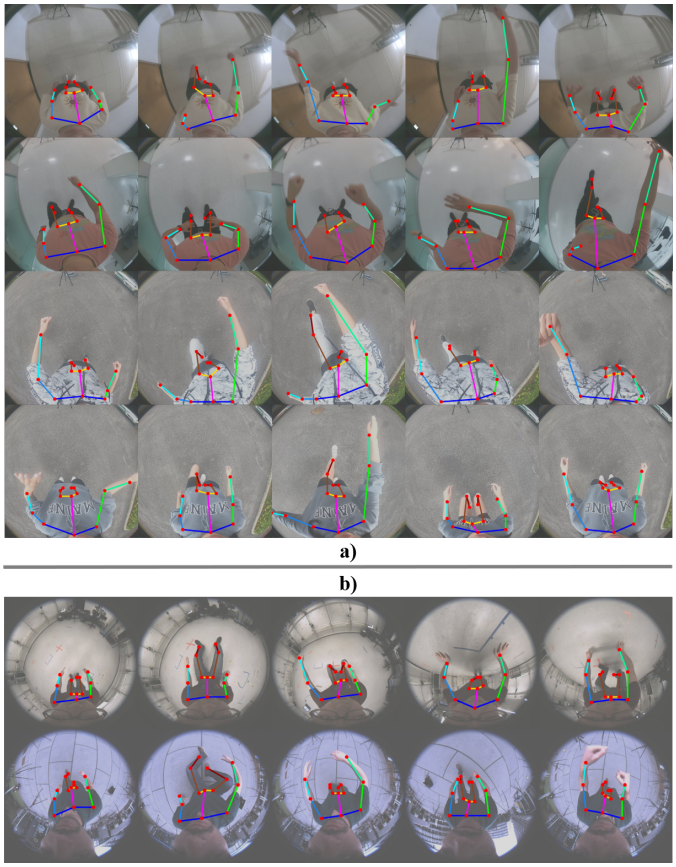

Fig. 6. Visualization results of egocentric 2D pose estimation by our proposed EgoFish3D. a) On ECHA dataset; b) On Mo^{2}Cap^2 dataset. The red points are the predicted joint positions and the colorized lines indicate the skeleton.

prior of 2D joints, the mask $\hat{M}_{c_{ego}}$ removes the heatmap with low confidence or out of human body, and the latent feature $\hat{FT}_{c_{ego}}$ preserves the original information from images and compensates for the incorrect 2D pose detection.

Can we directly use the triangulated 3D pose as the supervision? We perform the ablation study by removing the third-person-view module and direct apply the triangulated 3D pose as the supervision. As shown in model E, the performance is clearly worse than the full network architecture. This originates from the inaccurate 2D poses and trivial solution of the triangulated 3D pose in some circumstances. More importantly, with a well-trained third-person-view module, we can ease the data collection procedure by using only one third-person-view camera.

Does the coarse prior knowledge help the training of the network? The models F and G aim to highlight the significance of the prior knowledge on the 3D pose by triangulation and the rotation matrix by Aruco markers. With the help of these prior information, it can be seen that our full network (model A) can consistently improve the performance. Especially for $\mathcal{L}_{mat}^{itr}$ calculated by Aruco-based rotation, the network without this loss can hardly estimate the 3D pose under the egocentric coordinate system.

B. Qualitative Results

Fig.6 demonstrates the visualization results of egocentric 2D pose estimation on ECHA and Mo^{2}Cap^2 datasets by our proposed self-supervised learning based method. For ECHA dataset as in Fig.6(a), the proposed EgoFish3D can predict relatively accurate 2D pose even with the occlusion of lower limbs. The generalization ability of the 2D pose predicted by our model on Mo^{2}Cap^2 dataset is exhibited in Fig. 6(b), where the 2D pose estimator is trained by mixing ECHA dataset with a small number of synthetic data ($\sim$40k) in Mo^{2}Cap^2 dataset, without further finetuning on this dataset.

The egocentric 3D pose estimation results by our method on our ECHA, xR-EgoPose, and Mo^{2}Cap^2 datasets are presented in Fig. 7(a)-(c), respectively. To explore the generalization ability of our proposed method, we directly apply our trained model to the Mo^{2}Cap^2 dataset without finetuning. We retrain our EgoFish3D on xR-EgoPose [13] to conduct comparison on xR-EgoPose dataset. Given a single egocentric image captured by a fisheye lens, it can be seen that the proposed EgoFish3D can predict reasonable well 3D pose for different actions, even on unseen subjects & textures and some occluded body parts. More qualitative results can be found in our *supplementary material*.

VII. Conclusion and Future Work

This paper proposes a self-supervised egocentric pose estimation method called EgoFish3D to estimate both the 2D pose and 3D pose under the egocentric view from a single RGB image. This is achieved by leveraging the potential information from both the third-person view and the egocentric view. Specifically, the EgoFish3D incorporates three main modules: the third-person-view module, the egocentric module and the interactive module to improve the performance of our self-supervised method. This paper also proposes a real-world EgoCentric Human Action dataset called ECHA to capture the images from three different cameras circumventing the use of MoCap system to acquire the ground truth. Our experimental results demonstrate that our EgoFish3D can predict relatively accurate 2D and 3D pose. In future work, we aim to make our approach generalize well to the different placement of the egocentric camera and incorporate the human pose estimation with more egocentric vision tasks.

References

[1] S. Ardeshir and A. Borji, "Egocentric meets top-view," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 6, pp. 1353–1366, 2018.
[2] A. Bandini and J. Zariffa, "Analysis of the hands in egocentric vision: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2020.
[3] Y. Huang, X. Yang, *et al.*, "Holographic feature learning of egocentric-exocentric videos for multi-domain action recognition," *IEEE Trans. Multimedia*, 2021.
[4] H. Jiang and K. Grauman, "Seeing invisible poses: Estimating 3d body pose from egocentric video," in *Proc. CVPR*, 2017, pp. 3501–3509.
[5] M. Lu, *et al.*, "Deep attention network for egocentric action recognition," *IEEE Trans. Image Process.*, vol. 28, no. 8, pp. 3703–3713, 2019.
[6] Y. Zhang, C. Cao, *et al.*, "Egogesture: a new dataset and benchmark for egocentric hand gesture recognition," *IEEE Trans. Multimedia*, vol. 20, no. 5, pp. 1038–1050, 2018.
[7] S. Alletto, *et al.*, "Understanding social relationships in egocentric vision," *Pattern Recognit.*, vol. 48, no. 12, pp. 4082–4096, 2015.
[8] F. Ragusa, A. Furnari, *et al.*, "Ego-ch: Dataset and fundamental tasks for visitors behavioral understanding using egocentric vision," *Pattern Recognit. Lett.*, vol. 131, pp. 150–157, 2020.
[9] Z. Cao, G. Hidalgo, *et al.*, "Openpose: realtime multi-person 2d pose estimation using part affinity fields," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 1, pp. 172–186, 2019.

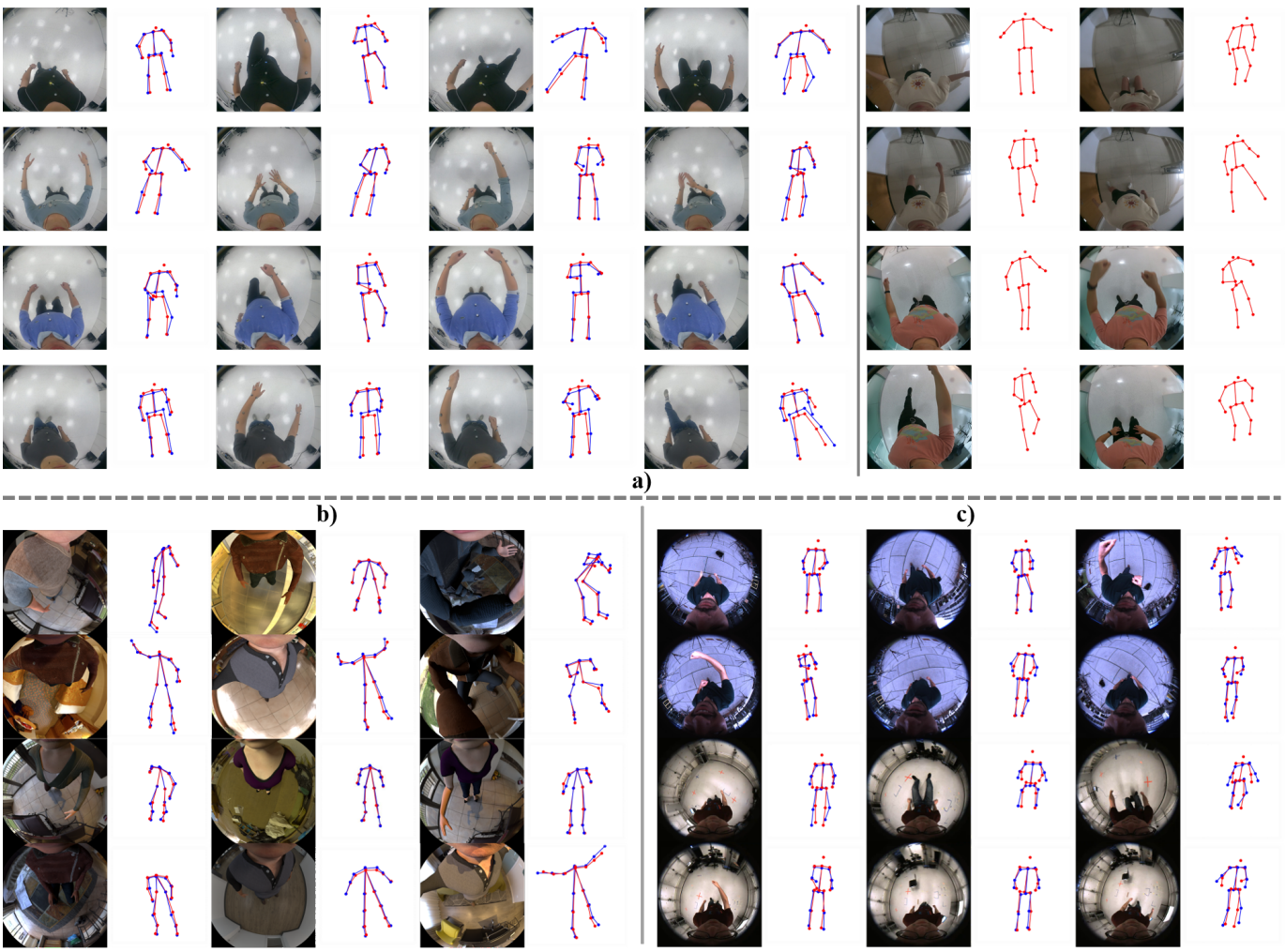

Fig. 7. Visualization of the egocentric 3D pose estimation by our EgoFish3D. a) On the ECHA dataset. Left: visualization results on the test set (two bottom rows are new subjects and all body textures are unseen in the training set); Right: visualization results on validation set. b) On xR-EgoPose dataset. We retrain our model on their dataset. c) On Mo^{2}Cap^2 dataset. We directly apply our model trained on ECHA dataset to their dataset. The red color is the predicted 3D pose by our EgoFish3D and the blue color represents the ground truth.

[10] M. Kocabas, *et al.*, "PARE: Part attention regressor for 3D human body estimation," in *Proc. ICCV*, Oct. 2021, pp. 11 127–11 137.

[11] Q. Zhang, Y. Jiang, *et al.*, "Single person dense pose estimation via geometric equivariance consistency," *IEEE Trans. Multimedia*, 2021.

[12] W. Li, H. Liu, *et al.*, "Exploiting temporal contexts with strided transformer for 3d human pose estimation," *IEEE Trans. Multimedia*, 2022.

[13] D. Tome, T. Alldieck, *et al.*, "Selfpose: 3d egocentric pose estimation from a headset mounted camera," *IEEE Trans. Pattern Anal. Mach. Intell.*, pp. 1–1, 2020.

[14] W. Xu, A. Chatterjee, *et al.*, "Mo 2 cap 2: Real-time mobile 3d motion capture with a cap-mounted fisheye camera," *IEEE Trans. Vis. Comput. Graph.*, vol. 25, no. 5, pp. 2093–2101, 2019.

[15] B. Wandt, M. Rudolph, *et al.*, "Canonpose: Self-supervised monocular 3d human pose estimation in the wild," in *Proc. CVPR*, June 2021, pp. 13 294–13 304.

[16] I. Chang, M.-G. Park, *et al.*, "Multi-view 3d human pose estimation with self-supervised learning," in *Proc. ICAIIC*. IEEE, 2021, pp. 255–257.

[17] A. Bouazizi, J. Wiederer, *et al.*, "Self-supervised 3d human pose estimation with multiple-view geometry," 2021.

[18] Y. Yuan and K. Kitani, "Ego-pose estimation and forecasting as real-time pd control," in *Proc. ICCV*, 2019, pp. 10 082–10 092.

[19] Z. Luo, R. Hachiuma, *et al.*, "Dynamics-regulated kinematic policy for egocentric pose estimation," *CoRR*, vol. abs/2106.05969, 2021.

[20] H. Rhodin, C. Richardt, *et al.*, "Egocap: egocentric marker-less motion capture with two fisheye cameras," *ACM Trans. Graph.*, vol. 35, no. 6, pp. 1–11, 2016.

[21] Y. Zhang, S. You, and T. Gevers, "Automatic calibration of the fisheye camera for egocentric 3d human pose estimation from a single image," in *WACV*, January 2021, pp. 1772–1781.

[22] H. Rhodin, J. Spörri, *et al.*, "Learning monocular 3d human pose estimation from multi-view images," in *CVPR*, 2018, pp. 8437–8446.

[23] X. Chen, K.-Y. Lin, *et al.*, "Weakly-supervised discovery of geometry-aware representation for 3d human pose estimation," in *Proc. CVPR*, 2019, pp. 10 895–10 904.

[24] M. Kocabas, *et al.*, "Self-supervised learning of 3d human pose using multi-view geometry," in *Proc. CVPR*, 2019, pp. 1077–1086.

[25] C.-H. Chen, A. Tyagi, *et al.*, "Unsupervised 3d pose estimation with geometric self-supervision," in *Proc. CVPR*, 2019, pp. 5714–5724.

[26] K. Iskakov, E. Burkov, *et al.*, "Learnable triangulation of human pose," in *Proc. ICCV*, 2019, pp. 7718–7727.

[27] U. Iqbal, *et al.*, "Weakly-supervised 3d human pose learning via multi-view images in the wild," in *Proc. CVPR*, 2020, pp. 5243–5252.

[28] H. Rhodin, *et al.*, "Unsupervised geometry-aware representation for 3d human pose estimation," in *Proc. ECCV*, 2018, pp. 750–767.

[29] "Automatic generation and detection of highly reliable fiducial markers under occlusion," *Pattern Recognit.*, vol. 47, no. 6, pp. 2280–2292, 2014.

[30] P. I. Corke, "The machine vision toolbox: a matlab toolbox for vision and vision-based control," *IEEE Robot. Autom. Mag.*, vol. 12, no. 4, pp. 16–25, 2005.

[31] P. Li, Y. Xu, Y. Wei, and Y. Yang, "Self-correction for human parsing," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2020.

[32] J. Martinez, R. Hossain, *et al.*, "A simple yet effective baseline for 3d human pose estimation," in *Proc. ICCV*, Oct 2017.