THESIS

EGOROOM: EGOCENTRIC 3D POSE ESTIMATION THROUGH MULTI-COORDINATES HEATMAPS

Submitted by

Changsoo Jung

Department of Computer Science

In partial fulfillment of the requirements

For the Degree of Master of Science

Colorado State University

Fort Collins, Colorado

Summer 2022

Master's Committee:

Advisor: Nathaniel Blanchard

Ross Beveridge
Benjamin Clegg

ABSTRACT

EGOROOM: EGOCENTRIC 3D POSE ESTIMATION THROUGH MULTI-COORDINATES HEATMAPS

Recent head-mounted virtual reality (VR) devices include fisheye lenses oriented to users' bodies, which enable full body pose estimation from video. However, traditional joint detection methods fail under this use case because fisheye lenses make joint depth information ambiguous, causing body parts to be self-occluded by the distorted torso. To resolve these problems, we propose a novel architecture, EgoRoom, that uses three different types of heatmaps in 3D to predict body joints, even if they are self-occluded. Our approach consists of three main modules. The first module transmutes the fisheye image into feature embeddings via an attention mechanism. Then, the second module utilizes three decoder branches to convert those features into a 3D coordinate system, with each branch corresponding to the xy, yz, and xz planes. Finally, the third module combines the three decoder heatmaps into the predicted 3D pose. Our method achieves state-of-the-art results on the xR-EgoPose dataset.

TABLE OF CONTENTS

ABSTRACT ii
LIST OF TABLES iv
LIST OF FIGURES v

Chapter 1 Introduction 1

Chapter 2 Related Works 4
2.1 Egocentric 3D Human Pose Estimation 4
2.2 3D Pose Estimation in space 5

Chapter 3 Methods 6
3.1 Body Attention Module 6
3.2 Multi-branch Decoder 7
3.3 3D Pose Extractor 9
3.4 Loss Function 11
3.5 Training Details 13

Chapter 4 Results 14
4.1 Evaluation on xR-EgoPose Dataset 14
4.2 Ablation Study 16
4.3 Speed by Variant 17

Chapter 5 Conclusion 19

LIST OF TABLES

4.1 Action Error Comparison 15
4.2 Joint Error Comparison 15
4.3 Ablation Study of Our Architecture 18
4.4 FPS Test for Variants of Ablation Study 18

LIST OF FIGURES

1.1 An Overview of Our Approach 2

3.1 Proposed Architecture: EgoRoom 7
3.2 Attention Module 8

4.1 Skeleton Comparison to State-of-the-art 16
4.2 Comparisons between Ground-truth and Prediction 17

INTRODUCTION

Metaverse, a virtual shared space, is posited to revolutionize long-distance interactions with others, be they meetings or hobbies. Accurately translating the position of the physical body to virtual reality (VR) via human pose estimation is key to enabling these interactions. For example, if a user is dancing, their key points need to be accurately recognized, translated to the VR context, and rendered in the Metaverse. Specifically, we focus on how body pose can be captured by a downward-facing fish-eye camera affixed to a Head Mounted Display (HMD) [12, 25, 31, 34, 37], allowing body joints to be inferred as 3D keypoints.

The accurate rendering of body orientation in VR enables a number of possibilities. First, human communication is multimodal, and because gesture is a key modality for communication [7, 20, 29, 35, 36, 38], accurately capturing body pose will enable better interactions with others in the Metaverse. Further, improving body pose rendering can allow the Metaverse to become a platform for more serious interactions — for example, an athletic trainer can virtually monitor an athlete's injury risk while they train, or lead virtual rehabilitation sessions with injured athletes [4]. Accurate egocentric pose estimation also facilitates human-AI interactions; accurate body pose information may be utilized by machine learning models to help coaches plan athletes' training plans [27].

Fisheye cameras are commonly used to capture egocentric views of body poses [12, 25, 31, 34, 37] because they make it possible to capture action from a wide perspective. Yet fisheye cameras tend to distort images and often cause body parts to be self-occluded by HMDs' emphasis on the torso. Traditional human body pose estimation fails because of the domain mismatch; for instance, an image of a straight arm can appear as a curved arm through the fisheye lens, thus necessitating a real-time solution for human body pose estimation.

Traditionally, body-centered pose estimation [1, 13, 31, 37] rather than a global pose [25, 34] struggles with changes in direction of movement, such as identifying that a figure is turning around. Estimating 3D poses in global space is more challenging because it requires both localization and mapping techniques, such as ORB-SLAM 2 [19, 34] in addition to processing local

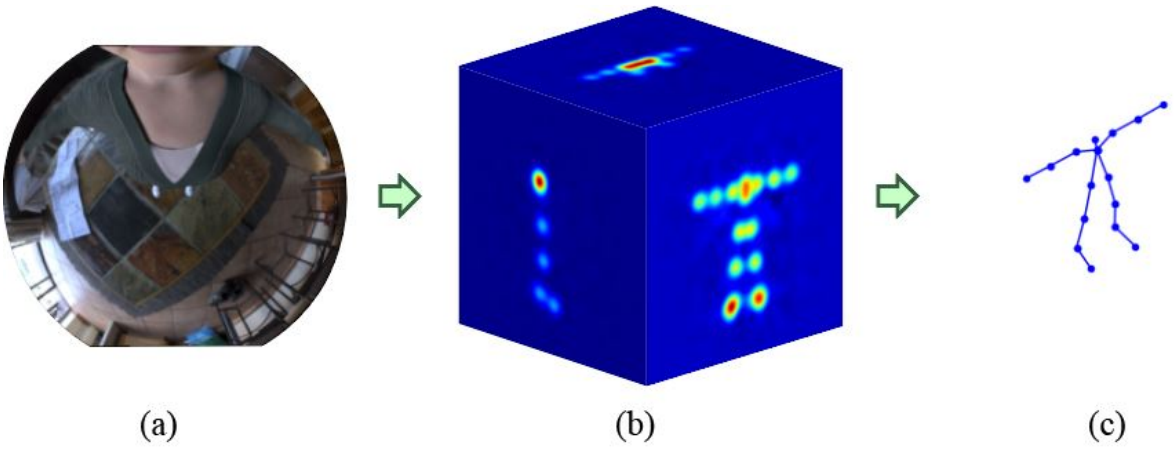

Figure 1.1: An overview of our approach to predicting a 3D human pose from an egocentric view image (a). Our first two modules, the Attention Module and the Multi-branch Decoder, predict key body points in three points of view. The outputs of the Multi-branch Decoder are body heatmaps on viewpoints of xy, yz, and xz planes (b). The predicted body heatmaps are transferred to 16 body keypoints into a 3D coordinate system through the 3D pose extractor. This module generates a figure including both ground truth and our prediction (c). In the figure, the blue skeleton represents ground truth and the red skeleton shows our prediction. Each axis length of (b) is 280 cm. The exact volume, which measures $280\,cm \times 280\,cm \times 280\,cm$, is EgoRoom (Chapter 4)

poses for location in global space. Our method predicts body joints for a specific human-centered space, making it more constrained than traditional global space estimation, while still overcoming traditional limitations in body-centered methods.

To address such issues related to image distortion and self-occlusion in body pose image detection, we propose a novel, real-time approach using a *Multi-branch Decoder*, visualized in Figure Figure 1.1. Our neural network produces three different types of heatmaps through two different 3D coordinate systems. We found that our approach outperforms the previous egocentric state-of-the-art result on xR-EgoPose [32]. Here are our contributions:

- We propose a novel approach for unstable estimation that corrects for image distortion and self-occlusion.
- We achieve state-of-the-art results for egocentric 3D pose estimation.
- Our solution runs in real-time, enabling real-world use.

- Ours is the first study about egocentric body pose estimation in a specific volume: $280\,cm \times 280\,cm \times 280\,cm$.

We organize the remainder of this paper as follows. First, we discuss relevant related work (Chapter 2). We then describe our approach and architecture (Chapter 3), and present our state-of-the-art, real-time results compared with others (Chapter 4). Finally, we discuss and conclude our findings (Chapter 5).

RELATED WORKS

A multitude of work has been done on the broad area of 3D human pose estimation; however, here we focus on the more constrained problem of 3D human pose estimation from an egocentric perspective. Typically, the egocentric body is captured with a fisheye lens rather than a traditional lens. This is because fisheye lenses provide a larger field of view than traditional lenses, allowing access to more body information, and thus the potential for improved accuracy in body pose estimation [12, 25, 34, 37, 40]. We similarly assess our method on a monocular fisheye camera dataset [32] and describe sensor-less human pose estimation on a specific space.

2.1 Egocentric 3D Human Pose Estimation

Several approaches to egocentric human pose estimation have been explored through varying settings of camera, types of datasets, or neural architectures. The word *egocentric* has mainly been interpreted in two ways. The "inside-out" view captures the egocentric perspective through a camera mounted on a person's chest [12, 13, 26] or head [39]. The "heads-down" view, on the other hand, refers to the egocentric perspective captured from a downward camera mounted on an HMD [31, 34, 40] or a helmet [25, 37]. Additional parameters for human pose estimation include the stereo and the monocular viewpoints. Stereo viewpoint provides greater depth and perspective than the monocular field of view [6, 25]. However, monocular viewpoint requires less computational knowledge and can be more easily installed than stereo viewpoint [31, 34, 37, 40]. In addition, egocentric datasets consist of multiple types: synthetic data [32, 37], real-world data [37], and global space data [34]. Synthetic data facilitates model training, but real-world datasets are typically required to fine-tune model since training on synthetic data alone rarely generalizes to real-world performance. In specific conditions that require human pose estimation globally, such as tracking of an athlete for preventing injuries [4], a model which is trained on the global space dataset is useful.

Two main approaches have been considered for obtaining 3D human keypoints. One approach is a directed 3D pose prediction from an image [10, 14, 17, 21, 30, 40, 43], and the other approach

is a two-step strategy which predicts 3D pose from estimated 2D joints [2, 3, 5, 12, 16, 18, 22, 24, 28, 31, 37, 41, 42]. Here, we propose a three-step approach which predicts 3D pose directly using monocular fisheye data to identify body joints. Our approach runs in real-time, an essential feature for real-world VR applications such as human-human communication.

2.2 3D Pose Estimation in space

Typically, egocentric human pose is expressed as self-centered pose and interpreted as human-centered plots in local space [13, 31, 40]. While some works [25, 34, 39] have attempted to estimate egocentric human pose in global space, the predictions are unstable because they are estimating across large areas, and SLAM [19] is required. Our approach predicts 3D human poses in a human-centered volume, which allows for prediction of occluded or out-of-frame joints since spatial information covers the limitations of egocentric view.

METHODS

An overview of our architecture for 3D human pose estimation is depicted in Figure 3.1. Our three-step approach involved three main modules: the Attention Module, the Multi-branch Decoder, and the 3D Pose Extractor. In the first step, the Attention Module estimated a focal point in the input image from which it extracted features from the input image and a highlighted image. Next, the Multi-branch Decoder used these extracted features to predict three types of heatmaps based upon two sets of 3D coordinates. Each type of heatmap implied distinguishable body information that appeared in a different perspective. Finally, the 3D Pose Extractor used the predicted three heatmaps to produce keypoints of an actual 3D pose.

3.1 Body Attention Module

In our approach, we utilized the Attention mechanism [9, 23, 33] to gather information about the features of the highlighted body. Our model predicted 3D body keypoints directly rather than basing predictions upon 2D pose estimations. We proposed that this direct approach would produce more accurate predictions of body pose position.

This module took a normalized RGB image $I \in \mathbb{R}^{445\times445\times3}$ as input. To normalize images, we subtracted their means and then divided by their standard deviations. The Attention Module consisted of the global and local branches, as described in Figure 3.2. The global branch produced a mask $M \in \mathbb{R}^{95\times95\times1}$ that highlighted the body of the RGB image I and applied the features used to generate the mask to the fully-connected layers. The global branch is composed of a pretrained layer *ResNet* [11], a pooling layer, a convolutional layer, and five subsequent deconvolutional layers. The output of the deconvolutional layers $H \in \mathbb{R}^{95\times95\times16}$ produced 16 heatmaps which corresponded to the positions of body-joints in the input image. Each heatmap H is merged as $\hat{H} \in \mathbb{R}^{95\times95\times1}$ with a maximum value of one for multiplying element-wisely with the small size of input image in the local branch.

In the local branch, the input image I is resized to a small image $\hat{I} \in \mathbb{R}^{95\times95\times3}$ to be multiplied by the mask M. This step directs focus to the body by precluding the image background. The

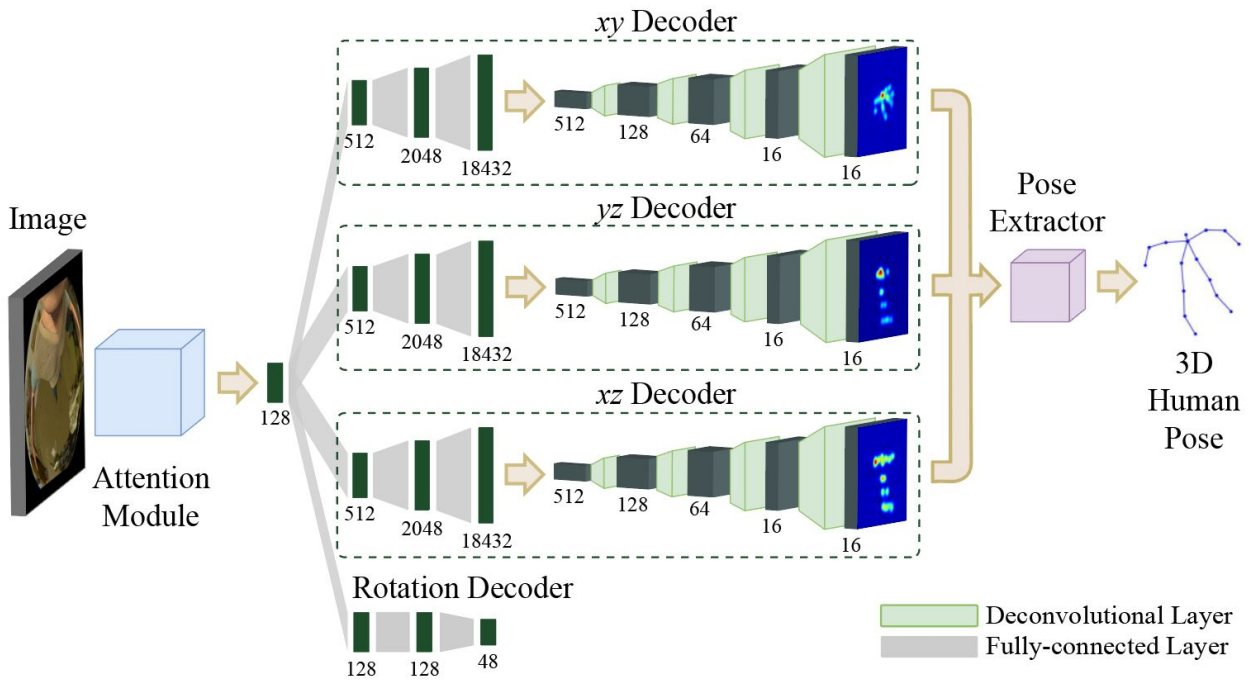

Figure 3.1: Our proposed architecture: an input image is passed through the Attention Module (described in Section 3.1), which produces embeddings in a latent space $\hat{ls}$. These embeddings are subsequently processed through a Multi-branch Decoder that predicts three different types of heatmaps with rotation vectors for each body-joint. The 3D pose extractor then extracts the position and rotation of each joint in 3D using the outputs of decoders.

highlighted body image plugs into a pretrained layer *ResNet* [11] and then into a pooling layer. The local branch output is flattened and added to the global branch output. The merged embeddings of both branches are applied to three fully-connected linear layers to generate a latent space $\hat{ls}$. Note that the last two layers of the *ResNet* [11] were discarded to manipulate the embeddings to the fully-connected layers.

3.2 Multi-branch Decoder

This module interprets human poses in 3D coordinate systems based upon the output of the Body Attention Module (Section 3.1). The merged features of the global and local branches are extracted to the small embeddings $\mathbb{R}^{128}$ by the fully-connected layers. The embeddings are delivered into the module as input, which the module uses to produce three distinct types of heatmaps. Each type of heatmap $\widehat{HM_{3D}} \in \mathbb{R}^{126\times126\times16}$ presents body joints from a different viewpoint of 3D coordinate system. By predicting three types of heatmaps, the Multi-branch Decoder clarifies each

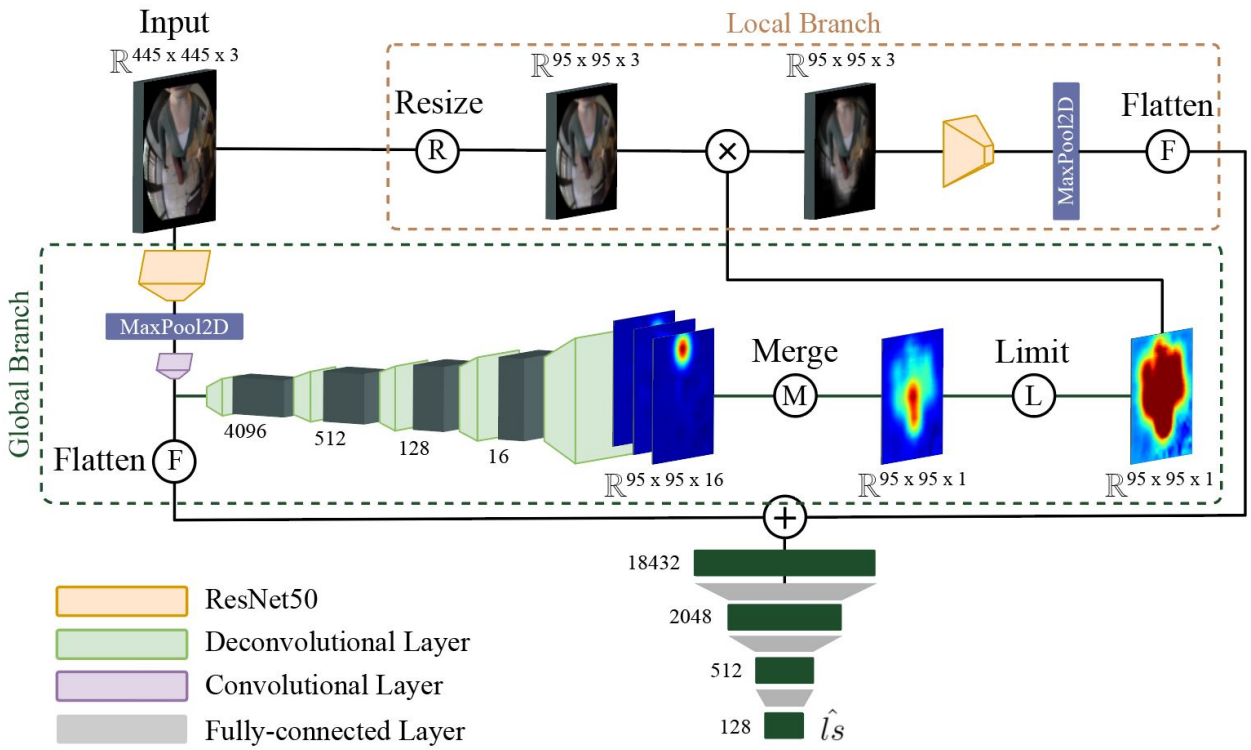

Figure 3.2: Details of Attention Module. An input image is applied to global and local branches. In the global branch, a mask is generated to highlight a body of the image. The features that are used to produce the mask are flattened to merge with the features of the local branch. The local branch extracts features from the body-focused small image. The outputs of both branches are combined to plug in the fully-connected layers.

body joint and verifies the predictions simultaneously through three points of view, in turn resolving self-occlusion and image distortion issues associated with downward-facing fisheye camera body pose capture.

To produce the output, we used a novel Multi-branch Decoder, with each branch generating heatmaps and rotation vectors of each body-joint. The embedding latent space $\hat{ls}$ passes through three fully-connected layers and three linear layers with LeakyReLU [15]. We used the 3D coordinate system, built with x, y, and z coordinates, to link the three fully-connected layers to the three branches which are xy decoder, yz decoder, and xz decoder. The outputs of these decoders are heatmaps that correspond to views of xy, yz, and xz planes. For example, the heatmaps of the view of xy plane are generated by the xy decoder. Note the outputs of the three branches benefit

when occluded body joints from one view are visible on another. We also include the prediction of rotation vectors from the Rotation Decoder.

Let the training set T have the set of body joints BJ and $BJ \subset T$, then a i-th joint $(x_i, y_i, z_i) \in BJ$. One of the ground truth joints of 3D Pose $(x_i\,', y_i\,', z_i\,') \in BJ_{gt}$ was defined as:

$$\begin{aligned} x_i\,' &= \lfloor \frac{x_i \times 126}{280} \rceil + 63 \\ y_i\,' &= \lfloor \frac{y_i \times 126}{280} \rceil + 63 \\ z_i\,' &= \lfloor \frac{\big(z_i + 80\big) \times 126}{280} \rceil \end{aligned} \tag{3.1}$$

We divided each coordinate value by an approximated value $Space_{max}$, 280.0 cm, which includes margins of 20 cm for generating heatmaps, calculated by:

$$Space_{max} = \max \Big\{ \Big(\max_{\forall x \in X} - \min_{\forall x \in X} \Big), \Big(\max_{\forall y \in Y} - \min_{\forall y \in Y} \Big), \Big(\max_{\forall z \in Z} - \min_{\forall z \in Z} \Big) \Big\} + 20 \quad \text{where } X, Y, Z \in BJ \tag{3.2}$$

We chose each maximum and minimum value based on the training set ground truth. We multiplied by 126 to reduce the length of ranges of all axes to 126 cm. We then added 63 to x and y axes and added 80 to z axis to match all the range of axes from 0 to 126.0 (Eq. 3.1). A set of new body joints BJ_{gt} were used to make heatmaps for loss functions (Section 3.4). The three types of heatmaps provided different viewpoints of three coordinate systems and these views were identical to xy, yz, and xz planes.

3.3 3D Pose Extractor

The extractor calculated 3D body keypoints from the outputs of three decoders (Eq. 3.1) using heatmaps and computed human body joints in 3D. Each output of the decoder contains body-joint position information in two different coordinate systems. The extractor provides predictions by merging the results from each coordinate.

Let the outputs of the Multi-branch Decoder be $\widehat{HM_{xy}}$, $\widehat{HM_{yz}}$, and $\widehat{HM_{xz}}$. A predicted i-th body joint $(\hat{x_i}, \hat{y_i}, \hat{z_i})$ is calculated as:

$$
\begin{aligned}
I_{xy_x}, I_{xy_y} &= I\big(\widehat{HM_{xy_i}}\big) \\
I_{yz_y}, I_{yz_z} &= I\big(\widehat{HM_{yz_i}}\big) \\
I_{xz_x}, I_{xz_z} &= I\big(\widehat{HM_{xz_i}}\big)
\end{aligned}
\qquad \text{with} \qquad (3.3)
$$

$$
I(hm) = \big\{\text{index of } v\big\}
$$

where

$$
hm \in HM, \text{ value } v \in hm \text{ and } v \geq 0.95 \max_{hm}
$$

We calculated the predicted position of each body keypoint using the indices of values greater than 95% of the maximum value of each heatmap. We estimated actual 3D body points using the set of indices for each coordinate.

$$
\begin{aligned}
\tilde{x} &= \frac{1}{n_x} \sum_{i=0}^{n_x - 1} I_{x_i} \text{ where } I_x = I_{xy_x} \cup I_{xz_x} \\
\tilde{y} &= \frac{1}{n_y} \sum_{i=0}^{n_y - 1} I_{y_i} \text{ where } I_y = I_{xy_y} \cup I_{yz_y} \\
\tilde{z} &= \frac{1}{n_z} \sum_{i=0}^{n_z - 1} I_{z_i} \text{ where } I_z = I_{yz_z} \cup I_{xz_z}
\end{aligned}
\qquad (3.4)
$$

Note that n_x, n_y, and n_z are numbers of elements of I_x, I_y, and I_z. We calculated the mean of the union of two sets which hold indices in each coordinate. For example, I_x represents predicted positions in the x axis of EgoRoom (Figure 1.1 on (c)). The mean of the index set of x coordinate I_x inferred the highest possibility of x's position of the corresponding joint. A 3D point $(\tilde{x}, \tilde{y}, \tilde{z})$ can be described as a predicted joint with the highest suspicion in the three-coordinate system. We then transferred the predicted joint to a $126\,cm \times 126\,cm \times 126\,cm$ space, which is a pose-centered volume.

$$\begin{aligned} \hat{x_i} &= \frac{(\tilde{x} - 63) \times 280}{126} \\ \hat{y_i} &= \frac{(\tilde{y} - 63) \times 280}{126} \\ \hat{z_i} &= \frac{\tilde{z} \times 280}{126} - 80 \end{aligned} \tag{3.5}$$

To predict body keypoints, we reversed equation 3.1; the module predicted 3D human body pose in a volume of $280\,cm \times 280\,cm \times 280\,cm$. The length of 280 cm enables the module to predict actions such as upper stretching or lower stretching.

3.4 Loss Function

Our whole architecture L_{all} consists of five small loss functions related to Attention Mask M, xy heatmaps, yz heatmaps, xz heatmaps, 3D body keypoints, bone lengths, bone angles, and rotation vectors. The bone length and bone angle predictions were based upon the predictions of 3D body keypoints produced by the 3D pose extractor. We trained the entire architecture with the loss function L_{all}. The main loss function L_{all} is described as:

$$L_{all} = \lambda_{HM} L_{HM} + \lambda_P L_P + \lambda_\ell L_\ell + \lambda_\theta L_\theta + \lambda_r L_r \tag{3.6}$$

where λ_{HM}, λ_P, λ_ℓ, λ_θ and λ_r are weights of loss functions that corresponded to 0.2, 0.1, 0.1, 10 and 0.05 respectively.

$$\begin{aligned} L_{HM} = \|M - \widehat{M}\|^2 + \|HM_{xy} - \widehat{HM_{xy}}\|^2 \\ + \|HM_{yz} - \widehat{HM_{yz}}\|^2 + \|HM_{xz} - \widehat{HM_{xz}}\|^2 \end{aligned} \tag{3.7}$$

L_{HM} is a sum of mean squared errors of four different heatmap types: Attention Mask M, xy heatmaps, yz heatmaps, and xz heatmaps.$\widehat{\ }$ denotes a prediction. For example, $\widehat{M}$ is a Attention Mask prediction from the Attention Module (Section 3.1). $\widehat{HM_{xy}}$, $\widehat{HM_{yz}}$, and $\widehat{HM_{xz}}$ are outputs of the three decoders in the Multi-branch Decoder (Section 3.2).

$$L_P = L_{torso} + L_{limb}$$

where

$$L_{torso} = \sum_{t}^{torso} \big(P_t - \hat{P}_t\big)^2 \tag{3.8}$$
$$L_{limb} = \sum_{l}^{limb} 4\big(P_l - \hat{P}_l\big)^2$$

We applied the extracted body keypoints P (see Eq. 3.5) in the keypoint loss function L_P. The loss function L_P was expressed as a sum of difference of torso and difference of limbs. We weighted limbs (see Eq. 3.8) to make limb prediction more robust. Since depth ambiguity on egocentric images can mis-estimate the precise position of legs, keypoints of limbs tended to produce higher rates of error than keypoints of torso.

$$L_\ell = \sum_{\ell}^{L} \big|\ell - \hat{\ell}\big| \tag{3.9}$$

We used absolute difference to calculate the loss of bone length. L is a set of bone lengths and ℓ where $\ell \in L$.

$$L_\theta = \sum_{b}^{B} \big|\frac{1}{\pi} \arccos\Big(\frac{\vec{b} \cdot \hat{\vec{b}}}{\big|\vec{b}\big|\big|\hat{\vec{b}}\big|}\Big)\big| \tag{3.10}$$

A bone b includes a set of bones B, and $\vec{b}$ is described as a bone vector. We applied cosine similarity to estimate the distance between two vectors. We calculated an absolute angle θ between two bone vectors.

$$L_r = \frac{1}{N_r} \sum_{r}^{R} \|r - \hat{r}\|^2 \tag{3.11}$$

A r denotes a rotation vector of a body-joint. N_r is a number of rotation vectors. L_r is expressed as an mean squared difference between the ground truth $\hat{r}$ and the prediction of the Rotation Decoder r.

Although L_P and L_{HM} produced the largest impacts in optimizing our model's performance, we proposed to use L_ℓ, L_θ, and L_r for more accurate estimations. We also used additional loss functions L_ℓ, L_θ, and L_r to calculate the confidence intervals for our predictions. The low values of L_ℓ, L_θ, and L_r indicate that our predictions were close to the ground truth. For increased accuracy, we relied on these calculations rather than only on L_P and L_{HM}.

3.5 Training Details

We trained our model on xR-EgoPose's training set [32], and used five epochs to calculate the state-of-the-art result with $1e^{-4}$ learning rate and 6 batches. The deconvolutional layers, fully-connected layers, and Multi-branch Decoder were initialized by *Xavier* [8]. The convolutional layer, pooling layers, and deconvolutional layers of the Attention Module had kernel size = 3 and stride = 2. All deconvolutional layers of the Multi-branch Decoder used kernel size = 4 and stride = 2. All the fully-connected layers except for the last layer had a leaky ReLU with 0.2 leakiness for an activation function. The last layers were calculated right before the latent space $\hat{ls}$ and deconvolutional layers of the Multi-branch Decoder.

RESULTS

We evaluated our model based on the performance of xR-EgoPose [32]. The xR-EgoPose test set excluded 12 synthetic subjects, including five females and seven males. Subjects in the test set were not included in training, and test subjects featured distinguishable characteristics from subjects in the training set.

The name "EgoRoom" refers to the fact that our model operates in a predictable space in an egocentric view to predict a 3D human pose.

To evaluate methods, we used Mean Per Joint Position Error (MPJPE). The MPJPE is expressed as:

$$MPJPE = \frac{1}{N_f}\frac{1}{N_j}\sum_{f=1}^{N_f}\sum_{j=1}^{N_j}\|P - \hat{P}\|_2 \tag{4.1}$$

where P and $\hat{P}$ denote keypoints of ground truth and prediction at f-th frame and j-th joint.

4.1 Evaluation on xR-EgoPose Dataset

We evaluated our proposed method by comparing to prior state-of-the-art result on the test set of xR-EgoPose [32]. Our qualitative results (Figure 4.1 and 4.2) show that our model's prediction is close to ground truth, and consistently outperforms other methods.

In our quantitative comparisons, EgoRoom outperformed the state-of-the-art result (Table 4.1 and 4.2). Table 4.1 showed that EgoRoom outperformed the state-of-the-art by more than 2% across all frames in the test set. However, when considering EgoRoom's performance across all actions, EgoRoom provides substantially better performance across a multitude of actions, and performance is far more consistent than in other methods (showcased by the low standard deviation). Even in cases where EgoRoom performs slightly worse than other methods (upper stretching, walking), performance is near equivalent. In particular, EgoRoom showed vastly improved results for actions like gaming, gesticulating, and greeting.

In Table 4.2, we examined how well EgoRoom performed in the context of predicting individual joints. Again, we find that EgoRoom produces substantially better results, which are also far

Table 4.1: Comparison with a state-of-the-art approach, Tome et al. [31], on the xR-EgoPose dataset. Our approach, EgoRoom, outperformed the previous state-of-the-art results on most actions except Upper Stretching and Walking, and, across actions, is far more consistent than other methods.

Action	Martinez [16]	Tome [31] - U-Net	**EgoRoom**
Gaming	109.6	52.5	**35.0**
Gesticulating	105.4	49.2	**32.3**
Greeting	119.3	72.0	**36.8**
Lower Stretching	125.8	37.3	37.3
Patting	93.0	53.0	**47.9**
Reacting	119.7	44.4	**34.2**
Talking	111.1	46.1	**32.4**
Upper Stretching	124.5	**39.3**	42.7
Walking	130.5	**37.2**	38.8
All Test Frames (mm)	122.1	41.0	**40.0**
Action Average (mm)	115.43	46.78	**37.49**
Action SD (mm)	11.76	12.49	**5.09**

Table 4.2: Joint error comparison with a state-of-the-art approach. Each value denotes an error distance in mm for each joint. Our approach, EgoRoom, outperformed on most joints except Left Leg, Right Leg, and Neck. EgoRoom showed drastic performance on Elbows and Hands.

Joint	Tome [31] ResNet50 (mm)	EgoRoom (mm)	Joint	Tome [31] ResNet50 (mm)	EgoRoom (mm)
Left Leg	**34.33**	38.23	Right Leg	**33.85**	38.26
Left Knee	62.57	**45.58**	Right Knee	61.36	**45.95**
Left Foot	70.08	**54.38**	Right Foot	68.17	**56.64**
Left Toe	76.43	**65.84**	Right Toe	71.94	**66.48**
Neck	**6.57**	16.27	Head	23.20	**11.81**
Left Arm	31.36	**22.91**	Right Arm	31.45	**21.56**
Left Elbow	60.89	**30.77**	Right Elbow	50.13	**30.79**
Left Hand	90.43	**46.49**	Right Hand	78.28	**47.75**
Joint Average	53.19	**39.98**	Joint SD	23.65	**16.71**

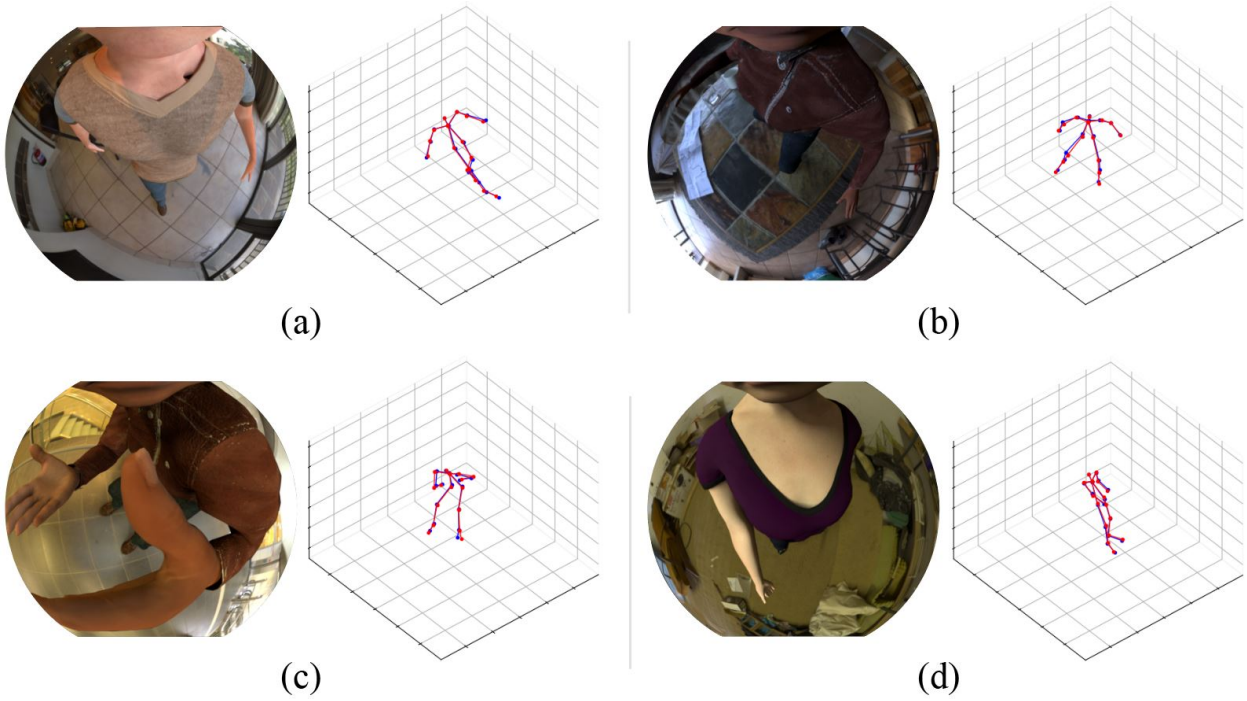

Figure 4.1: Qualitative results on xR-EgoPose [32] dataset. The left image is an input image and the right is the predicted (red) and ground truth (blue) pose. Our models predicted the the whole body precisely in all examples despite some of the body parts being self-occluded by torsos or hands.

more consistent than prior methods. Our approach more accurately predicted most joints while showcasing dramatic improvements on Elbows and Hands, a key target of our model since these body parts are often occluded. Notably, we find that prior methods outperform our model on Neck, but we believe Neck can be appropriately estimated using the head-mounted display (HMD) to which the camera camera will be affixed, making it a far less important joint to estimate.

4.2 Ablation Study

We conclude our results with an ablation study to investigate the influence of EgoRoom's modules/components (Table 4.3). Table 4.3 shows how final performance changes when components are added — specifically, we find that each component is essential for achieving substantial performance gains. For variant 1 to 5, we modified the last component of the linear layer to predict all body keypoints. For variants 1 and 2, the Attention Module and Multi-branch Decoder modules were replaced by a linear layer, and $\boldsymbol{L_\theta}$ was excluded to prevent untrainable losses. For the variant 3, 4, and 5, we added a linear layer to replace the decoders. For example, variant 3 needs a linear

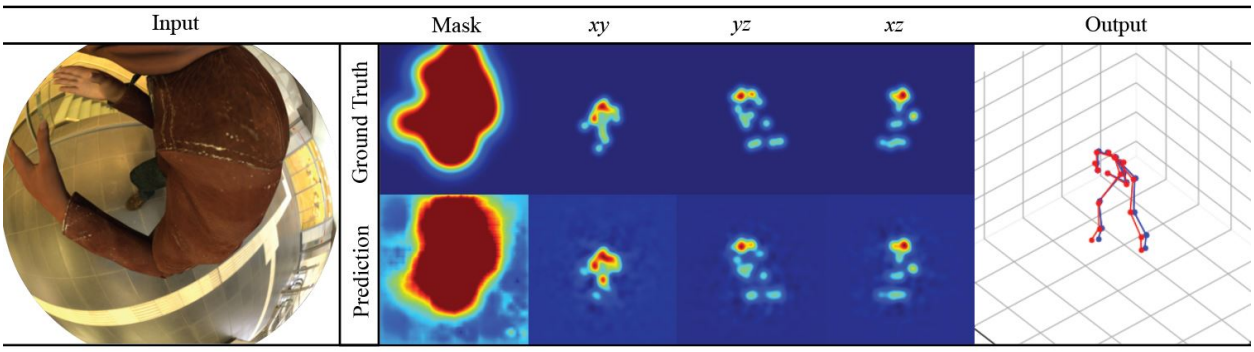

Figure 4.2: Comparison between ground truth and predictions of each step. An input image introduced into the Attention Module produces a mask for focusing an egocentric body. Then, the Multi-branch Decoder predicts three types of heatmaps that are shown in different viewpoints. xy, yz, and xz are predictions of 3D human pose in three viewpoints which are xy, yz, and xz planes. The 3D pose extractor refines the outputs of the Multi-branch Decoder into 3D body keypoints. The skeletons on the far right depict an expressed graph based on the predicted body keypoints. The red skeleton denotes our prediction and the blue skeleton represents ground truth.

layer to predict z coordinate of body points since it can get information of x and y from the $\boldsymbol{xy}$ decoder. variant 3 and 4 showcased that using two types of heatmaps from decoders improved results, and spatial information from two different viewpoints helps identify the joints in 3D. Finally, all heatmaps with rotation vectors of body-joints, our final model, produced the best performance.

4.3 Speed by Variant

We proposed variants (Section 4.2) to measure testing speed. We evaluated frames per second (FPS) on a RTX 3090 with a 10 core AMD Ryzen Threadripper 3960X. Our results show that all variants can predict egocentric body pose in real-time 4.4. Variant 10—which represents EgoRoom—can predict egocentric 3D human pose in real-time, enabling a multitude of real-world use applications such as embodied VR, multimodal VR interactions, and human-AI collaboration.

Table 4.3: Ablation study for our whole architecture. We compared variants based on Mean Per Joint Position Error (MPJPE) by ablating components of EgoRoom. Attention Module and Multi-branch Decoder were advanced to improve the performance through the variant 1 to 9. The variant 10 showed the rotation vectors significantly made better result.

Experiments	Attention Module	xy Decoder	yz Decoder	xz Decoder	Rotation Decoder	MPJPE (mm)
Variant 1						271.1
Variant 2	✓					71.2
Variant 3	✓	✓				56.8
Variant 4	✓		✓			56.4
Variant 5	✓			✓		47.8
Variant 6	✓		✓	✓		50.6
Variant 7	✓	✓		✓		50.3
Variant 8	✓	✓	✓			52.0
Variant 9	✓	✓	✓	✓		50.6
Variant 10	✓	✓	✓	✓	✓	**40.0**

Table 4.4: FPS of Variants on Testing. Variant numbers in this table correspond to those listed in 4.3. As our results show, EgoRoom can predict real-time body poses with precise estimation. The results collected for variants 3 to 10 indicate that these models function in real-time on 30 FPS video capture.

Variant	1	2	3	4	5	6	7	8	9	10
FPS	330	62	56	56	55	52	51	52	48	49

CONCLUSION

In this paper, we introduced a three-step approach for egocentric 3D human pose estimation that runs in real-time and achieves state-of-the-art results. Our architecture, EgoRoom, can predict centered human poses in 3D even if input images are distorted or include instances of self-occlusion. Specifically, our three-step architecture is composed of three modules: the Attention Module, which refines an egocentric image to feature embeddings; the Multi-branch Decoder Module, which transfers these embeddings to three different types of heatmaps; and the 3D Pose Extractor Module, which uses the heatmaps to identify 3D keypoints to estimate human poses. The multi-coordinate heatmaps predict body poses in three different views, enabling the model to estimate occluded joints on spatial information of a specific volume. We showcased the importance of each component with an ablation study, which compared our prediction performance and speed across multiple variants.

REFERENCES

[1] Ajanohoun, J., Paquette, E., & Vázquez, C. (2021). Multi-View Human Model Fitting Using Bone Orientation Constraint and Joints Triangulation [ISSN: 2381-8549]. *2021 IEEE International Conference on Image Processing (ICIP)*, 1094–1098. https://doi.org/10.1109/ICIP42928.2021.9506718

[2] Akhter, I., & Black, M. J. (2015). Pose-Conditioned Joint Angle Limits for 3D Human Pose Reconstruction. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1446–1455.

[3] Alldieck, T., Magnor, M., Xu, W., Theobalt, C., & Pons-Moll, G. (2018). Video Based Reconstruction of 3D People Models. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 8387–8397.

[4] Blanchard, N., Skinner, K., Kemp, A., Scheirer, W., & Flynn, P. (2019). "Keep Me In, Coach!": A Computer Vision Perspective on Assessing ACL Injury Risk in Female Athletes [ISSN: 1550-5790]. *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 1366–1374. https://doi.org/10.1109/WACV.2019.00150

[5] Bogo, F., Kanazawa, A., Lassner, C., Gehler, P., Romero, J., & Black, M. J. (2016). Keep It SMPL: Automatic Estimation of 3D Human Pose and Shape from a Single Image. In B. Leibe, J. Matas, N. Sebe, & M. Welling (Eds.), *Computer Vision – ECCV 2016* (pp. 561–578). Springer International Publishing.

[6] Cha, Y.-W., Shaik, H., Zhang, Q., Feng, F., State, A., Ilie, A., & Fuchs, H. (2021). Mobile. Egocentric Human Body Motion Reconstruction Using Only Eyeglasses-mounted Cameras and a Few Body-worn Inertial Sensors [ISSN: 2642-5254]. *2021 IEEE Virtual Reality and 3D User Interfaces (VR)*, 616–625. https://doi.org/10.1109/VR50410.2021.00087

[7] Fröhlich, M., Sievers, C., Townsend, S. W., Gruber, T., & van Schaik, C. P. (2019). Multimodal communication and language origins: Integrating gestures and vocalizations [Publisher: Wiley Online Library]. *Biological Reviews*, *94*(5), 1809–1829.

[8] Glorot, X., & Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks [ISSN: 1938-7228]. *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, 249–256. Retrieved November 14, 2021, from https://proceedings.mlr.press/v9/glorot10a.html

[9] Guan, Q., Huang, Y., Zhong, Z., Zheng, Z., Zheng, L., & Yang, Y. (2018). Diagnose like a radiologist: Attention guided convolutional neural network for thorax disease classification. *arXiv preprint arXiv:1801.09927*.

[10] Guo, Y., Zhao, L., Zhang, S., & Yang, J. (2019). Coarse-to-Fine 3D Human Pose Estimation. In Y. Zhao, N. Barnes, B. Chen, R. Westermann, X. Kong, & C. Lin (Eds.), *Image and Graphics* (pp. 579–592). Springer International Publishing.

[11] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep Residual Learning for Image Recognition [ISSN: 1063-6919]. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 770–778. https://doi.org/10.1109/CVPR.2016.90

[12] Hwang, D.-H., Aso, K., & Koike, H. (2019). MonoEye: Monocular Fisheye Camera-based 3D Human Pose Estimation [ISSN: 2642-5254]. *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, 988–989. https://doi.org/10.1109/VR.2019.8798267

[13] Jiang, H., & Grauman, K. (2017). Seeing Invisible Poses: Estimating 3D Body Pose from Egocentric Video [ISSN: 1063-6919]. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 3501–3509. https://doi.org/10.1109/CVPR.2017.373

[14] Li, S., & Chan, A. B. (2015). 3D Human Pose Estimation from Monocular Images with Deep Convolutional Neural Network. In D. Cremers, I. Reid, H. Saito, & M.-H. Yang (Eds.), *Computer Vision – ACCV 2014* (pp. 332–347). Springer International Publishing.

[15] Maas, A. L., Hannun, A. Y., Ng, A. Y. et al. (n.d.). Rectifier nonlinearities improve neural network acoustic models.

[16] Martinez, J., Hossain, R., Romero, J., & Little, J. J. (2017). A Simple Yet Effective Baseline for 3d Human Pose Estimation [ISSN: 2380-7504]. *2017 IEEE International Conference on Computer Vision (ICCV)*, 2659–2668. https://doi.org/10.1109/ICCV.2017.288

[17] Mehta, D., Rhodin, H., Casas, D., Fua, P., Sotnychenko, O., Xu, W., & Theobalt, C. (2017). Monocular 3D Human Pose Estimation in the Wild Using Improved CNN Supervision [ISSN: 2475-7888]. *2017 International Conference on 3D Vision (3DV)*, 506–516. https://doi.org/10.1109/3DV.2017.00064

[18] Moreno-Noguer, F. (2017). 3D Human Pose Estimation From a Single Image via Distance Matrix Regression. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2823–2832.

[19] Mur-Artal, R., Montiel, J. M. M., & Tardós, J. D. (2015). ORB-SLAM: A Versatile and Accurate Monocular SLAM System [Conference Name: IEEE Transactions on Robotics]. *IEEE Transactions on Robotics*, *31*(5), 1147–1163. https://doi.org/10.1109/TRO.2015.2463671

[20] Oh, J., & Kim, G. J. (2020). Effects of gestural exaggeration to user experience in virtual reality. *The 25th International Conference on 3D Web Technology*, 1–4.

[21] Park, S., Hwang, J., & Kwak, N. (2016). 3D Human Pose Estimation Using Convolutional Neural Networks with 2D Pose Information. In G. Hua & H. Jégou (Eds.), *Computer Vision – ECCV 2016 Workshops* (pp. 156–169). Springer International Publishing.

[22] Pavlakos, G., Choutas, V., Ghorbani, N., Bolkart, T., Osman, A. A. A., Tzionas, D., & Black, M. J. (2019). Expressive Body Capture: 3D Hands, Face, and Body From a Single Image. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 10975–10985.

[23] Ramachandran, P., Parmar, N., Vaswani, A., Bello, I., Levskaya, A., & Shlens, J. (2019). Stand-alone self-attention in vision models. *Advances in Neural Information Processing Systems*, *32*.

[24] Ramakrishna, V., Kanade, T., & Sheikh, Y. (2012). Reconstructing 3D Human Pose from 2D Image Landmarks. In A. Fitzgibbon, S. Lazebnik, P. Perona, Y. Sato, & C. Schmid (Eds.), *Computer Vision – ECCV 2012* (pp. 573–586). Springer.

[25] Rhodin, H., Richardt, C., Casas, D., Insafutdinov, E., Shafiei, M., Seidel, H.-P., Schiele, B., & Theobalt, C. (2016). EgoCap: Egocentric marker-less motion capture with two fisheye cameras. *ACM Transactions on Graphics*, *35*(6), 162:1–162:11. https://doi.org/10.1145/2980179.2980235

[26] Rogez, G., Supancic, J. S., & Ramanan, D. (2015). First-person pose recognition using egocentric workspaces. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 4325–4333. https://doi.org/10.1109/CVPR.2015.7299061

[27] Roygaga, C., Patil, D., Boyle, M., Pickard, W., Reiser, R., Bharati, A., & Blanchard, N. (2022). Ape-v: Athlete performance evaluation using video. *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 691–700.

[28] Sanzari, M., Ntouskos, V., & Pirri, F. (2016). Bayesian Image Based 3D Pose Estimation. In B. Leibe, J. Matas, N. Sebe, & M. Welling (Eds.), *Computer Vision – ECCV 2016* (pp. 566–582). Springer International Publishing.

[29] Sin, J., & Munteanu, C. (2020). Let's Go There: Combining Voice and Pointing in VR. *Proceedings of the 2nd Conference on Conversational User Interfaces*, 1–3.

[30] Tekin, B., Katircioglu, I., Salzmann, M., Lepetit, V., & Fua, P. (2016). Structured Prediction of 3D Human Pose with Deep Neural Networks. *Procedings of the British Machine Vision Conference 2016*, 130.1–130.11. https://doi.org/10.5244/C.30.130

[31] Tome, D., Alldieck, T., Peluse, P., Pons-Moll, G., Agapito, L., Badino, H., & De la Torre, F. (2020). SelfPose: 3D Egocentric Pose Estimation from a Headset Mounted Camera [Conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1–1. https://doi.org/10.1109/TPAMI.2020.3029700

[32] Tome, D., Peluse, P., Agapito, L., & Badino, H. (2019). xR-EgoPose: Egocentric 3D Human Pose From an HMD Camera. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 7727–7737. https://doi.org/10.1109/ICCV.2019.00782

[33] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, *30*.

[34] Wang, J., Liu, L., Xu, W., Sarkar, K., & Theobalt, C. (2021). Estimating egocentric 3d human pose in global space. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 11500–11509.

[35] Williams, A. S., & Ortega, F. R. (2020). Understanding gesture and speech multimodal interactions for manipulation tasks in augmented reality using unconstrained elicitation [Publisher: ACM New York, NY, USA]. *Proceedings of the ACM on Human-Computer Interaction*, *4*(ISS), 1–21.

[36] Williams, T., Bussing, M., Cabrol, S., Boyle, E., & Tran, N. (2019). Mixed reality deictic gesture for multi-modal robot communication. *2019 14th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, 191–201.

[37] Xu, W., Chatterjee, A., Zollhöfer, M., Rhodin, H., Fua, P., Seidel, H.-P., & Theobalt, C. (2019). Mo2Cap2: Real-time Mobile 3D Motion Capture with a Cap-mounted Fisheye Camera [Conference Name: IEEE Transactions on Visualization and Computer Graphics]. *IEEE Transactions on Visualization and Computer Graphics*, *25*(5), 2093–2101. https://doi.org/10.1109/TVCG.2019.2898650

[38] Yang, L. I., Huang, J., Feng, T., Hong-An, W., & Guo-Zhong, D. A. I. (2019). Gesture interaction in virtual reality [Publisher: Elsevier]. *Virtual Reality & Intelligent Hardware*, *1*(1), 84–112.

[39] Yuan, Y., & Kitani, K. (2019). Ego-Pose Estimation and Forecasting As Real-Time PD Control. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 10081–10091. https://doi.org/10.1109/ICCV.2019.01018

[40] Zhang, Y., You, S., & Gevers, T. (2021). Automatic Calibration of the Fisheye Camera for Egocentric 3D Human Pose Estimation from a Single Image. *2021 IEEE Winter Confer-*

ence on Applications of Computer Vision (WACV), 1771–1780. https://doi.org/10.1109/WACV48630.2021.00181

[41] Zhou, X., Zhu, M., Leonardos, S., & Daniilidis, K. (2017). Sparse Representation for 3D Shape Estimation: A Convex Relaxation Approach [Conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *39*(8), 1648–1661. https://doi.org/10.1109/TPAMI.2016.2605097

[42] Zhou, X., Zhu, M., Leonardos, S., Derpanis, K. G., & Daniilidis, K. (2016). Sparseness Meets Deepness: 3D Human Pose Estimation from Monocular Video [ISSN: 1063-6919]. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 4966–4975. https://doi.org/10.1109/CVPR.2016.537

[43] Zhou, X., Sun, X., Zhang, W., Liang, S., & Wei, Y. (2016). Deep Kinematic Pose Regression. In G. Hua & H. Jégou (Eds.), *Computer Vision – ECCV 2016 Workshops* (pp. 186–201). Springer International Publishing.