Automatic Calibration of the Fisheye Camera for Egocentric 3D Human Pose Estimation from a Single Image

Yahui Zhang, Shaodi You, Theo Gevers
University of Amsterdam
{y.zhang5, s.you, th.gevers}@uva.nl

Abstract

We propose a method for egocentric 3D human pose estimation from a single image captured by a fisheye camera. The problem of estimating the egocentric 3D pose for a fisheye camera is that images may be subject to strong image distortions (e.g. 2D poses on the image plane that pass through the line of sight of the fisheye lens).

Therefore, in this paper, we approach this problem by an automatic calibration module. Given a single image, our network first estimates 3D joint locations of a human in camera coordinates. To alleviate the impact of image distortions on 3D human pose estimation, we then use the automatic calibration to further regularize the 3D predictions. Experimental results demonstrate that the proposed method achieves state-of-the-art performance.

1. Introduction

Egocentric fisheye camera is used for human pose estimation or action recognition in different computer vision applications such as virtual reality (VR) or augmented reality (AR). These applications generally use a head mounted display to transform the user in a virtual world from a first-person viewpoint. Due to the large field of view, pose estimation from the egocentric fisheye viewpoint has many other valuable applications, such as robotics.

Current approaches focus on human pose estimation using pin-hole cameras. These methods show significant progress for different benchmarks, such as the Human3.6M [6] and MPI-INF-3DHP [16] datasets. To reduce the ambiguity, many methods estimate the root-relative 3D joint positions in camera coordinates. However, the problem of estimating the egocentric 3D pose for a fisheye camera is to predict the 3D human pose from a first-person viewpoint possibly subject to strong image distortions. These distortions may negatively influence the 3D poses when the 2D poses on the image plane pass through the line of sight of the fisheye lens. For example, as shown in Figure 1, two different 2D poses which are subject to different levels of image distortions correspond to the same 3D pose. Recent works [35, 28] propose methods for 3D human pose estimation from images captured by a fisheye camera to alleviate the problem of self-occlusion. However, their methods ignore the negative influence of the distortions on the 3D pose estimation.

To mitigate the effect of distortions on the 3D human pose estimation, we propose an automatic calibration module with self-correction to regularize 3D predictions. The proposed calibration module automatically estimates the intrinsic and distortion camera parameters with self-correction instead of using a post-processing step [35] to enforce the 3D predictions to be consistent with the corresponding distorted 2D poses. In this way, the effect of distortions on 3D pose estimation is alleviated. To assess the effectiveness of the proposed automatic calibration module, we modified the *x*R-EgoPose dataset [28], a recent public dataset for 3D human pose estimation collected by a fisheye camera, by adding different levels of image distortions. We show that our method outperforms previous state-of-the-art methods and significantly improves the performance by using the proposed automatic calibration.

The contributions of our approach are summarized as follows:

- We propose a method for egocentric 3D human pose estimation from a single image captured by a fisheye camera.
- We introduce an automatic calibration module with self-correction to mitigate the effect of image distortions for robust 3D human pose estimation.
- Our network shows state-of-the-art performance on the modified *x*R-EgoPose dataset containing images with different levels of distortions.

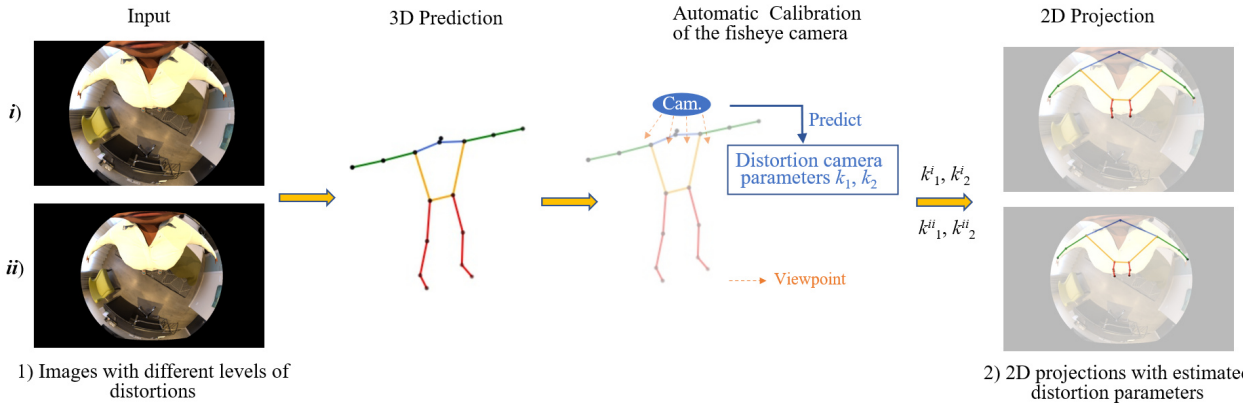


Figure 1. 3D pose prediction from a single image captured by a fisheye camera and 2D projection generated by our method. Note that although image (i) and (ii) appear different, they correspond to the same 3D pose. The proposed automatic calibration module alleviates the negative impact of image distortions on the 3D human pose estimation.

2. Related Work

In this section, we describe monocular 3D human pose estimation methods from a third-person viewpoint, a first-person viewpoint and wearable motion sensors.

Third-person 3D Pose Estimation Monocular 3D human pose estimation using external cameras show significant progress with the use of CNN's and with the availability of large-scale 2D [1, 8, 14] and 3D [6, 16] human pose datasets. In general, existing methods are categorised into two types: (i) Direct 3D human pose estimation from images with full supervision [13, 21, 27, 26, 39] and (ii) 3D pose estimation from intermediate 2D pose predictions [2, 4, 12, 15, 19, 34, 33, 20, 19, 37]. As direct pose estimation methods rely on extensive training data with 3D annotations, their generalization capability is limited. To mitigate the above problem, approaches attempt to create synthetic datasets based on Motion Capture (MoCap) systems [3, 30]. Nonetheless, differences still exist between synthetic images and real images, such as backgrounds, appearance and variety of details. On the other hand, using robust 2D pose detectors, 3D pose estimation methods decouple the task into 2D pose prediction and 3D pose lifting step. To reduce the requirement of 3D pose annotations, [38, 25] propose geometric constraints to regularize the 3D estimations. The human pose dataset with only 2D annotations is used to constrain the 3D predictions.

First-person 3D Pose Estimation A number of methods based on egocentric cameras focuses on hands, arms or torso detection [23, 36]. However, it is considerably more challenging to estimate the full 3D human pose from egocentric cameras. Jiang *et al*. [7] propose a method for 3D pose estimation based on videos taken from chest-mounted cameras by considering the motion of the surrounding scene. However, the predictions are less accurate and have low confidence. Rhodin *et al*. [22] present an approach for full human body reconstruction captured from a head-mounted camera pair. Only recently, egocentric monocular 3D human pose estimation based on fisheye cameras is proposed. Xu *et al*. [35] design a new head-mounted system, where a fisheye camera is placed at the rig of a standard baseball cap. To reduce the error of the lower body, their methods take two images — one original image and one 2 $\times$ zoomed central part of the original image, as input to compute the 3D pose estimation. Tome *et al*. [28] propose an auto-encoder with two branches for egocentric 3D human pose estimation based on a fisheye camera. However, their methods assume images with the same distortion and therefore ignoring the negative impact of different levels of distortions on 3D human pose estimation.

3D Pose Estimation from Wearable Motion Sensors Inertial Measurement Units (IMUs) are used to perform 3D pose estimation from a first-person viewpoint. However, a large number of sensors may cause the system to become intrusive and require more time to calibrate. Using less sensors becomes more challenging to reconstruct the 3D human pose in this configuration [31]. Shiratori *et al*. [24] introduce an alternative way to estimate the 3D human pose by structure-from-motion (SfM), with 16 cameras mounted at the human body joints. Nonetheless, this approach is difficult to use in real scenes due to motion blur, self-occlusion of limbs and missing textures in the background.

3. Automatic Calibration of Fisheye Cameras

The inherent problem of image distortions captured by fisheye cameras makes 3D human pose estimation challenging. Two images may correspond to the same 3D pose when 2D different poses on the image plane pass through the line of sight of the fisheye lens. Therefore, it is difficult to regress 3D human joint positions in camera coordinates without the distortion parameters. To alleviate this problem, we propose an automatic calibration module to enforce the

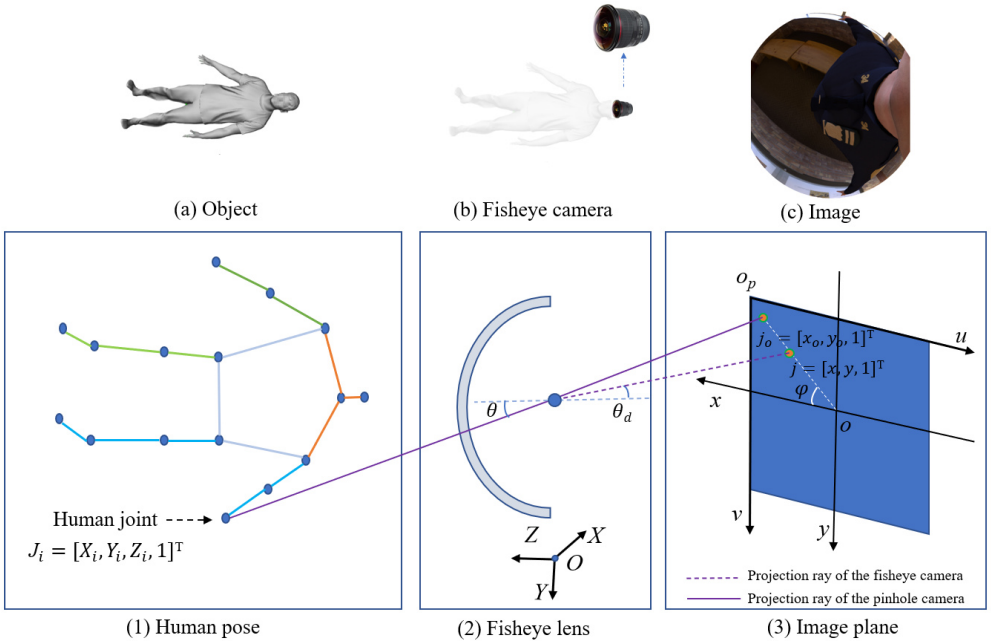


Figure 2. The imagery model of 3D-to-2D projection. The object — human joints $\boldsymbol{J}_i$ are located in camera coordinates $OXYZ$; The projected 2D pose (hand joint as an example) with the pinhole camera $\boldsymbol{j}_o$ and the fisheye camera $\boldsymbol{j}$ is on the image plane oxy; o_puv representing pixel coordinates. θ and θ_d indicate the angle of incidence and refraction with the fisheye lens respectively. φ represents the angle between the projected ray $\vec{oj}$ and x axis on the image plane.

3D predictions to be consistent with the corresponding distorted 2D poses.

3.1. Fisheye Camera Model

As shown in Figure 2.1, the human pose is represented by a set of joints $\boldsymbol{J}_i = [X_i, Y_i, Z_i, 1]^T$ located in the camera coordinate system. For the fisheye lens, shown in Figure 2.2, the angle of refraction from 3D locations $\boldsymbol{J}$ in Figure 2.1 is decreased from θ to θ_d. Then, the joint location $\boldsymbol{J}$ is projected on the image plane by $\boldsymbol{j} = [x, y, 1]^T$ in Figure 2.3. Particularly, the projected joint $\boldsymbol{j}_o = [x_o, y_o, 1]^T$ represents the projection based on the pin-hole camera model. It is because of the distortion that positions $\boldsymbol{j}$ and $\boldsymbol{j}_o$ are different.

3.2. Egocentric 3D Pose Estimation under a Fisheye Camera

From a single 2D image to 3D pose. 3D human pose estimation from a single image is an ill-posed geometric problem: there is no depth information. Previous methods attempt to solve this problem by learning the relation between 2D and 3D poses in a data-driven manner. However, with strong image distortions introduced by a fisheye camera, 3D human pose estimation is more challenging.

To alleviate the above issues, we propose an automatic calibration module to regularize 3D predictions. Instead of using a post-processing method [35] or ground truth, the proposed module automatically predicts the distortion camera parameters with self-correction. This is the first attempt to perform egocentric 3D human pose estimation by using automatic calibration of the fisheye camera.

From 3D pose to 2D pose. For a fisheye camera mounted on the head, the relative depth of human joints is comparable to the distance between the camera and the human joints. Therefore, weak perspective projection can not be used to approximate the 2D projections [9, 5, 32]. The 3D-to-2D projection process for the fisheye camera is illustrated in Figure 2.

Let $\mathbf{P}_{3D} = [\boldsymbol{J}_1, \boldsymbol{J}_2, ..., \boldsymbol{J}_n]$ denotes the human joint locations in camera coordinate $OXYZ$, where n is the number of human joints and $\boldsymbol{J}_i = [X_i, Y_i, Z_i, 1]^T$. The projected 2D pose from the fisheye camera and pinhole camera is defined by $\mathbf{p}_{2D}$ and $\mathbf{p}_{o2D}$, a 3 by n matrix with $\boldsymbol{j}_i = [x_i, y_i, 1]^T$ and $\boldsymbol{j}_{oi} = [x_{oi}, y_{oi}, 1]^T$ respectively.

Given the intrinsic ($\boldsymbol{K}$) and extrinsic ($\boldsymbol{R}$ and $\boldsymbol{T}$) camera parameters, the 2D projections $\mathbf{p}_{o2D}$ under the pin-hole camera model is as follows:

$$s \cdot \mathbf{p}_{o2D} = \boldsymbol{K}[\boldsymbol{R}|\boldsymbol{T}]\mathbf{P}_{3D}. \quad (1)$$

where the extrinsic camera parameters $\boldsymbol{R}$ and $\boldsymbol{T}$ are the identity matrix, s represents the scale factor and is equal to the Z value of the corresponding 3D joints in camera coordinates.

As the fisheye lens produces strong image distortions

compared to a pinhole camera, distortion matrix $\boldsymbol{D}$ needs to be considered to compute 2D projections from a fisheye camera:

$$s \cdot \mathbf{p}_{2\mathrm{D}} = \boldsymbol{K}\boldsymbol{D}[\boldsymbol{R}|\boldsymbol{T}]\mathbf{P}_{3\mathrm{D}}. \tag{2}$$

In this paper, $\boldsymbol{D}$ is defined by

$$\boldsymbol{D} = \begin{bmatrix} \theta_d/l & 0 & 0 \\ 0 & \theta_d/l & 0 \\ 0 & 0 & 1 \end{bmatrix}, \tag{3}$$

where $l = \frac{\sqrt{X^2+Y^2}}{Z}$, and θ_d indicates the angle of refraction. In this paper, we refer to [29, 10] to calculate the angle of refraction $\theta_d = \theta(1+k_1\theta^2+k_2\theta^4)$, where the angle of incidence $\theta = \arctan(l)$, and the number of radial distortion parameters to be estimated is set to two, *i.e.*, k_1, k_2.

Visually, the 2D projection $\boldsymbol{j}_o$ under the constraint of a pinhole camera is transformed to $\boldsymbol{j}$ for a fisheye camera in Figure 2.3 using Equation 2 for the distortion camera matrix.

3.3. Error Analysis of Estimated 3D Joints and 2D Projections

Different from other methods for 3D pose estimation from external viewpoints, *i.e.,* outside-in approaches, in our task, the depth variance of human joints is comparable to the distance between the human joints and the fisheye camera. Therefore, the depth of 3D joint locations has an effect on the 2D re-projection error. Besides, the level of distortions and the distance of 3D joint locations to the optical axis (*Z* axis) also influence the 2D re-projection error.

Equation 2 can be detailed as follows,

$$x = f\frac{\theta_d}{l}\frac{X}{Z} = f(\theta + k_1\theta^3 + k_2\theta^5)\frac{X}{\sqrt{X^2+Y^2}}, \tag{4}$$

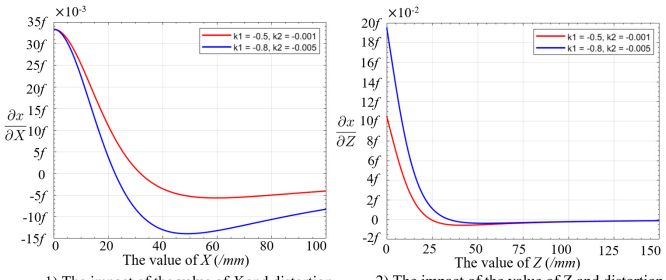
$$y = f\frac{\theta_d}{l}\frac{X}{Z} = f(\theta + k_1\theta^3 + k_2\theta^5)\frac{Y}{\sqrt{X^2+Y^2}}. \tag{5}$$

Without loss of generality, we only study the influence of the level of distortions, the depth *Z*, and the value of *X* (the same to *Y*) on the 2D re-projection error. Before calculating the derivation of Equation 4, we set Y to zero to simplify the formula, *i.e.*,

$$x = f\frac{\theta_d}{l}\frac{X}{Z} = f(\arctan\frac{X}{Z} + k_1\arctan^3\frac{X}{Z} + k_2\arctan^5\frac{X}{Z}). \tag{6}$$

The partial derivative of Equation 6 is taken:

$$\begin{aligned} \frac{\partial x}{\partial X} &= f\frac{Z}{X^2+Z^2}(1 + 3k_1\arctan^2\frac{X}{Z} + 5k_2\arctan^4\frac{X}{Z}), \\ \frac{\partial x}{\partial Z} &= -f\frac{X}{X^2+Z^2}(1 + 3k_1\arctan^2\frac{X}{Z} + 5k_2\arctan^4\frac{X}{Z}). \end{aligned} \tag{7}$$



1) The impact of the value of *X* and distortion parameters on reprojection error.

2) The impact of the value of Z and distortion parameters on reprojection error.

Figure 3. The impact of value of *X* and *Z* on the re-projection error under the fisheye camera with different distortion parameters k_1 and k_2. Due to the large range of hand and elbow joints, we plot the curve setting *Z* to be 30*mm* as shown in Figure 3.1. Since most joints such as shoulders, hips and knees have similar positions in the *XY* plane, we plot this curve setting *X* = 25*mm* as shown in Figure 3.2.

Figure 3 shows the impact of distortion parameters, the value of *X* and *Z* on re-projection error according to Equation 7: 1) The value of *X* and *Z* have different influences on the re-projection error with various distortion parameters. Specifically, the larger the image distortions, the larger the influence of 3D locations on the 2D re-projection error. 2) Under the same level of distortions, the 3D joint locations with larger distances to the camera (such as ankles, toes and hips joints in lower body) or with larger *X* (such as elbows and hands joints) are expected to cause smaller errors on the 2D projections. In other words, the error of 3D poses is larger for joints with larger distances to the camera in the depth or larger distances to the optical axis under the same error of 2D projections.

3.4. Self-correction for Calibrating the Fisheye Camera

An automatic calibration module is proposed to regularize the 3D predictions. Our calibration module predicts the intrinsic camera parameters $\boldsymbol{K}$ and the distortion camera parameters $\boldsymbol{D}$ automatically. Specifically, $\boldsymbol{K}$ includes focal length (f) and principal coordinates (u_0, v_0) while $\boldsymbol{D}$ contains the distortion parameters (k_1, k_2).

As discussed in Section 3.2, the re-projection error depends on the level of distortions, the depth, and the distance to the optical axis of the estimated 3D joints. Therefore, the commonly used $L2$ loss that constrains the camera parameters in the outside-in approaches [5, 32] cannot be used to update our automatic calibration module. The optimization process will focus on the upper body estimation, especially for neck and arm joints, due to the larger re-projection error. This may result in inaccurate estimation of hands, elbows and joints in lower body. We will verify this issue in Section 5.

To optimize our automatic calibration module, we mini-

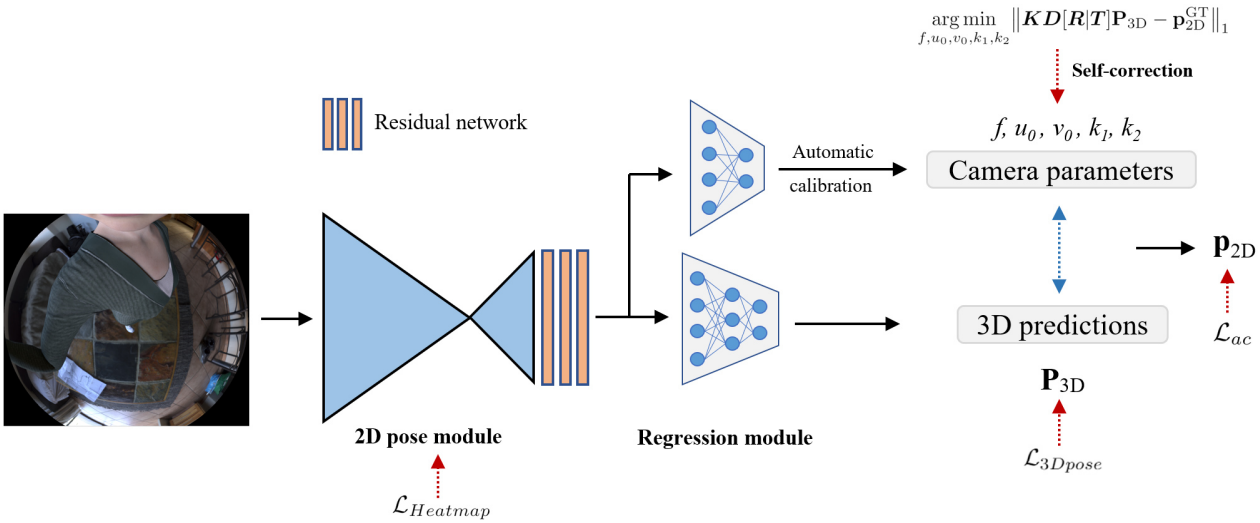


Figure 4. Overview of the proposed framework. We use *ResNet-50* as our backbone to detect 2D poses with heatmap representations. The 2D heatmaps are fed into a residual network with attention mechanism to further exploit the information in latent space. Then, we employ a series of fully connected layers to estimate the 3D pose in camera coordinates and camera parameters (*i.e.*, focal length f, principal coordinate u_0 and v_0 and distortion parameters k_1 and k_2). Finally, the estimated 3D pose are enforced to be consistent with 2D poses on image plane using the predicted camera parameters.

mize the absolute error between the projected 3D pose and 2D pose annotations $\mathbf{p}_{2D}^{GT}$. This avoids the optimization process to focus on the joints with larger re-projection errors:

$$\arg\min_{f,u_0,v_0,k_1,k_2} \left\|\boldsymbol{K}\boldsymbol{D}[\boldsymbol{R}|\boldsymbol{T}]\mathbf{P}_{3D} - \mathbf{p}_{2D}^{GT}\right\|_1. \quad (8)$$

Note that the camera parameters (f, u_0, v_0, k_1, k_2) and $\mathbf{P}_{3D}$ are updated simultaneously.

4. Network and Training Details

Given a single image captured by a fisheye camera, our method aims to regress 3D human joint locations in camera coordinates. In this section, we introduce our network design and training strategy of our network.

4.1. Network Design

Our framework consists of three modules as shown in Figure 4. First, we employ a 2D pose module to detect 2D heatmaps of human joint positions on the image plane. Second, a 3D pose regression module takes the fused features from 2D heatmaps as input to estimate 3D joint locations in camera coordinates. Finally, we use the proposed automatic calibration of the fisheye camera to enforce 3D predictions to be consistent with the corresponding 2D poses under the distortions.

2D Pose module. Considering the accuracy and computational costs, we adopt *ResNet-50* followed by three deconventional layers as our 2D pose module. Given a single image with a resolution of 256×256, 2D pose module infers 2D poses with heatmap representations $\mathbf{HM} \in \mathbb{R}^{16\times64\times64}$, where 16 indicates the number of human body joints with the space dimension of 64×64.

To train the 2D pose detector, we use the mean square error (MSE) to calculate the loss between the estimated **HM** and 2D ground-truth heatmaps $\mathbf{HM}^{GT}$. The loss function is defined by:

$$\mathcal{L}_{Heatmap} = \sum_{h}^{H}\sum_{w}^{W} \left\|\mathbf{HM}_{(h,w)} - \mathbf{HM}_{(h,w)}^{GT}\right\|_2, \quad (9)$$

where H and W indicate the resolution of the heatmaps. Specifically, we generate ground-truth heatmaps by using Gaussian distributions with kernel size of 13×13 and standard deviation of 2 pixels on each joint locations on the image plane.

2D-to-3D Regression Module. To regress the 3D human pose $\mathbf{P}_{3D}$ in camera coordinates, we employ several residual blocks with fully connected layers followed by batch normalization, ReLU non-linearity and Dropout. Considering the inference time and prediction accuracy, we use two residual blocks for 2D-to-3D regression.

We optimize the 3D pose regression module by minimizing the MSE error between 3D predictions $\mathbf{P}_{3D}$ and 3D pose ground truth $\mathbf{P}_{3D}^{GT}$. Given the dataset with the number of N samples, the loss function is defined by:

$$\mathcal{L}_{3D} = \frac{1}{N}\sum_{i=1}^{N} \left\|\mathbf{P}_{3D(i)} - \mathbf{P}_{3D\,(i)}^{GT}\right\|_2, \quad (10)$$

where i represents the index of the training set.

Table 1. Comparison with existing methods on the modified *x*R-EgoPose dataset.

Approach	Gaming	Gesticulating	Greeting	Lower Stretching	Patting	Reacting	Talking	Upper Stretching	Walking	Average
Martinez [15]	98.3	85.3	65.6	83.0	74.7	97.2	53.7	77.2	79.2	79.7
Ours (w/o $\mathcal{L}_{ac}$)	80.7	66.4	61.0	74.8	65.6	80.2	44.4	83.8	76.4	78.6
Ours	**75.3**	**66.0**	**54.1**	**68.7**	**65.4**	**78.3**	**43.0**	**67.4**	**69.2**	**67.7**

[1] Ours (w/o $\mathcal{L}_{ac}$) indicates that our method is trained without using the proposed automatic calibration module.

Automatic Calibration Module. As shown in Figure 4, there are two branches in the regression module. The first branch is the lifting module regressing 3D locations of human joints while a multi-layer perception is employed in the second branch to perform automatic calibration of the fisheye camera. Specifically, the second branch estimates the intrinsic camera parameters consisting of focal length (f), principal coordinate (c_x, c_y) and distortion parameters (k_1, k_2). Then we use Equation 2 to obtain the 2D projections, where 3D predictions are constrained by the 2D poses under the distortions. In this way, the impact of image distortions on 3D human pose estimation is alleviated. In this paper, automatic calibration module is only applied during the training phase.

As discussed in Section 3.2, the level of distortions, the depth and distance to the optical axis of estimated 3D joint locations have an influence on the errors of the corresponding 2D projections. Therefore, we minimize the absolute error (*i.e.*, $L1$ loss) between the projected 3D pose and 2D ground truth avoiding the optimization to focus on joints with large re-projection errors. The loss function is defined by:

$$\mathcal{L}_{ac} = \frac{1}{N}\sum_{i=1}^{N}\left\|\boldsymbol{K}\boldsymbol{D}[\boldsymbol{R}|\boldsymbol{T}]\mathbf{P}_{3\mathrm{D}(i)} - \mathbf{p}_{2\mathrm{D}\ (i)}^{\mathrm{GT}}\right\|_1, \qquad (11)$$

where $\boldsymbol{R}$ and $\boldsymbol{T}$ are the identity matrix.

4.2. Training

According to Equation 9 - 11, we train our full network by minimizing the following cost function:

$$\mathcal{L}_{pose} = \lambda_{HM}\mathcal{L}_{Heatmap} + \mathcal{L}_{3D} + \lambda_{ac}\mathcal{L}_{ac}, \qquad (12)$$

where λ_{HM} and λ_{rep} are loss weights to adjust the combination of the 2D heatmap loss, the 3D pose loss and the loss of automatic calibration module.

During training, we first pre-train the 2D pose module on the external perspective MPII dataset, because we found pre-trained 2D pose module obtains higher accuracy of 2D pose estimations for images captured by a fisheye camera. Then, we fine-tune the whole network on the modified *x*R-EgoPose dataset.

5. Experiments

Datasets. Recently, two datasets for egocentric 3D human pose estimation for a fisheye camera are released — *x*R-EgoPose [28] and Mo2Cap2 [35] datasets. Both datasets consist of a large number of frames of daily activities for different environments and lighting conditions. Considering images from current datasets with the same distortion, we modified the *x*R-EgoPose dataset using Equation 4 and Equation 5 to randomly add image distortions. For fast evaluation, the total number of images in the modified dataset is one-fifth of the total size in original *x*R-EgoPose dataset.

Evaluation Metrics. We use the Mean Per Joint Position Error (MPJPE) as the evaluation metric in the experiments. Note that we do not need to align the root joint for the evaluation as in the outside-in approaches.

Implement Details. The proposed network regresses 16 human body joints including the head joint. The head joint is estimated based on the position of head-mounted display from 2D images. We first pre-train our 2D pose module on the MPII dataset [1] and then train our full network for 36 epochs on the modified *x*R-EgoPose dataset using Adam [11] for optimization. The learning rate is set to 5×10^{-4}. The model is trained on two GTX 1080ti GPUs with a batch size of 64. The weights in the overall loss function are set to $\mathcal{L}_{hm} = 10^7$ and $\mathcal{L}_{ac} = 50$.

Method Comparisons. To assess the effectiveness of our method, we conduct experiments on the modified *x*R-EgoPose dataset compared with Martinez *et al*. [15], a simple but effective 3D pose estimation method from external camera viewpoints. Furthermore, we evaluate our method on the *x*R-EgoPose and Mo2Cap2 datasets compared with current state-of-the-art methods [35, 28] for egocentric 3D human pose estimation for fisheye cameras.

5.1. Evaluation on Modified *x*R-EgoPose Dataset

Overall performance. We first evaluate the proposed approach on the modified *x*R-EgoPose dataset. Since the existing methods [35, 28] do not release their codes, it is hard to make a fair comparison with them. Therefore, we compared our method with a state-of-the-art method [15] for 3D human pose estimation from external camera viewpoints.

Table 2. Average error for per joint performed by our method with $L1$ and $L2$ loss on the modified *x*R-EgoPose dataset.

Joint	Error ($/mm$)			Joint	Error ($/mm$)		
	Ours	Ours_$L2$	Improvement		Ours	Ours_$L2$	Improvement
Head	32.90	27.03	-5.87	Neck	20.53	17.20	-3.33
Left Arm	31.91	35.61	**3.69**	Right Arm	33.23	36.97	**3.74**
Left Elbow	47.69	51.82	**4.13**	Right Elbow	52.22	58.24	**6.02**
Left Hand	82.99	93.83	**10.84**	Right Hand	85.49	100.19	**14.71**
Left Hip	56.86	65.03	**8.17**	Right Hip	56.77	65.13	**8.35**
Left Knee	79.33	87.54	**8.21**	Right Knee	79.87	89.84	**9.97**
Left Foot	100.31	110.84	**10.53**	Right Foot	103.21	115.41	**12.20**
Left Toe	109.39	119.61	**10.22**	Right Toe	110.73	122.09	**11.36**

[1] Ours_$L2$ denotes our method use $L2$ loss to update the proposed automatic calibration module.

Table 3. Experimental results of our network on the modified *x*R-EgoPose dataset under less 3D ground truth.

Approach	3D gorund truth	MPJPE($/mm$)
Martinez *et al*. [15]	100%	79.7
Ours	100%	67.7
Ours	**80%**	76.9

Table 1 lists the experimental results showing that our method achieves the best performance in all activities, leading to an improvement of 15.1% in overall performance.

Effectiveness of automatic calibration module. We perform an ablation study on the modified *x*R-EgoPose dataset to assess the influence of our proposed automatic calibration module. The MPJPE of all activities are reported in Table 1, in which Ours (w/o $\mathcal{L}_{ac}$) refers to the proposed method without automatic calibration module. Our method obtained better performance than Ours (w/o $\mathcal{L}_{ac}$) with a $10.9mm$ improvement. The results show the effectiveness of the proposed automatic calibration module.

Update strategy of automatic calibration module. As discussed in Section 3.2, the level of distortions, the depth, and the distance to the optical axis of 3D joint locations have an influence on the error of the 2D projections. Based on the error analysis, we employ the $L1$ loss to train our automatic calibration module instead of the commonly used in the outside-in approach — $L2$ loss. In this way, our update strategy avoids the optimization process to focus on the estimated 3D joints with larger 2D re-projection errors. Otherwise, an inappropriate update strategy of our automatic calibration module may lead to overfitting of these joints and a decrease in overall performance.

We conduct a comparative experiment on the modified *x*R-EgoPose dataset to validate this strategy. Particularly, we denote our method using $L2$ loss as Ours_$L2$. Table 2 reports the average error for each estimated joint and the improvement by our method. It is shown that the proposed method achieves better performance for each joint except head and neck joints. Note that the error of joints in 1) lower body, such as knee, foot and toes, and 2) joints with large distances to the optical axis, such as hands and elbows in the 3D space are reduced significantly by our method, which validates our assumption.

5.2. Mixed 2D and 3D Ground Truth Datasets

Another advantage of the proposed method is that our network can be trained on a mixture of 2D and 3D pose datasets. Due to our automatic calibration module, the estimated 3D pose can be partially constrained by the 2D ground truth, alleviating the needs of 3D ground-truth labels. We test our model on the modified *x*R-EgoPose dataset with 80% of 3D annotations while the 2D ground truth labels are available in the training phase. Table 3 lists the experimental results. Our method still outperforms Martinze *et al*. [15] ($79.7mm$) with an error of $76.9mm$.

5.3. Evaluation on current datasets

Evaluations on *x*R-EgoPose dataset. We also validate our method on the original *x*R-EgoPose dataset. Specifically, the proposed method is compared with Martinez *et al*. [15] and Tome *et al*. [28] — the state-of-the-art egocentric pose estimation method. Table 4 shows MPJPE on this dataset, including the error on each activity and the average error. Our method shows the best performance with an average error of $50.0mm$, leading to an improvement of 14.1% on average compared to the state-of-the-art results.

Evaluations on Mo2Cap2 dataset. We further compare our method with current methods on the Mo2Cap2 dataset. Table 5 shows the experimental results, where 3DV'17 [17] and VNect [18] focus on pose estimation from external camera viewpoint while Xu *et al*. [35] and Tome *et al*. [28] are the current state-of-the-art egocentric pose estimation methods. Note that Xu *et al*. (i) take two images — one original image and one $2\times$ zoomed central part of the original image to regress 3D poses while we only use a single

Table 4. Comparison with existing methods on *x*R-EgoPose dataset.

Approach	Gaming	Gesticulating	Greeting	Lower Stretching	Patting	Reacting	Talking	Upper Stretching	Walking	Average
Martinez [15]	109.6	105.4	119.3	125.8	93.0	119.7	111.1	124.5	130.5	122.1
Tome [28]	56.0	50.2	44.6	51.1	59.4	60.8	43.9	**53.9**	57.7	58.2
Ours	**36.8**	**34.1**	**36.7**	**50.1**	**57.2**	**34.4**	**32.8**	54.3	**52.6**	**50.0**

Table 5. Comparison with existing methods on indoor set of Mo2Cap2 dataset.

Approach	Walking	Sitting	Crawling	Crouching	Boxing	Dancing	Stretching	Waving	Average
3DV'17 [17]	48.76	101.22	118.96	94.93	57.34	60.96	111.36	**64.50**	76.28
VNect [18]	65.28	129.59	133.08	120.39	78.43	82.46	153.17	83.91	97.85
Xu* [35]	38.41	70.94	94.31	81.90	48.55	55.19	99.34	60.92	61.40
Tome* [28]	38.39	61.59	69.53	51.14	37.67	42.10	58.32	44.77	48.16
Ours	**41.16**	**76.58**	**73.04**	**89.67**	**52.96**	**58.90**	**92.21**	71.55	**62.13**

[1] $*$ means the method uses extra information.

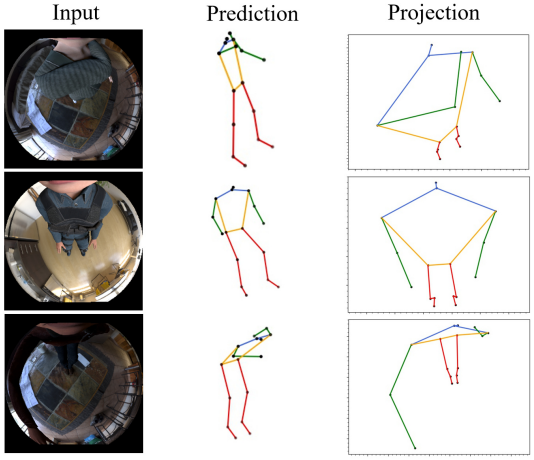


Figure 5. The visual results on the modified *x*R-EgoPose dataset predicted by the proposed method.

image as input; (ii) need the toolbox for calibration of the fisheye camera to obtain distortion camera parameters while we directly estimate the distortion camera parameters with self-correction in our framework. On the other hand, Tome *et al*. uses the estimated 2D heatmaps from Xu *et al*. to implement the evaluation. From Table 5, the proposed method achieves competitive results with an error of 62.13mm on the indoor set of Mo2Cap2 dataset, even with only a single image as input.

6. Conclusions

We presented a novel method for egocentric 3D human pose estimation from a single image captured by a fisheye camera. To alleviate the impact of image distortions on 3D human pose estimation, we proposed an automatic calibration module to enforce the 3D predictions to be consistent with the corresponding 2D projections under the distortions. Experimental results showed that our method obtained state-of-the-art performance on the modified *x*R-EgoPose and current datasets compared with existing methods.

References

[1] Mykhaylo Andriluka, Leonid Pishchulin, Peter Gehler, and Bernt Schiele. 2d human pose estimation: New benchmark and state of the art analysis. In *CVPR*, pages 3686–3693, 2014.

[2] Ching-Hang Chen and Deva Ramanan. 3d human pose estimation = 2d pose estimation + matching. In *CVPR*, pages 7035–7043, 2017.

[3] Wenzheng Chen, Huan Wang, Yangyan Li, Hao Su, Zhenhua Wang, Changhe Tu, Dani Lischinski, Daniel Cohen-Or, and Baoquan Chen. Synthesizing training images for boosting human 3d pose estimation. In *3DV*, pages 479–488. IEEE, 2016.

[4] Hao-Shu Fang, Yuanlu Xu, Wenguan Wang, Xiaobai Liu, and Song-Chun Zhu. Learning pose grammar to encode human body configuration for 3d pose estimation. In *AAAI*, 2018.

[5] Ikhsanul Habibie, Weipeng Xu, Dushyant Mehta, Gerard Pons-Moll, and Christian Theobalt. In the wild human pose estimation using explicit 2d features and intermediate 3d representations. In *CVPR*, pages 10905–10914, 2019.

[6] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3. 6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *TPAMI*, 36(7):1325–1339, 2013.

[7] Hao Jiang and Kristen Grauman. Seeing invisible poses: Estimating 3d body pose from egocentric video. In *CVPR*, pages 3501–3509. IEEE, 2017.

[8] Sam Johnson and Mark Everingham. Clustered pose and nonlinear appearance models for human pose estimation. In *BMVC*, 2010.

[9] Angjoo Kanazawa, Michael J Black, David W Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *CVPR*, pages 7122–7131, 2018.

[10] Juho Kannala and Sami Brandt. A generic camera calibration method for fish-eye lenses. In *ICPR 2004.*, volume 1, pages 10–13. IEEE, 2004.

[11] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[12] Chen Li and Gim Hee Lee. Generating multiple hypotheses for 3d human pose estimation with mixture density network. In *CVPR*, pages 9887–9895, 2019.

[13] Sijin Li and Antoni B Chan. 3d human pose estimation from monocular images with deep convolutional neural network. In *ACCV*, pages 332–347. Springer, 2014.

[14] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, pages 740–755, 2014.

[15] Julieta Martinez, Rayat Hossain, Javier Romero, and James J Little. A simple yet effective baseline for 3d human pose estimation. In *ICCV*, pages 2640–2649, 2017.

[16] Dushyant Mehta, Helge Rhodin, Dan Casas, Pascal Fua, Oleksandr Sotnychenko, Weipeng Xu, and Christian Theobalt. Monocular 3d human pose estimation in the wild using improved cnn supervision. In *3DV*, pages 506–516. IEEE, 2017.

[17] Dushyant Mehta, Helge Rhodin, Dan Casas, Pascal Fua, Oleksandr Sotnychenko, Weipeng Xu, and Christian Theobalt. Monocular 3d human pose estimation in the wild using improved cnn supervision. In *3DV*, pages 506–516. IEEE, 2017.

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

[19] Francesc Moreno-Noguer. 3d human pose estimation from a single image via distance matrix regression. In *CVPR*, pages 2823–2832, 2017.

[20] Georgios Pavlakos, Xiaowei Zhou, and Kostas Daniilidis. Ordinal depth supervision for 3d human pose estimation. In *CVPR*, pages 7307–7316, 2018.

[21] Georgios Pavlakos, Xiaowei Zhou, Konstantinos G Derpanis, and Kostas Daniilidis. Coarse-to-fine volumetric prediction for single-image 3d human pose. In *CVPR*, pages 7025–7034, 2017.

[22] Helge Rhodin, Christian Richardt, Dan Casas, Eldar Insafutdinov, Mohammad Shafiei, Hans-Peter Seidel, Bernt Schiele, and Christian Theobalt. Egocap: egocentric marker-less motion capture with two fisheye cameras. *ACM Transactions on Graphics (TOG)*, 35(6):1–11, 2016.

[23] Grégory Rogez, James S Supancic, and Deva Ramanan. First-person pose recognition using egocentric workspaces. In *CVPR*, pages 4325–4333, 2015.

[24] Takaaki Shiratori, Hyun Soo Park, Leonid Sigal, Yaser Sheikh, and Jessica K Hodgins. Motion capture from body-mounted cameras. In *ACM SIGGRAPH 2011 papers*, pages 1–10. 2011.

[25] Xiao Sun, Jiaxiang Shang, Shuang Liang, and Yichen Wei. Compositional human pose regression. In *ICCV*, pages 2602–2611, 2017.

[26] Xiao Sun, Bin Xiao, Fangyin Wei, Shuang Liang, and Yichen Wei. Integral human pose regression. In *ECCV*, pages 529–545, 2018.

[27] Bugra Tekin, Isinsu Katircioglu, Mathieu Salzmann, Vincent Lepetit, and Pascal Fua. Structured prediction of 3d human pose with deep neural networks. *BMVC*, 2016.

[28] Denis Tome, Patrick Peluse, Lourdes Agapito, and Hernan Badino. xr-egopose: Egocentric 3d human pose from an hmd camera. In *ICCV*, pages 7728–7738, 2019.

[29] FA Van den Heuvel, R Verwaal, and B Beers. Automated calibration of fisheye camera systems and the reduction of chromatic aberration. *PHOTOGRAMMETRIE FERNERKUNDUNG GEOINFORMATION*, 2007(3):157, 2007.

[30] Gul Varol, Javier Romero, Xavier Martin, Naureen Mahmood, Michael J Black, Ivan Laptev, and Cordelia Schmid. Learning from synthetic humans. In *CVPR*, pages 109–117, 2017.

[31] Timo von Marcard, Bodo Rosenhahn, Michael J Black, and Gerard Pons-Moll. Sparse inertial poser: Automatic 3d human pose estimation from sparse imus. In *Computer Graphics Forum*, volume 36, pages 349–360. Wiley Online Library, 2017.

[32] Bastian Wandt and Bodo Rosenhahn. Repnet: Weakly supervised training of an adversarial reprojection network for 3d human pose estimation. In *CVPR*, pages 7782–7791, 2019.

[33] Min Wang, Xipeng Chen, Liu Liu, Chen Qian, Liang Lin, and Lizhuang Ma. Drpose3d: Depth ranking in 3d human pose estimation. In *Twenty-Seventh International Joint Conference on Artificial Intelligence*, pages 978–984, 2018.

[34] Jiajun Wu, Tianfan Xue, Joseph J Lim, Yuandong Tian, Joshua B Tenenbaum, Antonio Torralba, and William T Freeman. Single image 3d interpreter network. In *ECCV*, pages 365–382. Springer, 2016.

[35] Weipeng Xu, Avishek Chatterjee, Michael Zollhoefer, Helge Rhodin, Pascal Fua, Hans-Peter Seidel, and Christian Theobalt. Mo2cap2: Real-time mobile 3d motion capture with a cap-mounted fisheye camera. *IEEE transactions on visualization and computer graphics*, 25(5):2093–2101, 2019.

[36] Haruka Yonemoto, Kazuhiko Murasaki, Tatsuya Osawa, Kyoko Sudo, Jun Shimamura, and Yukinobu Taniguchi. Egocentric articulated pose tracking for action recognition. In *2015 14th IAPR International Conference on Machine Vision Applications (MVA)*, pages 98–101. IEEE, 2015.

[37] Long Zhao, Xi Peng, Yu Tian, Mubbasir Kapadia, and Dimitris N Metaxas. Semantic graph convolutional networks for 3d human pose regression. In *CVPR*, pages 3425–3435, 2019.

[38] Xingyi Zhou, Qixing Huang, Xiao Sun, Xiangyang Xue, and Yichen Wei. Towards 3d human pose estimation in the wild: a weakly-supervised approach. In *ICCV*, pages 398–407, 2017.

[39] Xiaowei Zhou, Spyridon Leonardos, Xiaoyan Hu, and Kostas Daniilidis. 3d shape estimation from 2d landmarks: A convex relaxation approach. In *CVPR*, pages 4447–4455, 2015.