Camera Distance-aware Top-down Approach for 3D Multi-person Pose Estimation from a Single RGB Image

Gyeongsik Moon
Department of ECE, ASRI
Seoul National University
mks0601@snu.ac.kr

Ju Yong Chang
Department of EI
Kwangwoon University
juyong.chang@gmail.com

Kyoung Mu Lee
Department of ECE, ASRI
Seoul National University
kyoungmu@snu.ac.kr

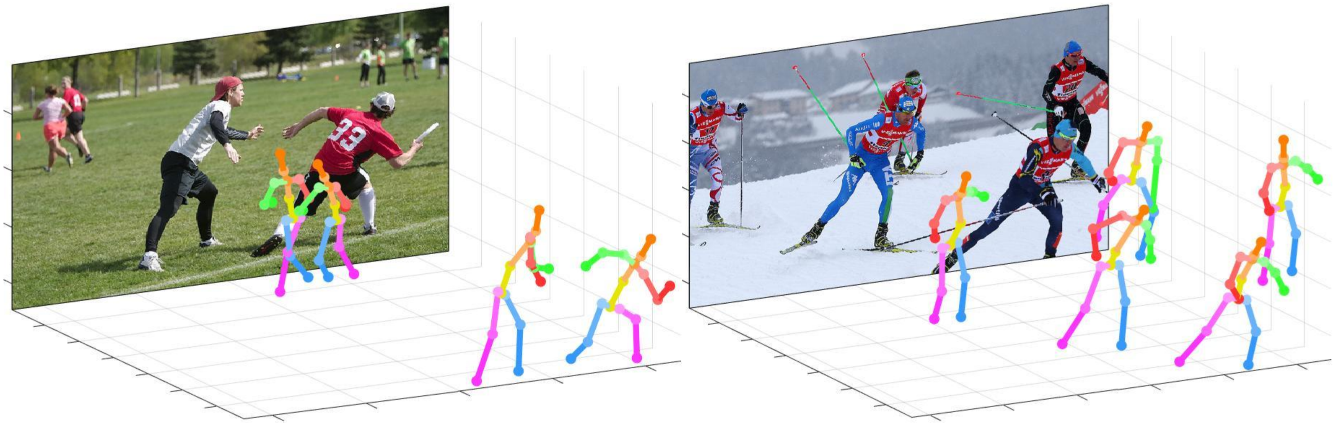

Figure 1: Qualitative results of applying our 3D multi-person pose estimation framework to COCO dataset [25] which consists of *in-the-wild* images. Most of the previous 3D human pose estimation studies mainly focused on the root-relative 3D single-person pose estimation. In this study, we propose a general framework which considers all required components for 3D multi-person pose estimation including human detection and 3D human root localization.

Abstract

Although significant improvement has been achieved in 3D human pose estimation, most of the previous methods only consider a single-person case. In this work, we firstly propose a fully learning-based, camera distance-aware top-down approach for 3D multi-person pose estimation from a single RGB image. The pipeline of the proposed system consists of human detection, absolute 3D human root localization, and root-relative 3D single-person pose estimation models. Our system achieves comparable results with the state-of-the-art 3D single-person pose estimation models without any groundtruth information and significantly outperforms previous 3D multi-person pose estimation methods on publicly available datasets. The code is available in [1][2].

1. Introduction

The goal of 3D human pose estimation is to localize semantic keypoints of a human body in 3D space. It is an essential technique for human behavior understanding and human-computer interaction. Recently, many methods [26, 37, 43, 44, 49, 52] utilize deep convolutional neural networks (CNNs) and have achieved noticeable performance improvement on large-scale publicly available datasets [16, 28].

Most of the previous 3D human pose estimation methods [26, 37, 43, 44, 49, 52] are designed for single-person case. They crop the human area in an input image with a groundtruth box or the box that is predicted from a human detection model [11]. The cropped patch of a human body is fed into the 3D pose estimation model, which then estimates the 3D location of each keypoint. As their models take a single cropped image, estimating the absolute camera-centered coordinate of each keypoint is difficult. To handle this issue, many methods [26,37,43,44,49,52] estimate the relative 3D pose to a reference point in the body, e.g., the center joint (*i.e.*, pelvis) of a human, called *root*. The final 3D pose is

[1] https://github.com/mks0601/3DMPPE_ROOTNET_RELEASE

[2] https://github.com/mks0601/3DMPPE_POSENET_RELEASE

obtained by adding the 3D coordinates of the root to the estimated root-relative 3D pose. Prior information on bone length [37] or groundtruth [44] has been commonly used for the localization of the root.

Recently, many top-down approaches [6, 13, 47] for the 2D multi-person pose estimation have shown noticeable performance improvement. These approaches first detect humans by using a human detection module, and then estimate the 2D pose of each human by a 2D single-person pose estimation module. Although they are straightforward when used in 2D cases, extending them to 3D cases is challenging. Note that for the estimation of 3D multi-person poses, we need to know the absolute distance to each human from the camera as well as the 2D bounding boxes. However, existing human detectors provide 2D bounding boxes only.

In this study, we propose a general framework for 3D multi-person pose estimation. To the best of our knowledge, this study is the first to propose a fully learning-based camera distance-aware top-down approach of which components are compatible with most of the previous human detection and 3D human pose estimation methods. The pipeline of the proposed system consists of three modules. First, a human detection network (DetectNet) detects the bounding boxes of humans in an input image. Second, the proposed 3D human root localization network (RootNet) estimates the camera-centered coordinates of the detected humans' roots. Third, a root-relative 3D single-person pose estimation network (PoseNet) estimates the root-relative 3D pose for each detected human. Figures 1 and 2 show the qualitative results and overall pipeline of our framework, respectively.

We show that our approach outperforms previous 3D multi-person pose estimation methods [29, 41] on several publicly available 3D single- and multi-person pose estimation datasets [16, 29] by a large margin. Also, even without any groundtruth information (*i.e.*, the bounding box and the 3D location of the root), our method achieves comparable performance with the state-of-the-art 3D single-person pose estimation methods that use the groundtruth in the inference time. Note that our framework is new but follows previous conventions of object detection and 3D human pose estimation networks. Thus, previous detection and pose estimation methods can be easily plugged into our framework, which makes the proposed framework flexible and easy to use.

Our contributions can be summarized as follows.

- We propose a new general framework for 3D multi-person pose estimation from a single RGB image. The framework is the first fully learning-based, camera distance-aware top-down approach, of which components are compatible with most of the previous human detection and 3D human pose estimation models.

- Our framework outputs the absolute camera-centered coordinates of multiple humans' keypoints. For this, we propose a 3D human root localization network (RootNet). This model makes it easy to extend the 3D single-person pose estimation techniques to the absolute 3D pose estimation of multi-person.

- We show that our method significantly outperforms previous 3D multi-person pose estimation methods on several publicly available datasets. Also, it achieves comparable performance with the state-of-the-art 3D single-person pose estimation methods without any groundtruth information.

2. Related works

2D multi-person pose estimation. There are two main approaches in the multi-person pose estimation. The first one, top-down approach, deploys a human detector that estimates the bounding boxes of humans. Each detected human area is cropped and fed into the pose estimation network. The second one, bottom-up approach, localizes all human body keypoints in an input image first, and then groups them using some clustering techniques.

[6,13,30,31,34,47] are based on the top-down approach. Papandreou *et al.* [34] predicted 2D offset vectors and 2D heatmaps for each joint. They fused the estimated vectors and heatmaps to generate highly localized heatmaps. Chen *et al.* [6] proposed a cascaded pyramid network whose cascaded structure refines an initially estimated pose by focusing on hard keypoints. Xiao *et al.* [47] used a simple pose estimation network that consists of a deep backbone network and several upsampling layers.

[3, 14, 21, 33, 38] are based on the bottom-up approach. Cao *et al.* [3] proposed the part affinity fields (PAFs) that model the association between human body keypoints. They grouped the localized keypoints of all persons in the input image by using the estimated PAFs. Newell *et al.* [33] introduced a pixel-wise tag value to assign localized keypoints to a certain human. Kocabas *et al.* [21] proposed a pose residual network for assigning detected keypoints to each person.

3D single-person pose estimation. Current 3D single-person pose estimation methods can be categorized into single- and two-stage approaches. The single-stage approach directly localizes the 3D body keypoints from the input image. The two-stage methods utilize the high accuracy of 2D human pose estimation. They initially localize body keypoints in a 2D space and lift them to a 3D space.

[23, 37, 43–45] are based on the single-stage approach. Li *et al.* [23] proposed a multi-task framework that jointly trains both the pose regression and body part detectors. Tekin *et al.* [45] modeled high-dimensional joint dependencies by adopting an auto-encoder structure. Pavlakos *et al.* [37] extended the U-net shaped network to estimate a 3D

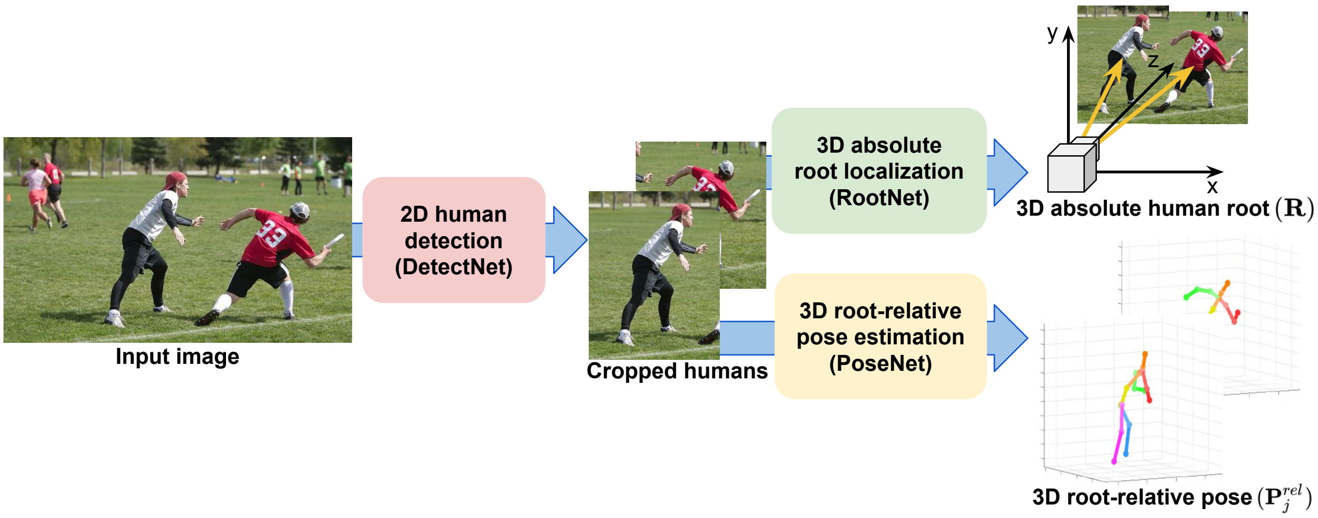
Input image
2D human detection (DetectNet)
Cropped humans
3D absolute root localization (RootNet)
3D root-relative pose estimation (PoseNet)
y
z
x
3D absolute human root $(\mathbf{R})$
3D root-relative pose $(\mathbf{P}_j^{rel})$

Figure 2: Overall pipeline of the proposed framework for 3D multi-person pose estimation from a single RGB image. The proposed framework can recover the absolute camera-centered coordinates of multiple persons' keypoints.

heatmap for each joint. They used a coarse-to-fine approach to boost performance. Sun *et al.* [43] introduced compositional loss to consider the joint connection structure. Sun *et al.* [44] used soft-argmax operation to obtain the 3D coordinates of body joints in a differentiable manner.

[4, 5, 7, 26, 35, 49, 52] are based on the two-stage approach. Park *et al.* [35] estimated the initial 2D pose and utilized it to regress the 3D pose. Martinez *et al.* [26] proposed a simple network that directly regresses the 3D coordinates of body joints from 2D coordinates. Zhou *et al.* [52] proposed a geometric loss to facilitate weakly supervised learning of the depth regression module with images in the wild. Yang *et al.* [49] utilized adversarial loss to handle the 3D human pose estimation in the wild.

3D multi-person pose estimation. Few studies have been conducted on 3D multi-person pose estimation from a single RGB image. Rogez *et al.* [41] proposed a top-down approach called LCR-Net, which consists of localization, classification, and regression parts. The localization part detects a human from an input image, and the classification part classifies the detected human into several anchor-poses. The anchor-pose is defined as a pair of 2D and root-relative 3D pose. It is generated by clustering poses in the training set. Then, the regression part refines the anchor-poses. Mehta *et al.* [29] proposed a bottom-up approach system. They introduced an occlusion-robust pose-map formulation which supports pose inference for more than one person through PAFs [3].

3D human root localization in 3D multi-person pose estimation. Rogez *et al.* [41] estimated both the 2D pose in the image coordinate space and the 3D pose in the camera-centered coordinate space simultaneously. They obtained the 3D location of the human root by minimizing the distance between the estimated 2D pose and projected 3D pose, similar to what Mehta *et al.* [28] did. However, this strategy cannot be generalized to other 3D human pose estimation methods because it requires both the 2D and 3D estimations. For example, many works [37, 44, 49, 52] estimate the 2D image coordinates and root-relative depth values of keypoints. As their methods do not output root-relative camera-centered coordinates of keypoints, such a distance minimization strategy cannot be used. Moreover, contextual information cannot be exploited because the image feature is not considered. For example, it cannot distinguish between a child close to the camera and an adult far from the camera because their scales in the 2D image is similar.

3. Overview of the proposed model

The goal of our system is to recover the absolute camera-centered coordinates of multiple persons' keypoints $\{\mathbf{P}_j^{abs}\}_{j=1}^J$, where J denotes the number of joints. To address this problem, we construct our system based on the top-down approach that consists of DetectNet, RootNet, and PoseNet. The DetectNet detects a human bounding box of each person in the input image. The RootNet takes the cropped human image from the DetectNet and localizes the root of the human $\mathbf{R} = (x_R, y_R, Z_R)$, in which x_R and y_R are pixel coordinates, and Z_R is absolute depth value. The same cropped human image is fed to the PoseNet, which estimates the root-relative 3D pose $\mathbf{P}_j^{rel} = (x_j, y_j, Z_j^{rel})$, in which x_j and y_j are pixel coordinates and Z_j^{rel} is root-relative depth value. We convert Z_j^{rel} into Z_j^{abs} by adding Z_R and transform x_j and y_j to the original image space before cropping. After back-projection formula, the final absolute 3D pose $\{\mathbf{P}_j^{abs}\}_{j=1}^J$ is obtained.

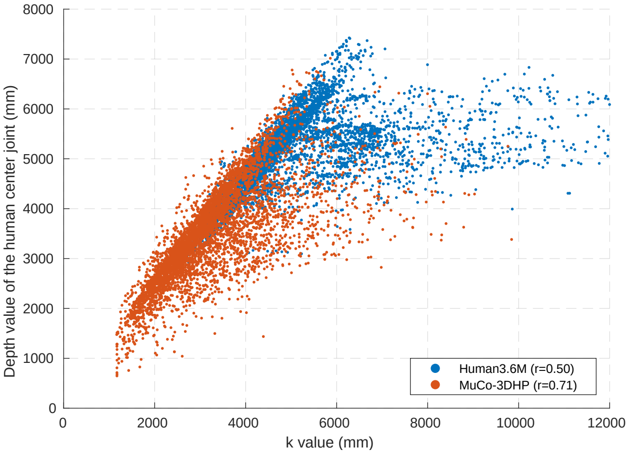

Figure 3: Correlation between k and depth value of the human root in 3D human pose estimation datasets (*i.e.*, Human3.6M [16] and MuCo-3DHP [29]). r represents Pearson correlation coefficient.

4. DetectNet

We use Mask R-CNN [11] as the framework of DetectNet. Mask R-CNN [11] consists of three parts. The first one, backbone, extracts useful local and global features from the input image by using deep residual network (ResNet) [12] and feature pyramid network [24]. Based on the extracted features, the second part, region proposal network, proposes human bounding box candidates. The RoIAlign layer extracts the features of each proposal and passes them to the third part, which is the classification head network. The head network determines whether the given proposal is a human or not and estimates bounding box refinement offsets. It achieves the state-of-the-art performance on publicly available object detection datasets [25]. Due to the high performance and publicly available code [9, 27], we use Mask R-CNN [11] as a DetectNet in our pipeline.

5. RootNet

5.1. Model design

The RootNet estimates the camera-centered coordinates of the human root $\mathbf{R} = (x_R, y_R, Z_R)$ from a cropped human image. To obtain them, RootNet separately estimates the 2D image coordinates (x_R, y_R) and a depth value (*i.e.*, the distance from the camera Z_R) of the human root. The estimated 2D image coordinates are back-projected to the camera-centered coordinate space using the estimated depth value, which becomes the final output.

Considering that an image provides sufficient information on where the human root is located in image space, the 2D estimation part can learn to localize it easily. By contrast, estimating the depth only from a cropped human image is difficult because the input does not provide information on the relative position of the camera and human. To resolve this issue, we introduce a new distance measure, k, which is defined as follows:

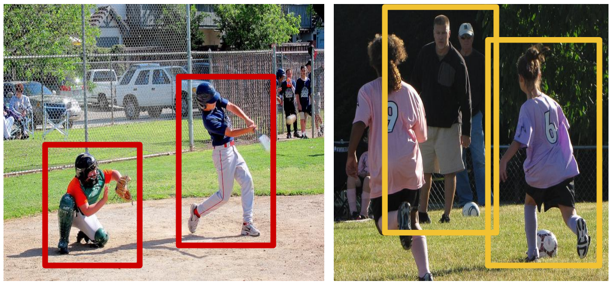
(a) Different area, same distance (b) Same area, different distance

Figure 4: Examples where k fails to represent the distance between a human and the camera because of incorrect A_{img}.

$$k = \sqrt{\alpha_x \alpha_y \frac{A_{real}}{A_{img}}}, \tag{1}$$

where α_x, α_y, A_{real}, and A_{img} are focal lengths divided by per-pixel distance factors (pixel) of x- and y-axes, the area of the human in real space (mm^2), and image space (pixel^2), respectively. k approximate the absolute depth from the camera to the object using ratio of the actual area and the imaged area of it, given camera parameters. Eq 1 can be easily derived by considering a pinhole camera projection model. The distance d (mm) between the camera and object can be calculated as follows:

$$d = \alpha_x \frac{l_{x,real}}{l_{x,img}} = \alpha_y \frac{l_{y,real}}{l_{y,img}}, \tag{2}$$

where $l_{x,real}$, $l_{x,img}$, $l_{y,real}$, $l_{y,img}$ are the lengths of an object in real space (mm) and in image space (pixel), on the x and y-axes, respectively. By multiplying the two representations of d in Eq 2 and taking the square root of it, we can have the 2D extended version of depth measure k in Eq 2. Assuming that A_{real} is constant and using α_x and α_y from datasets, the distance between the camera and an object can be measured from the area of the bounding box. As we only consider humans, we assume that A_{real} is $2000mm \times 2000mm$. The area of the human bounding box is used as A_{img} after extending it to fixed aspect ratio (*i.e.*, height:width = 1:1). Figure 3 shows that such an approximation provides a meaningful correlation between k and the real depth values of the human root in 3D human pose estimation datasets [16, 29].

Although k can represent how far the human is from the camera, it can be wrong in several cases because it assumes that A_{img} is an area of A_{real} (*i.e.*, $2000mm \times 2000mm$) in the image space when the distance between the human and

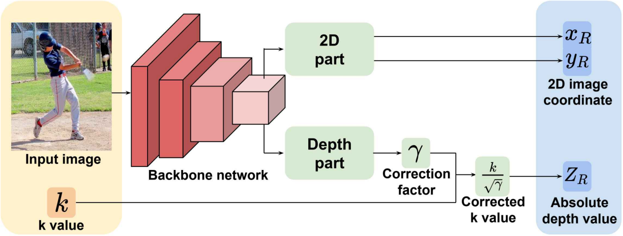

Figure 5: Network architecture of the RootNet. The RootNet estimates the 3D human root coordinate.

the camera is k. However, as A_{img} is obtained by extension of the 2D bounding box, it can have a different value according to its appearance, although the distance to the camera is the same. For example, as shown in Figure 4 (a), two humans have different A_{img} although they are at the same distance to the camera. On the other hand, in some cases, A_{img} can be the same, even with different distances from the camera. For example, in Figure 4 (b), a child and an adult have similar A_{img} however, the child is closer to the camera than the adult.

To handle this issue, we design the RootNet to utilize the image feature to correct A_{img}, eventually k. The image feature can give a clue to the RootNet about how much the A_{img} has to be changed. For example, in Figure 4 (a), the left image can tell the RootNet to increase the area because the human is in a crouching posture. Also, in Figure 4 (b), the right image can tell the RootNet to increase the area because the input image contains a child. Specifically, the RootNet outputs the correction factor γ from the image feature. The estimated γ is multiplied by the given A_{img}, which becomes A_{img}^{γ}. From A_{img}^{γ}, k is calculated and it becomes the final depth value.

5.2. Camera normalization

Our RootNet outputs correction factor γ only from an input image and does not belong to specific camera space. Therefore, the RootNet can be trained and tested with data of various α_x and α_y. Also, it can estimate 3D absolute coordinate of human root in the camera-normalized space (*i.e.*, $\alpha_x = \alpha_y = 1$) if α_x and α_y are unknown. This property makes our RootNet very flexible and useful.

5.3. Network architecture

The network architecture of the RootNet, which comprises three components, is visualized in Figure 5. First, a backbone network extracts the useful global feature of the input human image using ResNet [12]. Second, the 2D image coordinate estimation part takes a feature map from the backbone part and upsamples it using three consecutive deconvolutional layers with batch normalization layers [15] and ReLU activation function. Then, a 1-by-1 convolution is applied to produce a 2D heatmap of the root. Soft-argmax [44] extracts 2D image coordinates x_R, y_R from the 2D heatmap. The third component is the depth estimation part. It also takes a feature map from the backbone part and applies global average pooling. Then, the pooled feature map goes through a 1-by-1 convolution, which outputs a single scalar value γ. The final absolute depth value Z_R is obtained by multiplying k with $\frac{1}{\sqrt{\gamma}}$. In practical, we implemented the RootNet to output $\gamma' = \frac{1}{\sqrt{\gamma}}$ directly and multiply it with the k to obtain the absolute depth value Z_R (*i.e.*, $Z_R = \gamma' k$).

5.4. Loss function

We train the RootNet by minimizing the $L1$ distance between the estimated and groundtruth coordinates. The loss function L_{root} is defined as follows:

$$L_{root} = \|x_R^* - x_R\|_1 + \|y_R^* - y_R\|_1 + \|Z_R^* - Z_R\|_1, \quad (3)$$

where $*$ indicates the groundtruth.

6. PoseNet

6.1. Model design

The PoseNet estimates the root-relative 3D pose $\mathbf{P}_j^{rel} = (x_j, y_j, Z_j^{rel})$ from a cropped human image. Many works have been presented for this topic [26,28,37,43,44,49,52]. Among them, we use the model of Sun *et al.* [44], which is the current state-of-the-art method. This model consists of two parts. The first part is the backbone, which extracts a useful global feature from the cropped human image using ResNet [12]. Second, the pose estimation part takes a feature map from the backbone part and upsamples it using three consecutive deconvolutional layers with batch normalization layers [15] and ReLU activation function. A 1-by-1 convolution is applied to the upsampled feature map to produce the 3D heatmaps for each joint. The soft-argmax operation is used to extract the 2D image coordinates (x_j, y_j), and the root-relative depth values Z_j^{rel}.

6.2. Loss function

We train the PoseNet by minimizing the $L1$ distance between the estimated and groundtruth coordinates. The loss function L_{pose} is defined as follows:

$$L_{pose} = \frac{1}{J}\sum_{j=1}^{J}(\|x_j^* - x_j\|_1 + \|y_j^* - y_j\|_1 + \|Z_j^{rel*} - Z_j^{rel}\|_1), \quad (4)$$

where $*$ indicates groundtruth.

7. Implementation details

Publicly released Mask R-CNN model [27] pre-trained on the COCO dataset [25] is used for the DetectNet without fine-tuning on the human pose estimation datasets [16, 29]. For the RootNet and PoseNet, PyTorch [36] is used for implementation. Their backbone part is initialized with the publicly released ResNet-50 [12] pre-trained on the ImageNet dataset [42], and the weights of the remaining part are initialized by Gaussian distribution with $\sigma = 0.001$. The weights are updated by the Adam optimizer [20] with a mini-batch size of 128. The initial learning rate is set to 1×10^{-3} and reduced by a factor of 10 at the 17th epoch. We use 256×256 as the size of the input image of the RootNet and PoseNet. We perform data augmentation including rotation (±30°), horizontal flip, color jittering, and synthetic occlusion [51] in training. Horizontal flip augmentation is performed in testing for the PoseNet following Sun *et al.* [44]. We train the RootNet and PoseNet for 20 epochs with four NVIDIA 1080 Ti GPUs, which took two days, respectively.

8. Experiment

8.1. Dataset and evaluation metric

Human3.6M dataset. Human3.6M dataset [16] is the largest 3D single-person pose benchmark. It consists of 3.6 millions of video frames. 11 subjects performing 15 activities are captured from 4 camera viewpoints. The groundtruth 3D poses are obtained using a motion capture system. Two evaluation metrics are widely used. The first one is mean per joint position error (MPJPE) [16], which is calculated after aligning the human root of the estimated and groundtruth 3D poses. The second one is MPJPE after further alignment (*i.e.*, Procrustes analysis (PA) [10]). This metric is called PA MPJPE. To evaluate the localization of the absolute 3D human root, we introduce the mean of the Euclidean distance between the estimated coordinates of the root $\mathbf{R}$ and the ground truth ones $\mathbf{R}^*$, *i.e.*, the mean of the root position error (MRPE), as a new metric:

$$MRPE = \frac{1}{N} \sum_{i=1}^{N} ||\mathbf{R}^{(i)} - \mathbf{R}^{(i)*}||_2, \tag{5}$$

where superscript i is the index of the sample, and N denotes the total number of test samples.

MuCo-3DHP and MuPoTS-3D datasets. Mehta *et al.* [29] proposed a 3D multi-person pose estimation dataset. The training set, MuCo-3DHP, is generated by compositing the existing MPI-INF-3DHP 3D single-person pose estimation dataset [28]. The test set, MuPoTS-3D dataset, was captured at outdoors and it includes 20 real-world scenes with groundtruth 3D poses for up to three subjects. The groundtruth is obtained with a multi-view marker-less motion capture system. For evaluation, a 3D percentage of correct keypoints (3DPCK$_{rel}$) and area under 3DPCK curve from various threshold (AUC$_{rel}$) is used after root alignment with groundtruth. It treats a joint's prediction as correct if it lines within a 15cm from the groundtruth joint location. We additionally define 3DPCK$_{abs}$ which is the 3DPCK without root alignment to evaluate absolute camera-centered coordinates. To evaluate the localization of the absolute 3D human root, we use the average precision of 3D human root location (AP^{root}_{25}) which considers a prediction is correct when the Euclidean distance between the estimated and groundtruth coordinates is smaller than 25cm.

Settings	MRPE	MPJPE	Time
Joint learning	138.2	116.7	**0.132**
Disjointed learning (Ours)	**120.0**	**57.3**	0.141

Table 1: MRPE, MPJPE, and seconds per frame comparison between joint and disjointed learning on Human3.6M dataset.

DetectNet	RootNet	AP^{box}	AP^{root}_{25}	AUC$_{rel}$	3DPCK$_{abs}$
R-50	k	43.8	5.2	39.2	9.6
R-50	Ours	43.8	28.5	39.8	31.5
X-101-32	Ours	**45.0**	**31.0**	**39.8**	**31.5**
GT	Ours	100.0	31.4	39.8	31.6
GT	GT	100.0	100.0	39.8	80.2

Table 2: Overall performance comparison for different DetectNet and RootNet settings on the MuPoTS-3D dataset.

8.2. Experimental protocol

Human3.6M dataset. Two experimental protocols are widely used. *Protocol 1* uses six subjects (S1, S5, S6, S7, S8, S9) in training and S11 in testing. PA MPJPE is used as an evaluation metric. *Protocol 2* uses five subjects (S1, S5, S6, S7, S8) in training and two subjects (S9, S11) in testing. MPJPE is used as an evaluation metric. We use every 5th and 64th frame of videos in training and testing, respectively following [43, 44]. When training on the Human3.6M dataset, we used additional MPII 2D human pose estimation dataset [1] following [37, 43, 44, 52]. Each mini-batch consists of half Human3.6M and half MPII data. For MPII data, the loss value of the z-axis becomes zero for both of the RootNet and PoseNet following Sun *et al.* [44].

MuCo-3DHP and MuPoTS-3D datasets. Following the previous protocol, we composite 400K frames of which half are background augmented. For augmentation, we use images from the COCO dataset [25] except for images with humans. We use an additional COCO 2D human keypoint detection dataset [25] when training our models on the MuCo-3DHP dataset following Mehta *et al.* [29]. Each mini-batch consists of half MuCo-3DHP and half COCO data. For COCO data, loss value of z-axis becomes zero for both of the RootNet and PoseNet following Sun *et al.* [44].

Methods	Dir.	Dis.	Eat	Gre.	Phon.	Pose	Pur.	Sit	SitD.	Smo.	Phot.	Wait	Walk	WalkD.	WalkP.	Avg
With groundtruth information in inference time																
Yasin [50]	88.4	72.5	108.5	110.2	97.1	81.6	107.2	119.0	170.8	108.2	142.5	86.9	92.1	165.7	102.0	108.3
Rogez [40]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	88.1
Chen [5]	71.6	66.6	74.7	79.1	70.1	67.6	89.3	90.7	195.6	83.5	93.3	71.2	55.7	85.9	62.5	82.7
Moreno [32]	67.4	63.8	87.2	73.9	71.5	69.9	65.1	71.7	98.6	81.3	93.3	74.6	76.5	77.7	74.6	76.5
Zhou [53]	47.9	48.8	52.7	55.0	56.8	49.0	45.5	60.8	81.1	53.7	65.5	51.6	50.4	54.8	55.9	55.3
Martinez [26]	39.5	43.2	46.4	47.0	51.0	41.4	40.6	56.5	69.4	49.2	56.0	45.0	38.0	49.5	43.1	47.7
Kanazawa [19]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	56.8
Sun [43]	42.1	44.3	45.0	45.4	51.5	43.2	41.3	59.3	73.3	51.0	53.0	44.0	38.3	48.0	44.8	48.3
Fang [7]	38.2	41.7	43.7	44.9	48.5	40.2	38.2	54.5	64.4	47.2	55.3	44.3	36.7	47.3	41.7	45.7
Sun [44]	36.9	36.2	40.6	40.4	41.9	34.9	35.7	50.1	59.4	40.4	44.9	39.0	30.8	39.8	36.7	40.6
Ours (PoseNet)	**31.0**	**30.6**	**39.9**	**35.5**	**34.8**	**30.2**	**32.1**	**35.0**	**43.8**	**35.7**	**37.6**	**30.1**	**24.6**	**35.7**	**29.3**	**34.0**
Without groundtruth information in inference time																
Ours (Full)	**32.5**	**31.5**	**41.5**	**36.7**	**36.3**	**31.9**	**33.2**	**36.5**	**44.4**	**36.7**	**38.7**	**31.2**	**25.6**	**37.1**	**30.5**	**35.2**

Table 3: PA MPJPE comparison with state-of-the-art methods on the Human3.6M dataset using Protocol 1.

Methods	Dir.	Dis.	Eat	Gre.	Phon.	Pose	Pur.	Sit	SitD.	Smo.	Phot.	Wait	Walk	WalkD.	WalkP.	Avg
With groundtruth information in inference time																
Chen [5]	89.9	97.6	90.0	107.9	107.3	93.6	136.1	133.1	240.1	106.7	139.2	106.2	87.0	114.1	90.6	114.2
Tome [46]	65.0	73.5	76.8	86.4	86.3	68.9	74.8	110.2	173.9	85.0	110.7	85.8	71.4	86.3	73.1	88.4
Moreno [32]	69.5	80.2	78.2	87.0	100.8	76.0	69.7	104.7	113.9	89.7	102.7	98.5	79.2	82.4	77.2	87.3
Zhou [53]	68.7	74.8	67.8	76.4	76.3	84.0	70.2	88.0	113.8	78.0	98.4	90.1	62.6	75.1	73.6	79.9
Jahangiri [17]	74.4	66.7	67.9	75.2	77.3	70.6	64.5	95.6	127.3	79.6	79.1	73.4	67.4	71.8	72.8	77.6
Mehta [28]	57.5	68.6	59.6	67.3	78.1	56.9	69.1	98.0	117.5	69.5	82.4	68.0	55.3	76.5	61.4	72.9
Martinez [26]	51.8	56.2	58.1	59.0	69.5	55.2	58.1	74.0	94.6	62.3	78.4	59.1	49.5	65.1	52.4	62.9
Kanazawa [19]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	88.0
Fang [7]	50.1	54.3	57.0	57.1	66.6	53.4	55.7	72.8	88.6	60.3	73.3	57.7	47.5	62.7	50.6	60.4
Sun [43]	52.8	54.8	54.2	54.3	61.8	53.1	53.6	71.7	86.7	61.5	67.2	53.4	47.1	61.6	63.4	59.1
Sun [44]	**47.5**	**47.7**	**49.5**	**50.2**	**51.4**	**43.8**	**46.4**	**58.9**	**65.7**	**49.4**	**55.8**	**47.8**	**38.9**	**49.0**	**43.8**	**49.6**
Ours (PoseNet)	50.5	55.7	50.1	51.7	53.9	46.8	50.0	61.9	68.0	52.5	55.9	49.9	41.8	56.1	46.9	53.3
Without groundtruth information in inference time																
Rogez [41]	76.2	80.2	75.8	83.3	92.2	79.9	71.7	105.9	127.1	88.0	105.7	83.7	64.9	86.6	84.0	87.7
Mehta [29]	58.2	67.3	61.2	65.7	75.8	62.2	64.6	82.0	93.0	68.8	84.5	65.1	57.6	72.0	63.6	69.9
Ours (Full)	**51.5**	**56.8**	**51.2**	**52.2**	**55.2**	**47.7**	**50.9**	**63.3**	**69.9**	**54.2**	**57.4**	**50.4**	**42.5**	**57.5**	**47.7**	**54.4**

Table 4: MPJPE comparison with state-of-the-art methods on the Human3.6M dataset using Protocol 2.

8.3. Ablation study

In this study, we show how each component of our proposed framework affects the 3D multi-person pose estimation accuracy. To evaluate the performance of the DetectNet, we use the average precision of bounding box (AP^{box}) following metrics of the COCO object detection benchmark [25].

Disjointed pipeline. To demonstrate the effectiveness of the disjointed pipeline (*i.e.*, separated DetectNet, RootNet, and PoseNet), we compare MRPE, MPJPE, and running time of joint and disjointed learning of the RootNet and PoseNet in Table 1. The running time includes DetectNet and is measured using a single TitanX Maxwell GPU. For the joint learning, we combine the RootNet and PoseNet into a single model which shares backbone part (*i.e.*, ResNet [12]). The image feature from the backbone is fed to each branch of RootNet and PoseNet in a parallel way. Compared with the joint learning, our disjointed learning gives lower error under a similar running time. We believe that this is because each task of RootNet and PoseNet is not highly correlated so that jointly training all tasks can make training harder, resulting in lower accuracy.

Effect of the DetectNet. To show how the performance of the human detection affects the accuracy of the final 3D human root localization and 3D multi-person pose estimation, we compare AP_{25}^{root}, AUC_{rel}, and 3DPCK_{abs} using the DetectNet in various backbones (*i.e.*, ResNet-50 [12], ResNeXt-101-32 [48]) and groundtruth box in the second, third, and fourth row of Table 2, respectively. The table shows that based on the same RootNet (*i.e.*, Ours), better human detection model improves both of the 3D human root localization and 3D multi-person pose estimation performance. However, the groundtruth box does not improve overall accuracy considerably compared with other DetectNet models. Therefore, we have sufficient reasons to believe that the given boxes cover most of the person instances with such a high detection AP. We can also conclude that the bounding box estimation accuracy does not have a large impact on the 3D multi-person pose estimation accuracy.

Effect of the RootNet. To show how the performance

Methods	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12	S13	S14	S15	S16	S17	S18	S19	S20	Avg
Accuracy for all groundtruths																					
Rogez [41]	67.7	49.8	53.4	59.1	67.5	22.8	43.7	49.9	31.1	78.1	50.2	51.0	51.6	49.3	56.2	66.5	65.2	62.9	66.1	59.1	53.8
Mehta [29]	81.0	60.9	64.4	63.0	69.1	30.3	65.0	59.6	64.1	83.9	68.0	68.6	62.3	59.2	70.1	80.0	79.6	67.3	66.6	67.2	66.0
Ours	**94.4**	**77.5**	**79.0**	**81.9**	**85.3**	**72.8**	**81.9**	**75.7**	**90.2**	**90.4**	**79.2**	**79.9**	**75.1**	**72.7**	**81.1**	**89.9**	**89.6**	**81.8**	**81.7**	**76.2**	**81.8**
Accuracy only for matched groundtruths																					
Rogez [41]	69.1	67.3	54.6	61.7	74.5	25.2	48.4	63.3	69.0	78.1	53.8	52.2	60.5	60.9	59.1	70.5	76.0	70.0	77.1	81.4	62.4
Mehta [29]	81.0	65.3	64.6	63.9	75.0	30.3	65.1	61.1	64.1	83.9	72.4	69.9	71.0	72.9	71.3	83.6	79.6	73.5	78.9	**90.9**	70.8
Ours	**94.4**	**78.6**	**79.0**	**82.1**	**86.6**	**72.8**	**81.9**	**75.8**	**90.2**	**90.4**	**79.4**	**79.9**	**75.3**	**81.0**	**81.0**	**90.7**	**89.6**	**83.1**	**81.7**	77.3	**82.5**

Table 5: Sequence-wise 3DPCK$_{rel}$ comparison with state-of-the-art methods on the MuPoTS-3D dataset.

Methods	Hd.	Nck.	Sho.	Elb.	Wri.	Hip	Kn.	Ank.	Avg
Rogez [41]	49.4	67.4	57.1	51.4	41.3	84.6	56.3	36.3	53.8
Mehta [29]	62.1	81.2	77.9	57.7	47.2	**97.3**	66.3	47.6	66.0
Ours	**79.1**	**92.6**	**85.1**	**79.4**	**67.0**	96.6	**85.7**	**73.1**	**81.8**

Table 6: Joint-wise 3DPCK$_{rel}$ comparison with state-of-the-art methods on the MuPoTS-3D dataset. All groundtruths are used for evaluation.

of the 3D human root localization affects the accuracy of the 3D multi-person pose estimation, we compare AUC$_{rel}$ and 3DPCK$_{abs}$ using various RootNet settings in Table 2. The first and second rows show that based on the same DetectNet (*i.e.*, R-50), our RootNet exhibits significantly higher AP$_{25}^{root}$ and 3DPCK$_{abs}$ compared with the setting in which k is directly utilized as a depth value. We use the x and y of the RootNet when the k is used as a depth value. This result demonstrates that the RootNet successfully corrects the k value. The fourth and last rows show that the groundtruth human root provides similar AUC$_{rel}$, but significantly higher 3DPCK$_{abs}$ compared with our RootNet. This finding shows that better human root localization is required to achieve more accurate absolute 3D multi-person pose estimation results.

Effect of the PoseNet. All settings in Table 2 provides similar AUC$_{rel}$. Especially, the first and last rows of the table show that using groundtruth box and human root does not provide significantly higher AUC$_{rel}$. As the results in the table are based on the same PoseNet, we can conclude that AUC$_{rel}$, which is an evaluation of the root-relative 3D human pose estimation highly depends on the accuracy of the PoseNet.

8.4. Comparison with state-of-the-art methods

Human3.6M dataset. We compare our proposed system with state-of-the-art 3D human pose estimation methods on the Human3.6M dataset [16] in Tables 3 and 4. As most of the previous methods use the groundtruth information (*i.e.*, bounding box or 3D root location) in inference time, we report the performance of the PoseNet using the groundtruth 3D root location. Note that our full model does not require any groundtruth information in inference time. The tables show that our method achieves comparable performance despite not using any groundtruth information in inference time. Moreover, it significantly outperforms previous 3D multi-person pose estimation methods [25, 29].

MuCo-3DHP and MuPoTS-3D datasets. We compare our proposed system with the state-of-the-art 3D multi-person pose estimation methods on the MuPoTS-3D dataset [29] in Tables 5 and 6. The proposed system significantly outperforms them in most of the test sequences and joints.

Those comparisons clearly show that our approach outperforms previous 3D multi-person pose estimation methods.

9. Discussion

Although our proposed method outperforms previous 3D multi-person pose estimation methods by a large margin, room for improvement is substantial. As shown in Table 2, using the groundtruth 3D root location brings significant 3DPCK$_{abs}$ improvement. Recent advances in depth map estimation from a single RGB image [8, 22] can give a clue for improving the 3D human root localization model.

Our framework can also be used in applications other than 3D multi-person pose estimation. For example, recent methods for 3D human mesh model reconstruction [2, 18, 19] reconstruct full 3D mesh model from a single person. Joo *et al.* [18] utilized 2D multi-view input for 3D multi-person mesh model reconstruction. In our framework, if the PoseNet is replaced with existing human mesh reconstruction model [2, 18, 19], 3D multi-person mesh model reconstruction can be performed from *a single RGB image*. This shows our framework can be applied to many 3D instance-aware vision tasks which take a single RGB image as an input.

10. Conclusion

We propose a novel and general framework for 3D multi-person pose estimation from a single RGB image. Our framework consists of human detection, 3D human root localization, and root-relative 3D single-person pose estimation models. Since any existing human detection and 3D single-person pose estimation models can be plugged into our framework, it is very flexible and easy to use. The proposed system outperforms previous 3D multi-person pose estimation methods by a large margin and achieves comparable performance with 3D single-person pose estimation methods without any groundtruth information while they use it in inference time. To the best of our knowledge, this work is the first to propose a fully learning-based camera distance-aware top-down approach whose components are compatible with most of the previous human detection and 3D human pose estimation models. We hope that this study provides a new basis for 3D multi-person pose estimation, which has only barely been explored.

Supplementary Material of "Camera Distance-aware Top-down Approach for 3D Multi-person Pose Estimation from a Single RGB Image"

In this supplementary material, we present more experimental results that could not be included in the main manuscript due to the lack of space.

1. Derivation of Equation 1

We provide a derivation of Equation 1 of the main manuscript with reference to Figure 6 ,which shows a pinhole camera model. The green and blue arrows represent the human root joint centered x and y-axes, respectively. The yellow lines show rays, and c is the hole. d, f, and l_{sensor} are distance between camera and the human root joint (mm), focal length (mm), and the length of human on the image sensor (mm), respectively.

According to the definition of $\tan$,

$$\tan\theta_x = \frac{0.5 l_{x,real}}{d} = \frac{0.5 l_{x,sensor}}{f},$$

Let p_x be per pixel distance factor in x-axis. Then,

$$d = f\frac{l_{x,real}}{l_{x,sensor}} = f p_x \frac{l_{x,real}}{l_{x,sensor} p_x} = \alpha_x \frac{l_{x,real}}{l_{x,img}},$$

Above equations are also valid in y-axis. Therefore,

$$d = f\frac{l_{y,real}}{l_{y,sensor}} = f p_y \frac{l_{y,real}}{l_{y,sensor} p_y} = \alpha_y \frac{l_{y,real}}{l_{y,img}},$$

Finally,

$$d = \sqrt{\alpha_x \alpha_y \frac{l_{x,real}}{l_{x,img}} \frac{l_{y,real}}{l_{y,img}}} = \sqrt{\alpha_x \alpha_y \frac{A_{real}}{A_{img}}}.$$

2. Comparison of 3D human root localization with previous approaches

We compare previous absolute 3D human root localization methods [28,41] with the proposed RootNet on the Human3.6M dataset [16] based on protocol 2.

Previous approaches [28,41] simultaneously estimate 2D image coordinates and 3D camera-centered root-relative coordinates of keypoints. Then, absolute camera-centered coordinates of the human root are obtained by minimizing the distance between 2D predictions and projected 3D predictions. For optimization, linear least-squares formulation is used. To measure the errors of their method, we implemented and used ResNet-152-based model of Sun *et al.* [44] as a 2D pose estimator and model of Martinez *et al.* [26] as a 3D pose estimator, which are state-of-the-art methods. In addition, to minimize the effect of outliers in 3D-to-2D fitting, we excluded limb joints when fitting. Also, we performed RANSAC with a various number of joints to get optimal joint set for fitting instead of using heuristically selected joint set.

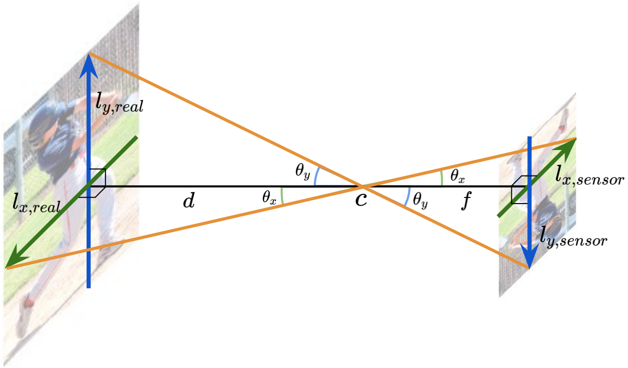

Figure 6: Visualization of a pinhole camera model.

Methods	MRPE	MRPE$_x$	MRPE$_y$	MRPE$_z$
Baseline [28,41]	267.8	27.5	28.3	261.9
W/o limb joints	226.2	24.5	24.9	220.2
RANSAC	213.1	24.3	24.3	207.1
RootNet (Ours)	**120.0**	**23.3**	**23.0**	**108.1**

Table 7: MRPE comparisons between previous distance minimization-based approaches [28, 41] and our RootNet on the Human3.6M dataset. MRPE$_x$, MRPE$_y$, and MRPE$_z$ represent the mean of the errors in the x, y, and z axes, respectively.

DetectNet	RootNet	PoseNet	**Total**
0.120	0.010	0.011	**0.141**

Table 8: Seconds per frame for each component of our framework.

Table 7 shows our RootNet significantly outperforms previous approaches. Furthermore, the RootNet can be designed independently of the PoseNet, giving design flexibility to both models. In contrast, the previous 3D root localization methods [28, 41] require both of 2D and 3D predictions for the root localization, which results in lack of generalizability.

3. Running time of the proposed framework

In Table 8, we report seconds per frame for each component of our framework. The running time is measured using a single TitanX Maxwell GPU. As the table shows, most of the running time is consumed by DetectNet. It is hard to directly compare running time with previous works [28,41] because they did not report it. However, we guess that there would be no big difference because models of [41] and [28] are similar with [39] and [3] whose speed is 0.2 and 0.11

Methods	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12	S13	S14	S15	S16	S17	S18	S19	S20	Avg
Accuracy for all groundtruths																					
Ours	59.5	44.7	51.4	46.0	52.2	27.4	23.7	26.4	39.1	23.6	18.3	14.9	38.2	26.5	36.8	23.4	14.4	19.7	18.8	25.1	31.5
Accuracy only for matched groundtruths																					
Ours	59.5	45.3	51.4	46.2	53.0	27.4	23.7	26.4	39.1	23.6	18.3	14.9	38.2	29.5	36.8	23.6	14.4	20.0	18.8	25.4	31.8

Table 9: Sequence-wise 3DPCK$_{abs}$ on the MuPoTS-3D dataset.

Methods	Hd.	Nck.	Sho.	Elb.	Wri.	Hip	Kn.	Ank.	Avg
Ours	37.3	35.3	33.7	33.8	30.4	30.3	31.0	25.0	31.5

Table 10: Joint-wise 3DPCK$_{abs}$ on the MuPoTS-3D dataset. All groundtruths are used for evaluation.

seconds per frame, respectively.

4. Absolute 3D multi-person pose estimation errors

For the continual study of the 3D multi-person pose estimation, we report 3DPCK$_{abs}$ in Table 9 and 10. As previous works [25, 29] did not report 3DPCK$_{abs}$, we only report our result.

5. Qualitative results

Figures 7 and 8 show qualitative results of our 3D multi-person pose estimation framework on the MuPoTS-3D [29] and COCO [25] datasets, respectively. Note that COCO dataset consists of *in-the-wild* images which are hardly included in the 3D human pose estimation training sets [16, 29].

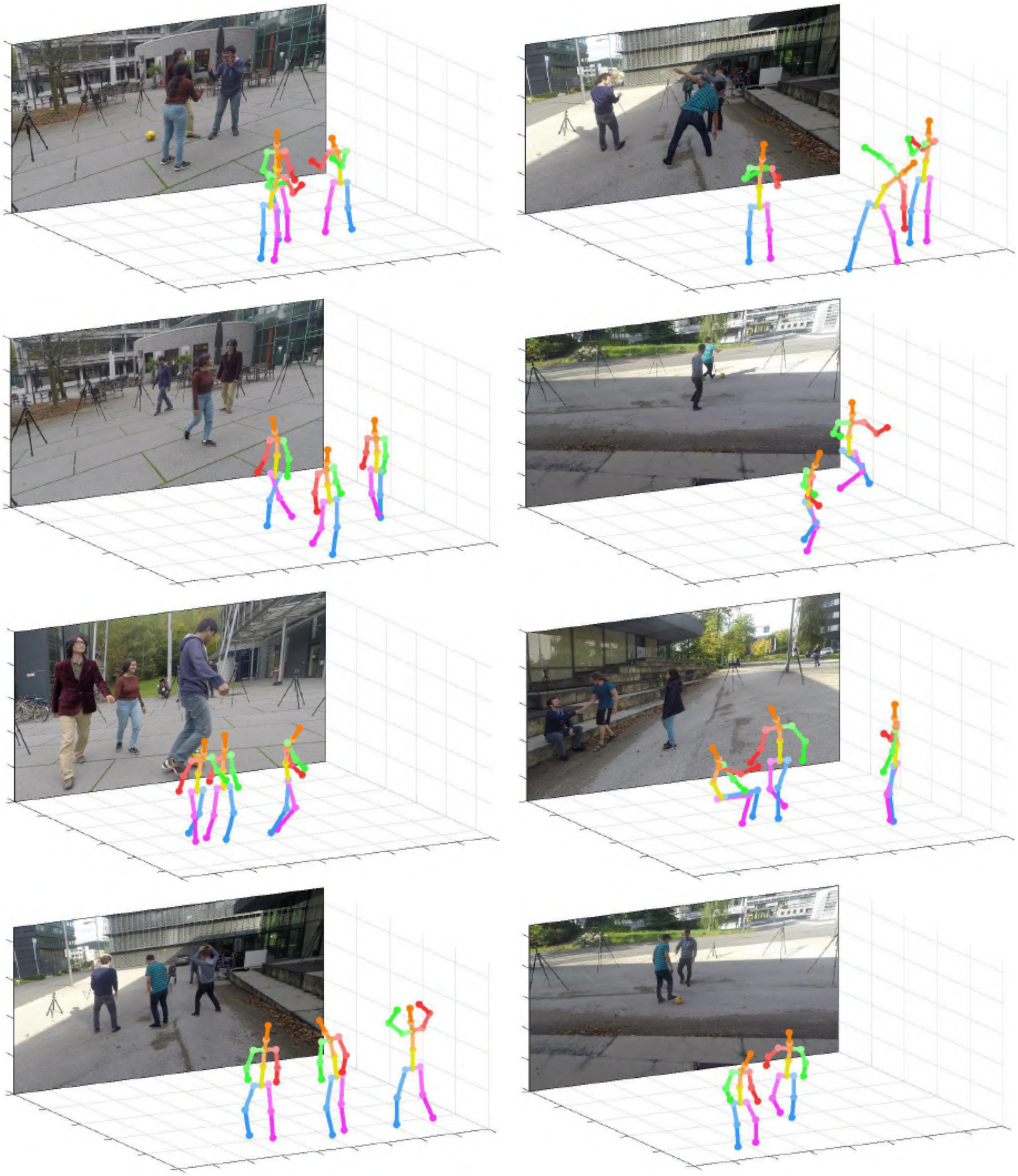

Figure 7: Qualitative results of applying our method on the MuPoTS-3D dataset [29].

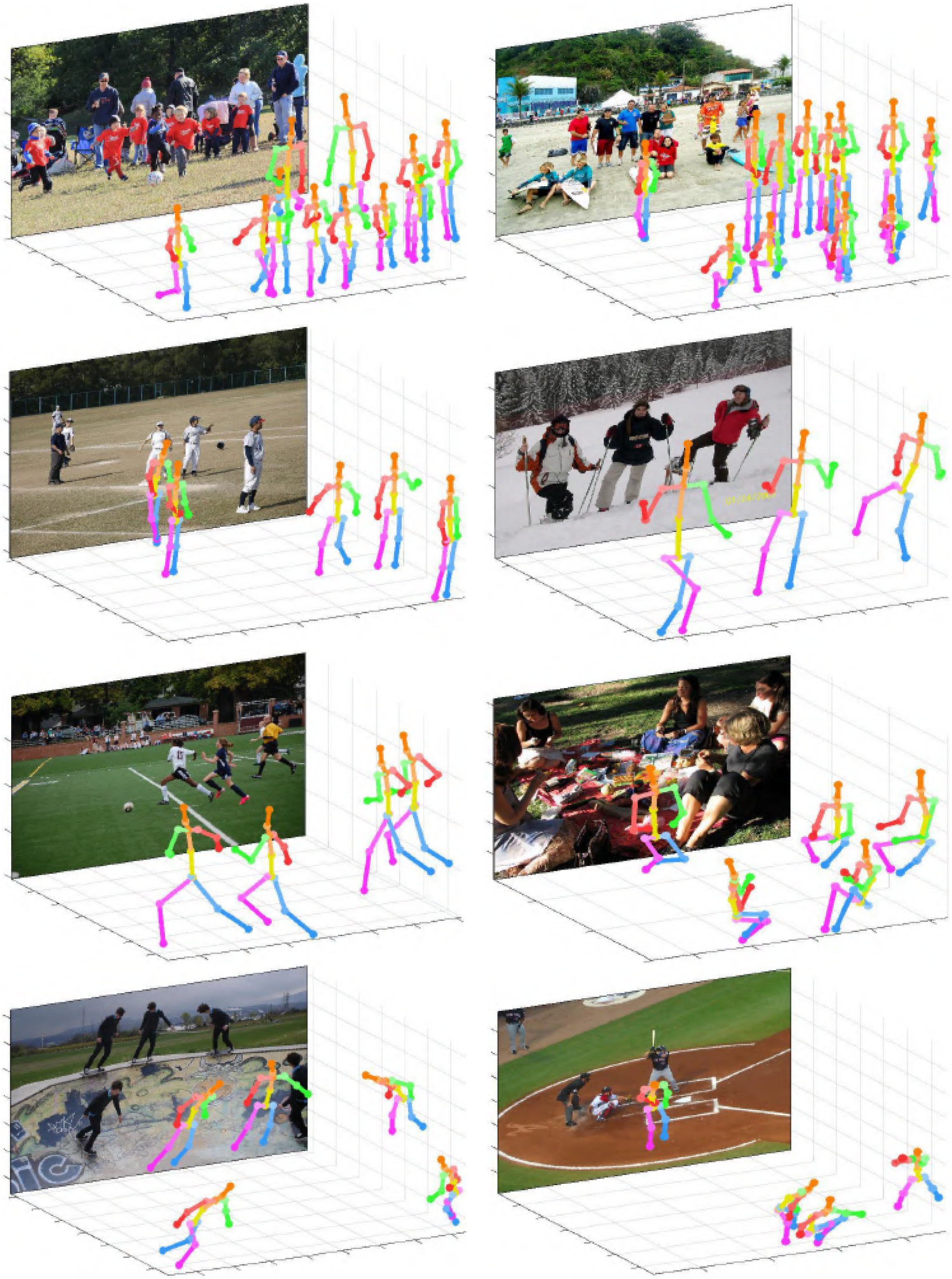

Figure 8: Qualitative results of applying our method on the COCO 2017 [25] validation set.

References

[1] M. Andriluka, L. Pishchulin, P. Gehler, and B. Schiele. 2d human pose estimation: New benchmark and state of the art analysis. In *CVPR*, 2014.

[2] F. Bogo, A. Kanazawa, C. Lassner, P. Gehler, J. Romero, and M. J. Black. Keep it smpl: Automatic estimation of 3d human pose and shape from a single image. In *ECCV*, 2016.

[3] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. *CVPR*, 2017.

[4] J. Y. Chang and K. M. Lee. 2d–3d pose consistency-based conditional random fields for 3d human pose estimation. *CVIU*, 2018.

[5] C.-H. Chen and D. Ramanan. 3d human pose estimation= 2d pose estimation+ matching. In *CVPR*, 2017.

[6] Y. Chen, Z. Wang, Y. Peng, Z. Zhang, G. Yu, and J. Sun. Cascaded pyramid network for multi-person pose estimation. In *CVPR*, 2018.

[7] H.-S. Fang, Y. Xu, W. Wang, X. Liu, and S.-C. Zhu. Learning pose grammar to encode human body configuration for 3d pose estimation. In *AAAI*, 2018.

[8] H. Fu, M. Gong, C. Wang, K. Batmanghelich, and D. Tao. Deep ordinal regression network for monocular depth estimation. In *CVPR*, 2018.

[9] R. Girshick, I. Radosavovic, G. Gkioxari, P. Dollár, and K. He. Detectron. `https://github.com/facebookresearch/detectron`, 2018.

[10] J. C. Gower. Generalized procrustes analysis. *Psychometrika*, 1975.

[11] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn. In *ICCV*, 2017.

[12] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[13] S. Huang, M. Gong, and D. Tao. A coarse-fine network for keypoint localization. In *ICCV*, 2017.

[14] E. Insafutdinov, L. Pishchulin, B. Andres, M. Andriluka, and B. Schiele. Deepercut: A deeper, stronger, and faster multi-person pose estimation model. In *ECCV*, 2016.

[15] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *ICML*, 2015.

[16] C. Ionescu, D. Papava, V. Olaru, and C. Sminchisescu. Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *TPAMI*, 2014.

[17] E. Jahangiri and A. L. Yuille. Generating multiple diverse hypotheses for human 3d pose consistent with 2d joint detections. In *ICCV*, 2017.

[18] H. Joo, T. Simon, and Y. Sheikh. Total capture: A 3d deformation model for tracking faces, hands, and bodies. In *CVPR*, 2018.

[19] A. Kanazawa, M. J. Black, D. W. Jacobs, and J. Malik. End-to-end recovery of human shape and pose. In *CVPR*, 2018.

[20] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *ICLR*, 2014.

[21] M. Kocabas, S. Karagoz, and E. Akbas. Multiposenet: Fast multi-person pose estimation using pose residual network. In *ECCV*, 2018.

[22] R. Li, K. Xian, C. Shen, Z. Cao, H. Lu, and L. Hang. Deep attention-based classification network for robust depth prediction. *arXiv preprint arXiv:1807.03959*, 2018.

[23] S. Li and A. B. Chan. 3d human pose estimation from monocular images with deep convolutional neural network. In *ACCV*, 2014.

[24] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie. Feature pyramid networks for object detection. In *CVPR*, 2017.

[25] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014.

[26] J. Martinez, R. Hossain, J. Romero, and J. J. Little. A simple yet effective baseline for 3d human pose estimation. In *ICCV*, 2017.

[27] F. Massa and R. Girshick. maskrcnn-benchmark: Fast, modular reference implementation of Instance Segmentation and Object Detection algorithms in PyTorch. `https://github.com/facebookresearch/maskrcnn-benchmark`, 2018.

[28] D. Mehta, H. Rhodin, D. Casas, P. Fua, O. Sotnychenko, W. Xu, and C. Theobalt. Monocular 3d human pose estimation in the wild using improved cnn supervision. In *3DV*, 2017.

[29] D. Mehta, O. Sotnychenko, F. Mueller, W. Xu, S. Sridhar, G. Pons-Moll, and C. Theobalt. Single-shot multi-person 3d pose estimation from monocular rgb. In *3DV*, 2018.

[30] G. Moon, J. Y. Chang, and K. M. Lee. Multi-scale aggregation r-cnn for 2d multi-person pose estimation. *CVPRW*, 2019.

[31] G. Moon, J. Y. Chang, and K. M. Lee. Posefix: Model-agnostic general human pose refinement network. In *CVPR*, 2019.

[32] F. Moreno-Noguer. 3d human pose estimation from a single image via distance matrix regression. In *CVPR*, 2017.

[33] A. Newell, Z. Huang, and J. Deng. Associative embedding: End-to-end learning for joint detection and grouping. In *NIPS*, 2017.

[34] G. Papandreou, T. Zhu, N. Kanazawa, A. Toshev, J. Tompson, C. Bregler, and K. Murphy. Towards accurate multi-person pose estimation in the wild. In *CVPR*, 2017.

[35] S. Park, J. Hwang, and N. Kwak. 3d human pose estimation using convolutional neural networks with 2d pose information. In *ECCV*, 2016.

[36] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer. Automatic differentiation in pytorch. 2017.

[37] G. Pavlakos, X. Zhou, K. G. Derpanis, and K. Daniilidis. Coarse-to-fine volumetric prediction for single-image 3d human pose. In *CVPR*, 2017.

[38] L. Pishchulin, E. Insafutdinov, S. Tang, B. Andres, M. Andriluka, P. V. Gehler, and B. Schiele. Deepcut: Joint subset partition and labeling for multi person pose estimation. In *CVPR*, 2016.

[39] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NIPS*, 2015.

[40] G. Rogez and C. Schmid. Mocap-guided data augmentation for 3d pose estimation in the wild. In *NIPS*, 2016.

[41] G. Rogez, P. Weinzaepfel, and C. Schmid. Lcr-net: Localization-classification-regression for human pose. In *CVPR*, 2017.

[42] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al. Imagenet large scale visual recognition challenge. *IJCV*, 2015.

[43] X. Sun, J. Shang, S. Liang, and Y. Wei. Compositional human pose regression. In *ICCV*, 2017.

[44] X. Sun, B. Xiao, F. Wei, S. Liang, and Y. Wei. Integral human pose regression. In *ECCV*, 2018.

[45] B. Tekin, I. Katircioglu, M. Salzmann, V. Lepetit, and P. Fua. Structured prediction of 3d human pose with deep neural networks. *BMVC*, 2016.

[46] D. Tome, C. Russell, and L. Agapito. Lifting from the deep: Convolutional 3d pose estimation from a single image. In *CVPR*, 2017.

[47] B. Xiao, H. Wu, and Y. Wei. Simple baselines for human pose estimation and tracking. In *ECCV*, 2018.

[48] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He. Aggregated residual transformations for deep neural networks. In *CVPR*, 2017.

[49] W. Yang, W. Ouyang, X. Wang, J. Ren, H. Li, and X. Wang. 3d human pose estimation in the wild by adversarial learning. In *CVPR*, 2018.

[50] H. Yasin, U. Iqbal, B. Kruger, A. Weber, and J. Gall. A dual-source approach for 3d pose estimation from a single image. In *CVPR*, 2016.

[51] Z. Zhong, L. Zheng, G. Kang, S. Li, and Y. Yang. Random erasing data augmentation. *arXiv preprint arXiv:1708.04896*, 2017.

[52] X. Zhou, Q. Huang, X. Sun, X. Xue, and Y. Wei. Weaklysupervised transfer for 3d human pose estimation in the wild. In *ICCV*, 2017.

[53] X. Zhou, M. Zhu, G. Pavlakos, S. Leonardos, K. G. Derpanis, and K. Daniilidis. Monocap: Monocular human motion capture using a cnn coupled with a geometric prior. *TPAMI*, 2019.