Supplementary Material: Monocular 3D Multi-Person Pose Estimation by Integrating Top-Down and Bottom-Up Networks

Yu Cheng[1], Bo Wang[2], Bo Yang[2], Robby T. Tan[1,3]
[1]National University of Singapore
[2]Tencent Game AI Research Center
[3]Yale-NUS College

e0321276@u.nus.edu, {bohawkwang,brandonyang}@tencent.com, robby.tan@nus.edu.sg

1. Network Structure

GCN Structure Unlike existing GCN methods which use an undirected graph [6, 2], we use a directed graph. The advantage of using directed graph is that more reliable joints with higher confidence are capable to influence the unreliable ones with low confidence with non-symmetric adjacency matrix. We adopt the GCN method following [4].

The features are propagated according to an adjacent matrix in GCNs, implying the edge values in the propagation graph. Given the heatmap H from the 2D pose estimator, we choose the location of the highest value in the map as a vertex in the graph for each joint, and the adjacency matrix is formed by the following equation:

$$\mathbb{A}_{i,j} = \begin{cases} \max(H_i)\exp(-order(i,j)) & (i \neq j) \\ \max(H_i) & (i = j) \end{cases}, \qquad (1)$$

where the $A_{i,j}$ is the outward weight from vertex i to vertex j. $max(H_i)$ stands for the confidence of the i_{th} joint. $order(i,j)$ is the minimal number of hops that is required to reach vertex j from vertex i. This formation of adjacency imposes more weight for close vertices and less for far ones. More details please refer to [4].

TCN Structure Our GCN can complete the pose under occlusion or missing information, yet produces jittering results because of its lack of temporal smoothness. Previous works on the Temporal Convolutional Network (TCN) show the effectiveness of a TCN to constrain the temporal smoothness of predicted 3D poses [21, 5]. We adopt the TCN structure [21]. As shown in Fig. 1, we utilize two TCNs to estimate the person-centric 3D poses (i.e., joints) and the camera-centric root joint depths, respectively. We named the two TCNs as: Joint-TCN and Root-TCN.

The Joint-TCN takes the 3D pose sequence produced by our GCN as input, and outputs the refined person-centric 3D poses by considering the temporal information. The loss is L2 between the estimated pose P^{TCN} and its ground-truth $\tilde{P}$, formulated as:

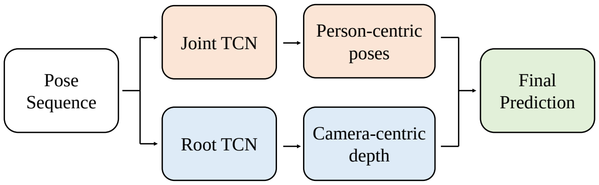

Figure 1. Pipeline of our TCNs. Our TCNs include one Joint TCN for relative pose estimation and one Root TCN for camera-centric root depth estimation.

$$L_{JTCN} = \frac{1}{K}\sum_{k=0}^{K} |P_k^{TCN} - \tilde{P}_k|^2, \qquad (2)$$

where K is the number of the joints.

The Root-TCN takes the 3D pose sequence generated by the GCN and the 2D pose sequence produced by the pose estimator as input, and outputs the estimated camera-centric root depths. Instead of directly estimating the camera-centric depth Z, we estimate the normalized root depth, which is $R^{TCN} = \frac{Z}{f}$ based on focal length f to avoid the influence of the camera intrinsic parameters. The loss function is L2 between the estimated R^{TCN} and its ground truth $\tilde{R}$:

$$L_{RTCN} = \frac{1}{K}\sum_{k=0}^{K} |R_k^{TCN} - \tilde{R}_k|^2 \qquad (3)$$

where K is the number of the joints. Based on the person-centric 3D pose from Eq. (2) and the root-joint depth from Eq. (3), the camera-centric 3D pose is obtained.

Illustration of the heatmaps estimated from the bottom-up network Fig. 2 illustrates an example output of the four heatmaps estimated by our bottom-up network. Top

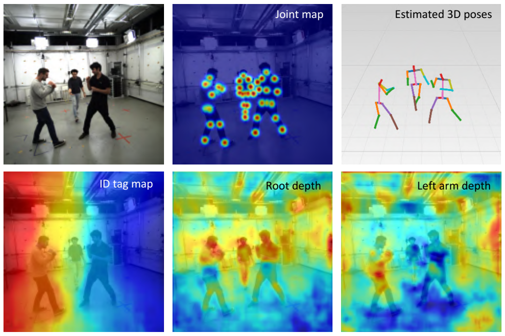

Figure 2. Visualization of estimated heatmaps from the bottom-up branch.

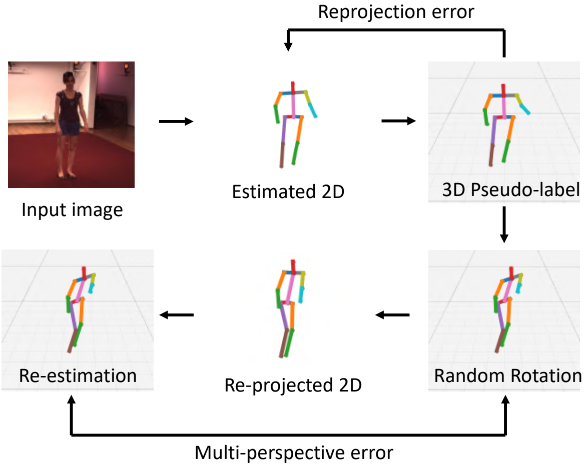

Figure 3. The illustration of our SSL pipeline. The SSL aims to keep two consistency: reprojection and multi-perspective.

left is an input image. Top middle is a joint map, which shows the heatmap of joints where all channels are merged together for better visualization of all joints. Top right is the estimated 3D poses. Bottom left shows the ID tag distribution. Bottom middle is the root depth map where the red represents a person is farther to camera than others. Bottom right is an example of relative depth map with respect to pelvis joint, where left arm depth is used as an example. The arm of left person is farther from the camera (red) compared to his pelvis while the right person's is closer to camera (blue) with respect to his pelvis.

Details of Semi-Supervised Learning Our Semi-supervised Learning (SSL) pipeline is shown in Fig. 3. First, we use the trained model to generate the pseudo-label of the unlabelled data, which is the COCO dataset in our experiment. Note that, we use only the images, and not the 2D ground-truths of the joints to mimic the unlabelled data scenario. Unfortunately, the pseudo-labels cannot be directly used because some of them are incorrect. Therefore, we use two consistency terms to measure the quality of all the pseudo-labels: the reprojection error and multi-perspective error as mentioned in the main paper.

As the pose variations of 2D datasets are more abundant than those of 3D datasets, e.g. COCO compared to H36M, the estimated 2D poses are more robust than the estimated 3D poses in terms of different environments and poses. Existing reprojection error [28] measures the deviation between generated 3D poses and detected 2D poses. Unlike this, we make use of the confidence of the joints from the 2D pose heatmap as weight in computing the reprojection error to adjust adaptively how much we should enforce the reprojected 3D poses to match the estimated 2D poses based on the confidence of the joints. Thus, the reprojection error is formulated as:

$$E_{rep} = \frac{1}{K} \sum_{k=1}^{K} C_k |rep(X_{3D,k}) - X_{2D,k}|^2 \qquad (4)$$

where the X_{3D} is the predicted 3D pose from the network, and X_{2D} stands for the 2D estimations from our multi-person 2D pose estimator. $rep(\cdot)$ is the reprojection function from 3D to 2D. K stands for the number of joints in total. Moreover, the error is a weighted sum of each joint's confidence score C_k, which is explained in Eq. (1).

We follow [3] to use a multi-perspective error as an additional measure to enforce the consistency of the predicted 3D poses from different viewing angles. Given a pseudo-label 3D pose P_{3D}^{pse}, we randomly rotate the pose along y axis (i.e., y-axis is perpendicular to the ground plane) to obtain $P_{3D}^{\prime pse}$, and re-project it to the 2D coordinates $P_{2D}^{\prime pse}$. Finally, we predict the $P_{3D}^{\prime\prime pse}$ based on the re-projection.

2. Implementation Details

Multi-Person Pose Estimator Our multi-person pose estimator uses HRNet-w32 [24] as the backbone and is trained on the combination of the MuCO and COCO dataset. We duplicate the COCO dataset twice to balance the training data between two datasets. The network is trained with the Adam optimizer with learning rate starts at 0.001 and decreases to $\frac{1}{10}$ at epoch 30 and 40. The network is trained for 50 epochs and it takes 35 hours to train on 8x RTX Quadro 8000 GPUs.

GCN and TCNs Our GCN and TCNs are trained based on the pre-extracted heatmaps from our multi-person pose estimator. We train the networks with the Adam optimizer with learning rate starts at 0.001 and decrease to $\frac{1}{10}$ every 40 epochs. The networks are trained with 100 epochs and takes 25 hours on single RTX 2080Ti GPU. We use the augmentation mentioned in [5] to train the network to better handle the occlusion.

Bottom-Up Network Our bottom-up network is trained based on the combination of the MuCO and COCO dataset. To balance the number of training samples, we duplicate the COCO dataset twice and combine with the MuCO dataset. The bottom-up network is trained with the Adam optimizer with learning rate starts at 0.001 and decrease to $\frac{1}{10}$ at the 30^{th} and 40^{th} epoch. The network is trained for 50 epochs and it takes 65 hours on 8x RTX Quadro 8000 GPUs.

Integration Network Our integration network contains 5 fully connected layers with layer size 512. The network is trained with the Adam optimizer with learning rate 0.001 in beginning, and decreased to $\frac{1}{10}$ every 50 epochs. The network is trained for 150 epochs and takes 3.5 hours on single RTX 2080Ti GPU. The data augmentation procedure is discussed in the main paper. We briefly explain here for clarity: 1) We use random masking to simulate the occlusion, where the occluded joints are masked to $(0, 0)$. 2) We apply a random shifting of joints based on a Gaussian random to simulate the inaccurate pose estimation. 3) We randomly make one of the poses in the pair to be zero, to simulate the unpaired poses.

3. Datasets Description

MuPoTS-3D [18] is a 3D multi-person testing set that consists of >8000 frames of 5 indoor and 15 outdoor scenes, and its corresponding training set is augmented from 3DHP, called MuCo-3DHP. The ground-truth 3D pose of each person in a video is obtained from multi-view markerless motion capture system, which is suitable for evaluating 3D multi-person pose estimation performance in both person-centric and camera-centric coordinates. Following [20], the training set (MuCo-3DHP) is used for training our bottom-up network, and MuPoTS-3D is used only for performance evaluation.

JTA [9] is a synthesized dataset from Grand Theft Auto V (GTA-V) game scene including various of illumination, viewpoints, and occlusion. It is a multi-person dataset with at most 32 persons appear in one frame. In addition, the images also demonstrate large person size variation as the crowd spread from close to camara and far from camera in various scenes. Because of these reasons, even it is a synthetic dataset, we'd like to perform evaluation on it. The dataset contains 512 videos, in which there are 256, 128, 128 for training, validation and testing, respectively. We follow the work [8] to estimate the F1 score under different distance threshold as a performance evaluation metric.

Human3.6M [11] is widely used for 3D human pose estimation. The dataset contains 3.6 million single-person images where an actor performs different activities in mocap studio at each video clip, so it is suitable for evaluation of 3D single-person pose estimation. Human3.6M is used for evaluating person-centric pose estimation performance. Following previous works [10, 21, 28], the subject 1,5,6,7,8 are used for training, and 9 and 11 for testing.

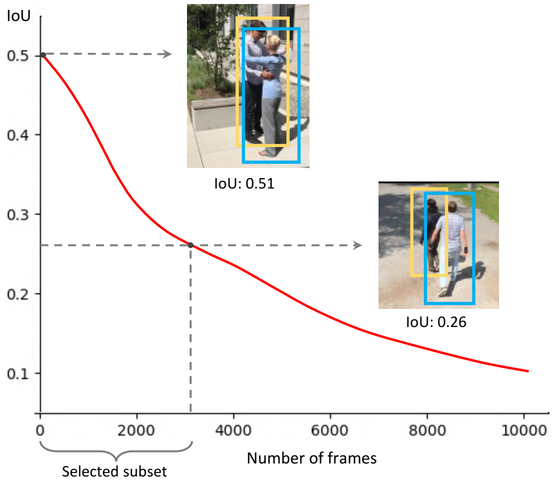

Figure 4. Interaction IoUs of 3DPW test set.

3DPW [27] is an outdoor multi-person video dataset for 3D human pose reconstruction. In each video, one target person wearing inertial measurement units (IMUs) performs daily activities outdoor, so 3D ground-truth is available for the target person only. Following previous methods [12, 25], we use 3DPW for testing without any finetuning. The ground-truth of 3DPW is SMPL 3D mesh model [17], where the definition of joints differs from what is used in 3D human pose estimation (skeleton-based) like Human3.6M, so 3DPW is rarely used in the evaluation of skeleton-based methods [26].

Evaluation errors on 3DPW cannot objectively reflect the performance of the skeleton-based methods, due to different definitions of joints. We select the top 3000 frames with the largest IoU between the target person (i.e., the person with 3D ground-truth label) and other persons based on detection out of 3DPW test set to create an inter-person occlusion subset, and then perform evaluation on it. The IoU statistics of the 3DPW test set is shown in Fig. 4, and the threshold at 3000^{th} frame is 0.26. Some samples of different occlusion level is shown in Fig. 5.

In fact, the error on this subset is still not a good performance indicator, the performance change of a method between the full testing set and this subset can measure how well the method can handle the inter-person occlusion problem. As shown in Table 6 in the main paper, our method shows the smallest error increase among all the existing methods, which demonstrates that our method is indeed capable of handling inter-person occlusion more effectively.

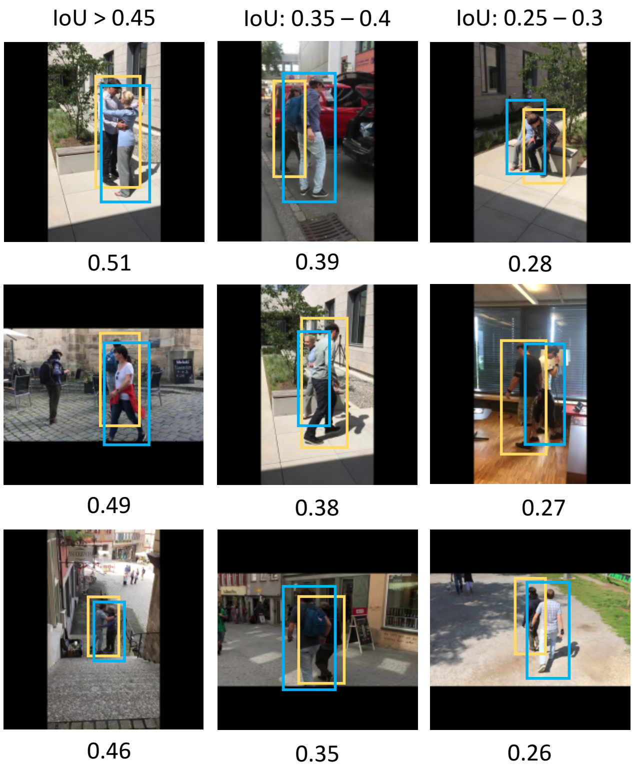

Figure 5. Some sample images of different IoU level that are selected for inter-person occlusion subset. IoU values are added below each image.

Training Datasets Both the 2D datasets and 3D datasets are used to train our networks. In the following, we explain the details of the used datasets in the training processes of our pose estimator, top-down and bottom-up networks, pose discriminator, and semi-supervised learning.

- 2D datasets for pose estimator training: We use both COCO and MuCO for training the multi-person pose estimator. Because the MuCO dataset is a synthesized dataset, solely training on the MuCO dataset will result in overfitting problem and produces unstable predictions on natural or wild images. Therefore, COCO is included for enhance the generalization ability of the network.

- 3D dataset for top-down network training: We use MuCO and its original 3DHP dataset to train the GCN and TCNs in the top-down network. MuCO and 3DHP are used for the GCN on single frame pose refinement, while the 3DHP is used to train the TCN that incorporates the temporal information. Since the network works on the x, y, z coordinates, no overfitting problem was observed from the trained models.

- 3D dataset for bottom-up network training: We use both MuCO and COCO to train the bottom-up network. We additionally include COCO, which is used only for training joint heatmaps and ID tag maps.

- 3D dataset for pose discriminator training: MuCO is used for training the integration net and pose discriminator. In addition, we do random translation and rotation of the poses to generate more synthesized interaction pairs.

- Additional 2D data for semi-supervised learning training: We use COCO for the unlabeled image dataset in training our semi-supervised learning.

Evaluation Protocols While we include the discussion of the datasets for the tables in the main paper, here we provide the details for the sake of clarity. Our model is trained with the datasets explained in the previous section (i.e., Training Datasets Used) for ablation study in Table 1 and 2, evaluations in Table 3 (MuPoTS-3D), Table 5 (Human3.6M), and Table 6 (3DPW).

The JTA dataset is captured from computer game, which has a domain gap to the real-world images. To perform the evaluation on the JTA dataset in Table 4 (JTA), we use the JTA training set to re-train the whole pipeline and perform the evaluation on the JTA test set.

As mentioned in the 3DPW dataset explanation, we follow the literature [12, 25] and only perform testing on 3DPW. Note that, the SOTA methods [14, 13] both use additional 2D and 3D datasets in training their networks. We do not use the 3DPW dataset to train our network, but used it to train the joint adaptation network [26], which transfers our predicted 3D poses of MuCO joint's definition to that of 3DPW defined on the SMPL model [17].

4. Detailed Experimental Results

As our method focuses on the 3D multi-person scenarios, our network is trained on the 3D multi-person datasets as discussed in section 3. To have a fair comparison against existing methods that are trained only with the single-person Human3.6M dataset, we re-trained the whole pipeline from scratch on H3.6M dataset following the training protocols [10, 21]. The evaluation result on the person-centric 3D human pose estimation is shown in Table 1. Similar to Table 5 in the main paper, our method achieves comparable performance against the SOTA top-down or bottom-up 3D multi-person pose estimation methods [20, 29, 15] on this single-person dataset.

Following [20], we also calculate our method's accuracy using the MPRE metrics, which measures the camera-centric 3D human pose estimation performance. In particular, [20] is 120.0, ours is 86.5, which shows 27.9% error reduction. HDNet [16] reports a better value on MPRE as 77.6, however, their method can only handle single-person

Method	MPJPE	PA-MPJPE
Moon et al., [20]	54.4	35.2
Zhen et al., [29]	54.1	n/a
Li et al., [15]	48.6	**30.5**
Ours	**42.1**	31.6

Table 1. Quantitative evaluation on Human3.6M for person-centric 3D human pose estimation. Best in **bold**, second best underlined.

cases, and performs poorly on multi-person cases where they value of PCK$_{abs}$ is 35.2, but ours is 48.0, which is a 36.4% improvement. Camera-centric 3D pose estimation is for multi-person scenario, only showing good result on a single-person dataset, and thus is not useful to solve the real problem in 3D multi-person pose estimation.

To have a better understanding on how our method compare with existing methods for each test sequence in MuPoTS-3D dataset, extended version of Table 3 in the main paper for each test sequence is summarized in Table 2 and 3 for the camera-centric and person-centric evaluations using PCK_{abs} and PCK metrics. We observe that our method consistently outperforms other methods in both the camera-centric and person-centric 3D multi-person pose estimation.

5. More Qualitative Results

In this section, we provide additional results compared with the SOTA 3D multi-person pose estimation methods. In the main paper, we already provided a qualitative comparison on 3DPW test set in Fig. 5, where the results of SMAP [29] is used as they released their code and we can perform testing with it.

Additional Comparison on MuPoTS-3D To compare with more methods, we provide additional results on MuPoTS-3D as RootNet [20] released their pretrained model on this dataset, so we can perform testing on MuPoTS-3D using their released model. Together with SMAP [29], we show the qualitative results of our method compared with that of the two SOTA methods RootNet (top-down) and SMAP (bottom-up) in Fig 6.

Additional Comparison on Wild Videos To further demonstrate the performance of our method compared with the SOTA 3D multi-person pose estimation method. We provide the qualitative results of our method compared with that of the SOTA bottom-up method SMAP [29] in Fig 7. The video clips are selected from MPII [1] dataset which is neither used for training or evaluation for both methods.

Additional Comparison on JTA As we reported our quantitative performance on JTA dataset in Table 4 in the main paper, we also provide the qualitative results of our method compared with that of the SOTA method reported and released their trained model on the JTA dataset [8] in Fig 8. The two video clips in Fig 8 show both inter-person occlusions and large multi-person scale variation where we observe our method can handle both challenges well and produce accurate camera-centric 3D multi-person pose estimation compared with LoCO [8].

References

[1] Mykhaylo Andriluka, Leonid Pishchulin, Peter Gehler, and Bernt Schiele. 5

[2] Yujun Cai, Liuhao Ge, Jun Liu, Jianfei Cai, Tat-Jen Cham, Junsong Yuan, and Nadia Magnenat Thalmann. Exploiting spatial-temporal relationships for 3d pose estimation via graph convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2272–2281, 2019. 1

[3] Ching-Hang Chen, Ambrish Tyagi, Amit Agrawal, Dylan Drover, Stefan Stojanov, and James M Rehg. Unsupervised 3d pose estimation with geometric self-supervision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5714–5724, 2019. 2

[4] Yu Cheng, Bo Wang, Bo Yang, and Robby T Tan. Graph and temporal convolutional networks for 3d multi-person pose estimation in monocular videos. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2021. 1

[5] Yu Cheng, Bo Yang, Bo Wang, Wending Yan, and Robby T Tan. Occlusion-aware networks for 3d human pose estimation in video. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 723–732, 2019. 1, 2

[6] Hai Ci, Chunyu Wang, Xiaoxuan Ma, and Yizhou Wang. Optimizing network structure for 3d human pose estimation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2262–2271, 2019. 1

[7] Rishabh Dabral, Anurag Mundhada, Uday Kusupati, Safeer Afaque, Abhishek Sharma, and Arjun Jain. Learning 3d human pose from structure and motion. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 668–683, 2018. 6

[8] Matteo Fabbri, Fabio Lanzi, Simone Calderara, Stefano Alletto, and Rita Cucchiara. Compressed volumetric heatmaps for multi-person 3d pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7204–7213, 2020. 3, 5, 9

[9] Matteo Fabbri, Fabio Lanzi, Simone Calderara, Andrea Palazzi, Roberto Vezzani, and Rita Cucchiara. Learning to detect and track visible and occluded body joints in a virtual world. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 430–446, 2018. 3

[10] Mir Rayat Imtiaz Hossain and James J Little. Exploiting temporal information for 3d human pose estimation. In *ECCV*, pages 69–86. Springer, 2018. 3, 4

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

Method	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12	S13	S14	S15	S16	S17	S18	S19	S20	Avg
Moon et al. [20]	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
Zhen et al. [29]	41.6	33.4	45.6	16.2	48.8	25.8	**46.5**	13.4	36.7	**73.5**	**43.6**	22.7	21.9	26.7	47.1	32.5	31.4	18.0	33.8	47.8	35.4
Ours	**69.2**	**57.1**	**49.3**	**68.9**	**55.1**	**36.1**	49.4	**33.0**	**43.5**	52.8	48.8	**36.5**	**51.2**	**37.1**	**47.3**	**52.0**	**20.3**	**43.7**	**57.5**	**50.4**	**48.0**

Table 2. PCK_{abs} on MuPoTS-3D dataset for all poses. Best in **bold**.

Method	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12	S13	S14	S15	S16	S17	S18	S19	S20	Avg
Rogez et al. [22]	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
Rogez et al. [23]	87.3	61.9	67.9	74.6	78.8	48.9	58.3	59.7	78.1	89.5	69.2	73.8	66.2	56.0	74.1	82.1	78.1	72.6	73.1	61.0	70.6
Dabral et al. [7]	85.1	67.9	73.5	76.2	74.9	52.5	65.7	63.6	56.3	77.8	76.4	70.1	65.3	51.7	69.5	87.0	82.1	80.3	78.5	70.7	71.3
Mehta et al. [18]	81.0	59.9	64.4	62.8	68.0	30.3	65.0	59.2	64.1	83.9	67.2	68.3	60.6	56.5	59.9	79.4	79.6	66.1	66.3	63.5	65.0
Mehta et al. [19]	88.4	65.1	68.2	72.5	76.2	46.2	65.8	64.1	75.1	82.4	74.1	72.4	64.4	58.8	73.7	80.4	84.3	67.2	74.3	67.8	70.4
Zhen et al. [29]	88.8	71.2	77.4	77.7	80.6	49.9	86.6	51.3	70.3	89.2	72.3	81.7	63.6	44.8	79.7	86.9	81.0	75.2	73.6	67.2	73.5
Moon et al. [20]	**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
Ours	93.4	**91.3**	**84.7**	**83.3**	**89.1**	**85.2**	**95.4**	**92.1**	89.5	**93.1**	**85.4**	**85.7**	**89.9**	**90.1**	**88.8**	**93.7**	**92.2**	**87.9**	**89.7**	**91.9**	**89.6**

Table 3. PCK on MuPoTS-3D dataset for all poses. Best in **bold**.

[12] Angjoo Kanazawa, Jason Y. Zhang, Panna Felsen, and Jitendra Malik. Learning 3d human dynamics from video. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 3, 4

[13] Muhammed Kocabas, Nikos Athanasiou, and Michael J Black. Vibe: Video inference for human body pose and shape estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5253–5263, 2020. 4

[14] Nikos Kolotouros, Georgios Pavlakos, Michael J Black, and Kostas Daniilidis. Learning to reconstruct 3d human pose and shape via model-fitting in the loop. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2252–2261, 2019. 4

[15] Jiefeng Li, Can Wang, Wentao Liu, Chen Qian, and Cewu Lu. Hmor: Hierarchical multi-person ordinal relations for monocular multi-person 3d pose estimation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020. 4, 5

[16] Jiahao Lin and Gim Hee Lee. Hdnet: Human depth estimation for multi-person camera-space localization. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020. 4

[17] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. Smpl: A skinned multi-person linear model. *ACM transactions on graphics (TOG)*, 34(6):248, 2015. 3, 4

[18] Dushyant Mehta, Oleksandr Sotnychenko, Franziska Mueller, Weipeng Xu, Srinath Sridhar, Gerard Pons-Moll, and Christian Theobalt. Single-shot multi-person 3d pose estimation from monocular rgb. In *2018 International Conference on 3D Vision (3DV)*, pages 120–130. IEEE, 2018. 3, 6

[19] 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):44, 2017. 6

[20] Gyeongsik Moon, Juyong Chang, and Kyoung Mu Lee. Camera distance-aware top-down approach for 3d multi-person pose estimation from a single rgb image. In *The IEEE Conference on International Conference on Computer Vision (ICCV)*, 2019. 3, 4, 5, 6, 7

[21] Dario Pavllo, Christoph Feichtenhofer, David Grangier, and Michael Auli. 3d human pose estimation in video with tem-

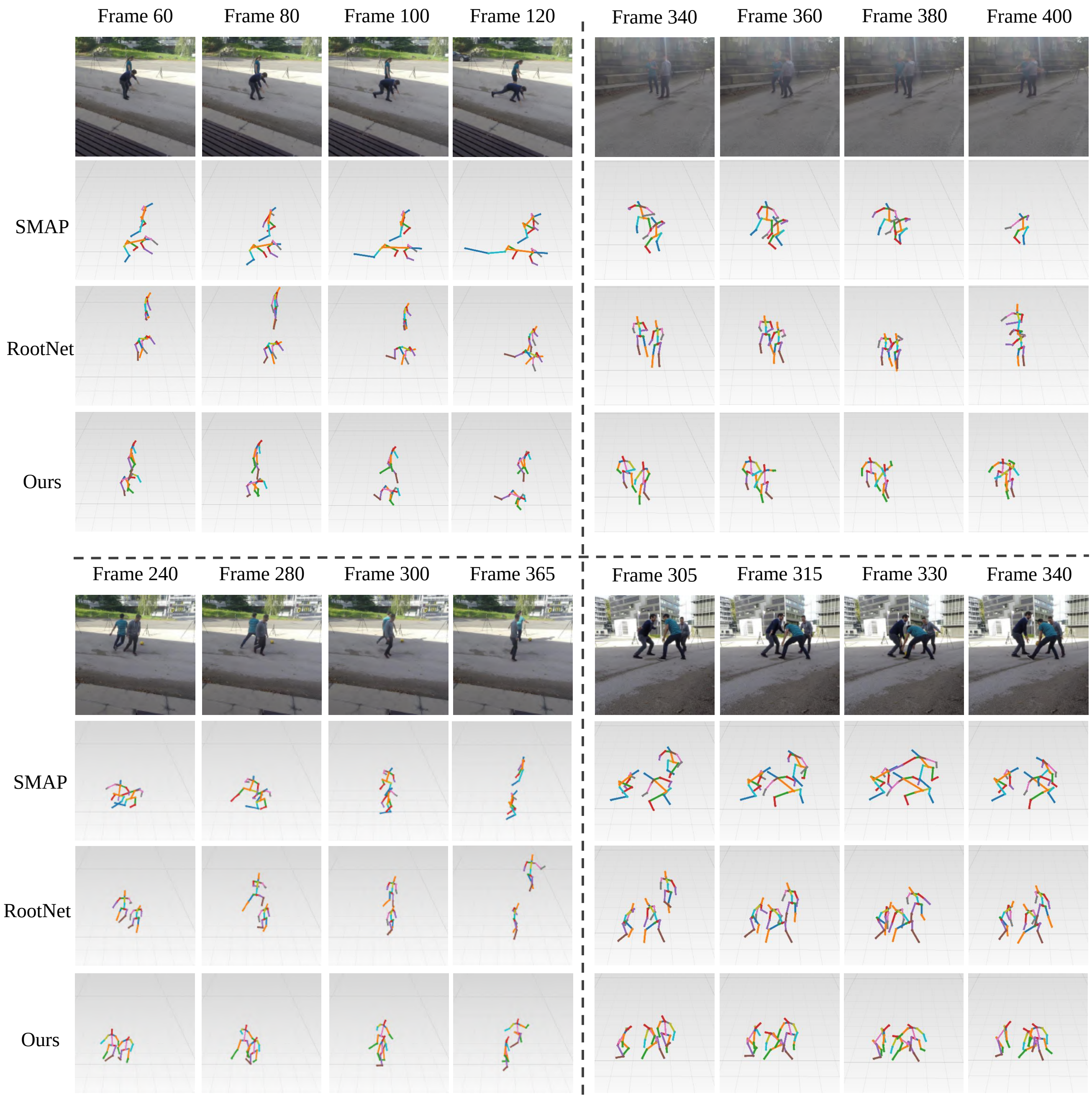

Figure 6. Results of our method compared with that of SMAP [29] (i.e., the SOTA bottom-up method) and RootNet [20] (i.e., the SOTA top-down method) on MuPoTS dataset. Results from four video clips are included: top-left, top-right, bottom-left, and bottom-right. For each video clip, the first row is the frames from the video clip; the second row is the result of SMAP; the third row is the result of RootNet; the fourth row is the result of our method. It is observed from these results that the SOTA methods suffer from inter-person occlusions while our method can handle these challenges and produce accurate camera-centric 3D multi-person pose estimation.

poral convolutions and semi-supervised training. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7753–7762, 2019. 1, 3, 4

[22] Gregory Rogez, Philippe Weinzaepfel, and Cordelia Schmid. Lcr-net: Localization-classification-regression for human pose. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3433–3441, 2017. 6

[23] Gregory Rogez, Philippe Weinzaepfel, and Cordelia Schmid. Lcr-net++: Multi-person 2d and 3d pose detection in natural images. *IEEE transactions on pattern analysis and machine*

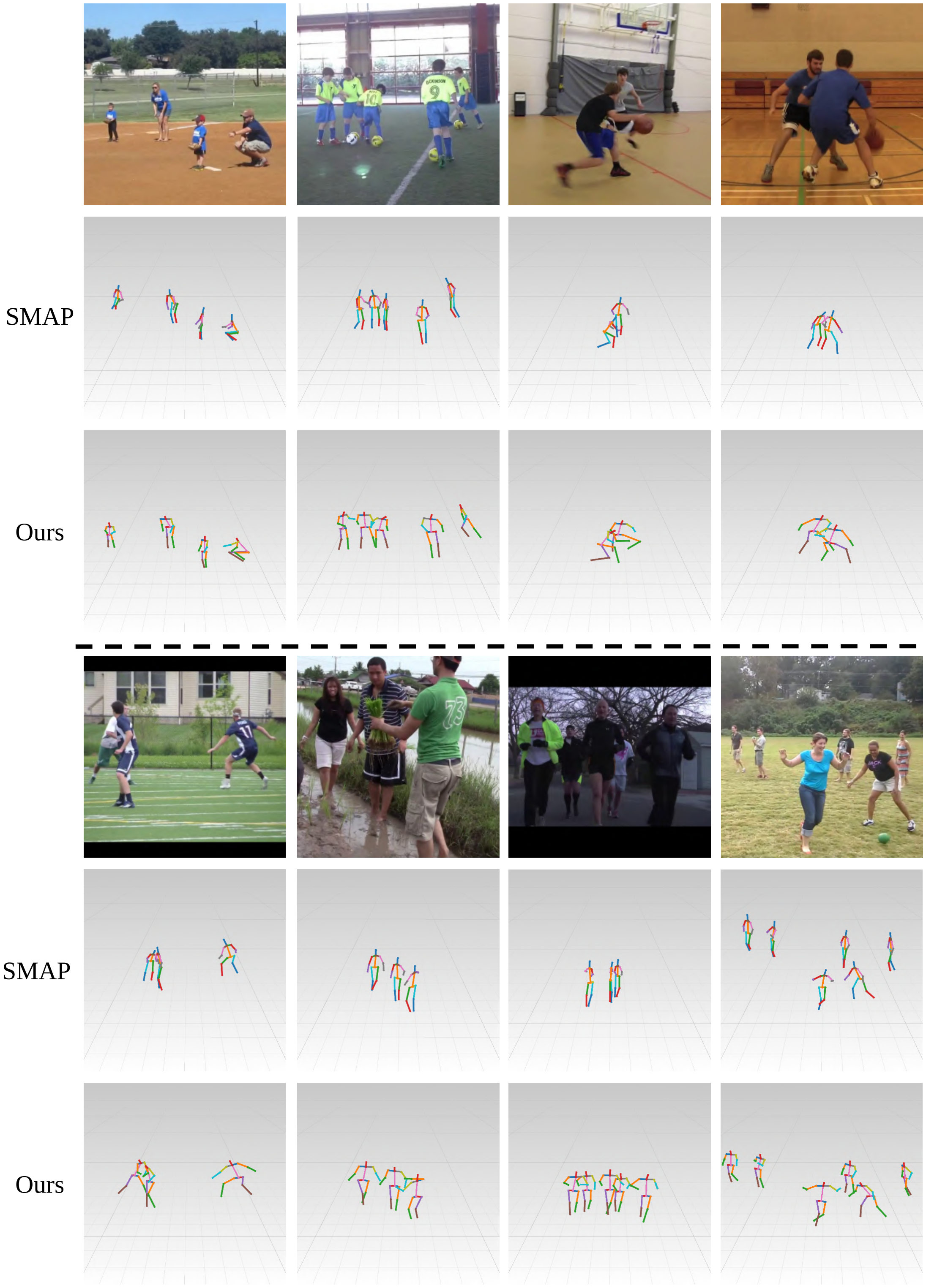

Figure 7. Results of our method compared with that of SMAP [29] (i.e., the SOTA bottom-up method) on wild videos. Results from eight video clips are included (i.e., one frame for each video). Four results are at top part of the figure, the other four are at the bottom, separated by the dashed line. For either part, the first row is the frames from the video clip; the second row is the results of SMAP; the third row is the results of our method. These results again show that the SOTA method cannot handle inter-person occlusions. In contrast, our method produces accurate camera-centric 3D multi-person pose estimation.

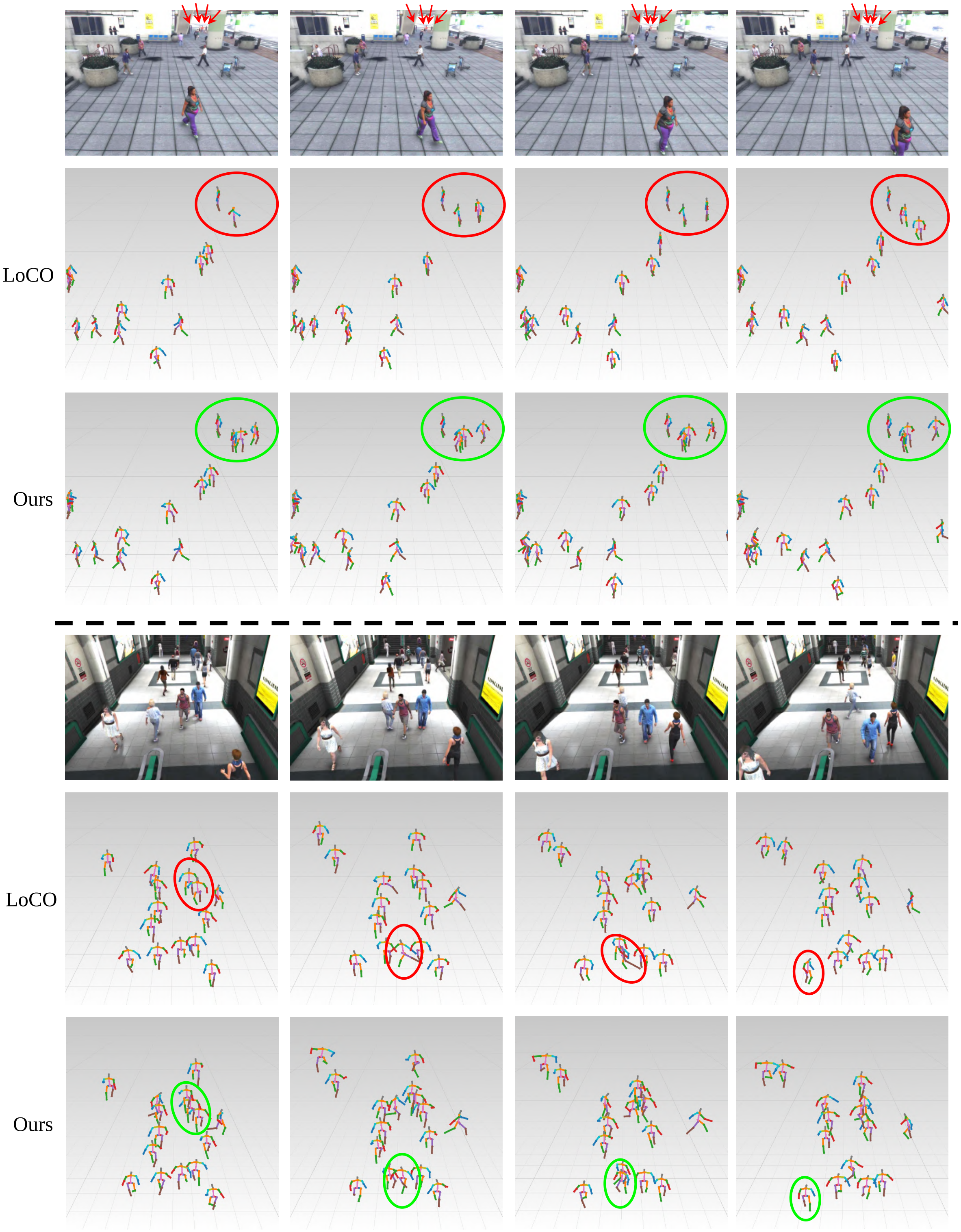

Figure 8. Result of our method compared with that of LoCO [8] (i.e., a SOTA method released trained model on JTA) on JTA dataset. Results from two video clips are included: top and bottom separated by the dashed line. For each video clip, the first row is the frames from the video clip; the second row is the result of LoCO; the third row is the result of our method. These results show that on this synthetic datasets, our method is able to produce more accurate and robust 3D multi-person pose estimation compared with other methods. We use red circle to indicate the wrong results of LoCO and green circle to point out the corresponding correct results of our method. In the first row of the top video clip, due the four persons are far from the camera which are small, we use four red arrows to indicate each of them.

intelligence, 2019. 6

[24] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for human pose estimation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 2

[25] Yu Sun, Yun Ye, Wu Liu, Wenpeng Gao, YiLi Fu, and Tao Mei. Human mesh recovery from monocular images via a skeleton-disentangled representation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5349–5358, 2019. 3, 4

[26] Shashank Tripathi, Siddhant Ranade, Ambrish Tyagi, and Amit Agrawal. Posenet3d: Unsupervised 3d human shape and pose estimation. *arXiv preprint arXiv:2003.03473*, 2020. 3, 4

[27] Timo von Marcard, Roberto Henschel, Michael J. Black, Bodo Rosenhahn, and Gerard Pons-Moll. Recovering accurate 3D human pose in the wild using IMUs and a moving camera. In *European Conference on Computer Vision (ECCV)*, pages 614–631, 2018. 3

[28] Bastian Wandt and Bodo Rosenhahn. Repnet: Weakly supervised training of an adversarial reprojection network for 3d human pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2, 3

[29] Jianan Zhen, Qi Fang, Jiaming Sun, Wentao Liu, Wei Jiang, Hujun Bao, and Xiaowei Zhou. Smap: Single-shot multi-person absolute 3d pose estimation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020. 4, 5, 6, 7, 8