POCO: 3D Pose and Shape Estimation with Confidence
Supplementary Material

In this Supplementary-Material document, we provide more details about our method. Additionally, please see the **Supplementary Video** for a summary of the method and more visualizations of the results.

1. Regressor Network Architecture

We use three variants of HPS regressors in POCO, i.e., PARE [7], HMR-EFT [4], CLIFF [11] as shown in Fig. S.1.

In PARE, the input image is first passed through a CNN backbone (HRNet-32W), and features are extracted before the average pooling layer. The features are then passed through two separate branches: a *2D Part Segmentation* branch and a *3D Body Feature* branch. The 2D part segmentation branch produces body-part attention features $S \in \mathbb{R}^{H\times W\times(J+1)}$, where $J = 24$ is the number of SMPL body parts, while a background mask is assigned to non-human pixels. The body feature branch is used to estimate SMPL body parameters. Both branches produce features of the same spatial dimensions, $H \times W$. The features from S pass through a spatial softmax normalization layer, κ. These are used as soft attention masks to aggregate 3D body features into final features, $F = \kappa(S)^\top \odot B$, where $S \in \mathbb{R}^{HW\times J}$, $B \in \mathbb{R}^{HW\times C}$ and $F \in \mathbb{R}^{J\times C}$; note that S and B are reshaped before the operation. Each feature row, $F_i \in \mathbb{R}^{1\times C}$ with $i \in \{1,\ldots,J\}$, passes through a separate MLP to get SMPL pose parameters, $\theta = \{\theta_i\}$. To estimate the camera, C, and SMPL shape, β, all final features F are fed, concatenated, to different MLPs.

HMR-EFT uses a simple network architecture for estimating HPS. The input image passes through a CNN backbone (ResNet-50) followed by a global average pooling layer. The features from the pooling layer are used to regress SMPL pose, θ, shape, β, and camera parameters, C, through separate MLPs. This regression is done through an iterative error feedback loop.

CLIFF uses a HRNet-w48 network architecture as a CNN backbone. Along with the a cropped image, CLIFF takes the bounding box location information (Bbox Info) as input to provide the location information of the person in the image. This helps to accurately predict the global rotation in the original camera coordinate frame. The bounding box formation contains the center of bounding box center relative to image center and focal length of the original camera which is calculated using image height and width. Contrary to PARE and HMR-EFT, CLIFF computes a 2D keypoint loss after projecting the body keypoints onto the original image plane.

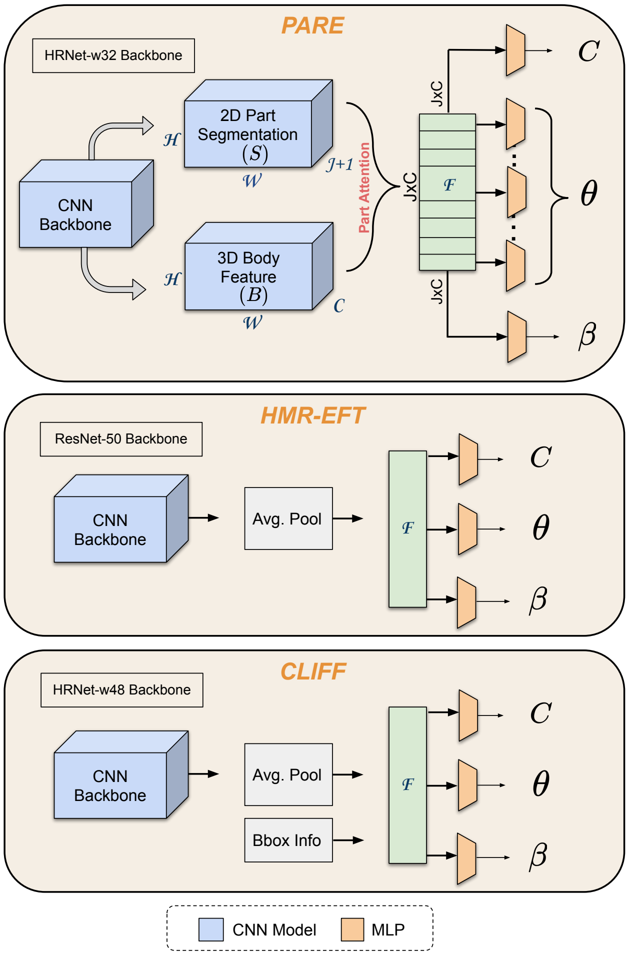

Figure S.1. **Regressor architecture.**

2. CLIFF Training and Evaluation

Since the CLIFF training code is not public, we reimplement it (“*CLIFF-Ours*”). We train CLIFF-Ours on

Method	HMR-EFT [17]		PARE [26]		CLIFF [37]	
	PVE ↓	PCC ↑	PVE ↓	PCC ↑	PVE ↓	PCC ↑
Baseline HPS	106.1	-	97.9	-	85.8	-
Gauss [22]	105.7	0.31	97.1	0.32	85.4	0.29
NFlow [35]	104.9	0.42	96.6	0.44	85.3	0.40
POCO-HPS	**101.1**	**0.52**	**95.3**	**0.54**	**84.6**	**0.51**

Table S.1. Evaluation of POCO and other uncertainty formulations for different HPS regressors.

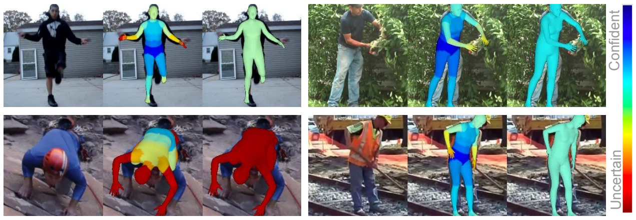

Figure S.2. **Per-part uncertainty of POCO-CLIFF.** For each image triplet: input image, per-part uncertainty of POCO and whole-body uncertainty of POCO.

COCO [12], MPII [1], MPI-INF-3D [13], H3.6M [3] and 3DPW [17] with the same dataset ratios used in HMR-EFT [4]. For 2D datasets, we use the pseudo ground-truth SMPL parameters provided by EFT [4] and, for other datasets, we use the the original annotations provided by the respective datasets. Following prior work [4, 7], we resize the cropped image to 224 × 224 for both training and evaluation. To compute the 2D keypoint loss on the full image, first we crop the keypoints according to the person bounding box and then project them back to the original image size. To evaluate on 3DPW test, we use the same bounding box scale and center used by prior work [7, 8].

3. Performance of Uncertainty Formulations for different HPS regressors

We compare the performance of our uncertainty formulation (i.e., our Dual Conditioning Strategy) with existing uncertainty formulations [6, 10] on 3DPW [17] for different HPS regressors [5, 7, 11] as shown in Tab. S.1. This complements Tab. 3 in the main paper. Our uncertainty formulation outperforms the prior formulations (*Gauss* and *NFlow*) for all HPS regressors in both the pose (PVE) and uncertainty (PCC) metrics. Note that the PCC metric should not be compared across different HPS methods on its own. Focusing *separately* on each HPS method, the important thing is that our novel uncertainty formulation consistently lowers PVE errors while increasing the PCC metric; this is indicative of a better uncertainty formulation.

Method	Train-Params	Test-Params	Inference Time
CLIFF	81.0 M	81.0 M	1.45 ms
POCO-CLIFF	82.6 M	81.3 M	1.49 ms

Table S.2. POCO's overhead when applied to HPS regressor.

4. Per-Part and Per-Vertex Uncertainty

POCO models the uncertainty of SMPL pose parameters in the following way. First, it estimates the uncertainty for the axis-angle rotation of each of SMPL's skeleton joints separately. This is important because each of these has a different amount of error. However, for downstream applications, having a single uncertainty value for the full body is more practical. To this end, we traverse SMPL's kinematic chain (i.e., recursively going in the direction from parent to child), and add the axis-angle uncertainties of the respective skeleton joints; as there are 24 joints in total, this produces a 24D vector. We then normalize the 24D vector to the range of $[0, 1]$ and compute the mean to get a single scalar uncertainty value; this represents the uncertainty for the full body. The per-part uncertainties and the full-body uncertainty are shown in Fig. S.2.

A few recent methods [14, 15] show per-vertex uncertainties. They do so by sampling multiple bodies and computes their per-vertex variance as a measure of uncertainty. While this is an interesting choice, modelling per-vertex uncertainties in a single feed-forward pass would be expensive. One would need to model the *base density function* and *scale network* to output 6890 (SMPL vertices) as compared to only 24 variables (SMPL joints) in the case of POCO.

5. Overhead of POCO Framework

POCO is a general uncertainty framework that can be applied to common HPS methods, extending them to also estimate uncertainty. It adds a *bDF* and *scale network* for estimating uncertainty in a single network pass. Tab. S.2 shows that POCO imposes only a small overhead. POCO-CLIFF has only 2% more training parameters than CLIFF due to adding the bDF and scale network. The former is unused at test time and the latter is just a small NN, so, adding POCO increases inference time only minimally.

6. Self-Improved HPS Training

POCO estimates an uncertainty measure that correlates with pose reconstruction quality. We use this measure to automatically curate SMPL estimates from the Charades dataset [16] and improve POCO, using the following steps.

We first sample every 100th frame from the videos to get a total of 130K images, and apply POCO on these. We then vary POCO's uncertainty threshold, and for each value we automatically curate the produced SMPL estimates and

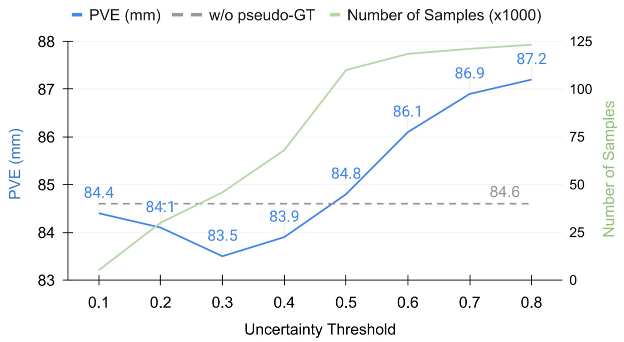

Figure S.3. **Uncertainty threshold for pseudo-GT selection.** Using an optimal uncertainty threshold of 0.3 for selecting pseudo-GT from Charades dataset [16] for training, POCO-CLIFF-pGT's performance on 3DPW test set is better than the model trained without it (dashed line). The performance degrades with higher uncertainty threshold. PVE denotes per-vertex error. The green line is for number of samples and axis is on the right.

Method	PVE ↓	MPJPE ↓	PA-MPJPE ↓	Filter pGT	# pGT
POCO-CLIFF	84.6	70.9	43.3	-	-
POCO-CLIFF-Whole	87.2	74.7	46.1	✗	130K
POCO-CLIFF-Rand	86.6	73.9	45.6	✗	46K
POCO-CLIFF-pGT	**83.5**	**69.7**	**42.8**	✓	46K

Table S.3. **Effect of uncertainty-filtered pGT data on 3DPW.** "Whole" trains with all data [16] without filtering, "Rand" with random samples, and "pGT" filters using POCO uncertainty.

extend POCO's training data. This results in multiple extended training datasets. We finetune POCO separately for each of these, and evaluate each finetuned model on 3DPW.

The evaluation results are shown in Fig. S.3. The dashed gray line shows POCO-CLIFF (with no additional pseudo-GT). The blue curve shows the PVE error (mm) of the finetuned variants. The green curve shows the number of curated samples for each threshold. With a low threshold (0.1) very few samples pass, thus, performance is almost unchanged. With a high threshold (≥ 0.45), as the threshold gets higher, more samples of decreasing quality pass, which can even harm performance. For thresholds in the range of $[0.2, 0.4]$ enough good-quality samples pass so that performance improves. The best performance is achieved for a threshold of 0.3, which results in adding roughly 46k samples in the training dataset; given the limited number of subjects and pose variation compared to the original dataset, the performance shows that this bootstrapping is promising. Note that the threshold is determined on the Charades dataset by visual inspection. 3DPW test data is not used in setting the threshold. Random samples of pseudo ground-truth generated by POCO-CLIFF on Charades dataset is shown in Fig. S.5.

To better understand the degree of self-improvement, we perform two additional baseline experiments. POCO-CLIFF-pGT uses 46K frames (out of 130K) from the Charades dataset, filtered using our uncertainty estimates. For comparison, we re-train POCO-CLIFF: (1) using all 130K frames *("Whole")*, and (2) using 46K frames randomly sampled from the 130K *("Rand")*. All methods use POCO-CLIFF SMPL estimates as pGT. Tab. S.3 shows that adding data without confidence filtering makes results worse, while our self-improvement process improves them.

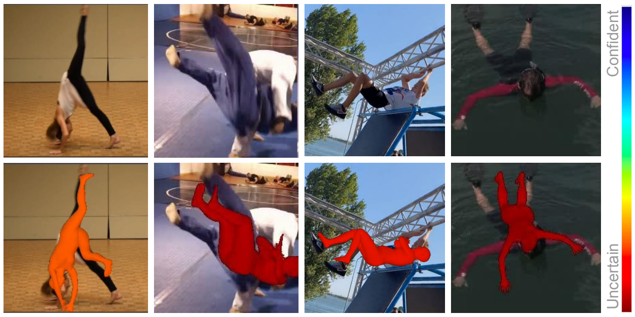

Figure S.4. **Failure cases of POCO.** In some cases of occlusion and out-of-distribution poses, POCO estimates high uncertainty even though the pose reconstructions are not totally implausible.

7. Details on infilling with uncertainty

For the second downstream task detailed in Sec. 5.4, we use POCO's uncertainty estimates to automatically detect and remove the uncertain pose estimates from a video sequence. Subsequently, we apply GLAMR [18] to inpaint the 3D bodies for frames with uncertain pose estimates. However, GLAMR has certain limitations and we use heuristics to avoid these. Specifically, we exclusively consider video sequences with a confidence level exceeding 0.3 for both the initial and final 5 frames; that is, infilling requires reliable pose estimates for the starting and ending video parts. Additionally, we exclude video sequences in which more than 15 consecutive frames exhibit an uncertainty exceeding 0.7, otherwise GLAMR's infiller is significantly challenged.

8. Failure Cases

In Fig. S.4, we show some representative cases in which POCO's prediction quality and its uncertainty estimate disagree. Typically, POCO produces more plausible poses than other HPS methods [2, 7], even for complex scenarios of heavy occlusion and out-of-distribution poses. However, sometimes POCO estimates high uncertainty, even if the poses it produces are reasonable; think of this a "false negative". In Fig. S.4 each image either contains an unusual pose, motion blur, occlusion, or dim lighting – in some cases more than one of these. It is reasonable for the network to be uncertain of its estimates in these cases, even if it happens to get the pose right (or close).

9. Effect of Occlusion on Uncertainty

POCO estimates 3D body parameters and their uncertainty in a single feed-forward pass. The uncertainty is correlated to image ambiguities and the quality of reconstruction. We analyze the correlation qualitatively on 3DPW for the POCO-HMR-EFT network. Specifically, we add a synthetic occluder that we swipe throughout the video frames to see the effect on uncertainty; see Fig. S.6. We observe that uncertainty increases when a body part is occluded.

10. Qualitative Results

We qualitatively compare POCO with the deterministic HPS methods like CLIFF [11], PARE [7], and the probabilistic methods ProHMR [9] and Sengupta et al. [14]. The results are shown in Fig. S.7 and Fig. S.8, respectively. Please see the *video* on our website for more examples.

References

[1] Mykhaylo Andriluka, Leonid Pishchulin, Peter Gehler, and Bernt Schiele. 2d human pose estimation: New benchmark and state of the art analysis. In *Computer Vision and Pattern Recognition (CVPR)*, 2014. 2

[2] Georgios Georgakis, Ren Li, Srikrishna Karanam, Terrence Chen, Jana Košecká, and Ziyan Wu. Hierarchical kinematic human mesh recovery. In *European Conference on Computer Vision (ECCV)*, 2020. 3, 7

[3] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3.6M: Large scale datasets and predictive methods for 3D human sensing in natural environments. *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2014. 2

[4] Hanbyul Joo, Natalia Neverova, and Andrea Vedaldi. Exemplar fine-tuning for 3D human pose fitting towards in-the-wild 3D human pose estimation. In *International Conference on 3D Vision (3DV)*, 2020. 1, 2

[5] Angjoo Kanazawa, Michael J. Black, David W. Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *Computer Vision and Pattern Recognition (CVPR)*, 2018. 2

[6] Alex Kendall and Yarin Gal. What uncertainties do we need in bayesian deep learning for computer vision? In *Conference on Neural Information Processing Systems (NeurIPS)*, 2017. 2

[7] Muhammed Kocabas, Chun-Hao P. Huang, Otmar Hilliges, and Michael J. Black. PARE: Part attention regressor for 3D human body estimation. In *International Conference on Computer Vision (ICCV)*, 2021. 1, 2, 3, 4, 6

[8] 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 *International Conference on Computer Vision (ICCV)*, 2019. 2

[9] Nikos Kolotouros, Georgios Pavlakos, Dinesh Jayaraman, and Kostas Daniilidis. Probabilistic modeling for human mesh recovery. In *International Conference on Computer Vision (ICCV)*, 2021. 4, 7

[10] Jiefeng Li, Siyuan Bian, Ailing Zeng, Can Wang, Bo Pang, Wentao Liu, and Cewu Lu. Human pose regression with residual log-likelihood estimation. In *International Conference on Computer Vision (ICCV)*, 2021. 2

[11] Zhihao Li, Jianzhuang Liu, Zhensong Zhang, Songcen Xu, and Youliang Yan. CLIFF: Carrying location information in full frames into human pose and shape estimation. In *European Conference on Computer Vision (ECCV)*, 2022. 1, 2, 4, 6

[12] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. In *European Conference on Computer Vision (ECCV)*, 2014. 2

[13] Dushyant Mehta, Helge Rhodin, Dan Casas, Pascal V. Fua, Oleksandr Sotnychenko, Weipeng Xu, and Christian Theobalt. Monocular 3d human pose estimation in the wild using improved cnn supervision. In *International Conference on 3D Vision (3DV)*, 2017. 2

[14] Akash Sengupta, Ignas Budvytis, and Roberto Cipolla. Hierarchical Kinematic Probability Distributions for 3D Human Shape and Pose Estimation from Images in the Wild. In *International Conference on Computer Vision (ICCV)*, 2021. 2, 4

[15] Akash Sengupta, Ignas Budvytis, and Roberto Cipolla. HuManiFlow: Ancestor-Conditioned Normalising Flows on SO(3) Manifolds for Human Pose and Shape Distribution Estimation. In *Computer Vision and Pattern Recognition (CVPR)*, 2023. 2

[16] Gunnar A. Sigurdsson, Gül Varol, Xiaolong Wang, Ali Farhadi, Ivan Laptev, and Abhinav Gupta. Hollywood in homes: Crowdsourcing data collection for activity understanding. In *European Conference on Computer Vision (ECCV)*, 2016. 2, 3, 5

[17] 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)*, 2018. 2

[18] Ye Yuan, Umar Iqbal, Pavlo Molchanov, Kris Kitani, and Jan Kautz. GLAMR: Global occlusion-aware human mesh recovery with dynamic cameras. In *Computer Vision and Pattern Recognition (CVPR)*, 2022. 3

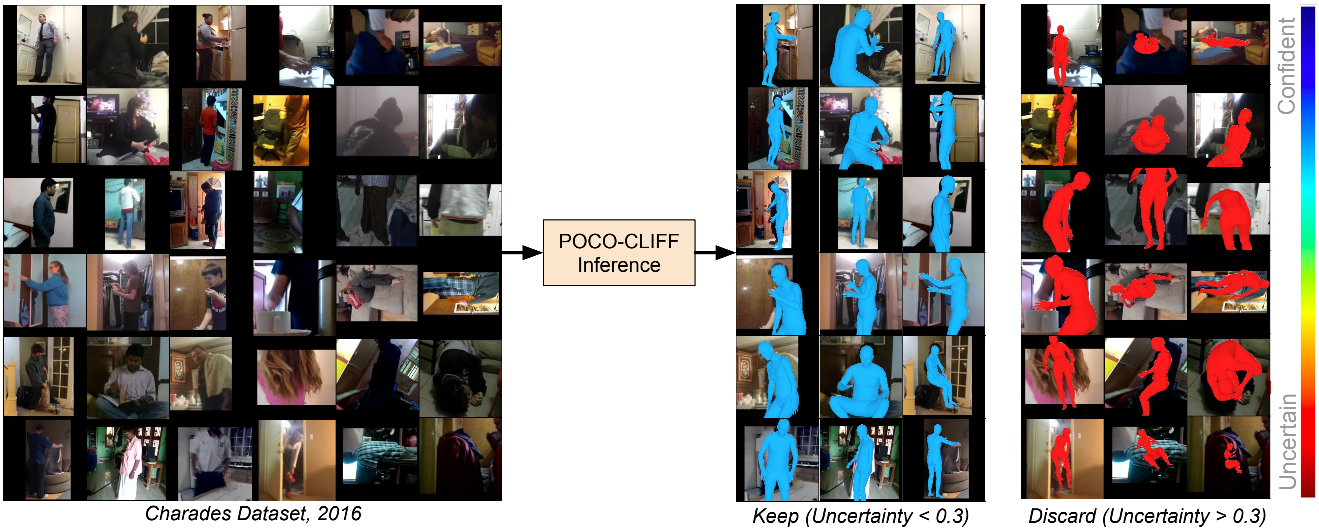

Figure S.5. **Automatic pseudo-GT.** Random samples of pseudo-GT generated by POCO-CLIFF on Charades [16] dataset. We keep the frames with lower uncertainty and treat the output SMPL parameters as pseudo ground-truth for re-training.

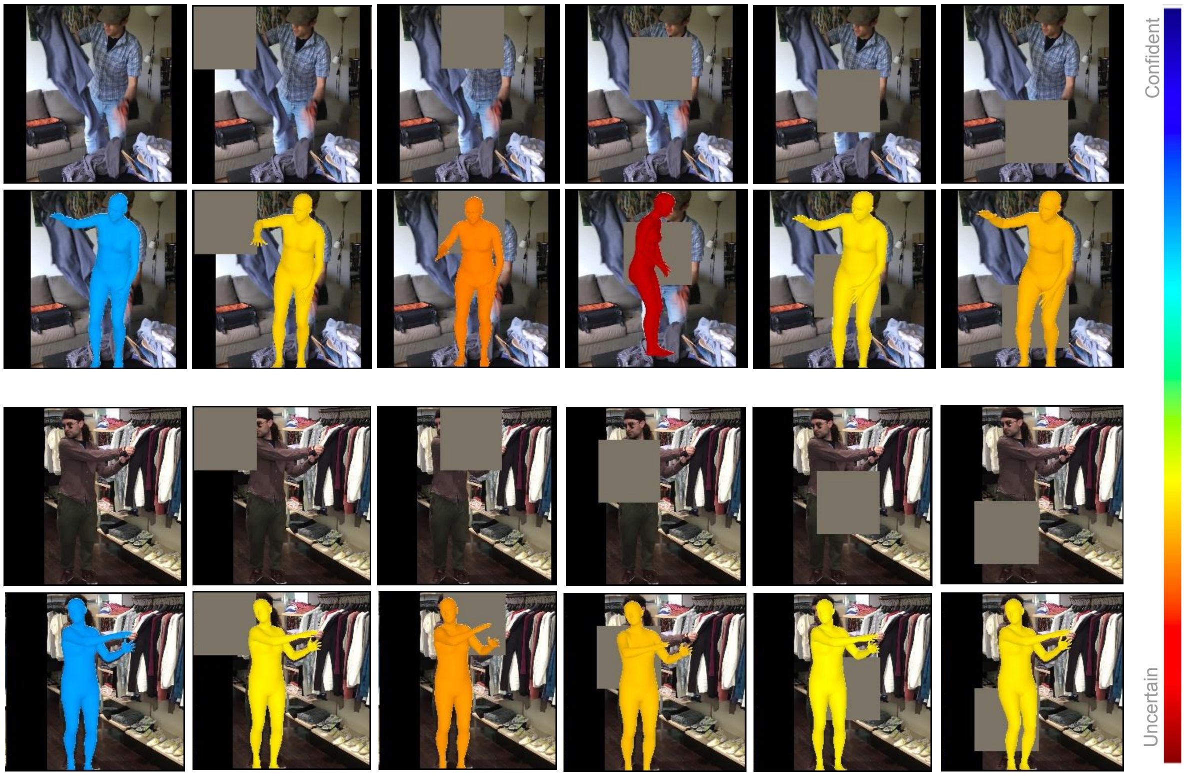

Figure S.6. **Effect of occlusion on uncertainty.** When an image becomes ambiguous due to a synthetic occluder, POCO-HMR-EFT estimates a higher uncertainty.

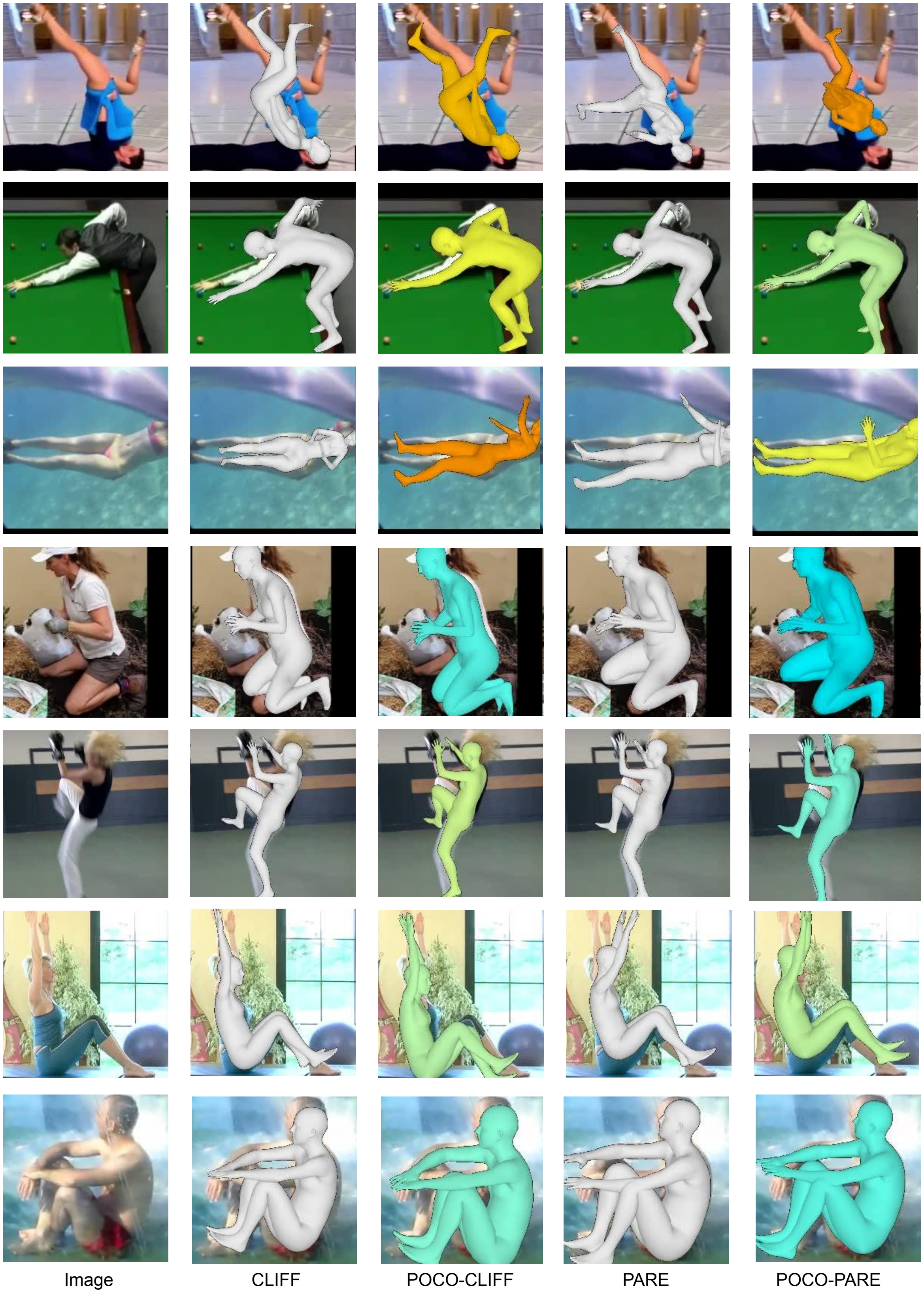

Figure S.7. **Qualitative evaluation for in-the-wild images.** We show results for CLIFF [11], PARE [7] and POCO versions of respective HPS methods, i.e., POCO-CLIFF and POCO-PARE.

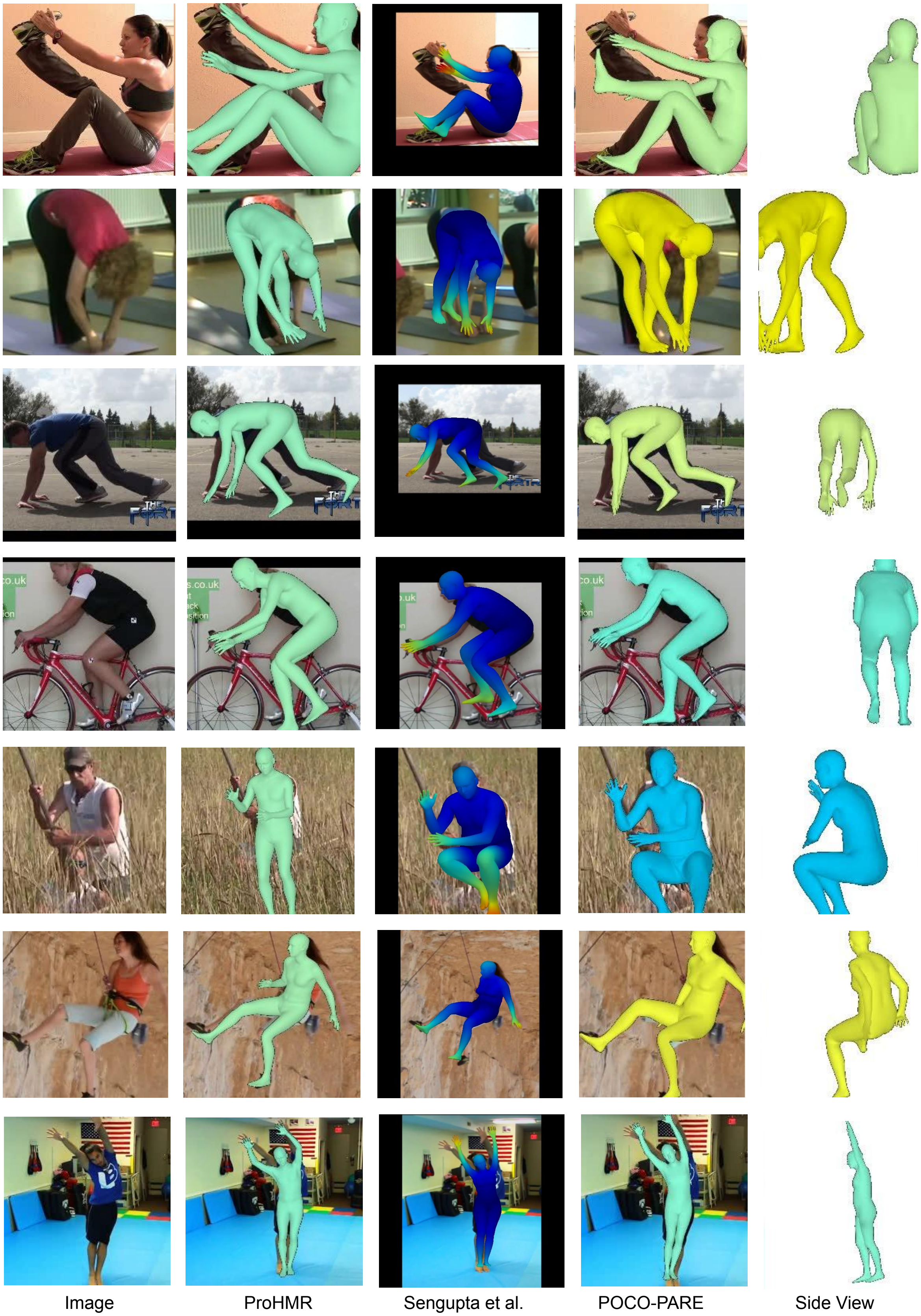

Figure S.8. **Qualitative evaluation for in-the-wild images.** We show results for ProHMR [9], Sengupta et al. [2], and our POCO-PARE.