Real-time Full Body Capture with Inter-part Correlations – Supplemental Document –

Yuxiao Zhou[1] Marc Habermann[2,3] Ikhsanul Habibie[2,3] Ayush Tewari[2,3] Christian Theobalt[2,3] Feng Xu[1*]
[1]BNRist and School of Software, Tsinghua University [2]Max Planck Institute for Informatics [3]Saarland Informatics Campus

In the following, we present more evaluation of our approach in Sec. 1 and Sec. 2, and explain technical details in Sec. 3.

1. Additional Qualitative Results

In Fig. 1, we present more qualitative results on in-the-wild videos. To process the image sequence, we first use the off-the-shell human detector [8] to obtain the body bounding box of the *first frame*. After that, for each frame, its body bounding box is updated according to the 2D keypoint estimation of the previous frame. In this way, our method tracks the subject and performs 3D capture fully automatically. As a frame-based approach, our method inevitably suffers from the temporal jittering, which is also shared by the previous work of Choutas et al. [2]. We adopt a basic temporal filter [1] for smooth visualization.

Further, we compare our results with the state-of-the-art approaches of Choutas et al. [2] and Xiang et al. [10] in Fig. 2, where we present results of equal visual quality but much faster inference speed. We present failure cases in Fig. 3. In the first row, our method cannot handle the hand-hand interaction very well. This is because distinguishing the two hands from monocular color input is a very challenging task, and such samples are rare in our training data. In the second row, our approach does not estimate the face color and the hand pose very well due to the unseen appearance: the face is occluded by the goggles, while the hands are under the gloves.

Finally, to illustrate the discrepancy in keypoint definitions of different datasets, we present the result of our model on the same image under different sets of dataset-specific *extended keypoints* in Fig. 4. The positions for the hips, shoulders, and neck are quite different, while the elbows, ankles, knees are always consistent across datasets.

Please refer to our supplementary video for more results.

*This work was supported by the National Key R&D Program of China 2018YFA0704000, the NSFC (No.61822111, 61727808), Beijing Natural Science Foundation (JQ19015), and the ERC Consolidator Grant 4DRepLy (770784). Feng Xu is the corresponding author.

Training \ Testing	MTC	HM36M	MPII3D	HUMBI
MTC	**0.89**	0.18	0.12	0.25
HM36M	0.05	0.81	0.05	0.10
MPII3D	0.31	0.23	0.50	0.33
HUMBI	0.78	0.39	0.34	**0.99**
Ours	0.84	**0.95**	**0.87**	**0.99**

Table 1. Cross-dataset evaluation. The models are exclusively trained on one dataset and evaluated on others, except for *Ours*. The metric is PCK at 150mm after Procrustes alignment.

2. Additional Quantitative Results

Evaluation of multiple datasets. In *DetNet* for body keypoint detection, we combine multiple datasets for superior generalization. To further evaluate the help of these datasets, for each dataset, we exclusively train a model on its train split, and evaluate it on all the datasets. To cope with the inconsistency in keypoint definitions of different datasets, we only compute the error of the *basic keypoints*, which are shared by all the datasets without ambiguity (see Sec. 3 for more details). The metric is percentage of correct keypoints (PCK) [6] with an error threshold of 150mm after Procrustes alignment. A larger PCK means a higher percent of the predicted keypoints are considered correct as their errors are smaller than the threshold. As shown in Tab. 1, all these datasets do not generalizes well on other datasets. On the other hand, by combining all the datasets, our approach performs similarly well or even better than the dataset-specific models on each dataset.

Evaluation of *FaceNet* on the EHF dataset. To further examine the quality of face capture and the generalization ability, we evaluate our *FaceNet* on the EHF[7] dataset. Note that the model has not seen this dataset during training. Due to the difference in mesh topologies, we cannot compute the vertex-to-vertex error as in [7]. Instead, we report the 2D landmark error and photometric error after projection as in the main manuscript, and compare with the work of Tewari et al. [9] in Tab. 2.

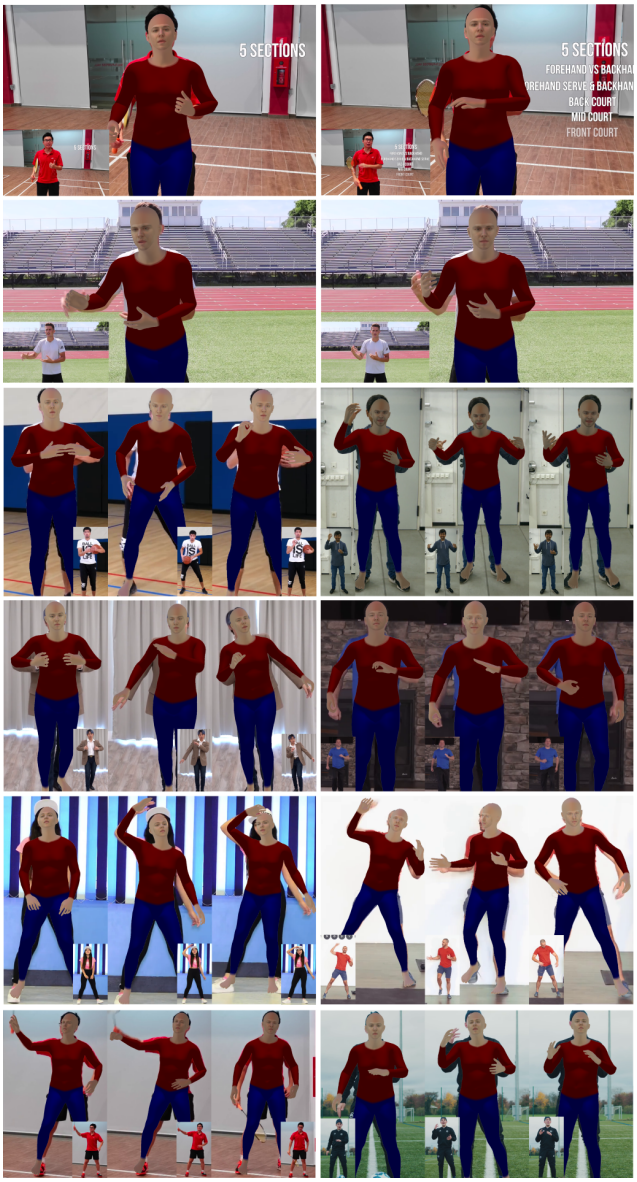

Figure 1. Qualitative results on in-the-wild videos. Given the bounding box of the first frame, our method captures the subject fully automatically in real-time.

Method	Landmark Error	Photometric Error
Tewari et al. [9]	5.81	0.2482
FaceNet	5.64	0.0468

Table 2. Evaluation of *FaceNet* on the EHF dataset.

3. Technical Details

PoseNet. We use an atomic module, *PoseNet*, for all keypoint detection tasks in *DetNet*. The network structure is illustrated in Fig. 5. *PoseNet* first estimates keypoint-maps K from the input features F. For keypoint i, its keypoint-map K_i is a one-channel map where the value at each pixel

Figure 2. Comparison with previous works. Despite much faster, our approach has equal visual quality as Choutas et al. [2] and Xiang et al. [10].

Figure 3. Failure cases. Top: our approach cannot handle hand-hand interaction very well. Bottom: our method fails to estimate the face color and the hand pose very well due to the unseen appearance: the subject is wearing goggles and gloves.

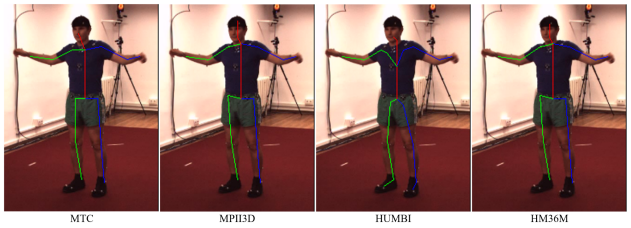

Figure 4. Visualization of different keypoint definitions on the same image.

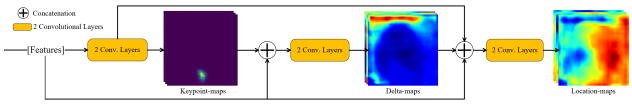

Figure 5. Our *PoseNet* network structure.

represents the confidence that this pixel is occupied by keypoint i. *PoseNet* then estimates delta-maps D from the concatenation of the input features F and the keypoint-maps K. For bone i, its delta-map D_i is a three-channel map that embeds the x, y and z component of the vector representing the direction of this bone. D serves as intermediate supervision during training. The ground truth delta-maps are obtained by tiling the unit vector of the bone to the size of K, but we do not force the network prediction to be unit vectors during training. Finally, F, K and D are concatenated and passed through two convolutional layers to obtain the location-maps L. For each keypoint i, its location-map L_i comprises three channels that store the estimated x, y and z coordinates relative to the pre-defined root keypoint. At reference, for keypoint i, its 2D coordinates (u_i, v_i) are determined as the location of K_i's maximum, and its 3D coordinates are retrieved from its location-map L_i at the position of (u_i, v_i). For body pose estimation, we directly use meter as the estimation unit, while for hands we use the length of the bone from the wrist joint to the middle finger root joint to normalize the coordinates. The estimated 3D positions are relative to a pre-defined root keypoint: for body, it is pelvis; for hand, it is the root of the middle finger.

Keypoint Definitions. To compensate for the issue that different datasets have different keypoint definitions, and the same keypoints are annotated differently in different datasets, we split the body keypoints into two subsets: universal *basic body keypoints*, and dataset-specific *extended body keypoints*. The *basic body keypoints* are listed as following: pelvis, left/right elbow, left/right wrist, left/right knee, and left/right ankle.

The input keypoints of *BodyIKNet* are defined on the HUMBI dataset. To provide sufficient information for IK, we manually select $N_{\text{humbi}} = 33$ keypoints as the *DetNet* prediction, which comprises 22 joints predefined in SMPL (the hand joints in SMPL are excluded), and 11 additional keypoints to reduce ambiguity, which are: head top, left/right eye, left/right hand index finger root joint, left/right hand little finger root joint, left/right foot big toe root joint, and left/right foot little toe joint. As HUMBI already provides ground truth in the form of SMPL mesh, we manually bind these keypoints to the corresponding vertices. We view HUMBI [11] and SPIN [4] as the same dataset as they all provide SMPL [5] meshes and thus have exactly the same keypoint definitions.

Evaluation on HUMBI. As mentioned above, we define $N_{\text{humbi}} = 33$ keypoints on the HUMBI dataset. For the quantitative evaluation on HUMBI, to be consistent with other datasets that usually contain around 15 keypoints, we select a subset of 14 keypoints for error computation, listed as following: pelvis, left/right hip, left/right knee, left/right ankle, lower neck, left/right shoulder, left/right elbow, and left/right wrist. HUMBI contains sequences of 772 subjects, from where we take 54 subjects for test and leave the remaining for training. The subjects for test are #1 to #60, excluding the following 6 subjects: 5, 6, 16, 22, 26, 43. They are excluded because the corresponding data provided by HUMBI is corrupted and we could not parse them. As HUMBI is a multi-view dataset, it is very often that the body is only partially seen in the image. Therefore, the keypoints outside the image are ignored during the computation of the error.

Training Details. During the training of *DetNet*, the composition of each mini-batch is: 2 samples from body+hands dataset (MTC); 2 samples from each 3D body dataset (HUMBI/MPII3D/HM36M/SPIN); 1 sample from each 2D body dataset (MPII2D/COCO); 1 sample from each hand dataset (FreiHand/STB/CMU-Hand). We set $w_k = 1$, $w_d = 10$, and $w_l = 10$ for the keypoint detection loss $\mathcal{L}_p$ of *PosNet* in Eq. 7 of the main paper. For the full loss of *DetNet* in Eq. 11 of the main paper, we set $\lambda_b = 10$, $\lambda_h = 1$, and $\lambda_f = 1$. The Adam optimizer [3] is used for training *DetNet*. The learning rate starts from 0.001, and is set to 0.0001 when the loss stops decreasing on the validation set. For *BodyIKNet* we use the following hyperparameters in Eq. 15 of the main paper: $\lambda_\alpha = 0.1$, $\lambda_\beta = 0.001$, $\lambda_\theta = 1.0$, $\lambda_\chi = 20.0$, and $\lambda_{\bar{\chi}} = 0.1$. For *HandIKNet* we use the following hyperparameters in Eq. 15 of the main paper: $\lambda_\alpha = 0.1$, $\lambda_\beta = 0.001$, $\lambda_\theta = 10.0$, $\lambda_\chi = 20.0$, and $\lambda_{\bar{\chi}} = 0.1$. We use the Adam optimizer to train both *BodyIKNet* and *HandIKNet* with a fixed learning rate of 0.0001.

References

[1] Géry Casiez, Nicolas Roussel, and Daniel Vogel. 1€ filter: a simple speed-based low-pass filter for noisy input in interactive systems. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 2527–2530, 2012.

[2] Vasileios Choutas, Georgios Pavlakos, Timo Bolkart, Dimitrios Tzionas, and Michael J. Black. Monocular expres-

sive body regression through body-driven attention. *arXiv preprint arXiv:2008.09062*, 2020.

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

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

[5] 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):1–16, 2015.

[6] 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. *international conference on computer graphics and interactive techniques*, 36(4):1–14, 2017.

[7] Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed A. Osman, Dimitrios Tzionas, and Michael J. Black. Expressive body capture: 3d hands, face, and body from a single image. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10975–10985, 2019.

[8] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.

[9] Ayush Tewari, Michael Zollhofer, Hyeongwoo Kim, Pablo Garrido, Florian Bernard, Patrick Perez, and Christian Theobalt. Mofa: Model-based deep convolutional face autoencoder for unsupervised monocular reconstruction. In *2017 IEEE International Conference on Computer Vision Workshops (ICCVW)*, pages 1274–1283, 2017.

[10] Donglai Xiang, Hanbyul Joo, and Yaser Sheikh. Monocular total capture: Posing face, body, and hands in the wild. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10965–10974, 2019.

[11] Zhixuan Yu, Jae Shin Yoon, In Kyu Lee, Prashanth Venkatesh, Jaesik Park, Jihun Yu, and Hyun Soo Park. Humbi: A large multiview dataset of human body expressions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2990–3000, 2020.