Supplementary Material for Monocular Real-Time Volumetric Performance Capture

1 Implementation Detail

1.1 Datasets

RenderPeople Similar to [18], we leverage high-quality photogrammetry scans of clothed humans with synthetic rendering to construct our training dataset. Aside from 466 static scans from RenderPeople [16] used in [18], we incorporate additional 167 rigged models from RenderPeople [16] and apply 32 animation sets from Mixamo [14] so that wider pose variations are covered for performance capture. Please refer to appendix A for the complete list of animations. By randomly selecting 3 frames from each animation, we obtain $466 + 167 \times 32 \times 3 = 16,498$ models. We split this into *training* and *validation* sets based on subject identities, resulting in $452 + 164 \times 32 \times 3 = 16,196$ meshes in the *training* set and $14 + 3 \times 32 \times 3 = 302$ meshes in the *validation* set. For training, each mesh is rendered with weak perspective camera at every 10 degrees around the yaw axis using Precomputed Radiance Transfer [20] and 163 second-order spherical harmonics derived from HDRI Haven [5]. For validation, we compute our loss metrics on the *validation* set rendered with 3 views sampled at 120-degree intervals around the yaw axis. The hyper-parameters α_{i}, β_{i}, α_{p} and β_{p} in our Online Hard Example Mining (OHEM) training strategy (see Eq. 6 in the paper) are chosen using the *validation* set.

BUFF To quantitatively evaluate the generalization ability of the proposed system and fairly compare with the existing methods, we propose to use the BUFF dataset [24] for the following reasons: First, the BUFF dataset provides high-fidelity geometry with photorealistic texture, approximating the modality of real images with detailed ground truth geometry. Secondly it contains large pose variations. Thus, accuracy of each method under various poses can be properly evaluated. Lastly, as the existing approaches [10,18,26] are trained with custom datasets, we can fairly compare on a dataset with which none of these methods are trained. The BUFF dataset consists of 5 subjects, each of which is captured with 1 or 2 unique outfits. In total, it contains 26 sequences with per-frame ground-truth 3D meshes and textures. As the large portion of poses are duplicated (e.g., T-pose), we apply K-Medoids to each sequence to obtain distinctive frames. By setting $K = 10$, we obtain $26 \times 10 = 260$ frames and render them from 3 views points at 120-degree intervals around the yaw axis, resulting in $260 \times 3 = 780$ images for *test* set (see Fig. 5 for sample images).

1.2 Network Architectures

We have made several architectural modifications to improve the efficiency and robustness of the original implementation of [18]. In this section, we provide

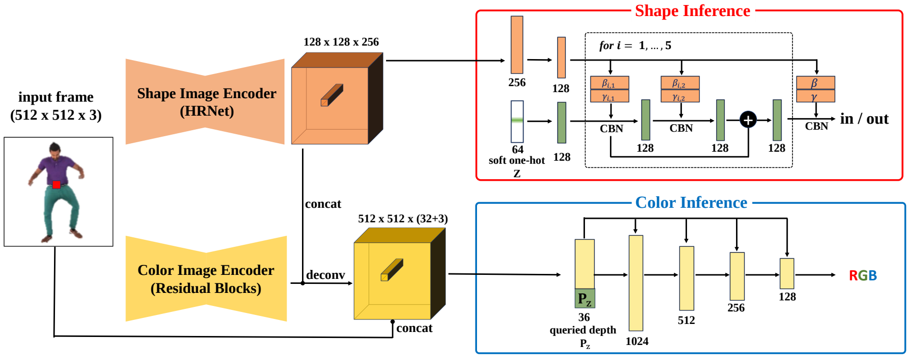

Fig. 1. The overview of our network architectures.

the implementation details as well as discussion about the effectiveness of each modification. Fig. 1 shows the overview of our network architectures.

Image Encoder For surface reconstruction, we replace the stacked hourglass network [15] with HRNetV2-W18-Small-v2 [21] in our image encoder for shape inference due to its superior performance in various tasks (e.g., semantic segmentations, human pose estimations) with faster computation (see Figure 1). The final feature resolution is 128 × 128 with the channel size of 256 as in [18]. Table 1.3 shows the ablation study on the choice of image encoders. HRNet not only shows better reconstruction accuracy but also faster runtime (14 fps vs 12 fps) with less parameters and computation. For color inference, we found that a higher spatial resolution for image features result in more detailed textures. To this end, we modify the architecture with 6 residual blocks [7] by upsampling the stacked output feature maps from shape and color image encoders from 128 × 128 to 512 × 512 with the output channel size of 32 using a transposed convolution.

Depth Representation Additionally, inspired by a multi-channel depth representation used in ordinal depth regression [3], we found that representing depth P_z as a multi-dimensional vector more effectively propagates depth information to the shape inference function f_O. More specifically, we convert $\{P_z \in \mathbb{R} | -1 \leq P_z \leq 1\}$ into a N-dimensional feature $\mathbf{Z} = \{Z_i\}_{i=0}^{N-1}$ as follows:

$$Z_i = \begin{cases} 1 + \lfloor (N-1) \cdot P'_z \rfloor - (N-1) \cdot P'_z & \text{if } i = \lfloor (N-1) \cdot P'_z \rfloor \\ (N-1) \cdot P'_z - \lfloor (N-1) \cdot P'_z \rfloor & \text{if } i = \lfloor (N-1) \cdot P'_z \rfloor + 1 \\ 0 & \text{otherwise,} \end{cases} \tag{1}$$

where $P'_z = 0.5 \cdot (P_z + 1.0)$ and $N = 64$ in our experiments. We term this multi-channel depth representation soft one-hot depth (SoftZ). Figure 2 and Table 1.3 demonstrates the faster convergence and more accurate reconstruction of the proposed depth representation.

Pixel-aligned 3D Lifting The original implementation of [18] lifts the pixel-aligned image features into 3D by feeding the image feature and the depth value P_z

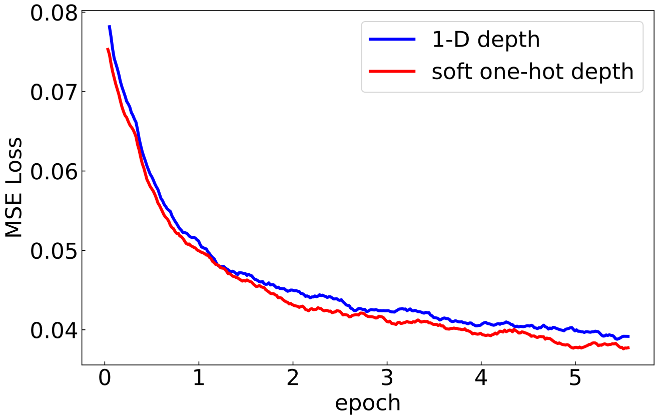

Fig. 2. Comparison on different depth representation. Representing depth z as a *soft* one-hot vector makes the network converge faster with higher accuracy. We use HRNet with CBN for both 1-D depth representation baseline and SoftZ.

into a multi-layer perceptron (MLP). To further reduce the channel size of intermediate layers, we adopt a conditional batch normalization (CBN) [1,2,13]. More specifically, the soft one-hot depth vector $\mathbf{Z}$ (our final model) or the depth value P_z (only for ablation study) is fed into a multi-layer perceptron (MLP) consisting of 5 blocks of an conditional batch normalization module (CBN) [1,2,13] where input feature vector for each CBN layer are normalized with the learnable multiplier $\gamma(c)$ and bias $\beta(c)$ taking as input a conditional vector c as follows:

$$f_{out} = \gamma(c) \frac{f_{in} - \mu}{\sqrt{\sigma^2 + \epsilon}} + \beta(c), \tag{2}$$

where f_{in} and f_{out} are the input and output features, μ is the statistical mean, σ is the standard deviation, and $\epsilon = 1.0 \times 10^{-5}$. Each layer is followed by non-linear ReLU activation. Note that unlike [13], our conditional vector c is pixel-aligned image features $\Phi(\mathbf{P}_{xy}, g_O(\mathbf{I}))$ to learn precise geometry aligned with an input image. Please refer to Figure 1 for the detailed architecture. We use the channel size of 128 for all the intermediate feature dimensions. In Tab. 1.3, CBN is referred to as 3d lifting with conditional batch normalization modules and MLP as the naive concatenation of a queried depth value and image features as in [18]. The number of parameters and computational overhead are further reduced while retaining the same level of reconstruction accuracy.

For color inference, we take as input the concatenation of the depth value P_z, RGB value from the corresponding pixel of the input image, and the learned image feature, resulting in 36 dimensional vector. They are fed into another MLP consisting of 5 layers with the channel size of 1024, 512, 256, 128, and 3 and skip connections at 1, 2, 3, and 4-th layers. Each layer is followed by the LeakyReLU activation except the last layer, and Tanh activation for the last layer.

Metric	Chamfer		P2S		Params	GFLOPS	Runtime
	RP	BUFF	RP	BUFF	(M)	(per 4096 calls)	(fps)
HG-MLP [18]	1.684	3.629	1.743	3.601	15.6	105.0+9.7	12
HRNet-MLP	1.602	3.623	1.691	3.617	8.8	16.0+9.7	14
HRNet-CBN	1.584	3.626	1.652	**3.585**	8.3	16.0+3.0	15
HRNet-CBN-SoftZ	**1.561**	**3.615**	**1.624**	3.613	8.3	16.0+3.0	15

Table 1. Ablation study.

Training procedure We use RMSProp [22] and Adam [9] for the surface reconstruction and texture inference respectively, with a learning rate of $1e-3$. Since the batch normalization layer in HRNet and CBN can benefit from large batch sizes, we use a batch size of 24 for both surface reconstruction and texture inference. The number of sampled points per image is 4096 in every training batch. We first train the surface reconstruction network for 5 epochs with the constant learning rate, then fix it and only train the texture inference network for 5 more epochs. The training of our networks for surface reconstruction and texture inference takes 3 days each on a single NVIDIA GV100 GPU.

1.3 Real-time Human Segmentation

As preprocessing, we require an efficient and accurate human segmentation network. To this end, we start by collecting high-quality data with accurate annotations. Because publicly available human segmentation datasets are either low-quality or biased to particular types of images (*e.g.*, portraits) [4,11,19,25], we collected $12,029$ human images with various backgrounds, lighting conditions, poses, and different outfits. Most of the images come from the LIP dataset [4], while the rest are collected from the internet. We obtained high-quality annotations of these images using a commercial website[1]. We use a U-Net [17] with ResNet-18 [6] as backbone with Adadelta [23] using an initial learning rate of 10.0. The learning rate is reduced by a factor of 0.95 after each epoch. The training converges after 100 epoches, which takes about 2 days on a single NVidia GV100. During inference, with 256×256 image resolution, this model run at 150 fps on NVidia GV100. Figure 6 and Figure 7 show the sampled training dataset and segmentation results of our real-time segmentation model, respectively.

2 Additional Results

We evaluate the robustness of our algorithm under different lighting conditions, viewpoints, and clothes topology in Figure 2. We also provide additional qualitative results from a video sequence (see Figure 8) and from internet photos (see Figure 9). The other video reconstruction results can be found in the supplemental video.

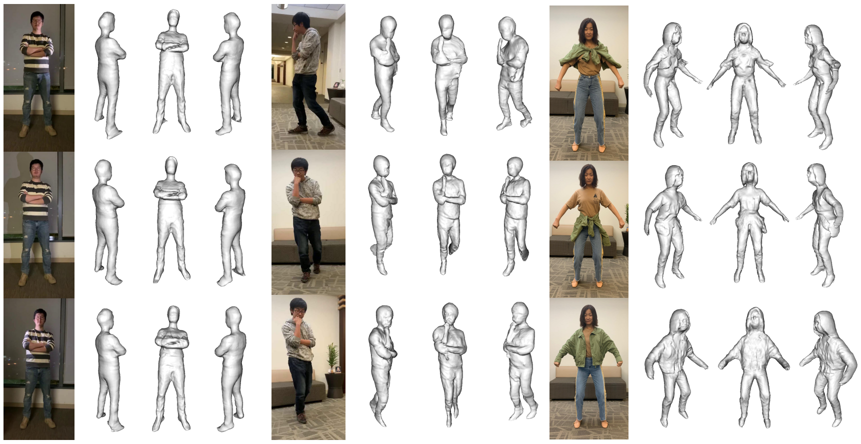

Fig. 3. We qualitatively evaluate the robustness of our approach by demonstrating the consistency of reconstruction with different lighting conditions, viewpoints and surface topology.

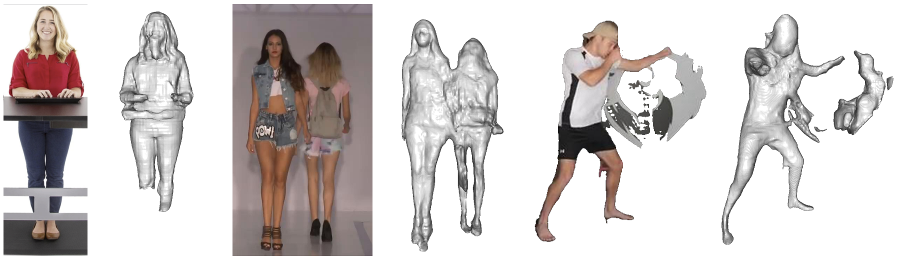

Fig. 4. Limitations.
Our current system may fail in the presence of inaccurate segmentation, multiple subjects, and severe occlusions.

2.1 Limitations

As our training data consists of only a single person at a time, the presence of multiple people confuses the network (see Figure 4). Modeling multiple subjects [8,12] is essential to understanding social interaction for a truly believable virtual experience. In the future, we plan to extend our approach to handle multiple people in a single monocular video. Another interesting direction is to handle occlusion by other objects, as a complete 3D reconstruction is difficult without explicitly modeling the occlusion occurring in natural scenes.

---

[1] https://www.remove.bg/

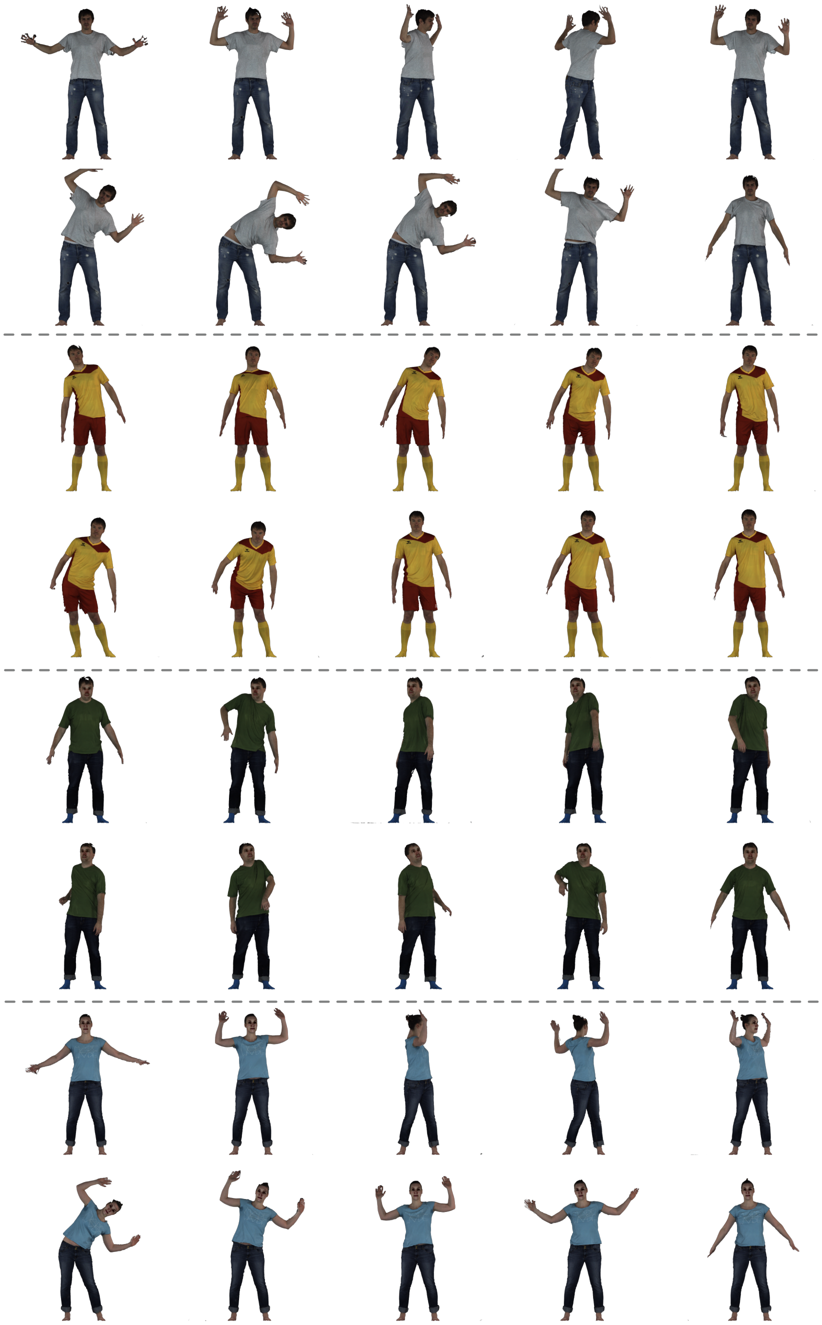

Fig. 5. Sampled BUFF benchmark. We apply K-Medoids to each sequence of BUFF dataset to construct the *test* set. Sufficient pose variations in BUFF dataset are covered with $K = 10$.

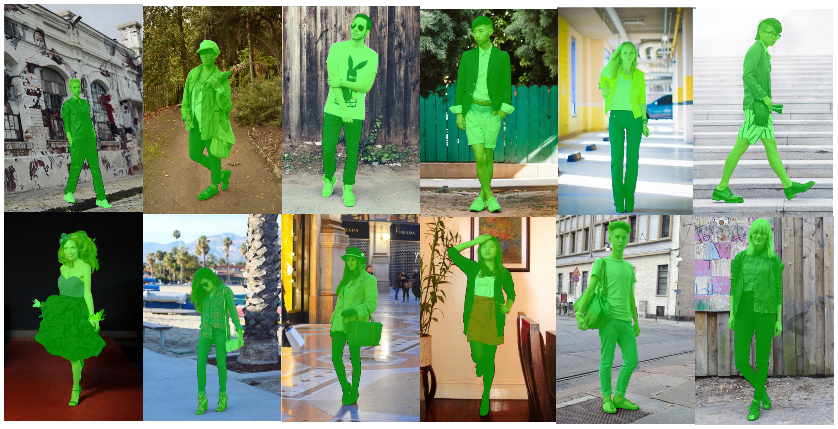

Fig. 6. Training data for our real-time segmentation network.

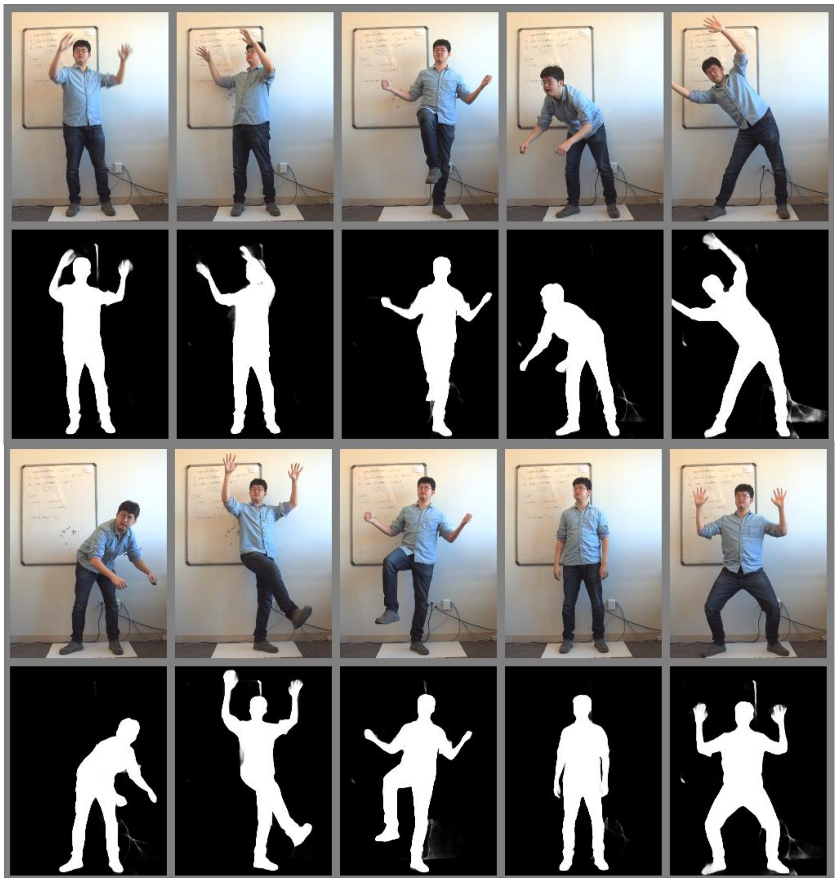

Fig. 7. Results of our segmentation network.

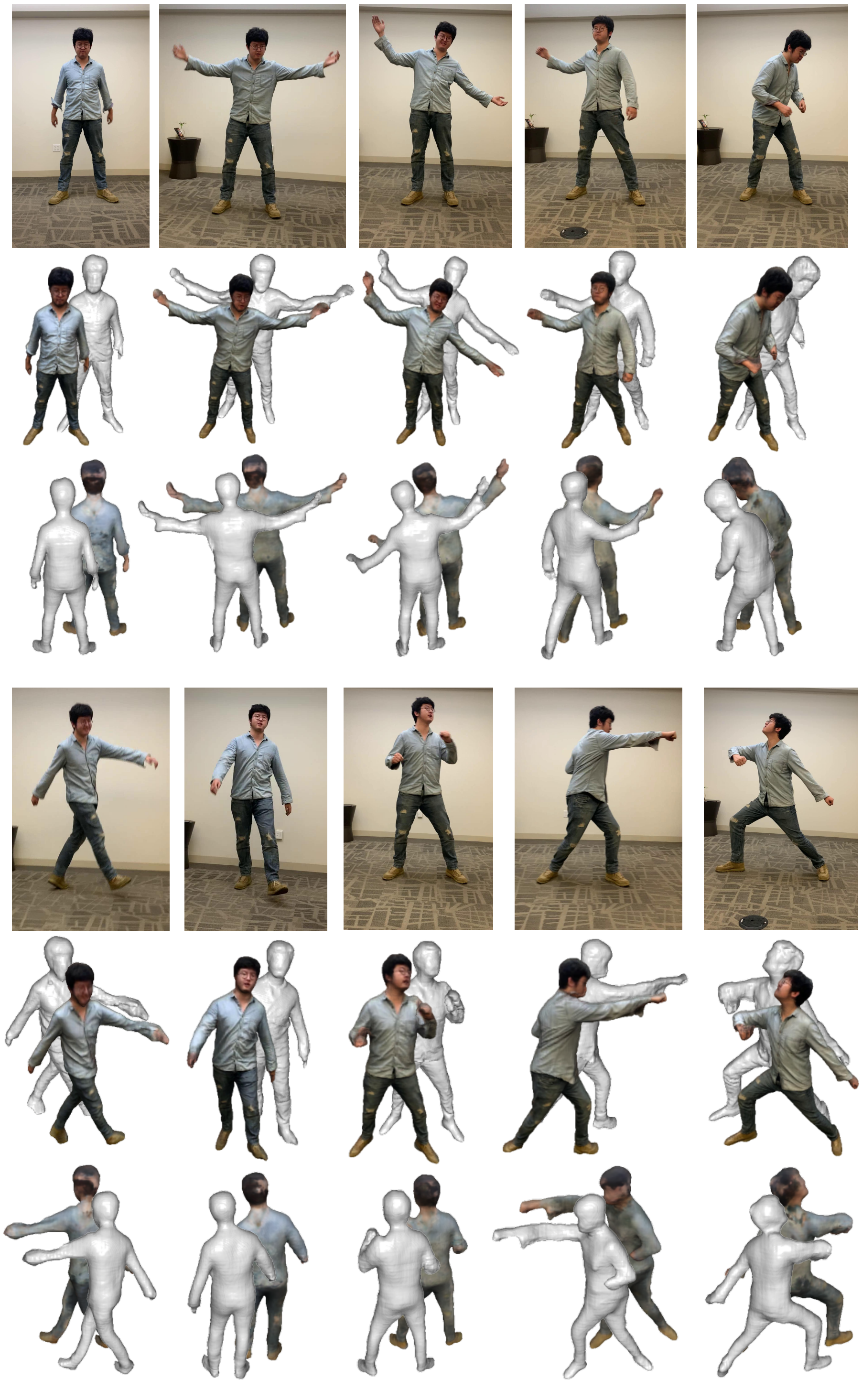

Fig. 8. Qualitative results on self-captured performances.

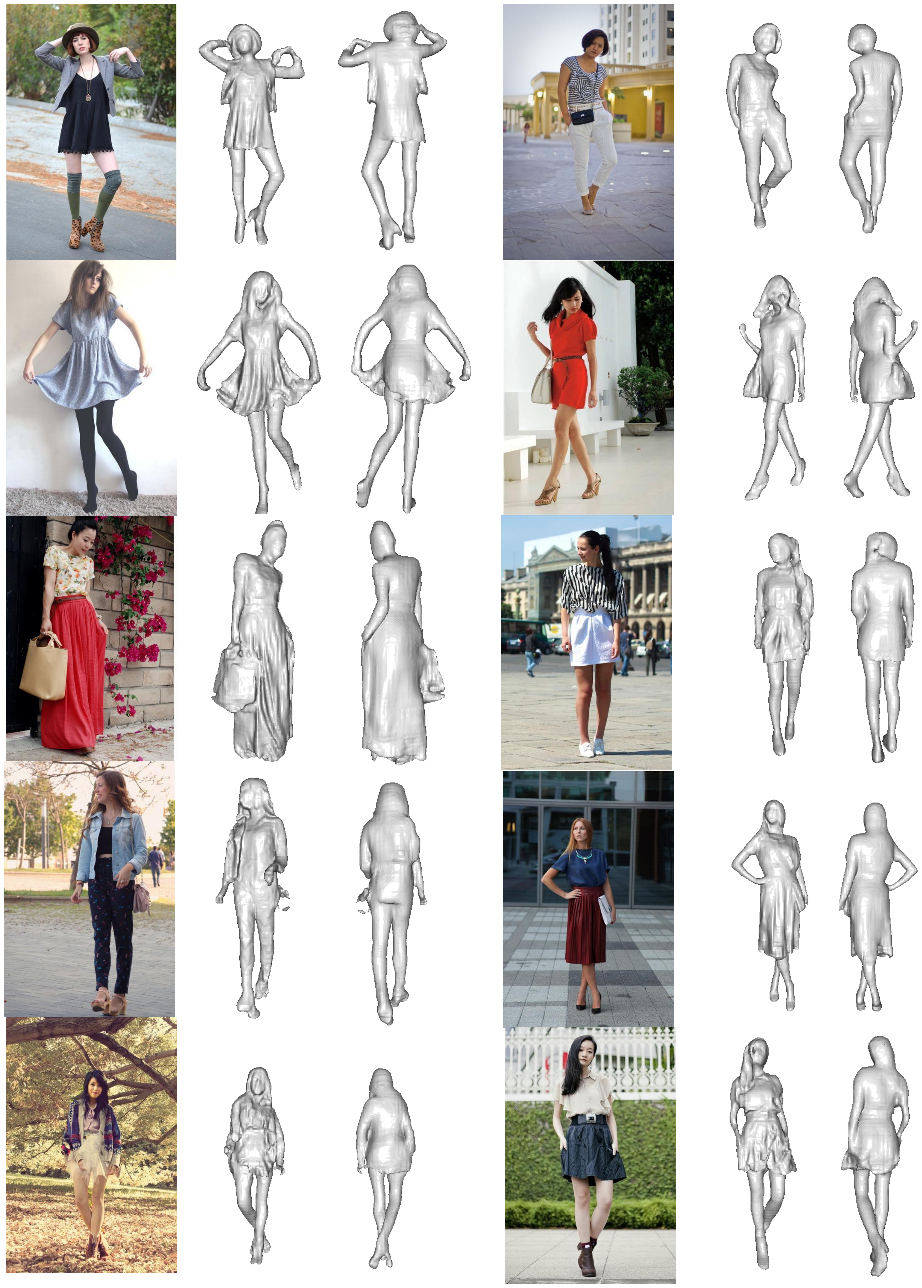

Fig. 9. Qualitative results on internet photos.

A Mixamo Animation Sets

Agreeing	Bored
Breakdance_Ready	Defeat
Defeated	Dwarf_Idle
Female_Tough_Walk	Hands_Forward_Gesture
Holding_Idle	Look_Over_Shoulder
Old_Man_Idle	Orc_Idle
Patting	Pointing
Put_Back_Rifle_Behind_Shoulder	Searching_Files_High
Shoulder_Rubbing	Standing_Clap
Standing_Greeting	Standing_Torch_Idle_02
Standing_Turn_90_Right	Standing_Turn_Left_90
Standing_W_Briefcase_Idle	Stop_Jumping_Jacks
Strut_Walking	Talking_On_Phone
Talking_Phone_Pacing	Talking
Talking_Turn_180	Walking
Yawn	Yelling

References

1. De Vries, H., Strub, F., Mary, J., Larochelle, H., Pietquin, O., Courville, A.C.: Modulating early visual processing by language. In: Advances in Neural Information Processing Systems. pp. 6594–6604 (2017)
2. Dumoulin, V., Belghazi, I., Poole, B., Mastropietro, O., Lamb, A., Arjovsky, M., Courville, A.: Adversarially learned inference. arXiv preprint arXiv:1606.00704 (2016)
3. Fu, H., Gong, M., Wang, C., Batmanghelich, K., Tao, D.: Deep ordinal regression network for monocular depth estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2002–2011 (2018)
4. Gong, K., Liang, X., Zhang, D., Shen, X., Lin, L.: Look into person: Self-supervised structure-sensitive learning and a new benchmark for human parsing. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 932–940 (2017)
5. HDRI Haven: (2018), https://hdrihaven.com/
6. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
7. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. In: European Conference on Computer Vision. pp. 694–711. Springer (2016)
8. Joo, H., Simon, T., Sheikh, Y.: Total capture: A 3d deformation model for tracking faces, hands, and bodies. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 8320–8329 (2018)
9. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
10. Kocabas, M., Athanasiou, N., Black, M.J.: Vibe: Video inference for human body pose and shape estimation. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2020)
11. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: ECCV. pp. 740–755 (2014)
12. Liu, Y., Stoll, C., Gall, J., Seidel, H.P., Theobalt, C.: Markerless motion capture of interacting characters using multi-view image segmentation. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 1249–1256 (2011)
13. Mescheder, L., Oechsle, M., Niemeyer, M., Nowozin, S., Geiger, A.: Occupancy networks: Learning 3d reconstruction in function space. arXiv preprint arXiv:1812.03828 (2018)
14. Mixamo: (2018), https://www.mixamo.com/
15. Newell, A., Yang, K., Deng, J.: Stacked hourglass networks for human pose estimation. In: European Conference on Computer Vision. pp. 483–499 (2016)
16. Renderpeople: (2018), https://renderpeople.com/3d-people
17. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: International Conference on Medical image computing and computer-assisted intervention. pp. 234–241. Springer (2015)
18. Saito, S., Huang, Z., Natsume, R., Morishima, S., Kanazawa, A., Li, H.: Pifu: Pixel-aligned implicit function for high-resolution clothed human digitization. In: ICCV (2019)
19. Shen, X., Hertzmann, A., Jia, J., Paris, S., Price, B., Shechtman, E., Sachs, I.: Automatic portrait segmentation for image stylization. Computer Graphics Forum **35**(2), 93–102 (2016)

20. Sloan, P.P., Kautz, J., Snyder, J.: Precomputed radiance transfer for real-time rendering in dynamic, low-frequency lighting environments. ACM Transactions on Graphics **21**(3), 527–536 (2002)
21. Sun, K., Zhao, Y., Jiang, B., Cheng, T., Xiao, B., Liu, D., Mu, Y., Wang, X., Liu, W., Wang, J.: High-resolution representations for labeling pixels and regions. arXiv preprint arXiv:1904.04514 (2019)
22. Tieleman, T., Hinton, G.: Lecture 6.5—RmsProp: Divide the gradient by a running average of its recent magnitude. COURSERA: Neural Networks for Machine Learning (2012)
23. Zeiler, M.D.: Adadelta: an adaptive learning rate method. arXiv preprint arXiv:1212.5701 (2012)
24. Zhang, C., Pujades, S., Black, M.J., Pons-Moll, G.: Detailed, accurate, human shape estimation from clothed 3d scan sequences. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4191–4200 (2017)
25. Zhang, S.H., Li, R., Dong, X., Rosin, P., Cai, Z., Han, X., Yang, D., Huang, H., Hu, S.M.: Pose2seg: Detection free human instance segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 889–898 (2019)
26. Zheng, Z., Yu, T., Wei, Y., Dai, Q., Liu, Y.: Deephuman: 3d human reconstruction from a single image. In: The IEEE International Conference on Computer Vision (ICCV) (October 2019)