Supplementary Material: Point-Based Modeling of Human Clothing

Ilya Zakharkin [1,2*], Kirill Mazur [1*], Artur Grigorev [1], Victor Lempitsky [1,2]

[1] Samsung AI Center, Moscow
[2] Skolkovo Institute of Science and Technology (Skoltech), Moscow

Figure 1: Outfits modeled from single in-the-wild images using our model are retargeted to novel poses. Here the same body shape and two challenging body poses are used for four different garment styles.

1. Draping Network Details

As the outfit style encoder we employ a five-layered perceptron with the first hidden layer having 256 units and the remaining hidden layers with 512 units.

The draping network is trained on eight NVIDIA Tesla P40, with the batch size 8 per GPU. As an optimizer we use the ADAM [6] optimizer. The learning rates for the Cloud Transformer [8], the outfit MLP encoder and the outfit codes are initialized to 0.0001, 0.001, and 0.1 respectively. We halve the learning rates every 50 epochs.

In the main text, we also describe the Cloth3D [4] dataset that our model is trained on. Though it has a diverse set of subjects with varying style, shape, and pose, important thing to note here is that it has quite specific clothing geometry when it comes to pants, shirts, and t-shirts. This peculiarity affects the final quality resulting in some bias when generating the garments of these types. For example, in Figure 8 one can observe wide sleeves of point clouds generated by our model. The bias towards wider than necessary sleeves comes from the training dataset.

*denotes equal contribution. VL is currently with Yandex and Skoltech.

2. Appearance Optimization Details

Our appearance optimization starts with optimizing the outfit code z^* for a new person and then fixing it. We then jointly optimize the neural descriptors T (one for each point in the cloth point cloud) and the parameters ψ of the rendering network R_ψ. The optimization process utilizes the whole training video sequence and takes around 16 hours with one NVIDIA Tesla P40 GPU. In our experiments, training sequences consist of roughly 2800 frames per person for the *AzurePeople* Dataset and roughly 600 frames for the *PeopleSnapshot* dataset.

The optimization is supervised with two loss functions: (1) the VGG19 perceptual loss between real and fake RGB-images and (2) the Dice loss between corresponding segmentation masks. Trained components are optimized using ADAM optimizer with parameters $\beta_1 = 0.5$, $\beta_2 = 0.99$, and the learning rates $lr_R = 1e^{-4}$ and $lr_T = 1e^{-2}$ for the rendering network and the neural descriptors respectively. Such difference in learning rates encourage more information to be stored within the neural descriptors rather than in the renderer parameters.

As a rendering network, we use a lightweight U-net with four downsample and upsample blocks. In total, it has around 2.2M parameters. Each neural descriptor t is a 16-dimensional vector. Overall, there are 8192 trained descriptors, same as the number of points in the outfit pointcloud.

Inpainting modification. Since for many samples in the dataset not every part of outfit is visible (e.g. due to the occlusion of the shoulders and the back by long hair) and therefore not present in the ground truth cloth segmentation masks, we modify our appearance optimization process so

Figure 2: Effect of modified appearance optimization process. Note missing regions in the original result (left), occluded by long hair in the training data, inpainted in the output of the modified model (right).

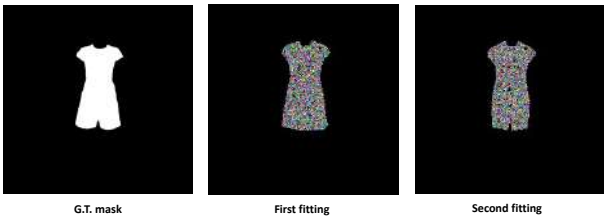


Figure 3: Given the ground truth silhouette (left), the first fitting run (middle) results in a short dress, while the second run (right) using the same method correctly captures the ground truth geometry. The resulting loss value is approximately the same for these two outfit codes.

that output images look better when rendered on top of the SMPL model.

In this modification, we add another loss function that ensures that every pixel within SMPL projection that contains a point from the cloth pointcloud is covered with predicted mask. Also, before calculating the perceptual and the segmentation loss functions we mask out pixels that are simultaneously *present* in the SMPL mask and the outfit mask (obtained from the rendered pointcloud with *floodfill* algorithm) and are *missing* in the ground truth mask. That prevents any supervision from the regions we want to inpaint. This modification results in blurring regions that are not visible in the ground truth data, but present in outfit geometry (see Figure 2)

3. Silhouette Fitting Failure Cases

We observed that our fitting method is sub-optimal on some specific clothing types worn by several people in the PeopleSnapshot [2] dataset. Namely, if a target person wears a pair of shorts, it becomes close to those wearing a short skirt in terms of its A-pose silhouette. As a reference, we show two independent runs of out silhouette fitting on the PeopleSnapshot dataset sample, which resulted in comparable target Chamfer loss (see Figure 3). However, only the second of the two runs correctly captures the clothing geometry. This limitation of our method remains to be addressed as future work.

4. User study protocol

For our user study, we fitted the geometry of human outfits to a single frame with each of the methods compared (*Ours*, Multi-garment net *MGN* [5], *Tex2Shape* [3], *Octopus* [1]). We then randomly sampled a number of body poses from the validation set of the *AzurePeople* dataset and rendered the videos of rotating 3D models in the sampled poses. Models fitted on people from *AzurePeople* Dataset were rendered in poses of the same person, while each *PeopleSnapshot* person was randomly matched with a person from *AzurePeople* due to very limited pose diversity in *PeopleSnapshot*. Since our model outputs a point cloud rather than a mesh, we rendered each point as a sphere (see Figure 4).

Since Multi-garment net [5] requires clothing classes to be explicitly set for each input, we manually labeled outfit types for all the subjects from *AzurePeople* and *PeopleSnapshot* datasets. Each person is assigned one or two classes from the following: Pants, ShortPants, ShirtNoCoat, TShirtNoCoat, and LongCoat (i.e. the classes that MGN was trained for). Note that some people from these datasets have clothing that is not properly represented by one of the mentioned garment types, namely, people in dresses and layered clothing like a shirt worn over a t-shirt. We set the LongCoat class in such cases in order for the model to have a larger SMPL [7] vertices coverage for offset predictions.

The user study was conducted using an online crowdsourcing platform. Participants were presented with two videos of rotating 3D models and a ground truth RGB image. They were asked to choose "which 3D model better represents outfit of the person". The Order of videos in the presented pairs was randomized. Each pair of 3D models was assessed by 30 participants. Each person from the *AzurePeople* appeared in different poses in 6 pairs of videos (total 48 pairs), each person from the *PeopleSnapshot* appeared in the pairs (total 51 pair). In total, user studies for the *AzurePeople* contained 1440 comparisons, and for the *PeopleSnapshot* contained 1530 comparisons.

We present extensive comparisons between our method and previous methods in the supplementary video as well as the images below.

Figure 4: An example of the user study pair presented to participants. This pair contains the output of our method (right), the ground truth image (center), and the output of *Tex2Shape* [3] method (right).

References

[1] T. Alldieck, M. Magnor, B. L. Bhatnagar, C. Theobalt, and G. Pons-Moll. Learning to reconstruct people in clothing from a single rgb camera. In *Proc. CVPR*, 2019. 2, 4, 5, 6, 7

[2] T. Alldieck, M. Magnor, W. Xu, C. Theobalt, and G. Pons-Moll. Video based reconstruction of 3d people models. In *Proc. CVPR*, pages 8387–8397, 2018. 2

[3] T. Alldieck, G. Pons-Moll, C. Theobalt, and M. Magnor. Tex2shape: Detailed full human body geometry from a single image. In *Proc. 3DV*, 2019. 2, 3, 4, 5, 6, 7

[4] H. Bertiche, M. Madadi, and S. Escalera. Cloth3d: Clothed 3d humans. In *Proc. ECCV*, 2020. 1

[5] B. L. Bhatnagar, G. Tiwari, C. Theobalt, and G. Pons-Moll. Multi-garment net: Learning to dress 3d people from images. In *Proc. ICCV*. IEEE, oct 2019. 2, 4, 5, 6, 7

[6] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization, 2017. 1

[7] M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, and M. J. Black. SMPL: A skinned multi-person linear model. *ACM Trans. Graphics (Proc. SIGGRAPH Asia)*, 34(6):248:1–248:16, Oct. 2015. 2

[8] K. Mazur and V. Lempitsky. Cloud transformers. In *Proc. ICCV*, 2021. 1

Figure 5: Comparisons of geometries produced by our method and other approaches for people from *AzurePeople* dataset

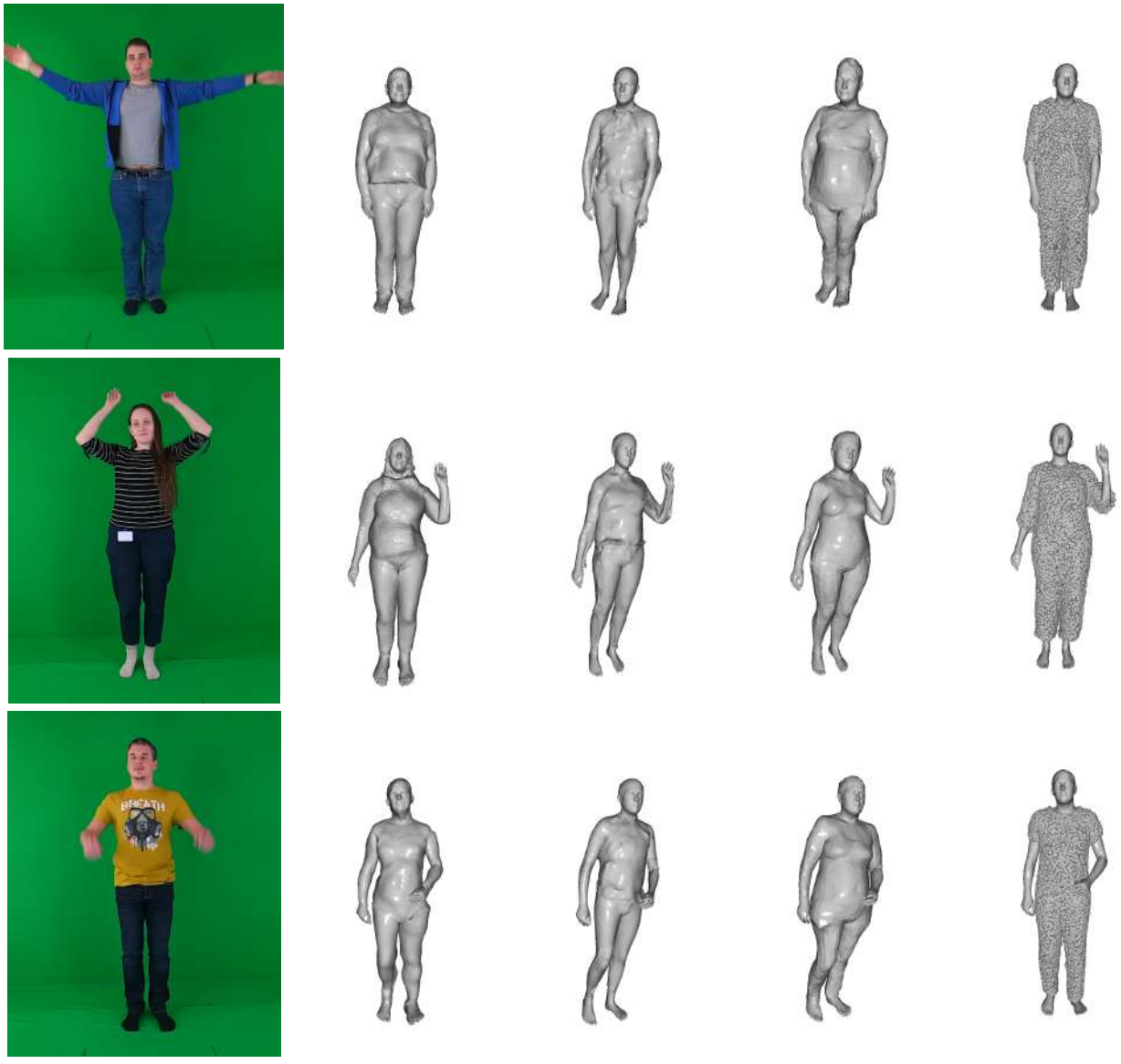
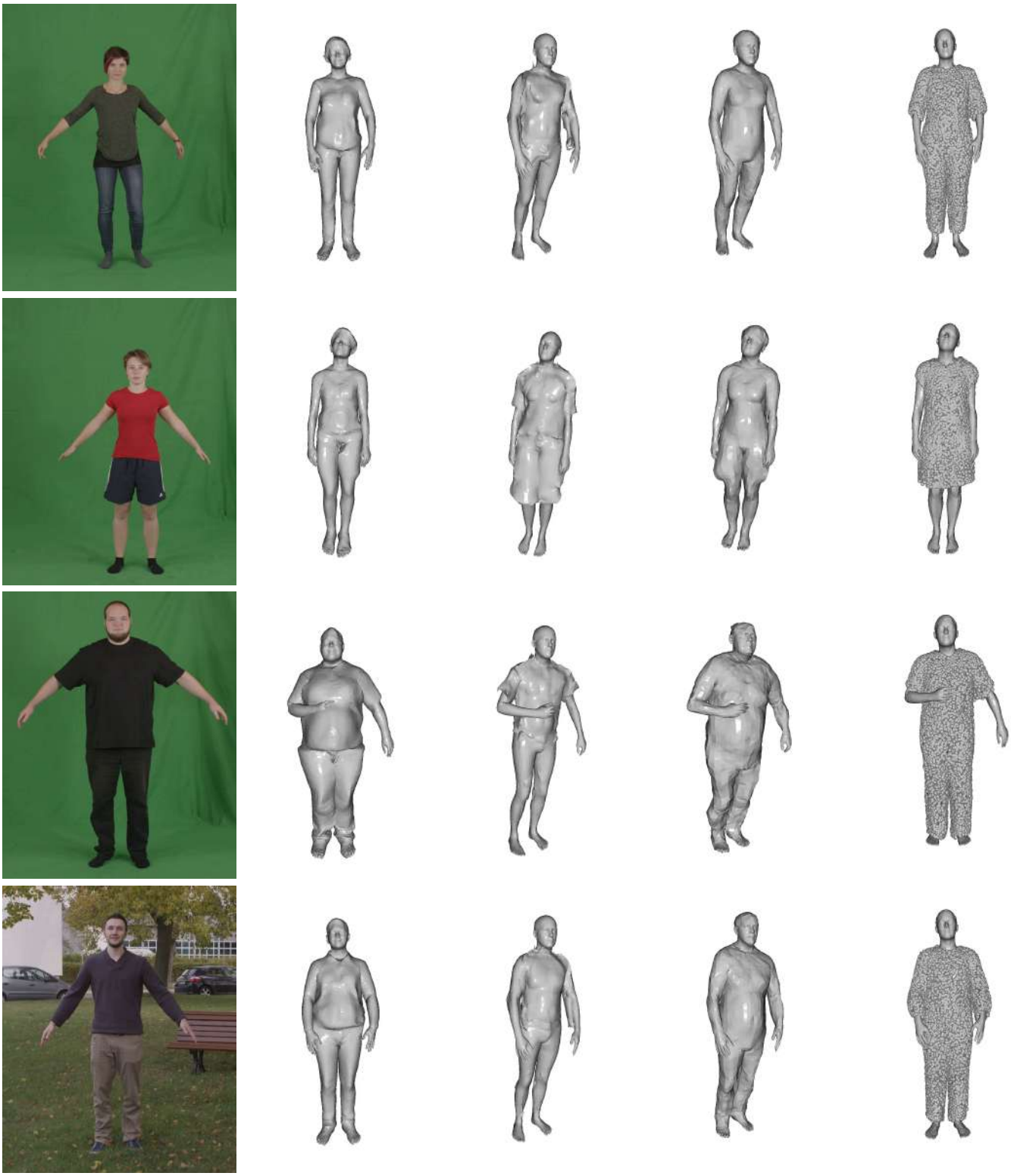


Figure 6: Comparisons of geometries produced by our method and other approaches for the people from *AzurePeople* dataset (Continued.)



Fitting frame | Tex2Shape [3] | MGN [5] | Octopus [1] | Ours

Figure 7: Comparisons of geometries produced by our method and other approaches for the people from *PeopleSnapshot* dataset

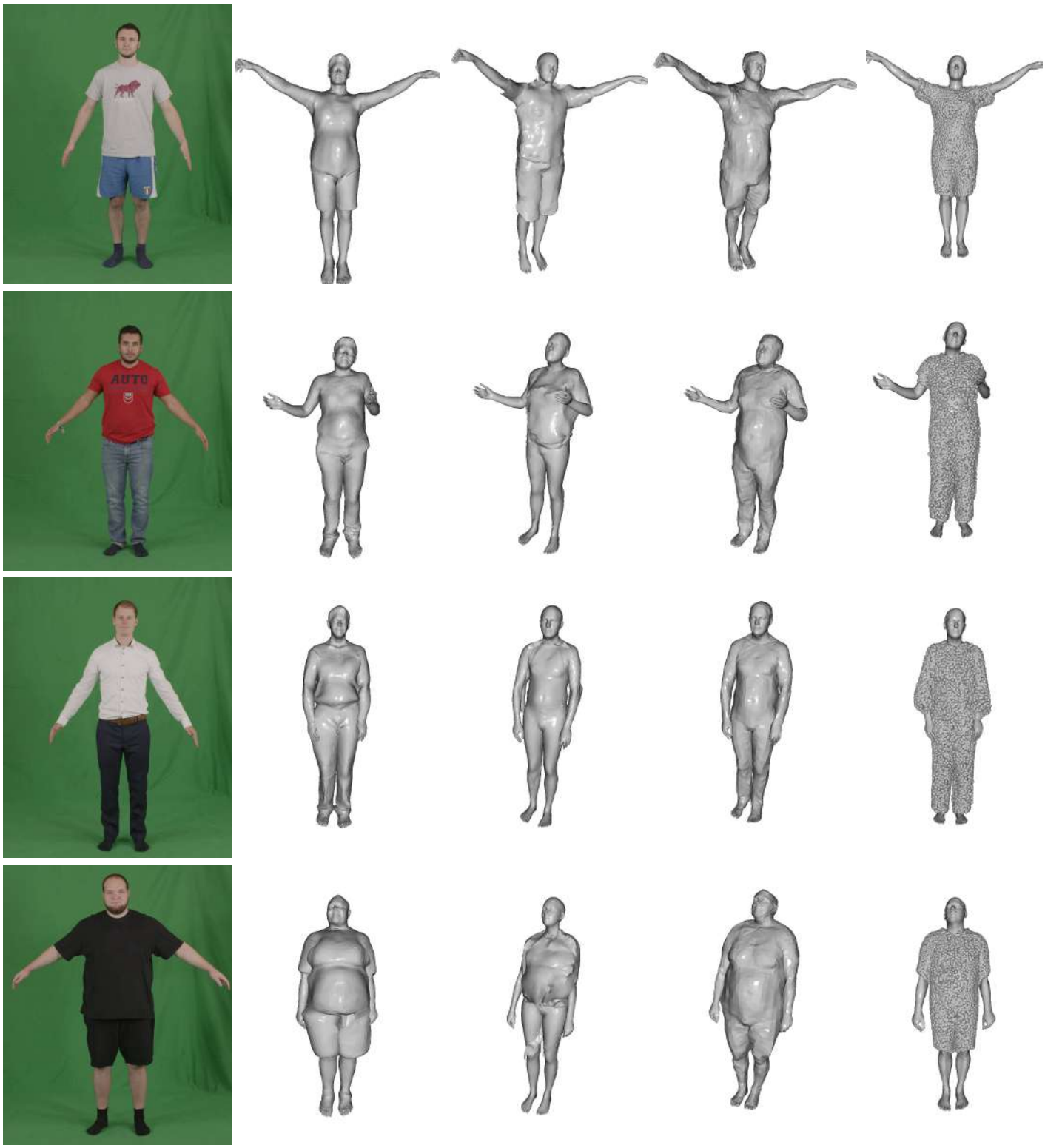


Figure 8: Comparisons of geometries produced by our method and other approaches for the people from the *PeopleSnapshot* dataset (Continued.)