The Power of Points for Modeling Humans in Clothing **Supplementary Material**

Qianli Ma[1,2] Jinlong Yang[1] Siyu Tang[2] Michael J. Black[1]
[1]Max Planck Institute for Intelligent Systems, Tübingen, Germany [2]ETH Zürich
{qma,jyang,black}@tuebingen.mpg.de, {qianli.ma, siyu.tang}@inf.ethz.ch

S1. Implementation Details

S1.1. Model Architecture

We use the SMPL [10] (for CAPE data) and SMPL-X [20] (for ReSynth data) UV maps of $128 \times 128 \times 3$ resolution as pose input, where each pixel is encoded into 64 channels by the pose encoder. The pose encoder is a standard UNet [24] that consists of seven [Conv2d, BatchNorm, LeakyReLU(0.2)] blocks, followed by seven [ReLU, ConvTranspose2d, BatchNorm] blocks. The final layer does not apply BatchNorm.

The geometric feature tensor has the same resolution as that of the pose feature tensor, i.e. $128 \times 128 \times 64$. It is learned in an auto-decoding [18] manner, being treated as a free variable that is optimized together with the network weights during training. The geometric feature tensor is followed by three learnable convolutional layers, each with a receptive field of 5, before feeding it to the shape decoder. We find that these convolutional layers help smooth the features spatially, resulting in a lower noise level in the outputs.

The pose and geometric feature tensors are concatenated along the feature channel. In all experiments, we query the feature tensor with a 256×256 UV map, i.e. the concatenated feature tensor is spatially $4\times$ bilinearly upsampled. The output point cloud has 50K points.

At each query location, the concatenated pose and geometry feature (64+64-dimensional), together with the 2D UV coordinate of the query point, are fed into an 8-layer MLP. The intermediate layers' dimensions are (256, 256, 256, 386, 256, 256, 256, 3), with a skip connection from the input to the 4th layer as in DeepSDF [18]. From the 6th layer, the network branches out 2 heads with the same architecture to predict the displacements and point normals, respectively. All but the last layer use BatchNorm and a Softplus non-linearity with $\beta = 20$. The predicted normals are normalized to unit length.

S1.2. Training

We train POP with the Adam [8] optimizer with a learning rate of 3.0×10^{-4}, a batch size of 4, for 400 epochs. The displacement and normal prediction modules are trained jointly. As the normal loss relies on the nearest neighbor ground truth points found by the Chamfer Distance, we only turn it on when $\mathcal{L}_d$ roughly stabilizes from the 250th epoch. The loss weights are set to $\lambda_{\text{d}} = 2.0 \times 10^4$, $\lambda_{\text{rd}} = 2.0 \times 10^3$, $\lambda_{\text{rg}} = 1.0$, $\lambda_{\text{n}} = 0.0$ at the beginning of the training, and $\lambda_{\text{n}} = 0.1$ from the 250th epoch.

S1.3. Data Processing

We normalize all the data examples by removing the body translation and global orientation from them. From each clothed body surface, we sample 40K points to serve as training ground truth. Note that we do not rely on any connectivity information in the registered meshes from the CAPE dataset.

S1.4. Baselines

NASA. We re-implement the NASA [5] model in PyTorch and ensure the performance is on par with that reported in the original paper. For evaluating NASA results, we first extract a surface using Marching Cubes [11] and then sample the same number of points (50K) from it for a fair comparison. The sampling is performed and averaged over three repetitions.

SCALE. We employ the same training schedule and the number of patches (798) as in the original SCALE paper [12], using the implementation released by the authors. We sample 64 points per patch at both training and inference to achieve the same number of output points as ours for a fair comparison.

LBS. In the main paper Sec. 4.3, we compare with the Linear Blend Skinning (LBS) in the single scan animation task. This is done with the help of the SMPL [10] body model: we first optimize the SMPL body shape and pose parameters to fit a minimally-clothed body to the given scan, and then displace the vertices such that the final surface mesh aligns with the scan. The fitted clothed body model is then reposed by the target pose parameters.

S1.5. User Study

We conduct a large-scale user study on the Amazon Mechanical Turk to get a quantitative evaluation of the visual quality of our model outputs against the point-based method SCALE [12]. We evaluate over 6,000 unseen test examples in the CAPE and ReSynth datasets, from different subjects, performing different poses. For each example, the point cloud output from POP and SCALE are both rendered with a surfel-based renderer by Open3D [30] under the same rendering settings (an example of such rendering is the Fig. 1 of the main paper). We then present both images side-by-side to the users and ask them to choose the one that they deem a higher visual quality. The left-right ordering of the images is randomly shuffled for each example to avoid users' bias to a certain side. The users are required to zoom-in the images before they are able to make the evaluation, and we do not set a time limit for the viewing. Each image pair is evaluated by three users, and the final results in the main paper is averaged from all the user choices on all examples.

S2. Datasets

ReSynth. The 24 outfits designed by the clothing designer include varied types and styles: shirts, T-shirts, pants, shorts, skirts, long dresses, jackets, to name a few. For each outfit, we find the SMPL-X [20] body (provided by the AGORA [19] dataset) that fits the subject's body shape in its corresponding original scan. We then use physics-based simulation [4] to drape the clothing on the bodies and animate them with a consistent set of motion sequences of the subject 00096 from the CAPE dataset. The simulation results are inspected manually to remove problematic frames, resulting in 984 frames for training and 347 frames for test for each outfit. Examples from ReSynth are shown in Fig. S1. We will release the dataset for research purposes.

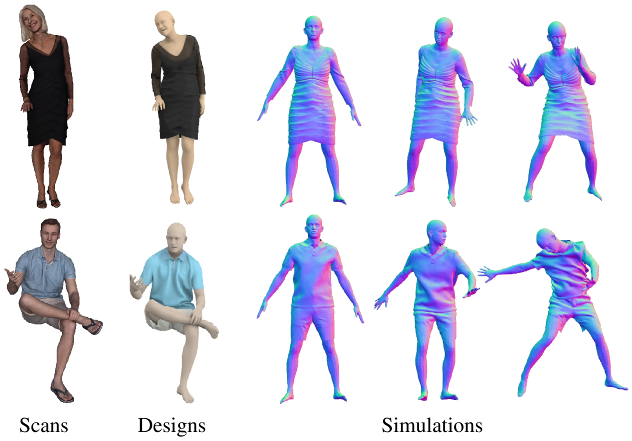

Figure S1: Examples from our ReSynth dataset. The clothing is designed based on real-world scans [23], draped on the SMPL-X [20] body model, and then simulated using Deform Dynamics [4].

CAPE. The CAPE dataset [13] provides registered mesh pairs of (unclothed body, clothed body) of humans in clothing performing motion sequences. The three subjects (00096, 00215, 03375) that we use in the experiments have in total 14 outfits comprising short/long T-shirts, short/long pants, a dress shirt, a polo shirt, and a blazer. For each outfit, the motion sequences are randomly split into training (70%) and test (30%) sets.

S3. Extended Results and Discussions

Here we provide extended analysis and discussions regarding the main paper Tab. 2. The implicit surface baseline, NASA [5], shows a much higher error than other methods. We find that it is majorly caused by the occasional missing body parts in its predictions. This happens more often for challenging, unseen body poses. The incomplete predictions thus lead to exceptionally high bi-directional Chamfer distance on a number of examples, hence a high average error.

Our approach is based on the SCALE [12] baseline, but it achieves on average 11.4% (on CAPE data) and 9.1% (on ReSynth) lower errors than SCALE, with both margins being statistically significant (p-value$\ll 1e$-4 in the Wilcoxon signed-rank test). Together with the user study results in the main paper, this shows a consistent improvement on the representation power.

In Figs. S3 and S4 we show extended qualitative comparisons with NASA [5] and SCALE [12] from the pose generalization experiment (Sec. 4.1 in the main paper). Please refer to the supplementary video at https://qianlim.github.io/POP for animated results.

S4. Run-time Comparison

Here we compare the inference speed of POP with the implicit surface baseline, NASA [5], and the patch-based baseline, SCALE [12].

To generate a point cloud with 50K points, POP takes on average 48.8ms, and SCALE takes 42.4ms. The optional meshing step using the Poisson Reconstruction [7] takes 1.1s if a mesh is desired. Both explicit representations have comparable run-time performance. In contrast, NASA requires densely evaluating occupancy values over the space in order to reconstruct an explicit surface, which takes 12.2s per example. This shows the speed advantage of the explicit representations over the implicit ones.

S5. Limitations and Failure Cases

As discussed in the final section of the main paper, the major limitation of our approach lies on the use of the UV map. Although the UV maps are widely used to reconstruct and model human faces [6, 14, 28], one can encounter additional challenges when applying this technique to human bodies. On a full body UV map such as that of SMPL, different body parts are represented as separate "islands" on the UV map, see Fig. 2 in the main paper. Consequently, the output may suffer from discontinuities at the UV islands' boundaries. Qualitatively, this may occasionally result in visible "seams" between certain body parts as shown in Fig. S2 (a-b), or an overly sparse distribution of points between the legs in the case of dresses as shown in Fig. S2 (c), leading to sub-optimal performance when training a unified model for both pants and skirts.

Note, however, that such discontinuities are not always the case. Intuitively, as the input UV positional map encodes (x, y, z) coordinates on the 3D body, the network can utilize not only the proximity in the UV space but also that in the original 3D space. We believe that the problem originates from the simple 2D convolution in the UNet pose encoder. A promising solution is to leverage a more continuous parameterization for the body surface manifold that is compatible with existing deep learning architectures. We leave this for future work.

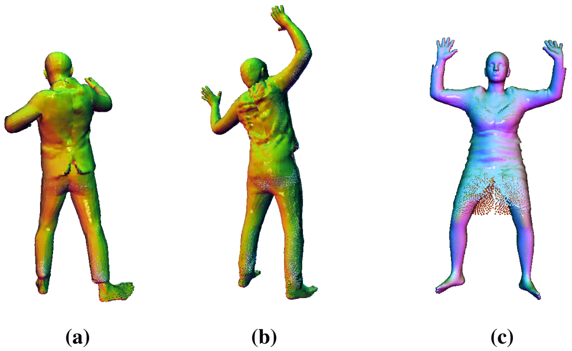

Figure S2: Illustrations of our limitations.

S6. Further Discussions on Related Work

Here we discuss the relationship of our method to recent work that uses similar techniques or that aims similar goals.

We represent clothing as a displacement field on the minimally-clothed body, in the canonical pose space. This helps factor out the effect of the articulated, rigid transformations that are directly accessible from the underlying posed body. In this way, the network can focus on the non-rigid, residual clothing deformation. Such technique is becoming increasingly popular for clothed human shape models that use meshes [13], point clouds [12], and implicit functions [2, 9, 17, 25, 29].

Our shape decoder is a coordinate-based multi-layer perceptron (MLP), reminiscent of the recent line of work on neural implicit surfaces [3, 15, 18] and neural radiance fields [16]. These methods learn to map a neural feature at a given query location into a certain quantity, e.g. a signed distance [18], occupancy [15], color and volume density [16], or a displacement vector in our case. Our work differs from others majorly in that the querying coordinates live on a 2-manifold (the body surface)[1] instead of $\mathbb{R}^3$. Moreover, our point cloud representation belongs to the *explicit* representation category, retaining an advantage in the inference speed compared to the implicit methods. With the recent progress in differentiable and efficient surface reconstruction from point clouds [21, 26], it becomes possible to flexibly convert between point clouds and meshes in various applications.

Recent work on deformable face modeling [14] and pose-controlled free-view human synthesis [9] employ similar network architectures as ours, despite the difference in the goals, tasks and detailed techniques. While the commonality implies the efficacy of the high-level architectural design, it remains interesting to combine the detailed technical practices from each piece of work. It is also interesting to note the connection between our geometric feature tensor and the *neural texture* in recent work on neural rendering [22, 27]: both concepts learn a spatial neural representation that controls the output, revealing a connection between modeling the 3D geometry and 2D appearances.

Finally, in concurrent work, MetaAvatar [29] also learns a multi-subject model of clothed humans, which can generalize to unseen subjects using only a few depth images. Unlike our auto-decoding learning of the geometric feature tensor, MetaAvatar uses meta-learning to learn a prior of pose-dependent cloth deformation across different subjects and clothing types that helps generalize to unseen subjects and clothing types at test-time. We believe both approaches will inspire future work on cross-garment modeling and automatic avatar creation.

[1] The concurrent work by Burov et al. [1] also maps the points on the SMPL body surface to a continuous clothing offset field, but the points' Euclidean coordinates (with positional encoding) are used to query the shape decoder.

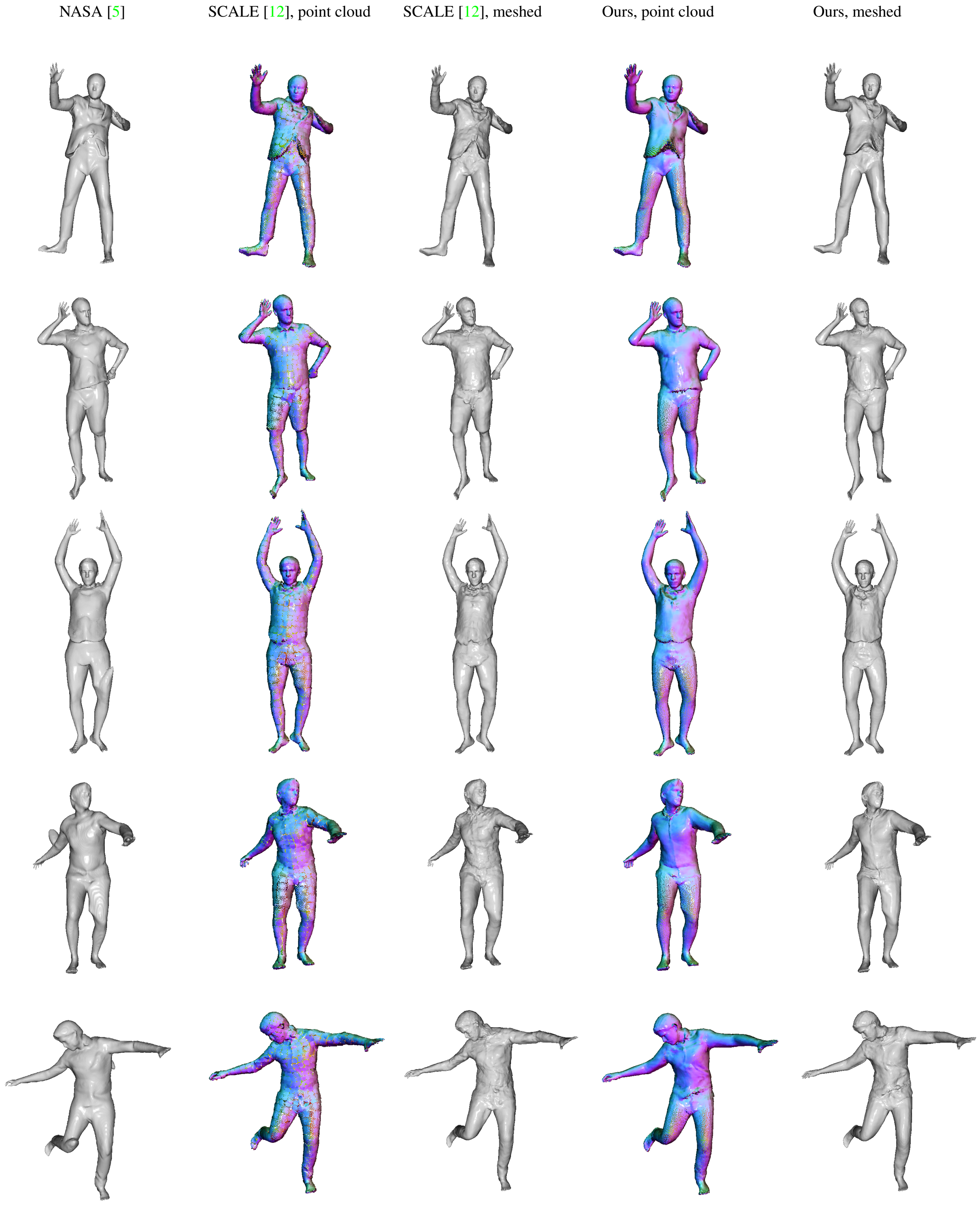

Figure S3: Extended qualitative results from the pose generalization experiment (main paper Sec. 4.1), on the CAPE dataset. Best viewed zoomed-in on a color screen.

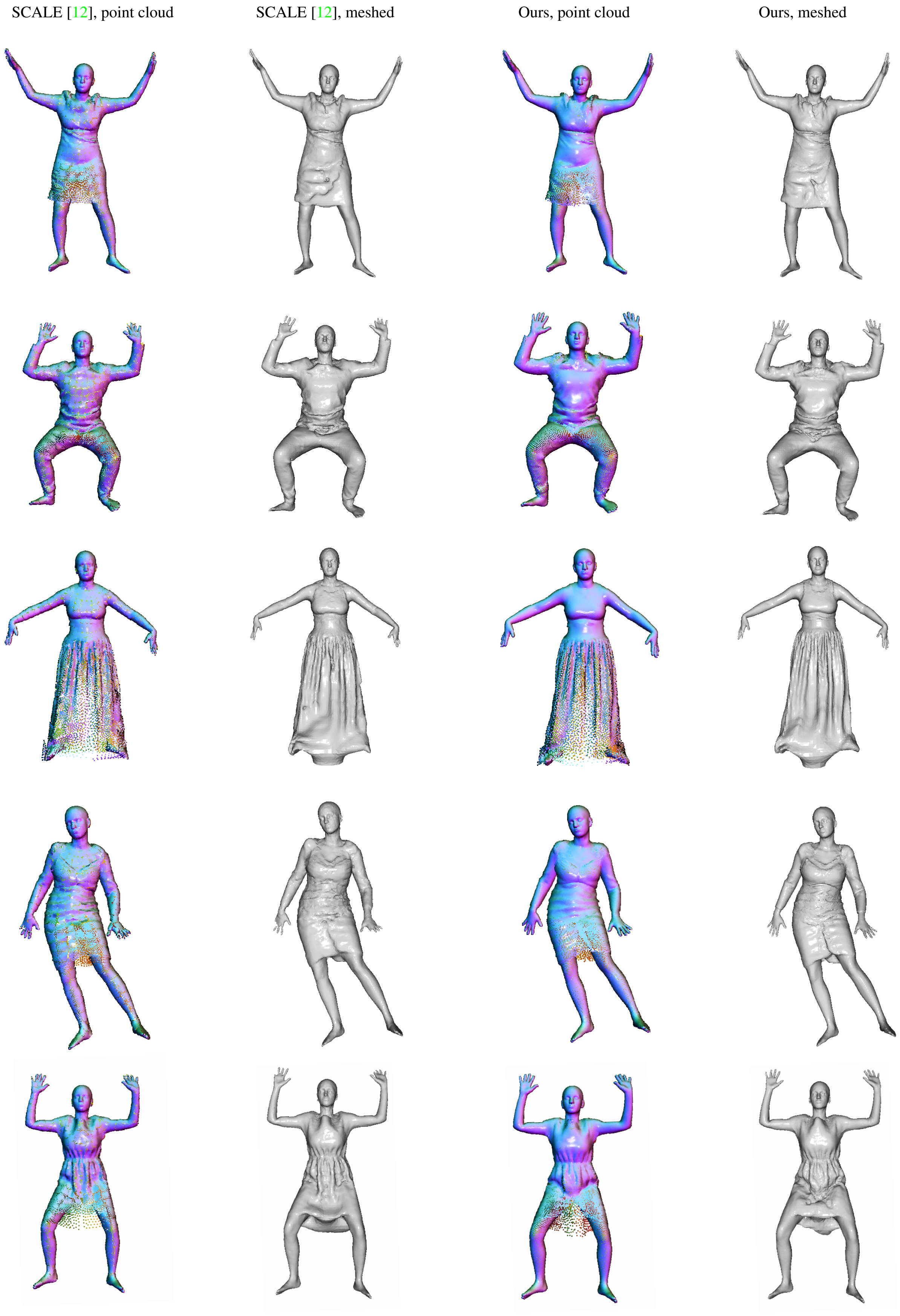

Figure S4: Extended qualitative results from the pose generalization experiment (main paper Sec. 4.1), on the ReSynth dataset. Best viewed zoomed-in on a color screen.

References

[1] Andrei Burov, Matthias Nießner, and Justus Thies. Dynamic surface function networks for clothed human bodies. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. 3

[2] Xu Chen, Yufeng Zheng, Michael J Black, Otmar Hilliges, and Andreas Geiger. SNARF: Differentiable forward skinning for animating non-rigid neural implicit shapes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. 3

[3] Zhiqin Chen and Hao Zhang. Learning implicit fields for generative shape modeling. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 5939–5948, 2019. 3

[4] Deform Dynamics. https://deformdynamics.com/. 2

[5] Boyang Deng, JP Lewis, Timothy Jeruzalski, Gerard Pons-Moll, Geoffrey Hinton, Mohammad Norouzi, and Andrea Tagliasacchi. Neural articulated shape approximation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 612–628, 2020. 1, 2, 4

[6] Yao Feng, Fan Wu, Xiaohu Shao, Yanfeng Wang, and Xi Zhou. Joint 3D face reconstruction and dense alignment with position map regression network. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 534–551, 2018. 3

[7] Michael Kazhdan, Matthew Bolitho, and Hugues Hoppe. Poisson surface reconstruction. In *Proceedings of the fourth Eurographics Symposium on Geometry Processing*, volume 7, 2006. 2

[8] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2015. 1

[9] Lingjie Liu, Marc Habermann, Viktor Rudnev, Kripasindhu Sarkar, Jiatao Gu, and Christian Theobalt. Neural Actor: Neural free-view synthesis of human actors with pose control. *arXiv preprint: 2106.02019*, 2021. 3

[10] 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. 1

[11] William E Lorensen and Harvey E Cline. Marching cubes: A high resolution 3D surface construction algorithm. In *ACM SIGGRAPH Computer Graphics*, volume 21, pages 163–169, 1987. 1

[12] Qianli Ma, Shunsuke Saito, Jinlong Yang, Siyu Tang, and Michael J. Black. SCALE: Modeling clothed humans with a surface codec of articulated local elements. In *Proceedings IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, June 2021. 1, 2, 3, 4, 5

[13] Qianli Ma, Jinlong Yang, Anurag Ranjan, Sergi Pujades, Gerard Pons-Moll, Siyu Tang, and Michael J. Black. Learning to dress 3D people in generative clothing. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 6468–6477, 2020. 2, 3

[14] Shugao Ma, Tomas Simon, Jason Saragih, Dawei Wang, Yuecheng Li, Fernando De la Torre, and Yaser Sheikh. Pixel codec avatars. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 64–73, June 2021. 3

[15] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3D reconstruction in function space. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 4460–4470, 2019. 3

[16] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. NeRF: Representing scenes as neural radiance fields for view synthesis. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 405–421. Springer, 2020. 3

[17] Pablo Palafox, Aljaz Bozic, Justus Thies, Matthias Nießner, and Angela Dai. Neural parametric models for 3D deformable shapes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. 3

[18] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. DeepSDF: Learning continuous signed distance functions for shape representation. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 165–174, 2019. 1, 3

[19] Priyanka Patel, Chun-Hao Huang Paul, Joachim Tesch, David Hoffmann, Shashank Tripathi, and Michael J. Black. AGORA: Avatars in geography optimized for regression analysis. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 13468–13478, June 2021. 2

[20] Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed A. A. Osman, Dimitrios Tzionas, and Michael J. Black. Expressive body capture: 3D hands, face, and body from a single image. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 10975–10985, 2019. 1, 2

[21] Songyou Peng, Chiyu Jiang, Yiyi Liao, Michael Niemeyer, Marc Pollefeys, and Andreas Geiger. Shape as points: A differentiable poisson solver. *arXiv preprint: 2106.03452*, 2021. 3

[22] Amit Raj, Julian Tanke, James Hays, Minh Vo, Carsten Stoll, and Christoph Lassner. ANR: Articulated neural rendering for virtual avatars. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 3722–3731, June 2021. 3

[23] Renderpeople, 2020. https://renderpeople.com. 2

[24] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pages 234–241, 2015. 1

[25] Shunsuke Saito, Jinlong Yang, Qianli Ma, and Michael J. Black. SCANimate: Weakly supervised learning of skinned clothed avatar networks. In *Proceedings IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, June 2021. 3

[26] Nicholas Sharp and Maks Ovsjanikov. PointTriNet: Learned triangulation of 3D point sets. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 762–778, 2020. 3

[27] Justus Thies, Michael Zollhöfer, and Matthias Nießner. Deferred neural rendering: Image synthesis using neural textures. *ACM Transactions on Graphics (TOG)*, 38(4):1–12,

2019. 3

[28] Diego Thomas and Rin-Ichiro Taniguchi. Augmented blendshapes for real-time simultaneous 3D head modeling and facial motion capture. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 3299–3308, June 2016. 3

[29] Shaofei Wang, Marko Mihajlovic, Qianli Ma, Andreas Geiger, and Siyu Tang. MetaAvatar: Learning animatable clothed human models from few depth images. *arXiv preprint:2106.11944*, 2021. 3

[30] Qian-Yi Zhou, Jaesik Park, and Vladlen Koltun. Open3D: A modern library for 3D data processing. *arXiv preprint: 1801.09847*, 2018. 2