# Supplementary Document for
# "High-Fidelity Human Avatars from a Single RGB Camera"

In this document, we provide the following supplementary contents:

- Loss Functions.
- User Study.
- Texture Maps Before and After Optimization.
- More Comparison Results.
- More Application Results.
- Failure Cases.

We also provide a demo video along with this document.

## 1. Loss Functions

### 1.1. Geometry Loss

**The 2D Joint Loss.** The detected 2D joint locations are used as weak supervision on the projected joints of *SMPL*, which is defined as:

$$\mathcal{L}_{J_{2D}} = \|J - \hat{J}\|_2, \tag{1}$$

where $J$ is the detected 2D joint locations predicted by a human pose estimation method [3], and $\hat{J}$ is the 2D joint locations regressed from the predicted vertices.

**The Silhouette Loss.** The binary segmentation masks are used to supervise the shape of person to capture instance-specific details, which is written as:

$$\mathcal{L}_{M_{2D}} = \|b(I) - \pi_c(M(\beta, \theta, \mathbf{D}))\|_2, \tag{2}$$

where $b(I)$ is the binary segmentation mask predicted by the matting method [6], $\pi_c$ is the perspective projection matrix, and $M(\cdot)$ is a function that maps pose and shape parameters into the vertices of *SMPL* model .

**The Photometric Loss.** This term is used to guide the vertices to be close to the right positions and relieve the misalignment of geometry, which is formulated as:

$$\mathcal{L}_{I_{2D}} = \|I - R(M(\beta, \theta, \mathbf{D}), \mathcal{T})\|_2, \tag{3}$$

where $R$ is the image formation function, $\mathcal{T}$ is the neural texture.

**The Laplacian Loss.** This term is used to prevent the vertices from moving freely and it serves as a local detail-preserving operator that encourages neighboring vertices to have the same movement, which is defined as:

$$\mathcal{L}_{lap} = \|L(M(\beta, \theta, \mathbf{D})) - L(M(\beta, \theta, \mathbf{0}))\|_2, \tag{4}$$

where $L$ is the Laplacian operator, and $\mathbf{0}$ is a zero matrix with the same size as the offsets $\mathbf{D}$.

**The Normal Loss.** This term is used to enhance the geometric details. We use the cosine distance to measure the difference between the predicted normal map and the ground truth, which is formulated as:

$$\mathcal{L}_n = -\frac{< N, \hat{N} >}{\|N\| \cdot \|\hat{N}\|}, \tag{5}$$

where $N$ is the pseudo ground truth estimated by [10], $\hat{N}$ is the predicted normal map, and $< \cdot, \cdot >$ is the cosine function.

Therefore, the overall geometry loss can be written as:

$$\begin{aligned}\mathcal{L}_g = \lambda_{J_{2D}} \mathcal{L}_{J_{2D}} + \lambda_{M_{2D}} \mathcal{L}_{M_{2D}} + \lambda_{I_{2D}} \mathcal{L}_{I_{2D}} \\ + \lambda_{lap} \mathcal{L}_{lap} + \lambda_n \mathcal{L}_n,\end{aligned} \tag{6}$$

where $\lambda_{J_{2D}}, \lambda_{M_{2D}}, \lambda_{I_{2D}}, \lambda_{lap}, \lambda_n$ are the weights that balance the contributions of individual loss terms.

### 1.2. Appearance Loss

**The Data Loss.** We employ an $\mathcal{L}_d$ loss between the generated image and the ground truth, which is defined as:

$$\mathcal{L}_d = \|I_p - I_g\|_1, \tag{7}$$

where $I_p$ and $I_g$ represent the predicted image and the ground truth, respectively.

**The Perceptual Loss.** We adopt a feature loss to increase the sharpness of the output images. The perceptual loss calculates the distances between activation layers of pre-trained VGG-16 network, which can be written as:

$$\mathcal{L}_p = \sum_i \|\phi_i(I_p) - \phi_i(I_g)\|_1, \tag{8}$$

| VideoAvatar [2] | Octopus [1] | Ours |
|---|---|---|
| 10.98% | 11.42% | **77.60%** |

Table 1. The percentage of each method considered to be ranked first on texture map generation.

| Neural Body [8] | HF-NHMT [5] | StylePeople [4] | Ours |
|---|---|---|---|
| 7.35% | 7.08% | 7.70% | **77.87%** |

Table 2. The percentage of each method considered to be ranked first on novel view synthesis.

| w/o TS | w/o REF | Full |
|---|---|---|
| 10.05% | 24.50% | **65.45%** |

Table 3. The percentage of each variant considered to be ranked first on ablation study of different supervision and sharpening schemes.

where $\phi_i$ means the $i$-th activation layer of VGG-16 network.

**The Adversarial Loss.** This term is employed to penalize the distribution difference between predicted (fake) images $I_p$ and target (real) images $I_g$, and to improve the visual quality of results. To stabilize the training of GANs, we adopt the Spectral Normalized Markovian Discriminator [11]. To discriminate whether the input is real or fake, the losses we used are as follows:

$$\mathcal{L}_G = -\mathbb{E}\left[D^{sn}(G(I_{uv}))\right], \tag{9}$$

$$\begin{aligned}\mathcal{L}_{D^{sn}} = {} & \mathbb{E}\left[ReLU\left(1 - D^{sn}(I_g)\right)\right] + \\ & \mathbb{E}\left[ReLU\left(1 + D^{sn}(G(I_{uv}))\right)\right],\end{aligned} \tag{10}$$

where $D^{sn}$ represents the spectral-normalized discriminator, and $G$ is the image generator that takes the UV-map $I_{uv}$ as the input.

Therefore, the overall appearance loss can be written as:

$$\mathcal{L}_a = \lambda_d \mathcal{L}_d + \lambda_p \mathcal{L}_p + \lambda_G \mathcal{L}_G, \tag{11}$$

where $\lambda_d, \lambda_p, \lambda_G$ are the weights that balance the contributions of individual loss terms.

## 2. User Study

To better evaluate the proposed method, we perform a perceptual evaluation with a user study which consists of 3 group tests. The first group shows the results of VideoAvatar [2], Octopus [1] and our method on 6 cases of texture map generation. The second group shows the results of Neural Body [8], HF-NHMT [5], StylePeople [4] and our method on 6 cases of novel view synthesis. The third group shows 2 cases of ablation study on reference branch and training scheme. The users are required to sort the results according to visual quality. We have collected answers from 188 participants, including 66 females and 122 males with different ages (5 users below 18, 157 users between 18 and 40, 20 users between 40 and 60, and 6 users beyond 60). We evaluate the percentage of each method considered to be ranked first, and calculate the median on each group test. The statistical results of 3 group tests are shown in Tables 1-3, respectively. As shown in Table 1 and Table 2, more than 75% of users consider that the results of our method have the best visual quality, which shows that our method outperforms the other methods on both texture map generation and novel view synthesis. Table 3 shows that more than 65% of users agree that the full model achieves the best visual results, which proves the effectiveness of our reference branch and training scheme.

## 3. Texture Maps Before and After Optimization

To generate a seamless and sharp texture map, we design a reference-based neural rendering network and exploit a sharpening-guided fine-tuning strategy in a coarse-to-fine manner. Figure 1 shows the texture maps before and after refinement. As shown in the figure, the quality of texture maps is obviously improved after our refinement. Our reference-based neural rendering network learns a joint representation between geometry and input image, which relieves the misalignment of geometry and enables to generate sharp and seamless texture maps.

## 4. More Comparison Results

Table 4 gives the comparison of whether to support monocular input, fully-textured avatar, high-fidelity novel view synthesis, and high-fidelity novel pose synthesis. As shown in the table, only our method supports all the cases.

We evaluate the avatar generation performance by comparing with two state-of-the-art video-based methods VideoAvatar [2] and Octopus [1]. More visual results on *People-Snapshot* dataset are shown in Figures 2-4. Compared with Octopus [1], our method generates seamless tex-

| Method | Monocular Input | Fully-textured Avatar | Novel View Synthesis | Novel Pose Synthesis |
|---|---|---|---|---|
| StylePeople [4] / ANR [9] | ✓ | ✗ | ✓ | ✓ |
| Neural Body [8] | ✓ | ✗ | ✓ | ✗ |
| Animatable Nerf [7] | ✗ | ✗ | ✓ | ✓ |
| HF-NHMT [5] | ✓ | ✗ | ✓ | ✓ |
| Octopus [1] / VideoAvatar [2] | ✓ | ✓ | ✗ | ✗ |
| Ours | ✓ | ✓ | ✓ | ✓ |

Table 4. Comparison with state-of-the-art methods. Novel view synthesis and novel pose synthesis mean high-fidelity results. ✗: not supported, ✓: supported.

| Method | Neural Body [8] | HF-NHMT [5] | StylePeople [4] | Ours |
|---|---|---|---|---|
| *SelfieVideo* | 81.8043 | 45.1285 | 63.8366 | **28.1964** |
| *PeopleSnapshot [2]* | 48.9066 | 91.0957 | - | **25.5928** |

Table 5. Quantitative comparison for novel view synthesis using FID. -: not available.

ture maps, and there are no texture mistakes or lost patterns in our generated texture maps. In a word, our method can generate seamless and sharp texture maps, with better quality compared with the state-of-the-art methods, which benefits from our coarse-to-fine framework and sharpening-guided fine-tuning strategy. Some estimated texture maps are given in Figure 5.

Besides, Figure 6 and Figure 7 show the comparison results of the reconstructed geometries. Our method can reconstruct more accurate and detailed geometry, benefiting from the design of dynamic surface network.

## 5. More Application Results

**Shape Editing.** Benefiting from our design of the dynamic surface network which disentangles the shape and texture of the person, our method can achieve shape editing by changing the parameters of the SMPL model. Figure 8 shows some neural rendering results of one person with the upper-bodies changing from thin to fat. It can be seen that the texture is not distorted as the shape changes, which proves that our method can disentangle the shape and texture of the person.

**Novel View Synthesis.** Given a target view, we can generate a view-conditioned UV-map with rasterization using z-buffer. With the corresponding UV-map, the geometry is rasterized using a neural texture by bilinear sampling and then is translated to an RGB image using a neural network. We compare our method with three state-of-the-art methods Neural Body [8], HF-NHMT [5] and StylePeople [4]. The trained models of Neural Body [8] and HF-NHMT [5] are generated by the official implementations, and the trained models of StylePeople [4] on 20 videos of *SelfieVideo* are provided by the authors. Table 5 gives the quantitative results on the two datasets. Due to lack of ground truths, FID is calculated by computing the distance between distributions of the generated images and the captured images. Our method outperforms the other methods.

## 6. Failure Cases

Although our method generates high-fidelity images with detailed textures in most cases, it cannot cope with extremely complex textures due to insufficient representation capacity of network. Figure 9 shows some examples of failure cases. For extremely complex patterns, our method fails to generate photo-realistic results and cannot generate sharp texture maps. In further work, we will combine efficient implicit representations, *e.g.*, implicit surfaces and NeRFs, to break through the limitation of the fixed topology, improve the representation capacity of the framework and generate more high-fidelity avatars.

## References

[1] Thiemo Alldieck, Marcus Magnor, Bharat Lal Bhatnagar, Christian Theobalt, and Gerard Pons-Moll. Learning to reconstruct people in clothing from a single RGB camera. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019. 2, 3, 5, 6, 7, 8, 9, 10

[2] Thiemo Alldieck, Marcus Magnor, Weipeng Xu, Christian Theobalt, and Gerard Pons-Moll. Video based reconstruction of 3D people models. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018. 2, 3, 5, 6, 7, 8, 9, 10

[3] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2017. 1

[4] Artur Grigorev, Karim Iskakov, Anastasia Ianina, Renat Bashirov, Ilya Zakharkin, Alexander Vakhitov, and Victor

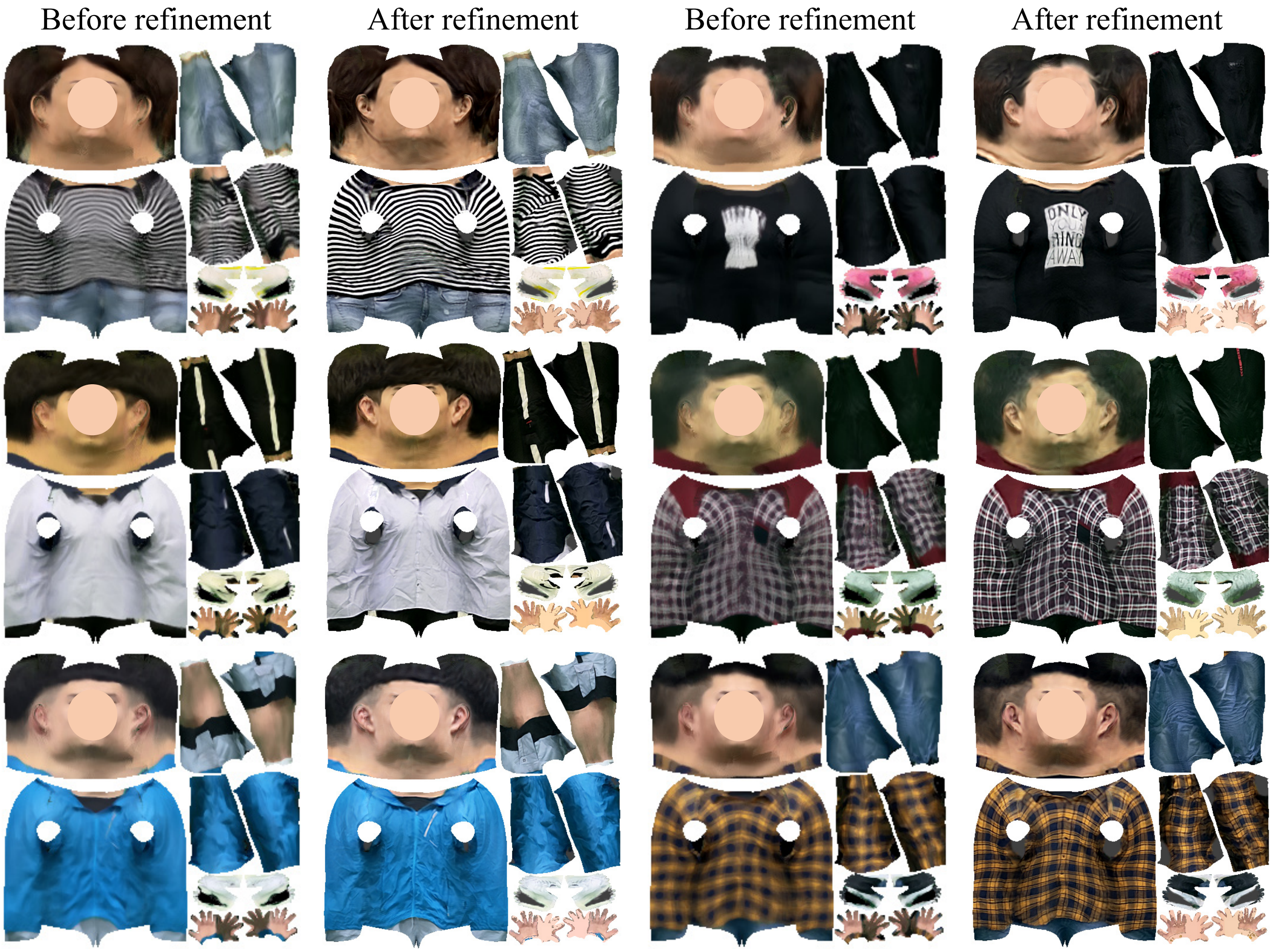

Figure 1. The texture maps before and after refinement.

Lempitsky. StylePeople: A generative model of fullbody human avatars. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021. 2, 3

[5] Moritz Kappel, Vladislav Golyanik, Mohamed Elgharib, Jann-Ole Henningson, Hans-Peter Seidel, Susana Castillo, Christian Theobalt, and Marcus Magnor. High-fidelity neural human motion transfer from monocular video. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021. 2, 3

[6] Zhanghan Ke, Kaican Li, Yurou Zhou, Qiuhua Wu, Xiangyu Mao, Qiong Yan, and Rynson W.H. Lau. Is a green screen really necessary for real-time portrait matting? *arXiv:2011.11961*, 2020. 1

[7] Sida Peng, Junting Dong, Qianqian Wang, Shangzhan Zhang, Qing Shuai, Hujun Bao, and Xiaowei Zhou. Animatable neural radiance fields for human body modeling. In *Int. Conf. Comput. Vis.*, 2021. 3

[8] Sida Peng, Yuanqing Zhang, Yinghao Xu, Qianqian Wang, Qing Shuai, Hujun Bao, and Xiaowei Zhou. Neural Body: Implicit neural representations with structured latent codes for novel view synthesis of dynamic humans. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021. 2, 3

[9] Amit Raj, Julian Tanke, James Hays, Minh Vo, Carsten Stoll, and Christoph Lassner. ANR-Articulated neural rendering for virtual avatars. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021. 3

[10] Shunsuke Saito, Tomas Simon, Jason Saragih, and Hanbyul Joo. PIFuHD: Multi-level pixel-aligned implicit function for high-resolution 3D human digitization. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020. 1

[11] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S Huang. Free-form image inpainting with gated convolution. *arXiv:1806.03589*, 2018. 2

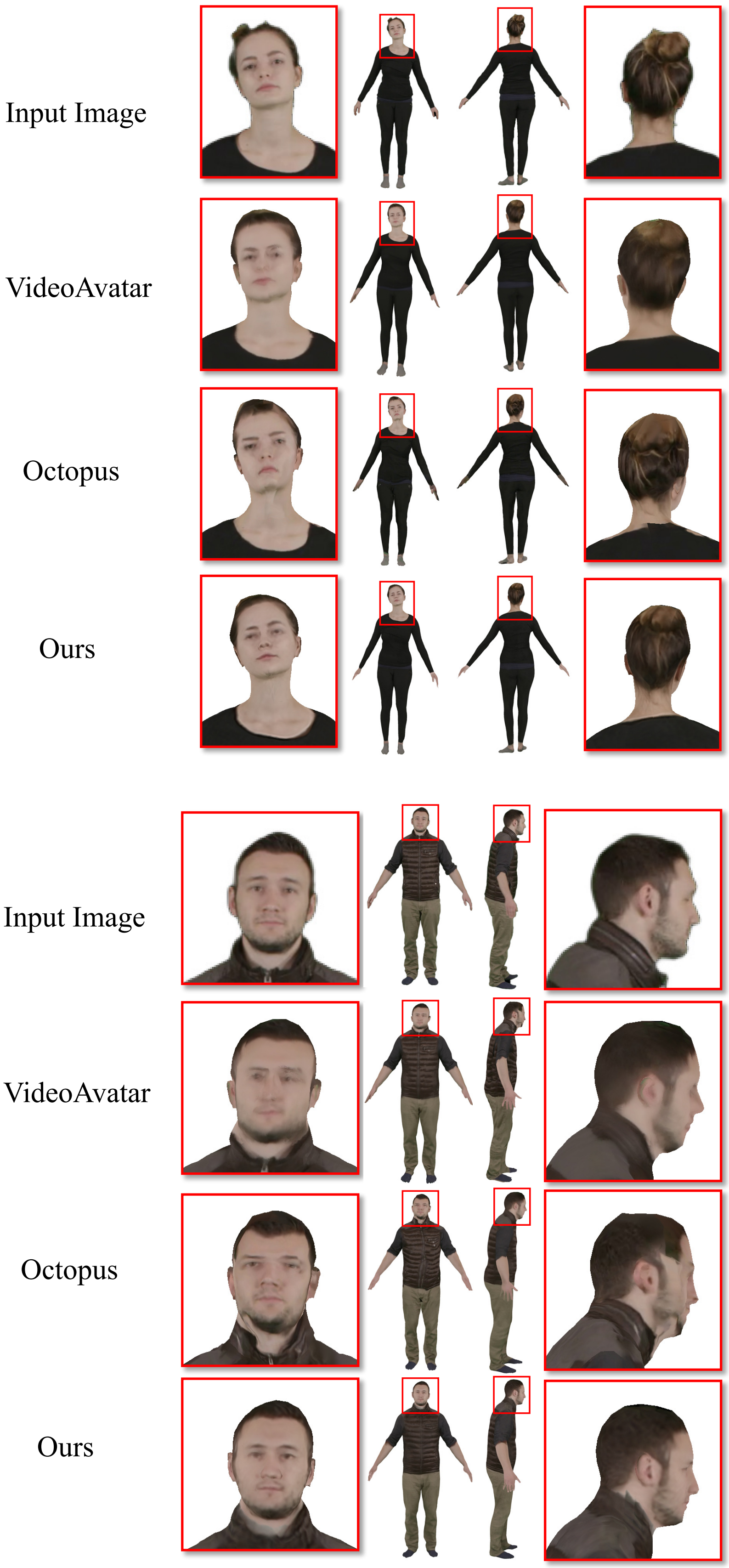

Figure 2. Reconstructed textured-avatars by VideoAvatar [2], Octopus [1] and ours on *People-Snapshot* [2].

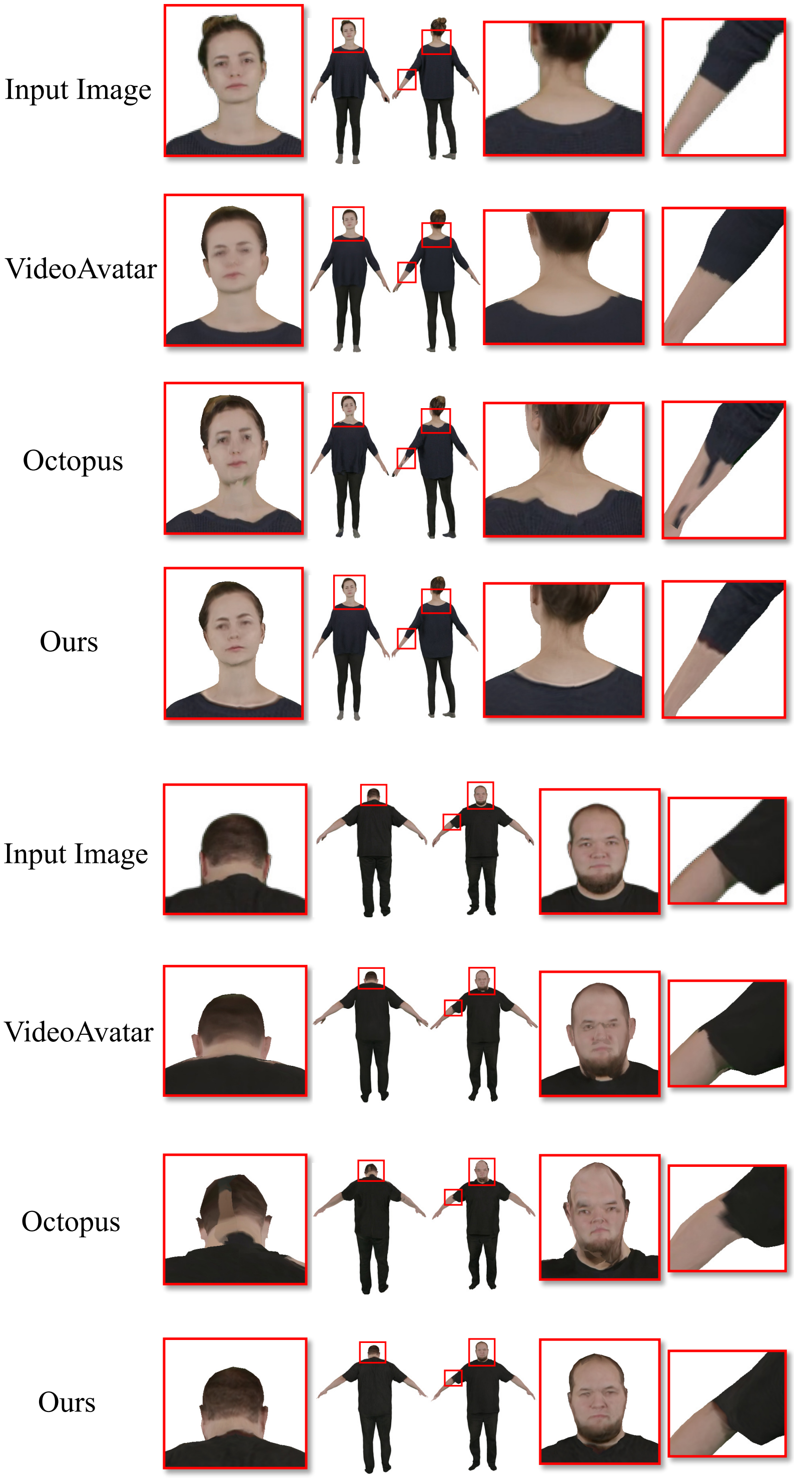

Figure 3. Reconstructed textured-avatars by VideoAvatar [2], Octopus [1] and ours on *People-Snapshot* [2].

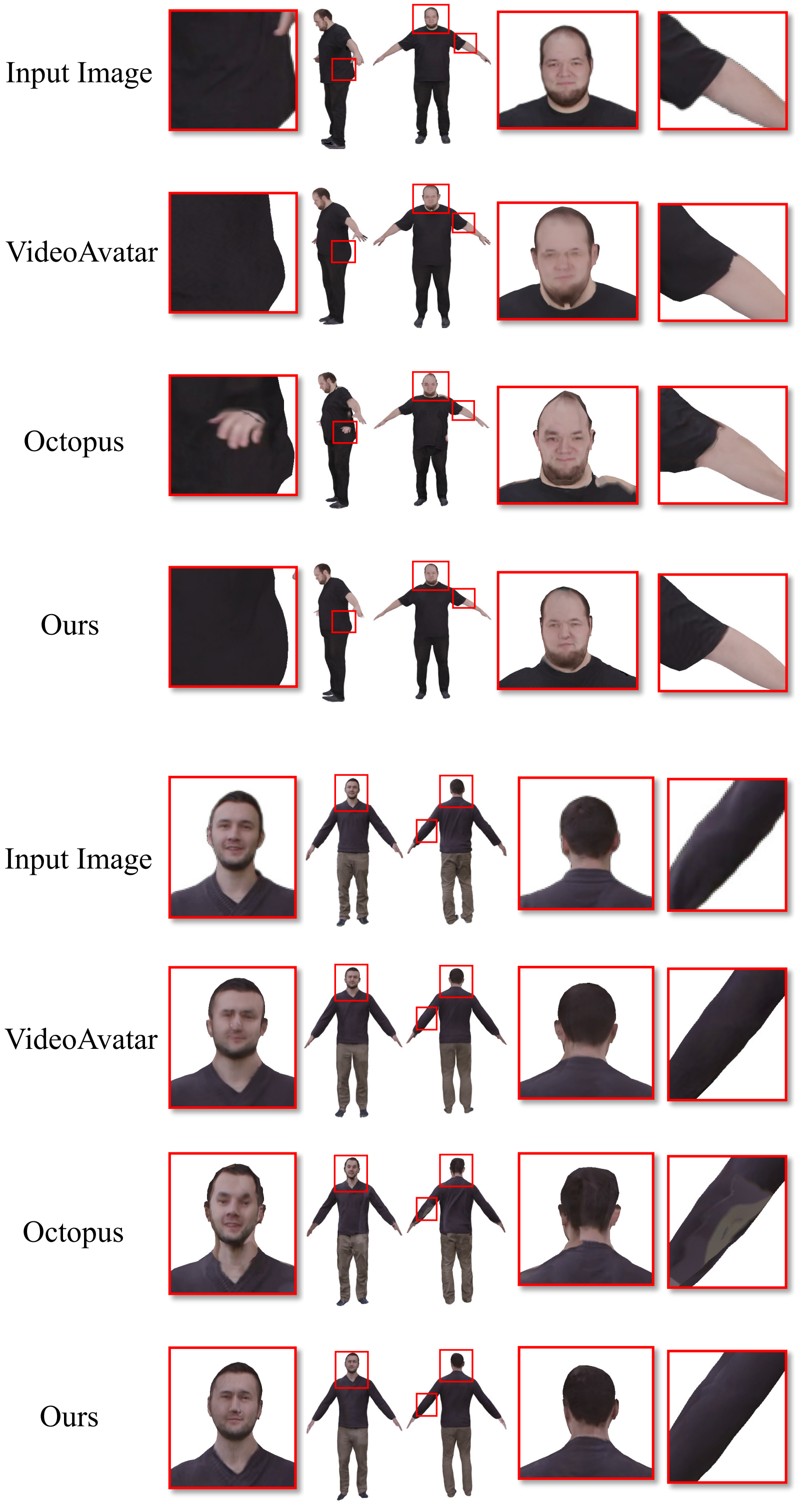

Figure 4. Reconstructed textured-avatars by VideoAvatar [2], Octopus [1] and ours on *People-Snapshot* [2].

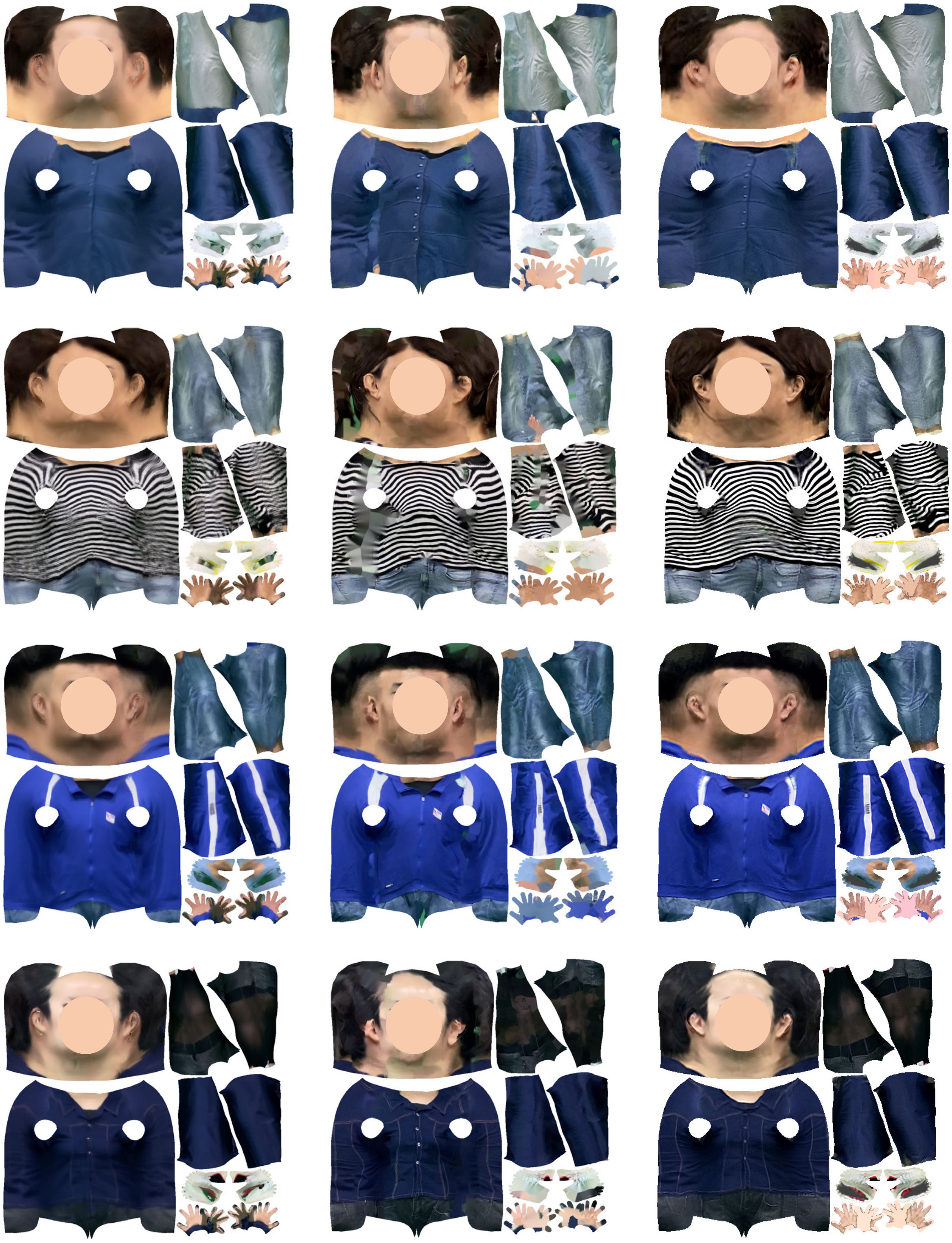

Figure 5. The texture maps estimated by VideoAvatar [2], Octopus [1] and ours.

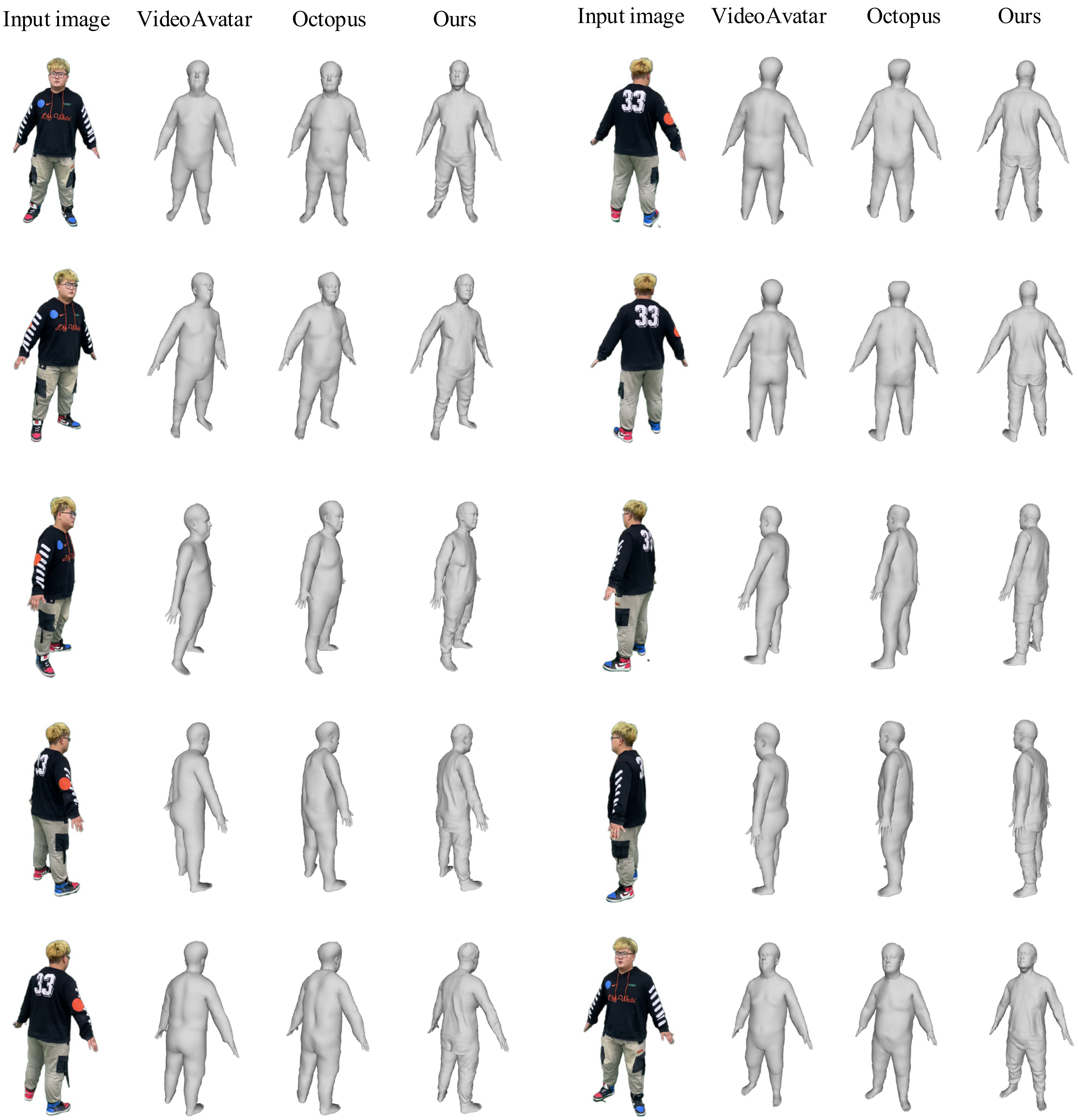

Figure 6. Reconstructed 3D geometries by VideoAvatar [2], Octopus [1] and our method.

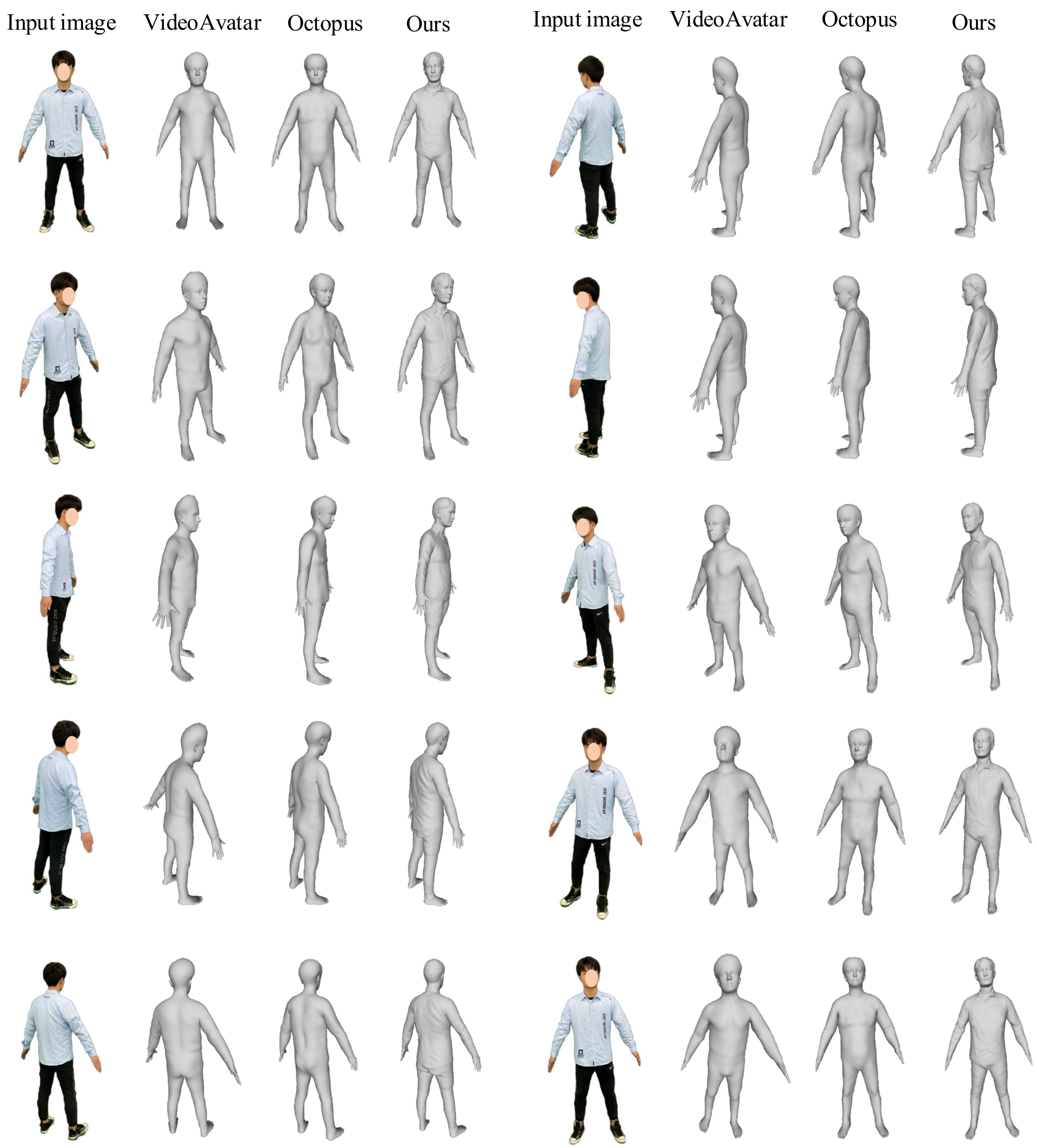

Figure 7. Reconstructed 3D geometries by VideoAvatar [2], Octopus [1] and our method.

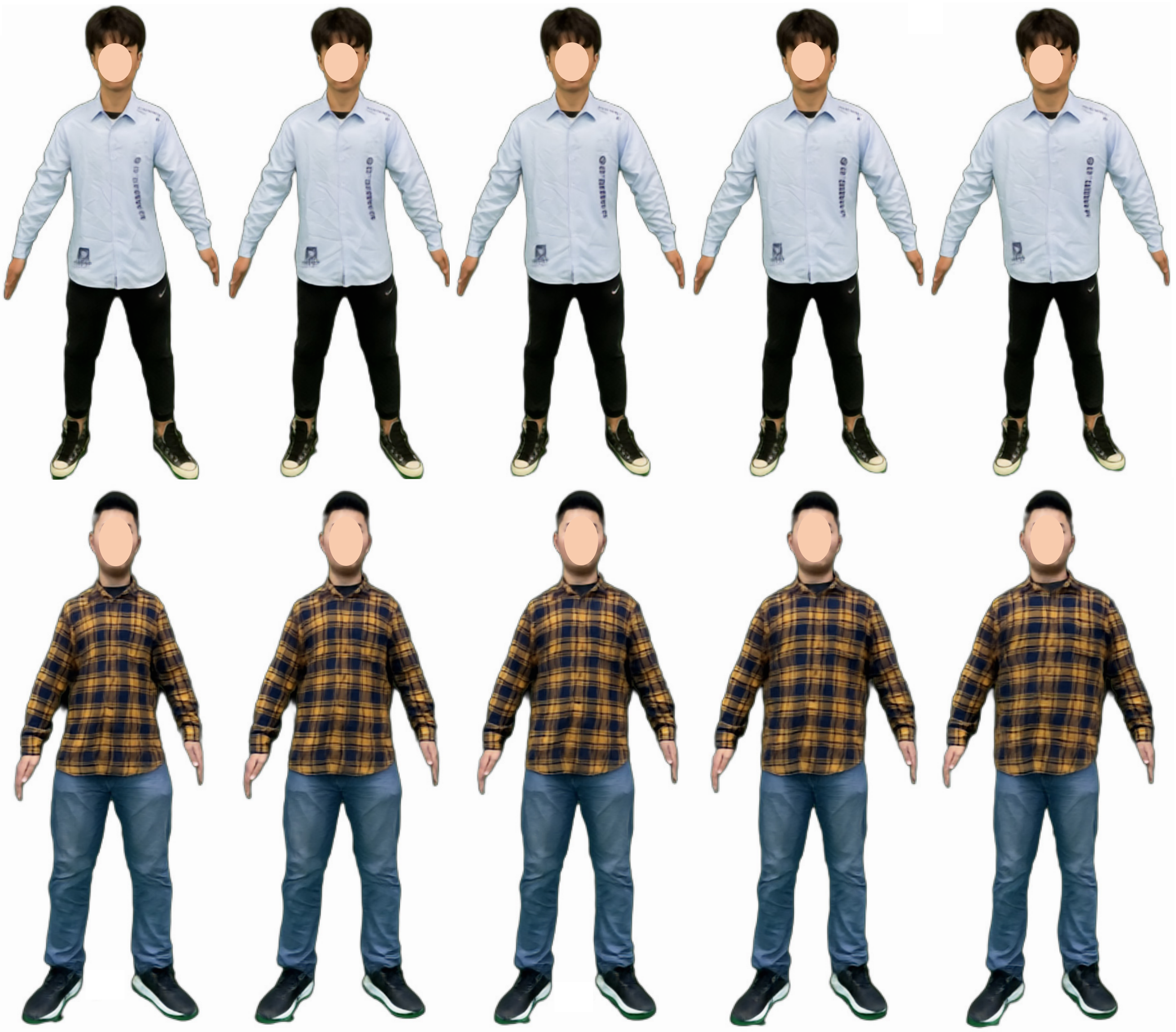

Figure 8. The results of shape editing. From left to right, we show the results of person changing from thinness to fatness.

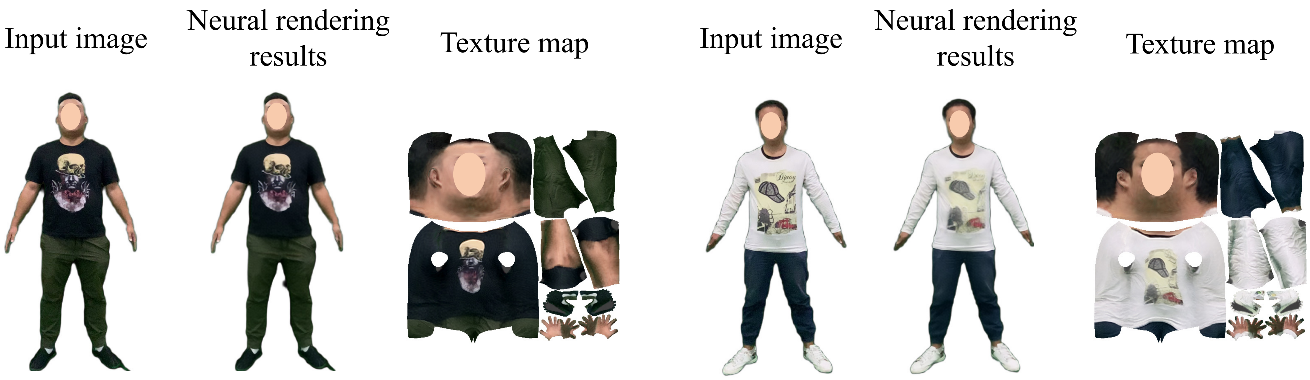

Figure 9. Some examples of failure cases.