AvatarOne: Monocular 3D Human Animation Supplementary Material

Akash Karthikeyan[1] Robert Ren[1] Yash Kant[1] Igor Gilitschenski[1,2]
[1]University of Toronto [2]Vector Institute for AI

Figure 1. **AvatarOne's Reposing Capabilities.** We present AvatarOne's novel pose editing. We repose the generated 3D avatar with help of only SMPL skeletons obtained from text prompts as in [6]

A. Network Architecture

We present the NeRF Network architecture in Fig. 2 which dynamically models the human surface and texture in canonical space. We use a occupancy grid of $112 \times 112 \times 112$ to capture the canonical space and utilize a voxel grid with dimensions $64 \times 64 \times 64$ voxels to parameterize the skinning weights, similar to the architecture described in [1].

B. Implementation Details

The framework achieves satisfactory results within approximately 15 minutes and scores saturate under 30 minutes when run on an RTX 4090 GPU.

- **Warm-up stage:** We randomly sample 10000 points from the SMPL canonical mesh and update the parameters of $F_{\Theta_{surf}}$ and use the bone weight loss to optimize the $F_{\Theta_{lbs}}$. The $F_{\Theta_{surf}}$ MLP uses the multi-hash encoding as in InstantNGP [3].

- **Stage I:** We employ randomized ray sampling on a downsampled-resolution of 512×512 image, initially selecting 4096 rays. These rays are subsequently thresholded based on their distance to the posed skeleton.

- **Stage II:** We freeze the $F_{\Theta_{lbs}}$ module and shift to a patch based sampling, we randomly sample 3 patches of 32×32 and additionally introduce the loss L_{LPIPS} to ensure that the model captures high-level features so that the output images are not overly muddied.

Pre-processing: We follow similar pre-processing steps as in TAVA [2] to cluster poses based on joint information.

- val_{pose}^{ood} : encompasses the most varied pose sequences and serves as an out-of-distribution validation set

- val_{pose}^{ind} : consist of new poses considered to be *in distribution* to train split.

- val_{pose}^{view} : contains data with the same poses as the training set but captured from different camera angles

C. Algorithm

In this section, we provide the pseudocode of the warmup and training stages of AvatarOne in Algorithm 1.

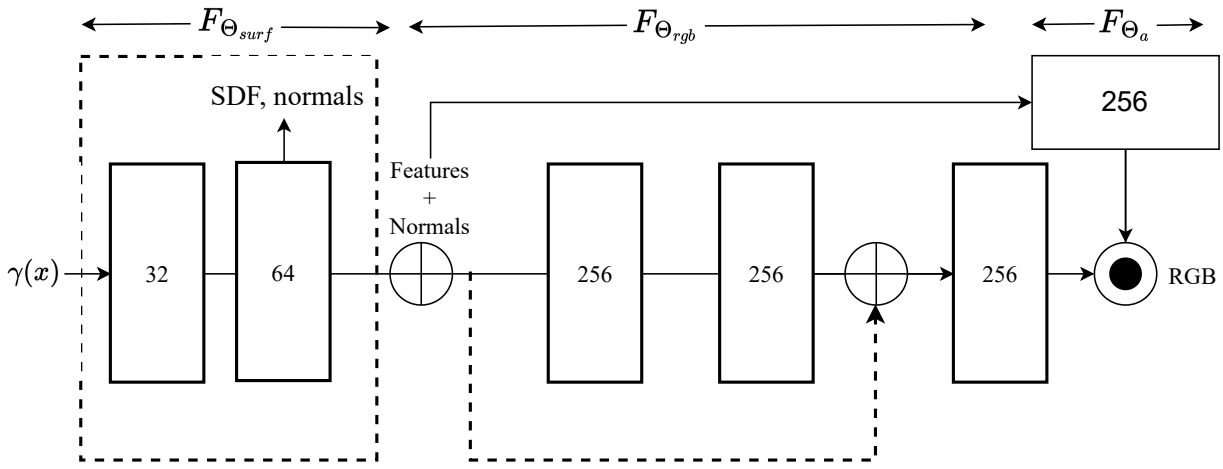

Figure 2. **Network architecture.** Given the positional encoding of canonical points $\gamma(x_c)$ based on [3], the network outputs the SDF, color, and normals. The input dimensions are indicated by the numbers in each block. ReLU activations for linear layers, except for the output layers of color and density, are excluded.

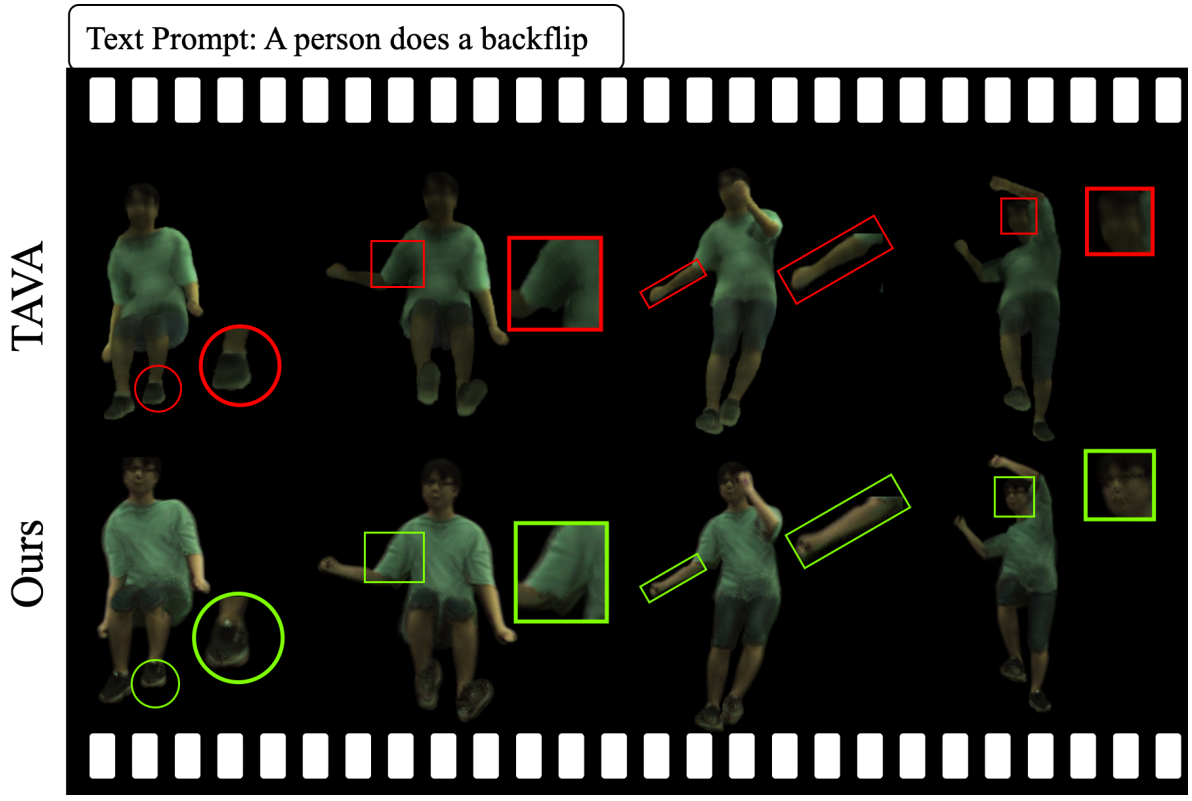

Figure 3. **Text-driven Novel pose.** AvatarOne provides detailed consistent results with all body parts intact

D. Further Analysis

In this section, we closely examine image results from our model and other baselines. Additional video demonstration of our model's output can be found in the video attached in the supplementary package.

D.1. Text-driven Novel Pose Rendering

We compare our model's novel pose synthesis capabilities with TAVA [2]. We produce novel pose rendering based on challenging poses generated by Motion-Diffusion-Model [6]. In Fig. 3, it can be clearly observed that

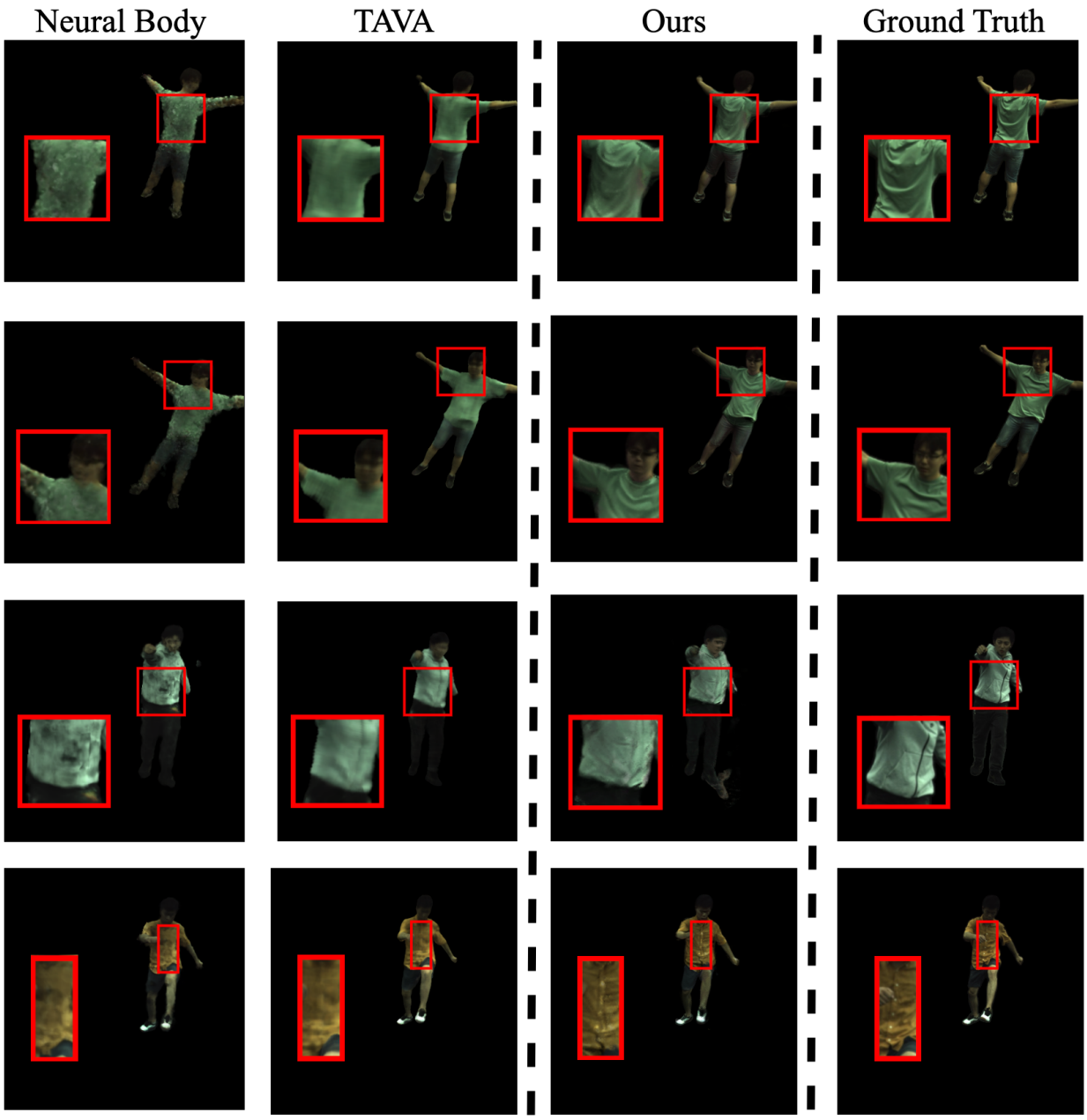

Figure 4. **Baseline Comparisons.** Novel pose comparisions on Neural Body [4] and TAVA [2]

AvatarOne successfully reconstructs intricate details of different human body parts such as the white stickers on the shoes, and visible glasses on the person's face. On the other hand, the novel pose results from TAVA is overly smoothened, and doesn't provide meaningful reconstruction at a finer level.

D.2. Novel View Rendering Comparison

We further compare our model's novel view rendering abilities with Neural Body [5] and TAVA [2]. In each comparison, we focus on different body parts in order to get a comprehensive understanding of all the models' performance. AvatarOne captures the wrinkles on clothing with far more details than Neural Body and TAVA. In addition for the subject in the last row, AvatarOne even successfully captures the buttons on the orange shirt, which is a detailed reconstruction neither Neural Body nor TAVA were able to achieve.

D.3. Additional Ablation Studies

In addition to the ablation studies discussed in the paper, we point out another key method in our model. We observe

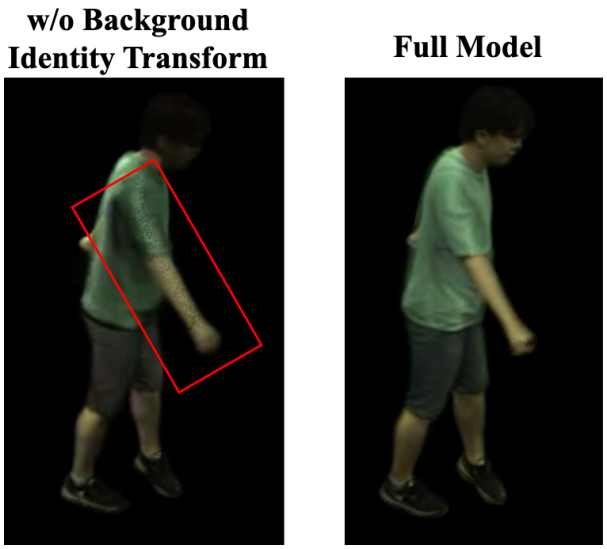

Figure 5. **Ablation Experiment.** We employ an identity transform as a control mechanism to prevent background and empty rays from being deformed during the skinning process.

that the human reconstruction contains many black floaters as can be seen in the left of Fig. 5. This is due to background and empty rays being wrongfully deformed in the skinning process, while they should not interact with the deformation process. By adopting a background identity transform, we largely filter out empty rays from being considered during skinning, thereby significantly improving the quality of our reconstruction.

E. Potential Social Impacts

Given that our model recovers realistic human avatars from monocular video input. Abuse, and misuse of 3D animatable avatars could lead to severe identity theft. Therefore, it is crucial to constrain the usage of such models and ensure that they are used legally and ethically.

F. Notations

In Tab. 1, we list the important variables used in this paper, along with their descriptions.

Algorithm 1 Warm-up and Traning Stages for AvatarOne

Inputs:
$\mathbf{x} \in$ set of points on SMPL canonical mesh
n_{smpl} surface normal of the SMPL mesh
$\bar{\mathbf{x}}_c$ points on the bones
$\boldsymbol{B}$ bone transformations
$\mathbf{w}_v$ skinning weights parameterised by $F_{\Theta_{lbs}}$ MLP
$\mathbf{d}, \mathbf{o}$ denote ray directions and origins, ray samples (x_o)

Warm-up: $F_{\Theta_{surf}}, F_{\Theta_{lbs}}$

for $\mathbf{x} \in \{\mathcal{S}_{\text{smpl}}\}$ **do** $\mathcal{S}_{\text{smpl}} = \{\, \mathbf{x} \mid F_{\Theta_{surf}}(\mathbf{x}) = 0 \,\}$.
subject to, $\mathcal{L}_{\text{n}}^i, \mathcal{L}_{\text{eik}}^i$
  for $k \leftarrow 0, n$ **do**
    $F_{\Theta_{lbs}}(\bar{\mathbf{x}}_c) \rightarrow w_1, ..., w_{n_b}$ subject to, $\mathcal{L}_w$
    $w_1, ..., w_{n_b} \leftarrow \mathbf{w}_{\sigma_w}(\mathbf{x}_v)$ ▷ $\mathbf{x}_v$ voxel weight field
  end for
end for

Training: $F_{\Theta_{surf}}, F_{\Theta_{lbs}}, F_{\Theta_{rgb}}, F_{\Theta_a}$

Define filter $f(\mathbf{d}, \mathbf{o})$ as:

$$f(\mathbf{d}, \mathbf{o}) = \begin{cases} 0 & \text{if dist}(\mathbf{d}, \mathbf{o}, B) < 0.3 \\ \mathbf{d}, \mathbf{o} & \text{otherwise} \end{cases}$$

For each $\mathbf{d}, \mathbf{o}$, apply $f(\mathbf{d}, \mathbf{o}) \rightarrow t_0, t_1$
For each $t_0, t_1 \rightarrow \mathbf{x_o}$, ▷ Get points from ray indicies
$\mathbf{T}_v \leftarrow \sum_{i=1}^{n_b} w_i \cdot \boldsymbol{B}_i$
for $\mathbf{x_o}, \mathbf{x_c}^*, \tilde{\mathbf{J}}^*$ **in parallel do** ▷ Initialize $\mathbf{x_c}^{1,2...9}$
  for $k \leftarrow 0, n$ **do**
    $\mathbf{T} \leftarrow \text{trilerp}(\mathbf{x}^k, \{\mathbf{T}_v\})$
    $\mathbf{x}^{k+1}, \tilde{\mathbf{J}}^{k+1} \leftarrow \text{broyden}(\mathbf{x}^k, \tilde{\mathbf{J}}^k, \mathbf{T}, \mathbf{x}')$
  end for
end for
$\mathbf{x_c}, \text{feat}, n_c = \arg\min\left(|F_{\Theta_{surf}}(\mathbf{x_c}^*)|\right)$ ▷ iso-surface
$n_c \rightarrow n_o$ ▷
$c, h \leftarrow F_{\Theta_{rgb}}(\text{feat}, n_o)$; $F_{\Theta_a} \leftarrow F_{\Theta_a}(h, B)$; $c = c \cdot ao$
subject to: $\mathcal{L}_{rgb}^i, \mathcal{L}_{mask}^i, \mathcal{L}_{\text{sparse}}^i, \mathcal{L}_{\text{eik}}^i, \mathcal{L}_{\text{n}}^i, \mathcal{L}_w$
return: $\{\mathbf{x_c}, c\}$

References

[1] Xu Chen, Tianjian Jiang, Jie Song, Max Rietmann, Andreas Geiger, Michael J. Black, and Otmar Hilliges. Fast-snarf: A fast deformer for articulated neural fields. *Pattern Analysis and Machine Intelligence (PAMI)*, 2023. 1

[2] Ruilong Li, Julian Tanke, Minh Vo, Michael Zollhofer, Jurgen Gall, Angjoo Kanazawa, and Christoph Lassner. Tava: Template-free animatable volumetric actors. *arXiv preprint arXiv:2206.08929*, 2022. 1, 2, 3

[3] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *arXiv:2201.05989*, Jan. 2022. 1, 2

[4] 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. *CVPR*, 2021. 3

[5] 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 *Proc. of CVPR*, 2021. 3

[6] Guy Tevet, Sigal Raab, Brian Gordon, Yoni Shafir, Daniel Cohen-or, and Amit Haim Bermano. Human motion diffusion model. In *The Eleventh International Conference on Learning Representations*, 2023. 1, 2

Table 1. **Notations.** A list of the important variables used in the paper.

Symbol	Description
$F_{\Theta_{surf}}$	MLP for point features in canonical space
$F_{\Theta_{rgb}}$	MLP for RGB and intermediate activation
F_{Θ_a}	MLP for ambient occlusion value
$F_{\Theta_{lbs}}$	MLP for skinning weights
$\mathbf{x_c}$	A point in canonical space
$\mathbf{x_o}$	A point along a ray in world space
$\mathbf{w}_v$	Low resolution voxel grid for skinning weight
$\mathcal{S}$	Canonical SMPL surface
$\mathbf{B}$	Bone transformations
$\mathbf{T}$	Ray transmittance for volume rendering
α	Ray opacity
$\mathbf{n}$	Surface normals
c	Color
$\mathcal{L}_{\text{rgb}}$	Pixel RGB reconstruction loss
$\mathcal{L}_{\text{mask}}$	Mask loss
$\mathcal{L}_{\text{eik}}$	Eikonal loss in canonical space
$\mathcal{L}_{\text{n}}$	Normal consistency loss with SMPL surface
$\mathcal{L}_{\text{w}}$	Bone weight loss
$\mathcal{L}_{\text{sparse}}$	Opacity sparseness regularization
$\mathcal{L}_{\text{LPIPS}}$	Perceptual Similarity loss between patches of image