Supplementary Material for CtrlAvatar: Controllable Avatars Generation via Disentangled Invertible Networks

Anonymous submission

1 Overview

In this Supplementary Material, we first detail the dataset processing flow that we use (see Sec. 2). Second, we describe the implementation details of our approach (see Sec. 3). Next, we illustrate the processing workflow for editing human cloths using our method (see Sec. 4). Finally, we describe the setup of the comparative methods and the more results of comparisons (see Sec. 5).

2 Dataset Details

In this section, we explain our dataset preparation from the original datasets (Shen et al. 2023; Ho et al. 2023).

SX-Humans To assess the efficacy of our method with limited data, we curate SX-Humans, a smaller scale dataset. We select 4 scanning actions from the complete action sequences of each subject which from the X-Humans dataset (Shen et al. 2023) at predetermined intervals. This process yields an average of 40 poses per subject for the training subset. For testing, we employ a set mirroring the complete action sequences from X-Humans to thoroughly evaluate the method's performance across different aspects of action reconstruction.

S-CustomHumans To enhance the diversity provided by the SX-Humans dataset, we meticulously select 10 subjects not contained in SX-Humans to create the S-CustomHumans subset from CustomHumans (Ho et al. 2023). For each subject within this subset, we designate 4 scans for the training set and 2 scans for the testing set. This approach ensures a comprehensive data coverage and accuracy in performance testing.

3 Implementation Details

In this section, we provide more training information about the network architecture and training detail.

3.1 Network Architectures

We begin by describing the structure of the IDN model used in Invertible Networks, as shown in Fig. 1. The orange segment represents f_{cond}, which extracts conditional features, while the green segment represents f_{delta}, which computes the offsets. Each linear layer, except the output layer, uses the softplus activation function. The output layer does not employ an activation function. For the weight field f_w and

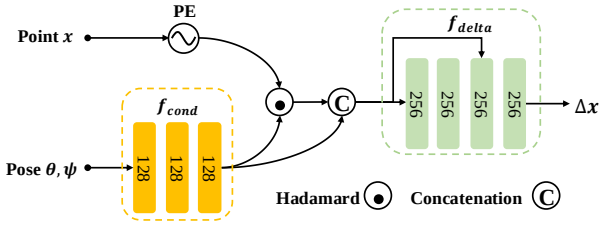

Figure 1: **Architecture of the IDN model.**

the occupancy field f_{occ}, we follow the approach used in XAvatar (Shen et al. 2023). The color field f_{color} is modeled using an MLP composed of 6 linear layers, each with 256 units. And the outputs are assigned to the range $[0, 1]$ using the sigmoid activation function. Finally, we implement the differentiable renderer $\mathcal{G}$ using PyTorch3D (Ravi et al. 2020). Starting with the canonical human mesh, oriented with the face forward, we rotate it clockwise by 90 degrees to obtain the right, back, and left camera viewpoints sequentially.

3.2 Training Detail

During the geometry training, we use a position encoding frequency of 4. The features obtained from this encoding are concatenated with the original points, resulting in final points position features dimension of $p = 27$. For color training, we use a position encoding frequency of 10, which produces a final point feature dimension of 126. In the multi-view constraint, we render the human body mesh into 800×800 images to calculate the loss, with the loss weights set to $\lambda_{\text{L1}} = 100$, $l_{\text{ssim}} = 10$, and $l_{\text{pre}} = 20$.

All modules are implemented using PyTorch and trained with the Adam optimizer (Kingma and Ba 2014). Both training and inference are conducted on a single NVIDIA RTX 4090 GPU, and all time measurements are based on this device. For each subject, our method requires 3-4 hours to train the occupancy field and an additional 2-3 hours to train the color field.

4 More Details of Editing the Human Avatars

In this section, we detail the process of editing the Human avatar. Benefiting from the use of 2D images as a constraint

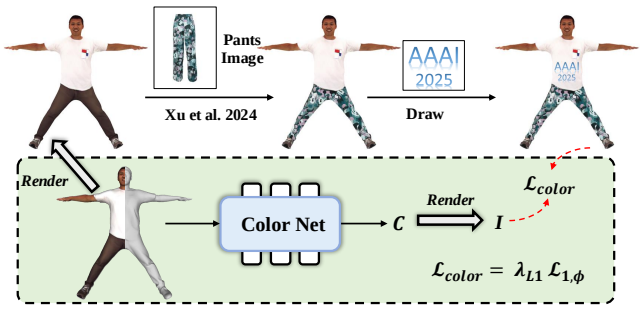

Figure 2: **Process of Editing Avatar.** Our method allows for editing the color of a 3D avatar's clothing by simply modifying a 2D image.

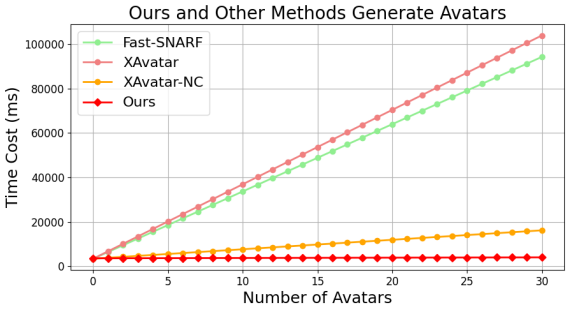

Figure 3: **Time Cost of Generate Avatars.** We measure the time cost in generate 30 different avatars. The results verify that the time cost by our method to generate a new avatar is significantly shorter than that required by other methods.

for texture, our model enables straightforward manipulation of Avatar colors through image editing.

Specifically, we render a pre-trained human avatar from multiple perspectives and then edit these 2D images to modify the original appearance. For example, color changes can be made directly or by using a 2D try-on technique,(Xu et al. 2024). These modified images then serve as ground truth in training the color network, as illustrated in Fig. 2. Since the 2D images originate from the initial avatar, they are perfectly aligned with the avatar's geometry, allowing us to update the colors using only L1 loss $\mathcal{L}_{1,\phi}$ as a constraint. Notably, although we use a T-pose and modifications from the front views in Fig. 2, the consistency of our geometric topology enables the use of any pose and camera parameters for fitting the color network. Additionally, we do not update the geometry and deformation networks during color updates, ensuring that the same network continues to control the avatar's deformation even after editing.

5 More Comparisons with State-of-the-art Methods

In this section, we first explain the setup of the comparison methods. We then present the comparative results of our method against other approaches in different subjects.

Setup of the comparison methods. To ensure a fair comparison, we benchmark our method against state-of-the-art approaches that can reconstruct human meshes such as XAvatar (Shen et al. 2023), Fast-SNARF (Chen et al. 2023). For Havefun (Yang et al. 2024), which relies on images from multiple viewpoints, we select 8 viewpoints to optimize reconstruction quality. We also use the method provided in the original code to map the rendered image to the human body to obtain the texture mapping. For Editable-Humans (Ho et al. 2023), we train the entire CustomHumans dataset using the published code. During testing, we focus exclusively on the subjects contained in the S-CustomHumans.

To be noted that, our comparisons focus solely on methods capable of extracting explicit meshes and textures, even though most of the Nerf-based (Weng et al. 2022) and 3DGS-based (Li et al. 2024) methods could provide a good appearance, our points-based methods have more advantages in the application.

More result. First, we present the results of human reconstruction across different subjects in Fig. 4 and the results of texture in Fig. 5. These results collectively highlight our method's effectiveness in generating human avatars with accurate geometry and detailed texture.

Next, we compare the average generation time of our method with other approaches, as illustrated in Fig. 3. The results demonstrate that our method significantly reduces the time required for avatar generation, especially as the number of avatars increases.

Additionally, we provide visualization results from the texture module ablation experiments, shown in Fig.6, further validating the robustness of our approach. Finally, we present the results of editing avatars with various clothing options and driving them by SMPL-X parameters(Pavlakos et al. 2019), as depicted in Fig.7. More visual comparison results can be found in the supplemental video.

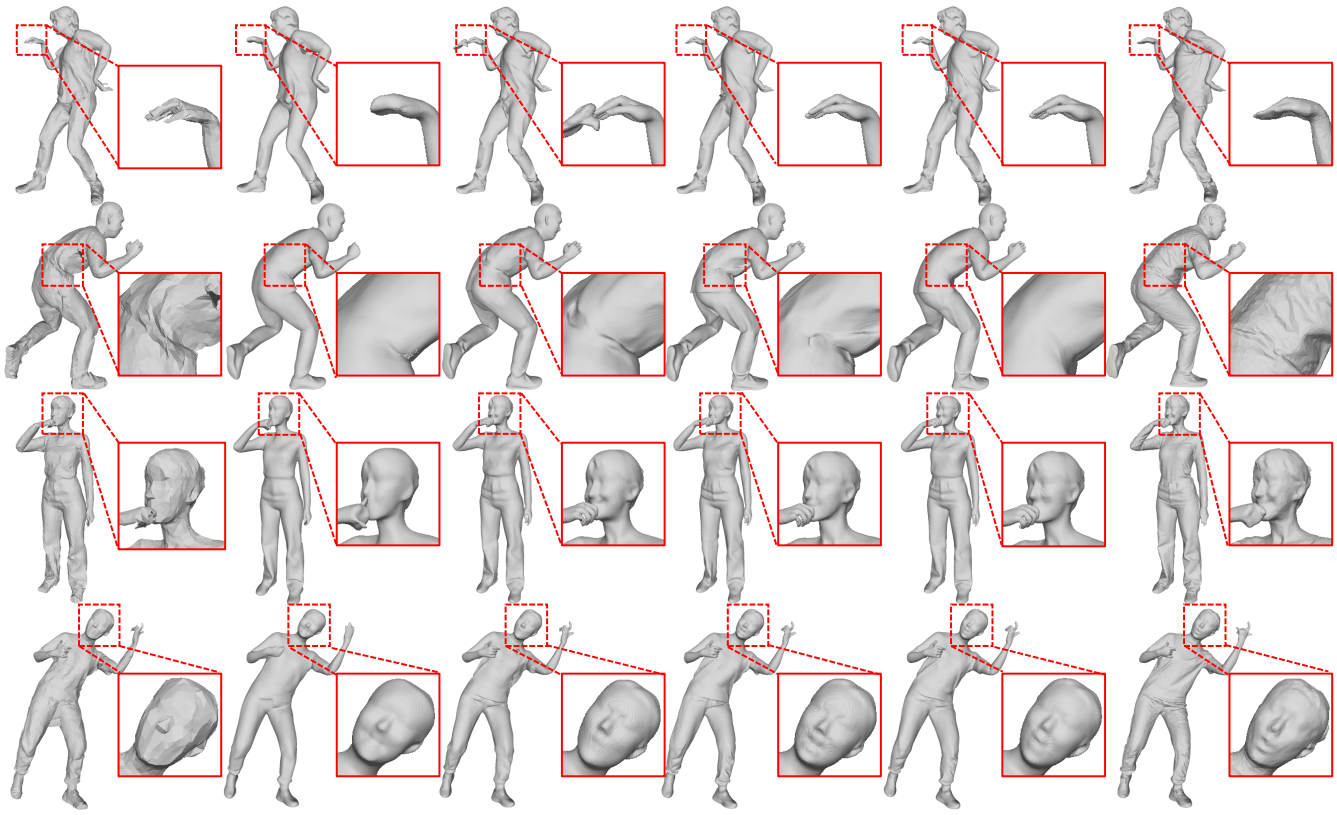

Figure 4: **Qualitative Results on the SX-Humans.** Our method avoids artifacts and captures richer details of the human body and face.

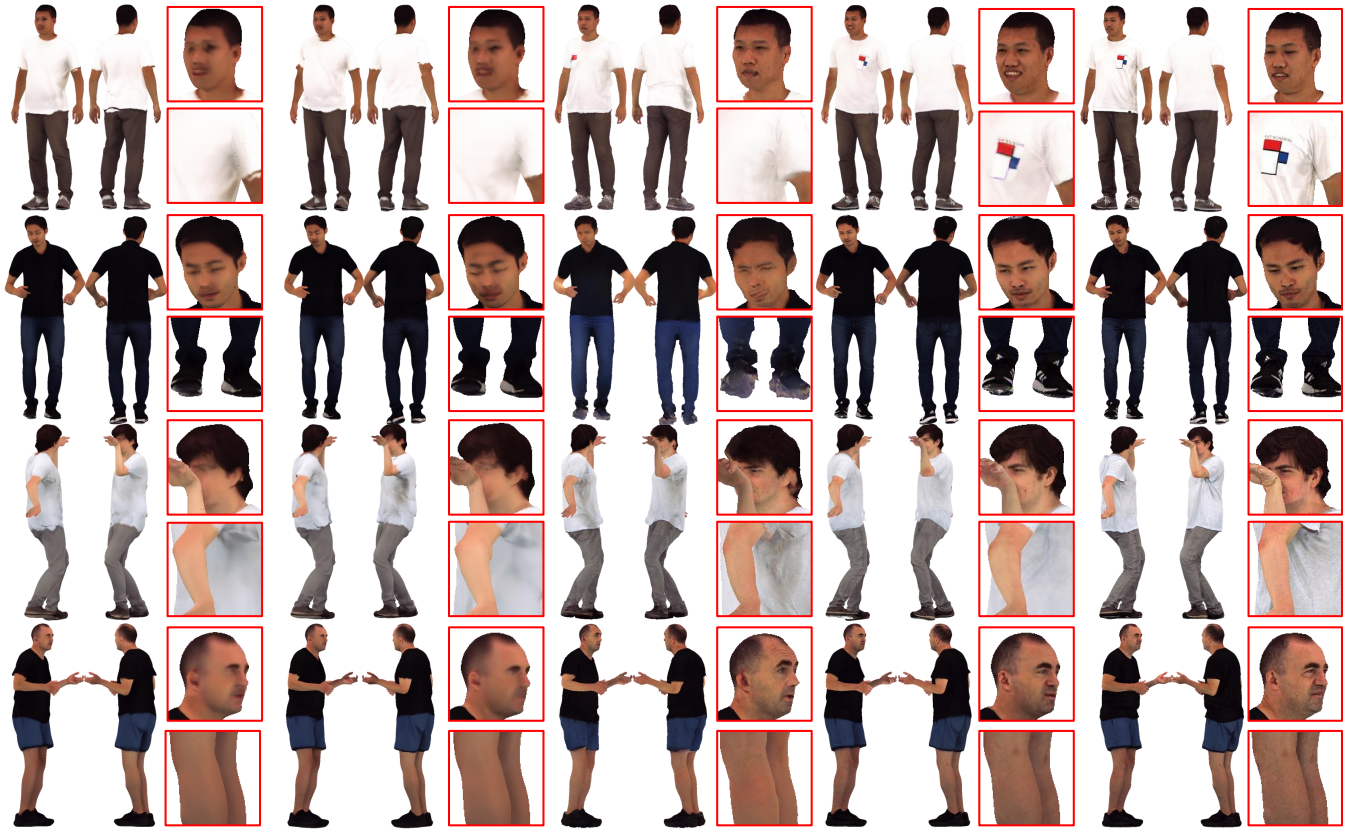

Figure 5: **Qualitative results on SX-Humans with Texture.** The results verify that our method performs best in texture appearance.

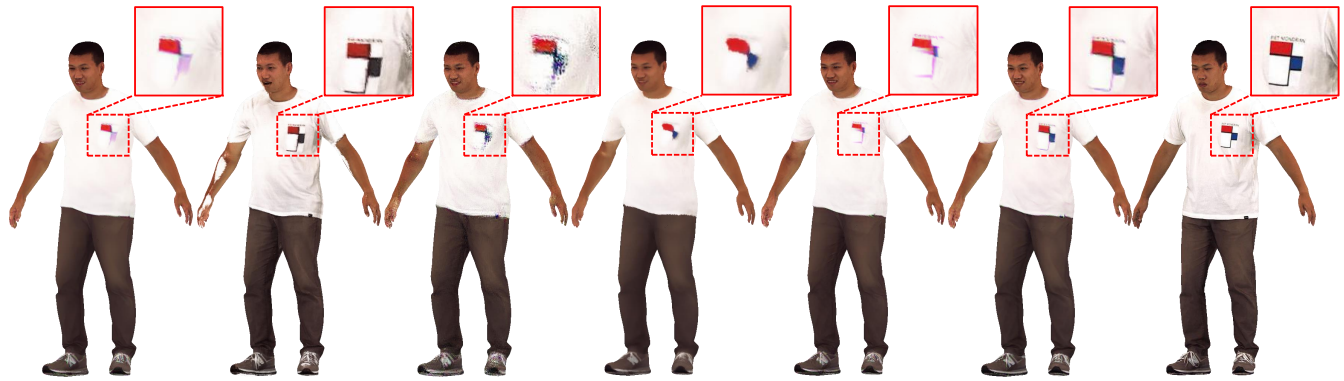

Figure 6: **Ablation Study for Texture Model.** The results verify the validity of our texture module design, enabling more realistic cloths textures.

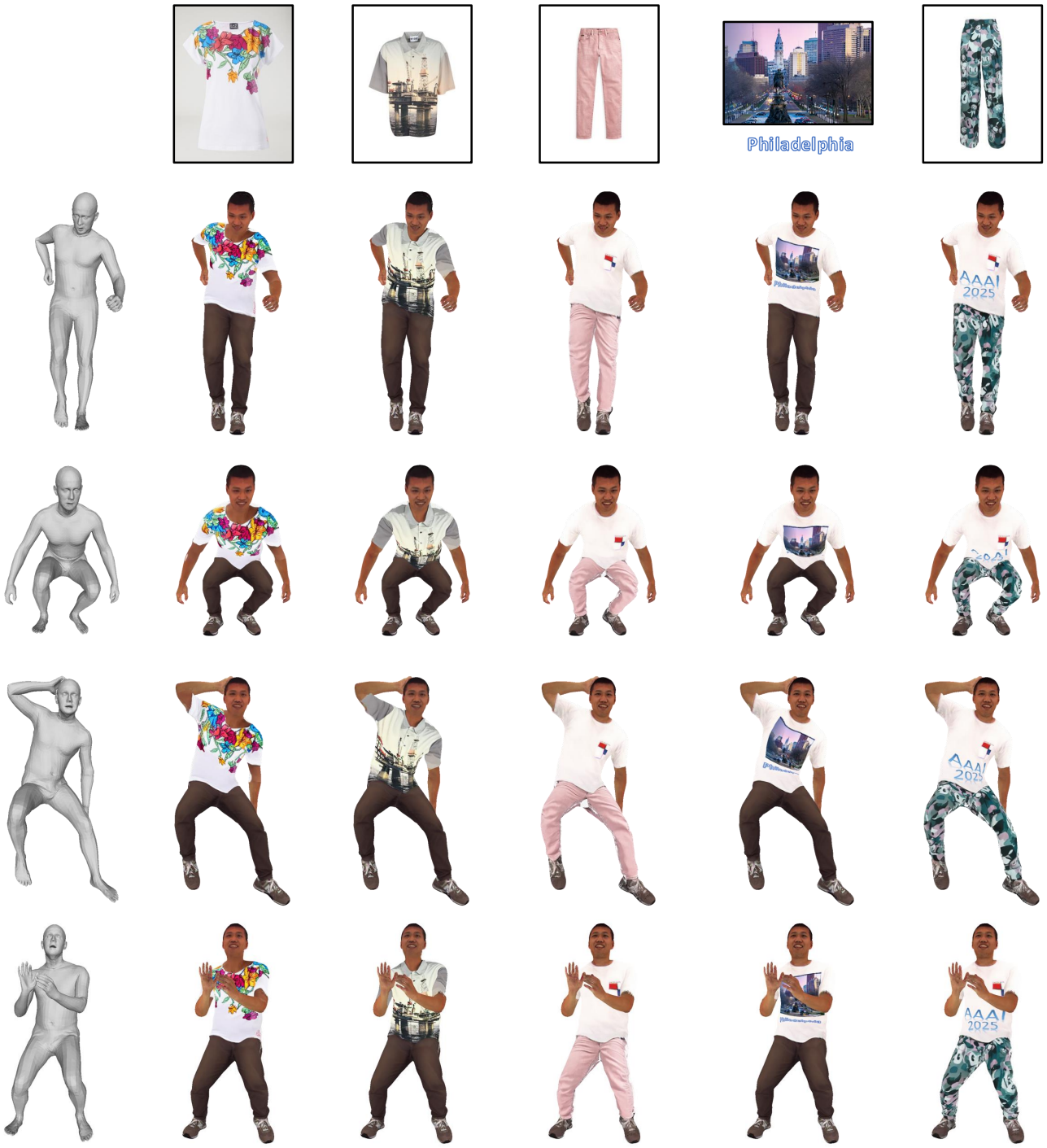

Figure 7: **Qualitative Results on Avatar Editing.** Our method supports editable textures and easily driven by SMPL-X parameters.

References

Chen, X.; Jiang, T.; Song, J.; Rietmann, M.; Geiger, A.; Black, M. J.; and Hilliges, O. 2023. Fast-SNARF: A fast deformer for articulated neural fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(10): 11796–11809.

Ho, H.-I.; Xue, L.; Song, J.; and Hilliges, O. 2023. Learning locally editable virtual humans. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 21024–21035.

Kingma, D. P.; and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Li, Z.; Zheng, Z.; Wang, L.; and Liu, Y. 2024. Animatable gaussians: Learning pose-dependent gaussian maps for high-fidelity human avatar modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 19711–19722.

Pavlakos, G.; Choutas, V.; Ghorbani, N.; Bolkart, T.; Osman, A. A.; Tzionas, D.; and Black, M. J. 2019. Expressive body capture: 3d hands, face, and body from a single image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 10975–10985.

Ravi, N.; Reizenstein, J.; Novotny, D.; Gordon, T.; Lo, W.-Y.; Johnson, J.; and Gkioxari, G. 2020. Accelerating 3D Deep Learning with PyTorch3D. *arXiv:2007.08501*.

Shen, K.; Guo, C.; Kaufmann, M.; Zarate, J. J.; Valentin, J.; Song, J.; and Hilliges, O. 2023. X-avatar: Expressive human avatars. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 16911–16921.

Weng, C.-Y.; Curless, B.; Srinivasan, P. P.; Barron, J. T.; and Kemelmacher-Shlizerman, I. 2022. Humannerf: Free-viewpoint rendering of moving people from monocular video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 16210–16220.

Xu, Y.; Gu, T.; Chen, W.; and Chen, C. 2024. Ootdiffusion: Outfitting fusion based latent diffusion for controllable virtual try-on. *arXiv preprint arXiv:2403.01779*.

Yang, X.; Chen, X.; Gao, D.; Wang, S.; Han, X.; and Wang, B. 2024. Have-fun: Human avatar reconstruction from few-shot unconstrained images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 742–752.