Animatable and Relightable Gaussians for High-fidelity Human Avatar Modeling

Zhe Li*, Yipengjing Sun*, Zerong Zheng, Lizhen Wang, Shengping Zhang and Yebin Liu, *Member, IEEE*

https://animatable-gaussians.github.io/relight

***Abstract*—Modeling animatable human avatars from RGB videos is a long-standing and challenging problem. Recent works usually adopt MLP-based neural radiance fields (NeRF) to represent 3D humans, but it remains difficult for pure MLPs to regress pose-dependent garment details. To this end, we introduce Animatable Gaussians, a new avatar representation that leverages powerful 2D CNNs and 3D Gaussian splatting to create high-fidelity avatars. To associate 3D Gaussians with the animatable avatar, we learn a parametric template from the input videos, and then parameterize the template on two front & back canonical Gaussian maps where each pixel represents a 3D Gaussian. The learned template is adaptive to the wearing garments for modeling looser clothes like dresses. Such template-guided 2D parameterization enables us to employ a powerful StyleGAN-based CNN to learn the pose-dependent Gaussian maps for modeling detailed dynamic appearances. Furthermore, we introduce a pose projection strategy for better generalization given novel poses. To tackle the realistic relighting of animatable avatars, we introduce physically-based rendering into the avatar representation for decomposing avatar materials and environment illumination. Overall, our method can create lifelike avatars with dynamic, realistic, generalized and relightable appearances. Experiments show that our method outperforms other state-of-the-art approaches.**

***Index Terms*—Animatable avatar, human reconstruction, view synthesis, animation, relighting.**

I. Introduction

ANIMATABLE human avatar modeling, due to its potential value in holoportation, Metaverse, game and movie industries, has been a popular topic in computer vision for decades. However, how to effectively represent the human avatar is still a challenging problem.

Explicit representations, including both meshes and point clouds, are the prevailing choices, not just in human avatars but also throughout the entire 3D vision and graphics. However, previous explicit avatar representations [1]–[3] necessitate dense reconstructed meshes to model human geometry, thus limiting their applications in sparse-view video-based avatar modeling. In the past few years, with the rise of implicit representations, particularly neural radiance fields (NeRF) [4], many researchers tend to represent the 3D human as a pose-conditioned NeRF [5]–[8] to automatically learn a neural avatar from RGB videos. However, implicit representations require a coordinate-based MLP to regress a continuous field, suffering from the low-frequency spectral bias [9] of MLPs. Although many works aim to enhance the avatar representation by texture feature [6] or structured local NeRFs [7], they fail to produce satisfactory results because they still rely on an MLP to output the continuous implicit fields.

Recently, 3D Gaussian splatting [10], an explicit and efficient point-based representation, has been proposed for both high-fidelity rendering quality and real-time rendering speed. In contrary to implicit representations, explicit point-based representations have the potential to be parameterized on 2D maps [3], thus enabling us to employ more powerful 2D networks for modeling higher-fidelity avatars. Based on this observation, we present *Animatable Gaussians*, a new avatar representation that leverages 3D Gaussian splatting and powerful 2D CNNs for realistic avatar modeling. The first challenge lies in modeling general garments including long dresses. Inspired by point-based geometric avatars [11], [12], we first reconstruct a parametric template from the input videos and inherit the parameters of SMPL [13] by diffusing the skinning weights [11]. The character-specific template models the basic shapes of the wearing garments, even for long dresses. This allows us to animate 3D Gaussians in accordance with the template motion while avoiding density control in standard Gaussians [10], thereby ensuring the maintenance of a temporally consistent structure for 3D Gaussians in the following 2D parameterization.

For compatibility with 2D networks, it is necessary to parameterize the 3D template onto 2D maps. However, it remains challenging to unwrap the template with arbitrary topologies onto a unified and continuous UV space. Regarding that the front & back views almost cover the entire canonical human, we achieve the parameterization by orthogonally projecting the canonical template to both views. In each view, we define every pixel within the template mask as a 3D Gaussian, represented by its position, covariance, opacity, and color attributes, resulting in two front & back Gaussian maps. Similarly, given the driving pose, we obtain two posed position maps that serve as the pose conditions. Such a template-guided parameterization enables predicting pose-dependent Gaussian maps from the pose conditions through a powerful StyleGAN-based [14]–[16] conditional generator, StyleUNet [17].

Benefiting from the powerful 2D CNNs and explicit 3D Gaussian splatting, our method can faithfully reconstruct human details under training poses. On the other hand, given novel poses, the generalization of animatable avatars has not been extensively explored. Due to the data-driven nature of

* indicates equal contribution.

Zhe Li, Lizhen Wang and Yebin Liu are with Department of Automation, Tsinghua University, Beijing 100084, P.R.China. Yipengjing Sun and Shengping Zhang are with the School of Computer Science and Technology, Harbin Institute of Technology, Weihai 264209, P.R. China. Zerong Zheng is with NNKosmos Technology, Hangzhou, P.R.China.

Corresponding author: Yebin Liu and Shengping Zhang.

Fig. 1. Lifelike relightable and animatable avatars with *highly dynamic*, *realistic* and *generalized* details created by our method. We show synthesized results animated by the same pose under the capture environment and novel lights.

learning-based avatar modeling, direct extrapolation to poses out of distribution will certainly yield unsatisfactory results. Therefore, we propose to employ Principal Component Analysis (PCA) to project the driving pose signal, represented by the position maps, into the PCA space, facilitating reasonable interpolation within the distribution of training poses. Such a pose projection strategy realizes reasonable and high-quality synthesis for novel poses.

A preliminary version of this work has been published in CVPR 2024 [18], in which we propose a novel avatar representation for modeling realistic animatable human avatars from multi-view videos. However, the preliminary work [18] can only animate the avatar under illumination from the capture environment. In the current version, we further introduce physically-based rendering (PBR) [19], [20] into our avatar representation for creating both animatable and relightable human avatars. Specifically, besides the original attributes of 3D Gaussians, our model also predicts albedo, roughness, and light visibility to decompose the avatar materials and illumination of the capture environment. As a result, our model can produce realistic animation under novel illumination.

We extend the preliminary version [18] in the following ways. First, we introduce PBR into our avatar representation (Animatable Gaussians) for creating relightable human avatars. Second, we compare our method with concurrent works on 3D Gaussian splatting-based avatars, and our method outperforms them on the avatar quality. Third, we compare our method with other works on human performance relighting, and our method can produce more realistic human relighting.

In summary, our technical contributions are:

- Animatable Gaussians, a new avatar representation that introduces explicit 3D Gaussian splatting into avatar modeling to employ powerful 2D CNNs for creating lifelike avatars with high-fidelity pose-dependent dynamics.
- Template-guided parameterization that learns a character-specific template for general clothes like dresses, and parameterizes 3D Gaussians onto front & back Gaussian maps for compatibility with 2D networks.
- A simple yet effective pose projection strategy that employs PCA on the driving signal, promoting better generalization to novel poses.
- We introduce physically-based rendering into Animatable Gaussians for photorealistic relighting under novel illumination.

Overall, benefiting from these contributions, our method can create lifelike animatable and relightable avatars with *highly dynamic*, *realistic* and *generalized* appearances as shown in Fig. 1. The code is available at https://github.com/lizhe00/AnimatableGaussians, and earns more than 700 stars.

II. Related Work

A. Mesh-based Human Avatars

The polygon mesh is the most popular 3D representation for its compatibility with traditional rendering pipelines. To model animatable human avatars using meshes, early approaches propose to reconstruct a character-specific textured mesh and animate it by physical simulation [21], [22] or retrieval from a database [23]. Recently, researchers tend to utilize neural networks to model dynamic textures and motions. Bagautdinov *et al.* [1], Xiang *et al.* [2], [24] and Halimi *et al.* [25] reconstruct topology-consistent meshes from dense multi-view videos and learn the dynamic texture in a UV space. DDC [26] and HDHumans [27] learn the deformation parameterized by both skeletons and embedded graph [28] of a pre-scanned template. DELIFFAS [29] employs DDC as a deformable template and parameterizes the light field around the body onto double surfaces for fast synthesis. These mesh-based methods require dense reconstruction, non-rigid tracking, or pre-scanned templates for representing dynamic humans. Besides, some works optimize the non-rigid deformation upon SMPL [13] from a monocular RGB [30]–[32] or RGB-D [33], [34] video, but the avatar quality is limited by the SMPL+D representation.

B. Implicit Function-based Human Avatars

Implicit function is a coordinate-based function, usually represented by an MLP, that outputs a continuous field, e.g., signed distance function (SDF) [35], [36], occupancy [37], and radiance (NeRF) [5] fields. In geometric avatar modeling, many works represent the human avatar as pose-conditioned SDF [38]–[42] or occupancy [37], [43]–[47] fields learned from human scans or depth sequences. In contrast, NeRF containing a density and color field is widely used in textured avatar modeling [48]–[56] because of its good differentiable property. Animatable NeRF [5] introduces SMPL deformation

into NeRF for animatable human modeling. Neural Actor [6] and UV volumes [57] parameterize 3D humans on SMPL or DensePose [58] UV space, thus limiting modeling loose clothes far from the human body. SLRF [7] defines local NeRF around sampled nodes upon SMPL and learns the pose-dependent dynamics in the local space. TAVA [8] models the human or animal deformation using only 3D skeletons without the requirement of a parametric model. ARAH [59] represents the avatar geometry as SDF and adopts SDF-based volume rendering [60], [61] for learning more plausible geometry from RGB videos. DANBO [62] employs GNNs to learn the part-based pose feature. Li *et al.* [63] introduce a learnable pose vocabulary to learn higher-frequency pose conditions for the conditional NeRF. Besides the body avatar, TotalSelfScan [64], X-Avatar [65] and AvatarReX [66] propose compositional full-body avatars for expressive control of the human body, hands and face. However, the implicit function-based methods usually adopt pure MLPs to represent the human avatar, yielding smooth or blurry quality due to the low-frequency bias of MLPs [9]. What's worse, the rendering speed of these methods is usually slow because rendering from implicit fields requires dense sampling along a ray.

C. Point-based Human Avatars

Point cloud is also a powerful and popular representation in human avatar modeling. Given 3D scans of a character, SCALE [67] and POP [3] learn the non-rigid deformation of dense points on SMPL UV maps to represent the dynamic garment wrinkles. FITE [11] and CloSET [68] extract pose features from projective maps or PointNet [69], [70] to avoid discontinuity on the UV map. SKiRT [12] and FITE learn a coarse template from the input scans and utilize learned or diffused skinning weights to animate loose clothes. Prokudin *et al.* [71] propose dynamic point fields for general dynamic reconstruction. This work and NPC [72] show results on avatars created from RGB videos using Point-NeRF [73]. However, applying Point-NeRF to avatar modeling still relies on a low-frequency coordinate-based MLP, struggling with the same problems in Sec. II-B. On the other hand, point-based rendering via splatting [74]–[82] offers another probability for animatable avatar modeling. PointAvatar [83] learns a canonical point cloud and deformation field to model head avatars from a monocular video via PyTorch3D's [84] differentiable point renderer.

Recently, 3D Gaussian splatting [10] (3DGS), an efficient differentiable point-based rendering method, has been proposed for real-time photo-realistic scene rendering. Along with extending 3DGS to dynamic scene modeling [85]–[88], many researchers also introduced 3DGS into animatable human avatars. Specifically, D3GA [89] leverages cage-based deformation to model the motion of 3D Gaussians. While other approaches like GART [90], 3DGS-Avatar [91], GauHuman [92] and HUGS [93] employ linear blend skinning (LBS) to model human motions and reconstruct a 3DGS-based animatable avatar from monocular videos. However, these approaches cannot produce highly realistic and dynamic human appearances because they all represent the canonical 3D human using MLPs, facing the same low-frequency problem as NeRF-based approaches (Sec. II-B). We observe such an explicit point-based representation can be combined with 2D CNNs for high-quality avatar modeling. Concurrent works including ASH [94] and GaussianAvatar [95] parameterize the 3D character on a 2D UV map to predict Gaussian attributes using 2D CNNs, sharing similar ideas with our method. Differently, we parameterize the canonical 3D human on the front and back views by orthographic projection.

D. Human Relighting

Human relighting aims to manipulate the reflectance field of the human surface, thus enabling an immersive fusion with novel illumination. Conventional approaches [96]–[103] propose capturing the reflectance characteristics of a human subject through a LightStage arrangement. This setup entails controlled illumination systems and dense camera arrays, facilitating the generation of photorealistic renderings under diverse lighting conditions. However, such configurations are both financially demanding to capture and not readily accessible to the public. With the advancement of neural implicit representations, recent methods [36], [104]–[108] necessitate solely multi-view or even monocular video recordings obtained under constant unknown illumination conditions to model both human motion and light transport properties. Relighting4D [104] employs NeuralBody [109] as the dynamic human model and utilizes neural inverse rendering to decompose it into a 3D reflectance field, enabling the estimation of materials and lighting properties. Building upon this framework, Sun *et al.* [106] apply LBS to transform the reflectance field from the canonical space to the observation space, facilitating animated sequences with relighting effects. However, this work does not address shadowing effects. RANA [105] pretrains an SMPL+D-based representation incorporating albedo and normal map refinement techniques. It utilizes a simplified spherical harmonics lighting model to achieve relighting effects but lacks the capability to accurately model specular effects and shadows. Lin *et al.* [107] propose to estimate pose-aware light visibility through part-wise MLPs, demonstrating improved shadowing estimation when generalizing to unseen poses. Xu *et al.* [36] utilize Hierarchical Distance Queries (HDQ) via sphere tracing to calculate correct SDF values under arbitrary human poses and then incorporate distant field soft shadow (DFSS) for estimating reasonable soft visibility maps. IntrinsicAvatar [108] employs explicit Monte-Carlo ray tracing in canonical space to capture secondary shading effects, thereby enabling precise estimation of materials and environmental lighting. However, most previous methods rely on NeRF-based techniques, which inherently yield limited rendering quality. Thanks to the proposed powerful Gaussian representation, our method can achieve more vivid and realistic relighting results under novel poses.

III. Method

A. Preliminary: 3D Gaussian Splatting

3D Gaussian splatting [10] is an explicit point-based 3D representation that consists of a set of 3D Gaussians. Each 3D

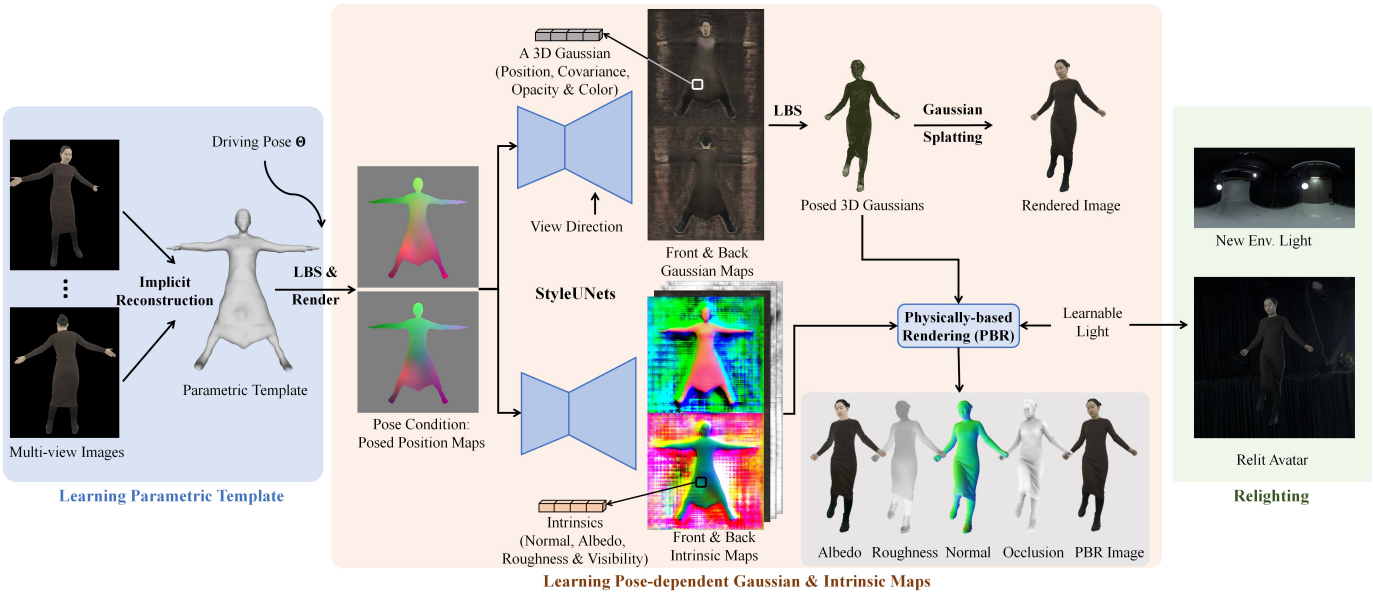

Fig. 2. **Illustration of the avatar modeling pipeline.** It contains two main steps: 1) Reconstruct a character-specific template from multi-view images. 2) Predict pose-dependent Gaussian and intrinsic maps through StyleUNets, and render the posed Gaussians by Gaussian splatting and physically-based rendering to learn both pose-dependent dynamics and avatar materials. Finally, given a novel environment light, we can animate the avatar with realistic dynamic appearances and shadow effects.

Gaussian is parameterized by its position (mean) $\boldsymbol{\mu}$, covariance matrix $\boldsymbol{\Sigma}$, opacity α and color $\mathbf{c}$, and its probability density function is formulated as

$$f(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right), \quad (1)$$

where we omit the constant factor in Eq. 1. For rendering a 2D image, the 3D Gaussians are splatted onto 2D planes, resulting in 2D Gaussians. The pixel color $\mathbf{C}$ is computed by blending N ordered 2D Gaussians overlapping this pixel:

$$\mathbf{C} = \sum_{i=1}^{N} \alpha_i \prod_{j=1}^{i-1}(1-\alpha_j)\mathbf{c}_i, \quad (2)$$

where $\mathbf{c}_i$ is the color of each 2D Gaussian, and α_i is the blending weight derived from the learned opacity and 2D Gaussian distribution [75].

B. Overview

Given multi-view RGB videos of a character and the corresponding SMPL-X [110] registrations about the per-frame pose and shared shape, our objective is to create a lifelike animatable avatar. As illustrated in Fig. 2, our method contains two main steps:

1) **Learning Parametric Template.** We begin by selecting a frame with a near A-pose from the input videos, and then optimize a canonical SDF and color field to fit the multi-view images through SMPL skinning and SDF-based volume rendering [60]. The template mesh is subsequently extracted from the canonical SDF field using Marching Cubes [111]. We then diffuse the skinning weights from the SMPL vertices to the template surface, obtaining a deformable parametric template.
2) **Learning Pose-dependent Gaussian and Intrinsic Maps.** Given a training pose, we first deform the template to the posed space via linear blend skinning (LBS) and render the posed vertex coordinates to canonical front & back views to obtain two position maps. The position maps serve as the pose condition and are translated into front & back Gaussian and intrinsic maps through StyleUNets [17]. We then extract valid 3D Gaussians inside the template mask from the Gaussian map, and deform the canonical 3D Gaussians to the posed space by LBS. In one branch, we directly render the posed 3D Gaussians to a camera view using the observed colors on the Gaussian maps. On the other hand, we render the posed 3D Gaussians using the PBR color computed from the intrinsic maps. These two branches allow our method not only to learn pose-dependent animation but also to decompose the avatar appearance into materials and light conditions for relighting purposes.

C. Avatar Representation

1) Learning Parametric Template: Given the multi-view videos, we first select one frame in which the character is under a near A-pose. Our goal is to reconstruct a canonical geometric model as the template from the multi-view images. Specifically, we represent the canonical character as an SDF and color field instantiated by an MLP. To associate the canonical and posed spaces, we precompute a skinning weight volume $\mathcal{W}$ in the canonical space by diffusing the weights from the SMPL surface throughout the whole 3D volume along the surface normal [11]. For each point in the posed space, we search its canonical correspondence by root finding [44]:

$$\min_{\mathbf{x}_c} \|\mathrm{LBS}(\mathbf{x}_c; \boldsymbol{\Theta}, \mathcal{W}) - \mathbf{x}_p\|_2^2, \quad (3)$$

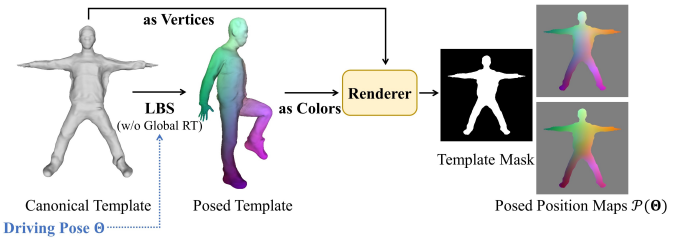

Fig. 3. **Illustration of the posed position maps.**

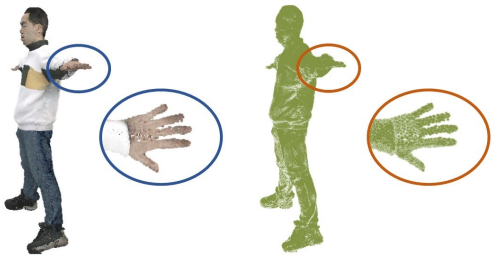

Fig. 4. **Canonical 3D Gaussians on side regions and hands.**

where LBS$(\cdot)$ is a linear blend skinning function that transforms a canonical point $\mathbf{x}_\mathrm{c}$ to its posed position $\mathbf{x}_\mathrm{p}$ in accordance with the SMPL pose $\mathbf{\Theta}$. Then the canonical correspondence is fed into the MLP to query its SDF and color, which are used to render RGB images by SDF-based volume rendering [60]. The rendered images are compared with the ground truth for optimizing the canonical fields via differentiable volume rendering. Finally, we extract the geometric template from the SDF field and query the skinning weights for each vertex in the precomputed weight volume $\mathcal{W}$, obtaining a deformable parametric template.

2) Template-guided Parameterization: Previous human avatar representations in NeRF-based approaches [5]–[7] necessitate the coordinate-based MLPs for the formulation of the implicit NeRF function. However, MLPs have demonstrated a low-frequency bias [9], hindering their ability to model high-frequency human dynamics. In light of this observation, we replace MLPs with more powerful 2D CNNs for creating higher-quality human avatars. To ensure compatibility with 2D networks, the 3D representation of the human avatar needs to be parameterized in 2D space. Therefore, we propose to parameterize the 3D Gaussians anchored on the canonical template onto front & back views via orthogonal projection. As illustrated in Fig. 3, given a driving pose $\mathbf{\Theta}$, we first deform the template to the posed space via LBS. Note that we do not consider the global transformation in this skinning process, because the global orientation and translation would not change the human dynamic details. Then we take the posed coordinate as the vertex color on the canonical template, and render it to both front & back views by orthogonal projection, obtaining posed position maps $\mathcal{P}_\mathrm{f}(\mathbf{\Theta})$ and $\mathcal{P}_\mathrm{b}(\mathbf{\Theta})$ that serve as pose conditions for the network.

3) Pose-dependent Gaussian Maps: We employ a powerful StyleGAN-based CNN , StyleUNet [17] $\mathcal{F}_\mathcal{G}$, to predict pose-dependent Gaussian maps from the pose conditions:

$$\mathcal{G}_\mathrm{f}(\mathbf{\Theta}), \mathcal{G}_\mathrm{b}(\mathbf{\Theta}) \leftarrow \mathcal{F}_\mathcal{G}(\mathcal{P}_\mathrm{f}(\mathbf{\Theta}), \mathcal{P}_\mathrm{b}(\mathbf{\Theta}), \mathcal{V}), \tag{4}$$

where $\mathcal{G}_\mathrm{f}(\mathbf{\Theta})$ and $\mathcal{G}_\mathrm{b}(\mathbf{\Theta})$ are front and back pose-dependent Gaussian maps, respectively, and each pixel represents a 3D Gaussian [10] including a position, covariance, opacity and color. To ensure that the position attribute of predicted Gaussian maps approximates the canonical human body, we opt to predict an offset map $\Delta\mathcal{O}(\mathbf{\Theta})$ on the parametric template instead of a global position map. We also modulate the output color attributes on Gaussian maps with a view direction map $\mathcal{V}$ to model view-dependent variance like NeRF-based approaches [5]. We extract canonical 3D Gaussians inside the template mask from the pose-dependent Gaussian maps. It is worth mentioning that despite utilizing only front and back views for parameterizing the 3D Gaussians, the resulting point clouds still cover the side regions and hands of the human body as demonstrated in Fig. 4. The reason is that the projection to front & back views is orthographic, thus there exist sufficient 3D Gaussians to model these parts.

4) LBS of 3D Gaussians: To render the synthesized avatar under the driving pose, we need to deform the canonical 3D Gaussians to the posed space. Specifically, given a canonical 3D Gaussian, we transform its position $\mathbf{p}_\mathrm{c}$ and covariance $\mathbf{\Sigma}_\mathrm{c}$ attributes:

$$\begin{aligned} \mathbf{p}_\mathrm{p} &= \mathbf{R}\mathbf{p}_\mathrm{c} + \mathbf{t}, \\ \mathbf{\Sigma}_\mathrm{p} &= \mathbf{R}\mathbf{\Sigma}_\mathrm{c}\mathbf{R}^\top, \end{aligned} \tag{5}$$

where $\mathbf{R}$ and $\mathbf{t}$ are the rotation matrix and translation vector calculated with the skinning weights of each 3D Gaussian. Finally, we render the posed 3D Gaussians to a desired camera view through splatting-based rasterization (Eq. 2).

5) Pose-dependent Intrinsic Maps: The pose-dependent Gaussian maps only represent the human appearances under the illumination of the capture environment, limiting animation under novel lighting conditions. To disentangle the avatar geometry, material and lighting conditions, we leverage the classic rendering equation [112] to simulate the rendering process:

$$L_o(\mathbf{x}, \boldsymbol{\omega}_o) = \int_\Omega L_i(\mathbf{x}, \boldsymbol{\omega}) f(\mathbf{x}, \boldsymbol{\omega}_i, \boldsymbol{\omega}_o; \alpha, \gamma, \mathbf{n})(\boldsymbol{\omega}_i \cdot \mathbf{n}) d\boldsymbol{\omega}_i, \tag{6}$$

where $L_o(\mathbf{x}, \boldsymbol{\omega}_o)$ and $L_i(\mathbf{x}, \boldsymbol{\omega}_i)$ are the outgoing and incident radiance at a surface position $\mathbf{x}$ along direction $\boldsymbol{\omega}_o$ and $\boldsymbol{\omega}_i$, respectively, and $\mathbf{n}$ is the normal vector at $\mathbf{x}$. $f(\mathbf{x}, \boldsymbol{\omega}_i, \boldsymbol{\omega}_o; \alpha, \gamma, \mathbf{n})$ is the Bidirectional Reflectance Distribution Function (BRDF) determined by the surface geometry (normal $\mathbf{n}$) and material properties including albedo α and roughness γ. Considering the light visibility, the incident radiance is further formulated as

$$L_\mathrm{i}(\mathbf{x}, \boldsymbol{\omega}_i) = V(\mathbf{x}, \boldsymbol{\omega}_i) L_i(\boldsymbol{\omega}_i), \tag{7}$$

where $V(\mathbf{x}, \boldsymbol{\omega}_i) \in \{0, 1\}$ indicates whether $\mathbf{x}$ is visible along the light direction $\boldsymbol{\omega}_i$. The global light $L_i(\boldsymbol{\omega}_i)$ is parametrized as learnable Spherical Harmonics (SH), which are optimized during the inverse rendering process.

Based on the physically-based rendering process, we additionally learn pose-dependent intrinsic maps including normal

$\mathcal{N}(\mathbf{\Theta})$, albedo $\mathcal{A}(\mathbf{\Theta})$, and roughness $\boldsymbol{\gamma}(\mathbf{\Theta})$ maps on the front and back canonical views:

$$\begin{aligned}\mathcal{N}_{\mathrm{f}}(\mathbf{\Theta}), \mathcal{N}_{\mathrm{b}}(\mathbf{\Theta}) &\leftarrow \mathcal{F}_{\mathcal{N}}(\mathcal{P}_{\mathrm{f}}(\mathbf{\Theta}), \mathcal{P}_{\mathrm{b}}(\mathbf{\Theta})),\\ \mathcal{A}_{\mathrm{f}}(\mathbf{\Theta}), \mathcal{A}_{\mathrm{b}}(\mathbf{\Theta}) &\leftarrow \mathcal{F}_{\mathcal{A}}(\mathcal{P}_{\mathrm{f}}(\mathbf{\Theta}), \mathcal{P}_{\mathrm{b}}(\mathbf{\Theta})),\\ \boldsymbol{\gamma}_{\mathrm{f}}(\mathbf{\Theta}), \boldsymbol{\gamma}_{\mathrm{b}}(\mathbf{\Theta}) &\leftarrow \mathcal{F}_{\boldsymbol{\gamma}}(\mathcal{P}_{\mathrm{f}}(\mathbf{\Theta}), \mathcal{P}_{\mathrm{b}}(\mathbf{\Theta})),\end{aligned} \tag{8}$$

where $\mathcal{F}_{\mathcal{N}}$, $\mathcal{F}_{\mathcal{A}}$ and $\mathcal{F}_{\boldsymbol{\gamma}}$ are StyleUNet modules. More specifically, the learned normal map $\mathcal{N}(\mathbf{\Theta})$ is an offset of the surface normal on the parametric template for better pose generalization. The light visibility $V(\mathbf{x}, \boldsymbol{\omega}_i)$ can be computed from the posed 3D Gaussians by point-based ray tracing [20]. However, as mentioned in [20], computing the light visibility during training is not preferred because of the computational complexity. Therefore, we train an additional network to predict light visibility and supervise the prediction by randomly sampling view directions. Specifically, we formulate the light-direction-dependent visibility as SH, and predict SH coefficient map $\mathcal{K}(\mathbf{\Theta})$ using a StyleUNet $\mathcal{F}_{\mathcal{K}}$:

$$\mathcal{K}_{\mathrm{f}}(\mathbf{\Theta}), \mathcal{K}_{\mathrm{b}}(\mathbf{\Theta}) \leftarrow \mathcal{F}_{\mathcal{K}}(\mathcal{P}_{\mathrm{f}}(\mathbf{\Theta}), \mathcal{P}_{\mathrm{b}}(\mathbf{\Theta})). \tag{9}$$

Similar to the Gaussian maps, each pixel value on normal, albedo, roughness and light visibility maps is associated with a 3D Gaussian. Following Relightable 3D Gaussian [20], we sample N incident light directions over the hemisphere space, and calculate the PBR color of each 3D Gaussian by the discrete form of Eq. 6:

$$\begin{aligned}&L_o(\boldsymbol{\mu}, \boldsymbol{\omega}_o) =\\ &\sum_{\boldsymbol{\omega}_i} L_i(\boldsymbol{\mu}, \boldsymbol{\omega}) f(\boldsymbol{\mu}, \boldsymbol{\omega}_i, \boldsymbol{\omega}_o; \alpha_{\boldsymbol{\mu}}, \gamma_{\boldsymbol{\mu}}, \mathbf{n}_{\boldsymbol{\mu}})(\boldsymbol{\omega}_i \cdot \mathbf{n}_{\mu}) \Delta\boldsymbol{\omega}_i,\end{aligned} \tag{10}$$

where $\boldsymbol{\mu}$ is the position of a 3D Gaussian, $\alpha_{\boldsymbol{\mu}}$, $\gamma_{\boldsymbol{\mu}}$ and $\mathbf{n}_{\boldsymbol{\mu}}$ is the corresponding albedo, roughness and normal attribute. Finally, the physically-based rendered image can be obtained by rasterization (Eq. 2).

D. Training

The optimizable parameters of the avatar model include the parameters of StyleUNets and an environment light map $\boldsymbol{l}_{\mathrm{env}}$. Our training loss consists of five parts: the reconstruction loss $\mathcal{L}_{\mathrm{recon}}$, PBR loss $\mathcal{L}_{\mathrm{PBR}}$, normal loss $\mathcal{L}_{\mathrm{normal}}$, visibility loss $\mathcal{L}_{\mathrm{vis}}$ and regularization loss $\mathcal{L}_{\mathrm{reg}}$:

$$\mathcal{L} = \mathcal{L}_{\mathrm{recon}} + \mathcal{L}_{\mathrm{PBR}} + \lambda_{\mathrm{normal}}\mathcal{L}_{\mathrm{normal}} + \lambda_{\mathrm{vis}}\mathcal{L}_{\mathrm{vis}} + \mathcal{L}_{\mathrm{reg}}. \tag{11}$$

1) Reconstruction Loss: The reconstruction loss aims to learn the geometry and the observed texture of the human avatar from the multi-view videos without considering the PBR process. We denote the rendered image using color attributes from the Gaussian maps as $\mathbf{C}_{\mathrm{OBS}}$. The reconstruction loss involves an L1 loss and a perceptual loss [113] between $\mathbf{C}_{\mathrm{OBS}}$ and the ground-truth image $\mathbf{C}_{\mathrm{GT}}$:

$$\mathcal{L}_{\mathrm{recon}} = \mathcal{L}_1(\mathbf{C}_{\mathrm{OBS}}, \mathbf{C}_{\mathrm{GT}}) + \lambda_{\mathrm{perceptual}}\mathcal{L}_{\mathrm{perceptual}}(\mathbf{C}_{\mathrm{OBS}}, \mathbf{C}_{\mathrm{GT}}). \tag{12}$$

2) PBR Loss: The PBR loss aims to disentangle the light condition and avatar materials from the multi-view observations. We denote the rendered image by the PBR process (i.e., Eq. 10) as $\mathbf{C}_{\mathrm{PBR}}$. The PBR loss involves an L1 loss and a perceptual loss between $\mathbf{C}_{\mathrm{PBR}}$ and $\mathbf{C}_{\mathrm{GT}}$:

$$\mathcal{L}_{\mathrm{PBR}} = \mathcal{L}_1(\mathbf{C}_{\mathrm{PBR}}, \mathbf{C}_{\mathrm{GT}}) + \lambda_{\mathrm{perceptual}}\mathcal{L}_{\mathrm{perceptual}}(\mathbf{C}_{\mathrm{PBR}}, \mathbf{C}_{\mathrm{GT}}). \tag{13}$$

3) Normal Loss: We employ a pretrained normal estimation network [114] to supervise our predicted normal. Specifically, we render a normal image with the predicted normal as additional channels by rasterization, and compare it with the estimated one by [114] using L1 loss.

4) Visibility Loss: In each iteration, for each 3D Gaussian, we randomly sample a light direction over the hemisphere around its normal. We compute L1 loss between the predicted visibility of the sampled directions and the ground-truth one obtained by point-based ray tracing [20].

5) Regularization Loss: For the stability and convergence of avatar training, we design several regularization losses on the geometry and materials. Specifically, the regularization losses involve a geometric regularization loss and smooth regularization losses on albedo and roughness:

$$\mathcal{L}_{\mathrm{reg}} = \lambda_{\mathrm{geo}}\mathcal{L}_{\mathrm{geo}} + \lambda_{\mathrm{albedo}}\mathcal{L}_{\mathrm{smooth}}(\mathbf{I}_{\mathcal{A}}) + \lambda_{\mathrm{roughness}}\mathcal{L}_{\mathrm{smooth}}(\mathbf{I}_{\boldsymbol{\gamma}}), \tag{14}$$

where $\mathcal{L}_{\mathrm{geo}} = \|\Delta\mathcal{O}\|_2^2$ restrains the predicted offset map $\Delta\mathcal{O}$ from being extremely large, and the smooth loss is a bilateral smoothness term [20] that constrains the material properties to change continuously in areas with smooth colors:

$$\mathcal{L}_{\mathrm{smooth}}(\mathbf{I}) = \|\nabla\mathbf{I}\| \exp(-\|\nabla\mathbf{C}_{\mathrm{GT}}\|), \tag{15}$$

where $\mathbf{I}$ is the rendered albedo or roughness map ($\mathbf{I}_{\mathcal{A}}$ or $\mathbf{I}_{\boldsymbol{\gamma}}$) via splatting-based rasterization.

E. Animation and Relighting

1) Pose Projection Strategy: Benefiting from the effective avatar representation, our method can reconstruct detailed human appearances under the training poses. However, given the inherently data-driven nature of learning-based avatars, addressing generalization to novel poses is also necessary and important. RAM-Avatar [115] trains a VAE to transform the testing pose into an in-distribution one. In this work, we propose to utilize Principal Component Analysis (PCA) to project a novel driving pose signal into the distribution of seen training poses for better generalization. Specifically, given a pose condition represented by posed position maps, we extract valid points and concatenate them as a vector $\mathbf{x}_t \in \mathbb{R}^{3M}$ (M is the point number). The vector of each training frame composes a matrix $\mathbf{X} = [\mathbf{x}_1, \cdots, \mathbf{x}_T]$, where T is the number of training frames. We perform PCA on $\mathbf{X}$, producing N principal components $\mathbf{S} = [\mathbf{s}_1, \cdots, \mathbf{s}_N] \in \mathbb{R}^{3M\times N}$ and standard deviation of each component σ_i. Given position maps derived by a novel driving pose, we project the corresponding vector $\mathbf{x}$ into the PCA space by

$$\boldsymbol{\beta} = \mathbf{S}^{\top} \cdot (\mathbf{x} - \bar{\mathbf{x}}), \tag{16}$$

Fig. 5. **Example animatable avatars with high-fidelity dynamic appearances created by our method.**

where $\bar{\mathbf{x}}$ is the mean of $\mathbf{X}$. Then we reconstruct the positions from the low-dimensional coefficient $\boldsymbol{\beta}$ by

$$\mathbf{x}_{\text{recon}} = \mathbf{S} \cdot \boldsymbol{\beta} + \bar{\mathbf{x}}, \tag{17}$$

then we reshape $\mathbf{x}_{\text{recon}}$ into a $M \times 3$ tensor, and scatter it onto the position maps. To constrain the reconstructed position maps to lie in the distribution of training poses, we clip each component of $\boldsymbol{\beta}$ within the bound of $[-2\sigma_i, 2\sigma_i]$. Overall, the pose projection strategy ensures reasonable interpolation within the distribution of training poses, enabling better generalization to novel poses as shown in Fig. 14.

2) Relighting: Thanks to the introduction of PBR into the avatar representation, our method is able to relight the avatar under novel illumination. Specifically, given a novel pose, we first feed it into the avatar networks to predict Gaussian, normal, albedo and roughness maps. Note that we do not predict light visibility maps during testing because we empirically find it cannot accurately generalize to novel poses. We hypothesize the reasons are that the visibility is heavily determined by global self-occlusions, and the variety of training poses is limited. To this end, we directly calculate the visibility of posed 3D Gaussians via point-based ray tracing [20]. Then, given a novel environment map, we inject it with the predicted normal, ableo, and roughness as well as calculated visibility into the PBR equation (Eq. 10) to compute the PBR color of each 3D Gaussian. Finally, the relit image is obtained by splatting-based rasterization. As illustrated in Fig. 7, our method can produce photorealistic animation under novel illumination.

IV. Experiments

A. Results

1) Animation: As shown in Fig. 1 and Fig. 5, our method can create realistic avatars with high-fidelity dynamic details from multi-view videos. We also show results animated by challenging out-of-distribution poses from AMASS dataset [116] in Fig. 6. Thanks to the effective avatar representation and pose projection strategy, our method can produce animation results with highly dynamic, realistic and generalized appearances.

2) Relighting: In addition to animation, our method extends support to relighting applications. By naturally incorporating physically-based inverse rendering techniques into our powerful avatar representation, we achieve accurate intrinsic decomposition and vivid relighting results, as demonstrated in Fig. 1 and Fig. 7.

Please refer to the supplementary video for more sequential animation and relighting results.

B. Dataset and Metric

We mainly utilize three public datasets for the experiments, including 3 sequences with 24 views from THuman4.0 dataset [7], 3 sequences with 16 views from AvatarReX dataset [66] and 5 sequences with 160 views from ActorsHQ dataset [117] (we only use 47 full-body views for avatar modeling). THuman4.0 and AvatarReX datasets also provide the SMPL-X [110] registrations. We fit SMPL-X for ActorsHQ dataset using the method proposed by Zhang *et al.* [118]. We split each sequence as training and testing chunks, and the training chunk contains 1500 $\sim$ 3000 frames.

Fig. 6. **Example sequential animation results by our method.** Each row is an animation sequence involving 3 subjects. Our method can generate realistic and reasonable dynamic details even under novel poses from the AMASS dataset [116].

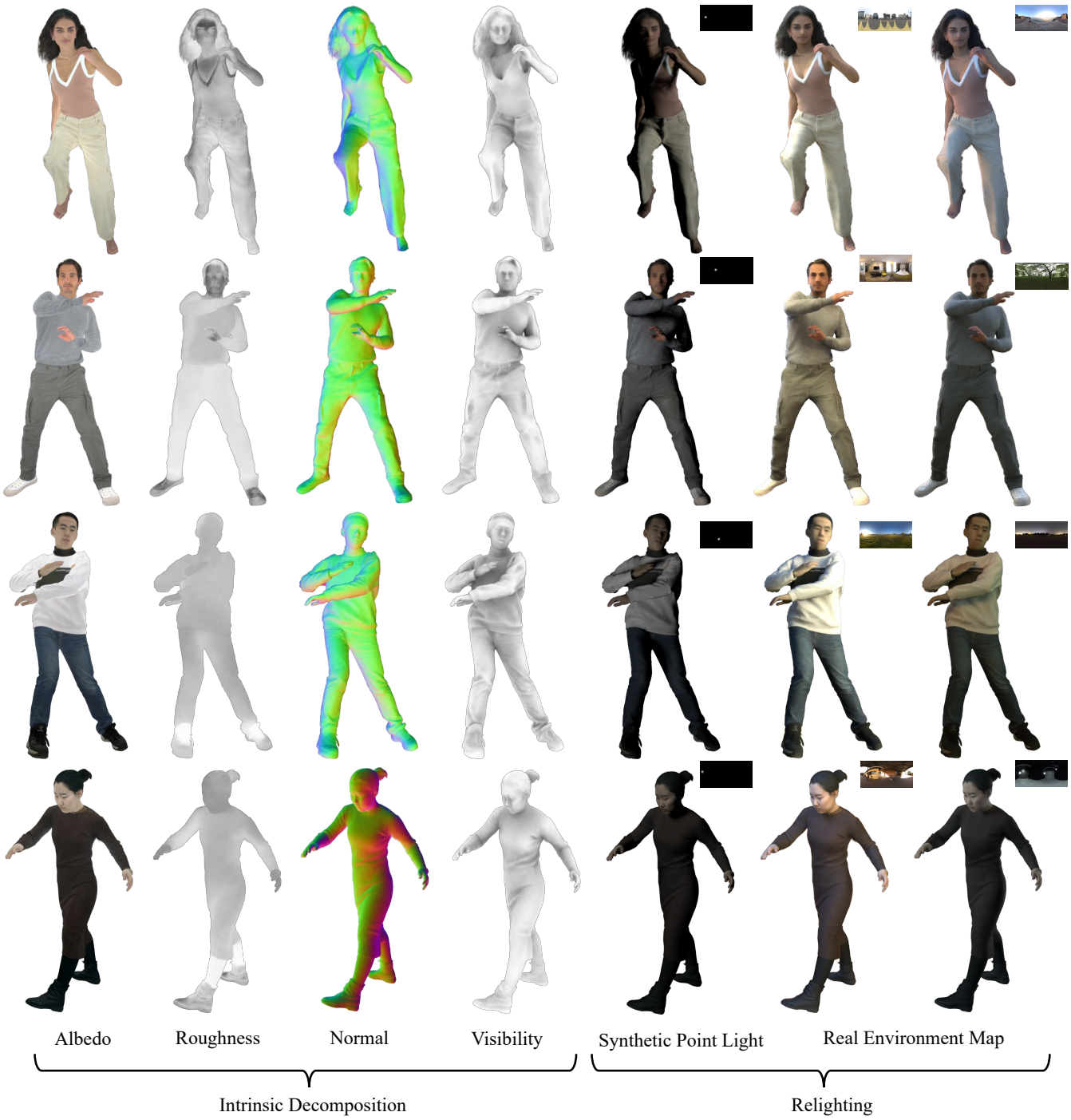

Fig. 7. **Example intrinsic decomposition and relighting results under novel poses by our method.** The first four columns display the decomposition results of material and geometry. The fifth column showcases the relighting outcomes with a directional point light, whereas the remaining columns are relit using real environment maps.

We adopt Peak Signal-to-Noise Ratio (PSNR), Structure Similarity Index Measure (SSIM) [119], Learned Perceptual Image Patch Similarity (LPIPS) [113] and Fréchet Inception Distance (FID) [120] for quantitative experiments. PSNR and SSIM are computed on the entire image at the original resolution, while LPIPS and FID are computed on the cropped minimal square that covers the human body.

C. Implementation Details

1) Template Reconstruction: We optimize an SDF and color field represented by an MLP consisting of intermediate layers with (512, 256, 256, 256, 256, 256) neurons. Given a posed point, we find accurate correspondence in the canonical space by root finding. Following ARAH [59], we initialize the correspondence as the canonical position that is computed by inverse skinning based on blending weights of the closest SMPL vertex. Different from SNARF [44] and ARAH [59]

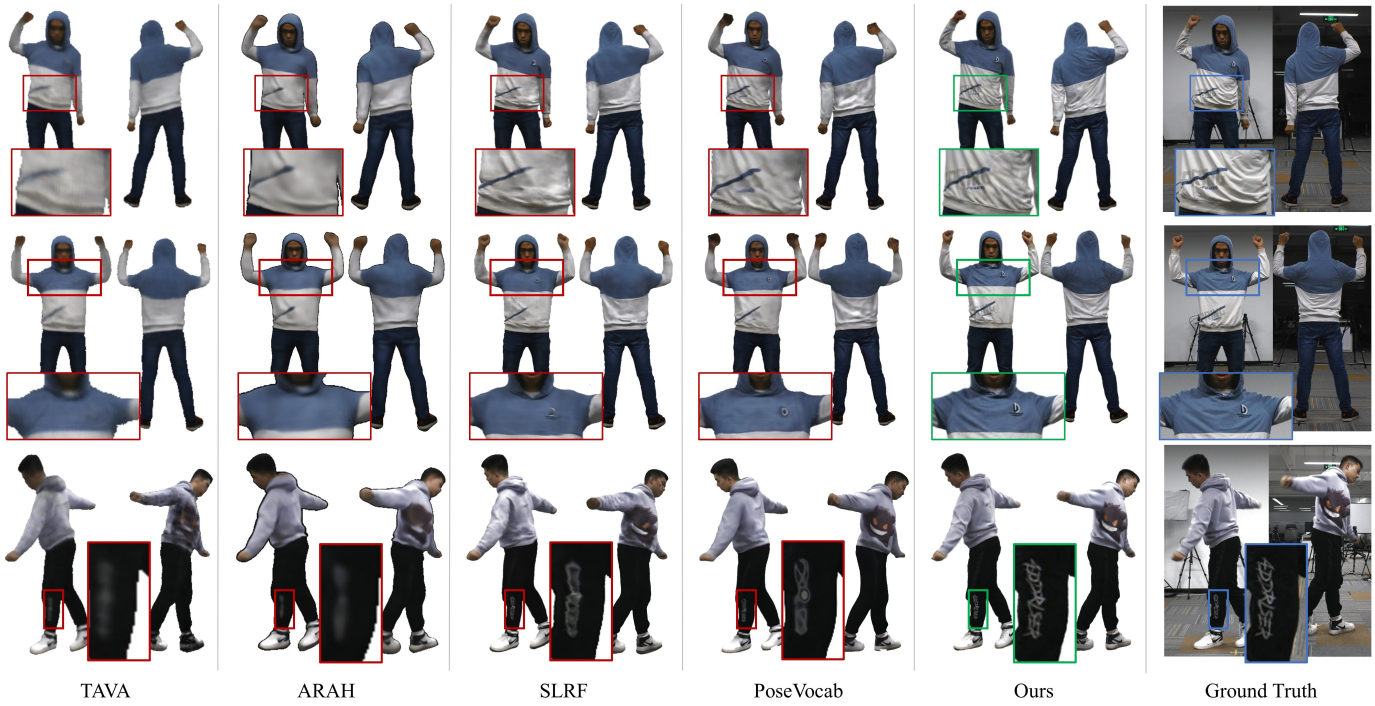

Fig. 8. **Qualitative comparison with state-of-the-art NeRF-based body-only avatars including TAVA [8], ARAH [59], SLRF [7] and PoseVocab [63] on novel pose synthesis.**

TABLE I
LOSS WEIGHTS IN THE TRAINING PROCESS.

Loss Weight	Value
$\lambda_{\text{perceptual}}$	0.1
λ_{normal}	0.2
λ_{vis}	0.1
λ_{geo}	0.01
λ_{albedo}	0.005
$\lambda_{\text{roughness}}$	0.005

TABLE II
QUANTITATIVE COMPARISON WITH STATE-OF-THE-ART NERF-BASED BODY-ONLY AVATARS.

Method	PSNR ↑	SSIM ↑	LPIPS ↓	FID ↓
Ours	**28.0714**	**0.9739**	**0.0515**	**29.4831**
PoseVocab [63]	26.3784	0.9707	0.0592	49.4541
SLRF [7]	26.9015	0.9724	0.0600	52.0613
ARAH [59]	22.3004	0.9616	0.1075	90.6077
TAVA [8]	26.8019	0.9705	0.0915	96.3474

that utilize the Broyden's method [121] to solve Eq. 3, we employ the Gauss-Newton method by implementing a customized CUDA kernel. The training loss of template reconstruction involves an RGB loss, a mask loss and an Eikonal loss [122].

2) Network Architecture: The network in our avatar representation is composed of StyleUNet [17], a conditional StyleGAN-based [14] generator. Differently, we adapt the original StyleUNet by incorporating two decoders to predict both front & back Gaussian and intrinsic maps. The resolution of the input position map is 512×512, and the resolution of the output Gaussian and intrinsic maps is 1024×1024. Specifically, we utilize five different StyleUNets to output color (3-channel), position (3-channel), other Gaussian attributes (8-channel), albedo & roughness (4-channel) and normal & visibility (19-channel). The visibility is represented as 16-dimensional SH coefficients. In the color StyleUNet, we modulate the color output with a view direction map to model view-dependent effects. Each pixel on the view direction map indicates the angle between the view direction and the template normal. The view direction map is encoded through a tiny CNN, then the encoded feature map is injected into an intermediate decoder layer of the color StyleUNet.

3) Training: We adopt the Adam optimizer [123] for training the network with a learning rate of 5×10^{-4}. The loss weights are set as illustrated in Tab. I. The batch size is 1, the total iteration number is 500k, and the training procedure takes about two days on one RTX 4090.

4) Running Time: For only animation, it takes around 0.13 secs to render one frame. Given novel illumination for relighting, it takes $4 \sim 10$ secs to synthesize one frame due to the additional computational cost of light visibility computation in PBR. The relighting time cost mainly depends on the numbers of 3D Gaussians and sampled rays.

D. Comparison on Avatar Animation

In the comparisons with state-of-the-art animatable avatars, we first compare our method with NeRF-based approaches including both body-only (TAVA [8], ARAH [59], SLRF [7], PoseVocab [63]) and full-body (AvatarReX [66]) avatars. Then we compare our method with concurrent 3D Gaussian splatting-based avatars including 3DGS-Avatar [91] and GaussianAvatar [95].

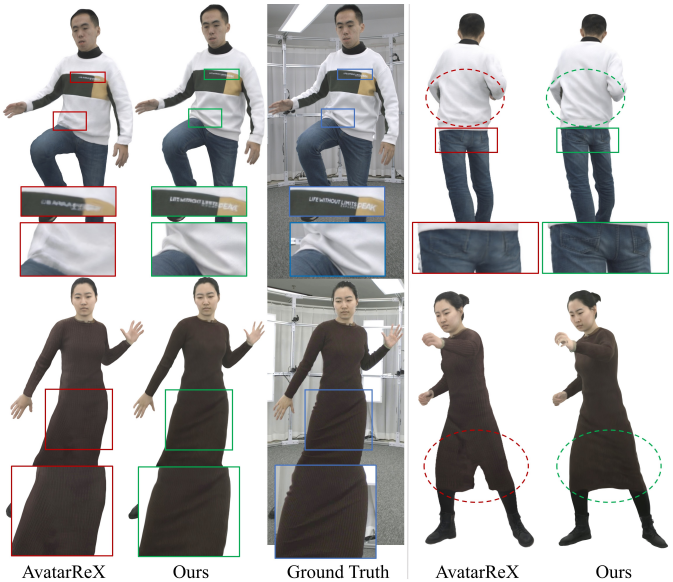

Fig. 9. **Qualitative comparison with AvatarReX [66] on both training pose reconstruction (a) and novel view synthesis (b).**

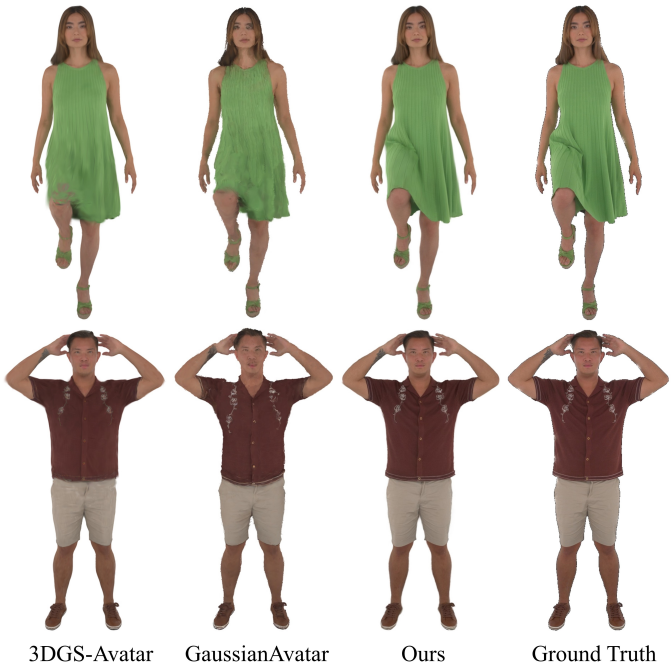

Fig. 10. **Qualitative comparison with state-of-the-art 3D Gaussian splatting-based avatars including 3DGS-Avatar [91] and GaussianAvatar [95].**

TABLE III
QUANTITATIVE COMPARISON WITH THE STATE-OF-THE-ART NERF-BASED FULL-BODY AVATAR.

Method	PSNR ↑	SSIM ↑	LPIPS ↓	FID ↓
Ours	**30.6143**	**0.9803**	**0.0290**	**13.2417**
AvatarReX [66]	23.2475	0.9567	0.0646	31.1387

TABLE IV
QUANTITATIVE COMPARISON WITH STATE-OF-THE-ART 3D GAUSSIAN SPLATTING-BASED AVATARS.

Method	PSNR ↑	SSIM ↑	LPIPS ↓	FID ↓
Ours	**30.3607**	**0.9682**	**0.0339**	**33.4665**
3DGS-Avatar [91]	28.7836	0.9511	0.0418	49.3673
GaussianAvatar [95]	26.9497	0.9389	0.0407	38.5387

1) NeRF-based Body-only Avatars: We compare our method with TAVA, ARAH, SLRF, and PoseVocab on "subject00" and "subject02" sequences of THuman4.0 dataset [7]. We run the released codes of TAVA, ARAH and PoseVocab on the dataset, and request the results of SLRF from the authors. We present qualitative comparisons on novel pose synthesis in Fig. 8. In contrast to other methods, our approach excels in animating highly realistic avatars with significant improvement on high-fidelity dynamic details, including garment wrinkles, logos and other textural patterns. The quantitative comparison is also performed on the testing chunk (the 2000-2500 frames and "cam18" view) of the "subject00" sequence as shown in Tab. II, and these numerical results prove that our method achieves more accurate animation. Although PoseVocab and SLRF introduce a learnable pose dictionary or local NeRFs to improve the representation ability of the NeRF MLP, they still suffer from the low-frequency bias [9] of MLPs and fail to create highly realistic avatars. Contrarily, our method leverages powerful 2D CNNs and explicit 3D Gaussian splatting, thus achieving modeling finer-grained dynamic appearances.

2) NeRF-based Full-body Avatars: Full-body avatars including TotalSelfScan [64], X-Avatar [65] and AvatarReX [66] can realize expressive control of the body, hands and face. TotalSelfScan reconstructs full-body avatars from monocular self-rotation videos, and only displays animations that appear very rigid. X-Avatar requires 3D human scans under different poses as input for creating avatars. AvatarReX is the most relevant work with our method, i.e., creating avatars from multi-view videos. Fig. 9 shows the comparison with AvatarReX on both training and novel poses. Fig. 9 (a) demonstrates that our method can reconstruct more faithful and vivid details compared with AvatarReX. Although AvatarReX introduces local feature patches to encode more details, it remains constrained by the representation ability of the conditional NeRF MLPs. Fig. 9 (b) shows that given a novel pose, our method not only generates more realistic details but also produces more reasonable non-rigid deformation, particularly for long dresses, in comparison with AvatarReX. This is attributed to the ability of our method to learn pose-dependent deformations on a character-specific template that has already modeled the basic shape of the wearing garments. In contrast, AvatarReX learns sparse node translations on the naked SMPL model, resulting in artifacts for long dresses. Tab. III reports the quantitative comparison on training pose reconstruction. Our method also outperforms AvatarReX on the reconstruction accuracy. The numerical results are evaluated on the first 500 frames and the "22010710" camera view in the "avatarrex_zzr" sequence from AvatarReX dataset.

3) 3D Gaussian Splatting-based Avatars: We qualitatively compare our method with 3DGS-Avatar [91] and GaussianAvatar [95] using "Actor01" and "Actor02" sequences from

Fig. 11. **Qualitative comparison with the state-of-the-art human performance relighting methods, R4D [104] and RA [107].**

TABLE V
QUANTITATIVE COMPARISON WITH STATE-OF-THE-ART HUMAN PERFORMANCE RELIGHTING METHOD.

Method	PSNR ↑	SSIM ↑	LPIPS ↓	FID ↓
Ours	**31.6339**	**0.9836**	**0.0208**	**24.1962**
RA [107]	24.8013	0.9694	0.0738	57.3957
R4D [104]	23.0883	0.9608	0.0968	112.9600

ActorsHQ dataset [117] in Fig. 10. It shows that our method outperforms other approaches by a large margin on the avatar quality, especially the dynamic wrinkles of the garments. 3DGS-Avatar [91] utilizes pure MLPs to regress the non-rigid deformations and pose-dependent appearances, suffering from the limited capacity of MLPs. Although GaussianAvatar [95] employs a 2D U-Net [124] to regress Gaussian parameters on SMPL UV space, it freezes the opacity and rotation attribute as pose-agnostic variables, deteriorating the modeling ability of the whole model. Moreover, these two methods utilize the naked SMPL model to parameterize 3D Gaussians and both fail to model detailed motions and appearances of loose clothes as shown in the top row of Fig. 10. We also report the numerical results in Tab. IV, and our method also quantitatively outperforms 3DGS-Avatar and GaussianAvatar. The numerical results are computed on the 48-548 frames and the "Cam127" camera view in the "Actor01/Sequence1" from ActorsHQ dataset.

E. Comparison on Human Performance Relighting

We compare our method with the state-of-the-art human performance relighting methods Relighting4D (R4D) [104] and Relightable and Animatable Avatar (RA) [107] on ActorsHQ [117] and AvatarReX [66] datasets.

We present the qualitative and quantitative comparison in Fig. 11 and Tab. V. Our method excels in precise intrinsic decomposition due to the proposed physically-based avatar representation, particularly evident in the albedo and normal details. This highlights the enhanced capabilities of our approach in capturing intricate details in both geometry and appearance. Moreover, the accurate material and geometry can then contribute to natural relighting under novel illumination. However, R4D [104] and RA [107], as NeRF-based implicit methods, fail to model detailed material and geometry of the character due the limited capacity of their avatar representations. These two approaches produce blurry results on both texture and geometry, leading to diminished relighting results.

F. Ablation Study

We evaluate the core contributions of our method in this subsection.

1) Parametric Template: We evaluate the learned parametric template by replacing it with a naked parametric model, SMPL-X [110]. Fig. 12 shows that SMPL-X fails to represent the long dress whose topology is not consistent with the SMPL-X model, yielding poor generalization to novel poses. Conversely, our character-specific template is adaptively reconstructed from the input video to model the basic shape of the wearing garments. We also quantitatively compare the

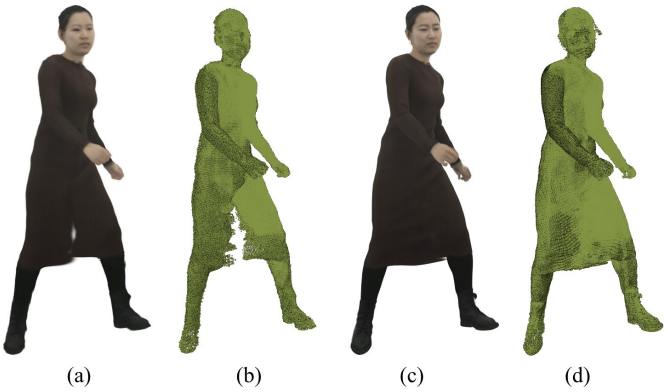

Fig. 12. **Ablation study of the parametric template.** (a,b) Rendered results and 3D Gaussians using SMPL-X. (c,d) Rendered results and 3D Gaussians using the character-specific template.

TABLE VI
QUANTITATIVE ABLATION STUDY ON THE PARAMETRIC TEMPLATE.

	PSNR ↑	SSIM ↑	LPIPS ↓	FID ↓
Parametric Template	**31.2183**	**0.9858**	**0.0344**	**36.9905**
SMPL-X	30.5241	0.9842	0.0401	47.5066

Fig. 13. **Comparison between representations with different backbones on training pose reconstruction.**

TABLE VII
QUANTITATIVE COMPARISON BETWEEN REPRESENTATIONS WITH DIFFERENT BACKBONES.

	PSNR ↑	SSIM ↑	LPIPS ↓	FID ↓
StyleUNet [17]	**29.3127**	**0.9664**	**0.0378**	**27.3143**
U-Net [124]	26.4255	0.9435	0.0507	31.3838
MLP	26.8961	0.9497	0.0650	87.0793

reconstructed parametric template with the naked SMPL-X model [110] on the animation accuracy in Tab. VI. It shows that the reconstructed template can animate the 3D Gaussians more accurately.

2) Backbones: To demonstrate the superior representation ability of 2D CNNs (StyleUNet in our settings), we replace StyleUNet with a coordinate-based MLP and a standard U-Net [124], respectively. The MLP takes a canonical point and pose vector as input, and returns the 3D Gaussian attributes of this point. While the standard U-Net replaces the StyleUNet as the backbone. Fig. 13 and Tab. VII show the qualitative and quantitative animation results of our method with StyleUNet and the baselines with MLPs and U-Net, respectively. First, it demonstrates that 2D CNNs are able to regress more detailed

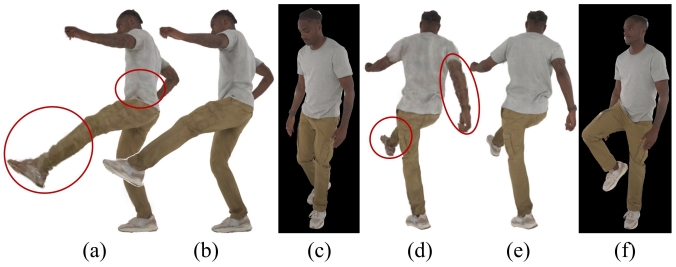

Fig. 14. **Ablation study of the pose projection strategy.** (a,d) and (b,e) are the animation results without and with the pose projection strategy, respectively. (c,f) are the reference images with the closest pose in the training dataset.

TABLE VIII
QUANTITATIVE ABLATION STUDY ON POSE PROJECTION.

	PSNR ↑	SSIM ↑	LPIPS ↓	FID ↓
w Pose Proj.	**24.9932**	**0.9285**	**0.0685**	**45.6266**
w/o Pose Proj.	23.5594	0.9189	0.0792	59.9083

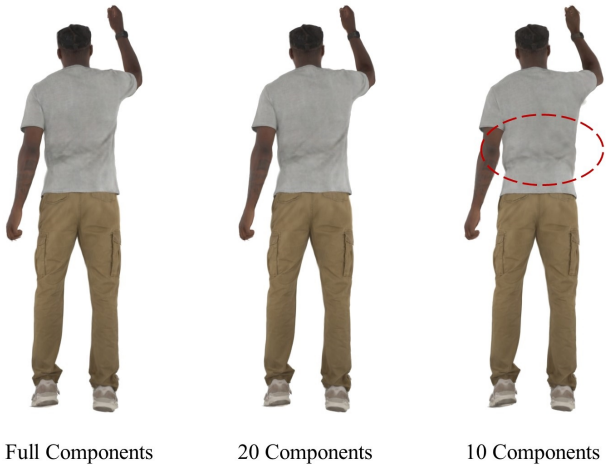

Fig. 15. **Ablation study on the component number in the pose projection.**

and realistic appearances, while MLPs suffer from limited representation ability, yielding blurry animation results. Second, StyleUNet outperforms the standard U-Net because of the additional modules (including style modulation and "To/From-RGB" modules) inherited from StyleGAN [14]. Overall, the 2D parameterization and StyleUNet enable our method to model high-quality human dynamic appearances.

3) Pose Projection: We evaluate the pose projection strategy by removing it, i.e., directly inputting the position map into the StyleUNet. Fig. 14 and Tab. VIII show the qualitative and quantitative animation results with and without the pose projection under novel poses, respectively. It demonstrates that direct extrapolation with the novel position map results in unreasonable 3D Gaussians, since no similar poses in the training dataset. In contrast, the pose projection guarantees that the reconstructed position maps (Eq. 17) lie within the distribution of training poses, leading to reasonable and vivid synthesized appearances.

4) Number of Principal Components in Pose Projection: Fig. 15 shows the animation results with different numbers

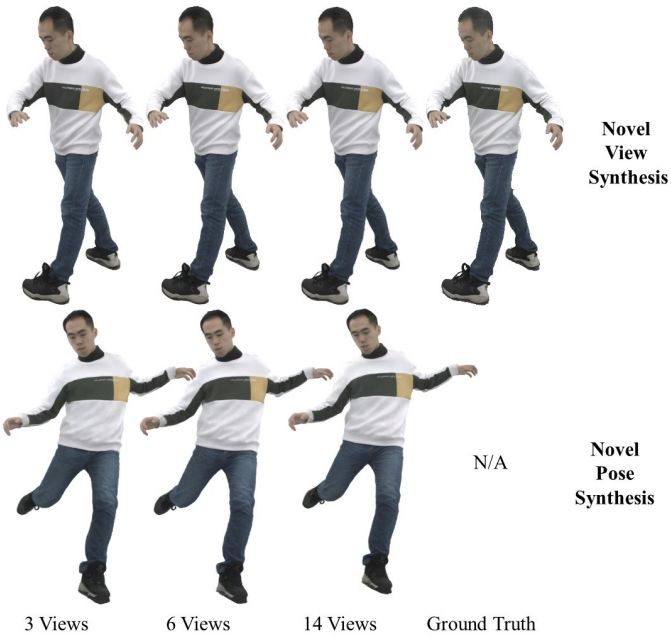

Fig. 16. **Animation results trained with different numbers of views.**

TABLE IX
QUANTITATIVE EVALUATION ON DIFFERENT VIEW NUMBERS.

	PSNR ↑	SSIM ↑	LPIPS ↓	FID ↓
3 Views	30.6123	0.9807	0.0306	11.3066
6 Views	30.3565	0.9803	0.0310	10.9966
14 Views	30.7622	0.9816	0.0297	10.6744

of principal components in the pose projection strategy. It demonstrates that although PCA can project a novel pose into the distribution of the training poses for better pose generalization as shown in Fig. 14, too few principal components may lose some fine-grained garment details. We empirically found that setting the number of principal components to 20 could produce both detailed and generalized animation.

5) View Number: We quantitatively and qualitatively show the animation results trained with 3 views, 6 views and 14 views in Tab. IX and Fig. 16. They demonstrate that our method also supports sparse-view input and can realize comparable high-fidelity results.

V. DISCUSSION

1) Conclusion: We present Animatable Gaussians, a new avatar representation for creating lifelike relightable and animatable human avatars with highly dynamic, realistic and generalized appearances from multi-view RGB videos. Compared with implicit NeRF-based approaches, we introduce the explicit point-based representation, 3D Gaussian splatting, into the avatar modeling, and leverage powerful 2D CNNs for modeling higher-fidelity human appearances. Based on the proposed template-guided parameterization and pose projection strategy, our method can not only faithfully reconstruct detailed human appearances, but also generate realistic garment dynamics for novel pose synthesis. By introducing physically-based rendering into the avatar representation, our

Fig. 17. **Failure cases.** (a,c) Animation results by our method, (b,d) ground-truth images. Our method fails to model the motion of hairs.

method can produce realistic avatar animation under different novel illuminations. Overall, our method outperforms other state-of-the-art avatar approaches, and we believe that the proposed 3D Gaussian splatting-based avatar representation will make progress towards effective and efficient 3D human representations.

2) Limitation: Our method entangles the modeling of the human body and clothes, limiting to changing the clothes of the avatar for applications like virtual try-on. A possible solution is to separately represent the body and clothes with multi-layer 3D Gaussians as NeRF-based approaches [50], [125]. Moreover, our method relies on the multi-view input to reconstruct a parametric template, limiting the application for modeling loose clothes from a monocular video. Finally, Our method fails to model the physical motion of components that are not driven by the body joints, e.g., the hairs, as illustrated in Fig. 17, since we model the whole body including clothes, hands and hairs as an entangled Gaussian representation. We leave for future work a disentangled and compositional representation for modeling the dynamics of different components of the character.

REFERENCES

[1] T. Bagautdinov, C. Wu, T. Simon, F. Prada, T. Shiratori, S.-E. Wei, W. Xu, Y. Sheikh, and J. Saragih, "Driving-signal aware full-body avatars," *TOG*, vol. 40, no. 4, pp. 1–17, 2021. 1, 2

[2] D. Xiang, T. Bagautdinov, T. Stuyck, F. Prada, J. Romero, W. Xu, S. Saito, J. Guo, B. Smith, T. Shiratori *et al.*, "Dressing avatars: Deep photorealistic appearance for physically simulated clothing," *TOG*, vol. 41, no. 6, pp. 1–15, 2022. 1, 2

[3] Q. Ma, J. Yang, S. Tang, and M. J. Black, "The power of points for modeling humans in clothing," in *ICCV*, 2021, pp. 10 974–10 984. 1, 3

[4] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "Nerf: Representing scenes as neural radiance fields for view synthesis," in *ECCV*. Springer, 2020, pp. 405–421. 1

[5] S. Peng, J. Dong, Q. Wang, S. Zhang, Q. Shuai, X. Zhou, and H. Bao, "Animatable neural radiance fields for modeling dynamic human bodies," in *ICCV*, 2021, pp. 14 314–14 323. 1, 2, 5

[6] L. Liu, M. Habermann, V. Rudnev, K. Sarkar, J. Gu, and C. Theobalt, "Neural actor: Neural free-view synthesis of human actors with pose control," *TOG*, vol. 40, no. 6, pp. 1–16, 2021. 1, 3, 5

[7] Z. Zheng, H. Huang, T. Yu, H. Zhang, Y. Guo, and Y. Liu, "Structured local radiance fields for human avatar modeling," in *CVPR*, 2022, pp. 15 893–15 903. 1, 3, 5, 7, 10, 11

[8] R. Li, J. Tanke, M. Vo, M. Zollhöfer, J. Gall, A. Kanazawa, and C. Lassner, "Tava: Template-free animatable volumetric actors," in *ECCV*. Springer, 2022, pp. 419–436. 1, 3, 10

[9] M. Tancik, P. Srinivasan, B. Mildenhall, S. Fridovich-Keil, N. Raghavan, U. Singhal, R. Ramamoorthi, J. Barron, and R. Ng, "Fourier features let networks learn high frequency functions in low dimensional domains," *NeurIPS*, vol. 33, pp. 7537–7547, 2020. 1, 3, 5, 11

[10] B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis, "3d gaussian splatting for real-time radiance field rendering," *TOG*, vol. 42, no. 4, pp. 1–14, 2023. 1, 3, 5

[11] S. Lin, H. Zhang, Z. Zheng, R. Shao, and Y. Liu, "Learning implicit templates for point-based clothed human modeling," in *ECCV*. Springer, 2022, pp. 210–228. 1, 3, 4

[12] Q. Ma, J. Yang, M. J. Black, and S. Tang, "Neural point-based shape modeling of humans in challenging clothing," in *3DV*. IEEE, 2022, pp. 679–689. 1, 3

[13] M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, and M. J. Black, "Smpl: A skinned multi-person linear model," *TOG*, vol. 34, no. 6, pp. 1–16, 2015. 1, 2

[14] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," in *CVPR*, 2019, pp. 4401–4410. 1, 10, 14

[15] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila, "Analyzing and improving the image quality of stylegan," in *CVPR*, 2020, pp. 8110–8119. 1

[16] T. Karras, M. Aittala, S. Laine, E. Härkönen, J. Hellsten, J. Lehtinen, and T. Aila, "Alias-free generative adversarial networks," *NeurIPS*, vol. 34, pp. 852–863, 2021. 1

[17] L. Wang, X. Zhao, J. Sun, Y. Zhang, H. Zhang, T. Yu, and Y. Liu, "Styleavatar: Real-time photo-realistic portrait avatar from a single video," in *SIGGRAPH Conference Proceedings*, 2023. 1, 4, 5, 10, 13

[18] Z. Li, Z. Zheng, L. Wang, and Y. Liu, "Animatable gaussians: Learning pose-dependent gaussian maps for high-fidelity human avatar modeling," in *CVPR*, 2024. 2

[19] H. Jin, I. Liu, P. Xu, X. Zhang, S. Han, S. Bi, X. Zhou, Z. Xu, and H. Su, "Tensoir: Tensorial inverse rendering," in *CVPR*, 2023, pp. 165–174. 2

[20] J. Gao, C. Gu, Y. Lin, H. Zhu, X. Cao, L. Zhang, and Y. Yao, "Relightable 3d gaussian: Real-time point cloud relighting with brdf decomposition and ray tracing," *arXiv preprint arXiv:2311.16043*, 2023. 2, 6, 7

[21] C. Stoll, J. Gall, E. De Aguiar, S. Thrun, and C. Theobalt, "Video-based reconstruction of animatable human characters," *TOG*, vol. 29, no. 6, pp. 1–10, 2010. 2

[22] P. Guan, L. Reiss, D. A. Hirshberg, A. Weiss, and M. J. Black, "Drape: Dressing any person," *TOG*, vol. 31, no. 4, pp. 1–10, 2012. 2

[23] F. Xu, Y. Liu, C. Stoll, J. Tompkin, G. Bharaj, Q. Dai, H.-P. Seidel, J. Kautz, and C. Theobalt, "Video-based characters: creating new human performances from a multi-view video database," *TOG*, vol. 30, no. 4, pp. 1–10, 2011. 2

[24] D. Xiang, F. Prada, T. Bagautdinov, W. Xu, Y. Dong, H. Wen, J. Hodgins, and C. Wu, "Modeling clothing as a separate layer for an animatable human avatar," *TOG*, vol. 40, no. 6, pp. 1–15, 2021. 2

[25] O. Halimi, T. Stuyck, D. Xiang, T. Bagautdinov, H. Wen, R. Kimmel, T. Shiratori, C. Wu, Y. Sheikh, and F. Prada, "Pattern-based cloth registration and sparse-view animation," *TOG*, vol. 41, no. 6, pp. 1–17, 2022. 2

[26] M. Habermann, L. Liu, W. Xu, M. Zollhoefer, G. Pons-Moll, and C. Theobalt, "Real-time deep dynamic characters," *TOG*, vol. 40, no. 4, pp. 1–16, 2021. 2

[27] M. Habermann, L. Liu, W. Xu, G. Pons-Moll, M. Zollhoefer, and C. Theobalt, "Hdhumans: A hybrid approach for high-fidelity digital humans," *ACM SCA*, vol. 6, no. 3, pp. 1–23, 2023. 2

[28] R. W. Sumner, J. Schmid, and M. Pauly, "Embedded deformation for shape manipulation," *TOG*, vol. 26, no. 3, pp. 80–es, 2007. 2

[29] Y. Kwon, L. Liu, H. Fuchs, M. Habermann, and C. Theobalt, "Deliffas: Deformable light fields for fast avatar synthesis," in *NeurIPS*, 2023. 2

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

[31] ——, "Detailed human avatars from monocular video," in *3DV*. IEEE, 2018, pp. 98–109. 2

[32] H. Zhao, J. Zhang, Y.-K. Lai, Z. Zheng, Y. Xie, Y. Liu, and K. Li, "High-fidelity human avatars from a single rgb camera," in *CVPR*, 2022, pp. 15 904–15 913. 2

[33] A. Burov, M. Nießner, and J. Thies, "Dynamic surface function networks for clothed human bodies," in *ICCV*, 2021, pp. 10 754–10 764. 2

[34] H. Kim, H. Nam, J. Kim, J. Park, and S. Lee, "Laplacianfusion: Detailed 3d clothed-human body reconstruction," *ACM Transactions on Graphics (TOG)*, vol. 41, no. 6, pp. 1–14, 2022. 2

[35] B. Jiang, Y. Hong, H. Bao, and J. Zhang, "Selfrecon: Self reconstruction your digital avatar from monocular video," in *CVPR*, 2022, pp. 5605–5615. 2

[36] Z. Xu, S. Peng, C. Geng, L. Mou, Z. Yan, J. Sun, H. Bao, and X. Zhou, "Relightable and animatable neural avatar from sparse-view video," *arXiv preprint arXiv:2308.07903*, 2023. 2, 3

[37] M. Mihajlovic, Y. Zhang, M. J. Black, and S. Tang, "Leap: Learning articulated occupancy of people," in *CVPR*, 2021, pp. 10 461–10 471. 2

[38] S. Saito, J. Yang, Q. Ma, and M. J. Black, "Scanimate: Weakly supervised learning of skinned clothed avatar networks," in *CVPR*, 2021, pp. 2886–2897. 2

[39] S. Wang, M. Mihajlovic, Q. Ma, A. Geiger, and S. Tang, "Metaavatar: Learning animatable clothed human models from few depth images," *NeurIPS*, vol. 34, 2021. 2

[40] G. Tiwari, N. Sarafianos, T. Tung, and G. Pons-Moll, "Neural-gif: Neural generalized implicit functions for animating people in clothing," in *ICCV*, 2021, pp. 11 708–11 718. 2

[41] Z. Dong, C. Guo, J. Song, X. Chen, A. Geiger, and O. Hilliges, "Pina: Learning a personalized implicit neural avatar from a single rgb-d video sequence," in *CVPR*, 2022. 2

[42] H.-I. Ho, L. Xue, J. Song, and O. Hilliges, "Learning locally editable virtual humans," in *CVPR*, 2023, pp. 21 024–21 035. 2

[43] B. Deng, J. P. Lewis, T. Jeruzalski, G. Pons-Moll, G. Hinton, M. Norouzi, and A. Tagliasacchi, "Nasa neural articulated shape approximation," in *ECCV*. Springer, 2020, pp. 612–628. 2

[44] X. Chen, Y. Zheng, M. J. Black, O. Hilliges, and A. Geiger, "Snarf: Differentiable forward skinning for animating non-rigid neural implicit shapes," in *ICCV*, 2021, pp. 11 594–11 604. 2, 4, 9

[45] X. Chen, T. Jiang, J. Song, M. Rietmann, A. Geiger, M. J. Black, and O. Hilliges, "Fast-snarf: A fast deformer for articulated neural fields," *IEEE T-PAMI*, 2023. 2

[46] M. Mihajlovic, S. Saito, A. Bansal, M. Zollhoefer, and S. Tang, "Coap: Compositional articulated occupancy of people," in *CVPR*, 2022, pp. 13 201–13 210. 2

[47] Z. Li, Z. Zheng, H. Zhang, C. Ji, and Y. Liu, "Avatarcap: Animatable avatar conditioned monocular human volumetric capture," in *ECCV*. Springer, 2022, pp. 322–341. 2

[48] S. Peng, S. Zhang, Z. Xu, C. Geng, B. Jiang, H. Bao, and X. Zhou, "Animatable neural implicit surfaces for creating avatars from videos," *arXiv preprint arXiv:2203.08133*, 2022. 2

[49] S.-Y. Su, F. Yu, M. Zollhöfer, and H. Rhodin, "A-nerf: Articulated neural radiance fields for learning human shape, appearance, and pose," *NeurIPS*, vol. 34, pp. 12 278–12 291, 2021. 2

[50] Y. Feng, J. Yang, M. Pollefeys, M. J. Black, and T. Bolkart, "Capturing and animation of body and clothing from monocular video," in *SIGGRAPH Asia 2022 Conference Proceedings*, ser. SA '22, 2022. 2, 15

[51] C.-Y. Weng, B. Curless, P. P. Srinivasan, J. T. Barron, and I. Kemelmacher-Shlizerman, "Humannerf: Free-viewpoint rendering of moving people from monocular video," in *CVPR*, 2022, pp. 16 210–16 220. 2

[52] G. Te, X. Li, X. Li, J. Wang, W. Hu, and Y. Lu, "Neural capture of animatable 3d human from monocular video," in *ECCV*. Springer, 2022, pp. 275–291. 2

[53] B. Peng, J. Hu, J. Zhou, and J. Zhang, "Selfnerf: Fast training nerf for human from monocular self-rotating video," *arXiv preprint arXiv:2210.01651*, 2022. 2

[54] C. Guo, T. Jiang, X. Chen, J. Song, and O. Hilliges, "Vid2avatar: 3d avatar reconstruction from videos in the wild via self-supervised scene decomposition," in *CVPR*, 2023, pp. 12 858–12 868. 2

[55] T. Jiang, X. Chen, J. Song, and O. Hilliges, "Instantavatar: Learning avatars from monocular video in 60 seconds," in *CVPR*, 2023, pp. 16 922–16 932. 2

[56] W. Jiang, K. M. Yi, G. Samei, O. Tuzel, and A. Ranjan, "Neuman: Neural human radiance field from a single video," in *ECCV*. Springer, 2022, pp. 402–418. 2

[57] Y. Chen, X. Wang, X. Chen, Q. Zhang, X. Li, Y. Guo, J. Wang, and F. Wang, "Uv volumes for real-time rendering of editable free-view human performance," in *CVPR*, 2023, pp. 16 621–16 631. 3

[58] R. A. Güler, N. Neverova, and I. Kokkinos, "Densepose: Dense human pose estimation in the wild," in *CVPR*, 2018, pp. 7297–7306. 3

[59] S. Wang, K. Schwarz, A. Geiger, and S. Tang, "Arah: Animatable volume rendering of articulated human sdfs," in *ECCV*. Springer, 2022, pp. 1–19. 3, 9, 10

[60] L. Yariv, J. Gu, Y. Kasten, and Y. Lipman, "Volume rendering of neural implicit surfaces," *NeurIPS*, vol. 34, pp. 4805–4815, 2021. 3, 4, 5

[61] P. Wang, L. Liu, Y. Liu, C. Theobalt, T. Komura, and W. Wang, "Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction," in *NeurIPS*, 2021. 3

[62] S.-Y. Su, T. Bagautdinov, and H. Rhodin, "Danbo: Disentangled articulated neural body representations via graph neural networks," in *ECCV*. Springer, 2022, pp. 107–124. 3

[63] Z. Li, Z. Zheng, Y. Liu, B. Zhou, and Y. Liu, "Posevocab: Learning joint-structured pose embeddings for human avatar modeling," in *ACM SIGGRAPH Conference Proceedings*, 2023. 3, 10

[64] J. Dong, Q. Fang, Y. Guo, S. Peng, Q. Shuai, X. Zhou, and H. Bao, "Totalselfscan: Learning full-body avatars from self-portrait videos of faces, hands, and bodies," in *NeurIPS*, 2022. 3, 11

[65] K. Shen, C. Guo, M. Kaufmann, J. J. Zarate, J. Valentin, J. Song, and O. Hilliges, "X-avatar: Expressive human avatars," in *CVPR*, 2023, pp. 16 911–16 921. 3, 11

[66] Z. Zheng, X. Zhao, H. Zhang, B. Liu, and Y. Liu, "Avatarrex: Real-time expressive full-body avatars," *TOG*, vol. 42, no. 4, 2023. 3, 7, 10, 11, 13

[67] Q. Ma, S. Saito, J. Yang, S. Tang, and M. J. Black, "Scale: Modeling clothed humans with a surface codec of articulated local elements," in *CVPR*, 2021, pp. 16 082–16 093. 3

[68] H. Zhang, S. Lin, R. Shao, Y. Zhang, Z. Zheng, H. Huang, Y. Guo, and Y. Liu, "Closet: Modeling clothed humans on continuous surface with explicit template decomposition," in *CVPR*, 2023. 3

[69] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "Pointnet: Deep learning on point sets for 3d classification and segmentation," in *CVPR*, 2017, pp. 652–660. 3

[70] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "Pointnet++: Deep hierarchical feature learning on point sets in a metric space," in *NeurIPS*, 2017. 3

[71] S. Prokudin, Q. Ma, M. Raafat, J. Valentin, and S. Tang, "Dynamic point fields," in *ICCV*, 2023, pp. 7964–7976. 3

[72] S.-Y. Su, T. Bagautdinov, and H. Rhodin, "Npc: Neural point characters from video," in *ICCV*, 2023. 3

[73] Q. Xu, Z. Xu, J. Philip, S. Bi, Z. Shu, K. Sunkavalli, and U. Neumann, "Point-nerf: Point-based neural radiance fields," in *CVPR*, 2022, pp. 5438–5448. 3

[74] H. Pfister, M. Zwicker, J. Van Baar, and M. Gross, "Surfels: Surface elements as rendering primitives," in *SIGGRAPH*, 2000, pp. 335–342. 3

[75] M. Zwicker, H. Pfister, J. Van Baar, and M. Gross, "Surface splatting," in *SIGGRAPH*, 2001, pp. 371–378. 3, 4

[76] M. Zwicker, M. Pauly, O. Knoll, and M. Gross, "Pointshop 3d: An interactive system for point-based surface editing," *TOG*, vol. 21, no. 3, pp. 322–329, 2002. 3

[77] M. Zwicker, J. Rasanen, M. Botsch, C. Dachsbacher, and M. Pauly, "Perspective accurate splatting," in *Proceedings-Graphics Interface*, 2004, pp. 247–254. 3

[78] W. Yifan, F. Serena, S. Wu, C. Öztireli, and O. Sorkine-Hornung, "Differentiable surface splatting for point-based geometry processing," *TOG*, vol. 38, no. 6, pp. 1–14, 2019. 3

[79] K.-A. Aliev, A. Sevastopolsky, M. Kolos, D. Ulyanov, and V. Lempitsky, "Neural point-based graphics," in *ECCV*. Springer, 2020, pp. 696–712. 3

[80] C. Lassner and M. Zollhofer, "Pulsar: Efficient sphere-based neural rendering," in *CVPR*, 2021, pp. 1440–1449. 3

[81] G. Kopanas, J. Philip, T. Leimkühler, and G. Drettakis, "Point-based neural rendering with per-view optimization," in *Computer Graphics Forum*, vol. 40, no. 4. Wiley Online Library, 2021, pp. 29–43. 3

[82] D. Rückert, L. Franke, and M. Stamminger, "Adop: Approximate differentiable one-pixel point rendering," *TOG*, vol. 41, no. 4, pp. 1–14, 2022. 3

[83] Y. Zheng, W. Yifan, G. Wetzstein, M. J. Black, and O. Hilliges, "Pointavatar: Deformable point-based head avatars from videos," in *CVPR*, 2023, pp. 21 057–21 067. 3

[84] N. Ravi, J. Reizenstein, D. Novotny, T. Gordon, W.-Y. Lo, J. Johnson, and G. Gkioxari, "Accelerating 3d deep learning with pytorch3d," *arXiv preprint arXiv:2007.08501*, 2020. 3

[85] J. Luiten, G. Kopanas, B. Leibe, and D. Ramanan, "Dynamic 3d gaussians: Tracking by persistent dynamic view synthesis," *arXiv preprint arXiv:2308.09713*, 2023. 3

[86] Z. Yang, X. Gao, W. Zhou, S. Jiao, Y. Zhang, and X. Jin, "Deformable 3d gaussians for high-fidelity monocular dynamic scene reconstruction," *arXiv preprint arXiv:2309.13101*, 2023. 3

[87] G. Wu, T. Yi, J. Fang, L. Xie, X. Zhang, W. Wei, W. Liu, Q. Tian, and X. Wang, "4d gaussian splatting for real-time dynamic scene rendering," *arXiv preprint arXiv:2310.08528*, 2023. 3

[88] Z. Yang, H. Yang, Z. Pan, X. Zhu, and L. Zhang, "Real-time photorealistic dynamic scene representation and rendering with 4d gaussian splatting," *arXiv preprint arXiv:2310.10642*, 2023. 3

[89] W. Zielonka, T. Bagautdinov, S. Saito, M. Zollhöfer, J. Thies, and J. Romero, "Drivable 3d gaussian avatars," *arXiv preprint arXiv:2311.08581*, 2023. 3

[90] J. Lei, Y. Wang, G. Pavlakos, L. Liu, and K. Daniilidis, "Gart: Gaussian articulated template models," in *CVPR*, 2024. 3

[91] Z. Qian, S. Wang, M. Mihajlovic, A. Geiger, and S. Tang, "3dgs-avatar: Animatable avatars via deformable 3d gaussian splatting," in *CVPR*, 2024. 3, 10, 11, 13

[92] S. Hu and Z. Liu, "Gauhuman: Articulated gaussian splatting from monocular human videos," in *CVPR*, 2024. 3

[93] M. Kocabas, J.-H. R. Chang, J. Gabriel, O. Tuzel, and A. Ranjan, "Hugs: Human gaussian splats," in *CVPR*, 2024. 3

[94] H. Pang, H. Zhu, A. Kortylewski, C. Theobalt, and M. Habermann, "Ash: Animatable gaussian splats for efficient and photoreal human rendering," in *CVPR*, 2024. 3

[95] L. Hu, H. Zhang, Y. Zhang, B. Zhou, B. Liu, S. Zhang, and L. Nie, "Gaussianavatar: Towards realistic human avatar modeling from a single video via animatable 3d gaussians," in *CVPR*, 2024. 3, 10, 11, 13

[96] C.-F. Chabert, P. Einarsson, A. Jones, B. Lamond, W.-C. Ma, S. Sylwan, T. Hawkins, and P. Debevec, "Relighting human locomotion with flowed reflectance fields," in *ACM SIGGRAPH 2006 Sketches*, 2006, pp. 76–es. 3

[97] P. Debevec, "The light stages and their applications to photoreal digital actors," *SIGGRAPH Asia*, vol. 2, no. 4, pp. 1–6, 2012. 3

[98] P. Debevec, T. Hawkins, C. Tchou, H.-P. Duiker, W. Sarokin, and M. Sagar, "Acquiring the reflectance field of a human face," in *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, 2000, pp. 145–156. 3

[99] P. Debevec, A. Wenger, C. Tchou, A. Gardner, J. Waese, and T. Hawkins, "A lighting reproduction approach to live-action compositing," *TOG*, vol. 21, no. 3, pp. 547–556, 2002. 3

[100] K. Guo, P. Lincoln, P. Davidson, J. Busch, X. Yu, M. Whalen, G. Harvey, S. Orts-Escolano, R. Pandey, J. Dourgarian *et al.*, "The relightables: Volumetric performance capture of humans with realistic relighting," *TOG*, vol. 38, no. 6, pp. 1–19, 2019. 3

[101] T. Hawkins, J. Cohen, and P. Debevec, "A photometric approach to digitizing cultural artifacts," in *Proceedings of the 2001 conference on Virtual reality, archeology, and cultural heritage*, 2001, pp. 333–342. 3

[102] A. Wenger, A. Gardner, C. Tchou, J. Unger, T. Hawkins, and P. Debevec, "Performance relighting and reflectance transformation with time-multiplexed illumination," *TOG*, vol. 24, no. 3, pp. 756–764, 2005. 3

[103] T. Weyrich, W. Matusik, H. Pfister, B. Bickel, C. Donner, C. Tu, J. McAndless, J. Lee, A. Ngan, H. W. Jensen *et al.*, "Analysis of human faces using a measurement-based skin reflectance model," *TOG*, vol. 25, no. 3, pp. 1013–1024, 2006. 3

[104] Z. Chen and Z. Liu, "Relighting4d: Neural relightable human from videos," in *ECCV*. Springer, 2022, pp. 606–623. 3, 12, 13

[105] U. Iqbal, A. Caliskan, K. Nagano, S. Khamis, P. Molchanov, and J. Kautz, "Rana: Relightable articulated neural avatars," in *ICCV*, 2023, pp. 23 142–23 153. 3

[106] W. Sun, Y. Che, H. Huang, and Y. Guo, "Neural reconstruction of relightable human model from monocular video," in *ICCV*, 2023, pp. 397–407. 3

[107] W. Lin, C. Zheng, J.-H. Yong, and F. Xu, "Relightable and animatable neural avatars from videos," in *AAAI*, vol. 38, no. 4, 2024, pp. 3486–3494. 3, 12, 13

[108] S. Wang, B. Antić, A. Geiger, and S. Tang, "Intrinsicavatar: Physically based inverse rendering of dynamic humans from monocular videos via explicit ray tracing," *arXiv preprint arXiv:2312.05210*, 2023. 3

[109] S. Peng, Y. Zhang, Y. Xu, Q. Wang, Q. Shuai, H. Bao, and X. Zhou, "Neural body: Implicit neural representations with structured latent codes for novel view synthesis of dynamic humans," in *CVPR*, 2021, pp. 9054–9063. 3

[110] G. Pavlakos, V. Choutas, N. Ghorbani, T. Bolkart, A. A. A. Osman, D. Tzionas, and M. J. Black, "Expressive body capture: 3d hands, face, and body from a single image," in *CVPR*, 2019. 4, 7, 13

[111] W. E. Lorensen and H. E. Cline, "Marching cubes: A high resolution 3d surface construction algorithm," *TOG*, vol. 21, no. 4, pp. 163–169, 1987. 4

[112] J. T. Kajiya, "The rendering equation," in *Proceedings of the 13th annual conference on Computer graphics and interactive techniques*, 1986, pp. 143–150. 5

[113] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *CVPR*, 2018, pp. 586–595. 6, 9

[114] S. Saito, T. Simon, J. Saragih, and H. Joo, "Pifuhd: Multi-level pixel-aligned implicit function for high-resolution 3d human digitization," in *CVPR*, June 2020. 6

[115] X. Deng, Z. Zheng, Y. Zhang, J. Sun, C. Xu, x. Yang, L. Wang, and Y. Liu, "Ram-avatar: Real-time photo-realistic avatar from monocular videos with full-body control," in *CVPR*, 2024. 6

[116] N. Mahmood, N. Ghorbani, N. F. Troje, G. Pons-Moll, and M. J. Black, "Amass: Archive of motion capture as surface shapes," in *ICCV*, 2019, pp. 5442–5451. 7, 8

[117] M. Işık, M. Rünz, M. Georgopoulos, T. Khakhulin, J. Starck, L. Agapito, and M. Nießner, "Humanrf: High-fidelity neural radiance fields for humans in motion," *TOG*, vol. 42, no. 4, pp. 1–12, 2023. 7, 13

[118] Y. Zhang, Z. Li, L. An, M. Li, T. Yu, and Y. Liu, "Lightweight multi-person total motion capture using sparse multi-view cameras," in *ICCV*, 2021, pp. 5560–5569. 7

[119] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE T-IP*, vol. 13, no. 4, pp. 600–612, 2004. 9

[120] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "Gans trained by a two time-scale update rule converge to a local nash equilibrium," *NeurIPS*, vol. 30, 2017. 9

[121] C. G. Broyden, "A class of methods for solving nonlinear simultaneous equations," *Mathematics of computation*, vol. 19, no. 92, pp. 577–593, 1965. 10

[122] A. Gropp, L. Yariv, N. Haim, M. Atzmon, and Y. Lipman, "Implicit geometric regularization for learning shapes," in *ICML*. PMLR, 2020, pp. 3789–3799. 10

[123] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *ICLR*, 2015. 10

[124] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *MICCAI*. Springer, 2015, pp. 234–241. 13

[125] Y. Feng, W. Liu, T. Bolkart, J. Yang, M. Pollefeys, and M. J. Black, "Learning disentangled avatars with hybrid 3d representations," *arXiv preprint arXiv:2309.06441*, 2023. 15