Relightable and Animatable Neural Avatar from Sparse-View Video

Zhen Xu [1] Sida Peng [1*] Chen Geng [1,2] Linzhan Mou [1]
Zihan Yan [3] Jiaming Sun [1] Hujun Bao [1] Xiaowei Zhou [1]

[1]Zhejiang University [2]Stanford University [3]University of Illinois Urbana-Champaign

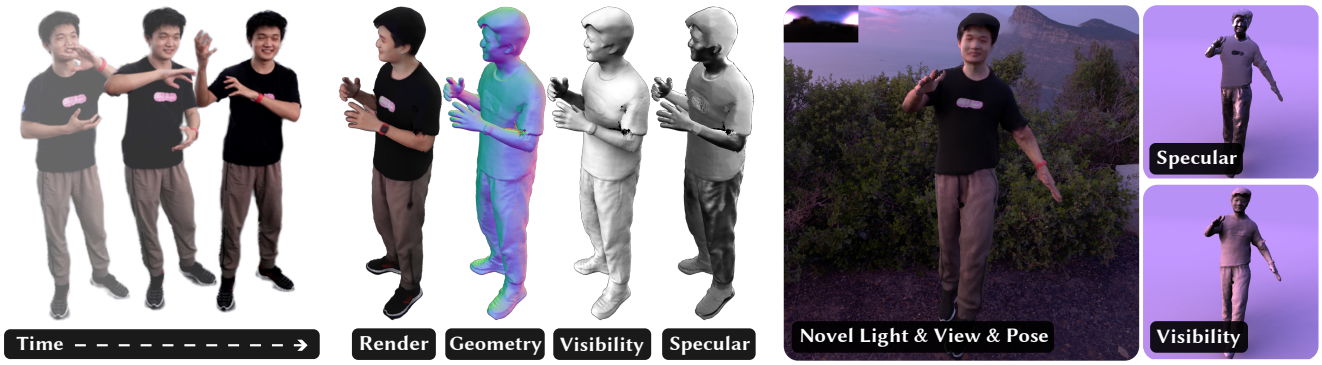

Figure 1. **Reconstructing relightable and animatable neural avatar from sparse-view (or monocular) video.** Our method takes only a sparse-view (or monocular) video as input and reconstructs a relightable and animatable neural avatar under unknown illumination, which can then be relit with arbitrary environment lights and animated with arbitrary motion sequences. **Note that our method successfully captures the shininess of the skin and pants as well as the specular highlights on the t-shirt's plastisol printings.**

Abstract

This paper tackles the problem of creating relightable and animatable neural avatars from sparse-view (or monocular) videos of dynamic humans under unknown illumination. Previous neural human reconstruction methods produce animatable avatars from sparse views using deformed Signed Distance Fields (SDF) but are non-relightable. While differentiable inverse rendering methods have succeeded in the material recovery of static objects, it is not straightforward to extend them to dynamic humans since it is computationally intensive to compute pixel-surface intersection and light visibility on deformed SDFs for relighting. To solve this challenge, we propose a Hierarchical Distance Query (HDQ) algorithm to approximate the world space SDF under arbitrary human poses. Specifically, we estimate coarse SDF based on a parametric human model and compute fine SDF by exploiting the invariance of SDF w.r.t. local deformation. Based on HDQ, we leverage sphere tracing to efficiently estimate the surface intersection and light visibility. This allows us to develop the first system to recover relightable and animatable neural avatars from sparse or monocular inputs. Experiments show that our approach produces superior results compared to state-of-the-art methods. Our project page is available at https://zju3dv.github.io/relightable_avatar.

The authors from Zhejiang University are affiliated with the State Key Lab of CAD&CG. *Corresponding author: Sida Peng.

1. Introduction

Realistic human avatars have a range of applications [13, 55] in various domains, e.g., virtual reality, filmmaking, and video games. This work focuses on the specific setting of creating animatable and relightable human avatars from sparse-view or monocular RGB videos. This problem is challenging due to the inherent ambiguity of acquiring human geometry, materials, and motions from sparse view images [20, 46]. Traditional methods [17, 19, 20, 22, 32, 55, 62] resolve this ambiguity via customized and costly capture devices, e.g., light stages with controllable illumination and dense camera arrays. However, such devices are restricted to professional users, impeding their universality and generalization.

Recent neural scene representation-based methods [38, 45, 60] have demonstrated the ability to extract detailed geometry and photorealistic appearance of human performers from sparse-view videos without sophisticated studio setup. These methods typically define the human model in canonical space and warp it into world space through a deformation module to represent human performers observed in videos. For example, AniSDF [45] models the human geometry and appearance as neural signed distance and radiance fields, and deforms them using linear blend skinning (LBS) [35] and learned local deformation networks. Albeit showing the capability of novel pose synthesis, the reconstructed avatars in these works [38, 44, 60] are not relightable as they

bake the shading and shadow into the appearance model. As a result, the shading of the avatars under novel poses is unrealistic and the environment illumination cannot be changed, which restricts the applicability of the avatars.

Another line of works attempts to create relightable models under natural illumination through inverse rendering techniques [10, 53, 67, 70, 71], which estimate surface material parameters from input images through differentiable physically-based rendering. Computing the visibility of 3D points to the environment light is essential for accurate estimation [70, 71], but the cost of visibility computation is high. To improve efficiency, L-Tracing [15] adopts a signed distance field to represent the scene geometry and estimates the light visibility through sphere tracing, which iteratively marches along a ray using distance values until hitting the surface. Although this strategy works well on static objects, it is not suitable for animatable neural avatars [45, 60, 63], which warp the canonical SDF to world space based on a non-rigid motion field, producing a deformed SDF. The reason is that sphere tracing might not converge on the deformed SDF [51] since the SDF is inherently defined in the canonical space, thereby yielding incorrect world-space distance.

In this work, we propose a novel approach for creating relightable and animatable human avatars from sparse-view (or monocular) videos via neural inverse rendering. Inspired by previous methods [45, 60], we parameterize the human avatar as MLP networks, which predict material parameters and signed distance for any 3D point in canonical space. These values are transformed into world space for rendering through a neural deformation field. Our innovation lies in designing a hierarchical query scheme that enables a consistent approximation of 3D points' distance to the surface of the neural avatar under arbitrary human poses. This allows us to perform sphere tracing for 3D points' pixel-surface intersection and light visibility for physically-based rendering. Specifically, we smoothly blend the world-space KNN (when query points are far from the surface) distances and canonical-space neural SDF (when query points are close to the surface), approximating an SDF defined on the world-space geometry of the neural avatar. In this way, vanilla sphere tracing [25] can be performed on the deformed SDF in world space when animating and relighting the avatar, avoiding the non-linearity of canonical sphere tracing, as well as the pitfalls of world space tracing with incorrect world-space distance.

Based on the Hierarchical Distance Query algorithm, we further develop a soft visibility computation scheme by incorporating traditional distance field soft shadow (DFSS) [43] onto the deformed SDF, which is essential to the photorealism of the relightable neural avatar. The soft shadow produced by an area light source typically requires multiple light samples to compute, while DFSS utilizes distance values to approximate the soft shadow coefficient with only a single sample. Note that it is not trivial to combine DFSS with previous methods [44, 60, 63], as they cannot produce world-space distance values from 3D points to the scene surface along an arbitrary direction.

To validate our approach, we collect a real-world multi-view dataset dubbed *MobileStage*, which captures the complex shading and shadow effects of dynamic humans with an array of mobile phone cameras. Furthermore, we extend the *SyntheticHuman* dataset [45] with novel illuminations, enabling the evaluation of relightable neural avatars with ground-truth photometric properties and relighting results. Experiments on relighting ability and novel pose synthesis show that our method outperforms the state-of-the-art with superior visual quality and physical accuracy on both real-world and synthetic datasets. Our code will be made publicly available for reproducibility.

Our contributions can be summarized as follows: (a) We propose a novel system for reconstructing relightable and animatable neural avatars from sparse-view (or monocular) videos. (b) We design a hierarchical distance query algorithm for efficient pixel-surface intersection and light visibility computation using sphere tracing. (c) We extend DFSS to drivable neural SDF, efficiently producing realistic soft shadows for the neural avatars. (d) We demonstrate quantitative and qualitative improvements compared to prior work.

2. Related work

Human avatars. To produce animatable human avatars, previous methods [13, 23, 24, 55, 58, 64] generally adopt a three-stage pipeline: they first reconstruct the human shape and appearance, then bind the shape to a skeleton, and finally animate the human model through linear blend skinning (LBS) algorithm [35]. Traditional methods tend to leverage complicated hardware, such as dense camera arrays [17, 21, 32, 54, 55] or depth sensors [3, 9, 52, 57], to create high-fidelity human models. Recently, some optimization-based methods [4, 30, 46, 61, 63] have attempted to reconstruct human models given sparse multi-view videos. For example, Neural Body [46] represents a dynamic human model by combining SMPL model [39] with neural radiance field (NeRF) [41]. Its model parameters are learned from images through differentiable volume rendering.

Animating human avatars. To animate the reconstructed human model, some [4, 26] retrieve the skinning weights of the SMPL model for performing the LBS algorithm. Several methods [14, 26, 44, 49] opt to optimize personalized skinning weights for the target human subject, where they represent the skinning weights as an MLP network and learn it from input data, such as human shapes [14, 49] or multi-view videos [26, 37, 44]. Another line of works [38, 45] introduces a neural displacement field to improve animation realism. The articulated deformation is represented by the

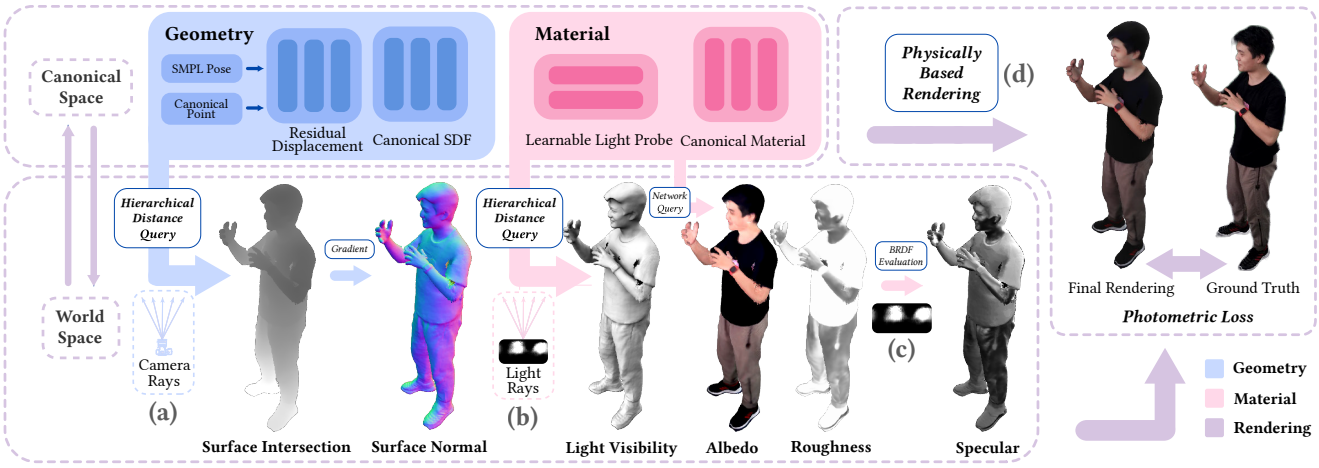

Figure 2. **Overview of the proposed sparse-view relightable and animatable human avatar.** (a) Given world space camera rays, we perform sphere tracing on the hierarchically queried distances (Sec. 3.2) to find surface intersections and canonical correspondences (Sec. 3.3). (b) Light rays generated by an optimizable light probe are also sphere traced with HDQ to compute the closest distances along the ray for soft visibility (Sec. 3.3). (c) Material properties (Sec. 3.4) and surface normals are queried on the canonical correspondences and warped to world space. (d) Then, the final pixel colors are computed using the rendering equation (Sec. 3.5).

LBS model of SMPL, and the non-rigid deformation is predicted using an MLP network. While neural animatable methods can produce dynamic avatars that appear realistic, they do not model the material properties of the avatars, making them unable to adapt to different lighting conditions.

Relighting human avatars. To relight objects, a typical approach is first acquiring their material properties and then rendering with new illumination through physically-based rendering. Traditional methods [20, 50] mostly require a known illumination for calculating the material parameters through photometric stereos. Light stage-based approaches [19, 20, 22, 62] build a controllable light array to capture images of target subjects under multiple illuminations. Based on these captured images and the known illuminations, they solve for the unknown material properties. Following this setting, some methods [8, 28, 36] achieve photorealistic relighting results by adopting a neural renderer to implicit learn the relightable appearance model from light-stage images. However, these methods typically require the geometry to be known a priori. More recently, neural inverse rendering methods [7, 10–12, 15, 34, 53, 56, 65, 67, 70, 71] explore more flexible capture settings, where the illumination is unknown or even variable. Motivated by its potential for many human-centric applications, research on human relighting has been widely conducted in the literature [40, 42, 66, 69]. Same as other objects, the material properties of human subjects can be recovered using neural inverse rendering methods. The difference is that human subjects exhibit more strong material priors. Therefore, some methods [5, 27, 29, 33, 42, 66] attempt to train neural networks to predict human materials from a single image. Recently, Relighting4D [16] have attempted to acquire human materials from sparse multi-view videos. However, Relighting4D is not designed to relight animatable avatars realistically, limiting its applicability.

3. Method

Given a sparse-view (or monocular) video of a human performer under natural and unknown illumination, we learn to reconstruct the drivable geometry and photometric properties of the human performer to create an animatable and relightable neural avatar. We assume the human poses and the foreground masks are provided as in [38, 44–46].

3.1. Overview

An overview of the relightable and animatable avatar can be found in Fig. 2. We formulate the relightable and animatable avatar using a set of canonical space neural fields and a warping between world and canonical space defined by the linear blend skinning algorithm [35] and a displacement field [38, 44, 45, 61]. In the canonical space, we define a set of geometry ($S(\boldsymbol{x})$) and material neural fields ($A(\boldsymbol{x})$ and $\Gamma(\boldsymbol{x})$) for the animated human model. The canonical space displacement field $F_{\Delta\boldsymbol{x}}$ provides additional pose-dependent deformation on top of SMPL inverse LBS. More details about the warping process are provided in Sec. 3.2.

The relightable and animatable neural avatar will be rendered by casting camera rays in world space and finding the surface intersection points $\boldsymbol{x}_s$ and their normals $\boldsymbol{n}_s$ using the Hierarchical Distance Query (HDQ) algorithm, whose material properties albedo α_s and roughness γ_s can be obtained from the canonical material MLP networks, composing the BRDF model R_s. Light visibility V_s can be computed by performing HDQ sphere tracing on all

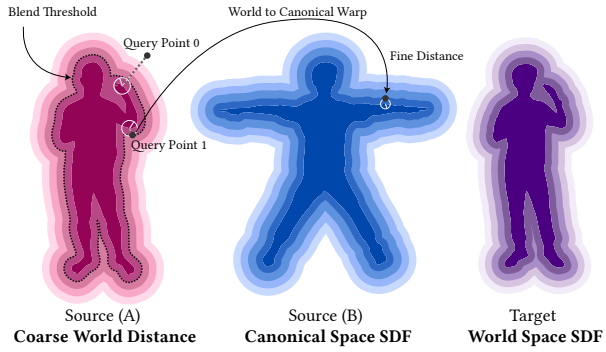

Figure 3. **Illustration of Hierarchical Distance Query.** For point 0, it falls out of the cut-off threshold, so its coarse distance is used directly as the world space distance. For point 1, it is within the range of local SDF values, so we blend the coarse world space distance and fine canonical distance to get the target SDF value.

incoming light directions. We also incorporate the Distance Field Soft Shadow (DFSS) algorithm [2, 6, 43] onto our drivable neural distance fields for soft-visibility computation. These properties are used to compute the radiance using the rendering equation [31]:

$$L_o = \int_\Omega L_s(\boldsymbol{\omega}_i) R_s(\boldsymbol{x}_s, \boldsymbol{\omega}_i, \boldsymbol{\omega}_o, \boldsymbol{n}_s) V_s(\boldsymbol{x}_s, \boldsymbol{\omega}_i)(\boldsymbol{n}_s \cdot \boldsymbol{\omega}_i) d\boldsymbol{\omega}_i, \tag{1}$$

where $L_o(\boldsymbol{x_s}, \boldsymbol{\omega}_o) \in \mathbb{R}^3$ is the outgoing radiance at the surface intersection point $\boldsymbol{x}_s$, $\boldsymbol{\omega}_o$ is the outgoing radiance direction, and $\boldsymbol{\omega}_i$ is the incoming radiance direction. In this paper, we use the Microfacet BRDF model in [59] which is defined in the canonical space of the animatable avatar, and an optimizable light probe image $L_s(\boldsymbol{\omega}_i) \in \mathbb{R}^{16\times32\times3}$. To make the optimization process more controllable, we separate the training process into two stages by postponing the training of the material and environment light probe after the geometry of the neural avatar has converged.

3.2. Hierarchical Distance Query

Given the world space query point $\boldsymbol{x}$, we approximate its world space distance $d^{world}(\boldsymbol{x})$ to the closest surface point on the neural avatar with the Hierarchical Distance Query algorithm $d^{world}(\boldsymbol{x}) \approx \tilde{d}^{world}(\boldsymbol{x}) = \text{HDQ}(\boldsymbol{x})$, which is later used for Sphere Tracing [25]. The query algorithm consists of four stages: (a) coarse distance query, (b) inverse warping, (c) fine distance query, and (d) smooth distance blending.

Coarse distance query. We first perform a geodesically-aware signed K Nearest Neighbor (GS-KNN) algorithm [48] on the posed vertices $\boldsymbol{v} \in \mathcal{V}$ of the driven parametric human model (SMPL-H [47]). GS-KNN produces the indices of the K closest points to $\boldsymbol{x}$ in $\mathcal{V}$, and its corresponding world-space closest vertices, distances, normals and blend weights. We set $K = 10$ through all experiments. The unsigned distance $\mathcal{D}$ is augmented with the sign of the dot product between $\boldsymbol{x} - \boldsymbol{v}$ and $\boldsymbol{n}$ to produce a coarse SDF. We additionally discard the k-th neighbor $\boldsymbol{v}_k$ if its canonical correspondence (T-Pose of SMPL-H) is too far from the canonical correspondence of the nearest neighbor. This strategy effectively exclude distant points from being assigned to the calculation of coarse world space SDF. The coarse level world space SDF is defined as $d^{world}_{coarse} = \frac{\sum_{k=0}^{K} d_k}{K}$.

Inverse warping. We follow the previous literature[38, 44] and use the linear blend skinning algorithm [35] to perform the inverse warping. The details can be found in the supplementary.

Fine distance query. Given the warped query point $\boldsymbol{x}'$, the pose-dependent displacement field $F_{\Delta\boldsymbol{x}}$ adds small perturbation to produce the final canonical space query point $\boldsymbol{x}''$. We implement $F_{\Delta\boldsymbol{x}}$ as an MLP with the human pose at the fth frame Θ_f and $\boldsymbol{x}'$ as input. The displaced canonical point $\boldsymbol{x}''$ fed into the canonical distance model S is defined as

$$\boldsymbol{x}'' = \boldsymbol{x}' + F_{\Delta\boldsymbol{x}}(\Theta_f, \boldsymbol{x}'). \tag{2}$$

Then, the fine canonical distance value can be obtained by querying the network $d^{can}_{fine} = S(\boldsymbol{x}'')$.

Smooth distance blending. Since SDF values of points close to the surface are hardly affected by LBS (Fig. 3 of the main paper and Fig. S3 of the supplementary), we propose to blend the fine canonical space distance value d^{can}_{fine} and the coarse world space distance d^{world}_{coarse} using a smooth function to produce the final approximated world space distance value $\tilde{d}^{world}$

$$\tilde{d}^{world} = \begin{cases} d^{world}_{coarse} & \text{, if } d^{world}_{coarse} > \tilde{T}_d \\ d^{can}_{fine}(1 - \frac{d^{can}_{fine}}{\tilde{T}_d}) + d^{world}_{coarse}\frac{d^{can}_{fine}}{\tilde{T}_d} & \text{, otherwise} \end{cases} \tag{3}$$

where $\tilde{T}_d$ is the distance threshold for cutting off coarse and fine distances, which is empirically set to 0.1. Note that we only perform the evaluation of S on points that satisfy the cutoff criteria $d^{world}_{coarse} \leq \tilde{T}_d$ for efficiency.

3.3. Geometry

Our physically based renderer requires the pixel-surface intersection $\boldsymbol{x}_s \in \mathbb{R}^3$, surface normal $\boldsymbol{n}_s \in \mathbb{R}^3$, and light visibility $V(\boldsymbol{x}_s, \boldsymbol{\omega}_i) \in \mathbb{R}$ as input. Using the Hierarchical Distance Query, these values can be easily obtained from the world space SDF of the neural avatar under arbitrary human poses.

Surface intersection. Given a camera ray and the neural avatar's SDF, we compute the location $\boldsymbol{x}_s$ at which the ray $\boldsymbol{r}(t) = \boldsymbol{o} + t\boldsymbol{d}$ from the camera origin $\boldsymbol{o}$ along the ray

direction $\boldsymbol{d}$ intersects the surface of the posed neural avatar. Specifically, we perform N_{st} Sphere Tracing iterations with the world space distance $\tilde{d}^{world} = \text{HDQ}(\boldsymbol{x})$ using Hierarchical Distance Query until the ray converges to the surface intersection point $\boldsymbol{x}_s$. The detailed algorithm is listed in the supplementary. N_{st} is set to 16 across all experiments.

Surface normal. The analytic normal direction $\boldsymbol{n}$ of any 3D points could be computed as the gradient of the neural SDF using $\nabla\tilde{d}^{world}(\boldsymbol{x})$. Although the hierarchical distance is differentiable, computing gradient through the whole query process is not efficient. Instead, we notice that surface intersections should satisfy the cutoff criteria of smooth distance blending in Section 3.2, that is

$$\tilde{d}^{world}(\boldsymbol{x}_s) = d^{can}_{fine}(\boldsymbol{x}_s), d^{world}_{coarse}(\boldsymbol{x}_s) \leq \tilde{T}_d. \tag{4}$$

Thus, the world space normal can be computed using $\nabla S(\boldsymbol{x}^{can}_s)$ and transformed from canonical to world space using the forward warping process. More details can be found in the supplementary.

Light visibility. Light visibility $V(\boldsymbol{x}, \boldsymbol{\omega}_i)$ from any 3D point $\boldsymbol{x}$ along any light direction $\boldsymbol{\omega}_i$ can be computed as whether the light path $\boldsymbol{x} + t\boldsymbol{\omega}_i$ is occluded by the geometry of the posed neural avatar, which is later integrated in the rendering equation [31] around the hemisphere. Since we use a discrete light probe $L_s(\boldsymbol{\omega}_i) \in \mathbb{R}^{16\times32\times3}$, the visibility term for every light direction needs to be integrated on the area of the pixel of $L_s(\boldsymbol{\omega}_i)$, which is time-consuming. Thanks to the global meaning of distance field, this occlusion and integration process can be approximated using Distance Field Soft Shadow (DFSS) [2, 6, 43], producing soft visibility with a single light sample. Specifically, we compute the visibility as the soft penumbra coefficient $p_s(\boldsymbol{x}_s, \boldsymbol{\omega}_i)$:

$$p_s(\boldsymbol{x}_s, \boldsymbol{\omega}_i) = \tag{5}$$
$$\min\left(\frac{\tilde{d}^{world}(\boldsymbol{x}_s + t_0\boldsymbol{\omega}_i)}{2t_0\sqrt{\frac{a}{\pi}}}, ..., \frac{\tilde{d}^{world}(\boldsymbol{x}_s + t_{N^{vis}_{st}}\boldsymbol{\omega}_i)}{2t_{N^{vis}_{st}}\sqrt{\frac{a}{\pi}}}\right),$$

for each surface point $\boldsymbol{x}_s$ along one of the 512 light directions $\boldsymbol{\omega}_i$ defined by $L_s(\boldsymbol{\omega}_i)$ during the N^{vis}_{st} sphere tracing steps, which is set to 4 for all experiments. The number of sphere tracing required for shadows are smaller as validated in Tab. 3. The ratio between the two tangent values $\frac{\tilde{d}^{world}}{t}$ and $2\sqrt{\frac{a}{\pi}}$ serves as an approximation of the ratio of light being occluded by the geometry from $\boldsymbol{x}_s$ along $\boldsymbol{\omega}_i$. Thanks to the smooth blending of d^{world}_{coarse} and d^{can}_{fine} in Sec. 3.2, our soft visibility scheme produces realistic and smooth soft shadow even when distance values from the parametric human model [47] and the canonical neural SDF are not perfectly aligned. A detailed listing of this algorithm is provided in the supplementary.

3.4. Reflectance

We adopt the Microfacet BRDF model in [59] for our material representation, which is composed of a diffuse albedo $\alpha \in \mathbb{R}^3$ term and a specular roughness $\gamma \in \mathbb{R}$ term. We use a fixed Fresnel term of 0.04. Similar to [16, 70, 71], we paramterize the albedo and roughness map with two MLPs $\alpha = A(\boldsymbol{x}'')$ and $\gamma = \Gamma(\boldsymbol{x}'')$, which is defined in the same canonical frame as $S(\boldsymbol{x}'')$ and $F_\Delta(\boldsymbol{x}')$ in Sec. 3.2. The BRDF model is denoted $R_s(\boldsymbol{x}_s, \boldsymbol{\omega}_i, \boldsymbol{\omega}_o, \boldsymbol{n}_s)$ where $\boldsymbol{\omega}_i$ is the incoming radiance direction, $\boldsymbol{\omega}_o$ is the outgoing radiance direction and $\boldsymbol{n}_s$ is the surface normal.

Given world space query point $\boldsymbol{x}_s$ and its corresponding canonical space point $\boldsymbol{x}''$, we obtain the albedo α and roughness γ by querying their canonical neural fields A and Γ, which can then be converted to BRDF values as defined in [59]. Our physically-based renderer also takes a light probe $L_s(\boldsymbol{\omega}_i) \in \mathbb{R}^{16\times32\times3}$ as input, which is represented by an optimizable neural texture during training and replaced with the designated environment map during relighting [16, 18, 70].

3.5. Training

We use 512 discrete incoming light directions defined by the light probe $L_s(\boldsymbol{\omega}_i) \in \mathbb{R}^{16\times32\times3}$ to approximate the Rendering Equation [31] as

$$L_o = \tag{6}$$
$$\sum_{\boldsymbol{\omega}_i} L_s(\boldsymbol{\omega}_i) R_s(\boldsymbol{x}_s, \boldsymbol{\omega}_i, \boldsymbol{\omega}_o, \boldsymbol{n}_s) V_s(\boldsymbol{x}_s, \boldsymbol{\omega}_i)(\boldsymbol{n}_s \cdot \boldsymbol{\omega}_i)\Delta\boldsymbol{\omega}_i,$$

where $\Delta\boldsymbol{\omega}_i$ is the solid angle of the incoming light $\boldsymbol{\omega}_i$ sampled from the light probe $L_s(\boldsymbol{\omega}_i)$ and $L_o(\boldsymbol{x_s}, \boldsymbol{\omega}_o) \in \mathbb{R}^3$ is the outgoing radiance at the surface intersection $\boldsymbol{x}_s$.

We optimize our relightable and animatable neural human avatar by rendering the image with given camera poses and comparing the pixel values L_o against the ground truth ones L_{gt}. The main loss function is defined as $\mathcal{L}_{data} = \sum_{\boldsymbol{r}\in\mathcal{R}} \|L_o(\boldsymbol{r}) - L_{gt}(\boldsymbol{r})\|_2$, where $\boldsymbol{r} = \boldsymbol{o} + t\boldsymbol{d} \in \mathcal{R}$ denotes all camera rays in the forward rendering process. Due to the ill-posed nature of the problem, we adopt a two-stage training strategy and add additional regularizations one the geometry (eikonal loss $\mathcal{L}_{eik}$) and material (sparsity loss $\mathcal{L}_{ent}$ and smoothness loss $\mathcal{L}_a, \mathcal{L}_r$). We elaborate on the details of each loss term and the training strategy in the supplementary. The training takes 20 hours on an Nvidia RTX 3090. Rendering a 512×512 image takes 5s.

4. Experiments

In this section, we conduct qualitative and quantitative experiments to investigate the performance of our relightable neural avatars. All hyperparameters are fixed through out the experiments unless otherwise specified. In Sec. 4.1, we

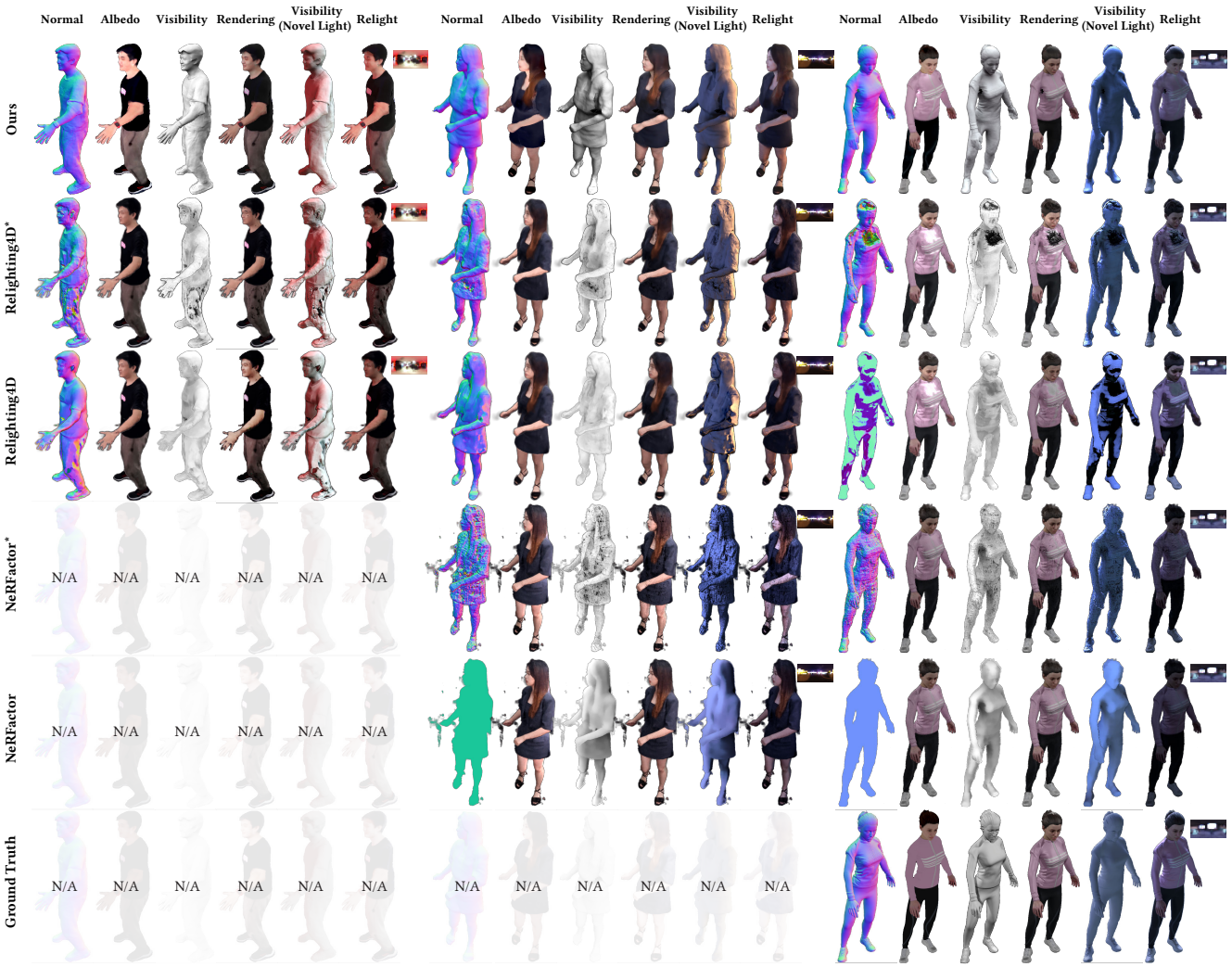

Figure 4. **Qualitative comparison of our method and baselines.** The first six columns display the results of synthesizing a character in a novel pose from the *MobileStage* dataset. The middle six columns depict a character in a training pose from the *MobileStage* dataset. For the last six columns, we show results from *SyntheticHuman++*, for which we have ground truth as reference. Note that NeRFactor is only trained on 1 frame. Relighting4D* and NeRFactor* denote directly computing normal and visibility using their density MLPs.

briefly introduce the datasets used for evaluation. Then we make quantitative and qualitative comparisons with three baseline methods in Sec. 4.2. Finally, we conduct ablation studies to investigate the effectiveness of our Hierarchical Distance Query and the soft visibility scheme in Sec. 4.3.

4.1. Datasets

We collect two datasets *MobileStage* and *SyntheticHuman++* for evaluation. *MobileStage* is a real-world multi-view (36 views) dataset created with synchronized mobile phone cameras on 4 real-world humans. We uniformly select 12 views for training. *SyntheticHuman++* contains 4 sequences (20 views) of dynamic 3D human models with ground truth shape and relighting information. We uniformly select 10 views for training for the sparse-view setting and we use the fourth view for the monocular setting. Please refer to the supplementary for more details.

4.2. Baseline Comparisons

Baselines. To the best of our knowledge, there are very few prior works that study the exact same setting as ours, i.e. training with unknown illumination and sparse-view (or monocular) videos while rendering with novel illumination and novel human poses. We take NeRFactor [70] and Relighting4D [16] as baselines and make comparisons with them on both real and synthetic datasets. Since NeRFactor is designed to handle static objects, we only train and evaluate it on the multi-view images of the first frame of each video. We

Table 1. **Quantitative comparison.** We compare our method with baselines on the *SyntheticHuman++* dataset. Following [46], the SSIM and LPIPS [68] metrics are computed in the bounding box of the human region, while the degree of normal and PSNR metrics are computed within the foreground mask. Due to the inherent scale ambiguity in the inverse rendering task, we align the rendered images and albedo with ground truth ones following [67] before computing metrics. Since NeRFactor [70] is designed to fit static objects, we train and evaluate it only on the first frame of each sequence. "$*$" denotes variants without the normal and visibility MLPs.

		Normal	**Diffuse Albedo**			**Relighting**			**Visibility**		
		Degree ↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
Sparse-View	Ours	**12.44**	**29.01**	**0.933**	**0.119**	**22.69**	**0.861**	**0.206**	**20.20**	**0.848**	**0.155**
	Relighting4D*	29.38	24.70	0.885	0.183	22.13	0.835	0.237	15.22	0.763	0.252
	Relighting4D	93.83	24.71	0.885	0.183	20.87	0.774	0.276	5.366	0.514	0.375
	NeRFactor* (1 frame)	34.29	22.23	0.817	0.226	21.04	0.758	0.313	11.37	0.581	0.387
	NeRFactor (1 frame)	51.92	22.23	0.817	0.226	20.70	0.757	0.299	10.56	0.597	0.361
	AniSDF	14.72	22.13	0.862	0.202	17.55	0.799	0.262	-	-	-
Monocular	Ours	**18.71**	23.42	**0.873**	**0.176**	**22.45**	**0.831**	**0.224**	**17.95**	**0.761**	**0.212**
	Relighting4D*	26.17	**25.37**	0.864	0.210	21.81	0.802	0.254	17.10	0.709	0.286
	Relighting4D	81.74	25.36	0.864	0.210	21.85	0.806	0.268	16.18	0.726	0.302
	AniSDF	20.36	21.51	0.812	0.255	18.29	0.745	0.297	-	-	-

observe that their normal and visibility MLPs often fail under complex human motions, thus we additionally compare with a Relighting4D* and NeRFactor* variant where we use the normal and visibility computed from the density MLP instead of the normal and visibility MLPs. To illustrate the effectiveness of our proposed components, we additionally compare with a non-relightable baseline [45].

Metrics. For quantitative analysis, we compare the normal (in degrees), albedo, light visibility and relighting results on 6 different light probes obtained from Polyhaven [1] using the PSNR, SSIM and LPIPS [68] metrics. Following [67], we align the diffuse albedo and rendered images with ground truth ones before computing metrics to mitigate the inherent scale ambiguity in the inverse rendering problem. Note that we evaluate PSNR using the same protocal as [44], only computing metrics on the human region. When computing metrics on the full image, our method achieves a PSNR of 28-30 dB. We do not compare the roughness term since Blender uses a different Principled BRDF model from [59]. Environment map of *SyntheticHuman* [45] is not available since they used programmatically defined light sources. We compare the uniform shading results to evaluate the visibility quality, where the BRDFs of the reconstructed avatars are set to 0.8 across all radiance directions (denoted "Visibility") when rendering. Since *SyntheticHuman* [45] does not provide ground truth models for novel poses, we only perform quantitative comparisons on training poses in Tab. 1, while qualitative analysis of animating the avatars can be found in Fig. 4 and the supplementary video.

Results. As shown in Fig. 4, our approach can successfully decompose the material and dynamic geometry of the neural avatar, generating a relightable neural avatar from only sparse-view (or monocular) video inputs. In comparison, NeRFactor[70] trained on 1 video frame overfits the training image with sparse-view input. Relighting4D [16] passes structured latent codes [46] to NeRFactor's MLPs, enabling it to relight a dynamic video of human performance. However, its quality decreases greatly when synthesizing novel poses. This is mainly because the visibility and normal MLP used in [16] is not generalizable to novel human motions. For the Relighting4D* variant, the density backbone still fails to generalize to novel poses [38, 45]. AniSDF [45] bakes illuminations effects like self-occlusions onto the rendering network, thus the reconstructed neural avatar looks unrealistic under novel illuminations. Qualitative results on monocular inputs can be found in the supplementary. Note thatRelighting4D and NeRFactor take 3s to render a 512×512 image, their "*" variants take 50s and our method takes 5s.

4.3. Ablation Studies

In this part, we ablate the effectiveness of our Hierarchical Distance Query and soft visibility scheme with the *jody* model of *SyntheticHuman++* under the sparse-view setting. More ablation studies can be found in the supplementary.

Effectiveness of Hierarchical Distance Query. In Fig. 5, We compare the results of performing sphere tracing on the canonical space distance ("*w/o* d^{world}_{coarse}"), coarse GS-KNN distance ("*w/o* d^{can}_{fine}") and our proposed hierarchically queried distance ("Ours"). As shown in the figure, the canonical space distance is incorrect when the query point is far from the actual surface of the human geometry, resulting in incorrect surface intersection points after the termination of the sphere tracing algorithm. Additionally, computing light visibility on this incorrect distance field would lead to false black regions since distances far from surface points are not reported correctly. Performing surface intersection

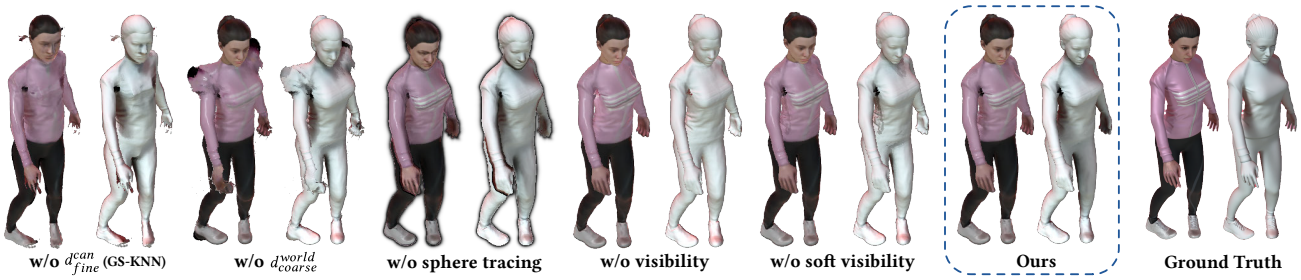

Figure 5. **Effectiveness of Hierarchical Distance Query**. Performing sphere tracing using only the canonical distance d^{can}_{fine} or coarse world distance d^{world}_{coarse} results in incorrect surface intersection and soft visibility, while tracing with our proposed Hierarchical Distance Query produces correct results. Using hard shadow ("*w/o* soft visibility") or no shadow ("*w/o* visibility") leads degraded perceptual quality.

and visibility computation on the coarse distance results in distorted rendering results. The "*w/o* sphere tracing" variant uses volume rendering of 128 samples per ray for surface intersection and light visibility computation, leading to an excessive rendering time of 60s per image for a resolution of 512×512, while our HDQ algorithm is able to obtain 10x speed-up at 5s per image with superior rendering quality.

Table 2. **Ablation study on Hierarchical Distance Query and soft visibility scheme.** The "*w/o* sphere tracing" variant uses naive volume rendering to compute pixel-surface intersection and visibility. More detailed description can be found in Sec. 4.3.

	Relighting			Visibility		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
Ours	**21.57**	**0.853**	**0.168**	20.53	0.869	**0.142**
w/o d^{can}_{fine} (GS-KNN)	19.56	0.784	0.245	14.63	0.758	0.233
w/o d^{world}_{coarse}	20.69	0.792	0.236	18.75	0.767	0.250
w/o sphere tracing	21.36	0.753	0.196	20.00	0.760	0.173
w/o visibility	21.00	0.844	0.175	20.88	0.869	0.143
w/o soft visibility	21.19	0.848	0.173	**21.27**	**0.873**	0.145

Effectiveness of the soft visibility scheme. We demonstrate the effectiveness of our soft visibility scheme by comparing it with two other variants where (a) hard visibility is used ("*w/o* soft visibility") and (b) no light visibility term is used ("*w/o* visibility"). The quantitative comparison of all three variants can be found in Tab. 2. Note that the visual quality of hard cast shadows in the "*w/o* soft visibility" is worse than ours, as indicated by the LPIPS metric and shown in Fig. 5.

Sensitivity analysis on hyper-parameters. We provide a runtime and sensitivity analysis regarding the cut-off value $\tilde{T}_d$ in Tab. 3. The cut-off for surface intersection is denoted $\tilde{T}_d$ and the cut-off for DFSS is denoted $\tilde{T}^{vis}_d$. The frame time and the rendering quality are roughly linear to the cut-off value up to a certain point, after which only diminishing returns can be observed by increasing the cut-off. However, a too-small value may result in incorrect surface intersection and visibility estimation, leading to degraded quality. Thus we choose the minimum cut-off value without a visible quality degradation ($\tilde{T}_d = 0.1, \tilde{T}^{vis}_d = 0.025$) as the default one. Setting the cut-off value to zero is effectively the same as the "*w/o* d^{can}_{fine} (GS-KNN)" variant in Fig. 5 , which greatly degrades the quality of relighting and rendering as shown in. Additional sensitivity analysis of hyperparameters can be found in the supplementary material.

Table 3. **Sensitivity study and runtime analysis on the cut-off value.** We choose the minimum cut-off value without a visible quality degradation ($\tilde{T}_d = 0.1, \tilde{T}^{vis}_d = 0.025$) as the default one.

		Relighting			Visibility		
	Frame Time↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
$\tilde{T}_d = 2.0, \tilde{T}^{vis}_d = 0.5$	7.659	**21.11**	**0.815**	0.200	18.10	0.795	0.193
$\tilde{T}_d = 1.0, \tilde{T}^{vis}_d = 0.25$	7.675	**21.11**	**0.815**	0.200	18.17	0.796	0.191
$\tilde{T}_d = 0.5, \tilde{T}^{vis}_d = 0.125$	7.320	21.10	**0.815**	0.199	18.17	0.797	0.189
$\tilde{T}_d = 0.1, \tilde{T}^{vis}_d = 0.025$	4.524	21.08	**0.815**	**0.197**	**18.19**	**0.798**	**0.187**
$\tilde{T}_d = 0.05, \tilde{T}^{vis}_d = 0.0125$	2.631	21.04	0.814	0.202	17.83	0.790	0.197
$\tilde{T}_d = 0.01, \tilde{T}^{vis}_d = 0.0025$	**0.976**	19.50	0.733	0.300	15.82	0.697	0.304

5. Conclusion

This paper presents a novel framework to reconstruct relightable and animatable neural avatars from only sparse-view (or monocular) video input. We generalize the canonical distance field to arbitrary human poses via a hierarchical distance query scheme, with which the photometric properties of the neural avatar can be easily retrieved for relighting. We demonstrate that together with other innovative components, our approach reconstructs high-quality animatable geometry and material, supporting realistic relighting.

Limitations. Although the proposed method produces high-quality relighting results from challenging sparse-view or monocular settings, it has the natural limitation of neural field methods in that it requires a long training time of 20 hours and could not render in real-time (5s per image). Future work could consider recent neural field acceleration methods to further increase the training and rendering speed. More discussions are presented in the supplementary.

Acknowledgement

The authors would like to acknowledge support from NSFC (No. 62172364) and Information Technology Center and State Key Lab of CAD&CG, Zhejiang University.

References

[1] Poly Haven, 2023. 7

[2] Sebastian Aaltonen. Gpu-based clay simulation and ray-tracing tech in claybook. *San Francisco, CA*, 2(5), 2018. 4, 5

[3] Kairat Aitpayev and Jaafar Gaber. Creation of 3d human avatar using kinect. *Asian Transactions on Fundamentals of Electronics, Communication & Multimedia*, 1(5):12–24, 2012. 2

[4] Thiemo Alldieck, Marcus Magnor, Weipeng Xu, Christian Theobalt, and Gerard Pons-Moll. Video based reconstruction of 3d people models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8387–8397, 2018. 2

[5] Thiemo Alldieck, Mihai Zanfir, and Cristian Sminchisescu. Photorealistic monocular 3d reconstruction of humans wearing clothing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1506–1515, 2022. 3

[6] Róbert Bán, Csaba Bálint, and Gábor Valasek. Area lights in signed distance function scenes. In *Eurographics (Short Papers)*, pages 85–88, 2019. 4, 5

[7] Shrisha Bharadwaj, Yufeng Zheng, Otmar Hilliges, Michael J. Black, and Victoria Fernandez Abrevaya. Flare: Fast learning of animatable and relightable mesh avatars. *ACM Transactions on Graphics*, 42:15, Dec. 2023. 3

[8] Sai Bi, Stephen Lombardi, Shunsuke Saito, Tomas Simon, Shih-En Wei, Kevyn Mcphail, Ravi Ramamoorthi, Yaser Sheikh, and Jason Saragih. Deep relightable appearance models for animatable faces. *ACM Transactions on Graphics (TOG)*, 40(4):1–15, 2021. 3

[9] Federica Bogo, Michael J Black, Matthew Loper, and Javier Romero. Detailed full-body reconstructions of moving people from monocular rgb-d sequences. In *Proceedings of the IEEE international conference on computer vision*, pages 2300–2308, 2015. 2

[10] Mark Boss, Raphael Braun, Varun Jampani, Jonathan T Barron, Ce Liu, and Hendrik Lensch. Nerd: Neural reflectance decomposition from image collections. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12684–12694, 2021. 2, 3

[11] Mark Boss, Andreas Engelhardt, Abhishek Kar, Yuanzhen Li, Deqing Sun, Jonathan T Barron, Hendrik Lensch, and Varun Jampani. Samurai: Shape and material from unconstrained real-world arbitrary image collections. *arXiv preprint arXiv:2205.15768*, 2022.

[12] Mark Boss, Varun Jampani, Raphael Braun, Ce Liu, Jonathan Barron, and Hendrik Lensch. Neural-pil: Neural pre-integrated lighting for reflectance decomposition. *Advances in Neural Information Processing Systems*, 34:10691–10704, 2021. 3

[13] Joel Carranza, Christian Theobalt, Marcus A Magnor, and Hans-Peter Seidel. Free-viewpoint video of human actors. *ACM transactions on graphics (TOG)*, 22(3):569–577, 2003. 1, 2

[14] Xu Chen, Yufeng Zheng, Michael J Black, Otmar Hilliges, and Andreas Geiger. Snarf: Differentiable forward skinning for animating non-rigid neural implicit shapes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11594–11604, 2021. 2

[15] Ziyu Chen, Chenjing Ding, Jianfei Guo, Dongliang Wang, Yikang Li, Xuan Xiao, Wei Wu, and Li Song. L-tracing: Fast light visibility estimation on neural surfaces by sphere tracing. In *European Conference on Computer Vision*, pages 217–233. Springer, 2022. 2, 3

[16] Zhaoxi Chen and Ziwei Liu. Relighting4d: Neural relightable human from videos. In *European Conference on Computer Vision*, pages 606–623. Springer, 2022. 3, 5, 6, 7

[17] Alvaro Collet, Ming Chuang, Pat Sweeney, Don Gillett, Dennis Evseev, David Calabrese, Hugues Hoppe, Adam Kirk, and Steve Sullivan. High-quality streamable free-viewpoint video. *ACM Transactions on Graphics (ToG)*, 34(4):1–13, 2015. 1, 2

[18] Paul Debevec. Rendering synthetic objects into real scenes: Bridging traditional and image-based graphics with global illumination and high dynamic range photography. In *Acm siggraph 2008 classes*, pages 1–10. 2008. 5

[19] Paul Debevec. The light stages and their applications to photoreal digital actors. Technical report, UNIVERSITY OF SOUTHERN CALIFORNIA LOS ANGELES, 2012. 1, 3

[20] Paul Debevec, Tim Hawkins, Chris Tchou, Haarm-Pieter Duiker, Westley Sarokin, and Mark Sagar. Acquiring the reflectance field of a human face. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 145–156, 2000. 1, 3

[21] Oliver Grau. Studio production system for dynamic 3d content. In *Visual Communications and Image Processing 2003*, volume 5150, pages 80–89. SPIE, 2003. 2

[22] Kaiwen Guo, Peter Lincoln, Philip Davidson, Jay Busch, Xueming Yu, Matt Whalen, Geoff Harvey, Sergio Orts-Escolano, Rohit Pandey, Jason Dourgarian, et al. The relightables: Volumetric performance capture of humans with realistic relighting. *ACM Transactions on Graphics (ToG)*, 38(6):1–19, 2019. 1, 3

[23] Marc Habermann, Weipeng Xu, Michael Zollhoefer, Gerard Pons-Moll, and Christian Theobalt. Livecap: Real-time human performance capture from monocular video. *ACM Transactions On Graphics (TOG)*, 38(2):1–17, 2019. 2

[24] Marc Habermann, Weipeng Xu, Michael Zollhofer, Gerard Pons-Moll, and Christian Theobalt. Deepcap: Monocular human performance capture using weak supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5052–5063, 2020. 2

[25] John C Hart et al. Sphere tracing: Simple robust antialiased rendering of distance-based implicit surfaces. In *Siggraph*, volume 93, pages 1–11, 1993. 2, 4

[26] Zeng Huang, Yuanlu Xu, Christoph Lassner, Hao Li, and Tony Tung. Arch: Animatable reconstruction of clothed humans. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3093–3102, 2020. 2

[27] Umar Iqbal, Akin Caliskan, Koki Nagano, Sameh Khamis, Pavlo Molchanov, and Jan Kautz. Rana: Relightable articulated neural avatars. *arXiv preprint arXiv:2212.03237*, 2022. 3

[28] Shun Iwase, Shunsuke Saito, Tomas Simon, Stephen Lombardi, Timur Bagautdinov, Rohan Joshi, Fabian Prada, Takaaki Shiratori, Yaser Sheikh, and Jason Saragih. Re-

lightablehands: Efficient neural relighting of articulated hand models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16663–16673, 2023. 3

[29] Chaonan Ji, Tao Yu, Kaiwen Guo, Jingxin Liu, and Yebin Liu. Geometry-aware single-image full-body human relighting. In *European Conference on Computer Vision*, pages 388–405. Springer, 2022. 3

[30] Wei Jiang, Kwang Moo Yi, Golnoosh Samei, Oncel Tuzel, and Anurag Ranjan. Neuman: Neural human radiance field from a single video. In *European Conference on Computer Vision*, pages 402–418. Springer, 2022. 2

[31] James T Kajiya. The rendering equation. In *Proceedings of the 13th annual conference on Computer graphics and interactive techniques*, pages 143–150, 1986. 4, 5

[32] Takeo Kanade, Peter Rander, and PJ Narayanan. Virtualized reality: Constructing virtual worlds from real scenes. *IEEE multimedia*, 4(1):34–47, 1997. 1, 2

[33] Yoshihiro Kanamori and Yuki Endo. Relighting humans: occlusion-aware inverse rendering for full-body human images. *arXiv preprint arXiv:1908.02714*, 2019. 3

[34] Zhengfei Kuang, Kyle Olszewski, Menglei Chai, Zeng Huang, Panos Achlioptas, and Sergey Tulyakov. Neroic: Neural rendering of objects from online image collections. *arXiv preprint arXiv:2201.02533*, 2022. 3

[35] John P Lewis, Matt Cordner, and Nickson Fong. Pose space deformation: a unified approach to shape interpolation and skeleton-driven deformation. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 165–172, 2000. 1, 2, 3, 4

[36] Junxuan Li, Shunsuke Saito, Tomas Simon, Stephen Lombardi, Hongdong Li, and Jason Saragih. Megane: Morphable eyeglass and avatar network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12769–12779, 2023. 3

[37] Zhe Li, Zerong Zheng, Yuxiao Liu, Boyao Zhou, and Yebin Liu. Posevocab: Learning joint-structured pose embeddings for human avatar modeling. In *ACM SIGGRAPH 2023 Conference Proceedings*, pages 1–11, 2023. 2

[38] Lingjie Liu, Marc Habermann, Viktor Rudnev, Kripasindhu Sarkar, Jiatao Gu, and Christian Theobalt. Neural actor: Neural free-view synthesis of human actors with pose control. *ACM Transactions on Graphics (TOG)*, 40(6):1–16, 2021. 1, 2, 3, 4, 7

[39] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. Smpl: A skinned multi-person linear model. *ACM transactions on graphics (TOG)*, 34(6):1–16, 2015. 2

[40] Abhimitra Meka, Rohit Pandey, Christian Haene, Sergio Orts-Escolano, Peter Barnum, Philip David-Son, Daniel Erickson, Yinda Zhang, Jonathan Taylor, Sofien Bouaziz, et al. Deep relightable textures: volumetric performance capture with neural rendering. *ACM Transactions on Graphics (TOG)*, 39(6):1–21, 2020. 3

[41] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021. 2

[42] Rohit Pandey, Sergio Orts Escolano, Chloe Legendre, Christian Haene, Sofien Bouaziz, Christoph Rhemann, Paul Debevec, and Sean Fanello. Total relighting: learning to relight portraits for background replacement. *ACM Transactions on Graphics (TOG)*, 40(4):1–21, 2021. 3

[43] Steven Parker, Peter Shirley, and Brian Smits. Single sample soft shadows. Technical report, Technical Report UUCS-98-019, Computer Science Department, University of Utah, 1998. 2, 4, 5

[44] Sida Peng, Junting Dong, Qianqian Wang, Shangzhan Zhang, Qing Shuai, Xiaowei Zhou, and Hujun Bao. Animatable neural radiance fields for modeling dynamic human bodies. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14314–14323, 2021. 1, 2, 3, 4, 7

[45] Sida Peng, Zhen Xu, Junting Dong, Qianqian Wang, Shangzhan Zhang, Qing Shuai, Hujun Bao, and Xiaowei Zhou. Animatable implicit neural representations for creating realistic avatars from videos. *arXiv preprint arXiv:2203.08133*, 2022. 1, 2, 3, 7

[46] Sida Peng, Yuanqing Zhang, Yinghao Xu, Qianqian Wang, Qing Shuai, Hujun Bao, and Xiaowei Zhou. Neural body: Implicit neural representations with structured latent codes for novel view synthesis of dynamic humans. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9054–9063, 2021. 1, 2, 3, 7

[47] Javier Romero, Dimitrios Tzionas, and Michael J Black. Embodied hands: Modeling and capturing hands and bodies together. *ACM ToG*, 2017. 4, 5

[48] Nick Roussopoulos, Stephen Kelley, and Frederic Vincent. Nearest neighbor queries. In *Proceedings of the 1995 ACM SIGMOD international conference on Management of data*, pages 71–79, 1995. 4

[49] Shunsuke Saito, Jinlong Yang, Qianli Ma, and Michael J Black. Scanimate: Weakly supervised learning of skinned clothed avatar networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2886–2897, 2021. 2

[50] Carolin Schmitt, Simon Donne, Gernot Riegler, Vladlen Koltun, and Andreas Geiger. On joint estimation of pose, geometry and svbrdf from a handheld scanner. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3493–3503, 2020. 3

[51] Dario Seyb, Alec Jacobson, Derek Nowrouzezahrai, and Wojciech Jarosz. Non-linear sphere tracing for rendering deformed signed distance fields. *ACM Transactions on Graphics*, 38(6), 2019. 2

[52] Ari Shapiro, Andrew Feng, Ruizhe Wang, Hao Li, Mark Bolas, Gerard Medioni, and Evan Suma. Rapid avatar capture and simulation using commodity depth sensors. *Computer Animation and Virtual Worlds*, 25(3-4):201–211, 2014. 2

[53] Pratul P Srinivasan, Boyang Deng, Xiuming Zhang, Matthew Tancik, Ben Mildenhall, and Jonathan T Barron. Nerv: Neural reflectance and visibility fields for relighting and view synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7495–7504, 2021. 2, 3

[54] Jonathan Starck and Adrian Hilton. Virtual view synthesis of people from multiple view video sequences. *Graphical Models*, 67(6):600–620, 2005. 2

[55] Jonathan Starck and Adrian Hilton. Surface capture for performance-based animation. *IEEE computer graphics and applications*, 27(3):21–31, 2007. 1, 2

[56] Wenzhang Sun, Yunlong Che, Han Huang, and Yandong Guo. Neural reconstruction of relightable human model from monocular video. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 397–407, 2023. 3

[57] Jing Tong, Jin Zhou, Ligang Liu, Zhigeng Pan, and Hao Yan. Scanning 3d full human bodies using kinects. *IEEE transactions on visualization and computer graphics*, 18(4):643–650, 2012. 2

[58] Daniel Vlasic, Ilya Baran, Wojciech Matusik, and Jovan Popović. Articulated mesh animation from multi-view silhouettes. In *Acm Siggraph 2008 papers*, pages 1–9. 2008. 2

[59] Bruce Walter, Stephen R Marschner, Hongsong Li, and Kenneth E Torrance. Microfacet models for refraction through rough surfaces. In *Proceedings of the 18th Eurographics conference on Rendering Techniques*, pages 195–206, 2007. 4, 5, 7

[60] Shaofei Wang, Katja Schwarz, Andreas Geiger, and Siyu Tang. Arah: Animatable volume rendering of articulated human sdfs. In *European conference on computer vision*, pages 1–19. Springer, 2022. 1, 2

[61] Chung-Yi Weng, Brian Curless, Pratul P Srinivasan, Jonathan T Barron, and Ira Kemelmacher-Shlizerman. Humannerf: Free-viewpoint rendering of moving people from monocular video. *arXiv preprint arXiv:2201.04127*, 2022. 2, 3

[62] Andreas Wenger, Andrew Gardner, Chris Tchou, Jonas Unger, Tim Hawkins, and Paul Debevec. Performance relighting and reflectance transformation with time-multiplexed illumination. *ACM Transactions on Graphics (TOG)*, 24(3):756–764, 2005. 1, 3

[63] Hongyi Xu, Thiemo Alldieck, and Cristian Sminchisescu. H-nerf: Neural radiance fields for rendering and temporal reconstruction of humans in motion. *Advances in Neural Information Processing Systems*, 34:14955–14966, 2021. 2

[64] Weipeng Xu, Avishek Chatterjee, Michael Zollhöfer, Helge Rhodin, Dushyant Mehta, Hans-Peter Seidel, and Christian Theobalt. Monoperfcap: Human performance capture from monocular video. *ACM Transactions on Graphics (ToG)*, 37(2):1–15, 2018. 2

[65] Haotian Yang, Mingwu Zheng, Wanquan Feng, Haibin Huang, Yu-Kun Lai, Pengfei Wan, Zhongyuan Wang, and Chongyang Ma. Towards practical capture of high-fidelity relightable avatars. *arXiv preprint arXiv:2309.04247*, 2023. 3

[66] Yu-Ying Yeh, Koki Nagano, Sameh Khamis, Jan Kautz, Ming-Yu Liu, and Ting-Chun Wang. Learning to relight portrait images via a virtual light stage and synthetic-to-real adaptation. *ACM Transactions on Graphics (TOG)*, 41(6):1–21, 2022. 3

[67] Kai Zhang, Fujun Luan, Qianqian Wang, Kavita Bala, and Noah Snavely. Physg: Inverse rendering with spherical gaussians for physics-based material editing and relighting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5453–5462, 2021. 2, 3, 7

[68] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018. 7

[69] Xiuming Zhang, Sean Fanello, Yun-Ta Tsai, Tiancheng Sun, Tianfan Xue, Rohit Pandey, Sergio Orts-Escolano, Philip Davidson, Christoph Rhemann, Paul Debevec, et al. Neural light transport for relighting and view synthesis. *ACM Transactions on Graphics (TOG)*, 40(1):1–17, 2021. 3

[70] Xiuming Zhang, Pratul P Srinivasan, Boyang Deng, Paul Debevec, William T Freeman, and Jonathan T Barron. Nerfactor: Neural factorization of shape and reflectance under an unknown illumination. *ACM Transactions on Graphics (TOG)*, 40(6):1–18, 2021. 2, 3, 5, 6, 7

[71] Yuanqing Zhang, Jiaming Sun, Xingyi He, Huan Fu, Rongfei Jia, and Xiaowei Zhou. Modeling indirect illumination for inverse rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18643–18652, 2022. 2, 3, 5