RGAvatar: Relightable 4D Gaussian Avatar from Monocular Videos

Zhe Fan, Shi-Sheng Huang, Yichi Zhang, Dachao Shang, Juyong Zhang, Yudong Guo, Hua Huang, *Senior Member, IEEE*

Abstract—Relightable 4D avatar reconstruction which enables high fidelity and real-time rendering continues to be a crucial but challenging problem, especially from monocular videos. Previous NeRF-based 4D avatars enable photo-realistic relighting but are too slow for rendering, while point-based or mesh-based 4D avatars are efficient but have limited rendering quality. The recent success of 3D Gaussian Splatting, i.e., 3DGS, has inspired a series of impressive 4D Gaussian avatars, however, most of which only focus on faithful appearance reconstruction but are not relightable. To address such issues, this paper proposes a new **R**elightable 4D **G**aussian **A**vatar, i.e., RGAvatar, tailored for high fidelity relightable rendering from monocular videos. Our key idea is to introduce a new relightable 4D Gaussian representation, based on which we can directly perform high fidelity Physically Based Rendering, and an effective joint learning mechanism for *compact* 4D Gaussian reconstruction with SDF regulation and *accurate* materials and lighting decomposition. By comparing with previous state-of-the-art approaches, RGAvatar can significantly outperform previous approaches in relightable rendering quality and speed. To our best knowledge, RGAvatar contributes a new state-of-the-art 4D Gaussian avatar from monocular videos, which enables high fidelity relightable rendering in a quite efficient manner.

Index Terms—4D Gaussian Splatting, Relightable 4D Gaussian Avatar, Relightable rendering, Neural head avatars.

1 Introduction

AUTOMATIC reconstructing *animatable* head avatars [1], [2] is an active research topic in computer graphics and computer vision communities, which has a wide range of applications in VR/AR, video games, teleconferencing, movie production, etc. There has been remarkable progress on animatable avatar creation work using single commodity sensors as input [3], [4], [5], [6], especially inspired by the impressive success of NeRF [7], [8], [9] as neural implicit representation, which have shown impressive applications such as novel view synthesis [4], novel expression [10], [11], [12], [13] and novel pose control [14], [15], and have also significantly accelerated the training process within even minutes [16], [17]. However, most previous 4D avatars focus on accurate avatar reconstruction, but couldn't support any relighting applications.

To relight objects, a typical approach is to perform Physically Based Rendering under new illumination with known or estimated material properties. One mainstream method is to directly learn relightable appearance model from lightstage devices, which show promising relighting for static [19], [20], [21], [22] and dynamic scenes [23], [24], [25]. However, those approaches are not feasible for 4D avatars from monocular videos. Another kind of relighting approach is neural inverse rendering [26], [27], [28], [29], which explores more flexible representation for unknown materials and lighting estimation from multi-view images or spare-view/monocular videos. But their essential requirement of light visibility estimation or tracing is often too slow to perform even for static scenes, which would be more time-consuming for 4D avatar scenario. For example, the latest work [30] still costs up to 60s when performing relightable rendering each frame. On the other hand, some recent works adopts to use explicit representation with differential rendering, such as points (PointAvatar [31]) and mesh (FLARE [18]), for efficient relightable 4D avatar reconstruction. However, the rendering quality for these approaches are still limited mainly due to the limited ability for point or mesh based rasterization.

The recent success of 3D Gaussian Splatting [32], i.e., 3DGS has inspired many impressive 4D Gaussian avatars [33], [34], [35], [36], [37]. However, most of those 4D Gaussian avatars only focus on accurate appearance reconstruction, but couldn't support any relightable applications. Recently, RGCA [38] propose a relightable Gaussian codec avatar model which enables real-time relighting of human head with impressive rendering quality. But this work still need known illumination captured from lightstage device as input, which couldn't work for monocular videos.

In this paper, we propose a new **R**elightable 4D **G**aussian **A**vatar (RGAvatar) model from monocular videos, which is especially tailored to high fidelity relightable human head rendering in a quite efficient manner. The key component of RGAvatar is a new relightable 4D Gaussian representation. Instead of approximating the Gaussian's appearance with spherical harmonics (SH) diffuse color and spherical Gaussian (SG) specular color as like RGCA [38], we

- *Zhe Fan and Yichi Zhang are with the School of Computing Science & Technology, Beijing Institute of Technology, Beijing 100081, China. E-mail: {fanzhe, 3220221006}@bit.edu.cn.*
- *Shi-Sheng Huang, Dachao Shang and Hua Huang are with the School of Artificial Intelligence, Beijing Normal University, Beijing 100875, China. E-mail: shishenghuang.net@gmail.com, {202111081029, huahuang}@bnu.edu.cn.*
- *Juyong Zhang and Yudong Guo are with the School of Mathematical Sciences, University of Science and Technology of China, Anhui 230026, China. E-mail: {juyong, yudong}@ustc.edu.cn.*

Hua Huang is the corresponding author.

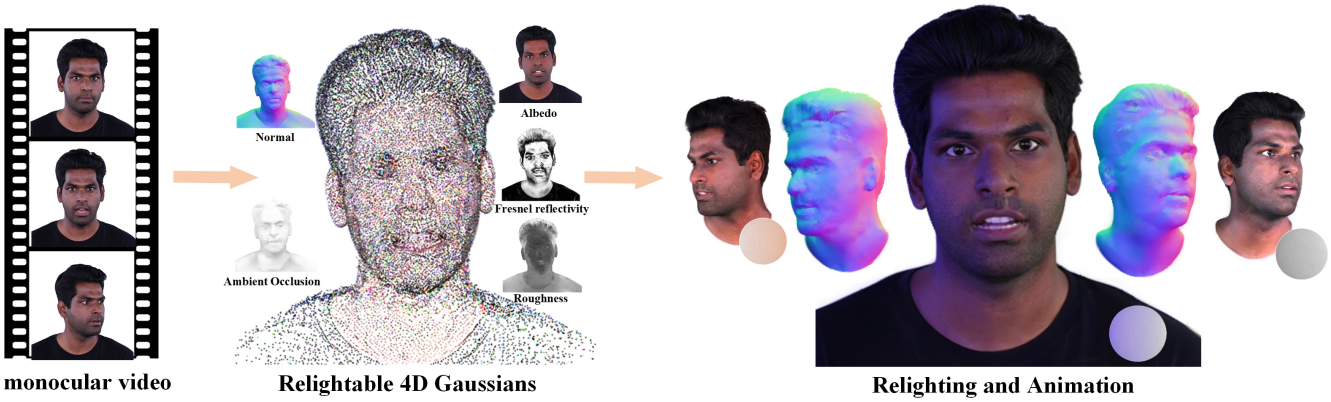

Fig. 1. This paper proposes a novel relightable 4D Gaussian avatar (RGAvatar) from monocular videos. Given RGB frames of a person's fixed viewport monocular video (left), we reconstruct the RGAvatar by accurately learning the relightable 4D Gaussian representation (R-4DGS) with *compact* 4D Gaussian reconstruction and accurate neural materials and incident light decomposition (middle). Our RGAvatar enables flexible novel view/pose/expression synthesis while performing relightable rendering using changing environmental light (right), which achieves much better rendering quality than previous approaches (FLARE [18]) and at very efficient rendering speed. RGAvatar contributes to be a new state-of-the-art relightable 4D avatar, in terms of both high fidelity relightable rendering quality and efficient rendering speed.

formulate a new relightable 4D Gaussians representation, which can perform Physically Based Rendering directly for more accurate and high fidelity appearance modeling. More importantly, we provide an effective joint learning of the relightable 4D Gaussians, which aims at a compact 4D Gaussian reconstruction and accurate BRDF materials and incident lighting decomposition simultaneously. Our key observation for the joint learning mechanism is to introduce effective regularization from an extra SDF field simultaneously learnt for the accurate learning of 4D Gaussian's geometry, materials and incident lighting factors from monocular videos. Once the relightable 4D Gaussians accurately learnt, we can efficiently perform relightable rendering for the avatar in high fidelity rendering quality.

To evaluate the effectiveness of RGAvatar, we have conducted extensive evaluation on public dataset by comparing with previous approaches. From the comparison, RGAvatar can achieve better 4D avatar reconstruction than previous 4D avatars such as INSTA [17], PointAvatar [31], FLARE [18] and recent 4D Gaussian avatars such as FlashAvatar [35] and SplattingAvatar [37]. For relighting application, RGAvatar can achieve significant better relighting quality than PointAvatar [31] and FLARE [18] while maintaining very efficient rendering speed. We summarize the main contribution of this paper in the following three folds:

1) We introduce a new relightable 4D Gaussian representation and a *compact* 4D Gaussian reconstruction, providing accurate geometry prior for materials and lighting decomposition.
2) We provide a neural material and lighting decomposition to learn the relightable 4D Gaussian representation, achieving high accurate geometry, material and incident lighting estimation.
3) We propose RGAvatar framework to efficiently reconstruct high fidelity relightable 4D Gaussian avatars from monocular videos, which significantly outperforms previous approaches as a new SOTA relightable 4D avatar method.

2 Related Works

Neural 4D Avatars. Following the pioneering work of 3DMM [39], plenty of works have made considerable efforts to *efficiently* generate realistic face models by introducing corrective blendshapes from subject shape [40], head pose and expression [41], emotion [42], and even biological factors such as bones and muscle [43]. These 3D morphable face models can be served as explicit representations for face priors, which benefit various tasks including face reconstruction [44], [45], [46], photo-realistic face synthesis [47], [48], and face reenactment [49], [50]. The recent work of NHA [4] proposed a full-head avatar model with view-dependent texture mapping. However, most of the current 3D morphable face models mainly focus on the face part with topology-fixed template mesh, thus limiting the rendering quality.

The impressive works such as NerFACE [3], HeadNeRF [5], IMAvatar [6], HAvatar [12] and LatentAvatar [16] proposed to utilize NeRF [7], [8], [9] for implicit head avatar generation. However, such dynamic NeRF-based deformation is over time-consuming, which would cost hours or even days for a single subject's head avatar training. NeRFBlendShape [10] reduces the training time to 20 minutes by introducing a hash table-based voxel field representation. AvatarMAV [11] decouples the motion and appearance with motion-aware neural voxels, which further reduces the head avatar training into 5 minutes.

On the other hand, DoubleField [51] provides a 3D avatar reconstruction by leveraging a neural surface field to guide the NeRF learning for high fidelity geometry reconstruction and rendering simultaneously, even under sparse-view scenarios. Some approaches adopt to use the 3D-Aware GAN to generate high fidelity controllable 3D avatars, such as Next3D [52] and AniPortraitGAN [53]. Very recently, the success of 3D Gaussian Splatting has inspired a serials of 4D Gaussian avatars [33], [34], [35], [36], [37], which achieves impressive head avatar creation results. However, most of previous neural 4D avatars focus on the avatar reconstruction but didn't support relightable

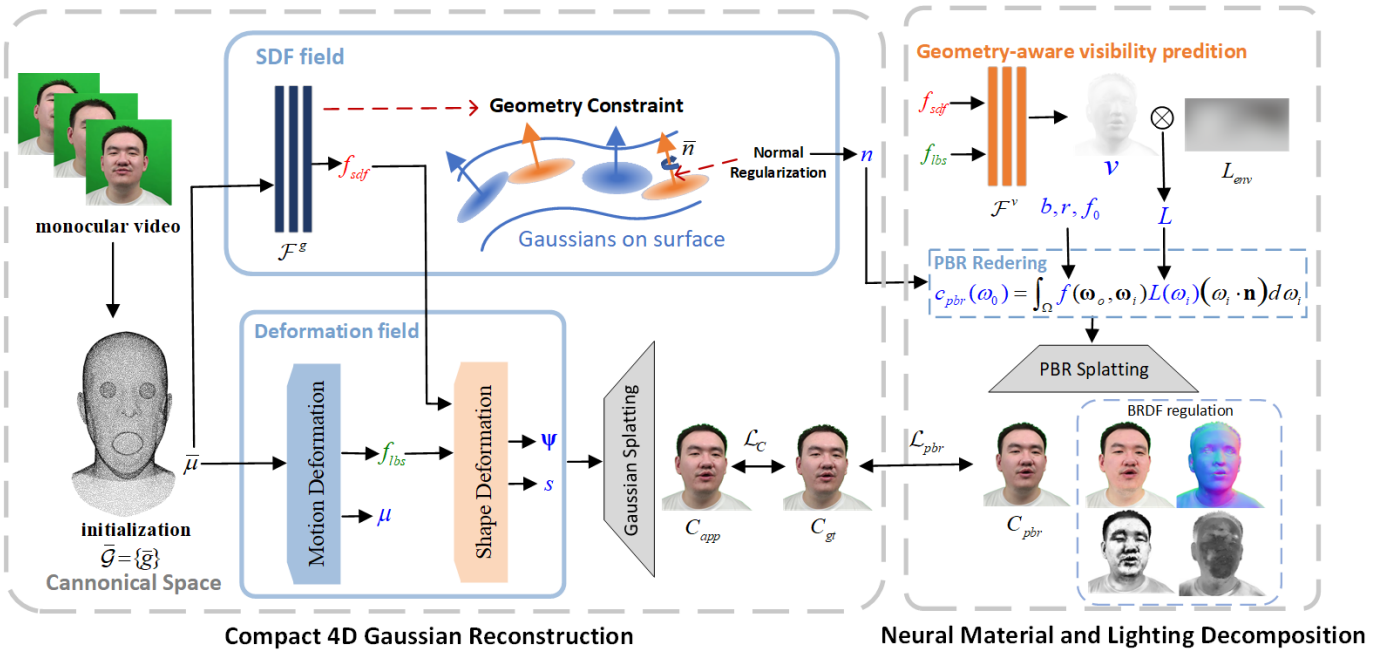

Fig. 2. The pipeline of RGAvatar. Given monocular video input, we propose to learn the R-4DGS with two learning stages. In the first stage, we aims at a compact 4D Gaussian reconstruction (left), which forces the R-4DGS compactly coherent to the underlying surface of an extra SDF field $\mathcal{F}^g$(left top). We deform the canonical R-4DGS $\bar{\mathcal{G}} = \{\bar{g}\}$ to g using a deformation field $\mathcal{D}$, in which a Motion Deformation Module is used to deform the centroid position $\bar{\mu}$ from canonical state to dynamic μ, and a Shape Deformation Module to deform the rotation and scale. In the next stage, we perform a neural material and lighting decomposition (right), by leveraging strong geometry prior from the compact 4D Gaussian reconstruction learning to guide the geometry-aware light visibility prediction using network $\mathcal{F}^v$, which takes the SDF encoding f_{sdf} and LBS deformation encodingf_{lbs} as input. This tailored visibility prediction module leading to more accurate materials (albedo b, roughness r and fresnel reflectivity f_0) and incident light (L_{env}) learning for the 4D Gaussians.

rendering. In contrast, this paper contributes a relightable 4D avatar, which enables high fidelity relightable rendering at very efficient speed.

Neural Relighting. There have been much progress made to learn relightable models using lightstage devices [19], [20], [21], [22], [23], [24], [25], and achieved impressive relighting results. To achieve more flexible relighting approaches which get rid of the usage of lightstage, neural inverse rendering [26], [27], [28], [29] is introduced to learn relightable 3D assets from multi-view images or spare-view/monocular videos. Munkberg et al [54] adopts neural SDFs to reconstruct high quality relightable 3D assets. Recently, SwitchLight [55] proposes a 2D relighting approach for impressive human portrait relighting results by combining a physics-guided architecture with a pre-training framework. Meanwhile, SwitchLight mainly focus on 2D image-based human portrait relighting, but is not feasible for controllable 3D head avatar relighting tasks.

For relightable 3D head avatars, some approaches [56], [57] leverage deformable priors from 3DMM [39] to learn neural materials and lighting from image or videos. Recently, Xu et al [30] propose to reconstruct relightable avatars by modeling appearance color with spherical harmonics (SH) diffuse color and spherical Gaussian (SG) specular color, which achieves impressive relightable rendering results. However, most of those previous approaches need very time-consuming light visibility tracing operations, which are limited for real-time relightable rendering applications. On the other hand, some recent works adopts to use explicit representation with differential rendering, such as points (PointAvatar [31]) and mesh (FLARE [18]), for efficient relightable 4D avatar reconstruction. However, their rendering quality are still limited mainly due to the limited ability for point or mesh based rasterization.

Very recently, RGCA [38] attempts to learn efficient relightable avatar based on 3D Gaussian splatting, but still requires known input lighting for accurate materials learning. Unlike previous approaches, this paper presents a new relightable 4D Gaussian representation, and accurately reconstruct a relightable 4D Gaussian avatar for high fidelity relightable rendering, and achieves fast rendering speed from monocular videos.

3 Method

Given a frame sequence $\mathcal{I} = \{I_1, ..., I_N\}$ of an individual person's portrait video, our goal is to reconstruct an animatable head avatar, called RGAvatar, represented by a set of relightable 4D Gaussians $\mathcal{G}(\phi, \theta, L_{env})$, where ϕ, θ represents the expression and pose parameters of FLAME model [41] respectively and L_{env} represents the environmental lighting. RGAvatar's key component is a joint learning mechanism to reconstruct the compact 4D Gaussians, and accurately decompose the materials and incident lighting simultaneously from frame sequence $\mathcal{I}$. Once accurately learnt, we can reanimate the head avatar using FLAME [41] parameters ϕ, θ for novel view/pose/expression synthesis, while performing relighting with changing environmental lighting L_{env} simultaneously. The system overview of RGAvatar is shown in Fig. 2.

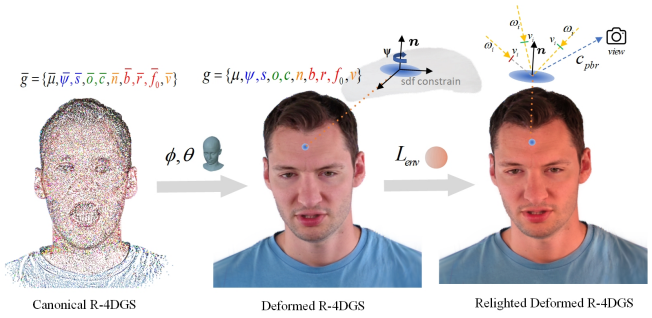

Fig. 3. The illustration of a person's relightable 4D Gaussian representation in canonical space (left) and its PBR rendering image in deformed space controlled by novel expression and pose parameters ϕ, θ (middle). Simultaneously, we can perform relighting using a changing environmental light L_{env} during novel pose/expression synthesis (right).

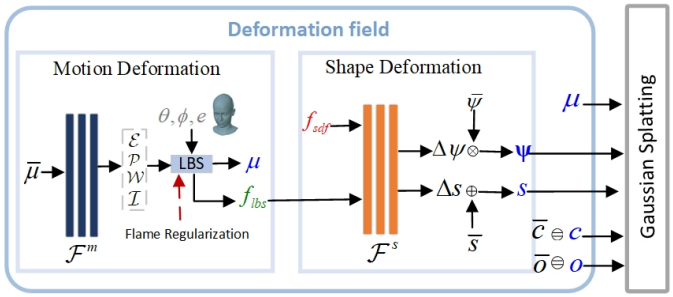

Fig. 4. The illustration of Deformation field. For R-4DGS's centroid position deformation, we utilize a network $\mathcal{F}^m$ to predict pose blendshapes $\mathcal{P}$, expression blendshapes $\mathcal{E}$, blend skinning weights $\mathcal{W}$ and eye blinking blendshapes $\mathcal{I}$. Then the linear blend skinning is performed using FLAME parameters θ, ϕ, e to calculate the dynamic centroid position μ and LBS deformation encoding f_{lbs}. For shape deformation, we employ a network $\mathcal{F}^s$, which takes f_{lbs} and SDF encoding f_{sdf} as input, to predict the the scale offset Δs and rotation offset $\Delta\psi$.

3.1 Relightable 4D Gaussians Representation

To enable relightable 4D Gaussian avatar reconstruction, we first introduce a new relightable 4D Gaussian representation, i.e., R-4DGS, which can perform Physically Based Rendering for relighting applications. In this paper, we define a 4D avatar's R-4DGS $\mathcal{G}(\phi, \theta, L_{env}) = \{g_i | i \in \mathbf{M}\}$ as a set of 4D Gaussians g_i ($\mathbf{M}$ is the number), which are deformed from their canonical space $\bar{\mathcal{G}} = \{\bar{g}_i | i \in \mathbf{M}\}$ using a deformation field $\mathcal{D}$.

Specifically, we define $g = \{\mu, \psi, s, o, c, n, b, r, f_0, v\}$ with μ, s, o, c representing the centroid position, scale, opacity and SH parameter respectively (as like the original 3DGS), n being the normal, ψ being the 1-DOF rotation angle around the normal, b, r, f_0 representing the BRDF material parameters albedo, roughness and fresnel reflectivity at normal incidence respectively, and v representing the light visibility. As shown in Fig. 3, we deform a canonical $\bar{g} = \{\bar{\mu}, \bar{\psi}, \bar{s}, \bar{o}, \bar{c}, \bar{n}, \bar{b}, \bar{r}, \bar{f}_0, \bar{v}\}$ to g using a deformation field $\mathcal{D}$, and compute its Gaussian-level PBR color $c_{pbr}(\omega_0)$ by performing Physically Based Rendering directly on each R-4DGS g for each given view ray ω_0. Finally, we compute the PBR color image by splatting the 4D Gaussians PBR color using a PBR splatting.

We group the parameters of each R-4DGS g into six parts: (1) *motion* part $g^m = \{\mu\}$ which is related to the dynamic motion of 4D Gaussians, (2) *shape* part $g^s = \{\psi, s\}$ which corresponds to the shape of each 4D Gaussian, (3) *appearance* part $g^a = \{o, c\}$ for the appearance component, (5) *material* part $g^{mat} = \{b, r, f_0\}$ and (6) *surface properties* part $g^l = \{n, v\}$ which counts for the surface properties for rendering. Since the *material* part g^{mat} represents the intrinsic materials of each 4D Gaussian, we keep this component fixed equivalent to the one in the canonical state, i.e., $g^{mat} = \bar{g}^{mat}$. For computation efficiency, we also keep the *appearance* part g^a fixed, i.e., $g^a = \bar{g}^a$.

Deformation field $\mathcal{D}$. As shown in Fig. 4, we build up the deformation field $\mathcal{D} = \{\mathcal{D}^m, \mathcal{D}^s\}$ with two components $(\mathcal{D}^m, \mathcal{D}^s)$, which deform the *motion* part g^m and *shape* part g^s from the corresponding canonical part $\bar{g}^m, \bar{g}^s$ respectively. Specifically, we follow PointAvatar [31] to construct $\mathcal{D}^m$, which deforms the centroid position $\bar{\mu}$ from canonical state to dynamic μ by combining the linear blend skinning (LBS) and an extra position offset field $\Delta\mu$ as:

$$\mu = \mathcal{D}^m(\bar{\mu}) \tag{1}$$
$$= LBS(\bar{\mu} + \mathcal{B}_P(\theta, \mathcal{P}) + \mathcal{B}_E(\phi, \mathcal{E}) + \mathcal{B}_I(e, \mathcal{I}), \theta, \mathcal{W}) + \Delta\mu,$$

where $\mathcal{B}_P(\cdot), \mathcal{B}_E(\cdot)$ compute the pose and expression offsets using the blendshape components $\mathcal{P}, \mathcal{E}$ respectively, $\mathcal{W}$ is the blend-skinning weights, θ, ϕ are the pose and expression parameters. Additionally, to adress the issue of eye blinking in PointAvatar [31] and FLARE [18], we compute extra eye blinking offsets $\mathcal{B}_I(\cdot)$ by incorporating the eye blinking parameters e and corresponding blendshape $\mathcal{I}$ from MICA [58]. Specifically, we train a MLP-based network $\mathcal{F}^m$ which, given canonical centroid position $\bar{\mu}$ as input, returns the pose blendshapes $\mathcal{P} \in R^{4\times9\times3}$, expression blendshapes $\mathcal{E} \in R^{100\times3}$,blend skinning weights $\mathcal{W} \in R^{n_j}$ and eye blinking blendshapes $\mathcal{I} \in R^{2\times3}$ i.e.,

$$\mathcal{F}^m(\bar{\mu}) \rightarrow \mathcal{P}, \mathcal{E}, \mathcal{W}, \mathcal{I}.$$

To construct the *shape* deformation $\mathcal{D}^s$, we train another MLP-based network $\mathcal{F}^s$ that, given the geometry feature f_g encoded from the canonical Gaussian state (see the details in Sec. 3.2) as input, returns the scale offset Δs and rotation offset $\Delta\psi$,

$$g^s = \bar{g}^s \oplus \mathcal{F}^s(f_g)$$
$$\Longrightarrow s = \bar{s} + \Delta s, \quad \psi = \bar{\psi} \otimes \Delta\psi. \tag{2}$$

PBR Rendering. Given any outgoing light direction ω_0, as shown in Fig. 3, we propose to compute the outgoing light radiance for each R-4DGS g by directly performing the Physically Based Rendering(PBR) . Specifically, following the rendering equation [59], the PBR color $c_{pbr}(\omega_0)$ for each g in the Gaussian level is computed as:

$$c_{pbr}(\omega_0) = \int_\Omega f(\boldsymbol{\omega}_o, \boldsymbol{\omega}_i) L(\omega_i) \Big(\omega_i \cdot \mathbf{n}\Big) d\omega_i, \tag{3}$$

where $f(\boldsymbol{\omega}_o, \boldsymbol{\omega}_i)$ models the BRDF properties of each g, $L(\omega_i)$ represents each 4D Gaussian g's incident light along the incoming light direction ω_i, n is the normal vector of g, and Ω signifies the hemispherical domain above the surface with the centroid position of g as the hemisphere center.

We adopt the simplified Disney BRDF model [60] and

divide the BRDF function $f(\boldsymbol{\omega}_o, \boldsymbol{\omega}_i)$ into diffuse term f_d and specular term f_s as:

$$f_d = \frac{\boldsymbol{b}}{\pi},$$
$$f_s(\boldsymbol{\omega}_o, \boldsymbol{\omega}_i) = \frac{D(\boldsymbol{h}; r) \cdot F(\boldsymbol{\omega}_o, \boldsymbol{h}; f_0) \cdot G(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o, h; r)}{(\boldsymbol{n} \cdot \boldsymbol{\omega}_i) \cdot (\boldsymbol{n} \cdot \boldsymbol{\omega}_o)} \quad (4)$$

where D is the microfacet distribution function, h is the half-vector, F is the Fresnel term, and G is the geometric attenuation factor.

As for the incident light $L(\omega_i)$, we assume that the person's monocular video is captured under a global environment light, and compute the incident light $L(\omega_i)$ for each 4D Gaussian g as:

$$L(\omega_i) = V(\omega_i) \cdot L_{env}(\omega_i), \quad (5)$$

where $V(\omega_i)$ is the light visibility term of each 4D Gaussian g and $L_{env}(\omega_i)$ is the global environmental light to be learnt. Both terms are parameterized as SH parameters.

So overall, when computing the PBR color for each 4D Gaussian in practical, we utilize Fibonacci sampling to sample N_s incident directions across the hemisphere space to numerically integrate the rendering equation [60]. The PBR color $c_{pbr}(\omega_o)$ of every 4D Gaussian g is then given by:

$$\boldsymbol{c_{pbr}(\omega_o)} = \sum_{i=0}^{N_s} (f_d + f_s(\boldsymbol{\omega}_o, \boldsymbol{\omega}_i)) L(\boldsymbol{\omega}_i)(\boldsymbol{\omega} \cdot \boldsymbol{n}) \Delta \boldsymbol{\omega}_i. \quad (6)$$

PBR Splatting. After we compute the PBR color $c_{pbr}(\omega_o)$ for each 4D Gaussian g, we render the PBR image C_{pbr} through a PBR splatting, which performs an alpha-blending of the PBR color $c_{pbr}(\omega_o)$ splatted on the image as:

$$C_{pbr} = \sum_{i \in \mathcal{N}_{cover}} T_i \alpha_i c^i_{pbr}, \quad T_i = \prod_{j=1}^{i-1} (1 - \alpha_j), \quad (7)$$

where $\mathcal{N}_{cover}$ is the set of 4D Gaussians which are splatted on the image covering the same image pixels.

Discussion. Given the definition of our R-4DGS above, we can render two kinds of color image using the original Gaussian splatting and our PBR splatting respectively. Specifically, we can render the appearance color image C_{app} by performing the 3D Gaussian splatting using the first five parameters of our R-4DGS, i.e., (μ, ψ, s, o, c), where C_{app} bakes the shading and shadow information together. Similarly, we can render the PBR color image C_{pbr} by performing our PBR rendering and splatting. In practical, we mainly use appearance color image C_{app} for the compact 4D reconstruction learning, and use the PBR color image C_{pbr} for the neural material and lighting decomposition respectively as shown in Sec. 3.2.2 and Sec. 3.3.2.

3.2 Compact 4D Gaussian Reconstruction

Previous 4D Gaussian avatars [33], [34], [35], [36] have shown impressive 4D Gaussian reconstruction by disentangling the 4D Gaussians' geometry from the appearance model. However, those solutions are not effective in our scenario, since we need to accurately decouple the 4D Gaussians' geometry, materials and incident lighting factors from the appearance rendering simultaneously, which is more complicated than those previous 4D Gaussian avatars. Instead, we propose to perform a *compact* 4D Gaussian reconstruction by forcing the 4D Gaussians *compactly* coherent to the underlying surface of the dynamic geometry, which is inspired by the recent progress for compact 4D Gaussian learning [61], [62].

3.2.1 Geometry Constraint Scheme

As shown in Fig. 2, we introduce an extra signed distance function (SDF) field in the canonical 4D Gaussian space, and leverage geometry and normal regularization from the SDF field to supervise a compact 4D Gaussian reconstruction learning.

SDF Field. As like PointAvatar [31], we create a MLP-based geometry network $\mathcal{F}^g$ that predicts the signed distance s for every position $x \in R^3$ in canonical 4D Gaussian space, i.e., $s = \mathcal{F}^g(x) \in R$. Using the geometry network $\mathcal{F}^g$, we propose to force the centroid position $\bar{\mu}$ of each canonical 4D Gaussian $\bar{g}$ to be located on the underlying surface $\mathcal{G}_s$ of the SDF field. The normal vector n_s for each on-surface point $x_s \in \mathcal{G}_s$ can be computed by the back propagation of the geometry network $\mathcal{F}^g$, i.e., $n_s = \frac{\partial \mathcal{F}^g}{\partial x_s}$.

Normal Regularization. On the other hand, after the canonical 4D Gaussian $\bar{g}$ is deformed to g in the deformed space, we force the normal vector n of g equivalent to deformed normal n_μ at μ measured by the deformation of SDF field. Following PointAvatar [31] and Eq. (1), we can compute the deformed normal from the n_μ at μ using

$$n_\mu = l_\mu \frac{\partial \mathcal{F}^g}{\partial \bar{\mu}} \left(\frac{\partial \mathcal{D}^m(\bar{\mu})}{\partial \bar{\mu}}\right)^{-1},$$

where l_μ is a normalizing scalar to ensure the normal is of unit length, $\bar{\mu}$ is the centroid position of canonical 4D Gaussian. $\bar{g}$. $\frac{\partial \mathcal{F}^g}{\partial \bar{\mu}}$ and $\frac{\partial \mathcal{D}^m(\bar{\mu})}{\partial \bar{\mu}}$ can be efficiently computed by back propagation of the MLP-based network $\mathcal{F}^g$ and $\mathcal{F}^m$ respectively. Thereafter, we set the normal vector n of g to be the deformed normal n_μ during the 4D Gaussian reconstruction, i.e., $n = n_\mu$.

So overall, during the 4D Gaussian reconstruction, we constraint the canonical 4D Gaussians located on the underlying surface of the SDF field (using the geometry regularization), and approximate the deformed normal vector of each 4D Gaussian equivalent to the deformed normal measured from the deformed space of the SDF field (using the normal regularization).

Geometry Features. Besides, as shown in Fig. 2, we also extract the geometry feature f_{sdf} of the SDF field to build up the geometry feature f_g in the shape deformation $\mathcal{D}^s$ as in Eq. (2), which also leverage geometry prior from SDF field to guide the 4D Gaussian's deformation field learning. Specifically, we set f_{sdf} as the latent vector of penultimate layer of the MLP-based network $\mathcal{F}^g$, and construct the geometry feature $f_g = \{f_{sdf}, f_{lbs}\}$ by concatenating f_{sdf} and the deformation encoding of the LBS deformation position f_{lbs}. We construct the deformation encoding f_{lbs} to represent the unique deformation state by concatenating the canonical centroid position $\bar{\mu}$, deformed centroid position μ, position offset $\Delta\mu$, normal n and an additional centroid position deformed only by pose parameters μ', which benefits to construct a more descriptive deformation encoding for better deformation field learning.

3.2.2 Compact Reconstruction Training Objectives

For a compact 4D Gaussian reconstruction, we optimize the canonical R-4DGS $\bar{\mathcal{G}} = \{\bar{g}_i | i \in \mathbf{M}\}$, three MLP-based networks $(\mathcal{F}^m, \mathcal{F}^s, \mathcal{F}^g)$ (for the motion deformation network, the shape deformation network and the geometry network of SDF respectively) by considering the following factors:

Appearance Rendering Loss. We perform the Gaussian splatting for the 4D Gaussians to achieve the appearance rendering image C_{app}, and use the input image C_{gt} as supervision for the 4D Gaussian reconstruction learning. Besides, we also use perceptual loss $\mathcal{L}_{\text{lpips}}$ [63] (with VGG [64] as the backbone) to measure the perceptional similarity between appearance rendering image C_{app} and supervision image C_{gt}, thus introducing a appearance rendering loss $\mathcal{L}_C$ as

$$\begin{aligned}\mathcal{L}_C =& \mathcal{L}_1(C_{app}, C_{gt}) + \lambda_{\text{lpips}}\mathcal{L}_{\text{lpips}}(C_{app}, C_{gt}) + \\ & \lambda_{\text{mouth}}\mathcal{L}_1(C_{app}\cdot\mathcal{K}_{mouth}, C_{gt}\cdot\mathcal{K}_{mouth})\end{aligned} \tag{8}$$

where $\mathcal{L}_1(\cdot)$ represents the L1 distance, $\mathcal{K}_{mouth}$ is the mouth mask, λ_{mouth},λ_{lpips} are weight parameters.

Flame Deformation Regularization. Similar with previous 4D avatars [6], [18], [31], we adopt to regularize the prediction of $\mathcal{F}^m$ in the motion deformation field $\mathcal{D}^m$ using the blendshapes and LBS weights from FLAME [41], by formulating a FLAME deformation regularization $\mathcal{L}_{\text{flame}}$ as

$$\begin{aligned}\mathcal{L}_{\text{flame}} =& \frac{1}{N}\sum_{i=1}^{N}(\lambda_e\|\mathcal{E}_i - \widehat{\mathcal{E}}_i\|_2 + \lambda_p\|\mathcal{P}_i - \widehat{\mathcal{P}_i}\|_2 + \\ & \lambda_w\|\mathcal{W}_i - \widehat{\mathcal{W}_i}\|_2) + \lambda_I\|\mathcal{I}_i - \widehat{\mathcal{I}}_i\|_2.\end{aligned} \tag{9}$$

where $\mathcal{E}$,$\mathcal{P}$,$\mathcal{I}$,$\mathcal{W}$ are the predicted expression, pose, eyeblinking blendshapes and blend skinning weights respectively, and $\widehat{\mathcal{E}}$,$\widehat{\mathcal{P}}$,$\widehat{\mathcal{W}}$,$\widehat{\mathcal{I}}$ are those from FLAME model.

SDF Regularization. With the aid of SDF field, we can trace the underlying surface $\mathcal{G}_s$ by extracting the zero level set of $\mathcal{F}^g$. Besides, we can also use the Eikonal regularization for all the sampled points $\mathbf{x_e}$ in the field, i.e., $|\nabla x_e| = 1$. Specifically, we use a SDF loss L_{sdf} performed on the centroid positions of all canonical 4D Gaussians, and a Eikonal loss L_{eik} performed on the sampled positions as

$$\mathcal{L}_{\text{sdf}} = \|\text{SDF}(\bar{\mu})\|^2, \tag{10}$$

$$\mathcal{L}_{\text{eik}} = (\|\nabla_{\mathbf{x_e}}\text{SDF}(\mathbf{x_e})\| - 1)^2. \tag{11}$$

So overall, we combine the above regularization loss functions to formulate a 4D Gaussian reconstruction loss $\mathcal{L}_r$ during the 4D Gaussian reconstruction learning as

$$\mathcal{L}_r = \lambda_c\mathcal{L}_C + \lambda_{flame}\mathcal{L}_{\text{flame}} + \lambda_{sdf}\mathcal{L}_{\text{sdf}} + \lambda_{eik}\mathcal{L}_{\text{eik}} \tag{12}$$

Besides, to accurate reconstruct the interior area of mouth, we also introduce a mouth interior modeling during the compact 4D Gaussian reconstruction, please refer to the Appendix for more details.

3.3 Neural Material and Lighting Decomposition

After the 4D Gaussians are compactly reconstructed, we also estimate the 4D Gaussians' material and incident light factors by decomposing them from the appearance. According to the PBR rendering for each 4D Gaussian in Eq. (6), since the BRDF material parameters b, r, f_0 and global environmental lighting L_{env} are intrinsic components and keep fixed, the only unknown component that we need to dynamically estimate is the light visibility v.

3.3.1 Geometry-Aware Light Visibility Prediction

For efficient computation, we use another MLP-based network $\mathcal{F}^v$ that, taking the geometry feature f_g as input, returns the SH coefficients of dynamic light visibility, i.e., $v = \mathcal{F}^v(f_g)$. What's more, for a more accurate material and lighting decomposition, we propose to leverage geometry prior from the compact 4D Gaussian reconstruction, which can make geometry-aware light visibility prediction as strong regularization thus leading to more accurate estimation of materials and incident lighting. Specifically, we construct the light visibility network $\mathcal{F}^v$ and shape deformation network $\mathcal{F}^s$ with the same network structure, and share weights between $\mathcal{F}^v$ and $\mathcal{F}^s$. In this way, we implicitly decode the light visibility using the geometry prior from the compact 4D Gaussian reconstruction to make accurate neural material and lighting decomposition. Furthermore, it is often too time-consuming to get the accurate visibility supervision from ray tracing, which will significantly decrease the training speed. To make a more efficient training, we propose to learn visibility using F^v with effective regularization, recovering reasonable visibility estimation.

3.3.2 Decomposition Training Objectives

To effectively decouple the geometry, materials and incident lighting, we jointly learn the canonical R-4DGS $\bar{\mathcal{G}} = \{\bar{g}_i | i \in \mathbf{M}\}$, four MLP-based networks $(\mathcal{F}^m, \mathcal{F}^s, \mathcal{F}^g, \mathcal{F}^v)$ and the global environmental lighting L_{env} by considering the following factors:

PBR Rendering Loss. We perform the PBR splatting for the 4D Gaussians to achieve the PBR rendering image C_{pbr}, and use the input image C_{gt} as supervision for the 4D Gaussian reconstruction learning, which formulates a PBR rendering loss $\mathcal{L}_{pbr}$ as

$$\mathcal{L}_{pbr} = \mathcal{L}_1(C_{pbr}, C_{gt}) + \lambda_{\text{lpips}}\mathcal{L}_{\text{lpips}}(C_{pbr}, C_{gt}). \tag{13}$$

Shading Regularization. As like previous work [18], [65], we adopt a white light regularization over the shading component with a regularization loss $\mathcal{L}_{light}$ as

$$\mathcal{L}_{light} = \sum_c(\mathcal{S}_c - \frac{1}{3}\sum_i \mathcal{S}_i), i, c \in \{\mathcal{R}, \mathcal{G}, \mathcal{B}\}. \tag{14}$$

where $\mathcal{S}_c$ is one channel of the diffuse shading.

BRDF Bilateral Smoothness. Besides, we also use the bilateral smoothness [65] to regularize the roughness, albedo and fresnel reflectivity parameters, but additionally adding a skin region roughness smoothness to weight the roughness in skin region not change too drastically as

$$\begin{aligned}&\mathcal{L}_{smooth,R} = \|\nabla R\|\exp(-\|\nabla C_{gt}\|) + \\ &\lambda_{skin}\|\nabla(R\cdot\mathcal{K}_{skin})\|\exp(-\|\nabla C_{gt}\cdot\mathcal{K}_{skin}\|),\end{aligned}$$

where R is the splatted (rendered) roughness map, given by $\sum_{i\in N} T_i\alpha_i r_i$, $\mathcal{K}_{skin}$ is the preprocessed skin mask. Similarly, the fresnel reflectivity and albedo smooth loss are:

$$\mathcal{L}_{smooth,F_0} = \|\nabla F_0\|\exp(-\|\nabla C_{gt}\|),$$

$$\mathcal{L}_{smooth,B} = \|\nabla B\|\exp(-\|\nabla C_{gt}\|),$$

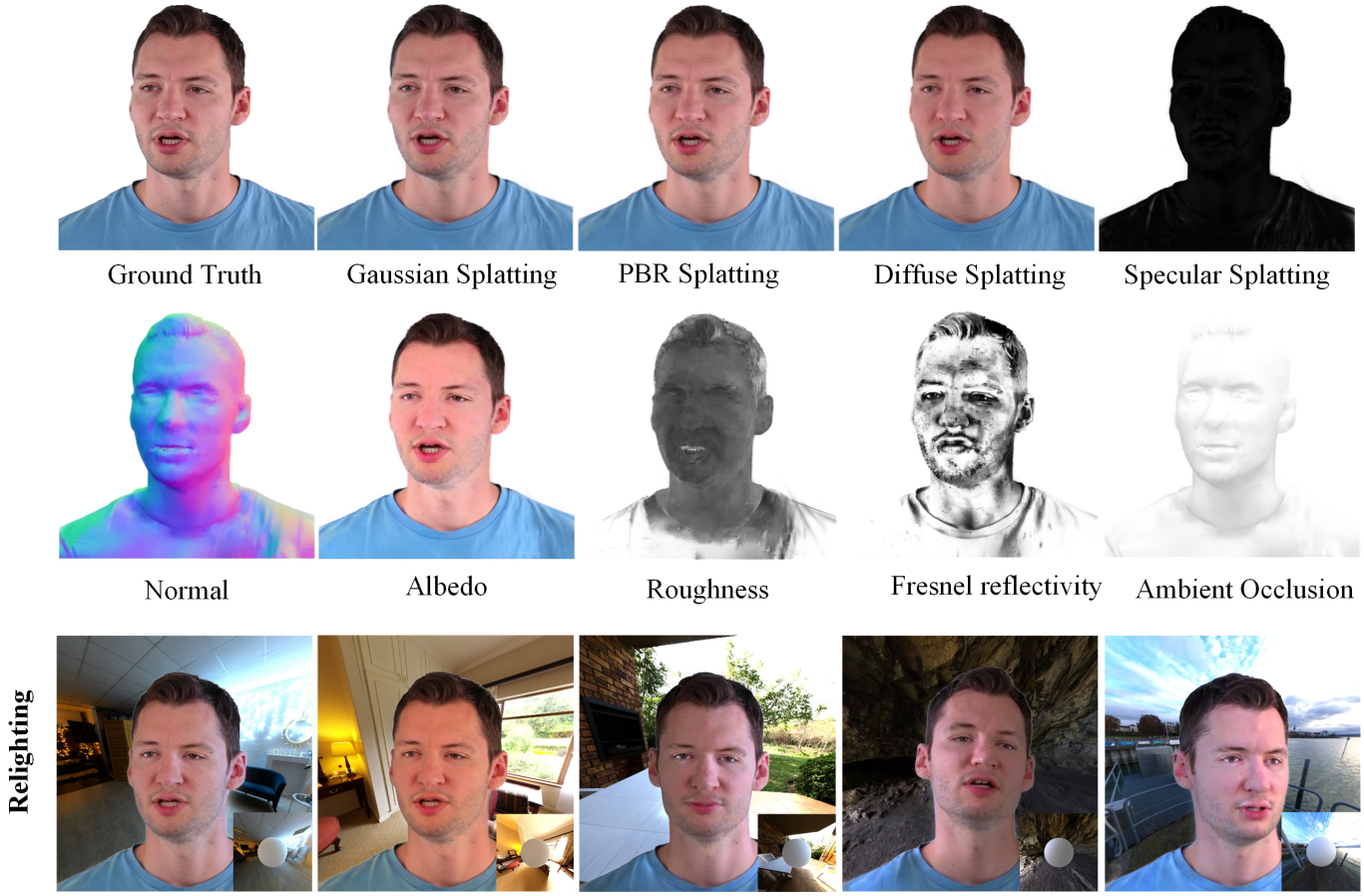

Fig. 5. The visual results of our R-4DGS reconstruction of different subjects. Specifically, We show the diffuse/specular components, normal, BRDFs and relighting results under different changing environmental lights.

where F_0 is the splatted fresnel reflectivity map, B is the splatted albedo map. The full BRDF smooth loss is sum:

$$\mathcal{L}_{smooth} = \mathcal{L}_{smooth,R} + \mathcal{L}_{smooth,F_0} + \mathcal{L}_{smooth,B} \tag{15}$$

Albedo Regularization. We realize the regularization of the albedo prediction using an albedo regularization $\mathcal{L}_b$ as

$$\mathcal{L}_b = \mathcal{L}_1(B, B_{target}) + \lambda_{\text{b}}\mathcal{L}_{SSIM}(B, B_{target}), \tag{16}$$

where B, B_{target} are the splatted albedo map and referential albedo map with reduced shadows and highlights, respectively. The determination of B_{target} relies on an approximation method. For details regarding the calculation of B_{target}, Please refer to the Appendix.

Visibility Regularization. The visibility term inferred by $\mathcal{F}^v$ sometimes can be entangled with the lightness of skin tone. To capture shadows rather than skin tone during the optimization process, we introduce a novel visibility regularization term, encouraging the ambient occlusion map [65](averaged visibility) C_V that approximates white. This decoupling process leads to a more reasonable visibility.

$$\mathcal{L}_V = \mathcal{L}_1(C_V, C_{white}). \tag{17}$$

Overall, we combine the above regularization loss functions together to learn an accurate material and lighting decomposition using loss function $\mathcal{L}_{ml}$ as

$$\mathcal{L}_{ml} = \lambda_{pbr}\mathcal{L}_{pbr} + \lambda_l\mathcal{L}_{light} + \lambda_{albedo}\mathcal{L}_b + \lambda_s\mathcal{L}_{smooth} + \lambda_v\mathcal{L}_V. \tag{18}$$

4 Experiments

4.1 Implementation Details

Data Reprocessing. For the given input RGB frames, we adopt MICA [58] as 3D avatar tracker to extract the pose and expression parameters for preprocessing of each frame. Besides, we also extract eye-blinking parameters by MICA, and perform the linear blend skinning (LBS) in the motion deformation field by combining the pose blendshapes, expression blendshapes, blend skinning weights from FLAME model [41] and eye-blinking blendshapes from MICA.

Lightweight Networks. We use a lightweight MLP to construct $\mathcal{F}^m$ in the motion deformation field with network structure as $[128, 128, 128, 128]$, and use softplus as activation function to predict the shape, pose, expression blendshape components. Similarly, we also use a four layers MLP to construct $\mathcal{F}^s$ and $\mathcal{F}^v$, with network structure is $[256, 256, 256, 256]$ to predict the shape components of 4D Gaussian and light visibility respectively. For the geometry network $\mathcal{F}^g$ in the SDF field, we use the similar MLP provided by PointAvatar [31] to predict signed distance. Please refer to our Appendix for the details of the network structures of $\mathcal{F}^m$, $\mathcal{F}^s$,$\mathcal{F}^v$ and $\mathcal{F}^g$ respectively.

Training Details. For compact 4D Gaussian reconstruction, we set the maximum training epoch as 60. For neural material and lighting decomposition, we set it as 25 epochs. In each training epoch, we randomly sample half of the frames from the training set as supervision for both appearance render loss and PBR rendering loss respectively. We implemented the full system using PyTorch framework, and adopt the Adam optimizer to perform RGAvatar learning. All of the training and evaluation experiments are performed on a platform with one RTX 3090Ti GPU.

Canonical R-4DGS Initialization. We adopt the same strategy from previous approaches [34], [35] to perform the canonical R-4DGS initialization. Specifically, we uniformly sample 28000 points from the canonical FLAME template mesh as centroid positions of the canonical R-4DGS initialization. After initialization, similar to FlashAvatar [35], we fix the number of Gaussians and avoid pruning and densification during the optimization process to reduce instability such as Gaussians number explosion for efficiency.

4.2 Dataset

To perform training and evaluation of RGAvatar, we randomly collect a dataset with individual persons' fixed viewport monocular videos from publicly released datasets, i.e., NeRFace [3], NHA [4], NeRFBlendShape [10], PointAvatar [31], and INSTA [17], which count for 12 subjects. Besides, we also captured extra 3 subjects using a fixed viewport webcam, thus making in total a dataset with 15 subjects in our collected dataset to perform the training and evaluation. In average, each subject's monocular video has around 3000 RGB frames, with image resolution set as 512×512. During the network training, we sample half of the frames for training, and the left frames for evaluation.

4.3 R-4DGS Reconstruction Evaluation

We first evaluate the 4D avatar reconstruction quality of RGAvatar, including the R-4DGS' geometry (4D Gaussians and normal), BRDF materials (albedo, roughness and Fresnel reflectivity). Besides, we also show the ambient occlusion map, which is the averaged visibility of sampled incident light directions and is rendered using this averaged visibility value $v_{avarage}$for splatting, given by $\sum_{i \in N} T_i \alpha_i v_{avarage}(i)$. Fig. 5 shows the R-4DGS reconstruction results of one subject from our collected dataset. Benefiting from compact 4D Gaussian reconstruction, we can see that our approach can accurately reconstruct the 4D Gaussians compactly approximate the underlying surface of the 4D avatar's dynamic geometry, and the splatted normal image also demonstrates accurate prediction of the normals, which are coherent to the geometry.

Relighting Evaluation. Based on the accurate R-4DGS reconstruction, we then evaluate the relighting quality by giving various changing environmental lights. As shown in Fig. 5, we demonstrate the relighting results when given some environmental lights, from soft lights to brighter lights. We can see that our approach can achieve high realistic relighting quality given different environmental lighting robustly. Besides, we also perform relighting under different novel pose and expression synthesis. As shown in Fig. 5, when the 4D avatar is animated using novel pose and expression parameters, our approach can also achieve high quality relighting results, which is benefit from the accurate relightable 4D Gaussians learning by our approach. Please refer to our supplemental video for dynamic visualization of the results.

TABLE 1
Quantitative comparison on the test set of our collected dataset from different comparison approaches, including INSTA [17], FlashAvatar [35], SplattingAvatar [37] (abbreviated as SPLAvatar for simplicity), PointAvatar [31], FLARE [18] and Ours, in terms of PSNR, SSIM, LPIPS, MSE and L1 accuracy metrics.

Method	PSNR ↑	SSIM ↑	LPIPS ↓	MSE ↓	L1↓
PointAvatar	28.6733	0.9279	0.0564	0.0375	0.0117
FlARE	25.8226	0.9097	0.0623	0.05319	0.0139
INSTA	27.2983	0.9423	0.0653	0.0485	0.0129
SPLAvatar	28.8963	0.9390	0.0540	0.0382	0.0109
FlashAvatar	28.5026	0.9474	0.0447	0.0379	0.0113
Ours	29.5255	0.9395	0.0414	0.0346	0.0095

4.4 Comparisons with State-of-the-Art 4D Avatars

To evaluate the effectiveness of our approach, we conduct comparison experiments between our approach and the state-of-the-art 4D avatars for monocular reconstruction, including INSTA [17], FlashAvatar [35], SplattingAvatar [37], PointAvatar [31] and FLARE [18]. Here the previous five avatars can be divided into two types: (1) non-relightable 4D avatars such as INSTA, FlashAvatar and SplattingAvatar, where we view INSTA as the representative method for previous NeRF-based 4D avatars, and FlashAvatar and SplattingAvatar are two recent 4D Gaussian avatars. (2) Relightable 4D avatars including PointAvatar and FLARE.

When performing the comparison, we use the public release demo code for INSTA[1], FlashAvatar[2], SplattingAvatar[3], PointAvatar[4] and FLARE[5] with the default settings for a fair comparison. Besides, since PointAvatar requires up to 80GB GPU memory storage which is much larger than other comparing approaches, we downsample the points number following the author's suggestions to reduce GPU memory consumption for fair comparison.

Quantitative Comparison. We first conduct quantitative comparison between these comparison approaches, by evaluating the rendering quality of predicted images in terms of accuracy metrics including PSNR, SSIM, LPIPS, MSE and L1 metrics, on the test set of our collected dataset. Table 1 shows the quantitative comparing results of those comparing approaches. We can see that our approach can achieve better PSNR accuracy than previous approaches for high fidelity rendering quality, and consistently better LPIPS, MSE and L1 accuracy values, which means that our approach can get consistently better 4D avatar reconstruction quality than previous approaches.

1. https://github.com/Zielon/INSTA
2. https://github.com/USTC3DV/FlashAvatar-code
3. https://github.com/initialneil/SplattingAvatar
4. https://github.com/zhengyuf/PointAvatar
5. https://github.com/initialneil/SplattingAvatar

Fig. 6. The visual comparison results for different 4D avatars, including INSTA, FlashAvatar, SplattingAvatar, PointAvatar, FLARE and Ours.

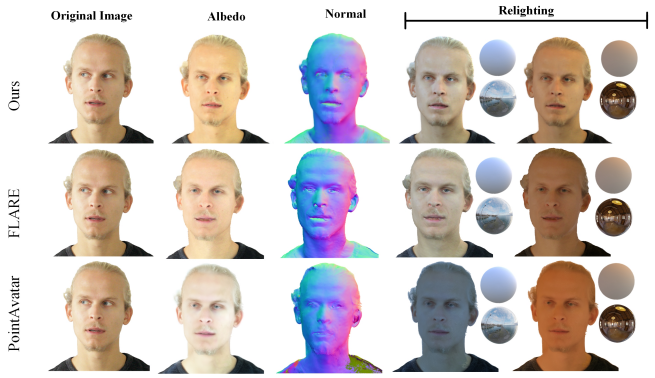

Fig. 7. The relighting comparison between PointAvatar [31], FLARE [18] and our approach, by giving different environmental lights.

One main reason that our approach outperforms previous 4D avatars quantitatively is the benefit from our compact 4D Gaussian reconstruction. Compared with previous Gaussian-based 4D avatars, our compact 4D Gaussian reconstruction provides more accurate geometry and appearance disentanglement for the 4D Gaussian learning, thus leading to more accurate results than those previous 4D avatars.

Qualitative Comparison. We also make qualitative comparison between those comparing approaches. Fig. 6 depicts the qualitative comparison between our approach and the other five comparing approaches. We can observe that the results of FLARE are limited mainly due to the limited expressive capability for complicated 4D avatar's geometry and deformation, showing relatively low visual quality with blur artifacts in teeth and mouth rendering. PointAvatar tends to exhibit point cracks during deformations and is not fine enough when the points number is insufficient. The results of INSTA tend to to be smooth, but doesn't faithfully reproduce enough details on some regions. In contrast, our approach can achieve better image rendering quality. Compared with other two 4D Gaussian avatars (FlashAvatar and SplattingAvatar), our approach can achieve the finest tooth details and the most faithful lips in the rendered images. From the zoom-in images, we can also see that our approach can produce more detailed surface textures, such as the nose in first row. Furthermore, benefit from our compact 4D Gaussian reconstruction, our approach exhibits minimal surface artifacts when representing complex features like beards (see the first row comparison results).

Relighting Comparison with PointAvatar and FLARE. To our best knowledge, PointAvatar [31] and FLARE [18] are two monocular-based relightable 4D avatars, which are most relevant to ours. Both methods predict normals and BRDFs. However, PointAvatar only predicts the diffuse BRDF(albedo), restricting the method to use diffuse shading models like the Lambertian model for relighting, while FLARE estimates both the diffuse (albedo) and specular (roughness and specular intensity) components, leading to natural relighting renderings with specular highlights. To

demonstrate the effectiveness on relighting applications, we conduct comparing experiments qualitatively with PointAvatar and FLARE using different environmental lighting. As shown in Fig. 7, our approach predicts more faithful albedo and achieves significantly better relighting results than PointAvatar and FLARE.

The alpha blending mechanism and flexible representation of 3D Gaussian Splatting plays important role in achieving natural relighting results, which is validated by the success of 3D Gaussian Splatting. But more importantly, our compact 4D Gaussian reconstruction and effective material and lighting decoupling mechanism serve as the main factor for the high quality and natural relighting compared to FLARE and PointAvatar. Please refer to our supplementary materials for more visual comparison relighting results between different approaches.

4D Gaussian Reconstruction Comparison. Different from previous 4D Gaussian avatars, like FlashAvatar and SplattingAvatar, our RGAvatar introduces effective geometry regularization from an extra SDF field, leading to more compact 4D Gaussian reconstruction. To demonstrate this point, we compare our approach with FlashAvatar and SplattingAvatar in terms of the 4D Gaussians' geometry reconstruction. Fig. 8 shows several 4D Gaussian geometry comparison by visualizing the deformed centroid positions $\bar{\mu}$ (point clouds) in varying poses, expressions and views. Since FlashAvatar and SplattingAvatar only model the head and neck regions, we present the comparison results for these areas. We can see that our approach can achieve more accurate 4D Gaussian geometry, with much better compactness coherent to the underlying surface.

To better illustrate our jointly optimized SDF, we extracted the on-surface meshes from the learned SDF and compared them with FLARE's canonical meshes qualitatively. Several comparison results are presented in Fig. 9. We can see that both methods can reconstruct reasonable head avatar geometry. However, the meshes reconstructed by FLARE exhibits some aliasing artifacts and holes. In contrast, the surface reconstructed by our SDF is more continuous and smoother with fewer artifacts, which can bring more faithful prior to guide the 4D Gaussian learning.

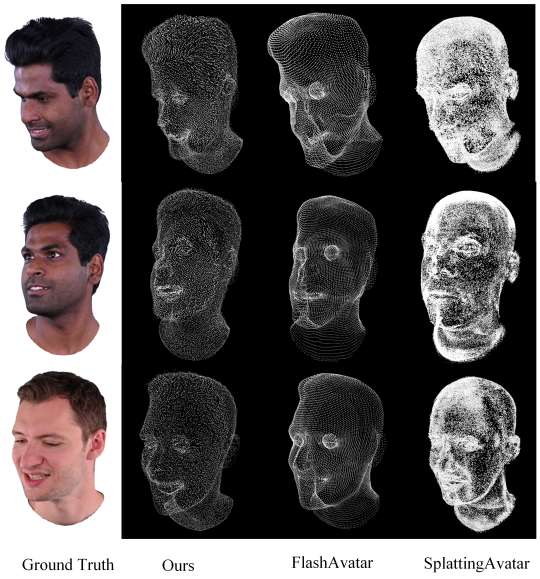

Fig. 8. The 4D Gaussian geometry reconstruction comparison between FlashAvatar [35], SplattingAvatar [37] and our approach. All these methods initialize the centroid position $\bar{\mu}$ using standard FLAME template mesh. However, our approach achieves more accurate geometry by incorporating SDF field constraint in conjunction with our tailored training strategy.

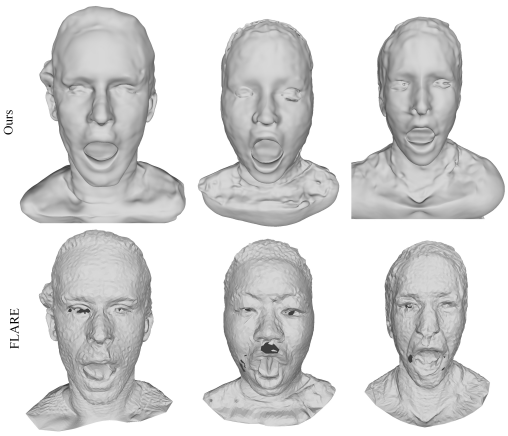

Fig. 9. Some visual comparison results of the surface extracted by our learned SDF (top row) and FLARE [18] (bottom row).

4.5 Ablation Study

SDF Field constraint. In order to verify the impact of the SDF field on the compact 4D Gaussian reconstruction, we implemented an additional version of our system that removes the SDF field (termed as 'w/o sdf') and experimentally compared with our full system (termed as 'w sdf'), by evaluating the image rendering accuracy on the test set of our collected dataset. Table 2 shows the quantitative comparison between such two systems. We can see that without using the Constraints from SDF field, the accuracy metrics including PSNR, SSIM, LPIPS, MSE and L1 consistently decrease compared with our full system. This shows that the extra SDF field takes effect for better 4D Gaussian reconstruction, thus achieving better image rendering results. Fig. 10 also shows a visual comparison results between such two systems. We can see that when using the SDF field constraint, our full system can achieve better reconstruction results with more accurate details, especially realizing compact surface reconstruction and exhibits significantly reduced surface artifacts benefit from our geometry Regularization and normal regularization. Besides, Fig. 11 demonstrates some relighting results between 'w sdf' and 'w/o sdf', we can see that without using the strong constraints of SDF for the 4D Gaussians reconstruction, the geometry, material and incident lighting couldn't be accurately decomposed by 'w/o sdf', which leads to unsatisfactory relighting results. But our full system can obtain more accurate materials estimation for high quality relighting.

Geometry-aware Light Visibility Prediction. One important point for accurate neural material and lighting decomposition is the geometry-aware light visibility prediction as described in Sec. 3.3. To evaluate the effectiveness of the designed geometry-aware light visibility prediction,

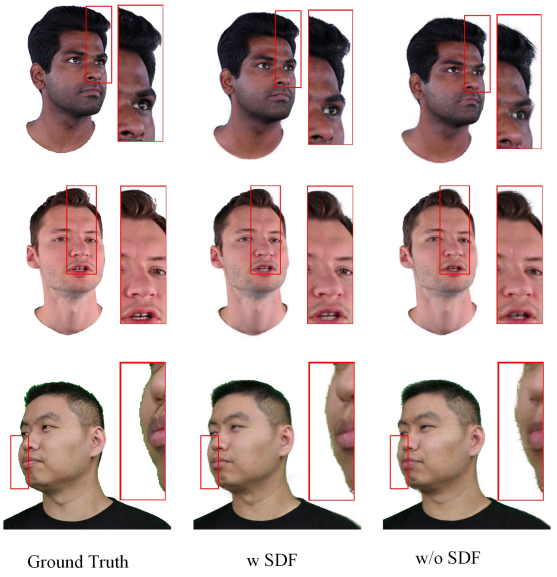

Fig. 10. The visual comparison results by two system variants with (left) or without (right) the SDF field constraints for the 4D Gaussian reconstruction respectively.

TABLE 2
Quantitative comparison between two variant systems with ('w sdf') or without ('w/o sdf') SDF field constraints.

Method	PSNR ↑	SSIM ↑	LPIPS ↓	MSE ↓	L1↓
w/o sdf	26.9784	0.9175	0.0478	0.0481	0.01282
w sdf	27.9482	0.9246	0.0411	0.0428	0.0114

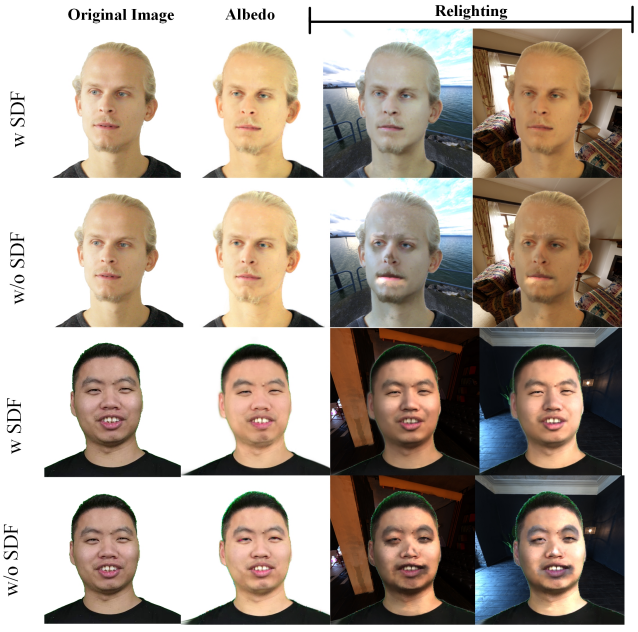

Fig. 11. The visual comparison results by two system variants with or without the SDF field constraints for the 4D Gaussian relighting respectively.

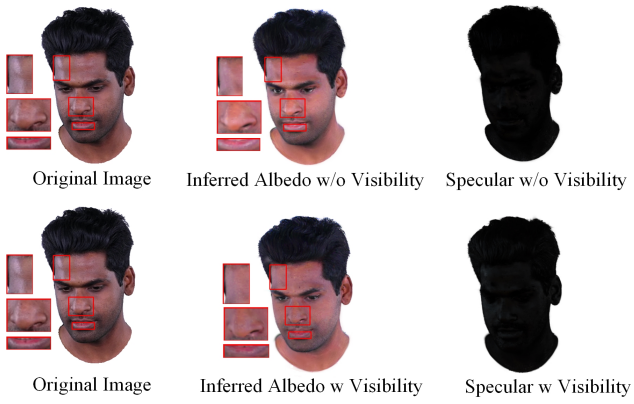

Fig. 12. Visual comparison for predicted albedo rendering with (bottom middle) or without (top middle) geometry-aware light visibility prediction. The right column shows the decoupled specular shading(related to roughness and fresnel reflectivity).The comparison between the first and second row indicates that with the geometry-aware light visibility prediction, the system can disentangle more accurate albedo and specular component from original images.

Fig. 13. The visual comparison for performing relighting with and without visibility $V(\omega_i)$ in Eq. (5). The right column shows ambient occlusion(averaged visibility) of 2 poses.

we implemented a variant of our full system (termed as 'w/o visibility') by removing the light visibility prediction module, and compare with our full system (termed as 'w visibility') on the evaluation. As shown in Fig. 12, without using the geometry-aware light visibility the albedo prediction will retain shadow and highlight information, while our full system can predicts more accurate intrinsic albedo and disentangles faithful specular shading component with more highlights from the original image.

Besides, Fig. 13 shows a visual comparison result for relighting with or without visibility. We can see that the relighting results using visibility handles reasonable self-occlusion and have better local shadow effects.

Neural Material and Lighting Decomposition. For the reflectance modeling, previous approaches like FLARE [18] and RGCA [38] proposed to use an separated MLP to predict the appearance, while we model the reflectance according to the Physically Based Rendering process, and learn to accurately decompose the material and incident lighting. To show the difference between such two reflectance modeling, we conduct experimental comparison. Specifically, we also implement a variant system of our approach by using the similar reflectance modeling like FLARE [18] and

Fig. 14. Visual comparison results of albedo, roughness and PBR image reconstruction from two reflectance modeling strategies, including the previous strategy like FLARE [18] (top row) and ours (bottom row) respectively.

RGCA [38], termed as 'MLP reflectance', and compare with our full system. As shown in Fig. 14, we can see that our approach can achieve better albedo, roughness and PBR rendering than the previous strategy, with more realistic appearance reconstruction from the input image.

In Fig. 15, we show the evolution of our learned environment map during training process. As we can see from the estimation of both irradiance and albedo respectively, our approach can effectively disentangle lighting color from skin tone, gradually decoupling the plausible environmental lighting and the albedo.

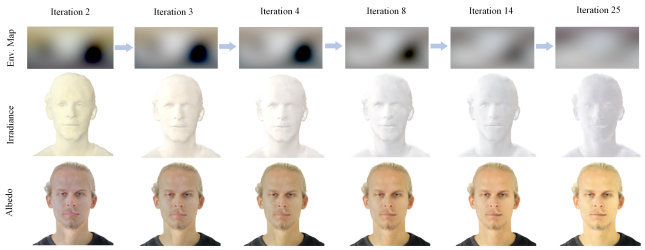

Fig. 15. The visualization of the learned environment map (top row) during the training process of our approach. The corresponding irradiance(middle row) and albedo map (bottom row) are also demonstrated.

SDF Constraint V.S. Flame Mesh. To provide a more comprehensive comparison, we conducted an ablation experiment comparing the performance of our method when bound to a jointly optimized FLAME mesh versus our SDF constraint. As shown in Fig. 16, it is evident that the geometry surface reconstructed by optimized SDF is more continuous and smoother, approximating the geometry of a real person. Similarly, the normals of the Gaussian avatar anchored to SDF exhibits the same characteristics. In contrast, the bounded FLAME mesh is constrained by the mesh resolution with accuracy limited and prone to artifacts. Finally, our approach can achieve better relighting results. Please refer to the Appendix for more details.

4.6 Time Efficiency Analysis

we carefully made a time efficiency analysis on our approach by comparing with previous approaches such as FLARE and PointAvatar. Table 3 shows the per-frame rendering time during the relighting for different comparing approaches. As we can see from the table, our approach can

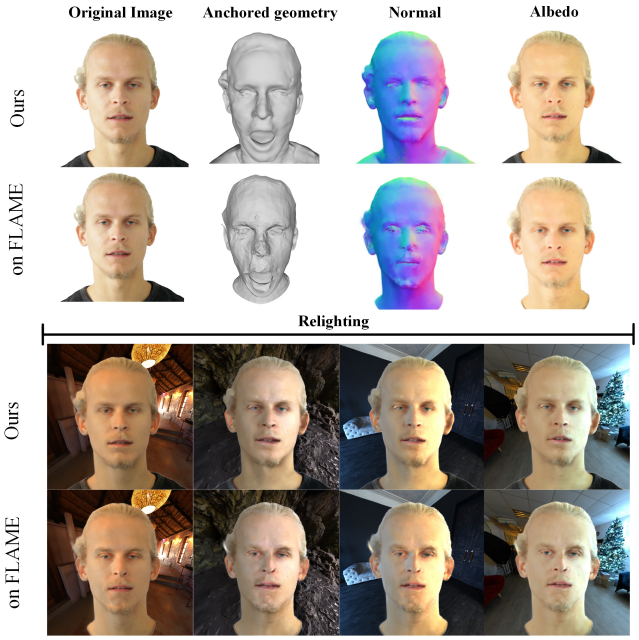

Fig. 16. Visual comparison results between two geometry constraint strategies, including the SDF regulation(ours) and bound Gaussians to a jointly optimized FLAME mesh respectively.

TABLE 3
Rendering efficiency comparision with other relighting methods. The rendering time here refers to the average total running time required to generate each relighted image. Ours (28000) means that the Gaussians' number is set as 28000 for the time evaluation.

Method	rendering time per image↓	FPS↑
PointAvatar	0.3968s	2.52
FlARE	0.1849s	5.41
Ours(28000)	0.0622s	16.06

achieve 16FPS rendering speed on average, while FLARE is 5.4 FPS and PointAvatar is 2.5 FPS (evaluated on the same RTX 3090 GPU device), which means that our approach is significantly faster than the two previous approaches.

More specifically, we split our rendering process into the deformation stage (including gaussians deformation and visibility prediction) and the subsequent PBR stage (including integrating rendering equations and rasterization), and separately measured their running time. For per-frame rendering of our approach (with the number of Gaussians set to 28,000), the deformation stage takes an average of 35ms, with motion deformation and normal deformation (related to deep geometry network $\mathcal{F}^g$ and network $\mathcal{F}^m$) accounting for 30ms, and shape deformation and visibility prediction (related to lightweight $\mathcal{F}^s$ and $\mathcal{F}^v$) accounting for 5ms. Obviously, the primary process affecting our deformation efficiency is the geometry constraint in motion/normal deformation, which is mainly due to the deep network structure and gradient computations for normal. The PBR stage takes an average of 27ms(with the number of integration samples set to 320).

5 Limitation and Discussion

RGAvatar consists some limitations. First, our current solution couldn't accurately model the material and incident light of human's eyes and eyeglasses due to their complex BRDF properties, thus often leading to unsatisfied material/lighting estimation for eye regions. Second, We model the illumination using a global environmental lighting combined with visibility, however, would fail to capture complicated illumination such as small sharp specular highlights on the face in some cases. This could be further enhanced by introducing a more complicated but realistic Physically Based Rendering illumination modeling for better results.

Besides, our approach would sometimes get unwanted silhouette in the boundary regions of the relighting results, which is mainly due to the limited precision of the portrait segmentation in our data preprocessing step. One straightforward solution is to use more accurate portrait segmentation such as SAM [66]. Since more accurate portrait segmentation is not the main contribution in this paper, we leave it for future work. Flickering artifacts would sometimes exist for mouth regions, which would mainly come from the limited precision of the preprocessed facial expression captured by MICA, bring inconsistencies for local facial regions especially when performing across-reenactment for expressions transferring across different identities. One possible solution would be to use more accurate 3DMM tracker (more accurate than MICA) which can also faithfully reconstruct any exaggerated expressions. But since it is a common challenge for most 4D avatar reconstruction approaches including ours, we also leave this as an interesting future work.

Ethic. The goal of RGAvatar is to enable a high fidelity, subject-specific relightable 4D avatar creation, which can be used to generate virtual portraits synthesis given novel poses/expressions. However, this would provide ways for new malicious content by training a subject from their monocular video on the internet and generating new content without their consent. Although the quality of RGAvatar has identifiable artifacts by current AI approaches or tool, the rapid progression of the field suggests these cues may diminish over time. Addressing this remains an important technical and legal challenge.

6 Conclusions

In this paper, we propose a new relightable 4D Gaussian avatar from monocular videos, by introducing a new relightable 4D Gaussian representation, and a joint learning mechanism for accurate R-4DGS learning. Our RGAvatar supports better relighting with significantly improved rendering quality and efficient rendering speed, which to our best knowledge becomes a new state-of-the-art relightable 4D avatar approach. We think the way we learn a compact 4D Gaussian reconstruction, while introducing geometry-aware light visibility prediction for accurate material and incident lighting decomposition, can bring fresh ideas for the 4D avatar creation community, especially for a high fidelity relightable 4D avatar while performing relightable rendering in a quite efficient manner.

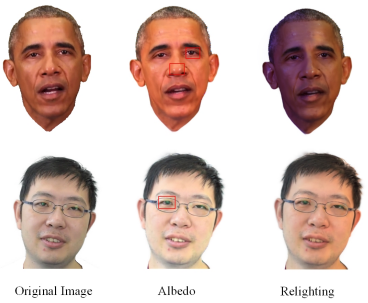

Fig. 17. Limitations. Our method is unable to handle some complicated lighting conditions, thereby sometimes failing to remove subtle specular highlights on the face (first row), leading to the decrease in accuracy of estimated albedo. Additionally, our approach does not model the complex material properties in eye regions, including the eyeball (first row) and eyeglasses (second row), which results in unsatisfied material/lighting estimation and relighting results for these regions.

Acknowledgments

We thank the reviewers for their insightful comments and suggestions. This work was supported in part by the National Natural Science Foundation of China(62202057, 62272433, 62402468). Specifically, Shi-Sheng Huang was supported by NSFC 62202057, Juyong Zhang was supported by NSFC 62272433, Yudong Guo was supported by NSFC 62402468, respectively.

References

[1] B. Egger, W. A. Smith, A. Tewari, S. Wuhrer, M. Zollhoefer, T. Beeler, F. Bernard, T. Bolkart, A. Kortylewski, S. Romdhani *et al.*, "3d morphable face models—past, present, and future," *ACM TOG*, vol. 39, no. 5, pp. 1–38, 2020.

[2] M. Zollhöfer, J. Thies, P. Garrido, D. Bradley, T. Beeler, P. Pérez, M. Stamminger, M. Nießner, and C. Theobalt, "State of the art on monocular 3d face reconstruction, tracking, and applications," in *Computer Graphics Forum*, vol. 37, no. 2, 2018, pp. 523–550.

[3] G. Gafni, J. Thies, M. Zollhofer, and M. Nießner, "Dynamic neural radiance fields for monocular 4d facial avatar reconstruction," in *IEEE CVPR*, 2021, pp. 8649–8658.

[4] P.-W. Grassal, M. Prinzler, T. Leistner, C. Rother, M. Nießner, and J. Thies, "Neural head avatars from monocular rgb videos," in *IEEE CVPR*, 2022, pp. 18 653–18 664.

[5] Y. Hong, B. Peng, H. Xiao, L. Liu, and J. Zhang, "Headnerf: A real-time nerf-based parametric head model," in *IEEE CVPR*, 2022, pp. 20 374–20 384.

[6] Y. Zheng, V. F. Abrevaya, M. C. Bühler, X. Chen, M. J. Black, and O. Hilliges, "Im avatar: Implicit morphable head avatars from videos," in *IEEE CVPR*, 2022, pp. 13 545–13 555.

[7] 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," *Communications of the ACM*, vol. 65, no. 1, pp. 99–106, 2021.

[8] A. Pumarola, E. Corona, G. Pons-Moll, and F. Moreno-Noguer, "D-nerf: Neural radiance fields for dynamic scenes," in *IEEE CVPR*, 2021, pp. 10 318–10 327.

[9] K. Park, U. Sinha, J. T. Barron, S. Bouaziz, D. B. Goldman, S. M. Seitz, and R. Martin-Brualla, "Nerfies: Deformable neural radiance fields," in *IEEE CVPR*, 2021, pp. 5865–5874.

[10] X. Gao, C. Zhong, J. Xiang, Y. Hong, Y. Guo, and J. Zhang, "Reconstructing personalized semantic facial nerf models from monocular video," *ACM TOG*, vol. 41, no. 6, pp. 1–12, 2022.

[11] Y. Xu, L. Wang, X. Zhao, H. Zhang, and Y. Liu, "Avatarmav: Fast 3d head avatar reconstruction using motion-aware neural voxels," in *ACM SIGGRAPH 2023 Conference Proceedings*, 2023.

[12] X. Zhao, L. Wang, J. Sun, H. Zhang, J. Suo, and Y. Liu, "Havatar: High-fidelity head avatar via facial model conditioned neural radiance field," *ACM TOG*, 2023.

[13] 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 *ACM SIGGRAPH 2023 Conference Proceedings*, 2023.

[14] Z. Zheng, X. Zhao, H. Zhang, B. Liu, and Y. Liu, "Avatarrex: Real-time expressive full-body avatars," *ACM TOG*, vol. 42, no. 4, 2023.

[15] 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 *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2024.

[16] Y. Xu, H. Zhang, L. Wang, X. Zhao, H. Han, Q. Guojun, and Y. Liu, "Latentavatar: Learning latent expression code for expressive neural head avatar," in *ACM SIGGRAPH 2023 Conference Proceedings*, 2023.

[17] W. Zielonka, T. Bolkart, and J. Thies, "Instant volumetric head avatars," in *IEEE CVPR*, 2023, pp. 4574–4584.

[18] S. Bharadwaj, Y. Zheng, O. Hilliges, M. J. Black, and V. Fernandez-Abrevaya, "Flare: Fast learning of animatable and relightable mesh avatars," *ACM Transactions on Graphics*, vol. 42, no. 6, 2023.

[19] D. Gao, G. Chen, Y. Dong, P. Peers, K. Xu, and X. Tong, "Deferred neural lighting: free-viewpoint relighting from unstructured photographs," *ACM TOG*, vol. 39, no. 6, pp. 1–15, 2020.

[20] X. Zhang, S. Fanello, Y.-T. Tsai, T. Sun, T. Xue, R. Pandey, S. Orts-Escolano, P. Davidson, C. Rhemann, P. Debevec *et al.*, "Neural light transport for relighting and view synthesis," *ACM TOG*, vol. 40, no. 1, pp. 1–17, 2021.

[21] K. Sarkar, M. C. Bühler, G. Li, D. Wang, D. Vicini, J. Riviere, Y. Zhang, S. Orts-Escolano, P. Gotardo, T. Beeler *et al.*, "Litnerf: Intrinsic radiance decomposition for high-quality view synthesis and relighting of faces," in *SIGGRAPH Asia 2023 Conference Papers*, 2023, pp. 1–11.

[22] Y. Xu, G. Zoss, P. Chandran, M. Gross, D. Bradley, and P. Gotardo, "Renerf: Relightable neural radiance fields with nearfield lighting," in *IEEE CVPR*, 2023, pp. 22 581–22 591.

[23] Z. Xu, K. Sunkavalli, S. Hadap, and R. Ramamoorthi, "Deep image-based relighting from optimal sparse samples," *ACM TOG*, vol. 37, no. 4, pp. 1–13, 2018.

[24] A. Meka, R. Pandey, C. Häne, S. Orts-Escolano, P. Barnum, P. David-Son, D. Erickson, Y. Zhang, J. Taylor, S. Bouaziz, C. Legendre, W.-C. Ma, R. Overbeck, T. Beeler, P. Debevec, S. Izadi, C. Theobalt, C. Rhemann, and S. Fanello, "Deep relightable textures: volumetric performance capture with neural rendering," *ACM TOG*, vol. 39, no. 6, nov 2020.

[25] A. Meka, C. Haene, R. Pandey, M. Zollhöfer, S. Fanello, G. Fyffe, A. Kowdle, X. Yu, J. Busch, J. Dourgarian *et al.*, "Deep reflectance fields: high-quality facial reflectance field inference from color gradient illumination," *ACM TOG*, vol. 38, no. 4, pp. 1–12, 2019.

[26] M. Boss, R. Braun, V. Jampani, J. T. Barron, C. Liu, and H. Lensch, "Nerd: Neural reflectance decomposition from image collections," in *IEEE CVPR*, 2021, pp. 12 684–12 694.

[27] P. P. Srinivasan, B. Deng, X. Zhang, M. Tancik, B. Mildenhall, and J. T. Barron, "Nerv: Neural reflectance and visibility fields for relighting and view synthesis," in *IEEE CVPR*, 2021, pp. 7495–7504.

[28] X. Zhang, P. P. Srinivasan, B. Deng, P. Debevec, W. T. Freeman, and J. T. Barron, "Nerfactor: Neural factorization of shape and reflectance under an unknown illumination," *ACM TOG*, vol. 40, no. 6, pp. 1–18, 2021.

[29] Y. Zhang, J. Sun, X. He, H. Fu, R. Jia, and X. Zhou, "Modeling indirect illumination for inverse rendering," in *IEEE CVPR*, 2022, pp. 18 643–18 652.

[30] 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," in *IEEE CVPR*, 2024.

[31] Y. Zheng, W. Yifan, G. Wetzstein, M. J. Black, and O. Hilliges, "Pointavatar: Deformable point-based head avatars from videos," in *IEEE (CVPR)*, 2023.

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

[33] S. Qian, T. Kirschstein, L. Schoneveld, D. Davoli, S. Giebenhain, and M. Nießner, "Gaussianavatars: Photorealistic head avatars with rigged 3d gaussians," in *IEEE CVPR*, 2024.

[34] 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 *IEEE CVPR*, 2024.

[35] J. Xiang, X. Gao, Y. Guo, and J. Zhang, "Flashavatar: High-fidelity head avatar with efficient gaussian embedding," in *IEEE CVPR*, 2024.

[36] Y. Xu, B. Chen, Z. Li, H. Zhang, L. Wang, Z. Zheng, and Y. Liu, "Gaussian head avatar: Ultra high-fidelity head avatar via dynamic gaussians," in *IEEE CVPR*, 2024.

[37] Z. Shao, Z. Wang, Z. Li, D. Wang, X. Lin, Y. Zhang, M. Fan, and Z. Wang, "SplattingAvatar: Realistic Real-Time Human Avatars with Mesh-Embedded Gaussian Splatting," in *IEEE CVPR*, June 2024, pp. 1606–1616.

[38] S. Saito, G. Schwartz, T. Simon, J. Li, and G. Nam, "Relightable gaussian codec avatars," in *IEEE CVPR*, 2024.

[39] V. Blanz and T. Vetter, "A morphable model for the synthesis of 3d faces," in *CGI*, 1999, pp. 187–194.

[40] S. Ploumpis, H. Wang, N. Pears, W. A. Smith, and S. Zafeiriou, "Combining 3d morphable models: A large scale face-and-head model," in *IEEE CVPR*, 2019, pp. 10 934–10 943.

[41] T. Li, T. Bolkart, M. J. Black, H. Li, and J. Romero, "Learning a model of facial shape and expression from 4d scans." *ACM TOG*, vol. 36, no. 6, pp. 194–1, 2017.

[42] R. Daněček, M. J. Black, and T. Bolkart, "Emoca: Emotion driven monocular face capture and animation," in *IEEE CVPR*, 2022, pp. 20 311–20 322.

[43] Z. Qiu, Y. Li, D. He, Q. Zhang, L. Zhang, Y. Zhang, J. Wang, L. Xu, X. Wang, Y. Zhang *et al.*, "Sculptor: Skeleton-consistent face creation using a learned parametric generator," *ACM TOG*, 2022.

[44] A. Lattas, S. Moschoglou, B. Gecer, S. Ploumpis, V. Triantafyllou, A. Ghosh, and S. Zafeiriou, "Avatarme: Realistically renderable 3d facial reconstruction" in-the-wild"," in *IEEE CVPR*, 2020, pp. 760–769.

[45] J. Shang, T. Shen, S. Li, L. Zhou, M. Zhen, T. Fang, and L. Quan, "Self-supervised monocular 3d face reconstruction by occlusion-aware multi-view geometry consistency," in *ECCV*, 2020, pp. 53–70.

[46] A. Tewari, F. Bernard, P. Garrido, G. Bharaj, M. Elgharib, H.-P. Seidel, P. Pérez, M. Zollhofer, and C. Theobalt, "Fml: Face model learning from videos," in *IEEE CVPR*, 2019, pp. 10 812–10 822.

[47] H. Kim, P. Garrido, A. Tewari, W. Xu, J. Thies, M. Niessner, P. Pérez, C. Richardt, M. Zollhöfer, and C. Theobalt, "Deep video portraits," *ACM TOG*, vol. 37, no. 4, pp. 1–14, 2018.

[48] X. Ji, H. Zhou, K. Wang, Q. Wu, W. Wu, F. Xu, and X. Cao, "Eamm: One-shot emotional talking face via audio-based emotion-aware motion model," in *SIGGRAPH Asia*, 2022.

[49] J. Thies, M. Zollhofer, M. Stamminger, C. Theobalt, and M. Nießner, "Face2face: Real-time face capture and reenactment of rgb videos," in *IEEE CVPR*, 2016, pp. 2387–2395.

[50] J. Thies, M. Elgharib, A. Tewari, C. Theobalt, and M. Nießner, "Neural voice puppetry: Audio-driven facial reenactment," in *ECCV*, 2020, pp. 716–731.

[51] R. Shao, H. Zhang, H. Zhang, M. Chen, Y.-P. Cao, T. Yu, and Y. Liu, "Doublefield: Bridging the neural surface and radiance fields for high-fidelity human reconstruction and rendering," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2022, pp. 15 872–15 882.

[52] J. Sun, X. Wang, L. Wang, X. Li, Y. Zhang, H. Zhang, and Y. Liu, "Next3d: Generative neural texture rasterization for 3d-aware head avatars," in *CVPR*, 2023.

[53] Y. Wu, S. Xu, J. Xiang, F. Wei, Q. Chen, J. Yang, and X. Tong, "Aniportraitgan: Animatable 3d portrait generation from 2d image collections," in *SIGGRAPH Asia 2023 Conference Papers*, 2023, pp. 1–9.

[54] J. Munkberg, J. Hasselgren, T. Shen, J. Gao, W. Chen, A. Evans, T. Müller, and S. Fidler, "Extracting triangular 3d models, materials, and lighting from images," in *IEEE CVPR*, 2022, pp. 8280–8290.

[55] H. Kim, M. Jang, W. Yoon, J. Lee, D. Na, and S. Woo, "Switchlight: Co-design of physics-driven architecture and pre-training framework for human portrait relighting," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2024, pp. 25 096–25 106.

[56] A. Dib, C. Thebault, J. Ahn, P.-H. Gosselin, C. Theobalt, and L. Chevallier, "Towards high fidelity monocular face reconstruction with rich reflectance using self-supervised learning and ray tracing," in *IEEE CVPR*, 2021, pp. 12 819–12 829.

[57] H. Feng, T. Bolkart, J. Tesch, M. J. Black, and V. Abrevaya, "Towards racially unbiased skin tone estimation via scene disambiguation," in *ECCV*. Springer, 2022, pp. 72–90.

[58] W. Zielonka, T. Bolkart, and J. Thies, "Towards metrical reconstruction of human faces," in *ECCV*. Springer, 2022, pp. 250–269.
[59] J. T. Kajiya, "The rendering equation," in *Proceedings of the 13th annual conference on Computer graphics and interactive techniques*, 1986, pp. 143–150.
[60] Y. Yao, J. Zhang, J. Liu, Y. Qu, T. Fang, D. McKinnon, Y. Tsin, and L. Quan, "Neilf: Neural incident light field for physically-based material estimation," in *ECCV*. Springer, 2022, pp. 700–716.
[61] Z. Peng, T. Shao, Y. Liu, J. Zhou, Y. Yang, J. Wang, and K. Zhou, "Rtg-slam: Real-time 3d reconstruction at scale using gaussian splatting," in *ACM SIGGRAPH 2024 Conference Proceedings*, 2024.
[62] D. Li, S.-S. Huang, Z. Lu, X. Duan, and H. Huang, "St-4dgs: Spatial-temporally consistent 4d gaussian splatting for efficient dynamic scene rendering," in *ACM SIGGRAPH 2024 Conference Proceedings*, 2024.
[63] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *IEEE CVPR*, 2018, pp. 586–595.
[64] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.
[65] 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.
[66] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo *et al.*, "Segment anything," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 4015–4026.