2DGS-Avatar: Geometrically Accurate Animatable Human Avatar with 2D Gaussian Surfels

Sihan Chen
ETH Zürich
sihchen@student.ethz.ch

Yiming Wang
ETH Zürich
wangyim@student.ethz.ch

Zehong Qiu
ETH Zürich
qiuzeh@student.ethz.ch

Yutong Chen
ETH Zürich
yutong.chen@inf.ethz.ch

Abstract

Learning animatable human avatars from sparse RGB video observations is crucial for numerous applications. Though recent Advances in 3DGS-Avatar have demonstrated impressive real-time rendering and high-quality novel view synthesis capabilities, they struggle to generate smooth and detailed surface reconstruction due to the intrinsic limitation of 3D gaussian splatting (3DGS). We propose 2DGS-Avatar, a new method to reconstruct animatable human avatars with high-fidelity geometry and appearance. Our approach combines 2D Gaussian surfels representation with deformable clothed human avatar modeling to achieve geometrically accurate surface extraction. Additionally, we enhance the animation results of our 2DGS-Avatar by introducing Lipschitz MLP and 6D rotation for improved extrapolation ability. We evaluate both qualitative and quantitative results of the reconstruction and animation, demonstrating that our method is on par with state-of-the-art 3DGS avatar methods in terms of novel view synthesis while achieving significantly better geometry reconstruction. Code are at `https://gitlab.inf.ethz.ch/wangyim/3dgs_avatar_dhproject.git`

1. Introduction

Reconstructing human avatars with high-fidelity geometry and appearance from sparse or monocular-view videos, which can be used to synthesis novel view images and conducting editing under user control, have significant applications, including VR/AR, video games, and telepresence. Traditional methods [3, 8] of human fidelity catpure rely on dense multi-view capture system, which are epensive and hard to setup. It's challenging to extend these methods to in-the-wild sceneraios with sparse or monocular view video inputs.

With recent advances in neural human representation based on the neural radiance fields (NeRF) [19] or 3D Gaussian Splatting (3DGS) [13], methods like 3DGS-Avatar [25] showcase fancy performance in efficient modeling of clothed human avatar, being able to achieve fast training and real-time rendering speed from sparse-view video inputs. Utilizing the human mesh template SMPL [16], these avatar methods can generate novel view synthesis in new pose under user control. However, these methods [25, 29] predominantly focus on appearance modelling, e.g. novel view synthesis, and ignore the reconstruction of human geometry. Additionally, these methods usually suffer from a dramatic performance drop for new pose synthesis. The generalization abiilty to unseen poses of the reconstructed avatar is not well evalauted with quanlitative metrics.

To enhance the quality of geometry reconstruction and improve generalization to unseen poses, we propose 2DGS-Avatar, a new method that combines gaussian surfels representation [10] and deformable clothed human avatar modelling. 2DGS/gaussian surfels flatten the original 3D gassian ellipsoid into a 2D gassian ellipse by setting the 'z' axis in the scale matrix to zero, and thus has a strict normal definition along this direction. Benefiting from this geometric accurate surface modelling of gaussian surfels, our method is able to extract smooth and detailed geometry for human avatar, which all previous 3DGS-Avavtar not able to achieve. Following 3DGS-avatar [25], we use the combined LBS deformation and non-rigid deformation to model the complex dynamics of human avatars and only create 2D-GS in the canonical spaces. The compressed dimension of 2d surf compared with 3d gs also helps us to reduce the learning difficulity for the deformation module and improve its generalizability to unseen pose. To further improve the generalizabiilty to unseen pose, we use Lipshchtiz bound and

6D rotation representation to enhance the non-rigid module with better extrapolation ability.

Our method is capable of generating real-time novel view synthesis results and high-fidelity geometry reconstruction after effcienct training from monocular or sparse-view video inputs. Experiment results demonstrate that our method is on par with the state-of-the-art 3DGS avatar methods in terms of novel view synthesis while achiving much better geometry reconstruction. We also conduct sufficient ablation studies of our proposed modules to show the effects.

Our contribution can be sumarized as follows:

- We introduce 2DGS-Avatar, a new method that combines gaussian surfels representation and deformable clothed human avatar modelling, enhanced with 6D rotation and lipschitz MLP for better unseen pose extrapolation ability.
- Our method generate real-time novel view synthesis results and high-fidelity geometry reconstruction after effcienct training from monocular or sparse-view video inputs.
- Experiment results demonstrate that our method is on par with the state-of-the-art 3DGS avatar methods in terms of novel view synthesis while achiving much better geometry reconstruction.

2. Related Work

2.1. Human Performance Capture

High fidelity reconstruction of dynamic clothed human has been widely explored, while many of them require additional input besides RGB images, e.g. pre-scanned templates [3, 8, 30] or depth sensors. Some methods [6, 21] attempt to utilize depth information from depth sensors to overcome the reliance on scanned templates. Although these methods produce impressive results, they necessitate a specialized capturing setup and are therefore not suitable for in-the-wild applications.

2.2. Deformable Neural Human Representation

The development of neural rendering techniques [13, 19] has enabled the use of neural rendering to reduce the requirement for input data while achieving impressive novel view synthesis results [25, 29]. Notably, 3D Gaussian Splatting facilitates fast training and real-time rendering with high-quality details. Many methods [11, 25] have capitalized on these advancements to enhance their performance. Beyong novel view synthesis, some methods [7] succeed in extracting detailed human geometry reconstruction, but are slow in trainig and rendering. Our work builds on these advancements by introducing 2D Gaussian Surfels [10], which address the limitations of existing methods in both term of geometry and rendering.

3. Method

3.1. Preliminaries

2D Gaussian Splatting. This work extends 3D Gaussian Splatting (3DGS) [13], which demonstrates impressive ability of real-time new view synthesis, to have more accurate geometry representation. 3DGS proposes to represent 3D scenes with 3D Gaussian primitives and render images using differentiable volume splatting, where each primitive (also known as Gaussian blob) is defined by a 3D covariance matrix Σ and its location $\mathbf{p}_k$:

$$G(\mathbf{p}) = \exp(-\frac{1}{2}(\mathbf{p}-\mathbf{p}_k)^\top \Sigma^{-1}(\mathbf{p}-\mathbf{p}_k)), \quad (1)$$

where the covariance maatrix $\mathbf{\Sigma} = \mathbf{R}\mathbf{S}\mathbf{S}^\top\mathbf{R}^\top$ is factorized into a scaling matrix $\mathbf{S}$ and a rotation matrix $\mathbf{R}$. When rendering, each primitive is transformed first by the world-to-camera transform matrix $\mathbf{W}$ and then projected into the image plane via a local affine transformation $\mathbf{J}$ [34], resulting in a 2D covariance matrix:

$$\Sigma^{2D} = (\mathbf{J}\mathbf{W}\mathbf{\Sigma}\mathbf{W}^\top\mathbf{J}^\top)_{1:2,1:2}, \quad (2)$$

where the operation $(-)_{1:2,1:2}$ means skipping the third row and column. Finally, all 2D primitives are accumulated from front to back by alpha blending to obtain the appearance:

$$\mathbf{c}(\mathbf{x}) = \sum_{k=1}^{K} \mathbf{c}_k \alpha_k G_k^{2D}(\mathbf{x}) \prod_{j=1}^{k-1}(1-\alpha_j G_j^{2D}(\mathbf{x})), \quad (3)$$

where k is the index of the Gaussian primitives, α_k denotes the alpha values and $\mathbf{c}_k$ is the view-dependent appearance.

Even though 3DGS has demonstrated promising progress in different domains including material modeling [12, 27], Huang et al. notices that it falls short in capturing intricate geometry since the volumetric 3D Gaussian, which models the complete angular radiance, conflicts with the thin nature of surfaces [10].

To incorporate the inductive bias of thin surfaces, Huang et al. [10] propose to flatten the Gaussian primitives into 2D. Specifically, each 2D Gaussian primitive is defined by its central point $\mathbf{p}_k \in \mathbb{R}^3$, a scaling vector $\mathbf{S} = (s_u, s_v) \in \mathbb{R}^2$ and two orthonormal tangent vectors $\mathbf{t}_u, \mathbf{t}_v \in \mathbb{R}^3$. The orientation of the 2D Gaussian splat can be defined by the rotation matrix $\mathbf{R} = [\mathbf{t}_u, \mathbf{t}_v, \mathbf{t}_w] \in \mathbb{R}^3$, where $\mathbf{t}_w = \mathbf{t}_u \times \mathbf{t}_v$ is the primitive normal. Each 2D Gaussian splat is now embedded in a local tangent plane parameterized by

$$P(u, v) = \mathbf{p}_k + s_u\mathbf{t}_u u + s_v\mathbf{t}_v v = \mathbf{H}(u, v, 1, 1)^\top, \quad (4)$$

where

$$\mathbf{H} = \begin{bmatrix} s_u \mathbf{t}_u & s_v \mathbf{t}_v & \mathbf{0} & \mathbf{p}_k \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{RS} & \mathbf{p}_k \\ \mathbf{0} & 1 \end{bmatrix} \tag{5}$$

For a 2D point in the local uv space, its 2D Gaussian value can be evaluated by standard Gaussian $G(\mathbf{u}) = \exp\left(-\frac{u^2+v^2}{2}\right)$.

Following 3DGS [13], each 2D Gaussian primitive also has learnable opacity α and view-dependent appearance c parameterized with spherical harmonics.

Given the world to screen space transformation $\mathbf{W} \in \mathbb{R}^{4\times4}$, the screen space points are obtained by

$$\mathbf{x} = (xz, yz, z, z)^\top = \mathbf{W}P(u,v) = \mathbf{WH}(u,v,1,1)^\top, \tag{6}$$

where $\mathbf{x}$ represents a homogeneous ray emitted from the camera and passing through pixel (x, y) and intersecting the splat at depth z.

The ray-splat intersection can be computed by parameterizing the ray of a pixel as the intersection of the x-plane $\mathbf{h}_x = (-1, 0, 0, x)$ and the y-plane $\mathbf{h}_y = (0, -1, 0, y)$. Using equation (6), we have

$$\mathbf{h}_u = (\mathbf{WH})^\top \mathbf{h}_x \quad \mathbf{h}_v = (\mathbf{WH})^\top \mathbf{h}_y \tag{7}$$

The intersection point should fall on the transformed x-plane and y-plane [10], hence

$$\mathbf{h}_u \cdot (u,v,1,1)^\top = \mathbf{h}_v \cdot (u,v,1,1)^\top = 0, \tag{8}$$

which yields an efficient solution for the intersection point $\mathbf{u}(\mathbf{x})$:

$$u(\mathbf{x}) = \frac{\mathbf{h}_u^2 \mathbf{h}_v^4 - \mathbf{h}_u^4 \mathbf{h}_v^2}{\mathbf{h}_u^1 \mathbf{h}_v^2 - \mathbf{h}_u^2 \mathbf{h}_v^1} \quad v(\mathbf{x}) = \frac{\mathbf{h}_u^4 \mathbf{h}_v^1 - \mathbf{h}_u^1 \mathbf{h}_v^4}{\mathbf{h}_u^1 \mathbf{h}_v^2 - \mathbf{h}_u^2 \mathbf{h}_v^1} \tag{9}$$

To deal with degenerate situations, Huang et al. [10] also propose to apply the object-space low-pass filter introduced in [2]:

$$\hat{G}(\mathbf{x}) = \max\left\{G(\mathbf{u}(\mathbf{x})), G(\frac{\mathbf{x}-\mathbf{c}}{\sigma})\right\} \tag{10}$$

Finally, 2DGS follows a similar rasterization process as in 3DGS [13], the only difference is to replace each $G_k^{2D}(\mathbf{x})$ with $\hat{G}_k(\mathbf{u}(\mathbf{x}))$ in equation (3).

Linear Blend Skinning. Linear Blend Skinning (LBS) [1, 9, 16, 22, 23, 23, 31] aims at transforming all vertices on the mesh given a set of rigid bone transformations $\{\mathbf{B}_b\}_{b=1}^B$. In the context of SMPL body model [16], $B = 24$ and aeach bone transformation is represented by a 4×4 rotation-translation matrix.

Suppose a point $\mathbf{x}$ is associated with a set of skinning weights $\mathbf{w} \in [0,1]^B$ s.t. $\sum_{b=1}^B \mathbf{w}_b = 1$, the transformations are linearly blended to transform $\mathbf{x}$:

$$\mathbf{x}' = \sum_{b=1}^{B} \mathbf{w}_b \mathbf{B}_b \mathbf{x} \tag{11}$$

In the context of human avatar, SMPL model [16] already provides a pretrained set of skinning weights $\{\mathbf{w}_\mathbf{x}\}$. Alternatively, the skinning weight can be modeled by a coordinate-based neural field $\mathbf{w}_\mathbf{x} = f_{\sigma_w}(\mathbf{x})$ [4,5,17,26,28].

3.2. Geometric Accurate 2D Gaussian Avatar

Deformable Avatar with 2D Gaussian Splatting. Inspired by 3DGS-Avatar [25], we propose using 2D Gaussian Splats to model the deformable avatar, given a monocular video with a calibarated camera, fitted SMPL [16] parameters. We first initialize $N = 50k$ points on the SMPL mesh surface as the inital centers of 2D gaussian splats $\{\mathcal{G}_c\}$ in the canonical space. For the set of 2D gaussian splats, we represent each splat with following attributes:position x, scaling factor s, rotation quaternion q, opacity α and a color feature vector f. Then we deform them into observation space and render images with given camera parameters using alpha composition. We also decompose the deformation into a non-rigid part which encodes pose-dependent cloth deformation, and a rigid part using LBS(decribed in 3.1) with the human skeleton.

We illustrate non-rigid deformation first, which can be formulated as:

$$\{\mathcal{G}_d\} = \mathcal{F}_{\theta_{nr}}(\{\mathcal{G}_c\}; \mathcal{Z}_p) \tag{12}$$

where $\{\mathcal{G}_d\}$ represents non-rigidly deformed 2D gaussians. θ_{nr} means the learnable parameters of non-rigid deformation module. z_p is a latent code which encodes SMPL pose and shape parameters (θ, β) using a hierachical pose encoder [18]. The module is a shallow MLP, taking the center of 2D gaussians x_c and latent code z_p as input, outputs the offsets of center$x$, scaling$s$, quaternion$q$, and a feature vector$z$, respectively, which can be formulated as:

$$(\delta\mathbf{x}, \delta\mathbf{s}, \delta\mathbf{q}, \mathbf{z}) = f_{\theta_{nr}}(\mathbf{x}_c; \mathcal{Z}_p) \tag{13}$$

The offsets can describe the deformed 2D gaussians:

$$\mathbf{x}_d = \mathbf{x}_c + \delta\mathbf{x} \tag{14}$$

$$\mathbf{s}_d = \mathbf{s}_c \cdot \exp(\delta\mathbf{s}) \tag{15}$$

$$\mathbf{q}_d = \mathbf{q}_c \cdot [1, \delta q_1, \delta q_2, \delta q_3] \tag{16}$$

where x_d, s_d, and q_d is the non-rigidly deformed position, scaling and quaternion. $\cdot$ operator means quaternion multiplying.

We further transform the non-rigidly deformed 2D gaussians to observation space:

$$\{\mathcal{G}_o\} = \mathcal{F}_{\theta_r}(\{\mathcal{G}_d\}; \{\mathbf{B_b}\}_{b=1}^{B}) \tag{17}$$

where the $\mathbf{B}_b$ is the global rigid transformation for joint b, $\mathcal{F}_{\theta_r}$ is a MLP predicting the skinning weights at the position x_d, we can do the rigid transformation using LBS:

$$\mathbf{T} = \sum_{b=1}^{B} f_{\theta_r}(\mathbf{x}_d)_b \mathbf{B}_b \tag{18}$$

$$\mathbf{x}_o = \mathbf{T}\mathbf{x}_d \tag{19}$$

$$\mathbf{R}_o = \mathbf{T}_{1:3,1:3}\mathbf{R}_d \tag{20}$$

where the x_o, and $\mathbf{R}_o$ are the position and rotation in observation space. $1:3$ operator means choosing the first three dimensions of transformation matrix $\mathbf{T}$.

Finally, we render images in the observation space. Following [25], instead of using spheical harmonics basis and learned coefficients directly. We use a MLP to model the appearance:

$$c = \mathcal{F}_{\theta_c}(\mathbf{f}, \mathbf{z}, \mathcal{Z}_c, \gamma(\hat{\mathbf{d}})) \tag{21}$$

where $\hat{\mathbf{d}}$ is the canonicalized ray direction(rigid transform back into canonical space from observation space ray direction d), $\mathcal{F}_{\theta_c}$ is a shallow MLP, γ and f are the spherical harmonics basis and learned coefficients, z is the pose-dependent feature vector output by non-rigid deformation module, and $\mathcal{Z}_c$ is per-frame latent code.

Geometry Regularization Unlike 3D gaussian splat, 2D gaussian splat is a plane, which could be perfectly aligned with the mesh surface of SMPL, leanding to much smoother geometry. To better make use of the unique property of 2D gaussians, we use several regularization like [10]:

$$\mathcal{L}_d = \sum_{i,j} \omega_i \omega_j |z_i - z_j| \tag{22}$$

where $\omega_i = \alpha_i G_i(\mathbf{u}(\mathbf{x})) \prod_{j=1}^{i-1}(1 - \alpha_j G_j(\mathbf{u}(\mathbf{x})))$ is the alpha composition weight of the $i - th$ intersection point and z_i is the depth of the point. This depth distortion regularization encourages gaussian splats in an local area to have similar depth, which is helpful to make the mesh geometry consistent. Compared to 3D gaussian splats, what we can get additionally is the normal of splat, which is useful when trying to attach the splats on the SMPL mesh surface. We can introduce another geometry regularization:

$$\mathcal{L}_n = \sum_{i} \omega_i (1 - \mathbf{n}_i^\top \mathbf{N}) \tag{23}$$

where n_i is the normal of 2D gaussian splat, N is the estimated normal from nearby depth point $\mathbf{p}$, using gradient:

$$\mathbf{N}(x, y) = \frac{\nabla_x \mathbf{p} \times \nabla_y \mathbf{p}}{|\nabla_x \mathbf{p} \times \nabla_y \mathbf{p}|} \tag{24}$$

where $\nabla_x \mathbf{p}$ and $\nabla_y \mathbf{p}$ represent x and y direction gradient of point $\mathbf{p}$. Besides, because the input is only a monocular video, the sparsity of input will lead to bad generalization of novel views and novel pose. We add as-rigid-as-possible constraint like [25]:

$$\mathcal{L}_{isopos} = \sum_{i=1}^{N} \sum_{j \in \mathcal{N}_k(i)} \left| d(\mathbf{x}_c^{(i)}, \mathbf{x}_c^{(j)}) - d(\mathbf{x}_o^{(i)}, \mathbf{x}_o^{(j)}) \right| \tag{25}$$

$$\mathcal{L}_{isocov} = \sum_{i=1}^{N} \sum_{j \in \mathcal{N}_k(i)} \left| d(\mathbf{\Sigma}_c^{(i)}, \mathbf{\Sigma}_c^{(j)}) - d(\mathbf{\Sigma}_o^{(i)}, \mathbf{\Sigma}_o^{(j)}) \right| \tag{26}$$

where $\mathcal{N}_k$ means the k-nearest neighborhood points. We use L2 loss for distance function $d(\cdot, \cdot)$.

3.3. Pose Generalizable Non-rigid Deformation

The main challenges of generating novel view results for animation lie in that the training pose is rather limited (300 frames). This problem becomes a on out-of-distribution poses which extraplation matters a lot. To improve the extrapolability of the non-rigid module, we first use Lipschitz MLP [15] with Lipschitz bound loss to replace the vanilla MLP $f z) = f_{\theta_{nr}}$. The Lipschitz MLP augments the ith layer $y = \sigma(W_i x + b_i)$ of the vanilla MLP with a trainable Lipschitz bound c_i as

$$y = \sigma\left(\widehat{W}_i x + b_i\right), \quad \widehat{W}_i = \text{norm}(W_i, \text{softplus}(c_i)) \tag{27}$$

where σ is the activation function, and function norm scales the weight matrix W_i to make each the absolute row-sum value of each row is less than or equal to softplus(c_i). The Lipschitz bound loss then is defined as:

$$\mathcal{L}_{\text{Lipschitz}} = \prod_{i=1}^{N} \text{softplus}(c_i) \tag{28}$$

where N the number of MLP layers.

By optimizing the Lipschitz bound loss during training, we gradually add constraints that the pose latent space for $f z) = f_{\theta_{nr}}$ should be Lipshchitz continuous, which lead to better extrapolation capacity and smoother interpolation results as shown in [15].

We also use the 6D rotation representation [33] instead of Quaternion representation used in 3dgs-avatar [**?**] to further enhance the continuity of the pose space. As illustrated in [33], rotation representations are discontinuous in four dimensions, e.g. Quaternion representation. In detail, the 3D rotation is represented using the first two 3 dimension vector $[a_1, a_2]$ as:

$$f_{6d}\left(\begin{bmatrix} a_1 & a_2 \end{bmatrix}\right) = \begin{bmatrix} b_1 & b_2 & b_3 \end{bmatrix} \tag{29}$$

where $b_1 = a_1$, $b_2 = a_2 - (a_1 \cdot a_2)a_1$, $b_3 = b_1 \times b_2$.

4. Experiments

4.1. Implementation Details

Baseline. We build our model on top of the 3DGS avatar baseline [25], by replacing the 3DGS rasterization with 2DGS rasterization [10].

Dataset. We choose ZJU-MoCap [24] as our dataset for training and evaluation (subject 393). Like what has been done in [25], we evaluate the novel pose image and mesh quality. To conduct both novel pose and novel view evaluation, we split each video ($\sim$ 300 frames) into two halves for training and testing, respectively.

Metrics. The quantitative evaluation is based on the Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index and Learned Perceptual Image Patch Similarity (LPIPS) [32].

Hyperparameters. We use the Adam optimizer ([14]) with learning rate 10^{-3}. The coefficient for Lipschitz MLP loss is $\lambda_{Lipschitz} = 10^{-3}$. The coefficients for geometry loss are $\lambda_{normal} = 0.1$ and $\lambda_{dist} = 1000$. The coefficients for AIAP loss are $\lambda_{isopos} = 1$ and $\lambda_{isocov} = 10^{-3}$. Following Qian et al. [25], we use a multi-level grid [20] to encode 3D positions as spatial features in the non-rigid transformation module with the parameters listed in table 1.

Parameter	Value
Number of levels	16
Feature dimension per level	2
Hash table size	2^{16}
Coarsest resolution	16
Finest resolution	2048

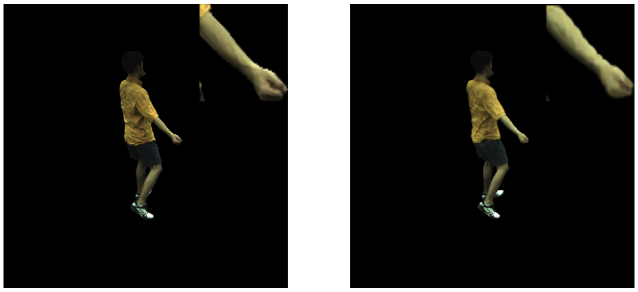
Table 1. Hash table parameters.

4.2. Reconstruction

We first test our model on novel view reconstruction. Table 2 (up and middle rows) shows the metrics comparison between 3DGS [25] baseline and our model. Figure 1 shows the rendering details of a given frame. The similar results correspond with the static scenes of 2DGS [10].

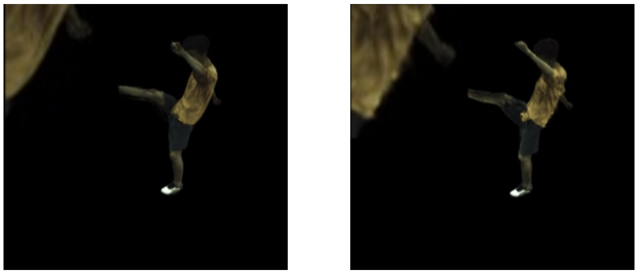
4.3. Pose Generation

We then evalute our model on out-of-distribution pose. Table 2(middle and bottom rows) shows more similar metrics compared to baseline. Due to the properties of 2DGS which are illustrated in Section 3.2, the geometry of our model is much better than baseline. In Figure 2 we can find there are many spikes around the edge of subject in baseline model, but the geometry of our model is much smoother. To further evaluate the geometry, we compare the meshes of our model and baseline in Figure 3. The result of our model gives a compact and relatively smooth mesh, but baseline gives a mesh with many holes and artifacts.


(a) ours

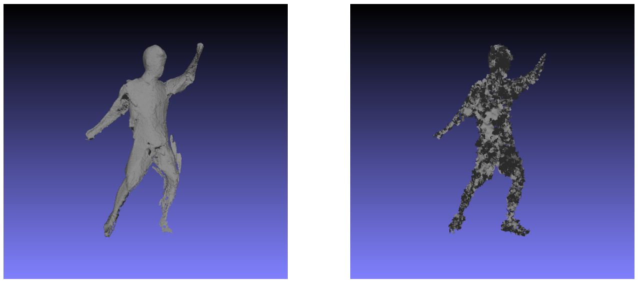
(b) 3DGS

Figure 1. Comparison of our model and 3DGS on novel view reconstruction


(a) ours

(b) 3DGS

Figure 2. Comparison of our model and 3DGS on novel pose rendering


(a) ours

(b) 3DGS

Figure 3. Comparison of our model and 3DGS on novel pose mesh

4.4. ablation study

Due to the lack of resources and time limit, the hyperparamters have not been carefully tuned. Regardless of looking marginal, we can still observe improvement of each module via quantitative or qualitative ablation study.

Table 2. Quantitative results

	3dgs-avatar			**Ours**		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
Training	34.986	0.988	0.0135	33.393	0.983	0.0162
Novel View	27.695	0.957	0.0427	27.401	0.955	0.0451
Novel Pose	27.956	0.960	0.0403	27.925	0.959	0.0416

Table 3. Quantitative results of ablation study.

	PSNR↑	SSIM↑	LPIPS↓
Ours (Full)	27.925	0.959	0.0416
Ours (w/o Pose Encoding)	27.896	0.959	0.0416
Ours (w/o Geometry Loss)	27.951	0.959	0.0425
Ours (w/o AIAP)	27.934	0.959	0.0414

Enhanced Pose Encoding Module. As discussed in Section 3.3, using 6d rotation representation and Lipschitz MLP [15] has the advantage of obtaining smoother network, which is critical for the performance of pose and view extrapolation. Table 3 demonstrates a quantitative improvement on PSNR introduced by this module.

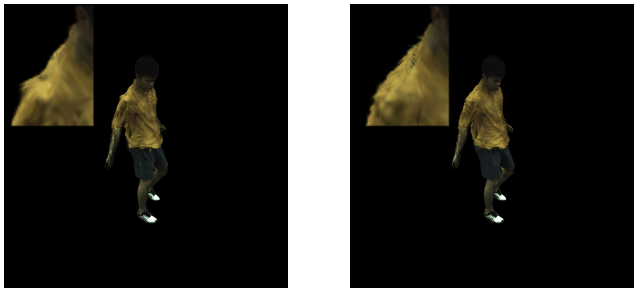

(a) Full model
(b) w/o enhanced pose encoding

Figure 4. Ablation Study on enhanced pose encoding, which removes the artifacts on highly articulated poses.

Figure 4 compares the rendered surfaces given by the models with and without the enhanced pose encoding module and shows that this module is able to reduce the spikes, hence we can conclude that it makes the network smoother as expected.

Geometry Loss. The geometry loss introduced by Huang et al. [10] acts as imposing inductive bias of surface geometry to the Gaussians. Figure 5 shows that this module is helpful at maintaining the continuity of the surface, which makes the new pose predictions look more natural and have less artifacts.

The visualization of surface normals (figure 6) futher proves that the geometry loss imposes surface geometry and continuity on the Gaussians.

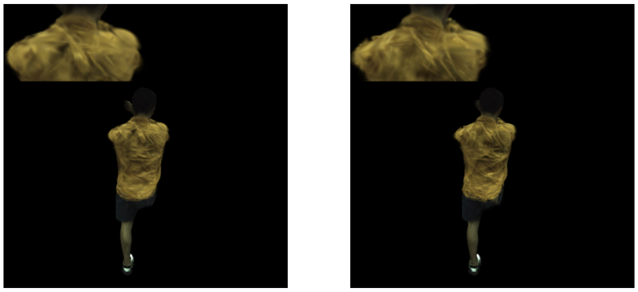

(a) Full model
(b) w/o geometry loss

Figure 5. Ablation Study on geometry loss, which removes the artifacts on highly articulated poses.

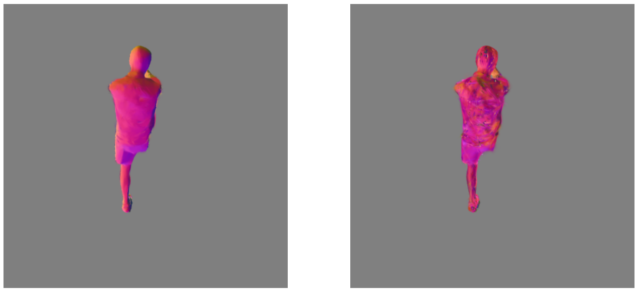
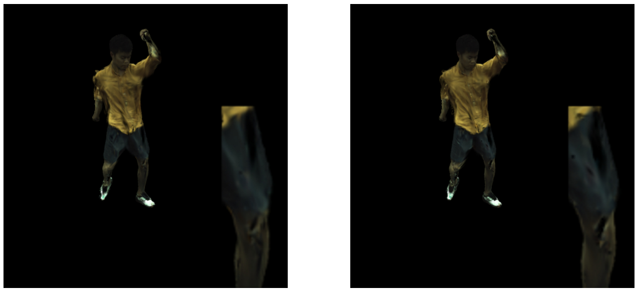

(a) Full model
(b) w/o geometry loss

Figure 6. Ablation Study on geometry loss, visualization of surface normals

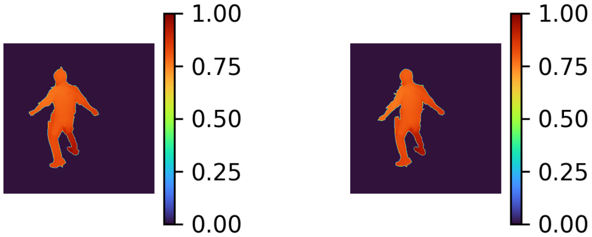
As-isometric-as-possible. The intuition behind AIAP loss is to constrain the Gaussians to comply with consistent movement during deformation, hence improving generalization on novel poses [25]. Even though there is even a slight improvement on metrics after removing the AIAP loss (table 3), we can observe the reduction of artifacts after using AIAP loss by spotting the rendered images (figure 7): large holes become smaller and small holes are fixed. The visualization of depth maps (figure 8) also shows that AIAP loss reduces the bulge on the leg by sticking to constant volume.



(a) Full model

(b) w/o AIAP loss

Figure 7. Ablation Study on as-isometric-as-possible regularization, which removes the artifacts on highly articulated poses.



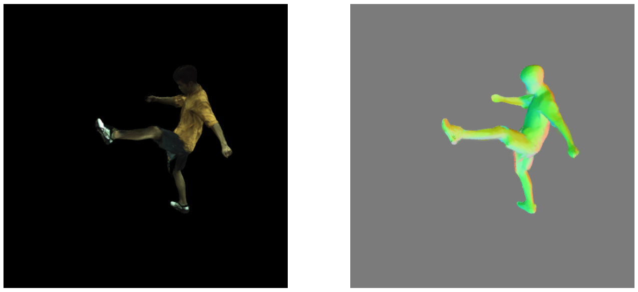
(a) Full model

(b) w/o AIAP loss

Figure 8. Ablation Study on as-isometric-as-possible regularization, depth maps.

4.5. Failure cases

Despite the aforementioned improvements, the model is still limited due to the restricted task setup. In particular, as in figure 9, while the geometry of the mesh, as indicated by the normal map, is satisfactory, the color and texture on the rendered result look messy and blurry, which implies the appearance module lacks far behind the rigid and non-rigid transformation modules.



(a) Rendered result

(b) Normal map

Figure 9. Failure case of the full model, geometry is much better than appearance.

In addition, the rendering of hands is particular challenges to our model (figure 10). The reason is that SMPL model [16], the inductive bias we use for the avatar, doesn't include parameters for hand posing, making non-rigid transformation the only source of hand posing.

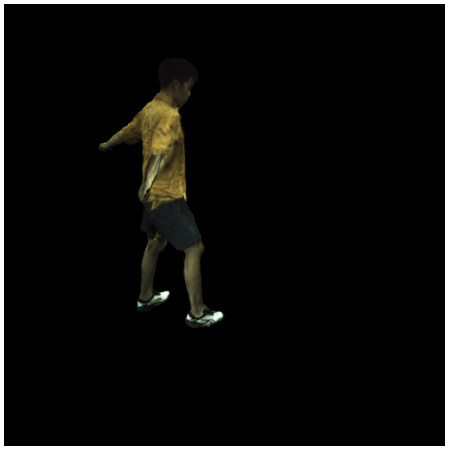


Figure 10. Failure case of the full model on hand posing.

5. Conclusion

In this paper, we propose 2DGS-Avatar, a novel method for reconstructing animatable human avatars with high-fidelity geometry and appearance from sparse or monocular RGB video observations. By using 2D Gaussian surfels representation and deformable clothed human avatar modeling, our approach overcomes the limitations of 3D Gaussian Splatting (3DGS), providing smooth and detailed surface reconstructions. We also enhance the animation capabilities using Lipschitz MLP and 6D rotation, significantly improving extrapolation to unseen poses. Experimental results show that 2DGS-Avatar achieves real-time novel view synthesis and superior geometry reconstruction compared to state-of-the-art 3DGS-Avatar methods.

Although our method improves surface extraction in 3DGS-Avatar, the extracted geometry is far from pefect, still lacking details and may have artifacts in unobserved obdy parts. Additionally, while the proposed modules enhance pose extrapolation, the method is not guaranteed to handle all unseen poses. Further work is needed to improve geometry detail and robustness across a wider range of poses and conditions.

6. Contributions of team members

Equal contributions for the course project in general. In terms of writing the final reports,

- Yiming Wang is responsible for introduction (sec 1), related work (sec 2), method section 3.3, conclusion (sec 5), in addition to a thorough summary of our work in the abstract.

- Zehong Qiu writes the preliminary part (sec 3.1) and experiment setup (sec 4.1), as well as analyzes module ablations (sec 4.4) and failure cases (sec 4.5).
- Sihan Chen takes charge of the methodology explanation (sec 3.2) and conducts comparative study on our model versus 3dgs avatar [25] (sec 4.2 and sec 4.3).

References

[1] Dragomir Anguelov, Praveen Srinivasan, Daphne Koller, Sebastian Thrun, Jim Rodgers, and James Davis. Scape: shape completion and animation of people. In *ACM SIGGRAPH 2005 Papers*, pages 408–416. 2005. 3

[2] Mario Botsch, Alexander Hornung, Matthias Zwicker, and Leif Kobbelt. High-quality surface splatting on today's gpus. In *Proceedings of the Second Eurographics / IEEE VGTC Conference on Point-Based Graphics*, SPBG'05, page 17–24, Goslar, DEU, 2005. Eurographics Association. 3

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

[4] Xu Chen, Tianjian Jiang, Jie Song, Max Rietmann, Andreas Geiger, Michael J Black, and Otmar Hilliges. Fast-snarf: A fast deformer for articulated neural fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023. 3

[5] 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. 3

[6] 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 Trans. Graph.*, 34(4), jul 2015. 2

[7] Chen Guo, Tianjian Jiang, Xu Chen, Jie Song, and Otmar Hilliges. Vid2avatar: 3d avatar reconstruction from videos in the wild via self-supervised scene decomposition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2023. 2

[8] Marc Habermann, Lingjie Liu, Weipeng Xu, Michael Zollhoefer, Gerard Pons-Moll, and Christian Theobalt. Real-time deep dynamic characters. *ACM Trans. Graph.*, 40(4), jul 2021. 1, 2

[9] Nils Hasler, Carsten Stoll, Martin Sunkel, Bodo Rosenhahn, and H-P Seidel. A statistical model of human pose and body shape. In *Computer graphics forum*, volume 28, pages 337–346. Wiley Online Library, 2009. 3

[10] Binbin Huang, Zehao Yu, Anpei Chen, Andreas Geiger, and Shenghua Gao. 2d gaussian splatting for geometrically accurate radiance fields. *arXiv preprint arXiv:2403.17888*, 2024. 1, 2, 3, 4, 5, 6

[11] Tianjian Jiang, Xu Chen, Jie Song, and Otmar Hilliges. Instantavatar: Learning avatars from monocular video in 60 seconds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16922–16932, 2023. 2

[12] Yingwenqi Jiang, Jiadong Tu, Yuan Liu, Xifeng Gao, Xiaoxiao Long, Wenping Wang, and Yuexin Ma. Gaussianshader: 3d gaussian splatting with shading functions for reflective surfaces. *arXiv preprint arXiv:2311.17977*, 2023. 2

[13] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkuehler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4), jul 2023. 1, 2, 3

[14] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 5

[15] Hsueh-Ti Derek Liu, Francis Williams, Alec Jacobson, Sanja Fidler, and Or Litany. Learning smooth neural functions via lipschitz regularization, 2022. 4, 6

[16] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. Smpl: A skinned multi-person linear model. In *Seminal Graphics Papers: Pushing the Boundaries, Volume 2*, pages 851–866. 2023. 1, 3, 7

[17] Mihajlovic Marko, Zhang Yan, and Siyu Tang. Leap: Learning articulated occupancy of people. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10461–10471, 2021. 3

[18] Marko Mihajlovic, Yan Zhang, Michael J Black, and Siyu Tang. Leap: Learning articulated occupancy of people. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10461–10471, 2021. 3

[19] 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. 1, 2

[20] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM transactions on graphics (TOG)*, 41(4):1–15, 2022. 5

[21] Richard A. Newcombe, Dieter Fox, and Steven M. Seitz. Dynamicfusion: Reconstruction and tracking of non-rigid scenes in real-time. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 343–352, 2015. 2

[22] Ahmed AA Osman, Timo Bolkart, and Michael J Black. Star: Sparse trained articulated human body regressor. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VI 16*, pages 598–613. Springer, 2020. 3

[23] Georgios Pavlakos, Luyang Zhu, Xiaowei Zhou, and Kostas Daniilidis. Learning to estimate 3d human pose and shape from a single color image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 459–468, 2018. 3

[24] 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. 5

[25] Zhiyin Qian, Shaofei Wang, Marko Mihajlovic, Andreas Geiger, and Siyu Tang. 3dgs-avatar: Animatable avatars via deformable 3d gaussian splatting. 2024. 1, 2, 3, 4, 5, 6, 8

[26] 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. 3

[27] Yahao Shi, Yanmin Wu, Chenming Wu, Xing Liu, Chen Zhao, Haocheng Feng, Jingtuo Liu, Liangjun Zhang, Jian Zhang, Bin Zhou, et al. Gir: 3d gaussian inverse rendering for relightable scene factorization. *arXiv preprint arXiv:2312.05133*, 2023. 2

[28] Shaofei Wang, Marko Mihajlovic, Qianli Ma, Andreas Geiger, and Siyu Tang. Metaavatar: Learning animatable clothed human models from few depth images. *Advances in Neural Information Processing Systems*, 34:2810–2822, 2021. 3

[29] 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. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16210–16220, June 2022. 1, 2

[30] Donglai Xiang, Fabian Prada, Zhe Cao, Kaiwen Guo, Chenglei Wu, Jessica Hodgins, and Timur Bagautdinov. Drivable avatar clothing: Faithful full-body telepresence with dynamic clothing driven by sparse rgb-d input. In *SIGGRAPH Asia 2023 Conference Papers*, SA '23, New York, NY, USA, 2023. Association for Computing Machinery. 2

[31] Hongyi Xu, Eduard Gabriel Bazavan, Andrei Zanfir, William T Freeman, Rahul Sukthankar, and Cristian Sminchisescu. Ghum & ghuml: Generative 3d human shape and articulated pose models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6184–6193, 2020. 3

[32] 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. 5

[33] Yi Zhou, Connelly Barnes, Lu Jingwan, Yang Jimei, and Li Hao. On the continuity of rotation representations in neural networks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 4

[34] Matthias Zwicker, Hanspeter Pfister, Jeroen Van Baar, and Markus Gross. Ewa volume splatting. In *Proceedings Visualization, 2001. VIS'01.*, pages 29–538. IEEE, 2001. 2