

# TeGA: Texture Space Gaussian Avatars for High-Resolution Dynamic Head Modeling

GENGYAN LI, ETH Zurich, Switzerland and Google, Switzerland

PAULO GOTARDO, Google, Switzerland

TIMO BOLKART, Google, Switzerland

STEPHAN GARBIN, Google, United Kingdom

KRIPASINDHU SARKAR, Google, Switzerland

ABHIMITRA MEKA, Google, United States

ALEXANDROS LATTAS, Google, Switzerland

THABO BEELER, Google, Switzerland

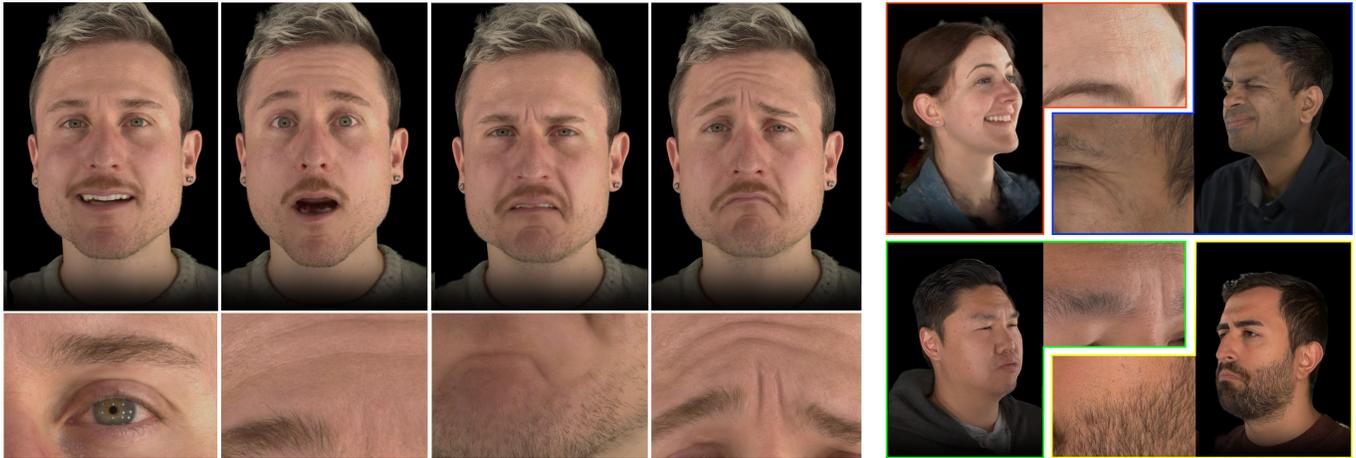


Fig. 1. We propose a novel method that builds photorealistic 3D head avatars for rendering at high resolution with controllable expressions and viewpoints. Given a deformable 3D mesh model, our method effectively learns a sparse, expression-dependent volumetric texture for rendering via 3D Gaussian splatting. It effectively leverages densification where most needed, estimating up to 4 million 3D Gaussians within the continuous UVD tangent space of the mesh. In this figure, we show multiple high fidelity novel face expression reconstructions.

Sparse volumetric reconstruction and rendering via 3D Gaussian splatting have recently enabled animatable 3D head avatars that are rendered under arbitrary viewpoints with impressive photorealism. Today, such photoreal avatars are seen as a key component in emerging applications in telepresence, extended reality, and entertainment. Building a photoreal avatar requires estimating the complex non-rigid motion of different facial components as seen in input video images; due to inaccurate motion estimation, animatable models typically present a loss of fidelity and detail when compared to their non-animatable counterparts, built from an individual facial expression. Also, recent state-of-the-art models are often affected by memory limitations that reduce the number of 3D Gaussians used for modeling, leading to lower detail and quality. To address these problems, we present a new high-detail 3D head avatar model that improves upon the state of the art, largely increasing the number of 3D Gaussians and modeling quality for rendering at 4K resolution. Our high-quality model is reconstructed from multiview input video and builds on top of a mesh-based 3D morphable model, which provides a coarse deformation layer for the head. Photoreal appearance is modelled by 3D Gaussians embedded within the continuous UVD tangent space of this mesh, allowing for more effective densification where most needed. Additionally, these Gaussians are warped by a novel UVD deformation field to capture subtle, localized motion. Our key contribution is the novel

deformable Gaussian encoding and overall fitting procedure that allows our head model to preserve appearance detail, while capturing facial motion and other transient high-frequency features such as skin wrinkling.

CCS Concepts: • **Computing methodologies** → *Point-based models; Animation; Rendering*; Rasterization; Volumetric models.

Additional Key Words and Phrases: Gaussian Splatting, Face Reconstruction, Face Animation, Multiview Stereo reconstruction

## 1 INTRODUCTION

The photorealistic modeling of human faces has enjoyed widespread attention in graphics and computer vision communities, and its applications have been impactful in numerous industries, including telecommunications, media, gaming and medicine to name a few. Telepresence, once science fiction, appears closer and closer, as hardware and algorithms are making great strides. In the pursuit to bridge the uncanny valley, numerous recent works have achieved pleasing high-resolution human animation. However, when considering very high resolution rendering as a requirement, for example, in visual effects in contemporary movies, many challenges arise

in the accurate reconstruction of details, which are paramount to achieve true likeness for a digital human.

A seminal step towards head modeling was the introduction of 3D Morphable Models (3DMM) [Blanz and Vetter 1999], which are still used over twenty years later. Controllable illumination capturing rigs [Debevec 2012] enabled the acquisition of high-quality and relightable facial appearance, though generation and manipulation has remained a challenge. With the introduction of Generative Adversarial Networks (GAN) [Goodfellow et al. 2014], and especially StyleGAN [Karras et al. 2020], the research community witnessed a big leap towards high-quality facial appearance modeling in camera-space [Nagano et al. 2018], UV-space [Gecer et al. 2019], or in 3D [Chan et al. 2022]. In addition, Latent Diffusion Models [Rombach et al. 2022] significantly improved such workstreams. At representation level, Neural Radiance Fields (NeRFs) [Mildenhall et al. 2020] revolutionized the space, providing a flexible medium for holistic human reconstruction and animation [Hong et al. 2022; Park et al. 2021; Sarkar et al. 2023; Zielonka et al. 2022], but were restricted by slow speeds and their difficulty to be manipulated. 3D Gaussian Splatting (3DGS) [Kerbl et al. 2023] recently filled these gaps, with an explicit representation with orders of magnitude faster optimization and unprecedented visual quality.

Multiple works have since explored with increasing success the space of high-quality 3DGS head modeling, following mostly two different paradigms: a) 3DMM-attached Gaussians (e.g. [Giebenhain et al. 2024; Qian et al. 2024]), and b) CNN-based generated Gaussians (e.g. [Saito et al. 2024b; Teotia et al. 2024]). Both approaches still underperform in producing sharp and detailed images when closely inspected. As such, an adequate exploration into the domain of very high resolution rendering is still needed. On one hand, 3DMM-based approaches are limited by the underlying linear model, or require the addition of expensive dense MLPs that learn to deform the Gaussians, and typically remain underconstrained and underperform in challenging cases. On the other hand, CNN-based methods generate a set number of Gaussians on a UV map of limited resolution, and cannot also adaptively allocate details, e.g., for facial hair. As we show in our extensive experiments, both cases perform poorly when attempting to capture fine facial details in high resolution.

Our work proposes a new hybrid approach for rendering digital head avatars at very high resolution (4K). To capture and animate fine facial details, our method effectively combines a 3DMM mesh, adaptive 3D Gaussian densification, and an expression-dependent deformation model that is agnostic to the number of 3D Gaussians. We achieve high-detail, 4K renderings without requiring super-resolution networks that operate on the camera image plane. Even though this is claimed by works similar to ours [Qian et al. 2024; Saito et al. 2024b], our extensive qualitative and quantitative experiments showcase the importance of our improvements, especially around the faithful depiction of facial details. In summary:

- (1) We propose a simple approach for rigging 3D Gaussians within the continuous tangent space of 3DMM face models, allowing Gaussians to move freely across mesh triangles.
- (2) We propose a novel CNN-based deformation model that is agnostic to the number of 3D Gaussians, naturally enabling adaptively densification of the representation to improve

detail where most needed, with expression-dependent shading.

- (3) We show significant improvements over baseline SOTA methods, and demonstrate the ability to render even extreme closeup images at high quality.

## 2 RELATED WORK

### 2.1 Head Modeling

Head modeling has been a very active field of research, arguably materialising with the seminal 3D Morphable Models work [Blanz and Vetter 1999; Egger et al. 2020], a PCA-based model of 3D face shape and appearance. Numerous works have followed, extending the quality and generalization of 3DMMs [Booth et al. 2016; Li et al. 2017; Paysan et al. 2009]. Appearance has been captured with linear models [Smith et al. 2020], GANs [Gecer et al. 2019; Lattas et al. 2020, 2023; Li et al. 2020; Luo et al. 2021], and diffusion models [Papantoniou et al. 2023; Zhang et al. 2023], with increasing quality. Nevertheless, non-skin areas pose a challenge for such models.

NeRFace [Gafni et al. 2021] combined a 3DMM with Neural Radiance Fields [Mildenhall et al. 2020], which provided a more flexible representation capable of also modeling non-skin regions, achieved higher quality appearance modeling. Several follow up works extend this to using a layered NeRF representation [Li et al. 2024b], create fast trainable avatars [Zielonka et al. 2022], or build NeRF-based parametric head models [Hong et al. 2022; Zhuang et al. 2022]. Despite their qualitative success, NeRF-based methods often lack high-frequency details, and suffer from slow evaluation. Our method instead uses 3D Gaussian Splatting to represent the neural, leading in higher quality and rendering performance.

### 2.2 3DGS-based Head Modeling

The recent work of 3D Gaussian Splatting [Kerbl et al. 2023] enabled radiance field training and rendering at much faster rates, and was shown to accurately track and reconstruct human motion [Luiten et al. 2024]. A main paradigm for 3DGS head modeling, introduced by GaussianAvatars [Qian et al. 2024], is to rig the 3D Gaussians on the triangles of 3DMMs, which can be optimized and densified locally, while being animated by 3DMM parameters. NPGA [Giebenhain et al. 2024] further extended this approach with non-linear 3DMMs, adaptive density for occluded facial areas, and learned latent features that enhance the learning of complex facial dynamic behaviour. The attachment of the Gaussians to the mesh may also be learned for additional flexibility [Shao et al. 2024; Zhao et al. 2024]. Moreover, Rig3DGS [Rivero et al. 2024] introduced a prior-based learnable deformation, to achieve the synthesis of scene and the head, while trained only on casually captured videos. 3DMM-based representations have also been extended with MLPs in order to model expression depended effects such as wrinkling [Dhamo et al. 2024; Xu et al. 2024]. Finally, hybrid approaches have also been proposed [Xiao et al. 2024], to leverage the benefits of both 3DGS and textured mesh rendering.

UV texture maps can be used to encode 3DGS parameters in a 2D ordered representation, that can be easily modeled with convolutional networks, while maintaining 3DMM-control [Abdal et al. 2024; Li et al. 2024c; Xiang et al. 2024]. This enables the end-to-end

optimization of person-specific 3DGS head models [Teotia et al. 2024]. Arguably, Relightable Gaussian Codec Avatars [Saito et al. 2024b], demonstrated impressive results when trained on densely captured multi-view data, and enabled gaze and illumination control. They have also been recently extended to a generalizable pipeline [Li et al. 2024a] that can be even optimized on handheld device training data. This generalizable framework can also be extended to training only on in-the-wild facial data, when paired with appropriate regularization [Kirschstein et al. 2024]. To reduce the computational requirements, such convolutional 3DGS avatars can be distilled to smaller PCA models for real-time rendering [Saito et al. 2024a; Zielonka et al. 2024]. Finally, recent works show that generic 3DGS models pre-training on synthetic data enables fast and accurate adaptation to real data [Saunders et al. 2024; Zielonka et al. 2025]. In common to this UV-space methods is their fixed number of Gaussian primitives, determined by the texels in the regressed Gaussian UV map. Without an option for densifying (and pruning) Gaussians during training, resulting avatars often either lack geometric details for smaller UV maps, or are memory intensive for larger maps. Our method instead significantly improves on visual quality with the proposed deformation model which retains the ability to adaptively densify more detailed regions.

### 3 METHOD

#### 3.1 Overview

Given a set of high-resolution, multiview training images, our primary goal is to reconstruct a photorealistic representation of the captured human geometry and appearance, for high-fidelity rendering under controllable expressions and viewpoints. To do so, and inspired by related work in Sec. 2, we leverage a 3DMM to define and track an underlying mesh-based deformation model that serves as control signal for both training and animating our head model.

The 3D Gaussians used for photorealistic rendering are defined relative to this underlying deformable mesh. Rather than anchoring Gaussians to a triangle [Qian et al. 2024], we instead define the 3D Gaussian positions in the continuous UVD space defined by the mesh’s UV texture coordinates and displacement  $D$  along the local surface normal. This representation allows the Gaussians to freely move across different triangles during optimization, which contrasts with the greedy assignment method in [Qian et al. 2024]. Our UVD-space optimization can thus recover from potentially suboptimal triangle assignments during densification, to increase modeling quality where most needed.

However, the coarse nature of the above mesh-based head model most often leads to imprecise deformation estimates and, thus, loss of spatial detail. We therefore complement the model with a new, continuous UVD deformation field and expression-dependent dynamic shading to adjust the 3D Gaussian attributes for representing facial expressions, such as wrinkle deformation with associated darkening through ambient occlusion. As shown in our experimental results (Sec. 5), our overall model with these additional dynamic components can maintain sharp spatial detail (obtained via Gaussian densification) while also modeling skin deformation and transient facial features due to dynamic expressions (*e.g.*, wrinkling). An overview of our method can be found in Fig. 2.

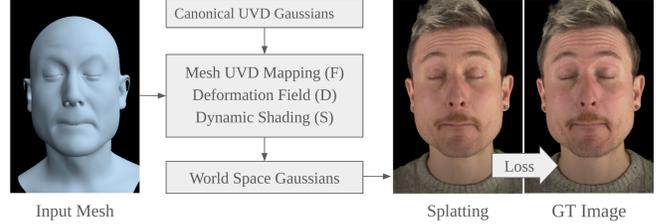


Fig. 2. Our method learns a dense set of canonical 3D Gaussians within the continuous UVD tangent space of a given mesh model, essentially encoding a sparse volumetric texture. An expression-dependent (dynamic) appearance model is also learned to protorealistically render an arbitrary input mesh.

#### 3.2 3D Gaussian Splatting

3D Gaussian Splatting [Kerbl et al. 2023] provides a fast reconstruction and rendering method, based on anisotropic 3D Gaussian primitives. The 3D Gaussians  $G := \{\mu, \mathbf{q}, \mathbf{s}, \mathbf{h}, \alpha\}$  are each parameterised by their location  $\mu \in \mathbb{R}^3$ , rotation quaternion  $\mathbf{q} \in \mathbb{R}^4$ , scaling factor  $\mathbf{s} \in \mathbb{R}^3$ , spherical harmonics parameters  $\mathbf{h} \in \mathbb{R}^{16 \times 3}$ , and opacity  $\alpha \in \mathbb{R}$ . These correspond essentially to a parametric ellipse that is rasterized, with scaling matrix  $S$ , rotation matrix  $R$  and therefore covariance matrix  $\Sigma = RSS^T R^T$ . Given a view direction  $\mathbf{v} \in \mathbb{R}^3$ , The view-dependent RGB color of each Gaussian is computed as  $c(\mathbf{v}) = \mathbf{b}_{SH}(\mathbf{v})\mathbf{h}$ , where  $\mathbf{b}_{SH}(\mathbf{v})$  is the spherical harmonics basis evaluated at  $\mathbf{v}$ . After splatting, the rendered pixel color is then a weighted blend of all the corresponding overlapping Gaussians, multiplied by their opacity.

#### 3.3 3DMM and Canonical UVD-space Gaussians

Akin to other models such as FLAME [Li et al. 2017], our 3DMM defines a subspace of 3D head meshes with varying facial expressions and head poses for a given identity. These meshes  $\mathcal{M}_t$  are parameterized by expression parameters  $\psi_t \in \mathbb{R}^{313}$  and pose parameters  $\theta_t \in \mathbb{R}^{15}$ , the latter capturing overall head translation as well as rotations of head, neck, and the two eyes. Note however that our model is not tied to our particular 3DMM implementation and can use, at training and test time, any sequence of 3D meshes  $\mathcal{M}_t, t \in [1, T]$  that are in correspondence. We demonstrate this fact in Sec. 5 by using head meshes provided with the Multiface dataset [Wuu et al. 2022].

Rather than defining 3D Gaussian positions directly in 3D world space, our model defines *canonical* 3D Gaussians within the continuous UVD tangent space, shared by the subject’s 3D meshes at training and test time. This representation allows us to establish a simple, continuous mapping between a Gaussian’s location in UVD space ( $\mu_{\text{uvd}}$ ) and world space ( $\mu_{\text{xyz}}$ ).

$$\mu_{\text{xyz}} = F(\mu_{\text{uvd}}). \quad (1)$$

Using traditional computer graphics, this function maps a UV coordinate to a 3D point on the corresponding mesh triangle, then uses the displacement coordinate  $D$  to offset the point along its normal (obtained from the triangle’s vertices via barycentric interpolation). As a result, even if the canonical 3D Gaussian remains static in UVD space, it still moves, rotates and stretches in world space together

with the underlying mesh surface. This full local transformation near a UVD point is captured by the Jacobian  $J_F$  of  $F(\cdot)$  at said point. We thus propose to use  $J_F$  to propagate the deformation of the mesh surface to the Gaussian’s covariance  $\Sigma_{\text{uvd}}$  in UVD to obtain the covariance  $\Sigma_{\text{xyz}}$  in world space:

$$\Sigma_{\text{xyz}} = J_F \Sigma_{\text{uvd}} J_F^T. \quad (2)$$

This model generalizes the Gaussian “rigging” by Qian *et al.* [2024], which without explicit formalization does in fact use the Jacobian of their (scaled) rigid transformation. Our Jacobian-based approach is simpler to implement, as it can be computed using automatic differentiation using PyTorch. Furthermore, it also generalizes to any continuous 3D space mapping, without being tied to a particular 3DMM or deformation model. For example, it is able to handle non-rigid triangle deformations such as when facial skin stretches. During optimization, canonical Gaussians are spawned, moved, and their covariance is  $\Sigma$  reshaped within UVD space. Our Gaussian rigging method maintains the full solution space and allows Gaussians to move across triangles during optimization, if that leads to better densification and modeling. In contrast, GaussianAvatars [Qian et al. 2024] adopts a greedy and potentially suboptimal solution in which the binding of each Gaussian to a triangle cannot be undone.

### 3.4 UVD Deformation Field

Continuous deformation fields have often been added to complement static (canonical) volumetric representations, turning the latter into deformable models [Chen et al. 2024; Lu et al. 2024; Park et al. 2021; Wang et al. 2024; Wu et al. 2024; Yang et al. 2024; Yu et al. 2024; Zielonka et al. 2022]. Typically, these fields must learn to output a multidimensional parameter vector for a transformation that includes translation, rotation, and scaling components; such complex deformation fields often remain underconstrained (poorly supervised), and usually lead to poor generalization to unseen facial expressions.

We now revisit this approach and propose using a simple 3D translation field that is easier to train. Leveraging the Jacobian method in Eq. 2, we derive complete transformations from such simpler field, including rotation and stretching components; the shape of each 3D Gaussian thus simply deforms according with the continuous mapping from UVD to world space.

Here, our goal is to complement the coarse mesh-based deformation described above with a residual motion field that helps the overall model to better capture localized and subtle facial deformations, which in turn leads to modeling with a higher level of detail. Leveraging our Jacobian method above, we propose a deformation field  $D(\cdot)$  that, given an input position  $\mu_{\text{uvd}}$  in UVD space, learns to output a residual 3D translation component in world space  $\Delta\mu_{\text{xyz}}$ . The mapping in Eq. 1 then expands to

$$\mu_{\text{xyz}} = F(\mu_{\text{uvd}}) + \Delta\mu_{\text{xyz}}, \quad \text{where} \quad \Delta\mu_{\text{xyz}} = D(\mu_{\text{uvd}}; \mathbf{f}_{\text{uv}}). \quad (3)$$

This residual translation field is conditioned on a learned, local feature vector,  $\mathbf{f}_{\text{uv}}$ , that depends only on the Gaussian’s UV coordinates and the underlying mesh deformation;  $\mathbf{f}_{\text{uv}}$  is further detailed below, with the architecture of our deformation field. With this additional

field, our Jacobian based covariance mapping in Eq. 2 becomes:

$$\Sigma_{\text{xyz}} = (J_F + J_D) \Sigma_{\text{uvd}} (J_F + J_D)^T = J_{(F+D)} \Sigma_{\text{uvd}} J_{(F+D)}^T, \quad (4)$$

where  $J_D$  is the new Jacobian of the deformation field relative to  $\mu_{\text{uvd}}$  only. The resulting Jacobian  $J_{(F+D)} = J_F + J_D$  can then be directly computed via automatic differentiation.

In contrast to the analytical mapping  $F(\cdot)$ , the deformation field  $D(\cdot)$  is learned end-to-end and is comprised of two main components: (i) a convolutional U-Net, operating in the UV texture grid to output the local deformation codes  $\mathbf{f}_{\text{uv}}$ ; and (ii) a shallow MLP network that implements  $D(\cdot)$  in Eq. 3. This architecture is illustrated in Fig. 3. The input to the UV deformation U-Net is a texture of rasterized XYZ vertex coordinates in world space; these coordinates are represented as vertex displacements (offsets) relative to a similar texture obtained for a neutral face, using simple texelwise subtraction. To compensate for rigid head motion, we zero out the pose parameters for these displacements, leaving only the pure expressions. The U-Net is then trained to translate such offsets into a  $N \times N \times 64$  latent texture from which the learned, local deformation codes  $\mathbf{f}_{\text{uv}}$  can be queried with continuous UV coordinates via bilinear interpolation. This model thus allows us to decouple the texture dimension  $N$  from the total number of Gaussians used for modeling (densification). In contrast, other methods [Li et al. 2024a; Teotia et al. 2024] that directly decode a texture of Gaussian attributes are limited to a fixed budget of at most  $N^2$  Gaussians. For our experiments, we use a value of 256 for  $N$ .

The continuous deformation field  $D(\cdot)$  is defined as a small MLP with two hidden layers, taking as input the position of a Gaussian in UVD space,  $\mu_{\text{uvd}}$  (with standard, sinusoidal positional encoding), and the corresponding input latent code  $\mathbf{f}_{\text{uv}}$ . As noted in Bai et al. [2023], conditioning this MLP on such local expression code allows the model to adapt to localized deformations in the input 3D mesh. Additionally, in contrast to related work, our UVD deformation field leverages the Jacobian method above to simplify its output, making the learning task simpler. Finally, as human eyes are rigid, we exclude any Gaussians placed on the eyeballs from being deformed by the deformation field.

### 3.5 Expression-Dependent Dynamic Colors

While the deformation model above can adjust the position and covariance of the 3D Gaussians for a given expression, the view-dependent RGB color of each Gaussian,  $\mathbf{c}(\mathbf{v}) \in \mathbb{R}^3$ , remains as an expression-agnostic “canonical” attribute. To model transient appearance features such as darkening within skin wrinkles, we now derive an additional network component,  $S(\cdot)$ , to provide each Gaussian with an expression-dependent dynamic shading.

To render our expression-dependent Gaussians in world space, we define a dynamic shading model  $\mathbf{c}_e(\mathbf{v}, \mathbf{f}_{\text{uv}})$  that is also a function of the latent facial expression code  $\mathbf{f}_{\text{uv}}$ ,

$$\mathbf{c}_e(\mathbf{v}, \mathbf{f}_{\text{uv}}) = \mathbf{c}(\mathbf{v}) S(\mu_{\text{uvd}}; \mathbf{f}_{\text{uv}}). \quad (5)$$

The canonical color term  $\mathbf{c}(\mathbf{v})$  has the standard Spherical Harmonics encoding  $\mathbf{h}$ , while the dynamic monochromatic shading term  $S(\cdot) \in \mathbb{R}$  is modeled as a small MLP with two hidden layers.

Empirically, we found it beneficial to duplicate the last layer of the deformation U-Net so that a dedicated (split) set of latents  $\mathbf{f}_{\text{uv}}^{(S)}$

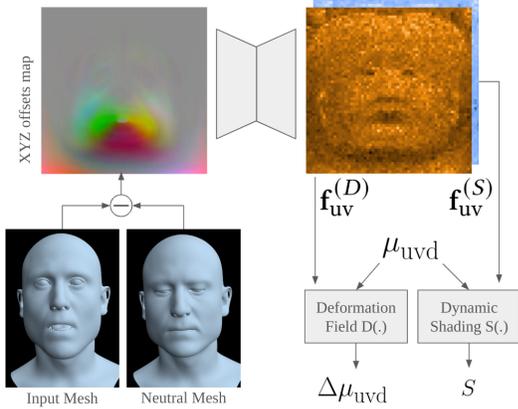


Fig. 3. The UVD deformation field and dynamic shading components are modeled as MLPs that are conditioned on expression-dependent latent features learned on the UV texture grid by a shared U-Net encoder.

is learned for dynamic colors, as shown in Fig. 3. This change allows us to better accommodate different regularization (smoothness) constraints on these two dynamic components. As detailed next, we found it advantageous to have a spatially smooth deformation field, without smoothing the dynamic shading network.

### 3.6 Training Losses and Regularization

Given input video frames captured in multiple views, we assume that a deformable 3D face mesh model (with fixed topology and UV map) has been tracked over these frames. We then proceed to train our new dynamic, photorealistic appearance model by minimizing an overall rendering loss, between captured  $I$  and rendered  $\hat{I}$  images, which is also subject to regularization constraints as described next.

Following related work, we utilize a combination of  $\mathcal{L}_1$ ,  $\mathcal{L}_{D-SSIM}$ , and  $\mathcal{L}_{VGG}$  losses as our primary objective,

$$\mathcal{L} = \lambda_1 \mathcal{L}_1 + \lambda_{D-SSIM} \mathcal{L}_{D-SSIM} + \lambda_{VGG} \mathcal{L}_{VGG} + \lambda_{J_D} \mathcal{L}_{J_D}, \quad (6)$$

with  $\lambda_1 = 0.8$  and  $\lambda_{D-SSIM} = 0.2$ , as originally proposed in [Kerbl et al. 2023]. The perceptual VGG loss is defined as:

$$\mathcal{L}_{VGG} = \|F_{VGG}(I) - F_{VGG}(\hat{I})\| \quad (7)$$

where  $F_{VGG}(\cdot)$  denotes the output feature of the first four layers of a pre-trained VGG network [Zhang et al. 2018]. While this perceptual loss helps reconstruct finer detail exceptionally well, our experiments show that it may also interfere with the optimization of the UVD deformation field during initial iterations. We thus follow a coarse-to-fine training strategy, in which we initially set  $\lambda_{VGG} = 0$ . After 50% of the training is done, when the model can start to focus on learning finer details, we set  $\lambda_{VGG} = 0.01$  and exponentially increase towards a final value of 0.1. This leads to better estimation of non-rigid motion across video frames, and sharper appearance detail as a result.

We also find it beneficial to constrain the UVD deformation field to be initially smoother. We enforce smoothness by penalizing the

spatial derivatives of this field,

$$\mathcal{L}_{J_D} = \sum_{s_{uvd}} \|J_D(s_{uvd})\|_F^2 \quad (8)$$

where  $\|\cdot\|_F$  denotes the Frobenius norm and  $J_D(s_{uvd})$  is the field’s Jacobian evaluated at 100K randomly sampled UVD points  $s_{uvd}$ , with  $UV \in [0, 1]$  and  $D \in [-50, 200]$ mm. The goal here is to have the entire deformation field be spatially smooth, not just at the Gaussian means. We initially set  $\lambda_{smooth} = 1.0$ , which then exponentially decays to 0.1 halfway during training. Also, following Li et al. [2024a], we restrict the size of each Gaussian to within a prescribed range, having standard deviations within  $[5mm, 0.02mm]$ .

We train our method over 60K total iterations on an NVIDIA H100, taking 10 to 16 hours. Our PyTorch implementation uses an Adam optimizer [Kingma and Ba 2014] with learning rates set as in Yang et al. [2024], decaying over the 60K iterations.

### 3.7 Initialization

We initialize the weights of the deformation and shading MLPs with zeros, which leads to a null initial field and uniform dynamic shading at 1.0 (using a sigmoid activation multiplied with 2). To initialize the canonical UVD Gaussians, we uniformly sample 500K random means  $\mu_{uvd}$  and, for the remaining attributes, we follow the procedure in Kerbl et al. [2023]. Because some Gaussian may be initialized outside valid areas of the UV plane, we project these means onto the nearest mesh triangle.

As Yang et al. [2024], we “warm up” the initial canonical 3D Gaussians by first optimizing them without the residual UVD deformation field, and only begin optimizing the field after obtaining an initial canonical volume (after 10K iterations). Unlike Yang et al. [2024], our method can warm up for substantially longer by leveraging the coarse deformation given by the underlying 3DMM mesh.

### 3.8 Adaptive Densification

Unlike many other deformable face avatar approaches [Chen et al. 2024; Li et al. 2024a; Xiang et al. 2024; Xu et al. 2024], we retain the adaptive densification from 3DGS [Kerbl et al. 2023] and their thresholding – we clone small Gaussians, and split large Gaussians with large view-space gradients. As these thresholds are computed using screen-space gradients, the densification algorithm works without any changes. More specifically, although densification thresholds are computed in world-space, splitting and cloning occurs in UVD space, which is itself a continuous 3D space. Finally, we set an upper bound of 4 million Gaussians to avoid running out of memory. In this manner, we can benefit by the 3DMM guidance and UV ordered topology, while being able to reconstruct and track fine facial and hair details, which vary significantly among individuals.

## 4 MULTIVIEW IMAGE DATASETS

We trained and evaluated our new method against baselines on two multiview image datasets: (1) the publicly available Multiface dataset [Wuu et al. 2022], and (2) our own dataset captured in-studio with the facial performance setup described next.

Our multiview, calibrated camera rig comprises 13 26MP cameras with 50mm lenses, evenly distributed over the frontal hemisphere

and tightly framed to capture the subject’s head down to their shoulders. Constant, uniform illumination is provided by 14 small area lights evenly placed in front of the subject. For our experiments, we captured 8 subjects, each performing a set of prescribed facial motions that included facial expressions of emotion, localized muscle activation (action units) and different lip shapes (visemes), for a total of 44 short video clips (at 30fps) per subject. On each video frame, dense multiview stereo [Gu et al. 2020] was run to reconstruct a 3D mesh, followed by registration of a template 3D mesh topology, which was done by tracking our own 3D morphable model (similar to FLAME [Li et al. 2017]) across the input videos. For training a head avatar model for each subject, we sampled 400 frames uniformly from the input video takes and downsampled the image dimensions by half, to  $3072 \times 2048$  pixels (due to GPU memory limitations). Finally, we run a matting algorithm [Pandey et al. 2021] to mask out background pixels.

For Multiface, we selected a subset of 40 cameras (excluding cameras behind the head and directly above). We trained and tested on subjects with IDs 5372021, 6795937, 8870559, 002643814. For each subject, we also uniformly sampled 400 frames for training. Instead of tracking a 3DMM, we use the 3D meshes provided with the dataset for all evaluated methods, including ours. We train at the original resolution of  $2048 \times 1334$  pixels without downsampling.

For both datasets, we consider test sequences where the subject goes through multiple expressions over the course of a few seconds.

## 5 RESULTS

In this section, we demonstrate facial performance reenactment as well as novel view synthesis. For each subject, we evaluate on test frames from three training views, using three cameras distributed horizontally in front of the face. We also evaluate novel view synthesis quality on images rendered on 3 holdout camera views, using training expression. We strongly encourage readers to view the supplementary materials for full resolution videos.

We compare LPIPS, SSIM and PSNR, computed relative to known ground-truth images. To evaluate expression alignment, we use an off-the-shelf landmark detector, and estimate the mean-squared difference between keypoints predicted on the ground truth versus rendered image. To improve consistency, we only use landmarks on the eye, mouth and nose.

### 5.1 Comparisons with the state of the art.

We compare our method against four representative baselines, GaussianAvatars (GA) [Qian et al. 2024], GaussianHeadAvatars (GHA) [Xu et al. 2024], RGCA [Saito et al. 2024b], and MVP [Lombardi et al. 2021] that focus on high-quality modeling from in-studio data. Thus, we do not include baselines that target monocular reconstruction in the wild, with lower modeling quality.

As RGCA is designed to be relightable, we adjust it to be compatible with a static lighting dataset without access to environment lighting information. To do this, we replace their specular and light direction dependent SH formulation with the original view dependent SH formulation. Other than that, we run their code as is, using the same geometry as used with our method. As our multiview dataset does not have accurate projected textures, we instead use

the same vertex delta map used as an input for our U-Net. MVP, using a similar architecture, is trained using the same inputs, but as it models only a static environment, requires no further changes.

GaussianAvatars is run on the same geometry as our method and RGCA. As Multiface provides tracked meshes, rather than a 3DMM, we disable pose optimization and input the mesh directly.

Finally, GHA uses supervision based on proximity to 3D keypoints. Therefore, we used their provided data processing pipeline.

As Fig. 8 shows, RGCA significantly underperforms in expression preservation (alignment). This is likely due to auto-encoding vertex locations: on novel test data, the auto-encoded expressions are reproduced less accurately, especially with more limited training data. Their convolutional decoder is also limited to a fixed budget of Gaussians (about 1 million), being unable to run adaptive densification. As such, their results are blurrier and suffer from “transparency” artifacts in novel views due to the representation not being dense enough (visible when the camera moves, as shown in our supplementary material). MVP, being limited to fixed voxel primitives, has similar issues and overall performs worse than RGCA.

GaussianAvatars performs well at lower resolutions, and is mostly able to reconstruct expressions. However, when zoomed in, their method fails to reconstruct fine details and has numerous artifacts. Despite using adaptive densification, their method typically still uses less than 200K Gaussians. For a fairer comparison, we rerun our method with a similar upper limit, and show that we outperform them even with a similar gaussian budget. Please refer to the several side-by-side videos in our supplementary material.

GHA suffers from noticeable misalignments as well as aliasing artifacts. GHA was notably the only method which could not use alternate geometry, and their custom reconstruction uses a purely keypoint based 3DMM fitting approach, resulting in overall worse fits due to a lack of photometric consistency. Furthermore, we note that both our dataset and the multiface dataset have much crisper images compared to the NeRFsemble dataset used by GHA. As a result, the previously already present aliasing artifacts are significantly more accentuated and noticeable.

Table 1. Quantitative comparison on our dataset; metrics evaluated on the full head. **Best**, **second best**, **third best** scores are highlighted.

	Test Reenactment				Novel View		
	Landmark ↓	LPIPS ↓	SSIM ↑	PSNR ↑	LPIPS ↓	SSIM ↑	PSNR ↑
No Warp	34.0 ± 32.7	0.169 ± 0.045	<b>0.779 ± 0.054</b>	24.3 ± 1.3	0.139 ± 0.042	0.828 ± 0.040	26.1 ± 1.7
No Shading	<b>33.5 ± 31.0</b>	<b>0.155 ± 0.053</b>	0.772 ± 0.060	24.0 ± 1.2	<b>0.128 ± 0.048</b>	0.847 ± 0.037	26.3 ± 1.4
No Network	38.2 ± 33.8	0.180 ± 0.050	0.769 ± 0.058	23.6 ± 1.4	0.150 ± 0.044	0.818 ± 0.043	25.1 ± 1.7
No Densify	122.7 ± 147	0.274 ± 0.072	0.738 ± 0.053	21.3 ± 1.5	0.211 ± 0.061	0.802 ± 0.044	24.9 ± 1.5
MLP	34.4 ± 31.7	0.167 ± 0.049	0.778 ± 0.058	<b>24.5 ± 1.3</b>	0.130 ± 0.049	0.833 ± 0.043	26.1 ± 1.5
No VGG	37.1 ± 34.8	0.178 ± 0.052	<b>0.787 ± 0.032</b>	<b>24.5 ± 1.2</b>	0.148 ± 0.052	<b>0.851 ± 0.037</b>	<b>26.8 ± 1.5</b>
No Triangle Updates	<b>31.7 ± 29.1</b>	<b>0.153 ± 0.016</b>	0.777 ± 0.061	24.3 ± 1.4	<b>0.123 ± 0.048</b>	<b>0.853 ± 0.039</b>	<b>27.0 ± 1.6</b>
GaussianAvatars	71.7 ± 67.7	0.249 ± 0.060	0.773 ± 0.050	24.1 ± 1.5	0.201 ± 0.060	0.813 ± 0.043	25.4 ± 1.6
RGCA	116.9 ± 95.6	0.212 ± 0.055	0.730 ± 0.058	23.0 ± 1.2	0.177 ± 0.055	0.823 ± 0.060	24.4 ± 5.1
GHA	1421 ± 1625	0.229 ± 0.075	0.669 ± 0.064	15.5 ± 2.3	0.180 ± 0.080	0.725 ± 0.073	17.1 ± 2.1
MVP	208.0 ± 122.	0.268 ± 0.071	0.748 ± 0.051	23.1 ± 1.3	0.226 ± 0.080	0.800 ± 0.049	26.0 ± 1.3
Ours	<b>31.7 ± 28.6</b>	<b>0.150 ± 0.055</b>	<b>0.779 ± 0.061</b>	<b>24.4 ± 1.2</b>	<b>0.123 ± 0.045</b>	<b>0.848 ± 0.038</b>	<b>26.7 ± 1.5</b>
Ours (200K GS.)	49.0 ± 26.1	0.207 ± 0.022	0.770 ± 0.029	24.3 ± 1.4	0.163 ± 0.013	0.807 ± 0.024	25.7 ± 1.3

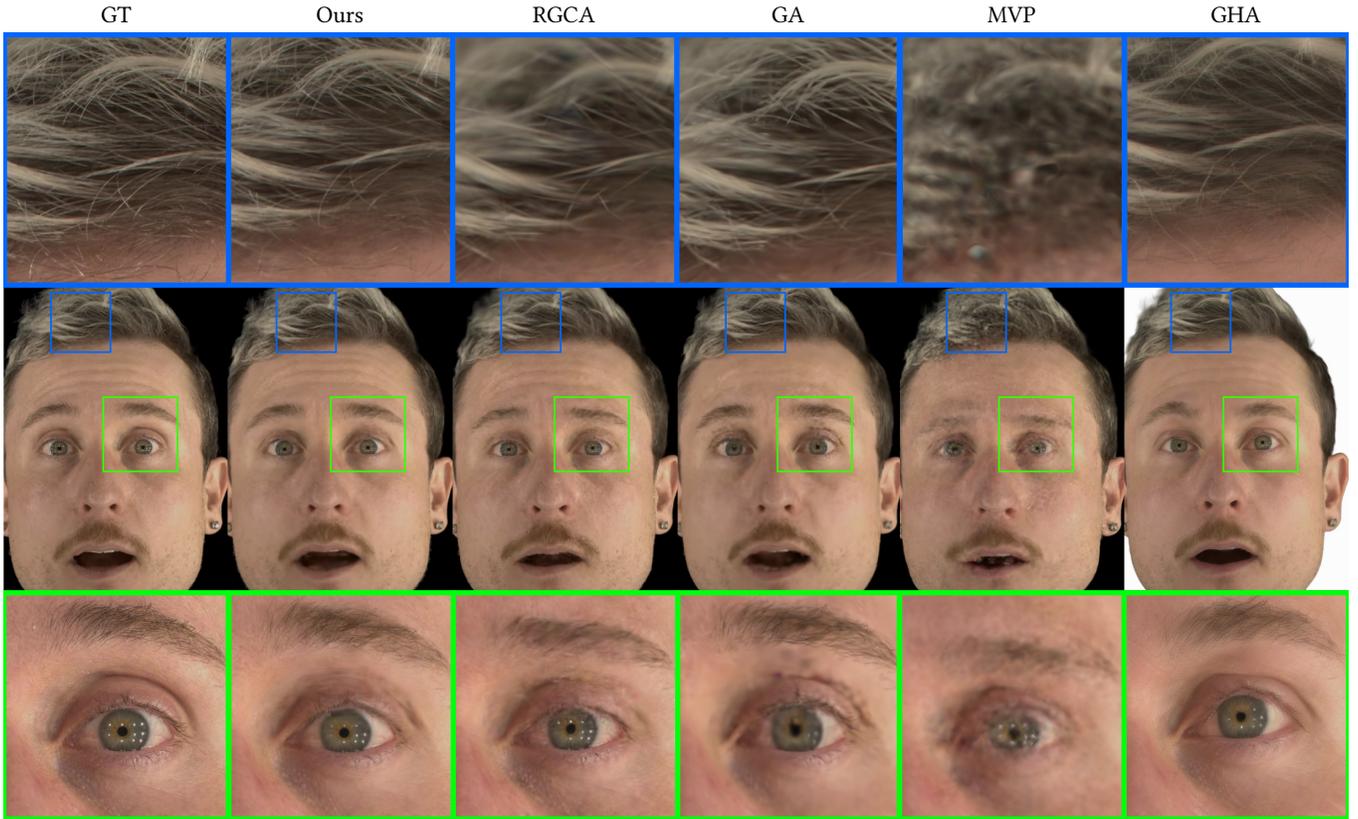


Fig. 4. Our method manages to extrapolate better when compared to RGCA, GA, MVP and GHA. Thanks to the warpfild and densification we can more faithfully capture hair and brows.

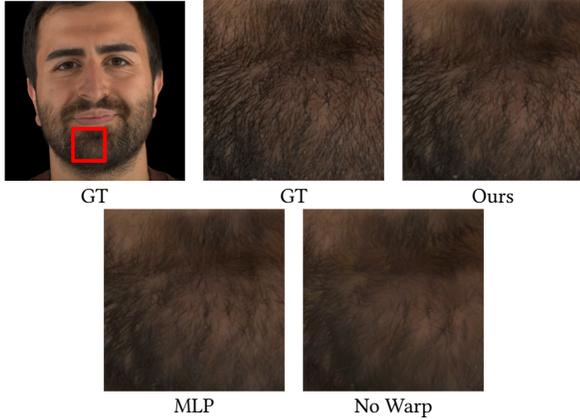


Fig. 5. We compare zoomed in patches of the red highlighted area.

## 5.2 Ablations

Table 1 shows ablations on our various design choices and settings to demonstrate their contribution to the method.

**5.2.1 Warp Field and Shading model.** The deformation and shading nets are inherently ambiguous and capable of compensating each other. We thus also evaluated our method trained without shading (No Shading), without deformation (No Warp), and without both (No Network). As Fig. 7 shows, the shading network is critical to reconstructing wrinkles well. Similarly, without a deformation network, our method is unable to reconstruct finer details nearly as well, Fig. 5. We also ablate our choice of a hybrid CNN/MLP approach, instead of a deep MLP: we replaced our UNET with an 8 layer CNN encoder which produces a single latent code for the entire expression, and use it to drive a positionally encoded MLP, similar to Yang et al. [2024] (MLP). However, due to the comparative expensiveness of a deep MLP, we can only train 1.5 million Gaussians with the same amount of GPU memory. Fig. 5 shows this is insufficient to best capture finer details.

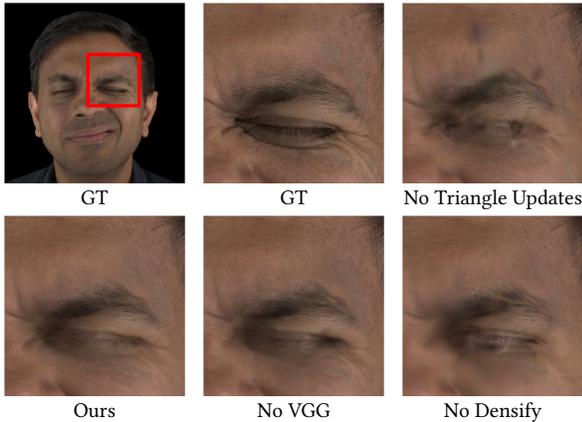


Fig. 6. We compare zoomed in patches of the red highlighted area.



Fig. 7. Close-up view of the highlighted patch, showing that our dynamic shading leads to better modeling of skin wrinkles.

**5.2.2 Triangle Updates.** We show the benefit of allowing Gaussians to move across triangles by comparing with the binding inheritance strategy from Qian et al. [2024] (No triangle updates). Without the ability to switch Gaussians between triangles, artifacts appear in many blinking poses, as seen in Fig. 6.

**5.2.3 Densification.** Critical to our method is the ability to adaptively densify. As seen in Figure 6, without densification (No Densify), our method dramatically drops in quality.

**5.2.4 VGG loss.** We note that neither GaussianAvatars nor RGCA use a perceptual VGG loss. Therefore, for fairness, we run our method without it (No VGG), and show that though we lose quality (Figure 6) we still outperform state-of-the-art methods without it.

## 6 CONCLUSION

We present a novel method for reconstructing photoreal head avatars that can be rendered at high resolution with controllable expressions and viewpoints. Our hybrid approach includes (i) a simple “rigging” model within the continuous tangent space of a facial 3DMM, where canonical Gaussians move freely across triangles for non-greedy optimization; and (ii) a network for expression-dependent deformation and dynamic shading that is agnostic to the number of 3D Gaussians, allowing for densification where most needed and for capturing both static and transient skin features in high detail. In comparison to baselines representing the state-of-the-art, our results show better reconstruction of both overall expression and level of detail.

Our method is however not without limitations: it relies on tracked 3D meshes for training, which are less accurate on areas such as the teeth, tongue, and mouth interior in general; loss of detail and reconstruction artifacts can thus be observed on these areas (which are also more poorly constrained by the training data). In addition, our dynamic shading term cannot change the chromaticity of each Gaussian and thus cannot account for dynamic skin appearance effects due to blood flow. Future work could investigate improving tongue and teeth reconstruction, as well as more comprehensive dynamic appearance effects.

## REFERENCES

- Rameen Abdal, Wang Yifan, Zifan Shi, Yinghao Xu, Ryan Po, Zhengfei Kuang, Qifeng Chen, Dit-Yan Yeung, and Gordon Wetzstein. 2024. Gaussian shell maps for efficient 3D human generation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. CVF / IEEE, 9441–9451.
- Ziqian Bai, Feitong Tan, Zeng Huang, Kripasindhu Sarkar, Danhang Tang, Di Qiu, Abhimitra Meka, Ruofei Du, Mingsong Dou, Sergio Orts-Escobedo, et al. 2023. Learning personalized high quality volumetric head avatars from monocular RGB videos. In

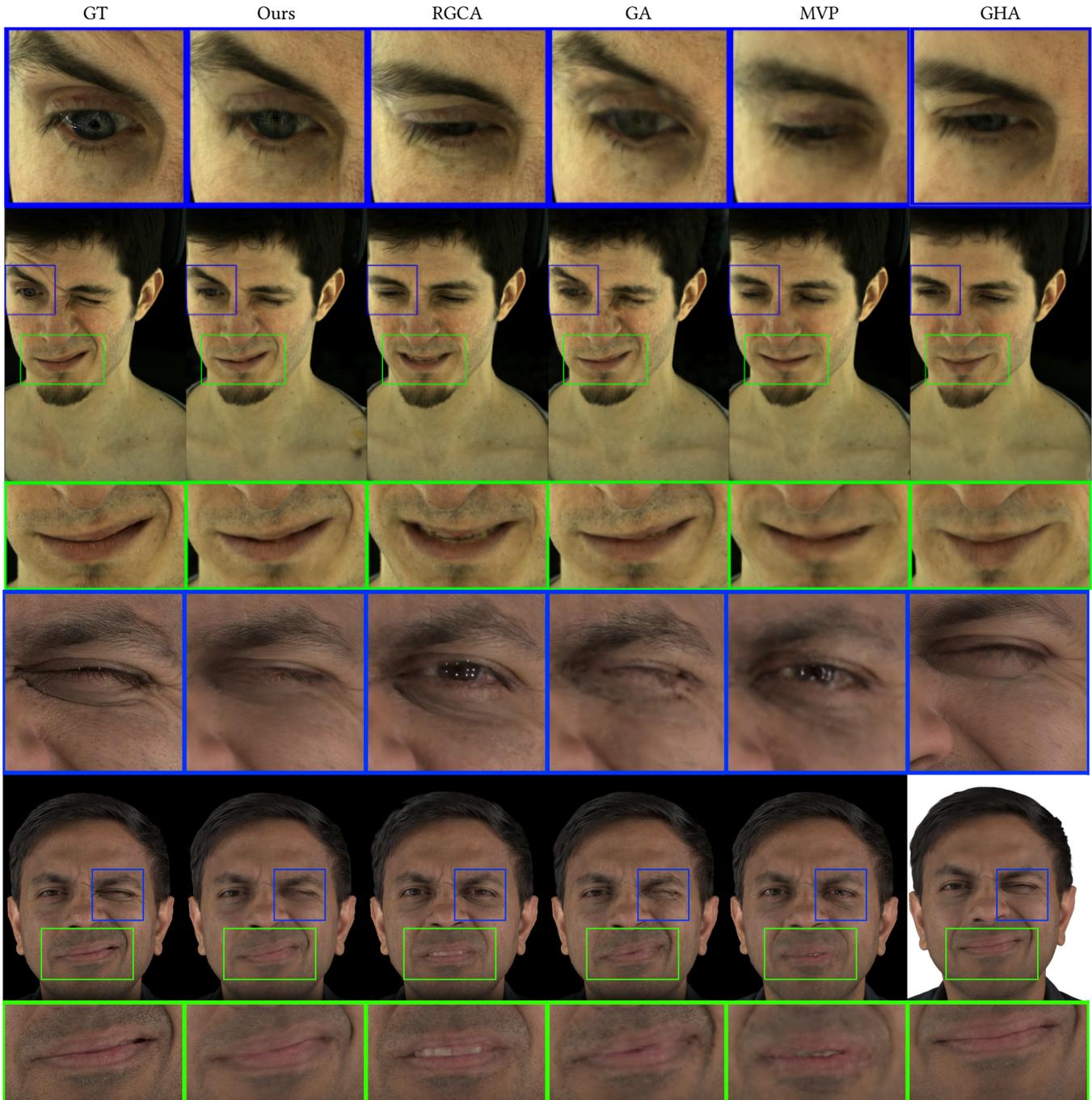


Fig. 8. Our method manages to extrapolate better when compared to RGCA, GA, MVP and GHA. Thanks to the warfield and densification we can more faithfully capture hair and brows (top-left). RGCA struggles to recreate expressions unseen during training (top, subject from Multiface[Wuu et al. 2022]) due to its architecture. GA extrapolates the coarse shape better, but suffers from reduced spatial detail and exhibits artefacts since it doesn't explicitly model dynamic appearance (i.e. lower-right). We encourage readers to zoom in further to see more details

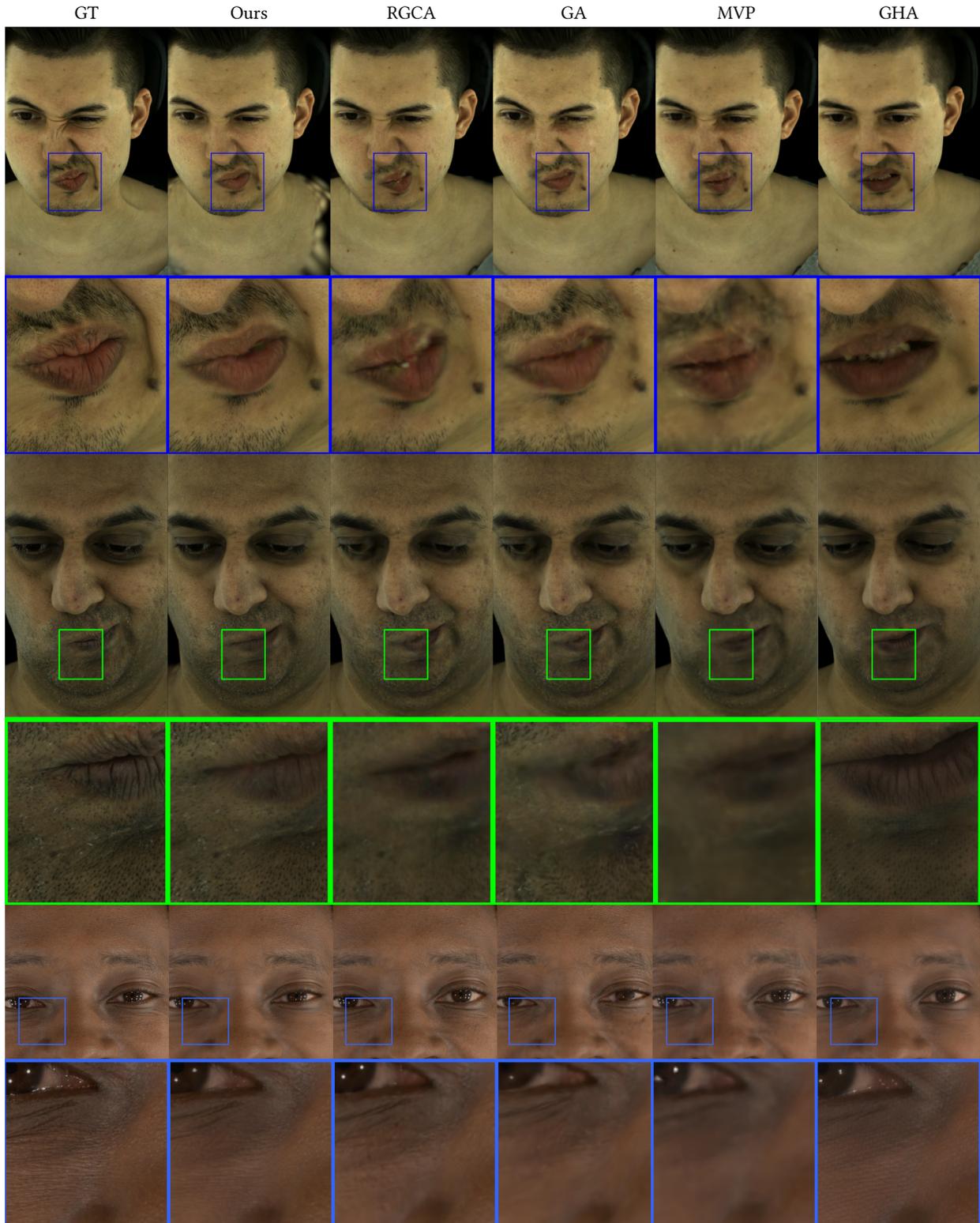


Fig. 9. In comparison to RGCA, GA, MVP and GHA our approach has significantly higher fidelity on details, such as stubble (top, middle, subjects from Multiface [Wuu et al. 2022]) or small wrinkles (bottom).

- Conference on Computer Vision and Pattern Recognition (CVPR)*. CVF / IEEE, 16890–16900.
- Volker Blanz and Thomas Vetter. 1999. A Morphable Model for the Synthesis of 3D Faces. In *SIGGRAPH*. 187–194.
- James Booth, Anastasios Roussos, Stefanos Zafeiriou, Allan Ponniah, and David Dunaway. 2016. A 3D morphable model learnt from 10,000 faces. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. CVF / IEEE, 5543–5552.
- Eric R Chan, Connor Z Lin, Matthew A Chan, Koki Nagano, Boxiao Pan, Shalini De Mello, Orazio Gallo, Leonidas J Guibas, Jonathan Tremblay, Sameh Khamis, et al. 2022. Efficient geometry-aware 3D generative adversarial networks. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. CVF / IEEE, 16123–16133.
- Yufan Chen, Lizhen Wang, Qijing Li, Hongjiang Xiao, Shengping Zhang, Hongxun Yao, and Yebin Liu. 2024. MonoGaussianAvatar: Monocular Gaussian Point-based Head Avatar. In *SIGGRAPH Conference Papers (SA)*. ACM, 58.
- Paul Debevec. 2012. The light stages and their applications to photoreal digital actors. *Transactions on Graphics, (Proc. SIGGRAPH Asia)* 2, 4 (2012), 1–6.
- Helisa Dhamo, Yinyu Nie, Arthur Moreau, Jifei Song, Richard Shaw, Yiren Zhou, and Eduardo Pérez-Pellitero. 2024. HeadGas: Real-Time Animatable Head Avatars via 3D Gaussian Splatting. In *European Conference on Computer Vision (ECCV)*, Vol. 15060. Springer, 459–476.
- Bernhard Egger, William AP Smith, Ayush Tewari, Stefanie Wuhler, Michael Zollhoefer, Thabo Beeler, Florian Bernard, Timo Bolkart, Adam Kortylewski, Sami Romdhani, et al. 2020. 3D morphable face models—past, present, and future. *Transactions on Graphics (TOG)* 39, 5 (2020), 1–38.
- Guy Gafni, Justus Thies, Michael Zollhöfer, and Matthias Nießner. 2021. Dynamic Neural Radiance Fields for Monocular 4D Facial Avatar Reconstruction. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. CVF / IEEE, 8649–8658.
- Baris Gecer, Stylianos Ploumpis, Irene Kotsia, and Stefanos Zafeiriou. 2019. GANFIT: Generative adversarial network fitting for high fidelity 3D face reconstruction. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. CVF / IEEE, 1155–1164.
- Simon Giebenhain, Tobias Kirschstein, Martin Rünz, Lourdes Agapito, and Matthias Nießner. 2024. NPGA: Neural Parametric Gaussian Avatars. In *SIGGRAPH Conference Papers (SA)*. 127:1–127:11.
- Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. *Advances in Neural Information Processing Systems (NeurIPS)* 27 (2014).
- Xiaodong Gu, Zhiwen Fan, Siyu Zhu, Zuozhuo Dai, Feitong Tan, and Ping Tan. 2020. Cascade Cost Volume for High-Resolution Multi-View Stereo and Stereo Matching. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. 2495–2504.
- Yang Hong, Bo Peng, Haiyao Xiao, Ligang Liu, and Juyong Zhang. 2022. HeadNeRF: A Real-time NeRF-based Parametric Head Model. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. CVF / IEEE, 20342–20352.
- Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. 2020. Analyzing and improving the image quality of stylegan. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. CVF / IEEE, 8110–8119.
- Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 2023. 3d gaussian splatting for real-time radiance field rendering. *Transactions on Graphics (TOG)* 42, 4 (2023), 139–1.
- Diederik P Kingma and Jimmy Ba. 2014. Adam: A Method for Stochastic Optimization. *ArXiv* (2014). doi:10.48550/arXiv.1412.6980
- Tobias Kirschstein, Simon Giebenhain, Jiapeng Tang, Markos Georgopoulos, and Matthias Nießner. 2024. GGHead: Fast and Generalizable 3D Gaussian Heads. In *SIGGRAPH Asia Conference Papers (SA)*. ACM, 126:1–126:11.
- Alexandros Lattas, Stylianos Moschoglou, Baris Gecer, Stylianos Ploumpis, Vasileios Triantafyllou, Abhijeet Ghosh, and Stefanos Zafeiriou. 2020. AvatarMe: Realistically Renderable 3D Facial Reconstruction in-the-wild. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. CVF / IEEE, 760–769.
- Alexandros Lattas, Stylianos Moschoglou, Stylianos Ploumpis, Baris Gecer, Jiankang Deng, and Stefanos Zafeiriou. 2023. FitMe: Deep photorealistic 3D morphable model avatars. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. CVF / IEEE, 8629–8640.
- Gengyan Li, Kripasindhu Sarker, Abhimitra Meka, Marcel Buehler, Franziska Mueller, Paulo Gotardo, Otmar Hilliges, and Thabo Beeler. 2024b. ShellNeRF: Learning a Controllable High-resolution Model of the Eye and Periocular Region. *Computer Graphics Forum* (2024). doi:10.1111/cgf.15041
- Junxuan Li, Chen Cao, Gabriel Schwartz, Rawal Khrodgar, Christian Richardt, Tomas Simon, Yaser Sheikh, and Shunsuke Saito. 2024a. URAvatar: Universal Relightable Gaussian Codec Avatars. In *SIGGRAPH Conference Papers (SA)*. 128:1–128:11.
- Ruilong Li, Karl Bladin, Yajie Zhao, Chinmay Chinara, Owen Ingraham, Pengda Xiang, Xinglei Ren, Pratusha Prasad, Bipin Kishore, Jun Xing, et al. 2020. Learning formation of physically-based face attributes. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. CVF / IEEE, 3410–3419.
- Tianye Li, Timo Bolkart, Michael J Black, Hao Li, and Javier Romero. 2017. Learning a model of facial shape and expression from 4D scans. 36, 6 (2017), 194–1.
- Zhe Li, Zerong Zheng, Lizhen Wang, and Yebin Liu. 2024c. Animatable Gaussians: Learning Pose-Dependent Gaussian Maps for High-Fidelity Human Avatar Modeling. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. CVF / IEEE, 19711–19722.
- Stephen Lombardi, Tomas Simon, Gabriel Schwartz, Michael Zollhoefer, Yaser Sheikh, and Jason Saragih. 2021. Mixture of Volumetric Primitives for Efficient Neural Rendering. *ACM Trans. Graph.* 40, 4, Article 59 (jul 2021), 13 pages. doi:10.1145/3450626.3459863
- Zhicheng Lu, Xiang Guo, Le Hui, Tianrui Chen, Ming Yang, Xiao Tang, Feng Zhu, and Yuchao Dai. 2024. 3D Geometry-aware Deformable Gaussian Splatting for Dynamic View Synthesis. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. CVF / IEEE, 8900–8910.
- Jonathon Luiten, Georgios Kopanas, Bastian Leibe, and Deva Ramanan. 2024. Dynamic 3D Gaussians: Tracking by persistent dynamic view synthesis. In *International Conference on 3D Vision (3DV)*. IEEE, 800–809.
- Huiwen Luo, Koki Nagano, Han-Wei Kung, Qingguo Xu, Zejian Wang, Lingyu Wei, Liwen Hu, and Hao Li. 2021. Normalized avatar synthesis using StyleGAN and perceptual refinement. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. CVF / IEEE, 11662–11672.
- Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. 2020. NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis. In *European Conference on Computer Vision (ECCV)*, Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm (Eds.), Vol. 12346. Springer, 405–421.
- Koki Nagano, Jaewoo Seo, Jun Xing, Lingyu Wei, Zimo Li, Shunsuke Saito, Aviral Agarwal, Jens Fursund, Hao Li, Richard Roberts, et al. 2018. paGAN: Real-time avatars using dynamic textures. *Transactions on Graphics (TOG)* 37, 6 (2018), 258.
- Rohit Pandey, Sergio Orts Escolano, Chloe Legendre, Christian Haene, Sofien Bouaziz, Christian Rhemann, Paul Debevec, and Sean Fanello. 2021. Total relighting: learning to relight portraits for background replacement. *ACM Transactions on Graphics (TOG)* 40, 4 (2021), 1–21.
- Foivos Paraperas Papanioniou, Alexandros Lattas, Stylianos Moschoglou, and Stefanos Zafeiriou. 2023. Relightify: Relightable 3d faces from a single image via diffusion models. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. CVF / IEEE, 8806–8817.
- Keunhong Park, Utkarsh Sinha, Jonathan T. Barron, Sofien Bouaziz, Dan B Goldman, Steven M. Seitz, and Ricardo Martin-Brualla. 2021. Nerfies: Deformable Neural Radiance Fields. *International Conference on Computer Vision (ICCV)* (2021).
- Pascal Paysan, Reinhard Kothke, Brian Amberg, Sami Romdhani, and Thomas Vetter. 2009. A 3D face model for pose and illumination invariant face recognition. In *International conference on advanced video and signal based surveillance*. IEEE, 296–301.
- Shenhan Qian, Tobias Kirschstein, Liam Schoneveld, Davide Davoli, Simon Giebenhain, and Matthias Nießner. 2024. GaussianAvatars: Photorealistic Head Avatars with Rigged 3D Gaussians. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. CVF / IEEE, 20299–20309.
- Alfredo Rivero, ShahRukh Athar, Zhixin Shu, and Dimitris Samaras. 2024. Rig3DGS: Creating controllable portraits from casual monocular videos. *arXiv preprint arXiv:2402.03723* (2024).
- Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-resolution image synthesis with latent diffusion models. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. CVF / IEEE, 10684–10695.
- Shunsuke Saito, Stanislav Pidhorskyi, Igor Santesteban, Forrest Iandola, Divam Gupta, Anuj Pahuja, Nemanja Bartolovic, Frank Yu, Emanuel Garbin, and Tomas Simon. 2024a. SqueezeMe: Efficient Gaussian Avatars for VR. *arXiv preprint arXiv:2412.15171* (2024).
- Shunsuke Saito, Gabriel Schwartz, Tomas Simon, Junxuan Li, and Giljoon Nam. 2024b. Relightable gaussian codec avatars. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. CVF / IEEE, 130–141.
- Kripasindhu Sarker, Marcel C. Buehler, Gengyan Li, Daoye Wang, Delio Vicini, Jérémy Riviere, Yinda Zhang, Sergio Orts-Escolano, Paulo Gotardo, Thabo Beeler, and Abhimitra Meka. 2023. LitNeRF: Intrinsic Radiance Decomposition for High-Quality View Synthesis and Relighting of Faces. In *ACM SIGGRAPH Asia 2023 Conference Papers, December 12–15, 2023, Sydney, NSW, Australia*. doi:10.1145/3610548.3618210
- Jack Saunders, Charlie Hewitt, Yanan Jian, Marek Kowalski, Tadas Baltrusaitis, Yiye Chen, Darren Cosker, Virginia Estellers, Nicholas Gyde, Vinay P. Nambodiri, and Benjamin E Lundell. 2024. GASP: Gaussian Avatars with Synthetic Priors. arXiv:2412.07739 [cs.CV] <https://arxiv.org/abs/2412.07739>
- Zhijiang Shao, Zhaolong Wang, Zhuang Li, Duotun Wang, Xiangru Lin, Yu Zhang, Mingfan Fan, and Zeyu Wang. 2024. SplattingAvatar: Realistic Real-Time Human Avatars with Mesh-Embedded Gaussian Splatting. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. CVF / IEEE, 1606–1616.
- William AP Smith, Alassane Seck, Hannah Dee, Bernard Tiddeman, Joshua B Tenenbaum, and Bernhard Egger. 2020. A morphable face albedo model. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. CVF / IEEE, 5011–5020.

- Kartik Teotia, Hyeonwoo Kim, Pablo Garrido, Marc Habermann, Mohamed Elgharib, and Christian Theobalt. 2024. GaussianHeads: End-to-End Learning of Drivable Gaussian Head Avatars from Coarse-to-fine Representations. *Transactions on Graphics (TOG)* 43, 6 (2024), 264:1–264:12.
- Jie Wang, Jiu-Cheng Xie, Xianyan Li, Feng Xu, Chi-Man Pun, and Hao Gao. 2024. GaussianHead: High-fidelity Head Avatars with Learnable Gaussian Diffusion. [arXiv:2312.01632 \[cs.CV\]](https://arxiv.org/abs/2312.01632)
- Guanjun Wu, Taoran Yi, Jiemin Fang, Lingxi Xie, Xiaopeng Zhang, Wei Wei, Wenyu Liu, Qi Tian, and Xinggang Wang. 2024. 4D Gaussian Splatting for Real-Time Dynamic Scene Rendering. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. CVF / IEEE, 20310–20320.
- Cheng-hsin Wu, Ningyuan Zheng, Scott Ardisson, Rohan Bali, Danielle Belko, Eric Brockmeyer, Lucas Evans, Timothy Godisart, Hyowon Ha, Xuhua Huang, Alexander Hypes, Taylor Koska, Steven Krenn, Stephen Lombardi, Xiaomin Luo, Keyvyn McPhail, Laura Millerschoen, Michal Perdoch, Mark Pitts, Alexander Richard, Jason Saragih, Junko Saragih, Takaaki Shiratori, Tomas Simon, Matt Stewart, Autumn Trimble, Xinshuo Weng, David Whitewolf, Chenglei Wu, Shou-I Yu, and Yaser Sheikh. 2022. Multiface: A Dataset for Neural Face Rendering. In *arXiv*. doi:10.48550/ARXIV.2207.11243
- Jun Xiang, Xuan Gao, Yudong Guo, and Juyong Zhang. 2024. FlashAvatar: High-fidelity Head Avatar with Efficient Gaussian Embedding. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. CVF / IEEE, 1802–1812.
- Yuting Xiao, Xuan Wang, Jiafei Li, Hongrui Cai, Yanbo Fan, Nan Xue, Minghui Yang, Yujun Shen, and Shenghua Gao. 2024. Bridging 3D Gaussian and Mesh for Freeview Video Rendering. *arXiv preprint arXiv:2403.11453* (2024).
- Yuelang Xu, Benwang Chen, Zhe Li, Hongwen Zhang, Lizhen Wang, Zerong Zheng, and Yebin Liu. 2024. Gaussian Head Avatar: Ultra High-fidelity Head Avatar via Dynamic Gaussians. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. CVF / IEEE, 1931–1941.
- Ziyi Yang, Xinyu Gao, Wen Zhou, Shaohui Jiao, Yuqing Zhang, and Xiaogang Jin. 2024. Deformable 3D Gaussians for High-Fidelity Monocular Dynamic Scene Reconstruction. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. CVF / IEEE, 20331–20341.
- Heng Yu, Joel Julin, Zoltán Ádám Milacski, Koichiro Niinuma, and László A. Jeni. 2024. CoGS: Controllable Gaussian Splatting. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. CVF / IEEE, 21624–21633.
- Longwen Zhang, Qiwei Qiu, Hongyang Lin, Qixuan Zhang, Cheng Shi, Wei Yang, Ye Shi, Sibe Yang, Lan Xu, and Jingyi Yu. 2023. DreamFace: Progressive Generation of Animatable 3D Faces under Text Guidance. *Transactions on Graphics (TOG)* 42, 4 (2023), 138:1–138:16.
- Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. 2018. The Unreasonable Effectiveness of Deep Features as a Perceptual Metric. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. CVF / IEEE, 586–595.
- Zhongyuan Zhao, Zhenyu Bao, Qing Li, Guoping Qiu, and Kanglin Liu. 2024. PSAvatar: A point-based morphable shape model for real-time head avatar creation with 3D Gaussian splatting. *arXiv preprint arXiv:2401.12900* (2024).
- Yiyu Zhuang, Hao Zhu, Xusen Sun, and Xun Cao. 2022. MoFaNeRF: Morphable Facial Neural Radiance Field. In *European Conference on Computer Vision (ECCV)*. 268–285.
- Wojciech Zielonka, Timo Bolkart, Thabo Beeler, and Justus Thies. 2024. Gaussian Eigen Models for Human Heads. [arXiv:2407.04545 \[cs.CV\]](https://arxiv.org/abs/2407.04545)
- Wojciech Zielonka, Timo Bolkart, and Justus Thies. 2022. Instant Volumetric Head Avatars. *Conference on Computer Vision and Pattern Recognition (CVPR)* (2022), 4574–4584.
- Wojciech Zielonka, Stephan J Garbin, Alexandros Lattas, George Kopanas, Paulo Gotardo, Thabo Beeler, Justus Thies, and Timo Bolkart. 2025. Synthetic Prior for Few-Shot Drivable Head Avatar Inversion. *arXiv preprint arXiv:2501.06903* (2025).