This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Gaussian Eigen Models for Human Heads

Wojciech Zielonka[1, 2] Timo Bolkart[3] Thabo Beeler[3] Justus Thies[1, 2]

[1]Max Planck Institute for Intelligent Systems, Tübingen, Germany
[2]Technical University of Darmstadt [3]Google
https://zielon.github.io/gem/

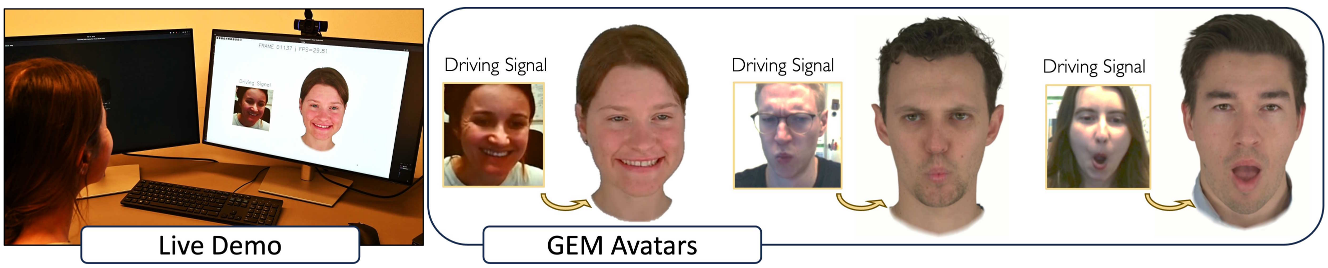


Figure 1. We propose a method that represents 3D Gaussian head avatars in a network-free form as ensembles of eigenbases (GEM). Only a linear combination of these bases is needed to generate new primitives, which can be splatted using 3D Gaussian Splatting. We demonstrate that the necessary coefficients for a specific expression can be regressed from single images, enabling real-time facial animation and cross-reenactment. The simplicity of GEM results in highly efficient storage and rendering times.

Abstract

Current personalized neural head avatars face a trade-off: lightweight models lack detail and realism, while high-quality, animatable avatars require significant computational resources, making them unsuitable for commodity devices. To address this gap, we introduce Gaussian Eigen Models (GEM), which provide high-quality, lightweight, and easily controllable head avatars. GEM utilizes 3D Gaussian primitives for representing the appearance combined with Gaussian splatting for rendering. Building on the success of mesh-based 3D morphable face models (3DMM), we define GEM as an ensemble of linear eigenbases for representing the head appearance of a specific subject. In particular, we construct linear bases to represent the position, scale, rotation, and opacity of the 3D Gaussians. This allows us to efficiently generate Gaussian primitives of a specific head shape by a linear combination of the basis vectors, only requiring a low-dimensional parameter vector that contains the respective coefficients. We propose to construct these linear bases (GEM) by distilling high-quality compute-intense CNN-based Gaussian avatar models that can generate expression-dependent appearance changes like wrinkles. These high-quality models are trained on multi-view videos of a subject and are distilled using a series of principle component analyses.

Once we have obtained the bases that represent the animatable appearance space of a specific human, we learn a regressor that takes a single RGB image as input and predicts the low-dimensional parameter vector that corresponds to the shown facial expression. We demonstrate that this regressor can be trained such that it effectively supports self- and cross-person reenactment from monocular videos without requiring prior mesh-based tracking. In a series of experiments, we compare GEM's self-reenactment and cross-person reenactment results to state-of-the-art 3D avatar methods, demonstrating GEM's higher visual quality and better generalization to new expressions. As our distilled linear model is highly efficient in generating novel animation states, we also show a real-time demo of GEMs driven by monocular webcam videos. The code and model will be released for research purposes.

1. Introduction

Half a century ago, Frederick Parke described a representation and animation technique to generate „animated sequences of a human face changing expressions" [40]. Using polygonal meshes, single facial expression states were described that could be combined with linear interpolation to generate new expression states (the „simplest way consistent with natural motion" [40]). Based on this principle, Blanz, and Vetter [2] introduced the so-called 3D morphable model (3DMM) - a statistical model of the 3D shape and appearance of human faces. Principle Component Analysis (PCA) is performed on a set of around 200 subjects that have been laser-scanned and registered to a consistent template to find the displacement vectors (principal components) of how faces change the most, in terms of geometry and albedo. With this PCA basis, new faces can be generated by specifying the coefficients for the principle components taking a dot product of the coefficients with the basis to obtain offsets, and adding them to the mean. State-of-the-art reports on face reconstruction and tracking [68] as well as on morphable models [6] state that this representation is widely used for facial performance capturing (regression-based and optimization-based) and builds the backbone of recent controllable photo-realistic 3D avatars that are equipped with neural rendering [10, 15, 48, 49, 62].

Inspired by the simplicity of such mesh-based linear morphable models and addressing the lack of appearance realism of current 3DMMs, we propose a personalized linear appearance model based on 3D Gaussians as geometry primitives following 3D Gaussian Splatting (3DGS) [21]. In contrast to the work on Dynamic 3D Gaussian Avatars [30, 38, 43, 45, 57, 60, 66], our goal is a compact and light representation that does not need vast amounts of compute resources to generate novel expressions of the human. Unfortunately, most of the methods show that to produce high-quality results, one needs to employ heavy CNN-based architectures which are not well suited for commodity devices and tend to slow down the rendering pipeline. Moreover, those models comprise dozens of millions of parameters creating heavy checkpoints that can easily exceed 500 MB. This ultimately creates a major issue for distributing and managing personalized models. We tackle this problem by distilling a CNN-based architecture, leading to a personalized **Gaussian Eigen Models for Human Heads**, **GEM** in short. Our approach builds on Gaussian maps predicted from a modified UNet architecture [53] which is used for the UV space normalization required to build linear eigenbases. Based on the per-subject trained CNN model, we bootstrap the GEM by computing an ensemble of linear bases on the predicted Gaussian maps of the training frames. The bases are refined on the training corpus using photometric losses while preserving their orthogonality.

These lightweight appearance bases are controlled with a relatively low number of parameters ranging from twenty up to fifty coefficients which can be specified w.r.t. the available compute resources and can for example be regressed by a ResNet-based model [8]. We demonstrate this for self-reenactment as well as cross-person animation, including a real-time demo in the suppl. video.

In summary, our main contributions are:

1. Gaussian Eigen Models for Human Heads (GEM), a distillation technique of 3D Gaussian head avatar models built upon an ensemble of eigenbases.
2. real-time (cross-person) animation of GEMs from single input images using a generalizable regressor.

2. Related Work

The majority of face representation and tracking techniques are based on parametric 3D morphable models (3DMM) [2, 29]. For a detailed overview, we refer to the state-of-the-art reports on face tracking and reconstruction [68], the report on morphable models [6], and the two neural rendering state-of-the-art reports [48, 49] that demonstrate how neural rendering can be leveraged for photo-realistic facial or full body avatars. Next, we review the recent methods for photo-realistic 3D avatars generation which build appearance models using neural radiance fields (NeRF) [35] or volumetric primitives like 3D Gaussians [21].

2.1. NeRF-based avatars

One of the first methods that combines a 3DMM and NeRF is NeRFace [10], where a neural radiance field is directly conditioned by expression codes of the Basel Face Model (BFM) [2, 50]. This idea gave rise to many methods [11, 15, 42, 56, 59, 62–64] following a similar approach, but attaching the radiance fields more explicitly to the surface of the 3DMM, e.g., by using the 3DMM-defined deformation field. For photorealistic results, some methods employ StyleGAN2-like architectures [20] with a NeRF-based renderer [1, 4, 19]. Generative methods like EG3D [4] and PanoHead [1] employ GAN-based training to predict triplane features that span a NeRF. GANAvatar [19] applies this scheme to reconstruct a personalized avatar.

Close to our method is StyleAvatar [53]. Based on 3DMM tracking the method learns a personalized avatar that benefits from a StyleUNet which incorporates StyleGAN [20] to decode the final image. Despite real-time capabilities, StyleAvatar suffers from artifacts produced by the image-to-image translation network that we explicitly avoid by using Gaussian maps which can compensate for tracking misalignments by predicting corrective fields for the 3D Gaussians.

2.2. 3D Avatars from Volumetric Primitives

Using multiview images with a variational auto-encoder [22] and volumetric integration, Neural Volumes (NV) [31] encodes dynamic scenes into a volume which can be deformed by traversing a latent code **z**. To better control the 3D space, Lombardi *et al*. [32] introduce Mixture of Volumetric Primitives (MVP) a hybrid representation based on primitives attached to a tracked mesh which ultimately replaced the encoder from NV. Each primitive is a volume

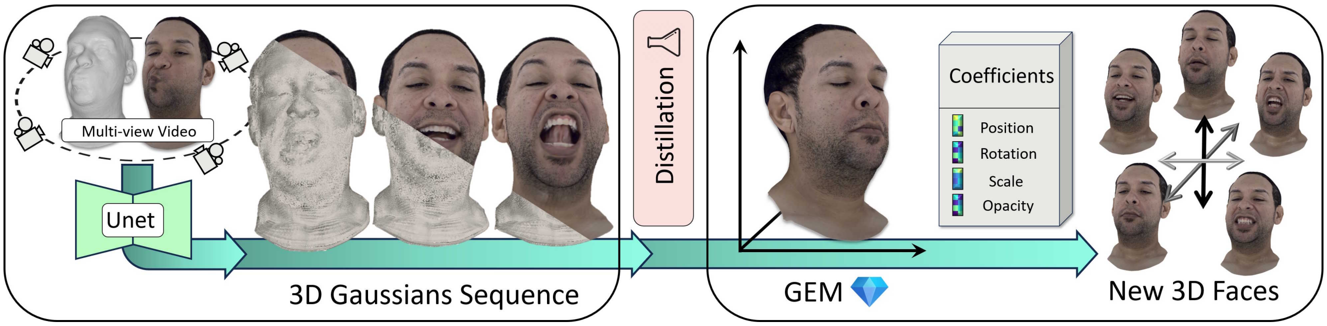


Figure 2. Given a multi-view video of a subject and mesh tracking, we create a dataset of 3D Gaussian point clouds for each frame in the sequence. Using this data, we distill a high-quality Gaussian Eigen Model (GEM). GEM is an ensemble of linear bases for each Gaussian primitive modality: position, opacity, scale, and rotation. Based on these bases, facial appearances are generated by a linear combination.

represented as a small voxel with 32^3 cells that store RGB and opacity values. The final color is obtained by integrating values along a pixel ray. This hybrid representation inspired many follow-up projects [3, 28, 44, 47, 54]. As an alternative to MVP primitives, 3D Gaussian Splatting (3DGS) [21] represents a volume as a set of anisotropic 3D Gaussians, which are equivalently described as ellipsoids, in contrast to isotropic spheres used in Pulsar [25].

Numerous methods [9, 13, 17, 24, 30, 38, 43, 45, 55, 57, 61, 66, 67] capitalize on the speed and quality of 3DGS. Qian *et al*. [43] attach 3D Gaussians to the FLAME [29] mesh surface and apply a deformation gradient similar to Zielonka et al. [64] to orient the Gaussians according to the local Frenet frames of the surface. This method, however, does not utilize any information about expressions and, thus, struggles with pose-dependent changes (e.g., wrinkles, self-shadows) and, despite high-quality results, retrieves only a global static appearance model. 3D Gaussian blendshapes [34] controls an avatar by linearly interpolating between optimized blendshapes using 3DMM expression coefficents. However, this method depends on an underlying 3DMM whereas GEM is a mesh-free representation. Li *et al*. [30] use a StyleUNet-like CNN architecture [53] to regress front and back Gaussian maps. Employing a powerful CNN network on position maps, they achieve impressive results for human bodies with effects like pose-dependent wrinkle formation.

Please note that in this work, we focus on methods that directly output Gaussian primitives. This is an important distinction from a branch of methods that follow Deferred Neural Rendering [51], where a refinement CNN translates splatted features or coarse colors into the final image; for instance, Gaussian Head Avatars [57] and NGPA [13]. This distinction is important because Gaussian primitives cannot be fully distilled into an eigenbasis in this context, as the refinement CNN network is required to complete the rendering directly in the image space.

2.3. 3DGS Compression Methods

Recently, several methods [7, 14, 26, 27, 36, 39] have been proposed to reduce the memory footprint of 3D Gaussian Splatting (3DGS). Papantonakis *et al*. [39] apply codebook quantization to the Gaussian primitive properties, alongside pruning of Spherical Harmonic (SH) coefficients based on their final contribution. In contrast to postprocessing approaches [7, 27, 39], Compact3D [36] employs a single-stage process that jointly optimizes both the codebook entries and the primitives. Fan *et al*. [7] calculate a significance score for each primitive by measuring its pixel hit count, thereby improving the pruning strategy. Most of these methods target static scenes or time-conditioned environments, unlike our approach, which focuses on efficient, fully controllable head avatars. Nonetheless, these compression techniques could be adapted to our animatable avatars to reduce memory usage.

3. Method

Recent dynamic 3D Gaussian Avatar methods show unprecedented quality, however, they require sophisticated and often compute-heavy CNN-based architectures [30, 38, 57] to capture high-frequency and dynamic details like pose-dependent wrinkles or self-shadows. The aim of this paper is to build on top of this quality but remove the compute-intense architecture during inference. Specifically, we propose to distill high-quality avatar models into lightweight linear animation models which we call GEMs. A GEM is defined by an ensemble of eigenbases that span the space of the 3D Gaussian primitives. These eigenbases are constructed via PCA applied on a dataset of per-frame Gaussian primitives, see Section 3.1.

An important distinction compared to other neural avatars [10, 15, 31, 53, 57, 64] is that GEM does not require a 3DMM [29, 41] at test time. We demonstrate that a GEM can be directly driven by a monocular video using a generalized image-based regression network, see Section 3.2.

3.1. Gaussian Eigen Model (GEM)

For our distillation, we reconstruct a sequence of normalized Gaussian primitives $\boldsymbol{D} = \{\boldsymbol{G}_0, ..., \boldsymbol{G}_{N-1}\}$. As input, we assume a multi-view video of the subject with N time frames. Per time frame i, we reconstruct the 3D Gaussian pointcloud $\boldsymbol{G}_i$, where $\boldsymbol{G}_i$ contains the parameters that define the 3D Gaussians such as rotation $\vec{\theta}$, position $\vec{\phi}$, opacity $\vec{\alpha}$, scale $\vec{\sigma}$, and color $\vec{c}$ such that $\boldsymbol{G}_i = \{\vec{\theta}, \vec{\phi}, \vec{\alpha}, \vec{\sigma}, \vec{c}\}$.

Reconstructing High-quality 3D Gaussian Primitives: We are following the idea of organizing the 3D Gaussians in 2D maps [30, 38, 45, 57], where each pixel represents a 3D Gaussian with its parameters. We propose an adapted CNN-architecture of Animatable Gaussians (AG) [30], by merging the separate Style-U-Nets, reducing the convolutional layers, and operating in the UV space of the FLAME head model. In addition, we are employing deformation gradients following Sumner *et al*. [46] to handle the transformation from canonical to deformed space and treat the color as a global parameter. We refer to the suppl. mat. for a detailed explanation of the architectural changes. In comparison to the original AG model, our proposed CNN model produces slightly better results while being more efficient in terms of computing and memory. Using this model, we generate the per-frame Gaussian primitives $\boldsymbol{G}_i$ in the canonical space for all training time-frames. Note that for this reconstruction, we follow Animatable Gaussians and, thus, FLAME-based tracking is required. However, during inference, our model is independent of FLAME.

Distillation: Given $\boldsymbol{D} = \{\boldsymbol{G}_0, ..., \boldsymbol{G}_{N-1}\}$, we build a personalized eigenbasis model, which is called GEM. We compute a statistical model for each Gaussian modality separately. Specifically, we create individual bases for rotation $\boldsymbol{B}_\theta$, position $\boldsymbol{B}_\phi$, opacity $\boldsymbol{B}_\alpha$, and scale $\boldsymbol{B}_\sigma$ with respective means $\vec{\mu}_\theta$, $\vec{\mu}_\phi$, $\vec{\mu}_\alpha$ and $\vec{\mu}_\sigma$ via Principle Complonent Analysis (PCA) [18]. Note that the color $\boldsymbol{C}$ is optimized globally and, thus, acts as a classical texture without the need to apply PCA. To accurately learn dynamically moving Gaussians, we fixed the color to prevent it from dominating the image representation, otherwise, Gaussians could change their semantic meaning (e.g., a Gaussian could represent the lip in one state, and the teeth in the other deformation state). Keeping the semantic meaning of specific Gaussians across deformation states is crucial for applying a PCA afterward.

A face model instance $\boldsymbol{G}$ is represented as a linear combination of these bases:

$$\boldsymbol{G} = \{\vec{\mu}_i + \mathbf{B}_i \mathbf{k}_i \mid i \in \{\theta, \phi, \alpha, \sigma\}, \vec{c}\}\,, \tag{1}$$

where $\mathbf{k}_\theta$, $\mathbf{k}_\phi$, $\mathbf{k}_\theta$ and $\mathbf{k}_\sigma \in \mathbb{R}^M$ are the linear coefficients which are defining the facial expression state, assuming M principal components. As an example, Figure 3 shows position parameter $\mathbf{k}_\phi$ sampled in the range of $[-3\sigma_\phi, 3\sigma_\phi]$ (σ_ϕ being the std. deviation).

Figure 3. **Samples of a GEM.** We display samples for the first three components of the position $\mathbf{k}_\phi$ eigenbasis of a GEM, showing diverse expressions. Note that GEM requires **no** parametric 3D face model like FLAME[29].

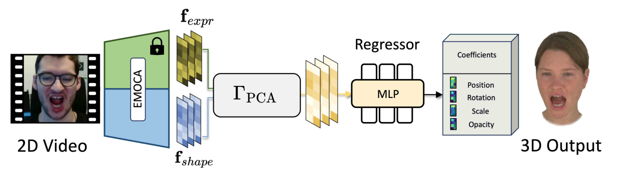


Figure 4. **Image-based animation.** One of the applications of our GEM is real-time (cross)-reenactment. For that, we utilize generalized features from EMOCA [5] and build a pipeline to regress the PCA coefficients of our model from an input image/video.

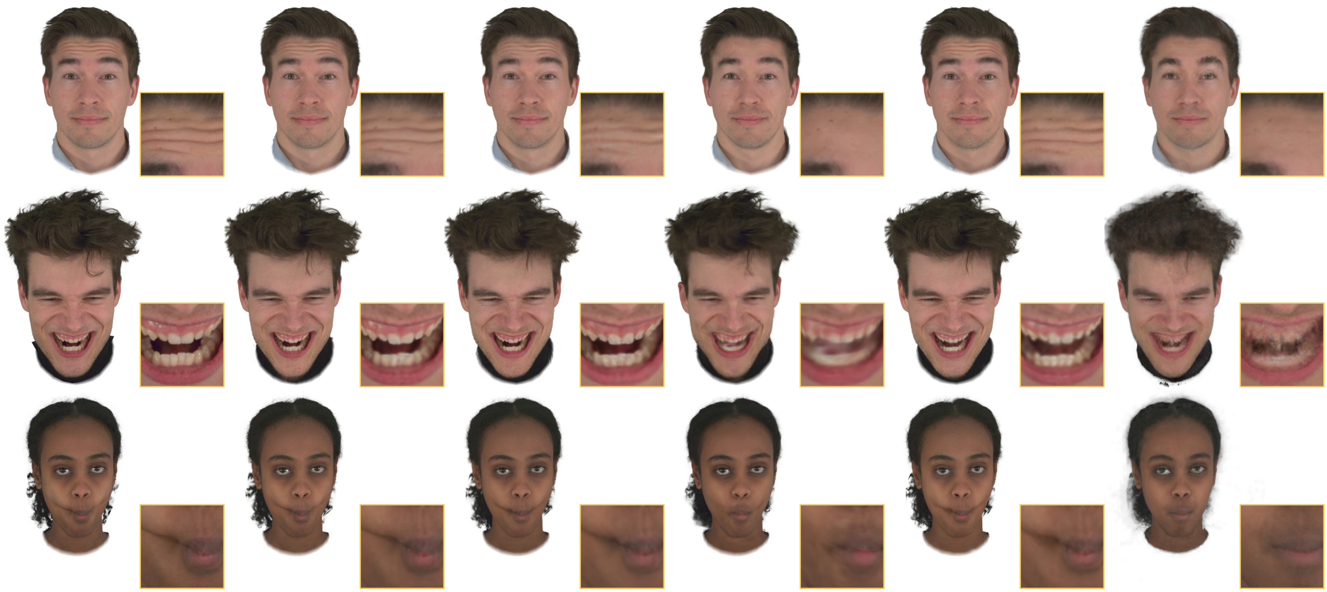
As the Gaussian primitives $\mathbf{D}$ might contain tracking failures and misalignments, the principle components $\boldsymbol{B}_{(\theta,\phi,\alpha,\sigma)}$ also contain artifacts as well. We, therefore, refine the bases using the training images directly, by applying a photometric reconstruction loss. We employ the same objectives from the CNN model training (see supp. mat).

$$\mathcal{L}_{Color} = (1-\omega)\mathcal{L}_1 + \omega\mathcal{L}_{\text{D-SSIM}} + \zeta\mathcal{L}_{\text{VGG}} \tag{2}$$

We refine the base vectors for around 30k iterations. To ensure that the individual bases stay orthonormal throughout this refinement, every 1k steps, we orthogonalize the bases using QR decomposition. This refinement improves the training PSNR errors from 34.75dB to 36.68dB and 36.85dB for the training steps 0k, 5k, and 30k, respectively. Throughout our experiments, we did not encounter overfitting issues with this scheme. The reconstruction metrics on two randomly selected test sequences with refinement are: PSNR: **31.51**, LPIPS: **0.091**, SSIM: **0.936**; and without: PSNR: 31.38, LPIPS: 0.094, SSIM: 0.933.

3.2. Image-based Animation

Expressions for a GEM are fully defined by their coefficients $\mathbf{k}_\theta$, $\mathbf{k}_\phi$, $\mathbf{k}_\theta$ and $\mathbf{k}_\sigma$. This is a similar idea to codec avatars [33], however, our approach does not need additional pixel shaders in the form of a small regression MLP. There are several ways to obtain the coefficients of a GEM, for example, one can employ analysis-by-synthesis-based optimization or regression. Analysis-by-synthesis [2] is the backbone of current avatar methods, as they use photomet-



Ground Truth Ours GEM Ours Net GA [43] AG [30] INSTA [64]

Figure 5. **Novel view synthesis.** Both, our CNN and GEM show better performance on novel views, especially, in the region of the mouth interior and wrinkles. In this experiment, we are following the evaluation of Gaussian Avatars [43] and demonstrate novel viewpoint generation. GEM is obtained throughout analysis-by-synthesis fitting [2, 50]. Note that the expressions are seen during training.

ric or depth-based face trackers to sequentially optimize the coefficients of the underlying 3DMMs like FLAME [15, 50, 64] which is typically slow. As a fast, but more imprecise alternative, regressors like DECA [8] or EMOCA [5] can be used which are built on a ResNet backbone and regress FLAME parameters directly from an image. We apply several modifications to the EMOCA model, see 4. We use intermediate features of the pre-trained EMOCA network denoted as $\Theta(\mathbf{I}_i)$ where $\mathbf{I}_i$ is the current image. EMOCA's architecture comprises two ResNet networks; one to extract expression features $\mathbf{f}_{expr} \in \mathbb{R}^{2048}$ and the second for shape $\mathbf{f}_{shape} \in \mathbb{R}^{2048}$, both are followed by final MLPs to regress corresponding FLAME parameters. As we do not rely on FLAME, we remove the last hidden layer of the final MLP obtaining two feature vectors which we combine into one $\mathbf{f} \in \mathbb{R}^{2\times 1024}$ vector. For these features, we build a PCA layer with a basis denoted as $\hat{\mathbf{R}}$ using the training frames from five frontal cameras of NeRSemble. Note that we use relative features $\mathbf{r} = \mathbf{f} - \mathbf{f}_{neutral}$ in this PCA layer. The neutral reference frame $\mathbf{f}_{neutral} = \Theta(\mathbf{I}_{neutral})$ to compute these relative features is selected manually from the video, similar to Face2Face [50]. During training, for each frame, we project $\mathbf{r}$ onto the PCA manifold using the first 50 principal components to restrict and regularize training. Finally, we use their corresponding PCA coefficients:

$$\kappa = (\mathbf{r} - \bar{\mathbf{R}})\hat{\mathbf{R}}^T, \tag{3}$$

where $\bar{\mathbf{R}}$ is the relative PCA model mean. The projected coefficients are passed through a small MLP that produces a vector of GEM coefficients $\mathbf{k} = \{\mathbf{k}_\theta, \mathbf{k}_\phi, \mathbf{k}_\theta, \mathbf{k}_\sigma\}$:

$$\mathbf{k} = 3 \cdot \sigma_k \cdot \tanh(\mathrm{MLP}(\kappa)). \tag{4}$$

The MLP has three hidden layers with 256 neurons each and ReLU activations. We use a scaled $\tanh$ activation function for the output to restrict the prediction to be in $[-3 \cdot \sigma_k, 3 \cdot \sigma_k]$, σ_k being the respective standard deviation of the coefficients $\mathbf{k}$, obtained from the PCA. The final primitives are obtained by Eq. 1 and splatted using 3DGS.

4. Results

We evaluate GEM on the NeRSemble [23], where tracked meshes [43] and synchronized images from 16 cameras with a resolution of 802×550 are available. Our baselines are Gaussian Avatars (**GA**) [43] which is neural network-free (Gaussians are attached to the FLAME model), our implementation of Animatable Gaussians (**AG**) [30] which is based on CNN-predicting Gaussian maps, and INSTA [64] which uses dynamic NeRF [35]. Note that all baselines require at least two stages: (i) construct the avatar, and (ii) get the parameters to drive it. Most of them use offline tracking with additional objectives like hair reconstruction [12, 43], which does not work for real-time applications despite the avatar model's rendering being real-time. Importantly, in our approach, we introduce a third step, i.e., the construction of the eigenbasis (GEM), which only introduces **negligible** computational costs ($\sim$ **1 min**) in comparison to the avatar reconstruction itself. For the comparison, we present both of our appearance models, the StyleUNet-based architecture (**Ours Net**) and the distilled linear Eigen model (**Ours GEM**) which we evaluate using analysis-by-synthesis fitting to the target images following [2, 50, 65]. Additionally, we present cross-reenactment results based on our coefficient regressor, compared to the baselines that

Figure 6. **Novel view and expression synthesis.** Our Gaussian Eigen Models for Human Heads shows better results in regions like teeth, wrinkles, and self-shadows compared to other methods that struggle with artifacts.

Method	PSNR ↑	LPIPS ↓	SSIM ↑	L1 ↓
AG [30]	32.4166	0.0712	0.9614	0.0066
GA [43]	31.3197	0.0786	0.9567	0.0075
INSTA [64]	27.7786	0.1232	0.9294	0.0163
Ours Net	32.4622	0.0713	0.9617	0.0067
Ours GEM	**33.5528**	**0.0678**	**0.9662**	**0.0061**

Table 1. **Novel viewpoint evaluation** is conducted on a withhold camera from the 16 cameras used for training. Note that the expression has been seen during training, and only the view is new.

Method	PSNR ↑	LPIPS ↓	SSIM ↑	L1 ↓
AG [30]	29.0114	0.0812	0.9429	0.0099
GA [43]	28.3137	0.0815	0.9433	0.0102
INSTA [64]	27.9181	0.1153	0.9340	0.0128
Ours Net	29.2454	0.0777	0.9448	0.0096
Ours GEM	**32.6781**	**0.0675**	**0.9633**	**0.0069**

Table 2. **Evaluation on novel expressions** and views show improved results of GEM optimized using analysis-by-synthesis compared to others. Figure 6 shows the corr. qualitative results.

use FLAME meshes regressed by DECA [8]. Relative expression transfer based on ground truth meshes [43] can be found in the supp. mat. All of the methods are evaluated using several image space metrics on novel expressions and novel views, following the test and novel-view split of Qian *et al*. [43]. For our GEM models, we use 50 components distilled from 256^2 textures which give around 60k active Gaussians. Animatable Gaussians [30] uses a similar amount of primitives for front and back textures and Gaussian Avatars [43] around 100k Gaussians.

4.1. Image Quality Evaluation

To evaluate our method, we measure the color error in the image space using the following metrics: PSNR (dB), LPIPS [58], L1 loss, and structural similarity (SSIM). We follow the evaluation scheme from Gaussian Avatars [43], using their train and validation split. The evaluation of GEM was generated by sequentially fitting the coefficients to each image using photometric objectives. Note that the baselines use the FLAME model with offsets for the tracking, while GEM can directly be used for tracking.

Table 2 presents results on novel expressions evaluated on all 16 cameras. Both the quantitative and qualitative results depicted in Figure 6 show that our PCA model produces fewer artifacts, especially for regions like teeth or facial wrinkles. Table 1 contains an evaluation where we measure errors on novel viewpoints. The results demonstrate that our CNN-based appearance model outperforms other neural methods, while our linear eigenbasis GEM achieves the highest quality. This is due to the 'direct' analysis-by-synthesis approach, which fully leverages the expressiveness and detail of our photorealistic appearance model, without the limitations imposed by 3DMMs such as FLAME. Moreover, Figure 5 shows qualitative results of our method on novel views. As can be seen, we better capture high-frequency details, pose-dependent wrinkles, and self-shadows - something which is not possible for methods like Gaussian Avatars [43] or INSTA [64], since they either do not use expression-dependent neural networks or limit the conditioning to a small region only.

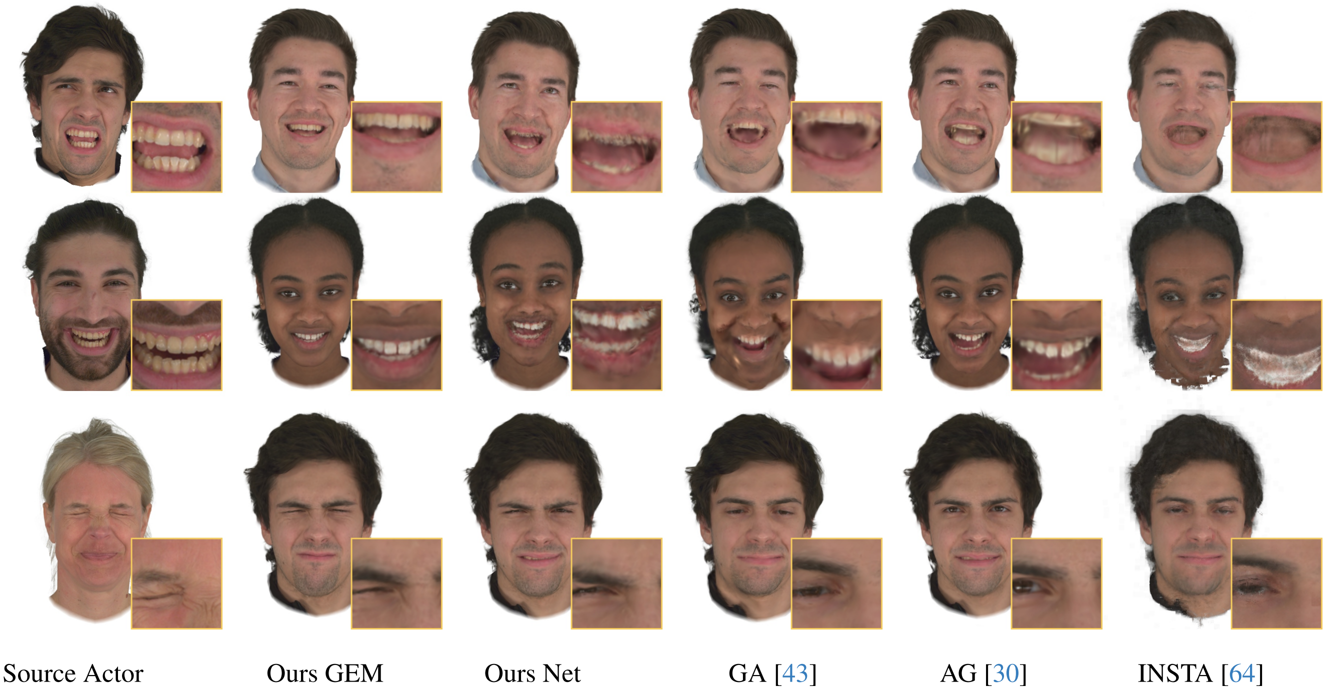


Figure 7. **Facial cross-person reenactment using an image-based regressor.** The reenactment of the baselines is performed using relative transfer between FLAME meshes regressed by EMOCA compared to our GEM regressor network (Ours GEM).

4.2. Cross-person Reenactment Evaluation

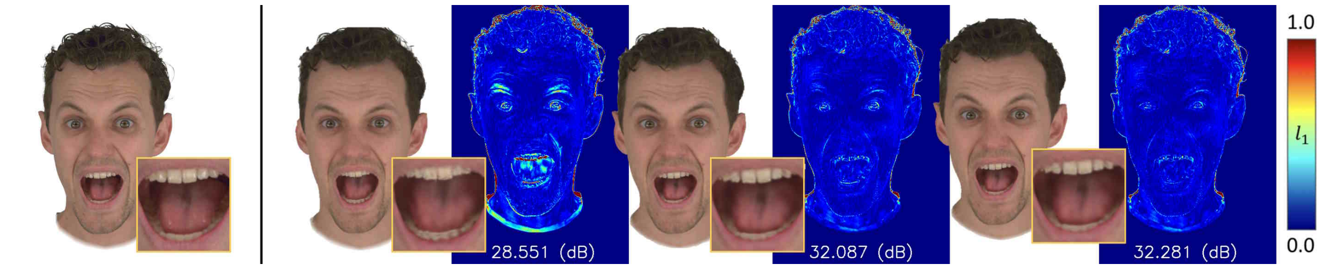
Facial cross-person reenactment transfers expressions from the source actor to the target actor. For this, the baseline methods require tracked meshes obtained by fitting the 3DMM model for each frame of the source actor sequence. As an alternative to optimization-based tracking, a (monocular) regressor like EMOCA [5], can predict such tracked meshes in real-time. We demonstrate this in Figure 7, where GEM is driven by our image-based regressor and the others by EMOCA. As shown, our network-based method and GEM produce sharp results, while the baseline methods struggle to extrapolate to new expressions, displaying severe artifacts in appearance. Our approach effectively regularizes the regressed coefficients, ensuring that the predicted avatar remains in the training distribution and thereby avoids artifacts seen in INSTA or Gaussian Avatars. Drawing inspiration from EMOCA [5], we further assess cross-re-enactment quantitatively by leveraging emotion recognition feature vectors from both the source image and the resulting cross-re-enactment, utilizing EmoNet [52]. For each pair of input and output images, we predict EmoNet features and measure cosine distance and $\mathcal{L}_1$ error between them. We report the numbers in the Table 3. Additionally, we also report FID scores [16] and rendering speed. Our method achieves on-par quality with the CNN-based solution while maintaining the highest frame rates and outperforming GA in terms of quality.

Method	$\mathbf{E}_{feat}\cos\uparrow$	$\mathbf{E}_{feat}\ \mathcal{L}_1\downarrow$	FID $\downarrow$	FPS $\uparrow$
AG	0.9396	5.3399	0.4093	16.51
GA	0.8917	6.6141	0.5593	142.71
INSTA	0.9087	6.3153	0.5299	20.62
Ours Net	0.9440	5.1044	0.3685	35.77
Ours GEM	0.9381	5.3197	0.4286	201.70

Table 3. **Cross-reenactment evaluation** employing EmoNet features and FID score.

4.3. GEM Ablation Studies

We are interested in the compression error introduced by the projection on different amounts of principal components used in GEM, also concerning the memory consumption. Our smallest model weighs as little as **7MB** using only 10 components of the eigenbasis. This is almost **12** times less than our smallest CNN-based model and almost **70** times less than Animatable Gaussians [30]. In contrast to neural networks, we can easily trade quality over size which is very useful in the context of different commodity devices with reduced compute capabilities. Table 4 presents how compression affects the quality of reconstruction, where we evaluate a sequence with $\sim$ 1k frames for a single actor under a novel view. As expected, using only 10 components impacts the quality the most, however, the results are still of high quality, see Figure 8. Gaussian Avatars [43] offers a small size of the stored Gaussians cloud, ranging from 5MB, and 14MB without the FLAME model for 128^2, 256^2 Gaussians, respectively. However, the quality of reconstruction lacks wrinkle details and sharpness as can be seen in Figure 10. In comparison to Gaussian Avatars [43], our model does not require FLAME during inference which is an additional **90MB**.



Ground Truth #10 #30 #50

Figure 8. **Compression error depending on the number of used principle components in GEM.** The heatmaps show the photometric ℓ_1-error for 10, 30, and 50 components using 128^2 Gaussian maps. See suppl. doc. for additional evaluations.

Figure 9. Despite fixed topology and predefined texture size GEM faithfully represents facial attributes like glasses.

Ground Truth Ours #30 128^2 GA [43] 128^2 GA [43] 256^2

Figure 10. The quality comparison to Gaussian Avatars. Note that we do **not** need an additional FLAME model which weighs 90MB.

#Comp	128^2			256^2			512^2		
	PSNR ↑	Size MB	FPS ↑	PSNR ↑	Size MB	FPS ↑	PSNR ↑	Size MB	FPS ↑
10	31.81	7	237.96	31.88	28	210.03	32.23	113	130.46
30	34.20	20	241.31	34.17	83	208.19	34.84	333	112.73
50	34.67	34	238.7	34.61	138	201.70	35.45	553	117.45
Ours Net	33.97	82	47.70	34.99	109	35.77	35.02	178	26.31
AG [30]	33.77	487	18.93	34.40	529	16.51	35.15	636	13.08

Table 4. **Ablation of GEM.** Even with 10 principle components a high PSNR of $31.81dB$ is achieved, while taking only 7MB of memory. In contrast to fixed-sized neural networks, the GEM can be adjusted on the fly depending on the hardware. Moreover, since evaluation requires a single dot product for forward pass the rendering speed is around four times higher than our network. The speed evaluation was done using a single Nvidia A100 GPU.

Figure 9 demonstrates that our method is able to handle different topologies (subject wearing glasses), despite utilizing a fixed UV space.

5. Discussion

We design a universal method capable of distilling 3DGS-based avatar solutions into a lightweight representation, GEM, provided that normalized input across training frames is available. The only requirement to successfully distill GEM is to have a dataset with Gaussian-image pairs across the training sequences. Our results show that a compact representation of the linear basis produces state-of-the-art results in terms of quality and speed. Note that to achieve wrinkle-level details, the generator itself has to produce high-quality outputs. Our distillation technique can be applied to existing methods like [67], making them lightweight and compact. GEM is well-suited for commodity devices, generating Gaussian primitives by a simple linear combination of the basis vectors. This potential has promising implications for tasks like holoportation, audio-driven avatars, and virtual reality.

Limitations: The PCA-based GEM models have a global extent which is useful for some applications, but it also means that we cannot control local changes and produce more combinations of local features. Thus, further work could include incorporating a localized PCA basis [37] for better avatar control, which could potentially enable a wider range of expressions outside the training set. Other limitations are; side-view generalization which results in unstable expressions and personalization. For new subjects a new representation has to be learned from multi-view data. An interesting future avenue is to create a statistical model across subjects.

6. Conclusion

We have proposed Gaussian Eigen Models for Human Heads, a linear appearance model that represents photo-realistic head avatars. The simplicity of this appearance model results in massively reduced compute requirements in comparison to CNN-based avatar methods. Although the idea is simple, it offers many interesting downstream applications. The lightweight representation could improve the management, sharing, and applicability of avatars. Moreover, GEM simplifies the process of online avatar animation from RGB images and increases flexibility by balancing memory and quality trade-offs through additional control over the number of eigenbases. Our distillation approach can be applied to existing methods, making them available for compression. We demonstrate how GEMs can be used in scenarios like self-reenactment and cross-person animation, even in real-time.

Acknowledgements The authors thank the International Max Planck Research School for Intelligent Systems (IMPRS-IS) for supporting WZ. JT was supported by the ERC Starting Grant LeMo (101162081). All the data were processed outside Google.

References

[1] Sizhe An, Hongyi Xu, Yichun Shi, Guoxian Song, Umit Y. Ogras, and Linjie Luo. PanoHead: Geometry-aware 3D full-head synthesis in 360°. *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 20950–20959, 2023. 2

[2] Volker Blanz and Thomas Vetter. A morphable model for the synthesis of 3D faces. In *SIGGRAPH*, pages 187–194, 1999. 2, 4, 5

[3] Chen Cao, Tomas Simon, Jin Kyu Kim, Gabriel Schwartz, Michael Zollhoefer, Shunsuke Saito, Stephen Lombardi, Shih-En Wei, Danielle Belko, Shoou-I Yu, Yaser Sheikh, and Jason M. Saragih. Authentic volumetric avatars from a phone scan. *Transactions on Graphics (TOG)*, 41:1 – 19, 2022. 3

[4] Eric Chan, Connor Z. Lin, Matthew Chan, Koki Nagano, Boxiao Pan, Shalini De Mello, Orazio Gallo, Leonidas J. Guibas, Jonathan Tremblay, S. Khamis, Tero Karras, and Gordon Wetzstein. Efficient geometry-aware 3D generative adversarial networks. *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16102–16112, 2021. 2

[5] Radek Danecek, Michael J. Black, and Timo Bolkart. EMOCA: Emotion driven monocular face capture and animation. *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 20279–20290, 2022. 4, 5, 7

[6] Bernhard Egger, William A. P. Smith, Ayush Tewari, Stefanie Wuhrer, Michael Zollhoefer, Thabo Beeler, Florian Bernard, Timo Bolkart, Adam Kortylewski, Sami Romdhani, Christian Theobalt, Volker Blanz, and Thomas Vetter. 3D morphable face models—past, present, and future. *Transactions on Graphics (TOG)*, 39(5), 2020. 2

[7] Zhiwen Fan, Kevin Wang, Kairun Wen, Zehao Zhu, Dejia Xu, and Zhangyang Wang. Lightgaussian: Unbounded 3d gaussian compression with 15x reduction and 200+ fps, 2023. 3

[8] Yao Feng, Haiwen Feng, Michael J. Black, and Timo Bolkart. Learning an animatable detailed 3D face model from in-the-wild images. *Transactions on Graphics (TOG)*, 40:1 – 13, 2020. 2, 5, 6

[9] Yutao Feng, Xiang Feng, Yintong Shang, Ying Jiang, Chang Yu, Zeshun Zong, Tianjia Shao, Hongzhi Wu, Kun Zhou, Chenfanfu Jiang, and Yin Yang. Gaussian splashing: Dynamic fluid synthesis with gaussian splatting, 2024. 3

[10] Guy Gafni, Justus Thies, Michael Zollhofer, and Matthias Nießner. Dynamic neural radiance fields for monocular 4D facial avatar reconstruction. *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8645–8654, 2020. 2, 3

[11] Xuan Gao, Chenglai Zhong, Jun Xiang, Yang Hong, Yudong Guo, and Juyong Zhang. Reconstructing personalized semantic facial nerf models from monocular video. *Transactions on Graphics (TOG)*, 41:1 – 12, 2022. 2

[12] Simon Giebenhain, Tobias Kirschstein, Markos Georgopoulos, Martin Rünz, Lourdes Agapito, and Matthias Nießner. Mononphm: Dynamic head reconstruction from monoculuar videos. 2024. 5

[13] Simon Giebenhain, Tobias Kirschstein, Martin Rünz, Lourdes Agapito, and Matthias Nießner. Npga: Neural parametric gaussian avatars, 2024. 3

[14] Sharath Girish, Kamal Gupta, and Abhinav Shrivastava. Eagles: Efficient accelerated 3d gaussians with lightweight encodings, 2024. 3

[15] Philip-William Grassal, Malte Prinzler, Titus Leistner, Carsten Rother, Matthias Nießner, and Justus Thies. Neural head avatars from monocular RGB videos. pages 18632–18643, 2022. 2, 3, 5

[16] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium, 2018. 7

[17] Ying Jiang, Chang Yu, Tianyi Xie, Xuan Li, Yutao Feng, Huamin Wang, Minchen Li, Henry Lau, Feng Gao, Yin Yang, and Chenfanfu Jiang. VR-GS: A physical dynamics-aware interactive gaussian splatting system in virtual reality, 2024. 3

[18] Ian T. Jolliffe. Principal component analysis and factor analysis. 1986. 4

[19] Berna Kabadayi, Wojciech Zielonka, Bharat Lal Bhatnagar, Gerard Pons-Moll, and Justus Thies. GAN-Avatar: Controllable personalized gan-based human head avatar. pages 882–892, 2024. 2

[20] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of StyleGAN. *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8107–8116, 2019. 2

[21] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkuehler, and George Drettakis. 3D gaussian splatting for real-time radiance field rendering. *Transactions on Graphics (TOG)*, 42:1 – 14, 2023. 2, 3

[22] Diederik P Kingma and Max Welling. Auto-encoding variational bayes, 2022. 2

[23] Tobias Kirschstein, Shenhan Qian, Simon Giebenhain, Tim Walter, and Matthias Nießner. NeRSemble: Multi-view radiance field reconstruction of human heads. *Transactions on Graphics (TOG)*, 42:1 – 14, 2023. 5

[24] Tobias Kirschstein, Simon Giebenhain, Jiapeng Tang, Markos Georgopoulos, and Matthias Nießner. Gghead: Fast and generalizable 3d gaussian heads, 2024. 3

[25] Christoph Lassner and Michael Zollhöfer. Pulsar: Efficient sphere-based neural rendering. *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1440–1449, 2021. 3

[26] Joo Chan Lee, Daniel Rho, Xiangyu Sun, Jong Hwan Ko, and Eunbyung Park. Compact 3d gaussian splatting for static and dynamic radiance fields. *arXiv preprint arXiv:2408.03822*, 2024. 3

[27] Joo Chan Lee, Daniel Rho, Xiangyu Sun, Jong Hwan Ko, and Eunbyung Park. Compact 3d gaussian representation for radiance field. pages 21719–21728, 2024. 3

[28] Junxuan Li, Shunsuke Saito, Tomas Simon, Stephen Lombardi, Hongdong Li, and Jason M. Saragih. MEGANE: Morphable eyeglass and avatar network. *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12769–12779, 2023. 3

[29] Tianye Li, Timo Bolkart, Michael. J. Black, Hao Li, and Javier Romero. Learning a model of facial shape and expression from 4D scans. *Transactions on Graphics, (Proc. SIGGRAPH Asia)*, 36(6):194:1–194:17, 2017. 2, 3, 4

[30] Zhe Li, Zerong Zheng, Lizhen Wang, and Yebin Liu. Animatable gaussians: Learning pose-dependent gaussian maps for high-fidelity human avatar modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 2, 3, 4, 5, 6, 7, 8

[31] Stephen Lombardi, Tomas Simon, Jason M. Saragih, Gabriel Schwartz, Andreas M. Lehrmann, and Yaser Sheikh. Neural volumes. *Transactions on Graphics (TOG)*, 38:1 – 14, 2019. 2, 3

[32] Stephen Lombardi, Tomas Simon, Gabriel Schwartz, Michael Zollhoefer, Yaser Sheikh, and Jason M. Saragih. Mixture of volumetric primitives for efficient neural rendering. *Transactions on Graphics (TOG)*, 40:1 – 13, 2021. 2

[33] Shugao Ma, Tomas Simon, Jason M. Saragih, Dawei Wang, Yuecheng Li, Fernando De la Torre, and Yaser Sheikh. Pixel codec avatars. pages 64–73, 2021. 4

[34] Shengjie Ma, Yanlin Weng, Tianjia Shao, and Kun Zhou. 3d gaussian blendshapes for head avatar animation. In *ACM SIGGRAPH Conference Proceedings, Denver, CO, United States, July 28 - August 1, 2024*, 2024. 3

[35] 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. In *European Conference on Computer Vision (ECCV)*, pages 405–421, 2020. 2, 5

[36] KL Navaneet, Kossar Pourahmadi Meibodi, Soroush Abbasi Koohpayegani, and Hamed Pirsiavash. Compgs: Smaller and faster gaussian splatting with vector quantization. *ECCV*, 2024. 3

[37] Thomas Neumann, Kiran Varanasi, Stephan Wenger, Markus Wacker, Marcus A. Magnor, and Christian Theobalt. Sparse localized deformation components. *Transactions on Graphics (TOG)*, 32:1 – 10, 2013. 8

[38] Haokai Pang, Heming Zhu, Adam Kortylewski, Christian Theobalt, and Marc Habermann. ASH: animatable gaussian splats for efficient and photoreal human rendering. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1165–1175. IEEE, 2024. 2, 3, 4

[39] Panagiotis Papantonakis, Georgios Kopanas, Bernhard Kerbl, Alexandre Lanvin, and George Drettakis. Reducing the memory footprint of 3d gaussian splatting. *Proceedings of the ACM on Computer Graphics and Interactive Techniques*, 7(1):1–17, 2024. 3

[40] Frederick I. Parke. Computer generated animation of faces. In *Proceedings of the ACM Annual Conference - Volume 1*, page 451–457, New York, NY, USA, 1972. Association for Computing Machinery. 1, 2

[41] P. Paysan, R. Knothe, B. Amberg, S. Romdhani, and T. Vetter. A 3D face model for pose and illumination invariant face recognition. *IEEE International Conference on Advanced Video and Signal based Surveillance (AVSS)*, 2009. 3

[42] Malte Prinzler, Otmar Hilliges, and Justus Thies. DINER: Depth-aware Image-based NEural Radiance fields. *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12449–12459, 2022. 2

[43] Shenhan Qian, Tobias Kirschstein, Liam Schoneveld, Davide Davoli, Simon Giebenhain, and Matthias Nießner. GaussianAvatars: Photorealistic head avatars with rigged 3D gaussians. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 20299–20309, 2024. 2, 3, 5, 6, 7, 8

[44] Edoardo Remelli, Timur M. Bagautdinov, Shunsuke Saito, Chenglei Wu, Tomas Simon, Shih-En Wei, Kaiwen Guo, Zhe Cao, Fabián Prada, Jason M. Saragih, and Yaser Sheikh. Drivable volumetric avatars using texel-aligned features. *SIGGRAPH Conference Papers (SA)*, 2022. 3

[45] Shunsuke Saito, Gabriel Schwartz, Tomas Simon, Junxuan Li, and Giljoo Nam. Relightable gaussian codec avatars. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 130–141, 2024. 2, 3, 4

[46] Robert W. Sumner and Jovan Popović. Deformation transfer for triangle meshes. *SIGGRAPH*, 2004. 4

[47] Kartik Teotia, R. MallikarjunB., Xingang Pan, Hyeon-Joong Kim, Pablo Garrido, Mohamed A. Elgharib, and Christian Theobalt. HQ3DAvatar: High quality controllable 3D head avatar. *Transactions on Graphics (TOG)*, 43(3):27:1–27:24, 2024. 3

[48] A. Tewari, O. Fried, J. Thies, V. Sitzmann, S. Lombardi, K. Sunkavalli, R. Martin-Brualla, T. Simon, J. Saragih, M. Nießner, R. Pandey, S. Fanello, G. Wetzstein, J.-Y. Zhu, C. Theobalt, M. Agrawala, E. Shechtman, D. B. Goldman, and M. Zollhöfer. State of the art on neural rendering. *EG*, 2020. 2

[49] Ayush Tewari, Justus Thies, Ben Mildenhall, Pratul Srinivasan, Edgar Tretschk, Yifan Wang, Christoph Lassner, Vincent Sitzmann, Ricardo Martin-Brualla, Stephen Lombardi, Tomas Simon, Christian Theobalt, Matthias Niessner, Jonathan T. Barron, Gordon Wetzstein, Michael Zollhoefer, and Vladislav Golyanik. Advances in neural rendering. *Computer Graphics Forum (CGF)*, pages 703–735, 2022. 2

[50] Justus Thies, Michael Zollhöfer, Marc Stamminger, Christian Theobalt, and Matthias Nießner. Face2Face: Real-time face capture and reenactment of RGB videos. *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2387–2395, 2016. 2, 5

[51] Justus Thies, Michael Zollhöfer, and Matthias Nießner. Deferred neural rendering. *ACM Transactions on Graphics (TOG)*, 38:1 – 12, 2019. 3

[52] Antoine Toisoul, Jean Kossaifi, Adrian Bulat, Georgios Tzimiropoulos, and Maja Pantic. Estimation of continuous valence and arousal levels from faces in naturalistic conditions. *Nature Machine Intelligence*, 2021. 7

[53] Lizhen Wang, Xiaochen Zhao, Jingxiang Sun, Yuxiang Zhang, Hongwen Zhang, Tao Yu, and Yebin Liu. StyleAvatar: Real-time photo-realistic portrait avatar from a single video. In *SIGGRAPH Conference Papers (SA)*, pages 67:1–67:10, 2023. 2, 3

[54] Zian Wang, Tianchang Shen, Merlin Nimier-David, Nicholas Sharp, Jun Gao, Alexander Keller, Sanja Fidler, Thomas Müller, and Zan Gojcic. Adaptive shells for efficient neural

radiance field rendering. *Transactions on Graphics (TOG)*, 42(6), 2023. 3

[55] Tianyi Xie, Zeshun Zong, Yuxing Qiu, Xuan Li, Yutao Feng, Yin Yang, and Chenfanfu Jiang. PhysGaussian: Physics-integrated 3D gaussians for generative dynamics. pages 4389–4398, 2024. 3

[56] Yuelang Xu, Hongwen Zhang, Lizhen Wang, Xiaochen Zhao, Huang Han, Qi Guojun, and Yebin Liu. LatentAvatar: Learning latent expression code for expressive neural head avatar. In *SIGGRAPH Conference Papers (SA)*, pages 86:1–86:10, 2023. 2

[57] Yuelang Xu, Bengwang Chen, Zhe Li, Hongwen Zhang, Lizhen Wang, Zerong Zheng, and Yebin Liu. Gaussian Head Avatar: Ultra high-fidelity head avatar via dynamic gaussians. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1931–1941. IEEE, 2024. 2, 3, 4

[58] Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. pages 586–595, 2018. 6

[59] Xiaochen Zhao, Lizhen Wang, Jingxiang Sun, Hongwen Zhang, Jinli Suo, and Yebin Liu. HAvatar: High-fidelity head avatar via facial model conditioned neural radiance field. *Transactions on Graphics (TOG)*, 2023. 2

[60] Shunyuan Zheng, Boyao Zhou, Ruizhi Shao, Boning Liu, Shengping Zhang, Liqiang Nie, and Yebin Liu. GPS-gaussian: Generalizable pixel-wise 3D gaussian splatting for real-time human novel view synthesis. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19680–19690. IEEE, 2024. 2

[61] Shunyuan Zheng, Boyao Zhou, Ruizhi Shao, Boning Liu, Shengping Zhang, Liqiang Nie, and Yebin Liu. GPS-Gaussian: Generalizable pixel-wise 3d gaussian splatting for real-time human novel view synthesis. pages 19680–19690, 2024. 3

[62] Yufeng Zheng, Victoria Fernández Abrevaya, Xu Chen, Marcel C. Buhler, Michael J. Black, and Otmar Hilliges. I M Avatar: Implicit morphable head avatars from videos. *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13535–13545, 2021. 2

[63] Yufeng Zheng, Yifan Wang, Gordon Wetzstein, Michael J. Black, and Otmar Hilliges. PointAvatar: Deformable point-based head avatars from videos. *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 21057–21067, 2022.

[64] Wojciech Zielonka, Timo Bolkart, and Justus Thies. Instant volumetric head avatars. *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4574–4584, 2022. 2, 3, 5, 6, 7

[65] Wojciech Zielonka, Timo Bolkart, and Justus Thies. Towards metrical reconstruction of human faces. In *European Conference on Computer Vision (ECCV)*, 2022. 5

[66] Wojciech Zielonka, Timur Bagautdinov, Shunsuke Saito, Michael Zollhöfer, Justus Thies, and Javier Romero. Drivable 3D gaussian avatars. In *International Conference on 3D Vision (3DV)*, 2025. 2, 3

[67] Wojciech Zielonka, Stephan J. Garbin, Alexandros Lattas, George Kopanas, Paulo Gotardo, Thabo Beeler, Justus Thies, and Timo Bolkart. Synthetic prior for few-shot drivable head avatar inversion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025. 3, 8

[68] Michael Zollhöfer, Justus Thies, Darek Bradley, Pablo Garrido, Thabo Beeler, Patrick Péerez, Marc Stamminger, Matthias Nießner, and Christian Theobalt. State of the art on monocular 3D face reconstruction, tracking, and applications. *Computer Graphics Forum (CGF)*, 37(2):523–550, 2018. 2