

Avat3r: Large Animatable Gaussian Reconstruction Model for High-fidelity 3D Head Avatars

Tobias Kirschstein^{1,2*} Javier Romero² Artem Sevastopolsky^{1,2*} Matthias Nießner¹ Shunsuke Saito²
Technical University of Munich¹ Meta Reality Labs Pittsburgh²

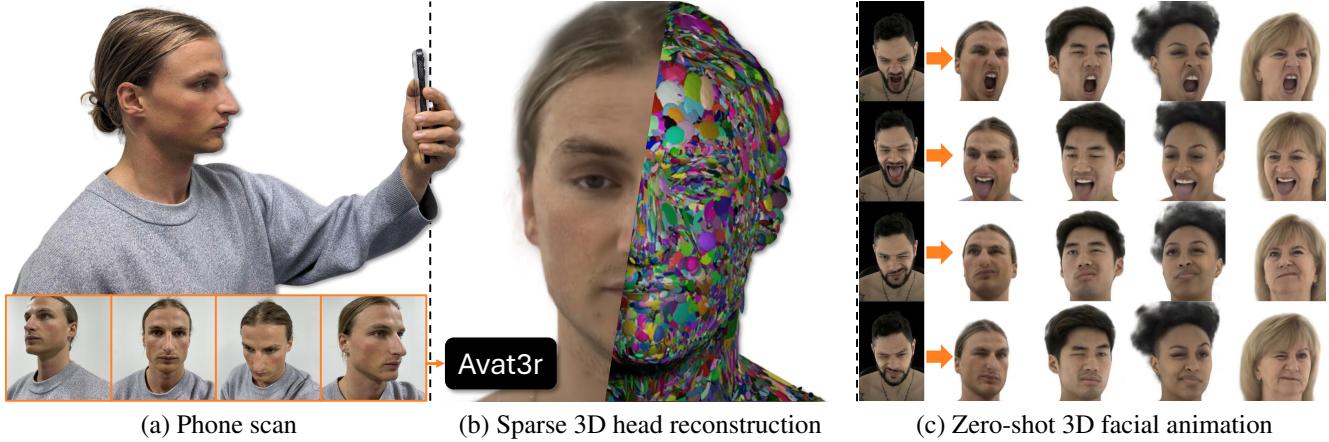


Figure 1. **Avat3r.** Given just four images of a person’s head (a), Avat3r achieves two things: (b) it creates a faithful 3D reconstruction of the head in a feed-forward manner, and (c) it allows facial animation without having seen any of the corresponding expressions of the person. This simplifies the capturing process as it drops the need for recording long sequences of facial movement. As a result, the entire pipeline from head scan to final 3D head avatar can be executed within a few minutes and runs on a single consumer-grade GPU.

Abstract

Traditionally, creating photo-realistic 3D head avatars requires a studio-level multi-view capture setup and expensive optimization during test-time, limiting the use of digital human doubles to the VFX industry or offline renderings. To address this shortcoming, we present Avat3r, which regresses a high-quality and animatable 3D head avatar from just a few input images, vastly reducing compute requirements during inference. More specifically, we make Large Reconstruction Models animatable and learn a powerful prior over 3D human heads from a large multi-view video dataset. For better 3D head reconstructions, we employ position maps from DUSt3R and generalized feature maps from the human foundation model Sapiens. To animate the 3D head, our key discovery is that simple cross-attention to an expression code is already sufficient. Finally, we increase robustness by feeding input images with different expressions to our model during training, enabling the

reconstruction of 3D head avatars from inconsistent inputs, e.g., an imperfect phone capture with accidental movement, or frames from a monocular video. We compare Avat3r with current state-of-the-art methods for few-input and single-input scenarios, and find that our method has a competitive advantage in both tasks. Finally, we demonstrate the wide applicability of our proposed model, creating 3D head avatars from images of different sources, smartphone captures, single images, and even out-of-domain inputs like antique busts.

1. Introduction

Lifelike 3D head avatars find use in a variety of applications such as telepresence, movie production, personalized games, or digital education. Typically, the creation of a high-quality avatar requires studio-quality videos that have to be recorded in hour-long sessions in expensive capture setups. This limits the applicability of a 3D head avatar creation pipeline, especially in casual settings where users may only want to use a few images taken with their smartphone to create a convincing digital double. In this paper,

*Work done while Tobias Kirschstein and Artem Sevastopolsky were interns at Reality Labs Research, Pittsburgh, PA, USA

Project website: <https://tobias-kirschstein.github.io/avat3r/>

we present Avat3r, a method that can reconstruct an animatable 3D head avatar from just a few images of a person.

Building a photo-realistic 3D head avatar from a few casual images involves several hard challenges. First, sparse 3D reconstruction poses a heavily under-constrained optimization problem since some regions such as the mouth interior or the side of the head may have simply not been seen in the available images. Next, animating the head requires plausibly morphing the reconstructed face without ever having seen that particular person talking or sticking their tongue out. Finally, the input images may be inconsistent, e.g., because the person could not remain completely static during a casual capture. All of these three challenges, i.e., sparse 3D reconstruction, face animation, and robust reconstruction, have to be solved simultaneously to successfully reconstruct a photo-realistic 3D head avatar given just a few images.

Due to the aforementioned challenges, the current best methods for high-quality avatar creation [16, 26, 37, 39] cannot be applied since their optimization-based approach requires many more observations than just a few images of a person. On the other hand, methods that aim at 3D-aware portrait animation [7, 8, 14, 43] learn a prior over 3D heads from large video datasets that include a variety of different people. As a result, existing methods can already convincingly animate faces from as few as just a single input image. However, such methods typically focus on frontal facial appearance and may sacrifice 3D-consistency or temporal consistency to reach satisfying image synthesis quality. This makes these methods less applicable in scenarios where a proper 3D model is required. Another approach is to build a photo-realistic 3D head model that learns the distribution of 3D face geometry, expression, and appearance, akin to classic 3DMMs. While such unconditional generative models are very versatile and recent works have shown promising results [50, 58], they typically suffer from limited generalizability along the identity axis due to existing 3D face datasets only providing a few hundred different persons [25, 34, 36, 51].

Our approach is motivated by the observation that 3D data for faces is limited in the identity axis, but not in the expression axis that is already well captured in existing datasets. We therefore design a system that is *conditional* on the identity of a person, but *generalizes* along the expression axis. Opposed to photo-realistic 3D face models, the network does not have to learn the full extent of human facial appearance but just needs to understand how to reconstruct it from a few example images, which simplifies the task. Inspired by recent Large Reconstruction Models [40, 49, 56], we devise an architecture that predicts 3D Gaussians for each pixel in the input images. As such, we deliberately avoid anchoring the 3D Gaussians on a template mesh, as common in the 3D Head Avatar do-

main [3, 37, 39]. The reason is that human facial appearance is complex and diverse. Limiting the 3D representation to a fixed number of Gaussians anchored on a template mesh with uniform topology will cause issues in regions such as hair or face accessories. Instead, when predicting Gaussians for each foreground pixel, a person with voluminous hair will receive more primitives to model their head than a bald person. To further simplify the task of sparse 3D reconstruction, we first compute position maps for each input image via DUS3R [46]. These position maps are injected into the network via skip connections and act as a coarse starting position for each Gaussian. We surprisingly find that DUS3R is still capable of producing reasonable position maps even if the input images are not consistent, e.g., because they show different facial expressions or are the result of a view-inconsistent image generation model. We exploit this remarkable capability of DUS3R to train our model on input images taken at different timesteps, which not only enables training and inference on monocular video datasets but also makes the model more robust to inconsistencies in the input images. Such inconsistencies can quickly occur in a real-life scenario where a person may not stay completely still during a phone capture of their head. Finally, to model the animation of the face, we find that simple cross-attention between intermediate feature maps and a descriptive expression code are already sufficient to generalize over the space of facial expressions. Taken together, our contributions are as follows:

- We design Avat3r, a novel pipeline for creating high-quality 3D head avatars from just a few images in a matter of minutes.
- We build the first animatable large 3D reconstruction model and show that simple cross-attention on expression token sequences is sufficient to model complex facial animations.
- To improve generalization and robustness, we integrate priors from Dust3r and Sapiens via skip connections, and feed input images from different timesteps to the network during training.

2. Related Work

2.1. Large 3D Reconstruction Models

Recently, generalized feed-forward models for 3D reconstruction from sparse input views have garnered considerable attention due to their applicability in heavily under-constrained scenarios. The Large Reconstruction Model (LRM) [18] uses a transformer-based encoder-decoder pipeline to infer a NeRF reconstruction from just a single image. Newer iterations have shifted the focus towards generating 3D Gaussian representations from four input images [5, 6, 31, 40, 49, 56], showing remarkable novel view synthesis results. The paradigm of transformer-based sparse

3D reconstruction has also successfully been applied to lifting monocular videos to 4D [38].

Yet, none of the existing works in the domain have studied the use-case of inferring *animatable* 3D representations from sparse input images, which is the focus of our work. To this end, we build on top of the Large Gaussian Reconstruction Model (GRM) [49].

2.2. 3D-aware Portrait Animation

A different line of work focuses on animating portraits in a 3D-aware manner. MegaPortraits [14] builds a 3D Volume given a source and driving image, and renders the animated source actor via orthographic projection with subsequent 2D neural rendering. 3D morphable models (3DMMs) [2] are extensively used to obtain more interpretable control over the portrait animation. For example, StyleRig [41] demonstrates how a 3DMM can be used to control the data generated from a pre-trained StyleGAN [20] network. ROME [22] predicts vertex offsets and texture of a FLAME [29] mesh from the input image. A TriPlane representation is inferred and animated via FLAME [29] in multiple methods like Portrait4D [11], Portrait4D-v2 [12], and GPAvatar [8]. Others, such as VOODOO 3D [43] and VOODOO XP [42], learn their own expression encoder to drive the source person in a more detailed manner.

All of the aforementioned methods require nothing more than a single image of a person to animate it. This allows them to train on large monocular video datasets to infer a very generic motion prior that even translates to paintings or cartoon characters. However, due to their task formulation, these methods mostly focus on image synthesis from a frontal camera, often trading 3D consistency for better image quality by using 2D screen-space neural renderers. In contrast, our work aims to produce a truthful and complete 3D avatar representation from the input images that can be viewed from any angle.

2.3. Photo-realistic 3D Face Models

The increasing availability of large-scale multi-view face datasets [25, 34, 36, 51] has enabled building photo-realistic 3D face models that learn a detailed prior over both geometry and appearance of human faces. HeadNeRF [17] conditions a Neural Radiance Field (NeRF) [35] on identity, expression, albedo, and illumination codes. VRMM [52] builds a high-quality and relightable 3D face model using volumetric primitives [32]. One2Avatar [54] extends a 3DMM by anchoring a radiance field to its surface. More recently, GPHM [50] and HeadGAP [58] have adopted 3D Gaussians to build a photo-realistic 3D face model.

Photo-realistic 3D face models learn a powerful prior over human facial appearance and geometry, which can be fitted to a single or multiple images of a person, effectively inferring a 3D head avatar. However, the fitting procedure itself

is non-trivial and often requires expensive test-time optimization, impeding casual use-cases on consumer-grade devices. While this limitation may be circumvented by learning a generalized encoder that maps images into the 3D face model’s latent space, another fundamental limitation remains. Even with more multi-view face datasets being published, the number of available training subjects rarely exceeds the thousands, making it hard to truly learn the full distribution of human facial appearance. Instead, our approach avoids generalizing over the identity axis by conditioning on some images of a person, and only generalizes over the expression axis for which plenty of data is available.

A similar motivation has inspired recent work on codec avatars where a generalized network infers an animatable 3D representation given a registered mesh of a person [3, 28]. The resulting avatars exhibit excellent quality at the cost of several minutes of video capture per subject and expensive test-time optimization. For example, URAvatar [28] finetunes their network on the given video recording for 3 hours on 8 A100 GPUs, making inference on consumer-grade devices impossible. In contrast, our approach directly regresses the final 3D head avatar from just four input images without the need for expensive test-time fine-tuning.

3. Method

The goal of Avat3r is to predict a high-quality 3D Gaussian representation of a human head from just a few posed input images while simultaneously animating the 3D head to match a desired facial expression:

$$\mathcal{G} = \text{AVAT3R}(I, \pi, z_{exp}) \quad (1)$$

where $I = \{I_1, \dots, I_V\}$ are the V input images, π their corresponding camera parameters, and z_{exp} a code describing the desired facial expression. In our experiments, $V = 4$. Note that the images I do not necessarily have to stem from multi-view camera systems or synthetic multi-view renderings of objects as it is common for large reconstruction models. Instead, the input images may show the scene at different instances in time, making the model amenable for training and inference on frames from monocular videos.

Under the hood, Avat3r employs a Vision Transformer backbone [13] to predict 3D Gaussian primitives for each pixel in the input images. Fig. 2 depicts the model architecture. In the following, we introduce the different components of our pipeline.

3.1. Pre-trained Foundation Models

We simplify the task by employing two foundation models: DUST3R [46], and Sapiens [23]. DUST3R predicts dense

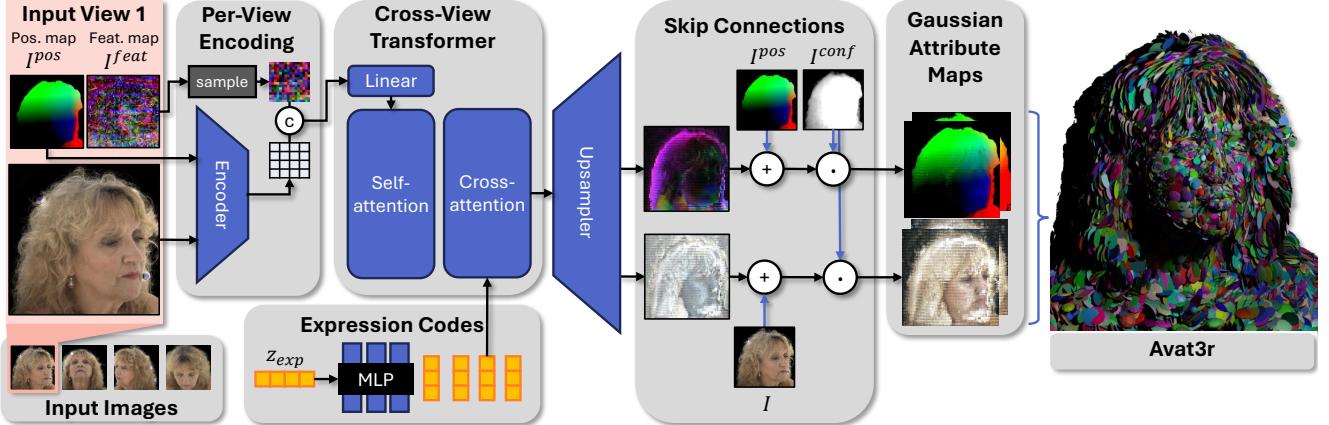


Figure 2. Method Overview: Avat3r reconstructs a high-quality 3D representation from just a few input images by predicting 3D Gaussian attributes for each input pixel. We first obtain position maps I^{pos} from DUST3R and feature maps I^{feat} from Sapiens for each view. These are then patchified to serve as image tokens for the Vision Transformer back-end. Dense Self-attention performs matching within tokens of the same image and across views to infer 3D structure. Dynamics are modelled via cross-attention layers that attend to a sequenced expression code. The resulting intermediate feature maps are decoded into Gaussian position, scale, rotation, color, and opacity maps, and then upsampled to the original input image resolution. Finally, we add skip connections to the predicted position and color maps. Gaussians that belong to pixels with a confidence larger than a pre-defined threshold are collected and can be rendered from arbitrary viewpoints.

position maps for a set of input images and globally aligns them with the known camera parameters π . These position maps will be beneficial for placing the 3D Gaussians. The Sapiens backbone predicts a rich low-resolution feature map for each image. The semantic nature of the extracted features will simplify the view matching task that has to be performed by the subsequent Transformer.

$$I^{pos}, I^{conf} = \text{DUST3R}(I, \pi) \quad (2)$$

$$I^{feat} = \text{SAPIENS}(I) \quad (3)$$

where I^{pos}, I^{conf} are position and confidence maps predicted by DUST3R, and I^{feat} are the Sapiens feature maps.

3.2. Animatable Large 3D Reconstruction Model

We adopt a Vision Transformer architecture following GRM [49] by first patchifying the input maps $I, I^{pos}, I^{pluck} \in \mathbb{R}^{V \times H \times W \times C}$ with a convolutional layer:

$$h = \text{CONV}([I, I^{pos}, I^{pluck}]) \quad (4)$$

where I^{pluck} are the per-ray plucker coordinates stored for each pixel and $h \in \mathbb{R}^{V \times H_p \times W_p \times D}$ will be the low-resolution intermediate feature maps for each input view. Next, we incorporate the pre-computed per-image feature maps $I^{feat} \in \mathbb{R}^{V \times H_f \times W_f \times C_f}$:

$$h^{feat} = \text{GRIDSAMPLE}(I^{feat}) \quad (5)$$

$$h \leftarrow \text{LINEAR}([h, h^{feat}]) \quad (6)$$

where the grid sampling step is necessary to transform the Sapiens features maps I^{feat} to the same $H_p \times W_p$ resolution

as the intermediate feature maps h . The subsequent linear layer then squashes the concatenated feature maps back to dimension D to avoid excessive compute increase in the following transformer layers.

The core of the Vision Transformer consists of several self-attention layers that perform cross-view matching between all $V \times H_p \times W_p$ tokens to infer 3D information from the given input images:

$$h \leftarrow \text{SELFATT}(h, h) \quad (7)$$

To make Avat3r animatable, we employ several cross-attention layers that allow each image token to attend to a sequence of expression tokens following [26, 53]:

$$f_{exp} = \text{MLP}(z_{exp}) \quad (8)$$

$$h \leftarrow \text{CROSSATT}(h, f_{exp}) \quad (9)$$

where the MLP projects a single descriptive expression code $z_{exp} \in \mathbb{R}^{C_{exp}}$ to the expression code sequence $f_{exp} \in \mathbb{R}^{S \times D}$. In our case, $S = 4$. Surprisingly, by simply enriching each image token in h with expression-dependent information, Avat3r is already capable of modeling complex facial animations in 3D.

We upsample the image tokens again to the original input resolution using the transformer-based upsampler from GRM [49]:

$$M = \text{UPSAMPLE}(h) \quad (10)$$

where $M \in \mathbb{R}^{V \times H \times W \times C_G}$ denote the Gaussian attribute maps consisting of per-pixel Gaussian positions, scales, rotations, colors, and opacities. After upsampling, we add two

skip connections for positions and colors:

$$M^{pos} \leftarrow M^{pos} + I^{pos} \quad (11)$$

$$M^{rgb} \leftarrow M^{rgb} + I \quad (12)$$

which incorporate the inductive bias that a pixel should primarily spawn a 3D Gaussian that models the pixel’s corresponding 3D geometry.

Finally, we make use of DUST3R’s per-pixel confidence maps I^{conf} to decide which pixel in the generated Gaussian attribute maps M should actually spawn a 3D Gaussian:

$$\mathcal{G} = \{M[x, y] : I^{conf}[x, y] > \tau\} \quad (13)$$

where $\tau = 0.5$ is a threshold describing the minimal confidence needed to create a 3D Gaussian. This serves two purposes: First, it avoids artifacts that arise from incorrect 3D position initializations from DUST3R. Second, it naturally adjusts the number of Gaussians to the person at hand. In that sense, the confidence mask acts much like a foreground segmentation: If the person had large voluminous hair, then more pixels will belong to the foreground and thus more Gaussians will be kept.

The final set of 3D Gaussians \mathcal{G} can then be rendered from any desired novel viewpoint π^{nv} using the tile-based differentiable rasterizer \mathcal{R} [21]:

$$I^{nv} = \mathcal{R}(\mathcal{G}, \pi^{nv}) \quad (14)$$

3.3. Losses

Avat3r can be trained with only photometric losses on novel viewpoints following the original 3D Gaussian Splatting work [21]:

$$\mathcal{L}_{l1} = \|I^{nv} - I^{gt}\|_1 \quad (15)$$

$$\mathcal{L}_{ssim} = \text{SSIM}(I^{nv}, I^{gt}) \quad (16)$$

Furthermore, we incorporate perceptual losses [57] to encourage the emergence of more high-frequency details:

$$\mathcal{L}_{lpips} = \text{LPIPS}(I^{nv}, I^{gt}) \quad (17)$$

The final loss term is comprised as follows:

$$\mathcal{L} = \lambda_{l1} \mathcal{L}_{l1} + \lambda_{ssim} \mathcal{L}_{ssim} + \lambda_{lpips} \mathcal{L}_{lpips} \quad (18)$$

with $\lambda_{l1} = 0.8$, $\lambda_{ssim} = 0.2$, and $\lambda_{lpips} = 0.01$.

4. Experimental Results

4.1. Training

We train Avat3r on multi-view video captures from the Ava256 dataset [34]. In particular, we use the 4TB version of the dataset which contains 7.5fps videos from 80 cameras (roughly 5000 frames each) at a resolution of 1024×667 .

Avat3r is trained on 244 persons from the dataset with the remaining 12 being held out. We use the provided tracked mesh to crop each image to a head-centric 512×512 portrait which will be fed as input to our model. For each training sample, we randomly select 4 reasonably diverse viewpoints and 4 timesteps with diverse expressions as input images I . On the other hand, the supervision images I^{gt} are left at the highest resolution and are randomly sampled over all available cameras and timesteps. For z_{exp} , we use the expression codes that are provided with the Ava256 dataset.

Avat3r is trained with Adam [24] using a learning rate of $5e - 5$. We only introduce \mathcal{L}_{lpips} after 3M optimization steps to avoid focusing on high-frequency details too early. In total, the model is trained for 3.5M steps with a batch size of 1 per GPU on 8 A100 GPUs which takes roughly 4 days. We further find that training convergence is significantly improved when supervising multiple viewpoints for each batch. I.e., a batch consists of 12 images: 4 with random expression that are used as input, and 8 viewpoints from the target expression that are used as supervision.

4.2. Experiment Setup

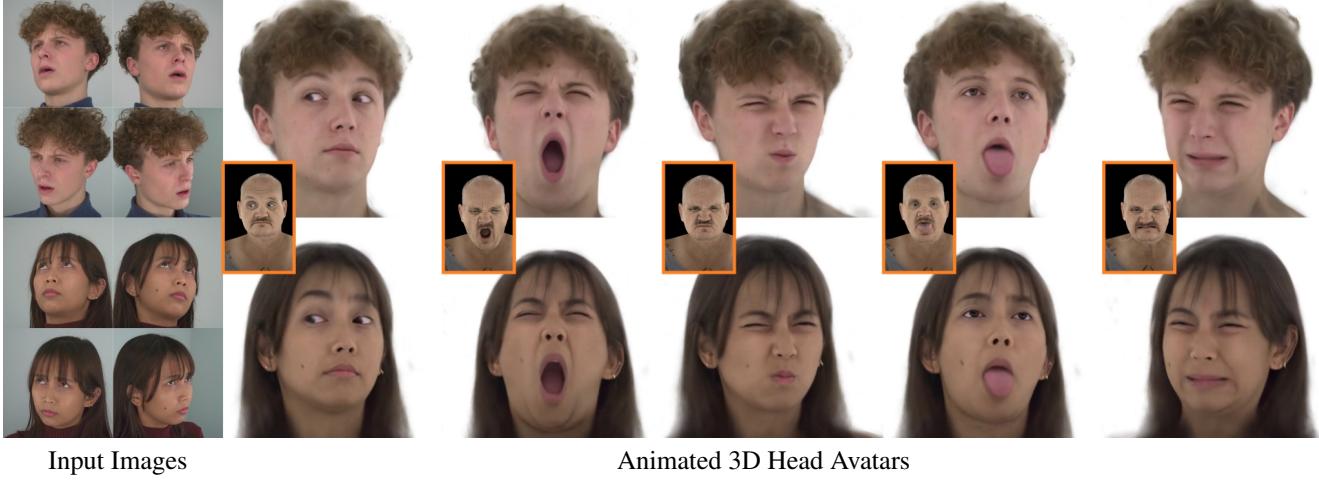
Task. We evaluate the ability to create a 3D head avatar from one or multiple images of an unseen person.

Metrics. We employ three paired-image metrics to measure the quality of individual rendered images: Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index (SSIM) [47], and Learned Perceptual Image Patch Similarity (LPIPS) [57]. Furthermore, we measure visual similarity of a rendered video to its ground truth counterpart with the perceptual video metric JOD [33]. Finally, we make use of two face-specific metrics: Average Keypoint Distance (AKD) measured in pixels with keypoints estimated from PIPNet [19], and cosine similarity (CSIM) of identity embeddings based on ArcFace [10].

Baselines. We compare our method with recent state-of-the-art systems for 3D head avatar creation from one and few input images.

GPAvatar [8]. A NeRF-based method that predicts canonical TriPlanes from multiple input images. Animations are then modeled by querying 3D points in an expression-dependent feature field inferred from FLAME.

GAGAvatar [7]. A method for 3D-aware portrait animation from a single image that constructs two sets of 3D Gaussians: The first set is obtained by predicting 2 Gaussians for each pixel in the input image with a CNN, the second set is decoded from FLAME vertices and automatically animated through the FLAME mesh.



Input Images

Animated 3D Head Avatars

Figure 3. **Performance on NeRSembla dataset.** We show reconstructed and animated avatars using 4 images from the NeRSembla dataset. Note that this dataset was not used during training and contains images with different lighting conditions, viewpoints, and camera intrinsics than the Ava-256 dataset that was used to train Avat3r.

	Image		Video		Face		
	PSNR↑	SSIM↑	LPIPS↓	JOD↑	AKD↓	CSIM↑	
4-in	GPAvatar	20.3	0.703	0.304	4.62	6.08	0.277
	Ours	22.0	0.732	0.309	5.20	5.93	0.595

Table 1. **3D Head Avatar Creation Comparison.** Our method outperforms GPAvatar [8] in almost all metrics.

4.3. Few-shot 3D Head Avatar Creation

We compare the ability to create a realistic 3D head avatar from just four input images with GPAvatar [8]. To this end, we train GPAvatar on the Ava256 dataset in the same setting as our model and use the high-quality tracked meshes provided with the dataset to drive GPAvatar. We evaluate both methods at 512×512 pixels resolution on the same set of 12 hold-out persons, sampling in total 1000 different target expressions from different viewpoints. For the video-based metric JOD, we render one video of 100 frames from the frontal camera for each hold-out person. The quantitative results are listed in Tab. 1. It can be seen that our method clearly outperforms the baseline. We observe the biggest difference in metrics measuring alignment (PSNR, SSIM), temporal consistency (JOD), and identity similarity (CSIM). This indicates that Avat3r produces 3D head avatars that better resemble the person in the input images and produces smoother video renderings since it does not require a screen-space super-resolution network as GPAvatar. We further show qualitative comparisons in Fig. 4. Our method produces more expressive avatars with higher rendering quality than the baseline and even performs well on a person with colored hair who wears a headband.

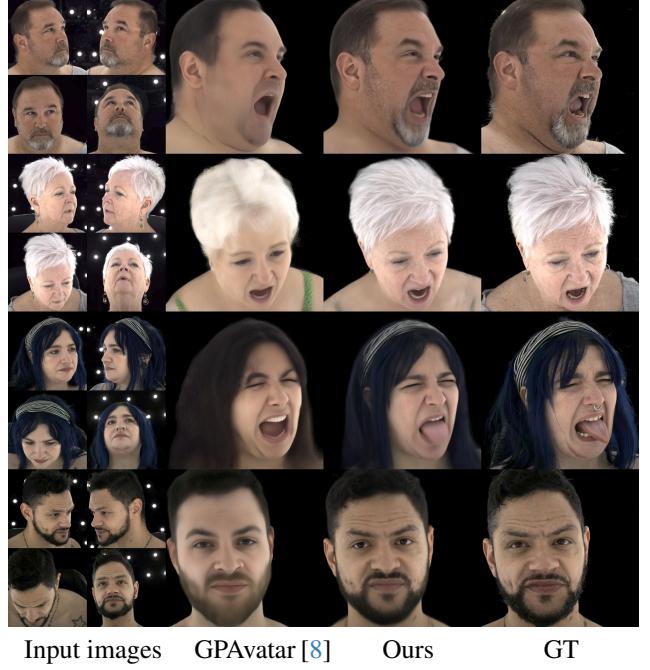


Figure 4. **Comparison for few-shot 3D Head Avatar creation.** We compare the ability of Avat3r to create 3D head avatars from four input images against GPAvatar [8]. Our method generalizes better and produces higher quality avatars.

To further assess Avat3r’s generalization capabilities, we create few-shot 3D head avatars for subjects from the multi-view video dataset NeRSembla [25] which has not been used during training. The results can be seen in Fig. 3. Note that the images in the NeRSembla dataset exhibit entirely different characteristics in terms of lighting, camera param-

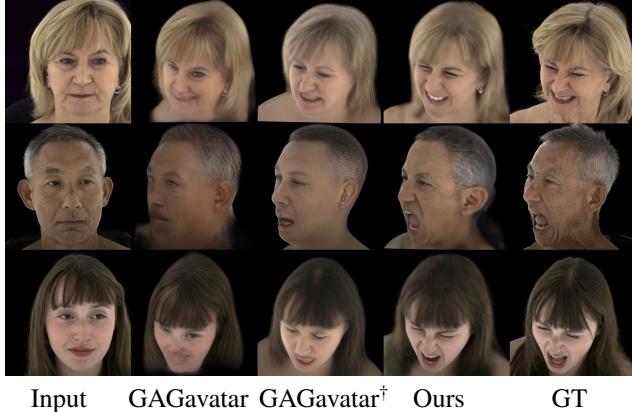


Figure 5. Comparison for single-shot 3D Head Avatar creation. We compare Avat3r with the recent 3D-aware portrait animation methods GAGAvatar [7] in a self-reenactment scenario on hold-out persons from the Ava256 dataset. GAGAvatar[†] denotes a version of the baseline that we trained on the Ava256 dataset. Our method shows better rendering quality especially for extreme expressions and viewing angles.

eters, and viewpoints, compared to the training images from the Ava256 dataset. Nevertheless, Avat3r generates realistic 3D head avatars in this setting as well.

4.4. Single-shot 3D Head Avatar Creation

To make Avat3r amenable for inference on only a single input image, we make use of a pre-trained 3D GAN [55] to first lift the single image to 3D and then render four views of the head. These renderings then constitute the input for Avat3r. We conduct comparisons with the recent 3D-aware portrait animation method GAGAvatar [7]. Specifically, we compare with two versions of GAGAvatar: One provided by the authors which is trained on VFHQ [48], and another version, denoted as GAGAvatar[†], that we train on the Ava256 dataset in the same setting as our method. To drive GAGAvatar, we use their monocular FLAME tracker to obtain tracked meshes. Fig. 5 shows qualitative comparisons between our method and GAGAvatar for single input images of hold out persons. Note that our method performs competitively compared to the single-input baselines despite never being trained for a single-shot scenario. We find that Avat3r produces more realistic facial expressions than GAGAvatar which is limited by FLAME’s expression space. Furthermore, our method allows much more extreme viewpoint changes without sacrificing rendering quality.

4.5. Ablations

In the following, we study the efficacy of our design choices for Avat3r. In particular, we look at the usefulness of DUST3R’s position maps, the Sapiens feature maps, and using input images with different expressions during training.

	\mathcal{S}	\mathcal{D}	\mathcal{T}	PSNR↑	LPIPS↓	AKD↓
Dynamic GRM	□	□	☒	20.8	0.439	8.60
w/o Sapiens	□	☒	☒	20.9	0.434	8.59
w/o Dust3r	☒	□	☒	21.1	0.429	9.60
w/o rand. timesteps	☒	☒	□	21.3	0.409	8.86
Avat3r	☒	☒	☒	21.6	0.410	8.08

Table 2. Ablation. We analyze the effect of Sapiens features (\mathcal{S}), DUST3R position maps (\mathcal{D}), and training on randomized input timesteps (\mathcal{T}). All ablation models are trained without LPIPS loss. Metrics are computed on 667×667 renderings.

Our findings are summarized in Tab. 2 and Fig. 6.

Effect of DUST3R position maps. Removing DUST3R from our pipeline mostly impairs geometric fidelity. In Fig. 6, it can be seen that Avat3r without DUST3R struggles with aligning the Gaussian predictions from the four input images.

Effect of Sapiens feature maps. Sapiens provides rich semantic information for the input images that mostly help in producing sharper predictions, e.g., in the hair area.

Effect of training with inconsistent input images. A model that is trained on only multi-view consistent input images produces slightly sharper images. However, any inconsistencies in the input images, that can easily happen in practice, carry over to the generated 3D representation causing unwanted artifacts.

4.6. Applications

Finally, we show some applications of Avat3r in Fig. 7. We study three use-cases: (i) Generating a novel 3D head with GGHead [27] and then animating it with Avat3r, (ii) Generating a portrait image with a text-to-image diffusion model [9], lifting it with a 3D GAN [55], and animating it with Avat3r, and (iii) Lifting an existing photograph with a 3D GAN [55] and animating it with Avat3r. The results demonstrate that, despite being designed for a few-shot scenario and only being trained on real studio data, Avat3r can be used in a variety of casual situations. For example, our findings for case (i) suggest that Avat3r may be used to animate images from any 3D GAN that generates 3D heads, effectively making any purely static 3D generation model animatable.

4.7. Limitations

Despite Avat3r’s great ability to create 3D head avatars from a few or even just a single input image, several limitations remain. For example, the current pipeline for in-

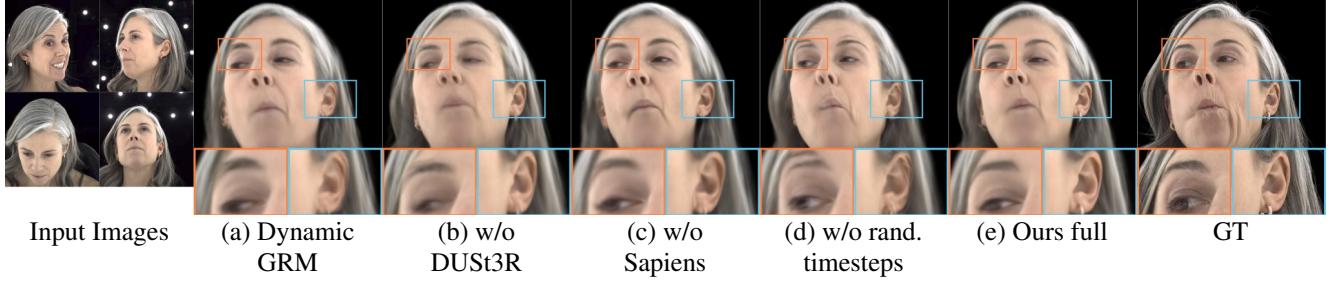


Figure 6. **Ablation study.** Adding Sapiens features (b) noticeably improves sharpness but still exhibits misalignments. Conversely, using position maps from Dust3r improves alignment but is overall less sharp (c). Combining both but training on multi-view images from the same timestep produces sharp results but is less robust to inconsistencies in the input (d). Our final model exhibits the best trade-off between alignment, sharpness, and robustness (e).

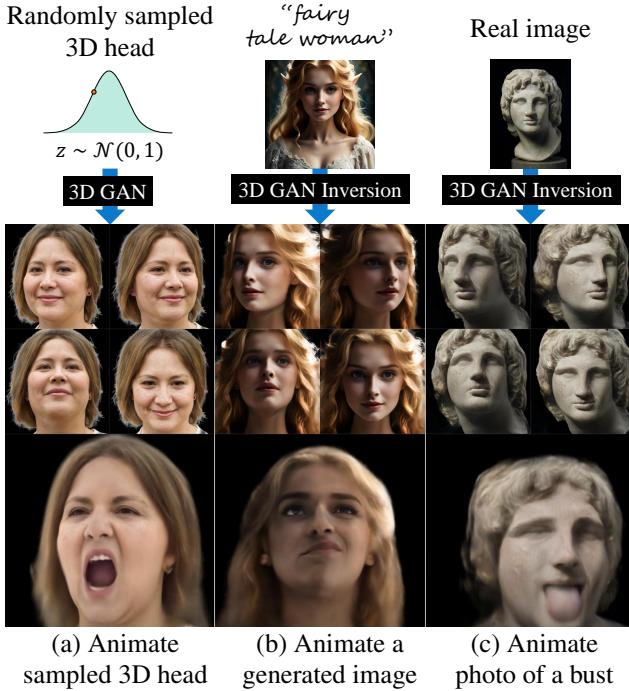


Figure 7. **Application scenarios of Avat3r.** We show-case various use-cases of our model such as animating a randomly sampled 3D head, animating an image generated by a text-to-image diffusion model, and animating a picture of a greek bust.

ferring avatars from a single image requires a 3D GAN for 3D lifting which accumulates errors in the process in two ways: First, the 3D GAN won’t be able to perfectly explain the person, inevitably loosing some details, and second, existing inversion frameworks employ NeRF-based 3D GANs that rely on screen-space super-resolution after rendering which introduces view-inconsistencies to the final images. Both aspects hamper the quality of the final avatar. However, these issues will decrease as newer and better 3D GANs become available that do not require screen-space rendering anymore [27, 44]. Another limitation of our

framework is the requirement of camera poses during inference. While reasonable estimates for face images can be obtained by head tracking algorithms [59], incorrect camera estimates still remain a potential source of error for real use-cases. Possible remedies may include teaching the network to deal with incorrect camera estimates already during training, or dropping the requirement for camera poses altogether [45]. Finally, our current pipeline does not provide control over the lighting and instead bakes in light effects from the input images. This limits applications that put avatars into novel environments since the generated head will look out of place. Adressing this issue by explicitly disentangling the lighting properties is an exciting avenue for future research.

5. Conclusion

We have presented Avat3r, a system for directly regressing an animatable 3D head avatar from just four input images. Our pipeline combines recent advancements in large reconstruction models with the powerful foundation models DUST3R and Sapiens to obtain high quality results, and models dynamics with simple cross attention layers. Moreover, we demonstrate that Avat3r can also be employed in a single image scenario by using a pre-trained 3D GAN. Our experiments show that Avat3r learns a powerful prior over 3D facial movement that even generalizes to out-of-distribution examples like pictures of antique busts or AI-generated images. We believe that our findings open up interesting avenues for future research. For example, we have shown that LRM-like architectures have the ability to infer 3D representations even from inconsistent input images such as frames of a monocular video. This opens up the possibility to train models on large-scale monocular video datasets with a much larger variety of identities. Finally, the underlying architecture could also be used as a 3D lifting and denoising module in a diffusion framework [1, 4], enabling training of diffusion models on monocular video data for unconditional 3D head avatar generation.

Acknowledgements

This work was supported by the ERC Starting Grant Scan2CAD (804724) and the German Research Foundation (DFG) Research Unit “Learning and Simulation in Visual Computing”. We would also like to thank Angela Dai for the video voice-over.

References

- [1] Titas Auciukevičius, Zexiang Xu, Matthew Fisher, Paul Henderson, Hakan Bilen, Niloy J Mitra, and Paul Guerrero. Rennderdiffusion: Image diffusion for 3d reconstruction, inpainting and generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12608–12618, 2023. [8](#)
- [2] V Blanz and T Vetter. A morphable model for the synthesis of 3d faces. In *26th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH 1999)*, pages 187–194. ACM Press, 1999. [3](#)
- [3] Chen Cao, Tomas Simon, Jin Kyu Kim, Gabriel Schwartz, Michael Zollhoefer, Shun-Suke Saito, Stephen Lombardi, Shih-En Wei, Danielle Belko, Shouo-I Yu, et al. Authentic volumetric avatars from a phone scan. 2022. [2](#), [3](#)
- [4] Eric R. Chan, Koki Nagano, Matthew A. Chan, Alexander W. Bergman, Jeong Joon Park, Axel Levy, Miika Aittala, Shalini De Mello, Tero Karras, and Gordon Wetzstein. GeNVS: Generative novel view synthesis with 3D-aware diffusion models. In *arXiv*, 2023. [8](#)
- [5] David Charatan, Sizhe Lester Li, Andrea Tagliasacchi, and Vincent Sitzmann. pixelsplat: 3d gaussian splats from image pairs for scalable generalizable 3d reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19457–19467, 2024. [2](#)
- [6] Yuedong Chen, Haofei Xu, Chuanxia Zheng, Bohan Zhuang, Marc Pollefeys, Andreas Geiger, Tat-Jen Cham, and Jianfei Cai. Mvsplat: Efficient 3d gaussian splatting from sparse multi-view images. In *European Conference on Computer Vision*, pages 370–386. Springer, 2025. [2](#)
- [7] Xiangeng Chu and Tatsuya Harada. Generalizable and animatable gaussian head avatar. *arXiv preprint arXiv:2410.07971*, 2024. [2](#), [5](#), [7](#)
- [8] Xiangeng Chu, Yu Li, Ailing Zeng, Tianyu Yang, Lijian Lin, Yunfei Liu, and Tatsuya Harada. Gpavatar: Generalizable and precise head avatar from image (s). *arXiv preprint arXiv:2401.10215*, 2024. [2](#), [3](#), [5](#), [6](#)
- [9] Xiaoliang Dai, Ji Hou, Chih-Yao Ma, Sam Tsai, Jialiang Wang, Rui Wang, Peizhao Zhang, Simon Vandenhende, Xiaofang Wang, Abhimanyu Dubey, et al. Emu: Enhancing image generation models using photogenic needles in a haystack. *arXiv preprint arXiv:2309.15807*, 2023. [7](#)
- [10] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4690–4699, 2019. [5](#)
- [11] Yu Deng, Duomin Wang, Xiaohang Ren, Xingyu Chen, and Baoyuan Wang. Portrait4d: Learning one-shot 4d head avatar synthesis using synthetic data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7119–7130, 2024. [3](#)
- [12] Yu Deng, Duomin Wang, and Baoyuan Wang. Portrait4d-v2: Pseudo multi-view data creates better 4d head synthesizer. *arXiv preprint arXiv:2403.13570*, 2024. [3](#)
- [13] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *ICLR*, 2021. [3](#)
- [14] Nikita Drobyshev, Jenya Chelishev, Taras Khakhulin, Alekssei Ivakhnenko, Victor Lempitsky, and Egor Zakharov. Megaportraits: One-shot megapixel neural head avatars. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 2663–2671, 2022. [2](#), [3](#)
- [15] Sankeerth Durvasula, Adrian Zhao, Fan Chen, Ruofan Liang, Pawan Kumar Sanjaya, and Nandita Vijaykumar. Distwar: Fast differentiable rendering on raster-based rendering pipelines. *arXiv preprint arXiv:2401.05345*, 2023. [2](#)
- [16] Simon Giebenhain, Tobias Kirschstein, Martin Rünz, Lourdes Agapito, and Matthias Nießner. Npga: Neural parametric gaussian avatars. *arXiv preprint arXiv:2405.19331*, 2024. [2](#)
- [17] Yang Hong, Bo Peng, Haiyao Xiao, Ligang Liu, and Juyong Zhang. Headnerf: A real-time nerf-based parametric head model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20374–20384, 2022. [3](#)
- [18] Yicong Hong, Kai Zhang, Jiuxiang Gu, Sai Bi, Yang Zhou, Difan Liu, Feng Liu, Kalyan Sunkavalli, Trung Bui, and Hao Tan. Lrm: Large reconstruction model for single image to 3d. *arXiv preprint arXiv:2311.04400*, 2023. [2](#)
- [19] Haibo Jin, Shengcai Liao, and Ling Shao. Pixel-in-pixel net: Towards efficient facial landmark detection in the wild. *International Journal of Computer Vision*, 2021. [5](#)
- [20] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410, 2019. [3](#)
- [21] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):139–1, 2023. [5](#)
- [22] Taras Khakhulin, Vanessa Sklyarova, Victor Lempitsky, and Egor Zakharov. Realistic one-shot mesh-based head avatars. In *European Conference on Computer Vision*, pages 345–362. Springer, 2022. [3](#)
- [23] Rawal Khirodkar, Timur Bagautdinov, Julieta Martinez, Su Zhaoen, Austin James, Peter Selednik, Stuart Anderson, and Shunsuke Saito. Sapiens: Foundation for human vision models. In *European Conference on Computer Vision*, pages 206–228. Springer, 2025. [3](#), [2](#)
- [24] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. [5](#)

- [25] Tobias Kirschstein, Shenhan Qian, Simon Giebenhain, Tim Walter, and Matthias Nießner. Nerensemble: Multi-view radiance field reconstruction of human heads. *ACM Transactions on Graphics (TOG)*, 42(4):1–14, 2023. 2, 3, 6
- [26] Tobias Kirschstein, Simon Giebenhain, and Matthias Nießner. Diffusionavatars: Deferred diffusion for high-fidelity 3d head avatars. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5481–5492, 2024. 2, 4
- [27] Tobias Kirschstein, Simon Giebenhain, Jiapeng Tang, Markos Georgopoulos, and Matthias Nießner. Gghead: Fast and generalizable 3d gaussian heads. *arXiv preprint arXiv:2406.09377*, 2024. 7, 8
- [28] Junxuan Li, Chen Cao, Gabriel Schwartz, Rawal Khirodkar, Christian Richardt, Tomas Simon, Yaser Sheikh, and Shunsuke Saito. Uravatar: Universal relightable gaussian codec avatars. *arXiv preprint arXiv:2410.24223*, 2024. 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. *ACM Trans. Graph.*, 36(6):194–1, 2017. 3
- [30] Shanchuan Lin, Andrey Ryabtsev, Soumyadip Sen-gupta, Brian Curless, Steve Seitz, and Ira Kemelmacher-Shlizerman. Real-time high-resolution background matting. *arXiv*, pages arXiv–2012, 2020. 1
- [31] Tianqi Liu, Guangcong Wang, Shoukang Hu, Liao Shen, Xinyi Ye, Yuhang Zang, Zhiguo Cao, Wei Li, and Ziwei Liu. Mvsgaussian: Fast generalizable gaussian splatting reconstruction from multi-view stereo. In *European Conference on Computer Vision*, pages 37–53. Springer, 2025. 2
- [32] Stephen Lombardi, Tomas Simon, Gabriel Schwartz, Michael Zollhoefer, Yaser Sheikh, and Jason Saragih. Mixture of volumetric primitives for efficient neural rendering. *ACM Transactions on Graphics (ToG)*, 40(4):1–13, 2021. 3
- [33] Rafał K. Mantiuk, Gyorgy Denes, Alexandre Chapiro, Anton Kaplanyan, Gizem Rufo, Romain Bachy, Trisha Lian, and Anjul Patney. Fovvideodp: A visible difference predictor for wide field-of-view video. *ACM Trans. Graph.*, 40(4), 2021. 5
- [34] Julieta Martinez, Emily Kim, Javier Romero, Timur Bagautdinov, Shunsuke Saito, Shou-I Yu, Stuart Anderson, Michael Zollhöfer, Te-Li Wang, Shaojie Bai, Chenghui Li, Shih-En Wei, Rohan Joshi, Wyatt Borsos, Tomas Simon, Jason Saragih, Paul Theodosis, Alexander Greene, Anjani Josyula, Silvio Mano Maeta, Andrew I. Jewett, Simon Venshtain, Christopher Heilman, Yueh-Tung Chen, Sidi Fu, Mohamed Ezzeldin A. Elshaer, Tingfang Du, Longhua Wu, Shen-Chi Chen, Kai Kang, Michael Wu, Youssef Emad, Steven Longay, Ashley Brewer, Hitesh Shah, James Booth, Taylor Koska, Kayla Hidle, Matt Andromalos, Joanna Hsu, Thomas Dauer, Peter Selednik, Tim Godisart, Scott Ardisson, Matthew Cipperly, Ben Humberston, Lon Farr, Bob Hansen, Peihong Guo, Dave Braun, Steven Krenn, He Wen, Lucas Evans, Natalia Fadeeva, Matthew Stewart, Gabriel Schwartz, Divam Gupta, Gyeongsik Moon, Kaiwen Guo, Yuan Dong, Yichen Xu, Takaaki Shiratori, Fabian Prada, Bernardo R. Pires, Bo Peng, Julia Buffalini, Autumn Trimble, Kevyn McPhail, Melissa Schoeller, and Yaser Sheikh. Codec Avatar Studio: Paired Human Captures for Complete, Driveable, and Generalizable Avatars. *NeurIPS Track on Datasets and Benchmarks*, 2024. 2, 3, 5, 1
- [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. *Communications of the ACM*, 65(1):99–106, 2021. 3
- [36] Dongwei Pan, Long Zhuo, Jingtan Piao, Huiwen Luo, Wei Cheng, Yuxin Wang, Siming Fan, Shengqi Liu, Lei Yang, Bo Dai, et al. Renderme-360: a large digital asset library and benchmarks towards high-fidelity head avatars. *Advances in Neural Information Processing Systems*, 36, 2024. 2, 3
- [37] Shenhan Qian, Tobias Kirschstein, Liam Schoneveld, Davide Davoli, Simon Giebenhain, and Matthias Nießner. Gaussianavatars: Photorealistic head avatars with rigged 3d gaussians. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20299–20309, 2024. 2
- [38] Jiawei Ren, Kevin Xie, Ashkan Mirzaei, Hanxue Liang, Xiaohui Zeng, Karsten Kreis, Ziwei Liu, Antonio Torralba, Sanja Fidler, Seung Wook Kim, et al. L4gm: Large 4d gaussian reconstruction model. *arXiv preprint arXiv:2406.10324*, 2024. 3
- [39] Shunsuke Saito, Gabriel Schwartz, Tomas Simon, Junxuan Li, and Giljoo Nam. Relightable gaussian codec avatars. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 130–141, 2024. 2
- [40] Jiaxiang Tang, Zhaoxi Chen, Xiaokang Chen, Tengfei Wang, Gang Zeng, and Ziwei Liu. Lgm: Large multi-view gaussian model for high-resolution 3d content creation. In *European Conference on Computer Vision*, pages 1–18. Springer, 2025. 2
- [41] Ayush Tewari, Mohamed Elgarib, Gaurav Bharaj, Florian Bernard, Hans-Peter Seidel, Patrick Pérez, Michael Zollhofer, and Christian Theobalt. Stylerig: Rigging stylegan for 3d control over portrait images. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6142–6151, 2020. 3
- [42] Phong Tran, Egor Zakharov, Long-Nhat Ho, Liwen Hu, Adilbek Karmanov, Aviral Agarwal, McLean Goldwhite, Ariana Bermudez Venegas, Anh Tuan Tran, and Hao Li. Voodoo xp: Expressive one-shot head reenactment for vr telepresence. *arXiv preprint arXiv:2405.16204*, 2024. 3
- [43] Phong Tran, Egor Zakharov, Long-Nhat Ho, Anh Tuan Tran, Liwen Hu, and Hao Li. Voodoo 3d: Volumetric portrait disentanglement for one-shot 3d head reenactment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10336–10348, 2024. 2, 3
- [44] Alex Trevithick, Matthew Chan, Towaki Takikawa, Umar Iqbal, Shalini De Mello, Manmohan Chandraker, Ravi Ramamoorthi, and Koki Nagano and. Rendering every pixel for high-fidelity geometry in 3d gans. In *arXiv*, 2023. 8
- [45] Peng Wang, Hao Tan, Sai Bi, Yinghao Xu, Fujun Luan, Kalyan Sunkavalli, Wenping Wang, Zexiang Xu, and Kai Zhang. Pf-lrm: Pose-free large reconstruction model for joint pose and shape prediction. *arXiv preprint arXiv:2311.12024*, 2023. 8

- [46] Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20697–20709, 2024. 2, 3
- [47] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004. 5
- [48] Liangbin Xie, Xintao Wang, Honglun Zhang, Chao Dong, and Ying Shan. Vfhq: A high-quality dataset and benchmark for video face super-resolution. In *The IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2022. 7
- [49] Yinghao Xu, Zifan Shi, Wang Yifan, Hansheng Chen, Ceyuan Yang, Sida Peng, Yujun Shen, and Gordon Wetzstein. Grm: Large gaussian reconstruction model for efficient 3d reconstruction and generation. *arXiv preprint arXiv:2403.14621*, 2024. 2, 3, 4
- [50] Yuelang Xu, Lizhen Wang, Zerong Zheng, Zhaoqi Su, and Yebin Liu. 3d gaussian parametric head model. In *European Conference on Computer Vision*, pages 129–147. Springer, 2025. 2, 3
- [51] Haotian Yang, Hao Zhu, Yanru Wang, Mingkai Huang, Qiu Shen, Ruigang Yang, and Xun Cao. Facescape: a large-scale high quality 3d face dataset and detailed riggable 3d face prediction. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, pages 601–610, 2020. 2, 3
- [52] Haotian Yang, Mingwu Zheng, Chongyang Ma, Yu-Kun Lai, Pengfei Wan, and Haibin Huang. Vrmm: A volumetric relightable morphable head model. In *ACM SIGGRAPH 2024 Conference Papers*, pages 1–11, 2024. 3
- [53] Hu Ye, Jun Zhang, Sibo Liu, Xiao Han, and Wei Yang. Ip-adapter: Text compatible image prompt adapter for text-to-image diffusion models. *arXiv preprint arXiv:2308.06721*, 2023. 4
- [54] Zhixuan Yu, Ziqian Bai, Abhimitra Meka, Feitong Tan, Qiangeng Xu, Rohit Pandey, Sean Fanello, Hyun Soo Park, and Yinda Zhang. One2avatar: Generative implicit head avatar for few-shot user adaptation. *arXiv preprint arXiv:2402.11909*, 2024. 3
- [55] Ziyang Yuan, Yiming Zhu, Yu Li, Hongyu Liu, and Chun Yuan. Make encoder great again in 3d gan inversion through geometry and occlusion-aware encoding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2437–2447, 2023. 7
- [56] Kai Zhang, Sai Bi, Hao Tan, Yuanbo Xiangli, Nanxuan Zhao, Kalyan Sunkavalli, and Zexiang Xu. Gs-lrm: Large reconstruction model for 3d gaussian splatting. In *European Conference on Computer Vision*, pages 1–19. Springer, 2025. 2
- [57] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018. 5
- [58] Xiaozheng Zheng, Chao Wen, Zhaohu Li, Weiyi Zhang, Zhuo Su, Xu Chang, Yang Zhao, Zheng Lv, Xiaoyuan Zhang, Yongjie Zhang, et al. Headgap: Few-shot 3d head avatar via generalizable gaussian priors. *arXiv preprint arXiv:2408.06019*, 2024. 2, 3
- [59] Wojciech Zienonka, Timo Bolkart, and Justus Thies. Towards metrical reconstruction of human faces. In *European conference on computer vision*, pages 250–269. Springer, 2022. 8

Avat3r: Large Animatable Gaussian Reconstruction Model for High-fidelity 3D Head Avatars

Supplementary Material

A. Analysis of Data Efficiency

In Fig. 8 we show how our model scales with the number of training subjects and input views. For the latter, we disable DUSt3R as it produces less reliable position maps for 2 input views, and cannot be executed at all for 1 input view. We see a clear improvement when using more training subjects as well as using more input views. However, further scaling the number of input views also has drawbacks, as it drastically increases runtime due to dense attention inside the transformer and the increased number of Gaussians that have to be rendered. We qualitatively analyze the effect of using more train subjects in Fig. 9 and the effect of the number of input views in Fig. 10.

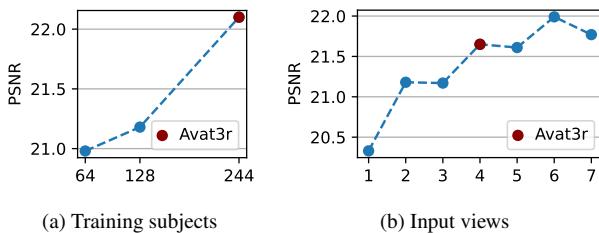


Figure 8. **Analysis of Data Efficiency.** We study how reconstruction and animation performance behaves when changing the number of training subjects and input views.



Figure 9. **Effect of Number of Train Subjects.** Training on a larger and more diverse set of people enhances the Avat3r’s generalization capabilities, as expected. This leads to more accurate reconstructions, with avatars better matching the identities shown in the input images. For instance, when dealing with complex hairstyles, a model trained on a broader range of individuals reproduces the hairstyle more accurately. All ablations are trained without LPIPS loss.

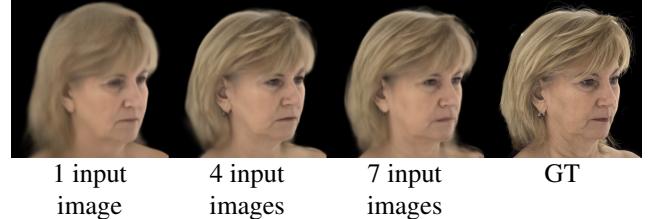


Figure 10. **Effect of Number of Input Views.** Training with just a single input image noticeably impairs quality. On the other hand, using more than 4 input images during training does not lead to significant improvements. Models are trained without DUSt3R position maps and without LPIPS loss in the interest of comparability.

B. Effect of Skip Connections

We analyze the effect of the proposed skip connections, i.e., omitting Eq. (11), Eq. (12), or both of the main paper. The results are listed in Tab. 3. We observe a noticeable hit in performance when either skip connection is removed. Furthermore, we qualitatively analyze the effect of skip connections in Fig. 11.

	\mathcal{C}	\mathcal{P}	PSNR↑	SSIM↑	LPIPS↓	JOD↑	AKD↓	CSIM↑
No skip	<input type="checkbox"/>	<input type="checkbox"/>	21.39	0.740	0.456	4.99	9.24	0.60
No pos. skip	<input checked="" type="checkbox"/>	<input type="checkbox"/>	21.76	0.746	0.443	5.03	9.04	0.611
No col. skip	<input type="checkbox"/>	<input checked="" type="checkbox"/>	21.55	0.745	0.435	5.00	7.69	0.648
Avat3r	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	22.05	0.751	0.421	5.15	7.99	0.689

Table 3. **Quantitative Ablation of Skip Connections.** We analyze the effect of the color (\mathcal{C}) and position (\mathcal{P}) skip connections. All ablation models are trained without LPIPS loss. Metrics are computed on 667×667 renderings.

C. Training Details

Dataset processing. We use the 4TB version of the Ava256 dataset [34] which contains 256 persons, 80 cameras, and roughly 5000 frames per person that are sampled at 7.5 fps. We compute foreground segmentation masks with BackgroundMattingV2 [30] and replace the background in all images with black pixels. We use the provided tracked mesh to find a 512×512 head-centered square crop for input images and 667×667 head-centered square crop for supervision views. This ensures that the pixels in the input images are used efficiently to show as much



Figure 11. **Qualitative Ablation of Skip Connections.** Not employing skip connections (a) causes misalignments, blurry renderings, and a slight color shift. Adding the color skip connection (b) already noticeably improves sharpness and color fidelity. On the other hand, if the position skip connection is added (c), geometric details are improved but the overall color slightly off. Using both skip connections (d) yields the best result.

as possible of the head, leading to more 3D Gaussians. The reason for also cropping the target images is to remove parts of the torso, as it is not the focus of this work.

DUST3R and Sapiens. Since both DUST3R [46] and Sapiens [23] are expensive foundation models, we pre-compute the position and feature maps for the input frames. For Dust3r, it is prohibitive to pre-compute all possible combinations of 4 input views out of the available 80 cameras. Instead, we choose 3 “reasonable partner views” for each input and only store the position map for that viewpoint. This assigns each input view exactly one position map, which is conceptually wrong since the position map from DUST3R should depend on the other 3 selected views. Nevertheless, we did not observe any disadvantages from this simplification strategy.

Head-centric coordinates. We further simplify the task by factoring the head poses from the provided tracked mesh into the camera poses instead of letting the network predict them. That way, our model can always predict the head in canonical pose, making the task easier. This is possible because modeling the torso, which in head-centric coordinates moves a lot when the person shakes their head, is not the focus of this work.

k-farthest viewpoint sampling. To ensure that the 4 input images always follow a reasonable viewpoint distribution, we employ k-farthest viewpoint sampling. Specifically, we first start from a random camera and collect a set of 10 candidate cameras that are evenly spread out using farthest point sampling. From this candidate set, we then randomly select 4 cameras as input. This two-stage approach ensures that the input cameras are sufficiently random during training but also reasonably spread out to avoid seeing a person only from one side. During sampling input

viewpoints, we exclude cameras that only observe the person from the back since those are not realistic inputs during test-time.

Input timestep sampling. To improve robustness of our model, we sample different timesteps for each of the 4 input images. This ensures that the model can deal with inconsistencies in the input. To maximize the diversity in the input expressions, we uniformly sample 10 timesteps in the segments: EXP_eye_wide, EXP_tongue001, and EXP_jaw003 from the recordings of the Ava256 dataset. This covers the most extreme facial expressions while avoiding having to pre-compute DUST3R and Sapiens maps for every single image in the dataset.

Speed. To speed-up training, we employ the 3D Gaussian Splatting performance improvements of DISTWAR [15].

D. Hyperparameters

In Tab. 4, we list the most important hyperparameters for training Avat3r.

	Hyperparameter	Value
ViT & GRM	ViT patch size	8×8
	hidden dimension D	768
	#self-attention layers	8
	#cross-attention layers	8
	#GRM transformer upsampler step	1
Input & Output	Sapiens version	2b
	Sapiens feature dimension	1920
	Input image resolution	512×512
	Gaussian attribute map resolution	512×512
	Train render resolution	667×667
Expression MLP	Dimension of expression code	256
	#expression sequence MLP layers	2
	Dimension of expression sequence MLP	256
	Expression sequence MLP activation	ReLU

Table 4. **Hyperparameters.**