GaussianStyle: Gaussian Head Avatar via StyleGAN

Pinxin Liu[1] Luchuan Song[1*] Daoan Zhang[1] Yunlong Tang[1]
Hang Hua[1] Huaijin Tu[2] Jiebo Luo[1] Chenliang Xu[1]
[1]University of Rochester [2]Georgia Institute of Technology
{pliu23, lsong11, daoan.zhang}@ur.rochester.edu
{yunlong.tang, hhua2@cs., jluo@cs., chenliang.xu@}rochester.edu
htu35@gatech.edu

Figure 1. We present **GaussianStyle**, a novel method designed for high-fidelity volumetric avatar reconstruction from a short monocular video. Our pipeline can be utilized for portrait reenactment, high-fidelity editing, and novel view synthesis.

Abstract

Existing methods like Neural Radiation Fields (NeRF) and 3D Gaussian Splatting (3DGS) have made significant strides in facial attribute control such as facial animation and components editing, yet they struggle with fine-grained representation and scalability in dynamic head modeling. To address these limitations, we propose GaussianStyle, a novel framework that integrates the volumetric strengths of 3DGS with the powerful implicit representation of StyleGAN. The GaussianStyle preserves structural information, such as expressions and poses, using Gaussian points, while projecting the implicit volumetric representation into StyleGAN to capture high-frequency details and mitigate the over-smoothing commonly observed in neural texture rendering. Experimental outcomes indicate that our method achieves state-of-the-art performance in reenactment, novel view synthesis, and animation.

1. Introduction

Learning head avatars from a given monocular video has become popular in recent years. It aims to achieve diversity control in terms of facial expression and head pose. Many works incorporate NeRF [7, 9, 36, 56, 58] and 3DGS [44, 48] into head avatar training via tracked parametric facial template. Generally, those methods are maintain a relatively canonical feature space by the implicit topology (for NeRF) or explicit topology (for 3DGS), and enable the queried voxel or Gaussian points to learn the neural texture features from the movement of head in video.

Though this strategy improve the movement stability, it overlooks a critical challenge inherent within dynamic 3D head modeling: the assumption that a fixed 3D coordinate in the canonical space will always correspond to the same facial region throughout the entire sequence. In reality, as the head motion and dynamic expressions, the relative positions of facial features will shift significantly. For example, a point that initially corresponds to the corner of the mouth in a canonical expression will shift toward the cheek when smiles or lip motions. This movement causes the fixed coordinate canonical template to misalign with the actual facial regions it is supposed to observe.

This limitation manifests as over-smoothing in dynamic head and facial movement rendering scenarios. Since the fixed 3D coordinates do not accurately track the evolving geometry of the face, the model tends to produce averaged or blurred features rather than sharp and precise details.

[1]corresponding author.

This over-smoothing effect is particularly pronounced in areas where there is a high degree of motion or expression variation, resulting in a loss of the fine-grained details necessary for realistic and expressive head avatars. This problem is further exacerbated during the cross-reenactment scenarios when the motion is conditioned on novel expression, pose, or camera perspectives.

Drawing inspiration from the deferred neural rendering [42] (DNR), which first samples the noised UV space features and then leverages the neural network to translate the texture space to pixel space, we believe this coarse-to-fine strategy has the potential to address the over-smoothing issue. However, this UV-projected texture (or called neural texture) is difficult to extend to 3D space, leading to limitations in novel-view synthesis and flexible control.

To address these challenges, we propose **GaussianStyle**, a novel framework that integrates dynamic neural rendering with 3DGS. By leveraging the powerful implicit representation of StyleGAN, GaussianStyle improves the fine-grained texture quality based on the volumetric representation provided by 3DGS. Specifically, we first propose a more robust dynamic Gaussian representation. Inspired by the Triplane-Gaussian [35], we construct a temporal-aware Tri-plane as an implicit and low-dimensional Gaussian representation. This design allows for more effective 4D Gaussian modeling by leveraging cross-attention to learn the correspondence between Gaussian points and motion-control parameters. To extend StyleGAN on the 3DGS features, we introduce a multi-view PTI initialization that minimizes disruptions to pre-trained StyleGAN parameters while personalizing the rendering for the target avatar. Additionally, we propose an optimal method for projecting Gaussian features into the StyleGAN architecture, informed by a comprehensive analysis of its structure.

We validate the efficacy of our framework through both quantitative and qualitative experiments on self/cross-reenactment. Our contributions are summarized as follows:

- We present **GaussianStyle**, a framework that integrates 3DGS with StyleGAN representations. This integration enhances the controllability of head pose, facial expression, and fine-grained facial details, enabling high-quality volumetric avatar generation from monocular videos.
- We refine the hybrid triplane-Gaussian representation by introducing a temporal-aware design and an attention-based deformation module. This improves the deformability of Gaussian points, leading to more robust and accurate 3D face rendering.
- We design a pipeline that effectively maps dynamic 3D representations to the latent space of StyleGAN for volumetric rendering. This approach requires training only a small number of parameters, achieving an flexibility editable neural representation with inference speeds exceeding 30 FPS while maintaining high fidelity.

2. Related Work

Video Portrait Animation. Mainstream approaches for facial reconstruction and animation primarily relied on 3D Morphable Models(3DMM) [5] or relying on implicit neural representations [10, 34, 37, 38, 56]. IMAvatar [56] and INSTA [58] shifted towards using implicit geometry to overcome the limitations from mesh templates. The point-cloud based models combine explicit point clouds with neural networks' implicit representations to enhance image quality [57]. Recent works [27, 44] have shifted the direction towards 3DGS for head modeling, aiming to leverage the benefits of rapid training and rendering while still achieving competitive levels of photorealism. GaussianAvatars [27] reconstructed head avatars through rigging 3D Gaussians on FLAME [19] template. MonoGaussianAvatar [4] learned explicit head avatars by deforming 3D Gaussians from canonical space with Linear Blend Skinning (LBS) and simultaneously. GaussianHead [44] adopted a motion deformation field to adapt to facial movements while preserving head geometry. FlashAvatar [47] initializes 3D Gaussians based on the UV coordinates and learns the deformation offset conditioned on tracking parameters. However, none of these methods considers the dynamic coordinate change of Gaussian points and thus cannot present a robust performance towards novel poses and camera views.

StyleGAN-based Portrait Editing and Rendering Portrait animation and rendering have drawn considerable views recently [32, 35, 38, 41, 54]. With diverse style distribution, StyleGAN significantly promotes the capabilities of facial editing [8, 29, 31, 49]. DeformToon3D [53] further extends geometry-aware 3D editing. Portrait rendering techniques also benefited from StyleGAN. StyleHEAT [52] optimizes latent codes through inversion and leverages audio features for motion, further refined by OTAvatar [22] that applies EG3D to perform geometry-aware rendering. Next3D [40] and IDE-3D [39] disentangle semantics and geometry for 3D-aware controlled avatar rendering. However, these works mainly focus on aligned faces and are not applicable to avatars with torso.

3. Method

As depicted in Fig. 2, our framework combines Gaussian with StyleGAN [14] for Volume Rendering. StyleGAN's high-quality generation capability and style control proficiency make it suitable for our objectives. We first present a temporal-aware hybrid triplane-gaussian representation with attention-based deformation to achieve robust pose and expression control. (Sec. 3.1) To counteract the oversmooth problems, we explored an effective strategy of mapping Gaussian representation into StyleGAN's latent (Sec. 3.2).

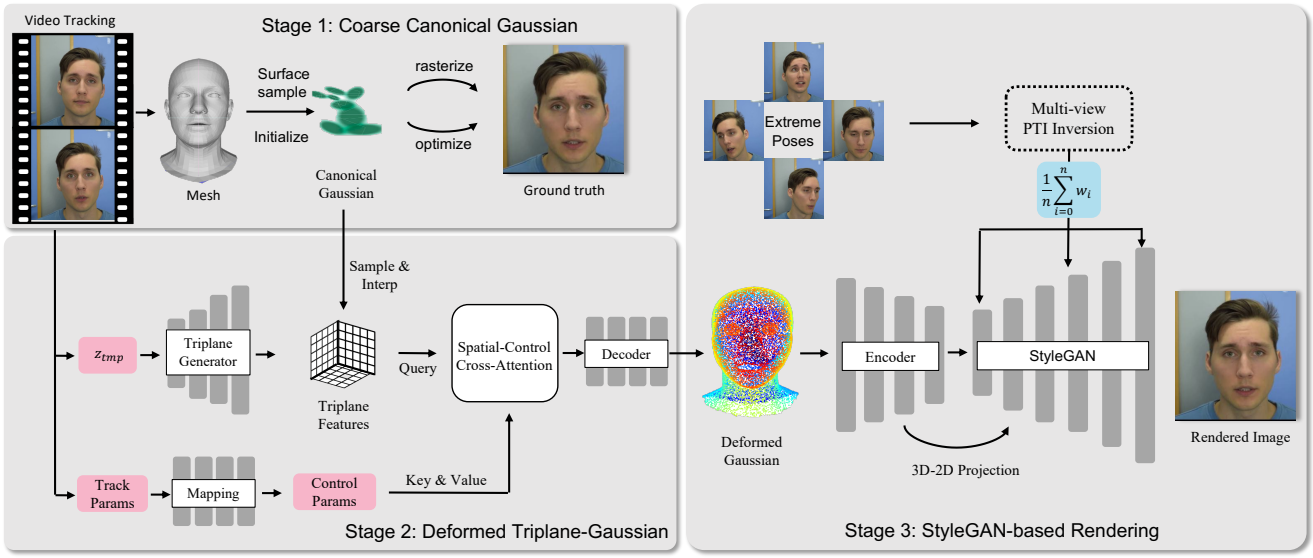

Figure 2. The proposed Tri-Stage training strategy includes StyleGAN-based Volumetric Rendering. In Stage 1, we construct static coarse canonical Gaussians. In Stage 2, Gaussians are queried from a temporal-aware triplane for attention-based deformation. In Stage 3, we initialize the StyleGAN through multi-view PTI initialization and project dynamic Gaussian prior into StyleGAN for volumetric rendering.

3.1. Deformable Triplane-Gaussian

Temporal-ware Hybrid Representation Recent studies [44, 59] have demonstrated that hybrid triplane-Gaussian representations are effective in capturing continuous, structural, and low-dimensional features for 3D modeling. We extend this strategy to develop a deformable hybrid representation for 4D head modeling. Our approach employs a convolutional neural network generator, inspired by the StyleGAN architecture [16], to synthesize features within a triplane representation. To incorporate temporal dynamics, we introduce a frame-specific latent code, denoted as z_{tmp}, into the generator for each input frame. For each 3D Gaussian centered at μ, the coordinates are normalized, and corresponding features are obtained by interpolating the position on a regularly spaced 2D grid for each plane. These features are concatenated $\bigcup$ across dimensions to produce a final feature vector $F(\mu)$ or each canonical Gaussian position μ_c:

$$F(\mu) = \bigcup \text{interp}(plane, \mathbf{P}(\mu)) \tag{1}$$

where $\mathbf{P}(\mu)$ denotes a projection of μ onto the plane and 'interp' represents bilinear interpolation on the 2D grid.

Attention-based Deformation Traditional deformable Gaussian models [46, 47, 51] typically concatenate conditioning parameters with Gaussian points to predict offsets for dynamic rendering. However, this approach overlooks a critical challenge inherent in dynamic 3D head modeling: the assumption that a fixed 3D coordinate in the canonical space will always correspond to the same facial region throughout the entire sequence.

To address this limitation and improve the correspondence between Gaussian points and conditioning parameters (such as facial expressions and head poses), we introduce a cross-attention mechanism. This mechanism fuses the spatial feature embeddings $F(\mu_c)$ of the canonical 3D Gaussians with the conditioning parameters, capturing how input expression and other factors influence the movement of the 3D Gaussians. The cross-attention mechanism layer $\text{CA}(\cdot)$ and MLP layer $\text{FFN}(\cdot)$, each connected via skip connections. The process is defined as follows:

$$F(\mu)' = \text{CA}(F(\mu), \mathbf{c}n) + F(\mu), \tag{2}$$

$$Z(\mu) = \text{FFN}(F(\mu)') + F(\mu)', \tag{3}$$

where the cross-attention is computed between the triplane feature $F(\mu)$ and the conditional feature c_n of the n-th image frame. The output feature $Z(\mu)$ effectively integrates the conditioning information with the detailed facial features captured by each 3D Gaussian.

Finally, based on the condition-aware feature representation given the cross-attention, we leverage a deformation MLP $\text{Deform}(\cdot)$ for predicting the spatial dynamic offsets of Gaussians:

$$\Delta\mathbf{c}, \Delta\mu, \Delta\mathbf{r}, \Delta\mathbf{s} = \text{Deform}(Z(\mu)) \tag{4}$$

3.2. Extended StyleGAN on 3D Gaussian

Pre-trained on the aligned FFHQ dataset, StyleGAN struggles with unaligned avatars commonly found in portrait videos. We defer a detailed analysis to the Appendix. Additionally, the lack of geometric awareness prevents Style-

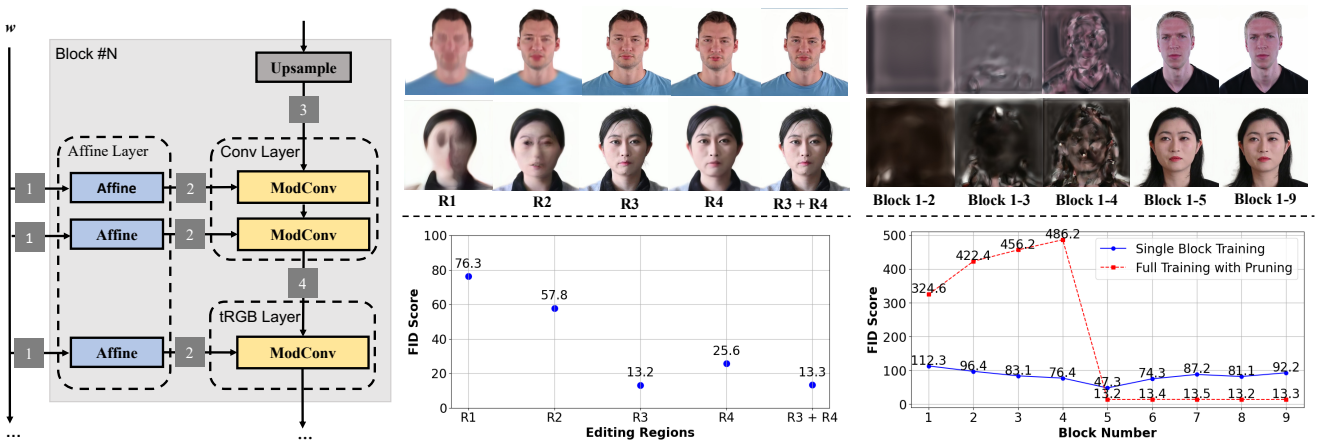

Figure 3. Left: Four regions within a single StyleGAN Block for features manipulation. Mid: Integration to R3 performs the best. R3+R4 does not bring improvement. Right: Blocks 1 to 5 are effective for volumetric projection. The upper refers to the block pruning results.

GAN from effectively handling novel view and pose reconstruction, which is essential for video avatar rendering. To address this limitation, we present a novel methodology that efficiently retains StyleGAN's pre-trained generalization abilities while encoding animatable 3D Gaussian representations into its latent space. This extension enhances its capability to generate and edit drivable video portraits.

Multi-view PTI initialization First, we employ PTI inversion [29] from multiple images from the training dataset with extreme head poses to embed the target portrait within StyleGAN's latent distribution by subtly modifying the original model parameters. We first fix StyleGAN's parameters and optimize style code w to minimize the discrepancy between the generated and target images, indicating that w closely aligns with our target in the latent space. Subsequently, we fix w and fine-tune StyleGAN to enhance the similarity of the generated image to the target at this w.

Volumetric Projection to StyleGAN Next, we conduct a comprehensive analysis of StyleGAN's structure. Directly leveraging StyleGAN for unaligned 3D presentations is a non-trivial task. To resolve this issue, we investigate StyleGAN's architecture for the integration of dynamic volumetric representations from the Gaussians. Inspired by Pixel2Style2Pixel [28], we designed a Convolutional Encoder to encode Gaussian priors. However, directly formulating the 3D feature projection through manipulating style code w presents unwanted results. To explore the effective strategy of volumetric feature projection to StyleGAN, we further investigate StyleGAN's architecture. Fig.3 Left presents the four regions for 3D projection within a single StyleGAN Block. R1 refers to style code w manipulation. We in addition propose R2: Altering the style latent post Affine-Layer mapping. R3: Integrate the prior volumetric feature with the Conv-Layer feature. R4: Integrate the prior volumetric feature with the tRGB-Layer feature. We defer the details of the model structure in the Appendix.

Volumetric Feature Integration We train StyleGAN with modification within four regions over 10 epochs. Our analysis, illustrated in Fig. 3 Middle, reveals that R1, R2, and R4 yield imprecise facial details, albeit preserving the general facial positioning within images. This suggests that modifications at the latent code, Affine-Layer levels, or intermediate tRGBs fail to impart adequate texture detail to StyleGAN. Conversely, we discover that feature integration to Conv-Layers suffices for embedding volumetric Gaussian priors into StyleGAN. We in addition explored R3 + R4, resulting in no performance difference. Consequently, we opt for exclusively modifying only the Conv-Layers.

Effective StyleGAN Blocks Our subsequent investigation focuses on determining the effective blocks for the modification. We adopt two strategies for the investigation. (1) Integrate projection to a single block (2) Integrate projections across all StyleGAN blocks during training, and prune projections during inference. Shown in Fig. 3 Right, our study for (1) reveals that Blocks 1 to 5 (4×4 to 64×64) are effective with Block 5 significantly better than others. In addition, (2) presents that during pruning, Blocks 1 to 4 are instrumental for geometry while Block 5 refines texture. The others can be pruned with no detrimental impact. We thus choose the first five blocks for the volumetric feature projection.

3.3. Training strategy of Volumetric Rendering

Canonical Gaussian We first reconstruct the mean shape of the talking face, by optimizing the positions of 3D Gaussians and the triplane generator. We initialize the 3D Gaussian center positions by sampling the surface of the mesh from video tracking. This preserves the shape topology of the face and landmarks of the target avatar. Unlike conventional Gaussian rendering that only considers RGB, we render a 32-channel volumetric feature for better 3D representation. We employ both L_1 loss and LPIPS loss for

aligning synthesized images with ground truth. We focus on the first three channels $I_{gs}^{(1:3)}$, comparing them against RGB ground-truth images I_{gt}. Other channels will be deferred to StyleGAN-based rendering.

$$\mathcal{L}_{rgb} = ||I_{gs}^{(1:3)} - I_{gt}||_1 \tag{5}$$

Deformation We optimize the triplane generator, cross-attention, and deformer for deformation. In addition to the previous L1 loss, we additionally focus on the eye and mouth regions for learning the expression.

$$\mathcal{L}_{lmk} = ||R_n(I_{gs}) - R_n(I_{gt})||_1 \tag{6}$$

R_n is either the eyes or mouth region extracted using RoI-align on the bounding boxes calculated using landmarks.

StyleGAN-based Volumetric Synthesis The final stage optimizes the triplane generator, the deformer, the encoder, and the projection layers while freezing the whole StyleGAN. We employ the LPIPS loss [13] between synthesized and ground-truth images.

$$\mathcal{L}_{perp} = \text{LPIPS}(I_{gan}, I_{gt}). \tag{7}$$

To further enhance image fidelity, we introduce a conditional discriminator, using UV maps as a condition to compare generated images with ground truth. This method, employing conditional adversarial loss [24]:

$$\begin{aligned}\mathcal{L}_{cGAN}(G, D) =& \mathbb{E}_{I_{gt},uv}\left[\log D\left(I_{gt}, uv\right)\right] \\ &+ \mathbb{E}_{uv}\left[\log\left(1 - D\left(uv, I_{gan}\right)\right]\right.\end{aligned} \tag{8}$$

where D aims to distinguish between $\{(I_{gt}, uv)\}$ and $\{(I_{vr}, uv)\}$, and $(\cdot, \cdot)$ denotes concatenation.

4. Experiment

Implementation Details We implement our model with PyTorch and a single A6000 GPU. We use StyleGAN2 distill-version, MobileStyleGAN 1024x1024, pre-trained on FFHQ as the generator for all studies. For Coarse 3D Gaussian and Deformation jointly, We train the model for 10,000 iterations. For the StyleGAN-based synthesis stage, we train the triplane generator, the deformer, the encoder, and the projection layers while freezing the StyleGAN with a batch size of 4 for 50,000 iterations. The Adam optimizer [18] is adopted for all learnable parameters with a learning rate of $1e^{-4}$. We present the details of the encoder and projection layer in the Appendix.

Dataset Our method takes a monocular video as input and leverages expression parameters from tracking to achieve video portrait rendering and editing. We primarily conduct experiments on data from INSTA [25], NeRFFace [6], NerfBlendShape [9] and Tri^2-Plane [37]. Each data sample captures a diverse range of facial motions in an average of 5-minute duration. All videos are resized to 1024^2 for our model. We divide the training set and testing set from each video to 80% and 20% of all frames, respectively. We conduct experiments comparing self/cross-reenactment with the current drivable portrait rendering techniques for avatar reconstruction and control.

Methods	F-LMD↓	SD↓	PSNR↑	LPIPS↓	MOS_1	MOS_2	MOS_3
	Quantitative Results				User Study		
	Dataset A				Self-Reenactment		
FlashAvatar	2.96	9.43	27.97	29.42	2.26	1.97	2.05
PointAvatar	2.55	8.42	28.39	23.64	3.67	4.03	3.96
SplatAvatar	2.88	4.54	32.53	25.47	3.88	3.76	4.21
Ours	**2.42**	**3.38**	**34.43**	**13.14**	**4.47**	**4.27**	**4.73**
	Dataset B				Cross-Reenactment		
FlashAvatar	3.82	10.67	25.43	25.35	1.51	2.12	1.13
PointAvatar	2.64	8.24	26.19	21.42	3.31	2.89	3.80
SplatAvatar	3.11	5.23	28.32	19.46	3.45	2.76	3.11
Ours	**2.31**	**2.84**	**30.44**	**11.82**	**4.21**	**3.83**	**3.89**

Table 1. (1) Left: Quantitative results of FlashAvatar [47], PointAvatar [57], SplatAvatar [30]. We bold the best. The values of SD and LPIPS are multiplied by 10^{-1} and 10^2 respectively. (2) Right: The MOS score for human evaluation. Each one comes from a 5-point Likert scale (from 1 to 5 are correspond to poor to excellent). The closer to 5 the better, we bold the best.

Baseline Methods We benchmark GaussianStyle against the following methods for monocular video avatar rendering: (1) FlashAvatar [47], (2) PointAvatar [57], (3) SplatAvatar [30]. We do not include GaussianAvatars [27] and Gaussian-Head-Avatar [48], which reconstruct human heads from multi-views, unlike single-view monocular videos. We defer the comparison with NeRF-based and StyleGAN-based rendering methods in the Appendix.

4.1. Quantitative Evaluation

Evaluation Metrics We evaluate the effectiveness of our method on three aspects: (1) F-LMD [3]: The differences in head pose and facial expression landmark positions calculated via MediaPipe [21]. (2) The Sharpness Difference (SD) [23]: It is used to evaluate the sharpness difference between the source and generated images at the pixel level. (3) Image Spatial Quality: we adopt the PSNR, the Learned Perceptual Image Patch Similarity (LPIPS) [55], and SSIM for image generation quality evaluation.

Evaluation Results Table 1 summarizes the quantitative results, where our method consistently outperforms the baselines in terms of image quality (PSNR, LPIPS), sharpness (SD), and motion accuracy (F-LMD). The significant improvement in LPIPS and SD highlights our method's ability to enhance detail intensity and perceptual similarity.

Figure 4. Our model outperforms other monocular avatar rendering methods in detail such as eyes and teeth.

4.2. Qualitative Results

Fig.4 and Fig.5 provide qualitative comparisons for self-reenactment and cross-reenactment scenarios, respectively. The highlighted zoomed-in regions of each generated image demonstrate the distinct strengths and weaknesses of each method. FlashAvatar produces noisy point clouds and blurred outputs due to inadequate regularization of implicit pose and expression deformations. SplatAvatar excels in shape reconstruction but struggles with appearance recovery, particularly in challenging areas like the torso, mouth, and eyes during cross-reenactment. Point-Avatar manages to recover specific features such as glasses but delivers overly smooth facial representations, failing to capture subtle expressions and clear teeth despite significant computational demands. In cross-reenactment, the baseline methods generally exhibit blurring and noise due to out-of-domain expressions and head poses. In contrast, our approach, leveraging cross-attention for deformation, maintains a robust representation of 3D points conditioned on tracking parameters. This enables precise recovery of high-frequency details and accurate control of head pose and facial expressions, outperforming all comparative baselines across both self-reenactment and cross-reenactment scenarios.

User Study We conducted a user study to evaluate the visual quality of our method, following the protocols established in Deep Video Portrait [17]. The study in-

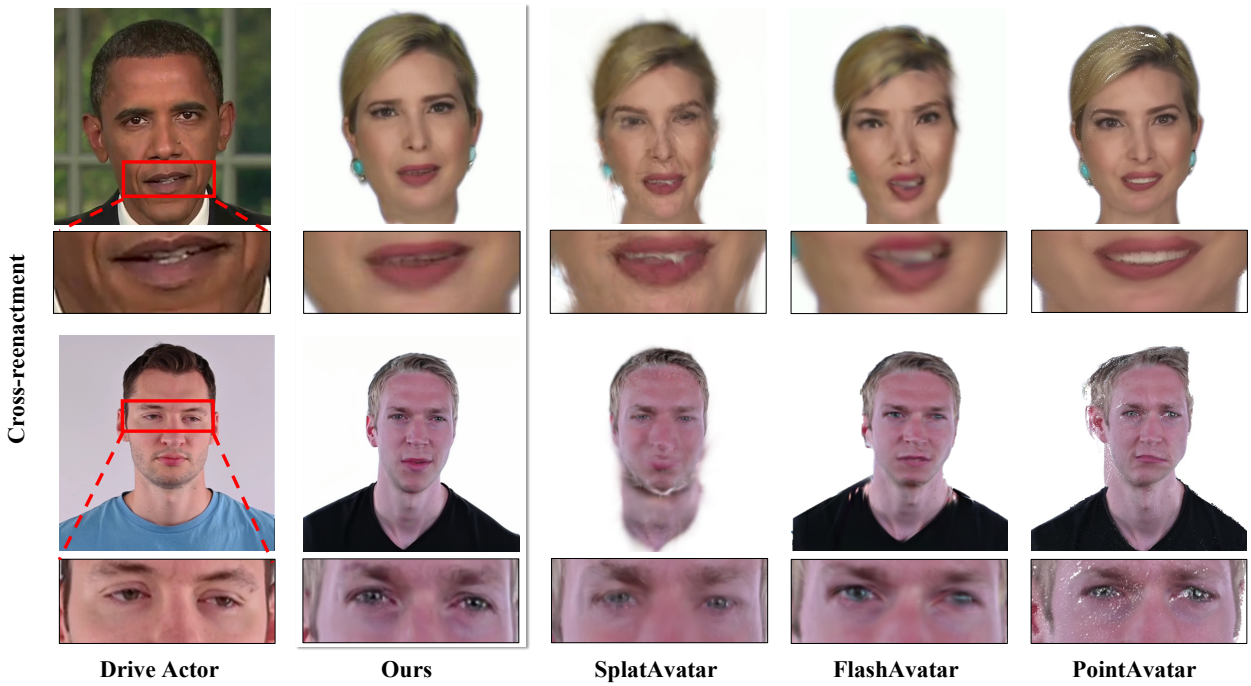

Figure 5. Other methods are not robust to novel views, expressions, or head poses and thus exhibit noisy point clouds and blurred results.

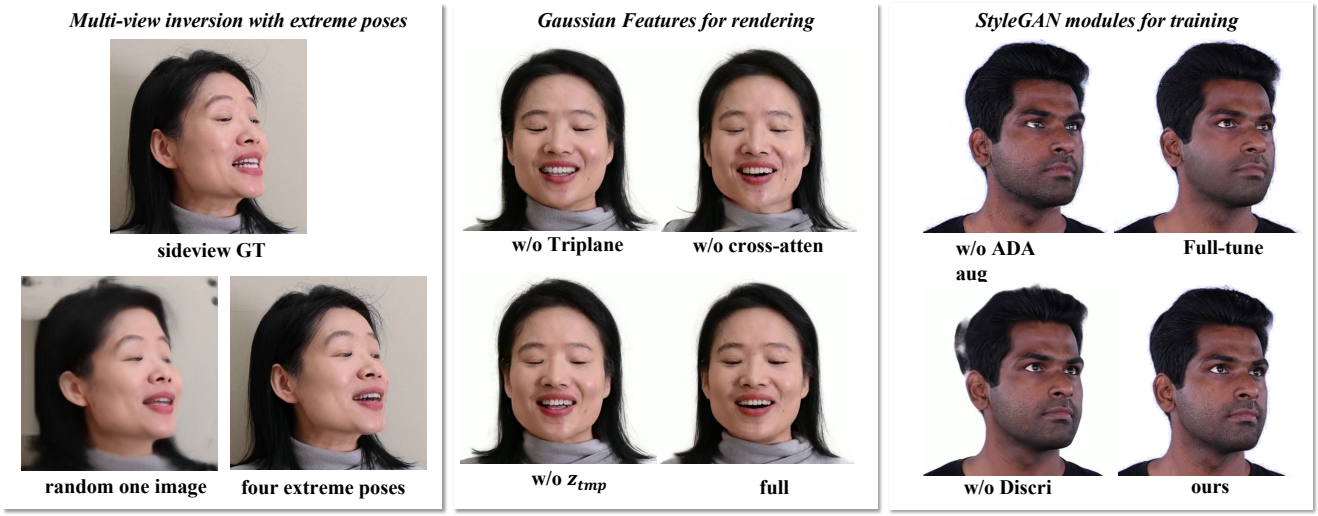

Figure 6. Ablations for multi-view PTI inversion, gaussian-feature contribution, and StyleGAN training.

cluded 40 videos—20 for self-reenactment and 20 for cross-reenactment. We recruited 20 participants via Amazon Web Services (AWS) to assess the quality based on several criteria, using the Mean Opinion Scores (MOS) rating system. Participants rated the videos on: (1) MOS_1: *"How is the image quality in the video?"*, (2) MOS_2: *"How realistic does the video appear?"*, and (3) MOS_3: *"Are the facial motions synchronized between the two videos?"*. The videos were presented in random order to capture participants' initial impressions. As shown in the right section of Table 1, our method outperformed others across all criteria, demonstrating superior video quality, realism, and motion synchronization.

4.3. Ablation Studies

Multi-view PTI Inversion We investigate strategies of PTI inversion for StyleGAN initialization: (1) a single random image, (2) multiple random images, and (3) multiple images depicting extreme poses. In Fig. 6, PTI with a single image produces blurred results for side views. However, initializing with four images capturing extreme poses along the x

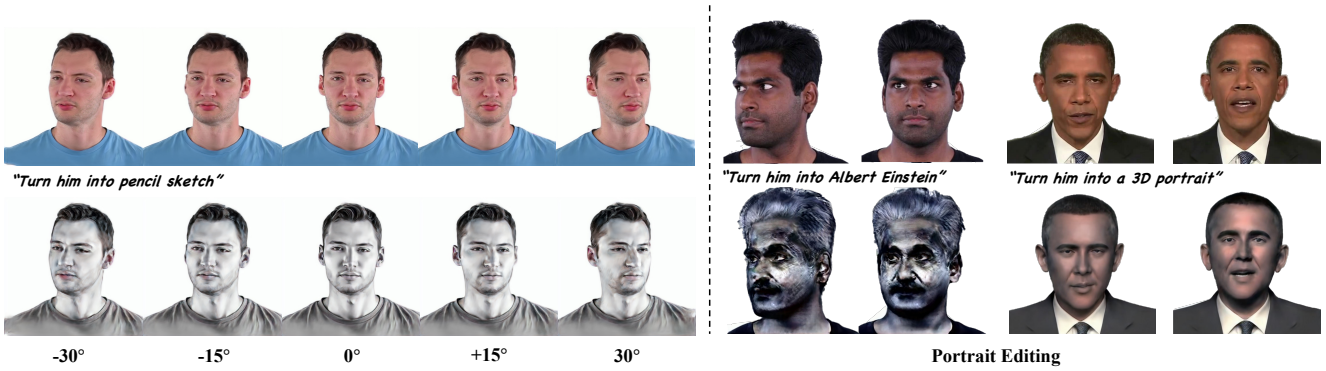

Figure 7. Left: We apply -30 to +30 degree rotation to the camera. The visualized results are consistent over multi-views. Right: 200 images are sampled for fine-tuning. IN2N is applied for text-driven editing guidance and takes about 10 minutes for each subject.

View Init.	LPIPS↓	PSNR↑	SSIM↑
1 random	18.51	32.47	0.842
4 random	16.42	33.36	0.839
2 extreme	15.63	34.02	0.847
8 extreme	14.57	34.21	0.873
Ours	**13.14**	**34.43**	**0.886**

(a) The effect of multi-view PTI Inversion

Gauss Feats.	PSNR↑	SSIM↑	Train↓
w/o triplane	30.32	0.818	**15 min**
w/o cross-atten	32.38	0.834	**15 min**
w/o z_{tmp}	33.15	0.857	**15 min**
w/o init	28.14	0.764	25 min
Ours	**34.43**	**0.886**	**15 min**

(b) The effect of different types of features.

StyleGAN feats.	PSNR↑	Train↓	In-Speed↑
StyleGAN2	**35.53**	3.5h	18fps
w/o ADA-aug	30.62	2.5h	**30fps**
w/o Discri	30.44	**2h**	**30fps**
full-tune	34.32	5.5h	**30fps**
MobileStyleGAN	34.43	2.5h	**30fps**

(c) The effect of StyleGAN training.

Table 2. Ablations of our method. We vary views for initialization, feature types, and StyleGAN training to investigate their effectiveness.

and y axes effectively addresses this issue, shown in Tab 2a. Inversion with additional images lowers the performance.

Rendering Features To assess the contribution of each component, we design the following variations (1) w/o triplane: Pure Gaussian representation without triplane. (2) w/o cross-atten: No cross-attention but only MLP for deformation used in DeformableGaussian [51] and FlashAvatar [47] for Gaussian offsets. (3) w/o triplane generator: triplane not generated by a Stylegan-like generator but learned as in GaussianHead [44] and 4DGaussian [46] (4) w/o init: No Gaussian initialization based on mesh but optimized from scratch. As detailed in Tab. 2b and Fig. 6, our design of Deformable Hybrid Triplane-Gaussian representation significantly improves the rendering quality.

StyleGAN Features To assess the contribution of each component, we design the following variations (1) StyleGAN: all other modules remain the same except replacing MobileStyleGAN with StyleGAN2 (2) w/o ADA-aug: no data augmentation used in StyleGAN-ADA [15] applied during training (3) w/o Discri: No distriminator used during training. (4) full-tune: No PTI inversion and fully finetuned StyleGAN without freezing. As detailed in Tab. 2b and Fig. 6, our strategy that preserves StyleGAN's latent distribution helps reduce the training time while improving quality. In addition, the data augmentation significantly improves the performance in the monocular video setting. MobileStyleGAN achieves a compatible performance but much faster than StyleGAN2. We thus take MobileStyleGAN as the final generator.

4.4. Applications: 3D Editing and Novel View

GaussainStyle is a general representation of volumetric head avatars and can be easily extended for novel view synthesis and portrait editing.

Novel View Synthesis We fix the control parameters and apply -30 to +30 degree rotation to the camera. Fig. 7 shows our model is robust in novel view synthesis.

Portrait Editing Following TextToon [35], we use InstructionPixel2Pixel [1] for guidance. After training on realistic faces, we randomly select 200 images from the dataset for iterative dataset update. We freeze the Gaussian, deformation modules, and StyleGAN, with only the projection layer trainable. All other settings remain the same as in IN2N. The typical editing time is around 10 minutes per subject on a single GPU.

5. Conclusion

In this work, we present **GaussianStyle**, a novel framework that combines 3D Gaussian splatting with StyleGAN for high-fidelity volumetric avatar generation. Our temporal-aware tri-plane and attention-based deformation module refine the Gaussian representation for robust dynamic face rendering. By mapping dynamic 3D representations to StyleGAN's latent space, we retain the pretrained model's generalization abilities while enabling editable neural representations. We achieve an inference speed of over 30 FPS while maintaining high fidelity across novel-view and self/cross-reenact synthesis scenarios

References

[1] Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. *arXiv preprint arXiv:2211.09800*, 2022. 8, 1

[2] Eric R Chan, Connor Z Lin, Matthew A Chan, Koki Nagano, Boxiao Pan, Shalini De Mello, Orazio Gallo, Leonidas J Guibas, Jonathan Tremblay, Sameh Khamis, et al. Efficient geometry-aware 3d generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16123–16133, 2022. 1, 4

[3] Lele Chen, Zhiheng Li, Ross K. Maddox, Zhiyao Duan, and Chenliang Xu. Lip movements generation at a glance, 2018. 5

[4] Yufan Chen, Lizhen Wang, Qijing Li, Hongjiang Xiao, Shengping Zhang, Hongxun Yao, and Yebin Liu. Monogaussianavatar: Monocular gaussian point-based head avatar. In *ACM SIGGRAPH 2024 Conference Papers*, pages 1–9, 2024. 2

[5] Yu Deng, Jiaolong Yang, Sicheng Xu, Dong Chen, Yunde Jia, and Xin Tong. Accurate 3d face reconstruction with weakly-supervised learning: From single image to image set. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 0–0, 2019. 2

[6] Guy Gafni, Justus Thies, Michael Zollhöfer, and Matthias Nießner. Dynamic neural radiance fields for monocular 4d facial avatar reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8649–8658, 2021. 5

[7] Guy Gafni, Justus Thies, Michael Zollhofer, and Matthias Nießner. Dynamic neural radiance fields for monocular 4d facial avatar reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8649–8658, 2021. 1

[8] Rinon Gal, Or Patashnik, Haggai Maron, Amit H Bermano, Gal Chechik, and Daniel Cohen-Or. Stylegan-nada: Clip-guided domain adaptation of image generators. *ACM Transactions on Graphics (TOG)*, 41(4):1–13, 2022. 2, 4

[9] Xuan Gao, Chenglai Zhong, Jun Xiang, Yang Hong, Yudong Guo, and Juyong Zhang. Reconstructing personalized semantic facial nerf models from monocular video. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia)*, 41(6), 2022. 1, 5

[10] Xuan Gao, Chenglai Zhong, Jun Xiang, Yang Hong, Yudong Guo, and Juyong Zhang. Reconstructing personalized semantic facial nerf models from monocular video. *ACM Transactions on Graphics (TOG)*, 41(6):1–12, 2022. 2

[11] Michal Geyer, Omer Bar-Tal, Shai Bagon, and Tali Dekel. Tokenflow: Consistent diffusion features for consistent video editing. *arXiv preprint arxiv:2307.10373*, 2023. 1, 4

[12] Ayaan Haque, Matthew Tancik, Alexei A Efros, Aleksander Holynski, and Angjoo Kanazawa. Instruct-nerf2nerf: Editing 3d scenes with instructions. *arXiv preprint arXiv:2303.12789*, 2023. 1, 5

[13] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14*, pages 694–711. Springer, 2016. 5

[14] 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. 2

[15] Tero Karras, Miika Aittala, Janne Hellsten, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Training generative adversarial networks with limited data. In *Proc. NeurIPS*, 2020. 8

[16] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8110–8119, 2020. 3

[17] Hyeongwoo Kim, Pablo Garrido, Ayush Tewari, Weipeng Xu, Justus Thies, Matthias Nießner, Patrick Pérez, Christian Richardt, Michael Zollöfer, and Christian Theobalt. Deep video portraits. *ACM Transactions on Graphics (TOG)*, 37 (4):163, 2018. 6, 1, 3

[18] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017. 5

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

[20] Hanxiao Liu, Zihang Dai, David R. So, and Quoc V. Le. Pay attention to mlps, 2021. 1

[21] Camillo Lugaresi, Jiuqiang Tang, Hadon Nash, Chris McClanahan, Esha Uboweja, Michael Hays, Fan Zhang, Chuo-Ling Chang, Ming Guang Yong, Juhyun Lee, et al. Mediapipe: A framework for building perception pipelines. *arXiv preprint arXiv:1906.08172*, 2019. 5

[22] Zhiyuan Ma, Xiangyu Zhu, Guojun Qi, Zhen Lei, and Lei Zhang. Otavatar: One-shot talking face avatar with controllable tri-plane rendering. *arXiv preprint arXiv:2303.14662*, 2023. 2, 1, 3

[23] Michael Mathieu, Camille Couprie, and Yann LeCun. Deep multi-scale video prediction beyond mean square error. *arXiv preprint arXiv:1511.05440*, 2015. 5

[24] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets, 2014. 5

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

[26] Hao Ouyang, Qiuyu Wang, Yuxi Xiao, Qingyan Bai, Juntao Zhang, Kecheng Zheng, Xiaowei Zhou, Qifeng Chen, and Yujun Shen. Codef: Content deformation fields for temporally consistent video processing. *arXiv preprint arXiv:2308.07926*, 2023. 1, 5

[27] Shenhan Qian, Tobias Kirschstein, Liam Schoneveld, Davide Davoli, Simon Giebenhain, and Matthias Nießner. Gaussianavatars: Photorealistic head avatars with rigged 3d gaussians. *arXiv preprint arXiv:2312.02069*, 2023. 2, 5

[28] Elad Richardson, Yuval Alaluf, Or Patashnik, Yotam Nitzan, Yaniv Azar, Stav Shapiro, and Daniel Cohen-Or. Encoding in style: a stylegan encoder for image-to-image translation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 4

[29] Daniel Roich, Ron Mokady, Amit H Bermano, and Daniel Cohen-Or. Pivotal tuning for latent-based editing of real images. *ACM Transactions on graphics (TOG)*, 42(1):1–13, 2022. 2, 4

[30] Zhijing Shao, Zhaolong Wang, Zhuang Li, Duotun Wang, Xiangru Lin, Yu Zhang, Mingming Fan, and Zeyu Wang. SplattingAvatar: Realistic Real-Time Human Avatars with Mesh-Embedded Gaussian Splatting. In *Computer Vision and Pattern Recognition (CVPR)*, 2024. 5, 1

[31] Luchuan Song, Bin Liu, and Nenghai Yu. Talking face video generation with editable expression. In *Image and Graphics: 11th International Conference, ICIG 2021, Haikou, China, August 6–8, 2021, Proceedings, Part III 11*, pages 753–764. Springer, 2021. 2

[32] Luchuan Song, Zheng Fang, Xiaodan Li, Xiaoyi Dong, Zhenchao Jin, Yuefeng Chen, and Siwei Lyu. Adaptive face forgery detection in cross domain. In *European Conference on Computer Vision*, pages 467–484. Springer, 2022. 2, 5

[33] Luchuan Song, Xiaodan Li, Zheng Fang, Zhenchao Jin, YueFeng Chen, and Chenliang Xu. Face forgery detection via symmetric transformer. In *Proceedings of the 30th ACM international conference on multimedia*, pages 4102–4111, 2022. 5

[34] Luchuan Song, Guojun Yin, Zhenchao Jin, Xiaoyi Dong, and Chenliang Xu. Emotional listener portrait: Realistic listener motion simulation in conversation. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 20782–20792. IEEE, 2023. 2

[35] Luchuan Song, Lele Chen, Celong Liu, Pinxin Liu, and Chenliang Xu. Texttoon: Real-time text toonify head avatar from single video. In *SIGGRAPH Asia 2024 Conference Papers*, pages 1–11, 2024. 2, 8, 1

[36] Luchuan Song, Pinxin Liu, Lele Chen, Celong Liu, and Chenliang Xu. Tri^2-plane: Volumetric avatar reconstruction with feature pyramid, 2024. 1

[37] Luchuan Song, Pinxin Liu, Lele Chen, Guojun Yin, and Chenliang Xu. Tri 2-plane: Thinking head avatar via feature pyramid. In *European Conference on Computer Vision*, pages 1–20, 2024. 2, 5

[38] Luchuan Song, Pinxin Liu, Guojun Yin, and Chenliang Xu. Adaptive super resolution for one-shot talking-head generation. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4115–4119. IEEE, 2024. 2

[39] Jingxiang Sun, Xuan Wang, Yichun Shi, Lizhen Wang, Jue Wang, and Yebin Liu. Ide-3d: Interactive disentangled editing for high-resolution 3d-aware portrait synthesis. *ACM Transactions on Graphics (ToG)*, 41(6):1–10, 2022. 2

[40] Jingxiang Sun, Xuan Wang, Lizhen Wang, Xiaoyu Li, Yong Zhang, Hongwen Zhang, and Yebin Liu. Next3d: Generative neural texture rasterization for 3d-aware head avatars. In *CVPR*, 2023. 2, 1, 4

[41] Yunlong Tang, Junjia Guo, Pinxin Liu, Zhiyuan Wang, Hang Hua, Jia-Xing Zhong, Yunzhong Xiao, Chao Huang, Luchuan Song, Susan Liang, Yizhi Song, Liu He, Jing Bi, Mingqian Feng, Xinyang Li, Zeliang Zhang, and Chenliang Xu. Generative ai for cel-animation: A survey, 2025. 2

[42] Justus Thies, Michael Zollhöfer, and Matthias Nießner. Deferred neural rendering: Image synthesis using neural textures, 2019. 2

[43] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2023. 1

[44] Jie Wang, Jiu-Cheng Xie, Xianyan Li, Feng Xu, Chi-Man Pun, and Hao Gao. Gaussianhead: High-fidelity head avatars with learnable gaussian derivation, 2024. 1, 2, 3, 8

[45] 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. *arXiv preprint arXiv:2305.00942*, 2023. 1, 3

[46] Guanjun Wu, Taoran Yi, Jiemin Fang, Lingxi Xie, Xiaopeng Zhang, Wei Wei, Wenyu Liu, Qi Tian, and Xinggang Wang. 4d gaussian splatting for real-time dynamic scene rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 20310–20320, 2024. 3, 8

[47] Jun Xiang, Xuan Gao, Yudong Guo, and Juyong Zhang. Flashavatar: High-fidelity head avatar with efficient gaussian embedding. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 2, 3, 5, 8, 1

[48] Yuelang Xu, Benwang Chen, Zhe Li, Hongwen Zhang, Lizhen Wang, Zerong Zheng, and Yebin Liu. Gaussian head avatar: Ultra high-fidelity head avatar via dynamic gaussians. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 1, 5

[49] Shuai Yang, Liming Jiang, Ziwei Liu, and Chen Change Loy. Pastiche master: Exemplar-based high-resolution portrait style transfer. In *CVPR*, 2022. 2

[50] Shuai Yang, Yifan Zhou, Ziwei Liu, , and Chen Change Loy. Rerender a video: Zero-shot text-guided video-to-video translation. In *ACM SIGGRAPH Asia Conference Proceedings*, 2023. 1, 5

[51] Ziyi Yang, Xinyu Gao, Wen Zhou, Shaohui Jiao, Yuqing Zhang, and Xiaogang Jin. Deformable 3d gaussians for high-fidelity monocular dynamic scene reconstruction. *arXiv preprint arXiv:2309.13101*, 2023. 3, 8

[52] Fei Yin, Yong Zhang, Xiaodong Cun, Mingdeng Cao, Yanbo Fan, Xuan Wang, Qingyan Bai, Baoyuan Wu, Jue Wang, and Yujiu Yang. Styleheat: One-shot high-resolution editable talking face generation via pre-trained stylegan. In *European conference on computer vision*, pages 85–101. Springer, 2022. 2, 1, 3

[53] Junzhe Zhang, Yushi Lan, Shuai Yang, Fangzhou Hong, Quan Wang, Chai Kiat Yeo, Ziwei Liu, and Chen Change Loy. Deformtoon3d: Deformable neural radiance fields for 3d toonification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9144–9154, 2023. 2

[54] Pengfei Zhang, Pinxin Liu, Hyeongwoo Kim, Pablo Garrido, and Bindita Chaudhuri. Kinmo: Kinematic-aware human motion understanding and generation, 2024. 2

[55] 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

[56] Yufeng Zheng, Victoria Fernández Abrevaya, Marcel C Bühler, Xu Chen, Michael J Black, and Otmar Hilliges. Im avatar: Implicit morphable head avatars from videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13545–13555, 2022. 1, 2, 3, 4

[57] Yufeng Zheng, Wang Yifan, Gordon Wetzstein, Michael J. Black, and Otmar Hilliges. Pointavatar: Deformable point-based head avatars from videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 2, 5, 4

[58] Wojciech Zielonka, Timo Bolkart, and Justus Thies. Instant volumetric head avatars. In *Conference on Computer Vision and Pattern Recognition*, 2023. 1, 2, 3

[59] Zi-Xin Zou, Zhipeng Yu, Yuan-Chen Guo, Yangguang Li, Ding Liang, Yan-Pei Cao, and Song-Hai Zhang. Triplane meets gaussian splatting: Fast and generalizable single-view 3d reconstruction with transformers. *arXiv preprint arXiv:2312.09147*, 2023. 3

GaussianStyle: Gaussian Head Avatar via StyleGAN

Supplementary Material

5.1. Model Structure

Transformer for Gaussian Deformation We use 2-layers of transformer blocks, each with a cross-attention layer and a Feed-Forward layer. Unlike the vanilla transformer [43], we use gated MLP [20] for the Feed-Forward layer.

Encoder The encoder is purely convolutional. It accepts both our modified 32-dimensional Gaussian representations. We obtain the projection layer features from the encoder and the initial input to StyleGAN, as seen in Fig. 8

Volumetric Projection Volumetric Projections utilize only two convolutions to fuse the 3D feature with the StyleGAN intermediate features. Please see Fig. 8 for more information.

Triplane Generator We use a lightweight StyleGAN to generate the Triplane for Gaussian representation. The structure is similar to EG3D [2]. The latent dimension is 64 as the embedding for the frame index. During the self-reenactment or cross-reenactment, we fixed the frame index to 0 for inference.

5.2. Training details

Training Strategy We applied StyleGAN-ADA's geometric transformation during the training to improve the robustness. Fig. 9 shows the effectiveness of geometric transformation applied to UV maps during training, which allows the model to learn the relative position between the facial and the torso regions based on the UV map. This strategy significantly improves the self/cross-reenactment during extreme poses. For unseen poses, without geometric transformation, the generated portrait always contains a wrong facial shape.

Training/Inference Time To present a fair comparison between our methods and others, we present the training time and inference time in Tab. 3 for volumetric rendering and editing separately.

5.3. Editing details

We applied Instruct-Pixel2Pixel [1] (IP2P) as the guidance tool for editing following TextToon [35]. We discover that the raw IP2P model does not present consistency for different views. To address this problem, we first take the novel view synthesis based on our model and feed these data with sampled 200 images to IP2P for finetuning. The finetuning process significantly improves the editing quality. During editing, we freeze all other parameters except the projection layers to the StyleGAN module.

Method	Training Time	Inference Time
Reconstruction		
IM-Avatar [56]	48h	0.1 fps
PointAvatar [56]	4h	15fps
INSTA [58]	2h	20fps
DVP [17]	12h	25fps
StyleAvatar [45]	6h	25fps
FlashAvatar [47]	0.5h	300 fps
SplattingAvatar [30]	0.5h	80fps
Next3D [40]	10h	20fps
StyleHeat [52]	8h	30fps
OTAvatar [22]	8h	20fps
Ours	2.5h	35fps
Editing		
TokenFlow [11]	30 min	0.5 fps
RAV [50]	30 min	0.8 fps
CoDeF [26]	30 min	40fps
IN2N[12] + GaussianStyle	10 min	35 fps

Table 3. Training/Inference Time Comparison for Avatar Rendering Methods and editing methods

6. Analysis of StyleGAN

We evaluate StyleGAN's ability to generate animatable video portraits, which involves capturing varying expressions, continuous facial motions, and cohesive upper body movement during head rotations. Unlike the aligned images in the pre-trained FFHQ dataset, animatable portraits are often unaligned and captured in diverse settings, with a variety of head positions and orientations.

To assess StyleGAN's effectiveness, we applied the GAN inversion method on both aligned and unaligned portraits, comparing the rendering results. This was crucial to determine if StyleGAN could accurately represent a dynamic portrait video. Our evaluation focused on frames showing extreme left and right head poses from videos as inputs for GAN inversion. This approach tested StyleGAN's limits in rendering realistic, continuous motion and its ability to capture the nuanced changes in facial orientation and expression. The insights gained from this assessment were instrumental in shaping the GaussianStyle framework, enhancing our understanding of the capabilities and limitations of StyleGAN in animatable portrait generation.

Inability for Unaligned portrait generation In Figure 10, the linear interpolation of latent codes for extreme poses is presented in two rows: the first for aligned and the second for unaligned inversion. With aligned inversion, interpolating between two style codes yields images that main-

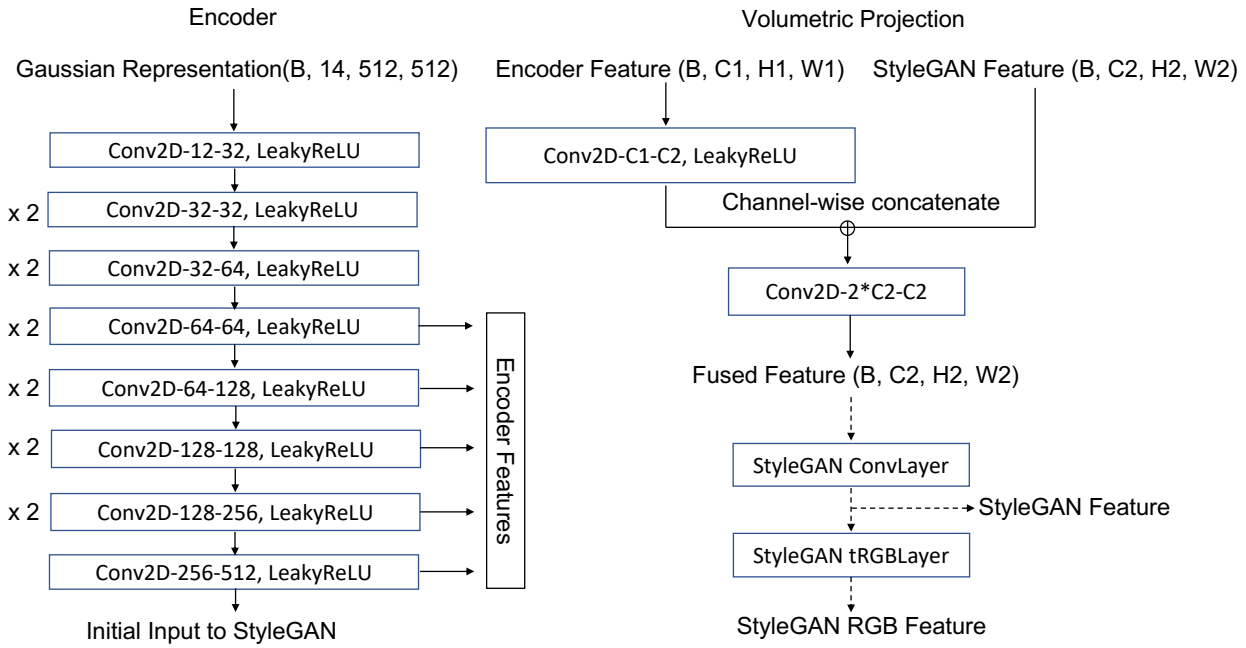

Figure 8. Both encoder and projections are purely convolutional. We obtain intermediate features from the encoder providing StyleGAN with dynamic Gaussian representations

Figure 9. Geometric transformation helps improve the performance of unseen novel views for self/cross-reenactment settings.

tain texture quality and exhibit consistent, smooth transitions in facial expressions and poses. This demonstrates StyleGAN's capability in handling aligned facial data. In contrast, the unaligned inversion results reveal StyleGAN's limitations. When processing unaligned faces, particularly in the animatable portrait domain, the model struggles, leading to blurred images. This blurring highlights its difficulty in accurately reconstructing the complex, varied aspects of unaligned faces, including nuanced head movements and expressions. This comparison underlines a key finding: while pre-trained StyleGAN is effective for aligned facial

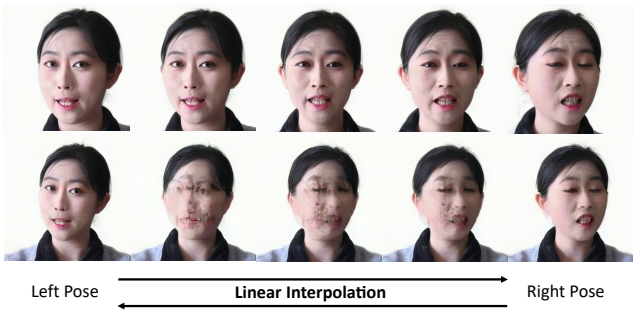

Figure 10. **Interpolation of GAN inversion**: Latent code interpolation between extreme pose parameters along the x-axis for aligned (upper) and unaligned (lower) video portraits.

portraits, it falls short in encoding complete portraits with upper body information, unable to capture the full range of portrait dynamics.

StyleGAN's latent Space In addition, we discover that StyleGAN can obtain a consistent neural representation of the target avatar. From the first line in Figure 10, we observe that even though only two images from extreme poses in the left and right directions are used for GAN inversion, StyleGAN is still capable of rendering relatively good intermediate images when interpolating the latent codes. This suggests that after GAN Inversion, the latent space encoded in StyleGAN remains continuous, motion-aware, and can be effectively sampled. Therefore, we can sample a small number of images from the video to perform GAN inversion, thereby obtaining the video's neural representation

Figure 11. Upper: Comparison with monocular video portrait rendering methods. Lower: Comparison with StyleGAN-based reenactment methods. The comparison suggests that existing methods are unable to deal with unaligned faces and extreme poses.

model.

7. Additional Experiments

In this section, we mainly present the comparison with the NeRF-based or 2D or StyleGAN based models for self/cross-reenactment.

7.1. Self/cross-reenactment

We further compared our method with the existing monocular video portrait rendering techniques, including Deep Video Portrait (DVP) [17], INSTA [58], IM-Avatar [56] and StyleGAN-based reenactment models, including StyleHEAT [52], OTAvatar [22] and StyleAvatar [45]. Specifically, OTAvatar and StyleHEAT are designed for aligned one-shot reenactment. To adapt them to unaligned situations and for a fair comparison, we finetuned their models on our video for 10 epochs. It takes about 1 day on a single A6000 to finish fine-tuning.

Fig. 11 shows the comparison between our methods with the existing NeRF-based and StyleGAN-based reenactment methods. INSTA has bad predictions for the non-facial areas. IM-Avatar presents over-smoothing results. DVP utilizs PNCC to Image translation, but struggles with the fine-grained details. StyleHeat cannot deal with unaligned faces

Figure 12. Next3D, after fine-tuning on target person video, is deficient in domain transfer, as visualized by the artifact for mouth regions.

and thus generates explicit artifacts during both self/cross-reenactment. OTAvatar utilized a Triplane [2] for geometry-aware 3D modeling of the target portrait. It cannot disentangle the movement of heads from the torso area. StyleAvatar stands out in cross-reenactment, while not as robust as our methods in dealing with extreme poses.

7.2. Editing

For editing comparison, we further include Next3D [40]. Since it cannot deal with unaligned data, we crop the images from videos. We fine-tuned Next3D on the cropped aligned videos for a fair comparison.

In Fig.12, we apply StyleGAN-NADA[8] to Next3D following fine-tuning on the aligned target portrait videos. Unlike Diffusion, the use of CLIP in Next3D does not ensure consistent intensity for editing. Furthermore, in contrast to our approach, which preserves StyleGAN's domain generalization capability by training only the projection layers while keeping StyleGAN frozen, our fine-tuning on Next3D diminishes its ability to render normal mouth areas, as evidenced by explicit artifacts in these regions.

7.3. Novel View Synthesis

We present the novel view results for 3D geometry evaluations. In case our method is trained on a short monocular portrait video without multi-view inputs, we range the reconstructed results under the viewpoints ranging from -30° to +30°, as shown in Fig. 13, the novel views maintain good visual quality within the range.

8. Baseline Details

8.1. Self/cross-reenactment

To demonstrate the fairness of our comparison with the baselines, we provide specific details on the various baselines and indicate how they differ from the original reports. Since part of the methods do not release source code, we reproduce them by ourselves with fairness.

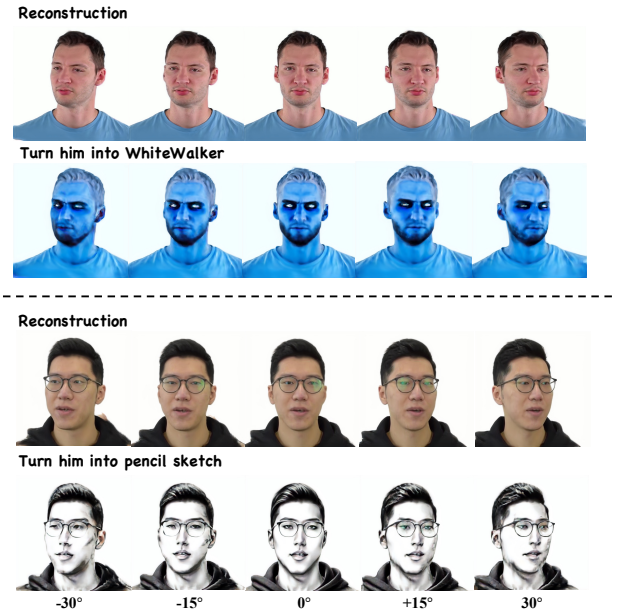

Figure 13. Our reconstruction and editing is consistent for novel views under various conditions.

FlashAvatar We adopt tracking parameters given by the authors and implement the training following the official GitHub repo.

PointAvatar and IM-Avatar The Point-Avatar [57] and IM-Avatar [56] shares the same data preprocessing. We follow the official report to perform the reconstruction.

SplattingAvatar This work adopts the same data preprocessing as in the previous two, we follow the official GitHub repo to reimplement the code.

INSTA We follow the provided official pipeline in the report.

StyleAvatar We reprocess our data via the FaceVerse in StyleAvatar and retrain it from the code in the official repo.

8.2. Portrait Editing

We compared our method with both guidance-based and video-based editing methods. Given the limitations of CoDEF and TokenFlow in handling long video sequences and the increasing GPU memory requirements with video length, respectively, we standardized our evaluation on 3-second video segments, roughly comprising 75 frames for a balanced comparison.

TokenFlow It first did inversion and then editing. We followed the official code provided by TokenFlow [11] for data preprocessing and editing.

Rerender-A-Video We apply the same prompt as that used

in TokenFlow for video-based editing following the officially released code by RAV [50]

CoDeF CoDeF's editing process involves modifying a canonical image via Instruct-Pix2Pix and generating the final edited video according to the deformation field. For the other procedures, we follow the officially released code by CoDeF [26] for data processing, training, and editing.

Insturct-NeRF2NeRF Compared with the original setting in IN2N [12], instead of training the model from scratch and iteratively updating the dataset. We selected a subset containing 200 images with our novel view synthesis as psueduo ground-truth for the model to finetune the model. It takes about 10 epochs to converge.

9. Metrics Detail

Peak Signal-to-Noise Ratio (PSNR). The PSNR is used to eval the generated image quality with ground truth. It is widely used in the field of evaluation image generation

Learned Perceptual Image Patch Similarity (LPIPS). The LPIPS is to apply the perceptual function at the patch level to calculate the feature distance between the generated image and ground truth.

Structural Similarity Index (SSIM). SSIM evaluates the visual impact of three key components: luminance, contrast, and structure.

Blind Image Spatial Quality Evaluator (BIQ). It is a metric to evaluate the generated images without ground truth.

10. Limitations

Although GaussianStyle is able to synthesize photo-realistic and fully animatable head avatars with editing capabilities, there are still areas for improvement:

(1) GaussianStyle relies on video tracking parameters. Inaccurate tracking of landmarks and expressions might introduce potential errors into our model, leading to artifacts and degraded facial details. Our method could benefit from a more accurate video tracking estimation method or corrective operations.

(2) GaussianStyle utilizes tracking parameters for Gaussian Point Deformation, which could introduce errors due to a lack of explicit regularization for landmark matching. In addition, the tracking always present the average expression but cannot capture the extreme expressions. Exploring more robust and accurate techniques could open new directions for future work.

(3) GaussianStyle is still sensitive to extreme views and poses. For out-of-domain camera views and head poses, our methods show degradation in rendering, as illustrated in Fig. 13.

11. Ethical Consideration

Our research primarily focuses on simulating high-fidelity facial avatars. However, due to its photo-realistic facial rendering capabilities, there exists a potential for misuse. For example, creating speech videos of public figures portraying events or statements that never occurred. The risk of such abuses is a longstanding concern in the field of AI-synthesized photo-realistic humans, evident in phenomena like deepfake swapping and talking head generation.

While it is challenging to completely prevent the misuse of this technology, our paper contributes by providing a technical analysis of facial synthesis. This insight allows users to better understand the field and recognize the limitations of AI synthesis to a certain extent, including aspects like tooth detail and temporal consistency [32, 33].

Furthermore, we advocate for responsible usage practices. These include measures like embedding watermarks in generated videos and employing synthetic face detection technologies for photo-realistic portraits. Such steps are crucial in mitigating the risks associated with this technology.