FLARE: FAST LIGHTWEIGHT AVATARS WITH REAL-TIME EXPRESSION

Anonymous authors
Paper under double-blind review

ABSTRACT

The deployment of interactive digital avatars, particularly for low-latency applications like LLM-driven interaction, is often hindered by a trade-off between rendering speed, visual quality, and generalization. While many methods achieve real-time frame rates, they can fall short of the high-throughput demands required for truly scalable and responsive systems. To address this, we present Fast Lightweight Avatars with Real-time Expression (FLARE), a one-shot 3D head avatar framework designed for high-speed synthesis. Our approach uses a decoupled Core-Shell Gaussian representation, with a dynamic FLAME-based Core and a static Shell for finer details. Crucially, we replace heavy per-frame neural deformation with a Hybrid Kinematic-Neural Deformation architecture. This relies on a one-time geometric binding based on barycentric coordinates as a gross kinematic base, paired with an lightweight Multi-Layer Perceptron (MLP) for high-frequency residual deformation and opacity control. This decoupled design effectively eliminates key computational bottlenecks in prior work. Consequently, our method achieves rendering speeds of over 600 FPS for motion reenactment and over 350 FPS for end-to-end speech-driven generation, measured at 512^2 resolution on a single NVIDIA A100 GPU. Extensive experiments show FLARE achieves state-of-the-art inference speed while maintaining competitive reconstruction quality, offering a practical and scalable solution for deploying high-fidelity avatars in latency-sensitive applications. *Code will be available upon publication.*

1 INTRODUCTION

The creation of lifelike, animatable digital humans from a single photograph is a cornerstone of modern computer graphics. Such technology is poised to redefine human interaction across immersive telepresence, virtual reality, and digital companionship. A key objective in this domain is the development of dynamic, responsive avatars that can be generated instantaneously and animated in real-time. This demand for real-time performance extends beyond mere visual fluidity; it is a functional prerequisite for both low-latency interaction and computational scalability. For conversational AI, the perceptual latency budget is already strained by a 500-1000 ms delay from the ASR-LLM-TTS pipeline. An inefficient rendering stage consuming 50 ms per frame (20 FPS) critically exacerbates this delay, making natural interaction impossible. From a scalability perspective, the high throughput afforded by an efficient renderer is equally critical. A method with high per-frame inference cost cannot be deployed at scale, as the computational resources required would be prohibitive. Therefore, extreme rendering speed is not just a performance metric, but a direct measure of a model's potential for both real-world interactivity and large-scale deployment.

However, achieving this level of performance without compromising other key attributes has proven exceptionally difficult. The field has long navigated a complex set of trade-offs between three critical features: **one-shot generalization**, **photorealistic quality**, and the aforementioned **high-performance rendering**. The research landscape exemplifies this trilemma, with state-of-the-art methods consistently sacrificing at least one pillar. While 2D generative models like VASA-1 Xu et al. (2024) offer impressive realism, their iterative nature is too slow for interaction. Consequently, the field has focused on 3D-aware methods, yet these are also caught in compromise. Person-specific systems Qian et al. (2024) prove that high speed and quality are attainable, but they abandon one-shot generalization. Conversely, generalizable one-shot methods struggle to balance the remaining pillars. Most sacrifice speed for quality (e.g., SEGA Guo et al. (2025) at 20 FPS); a recent standout, LAM He

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

et al. (2025), achieves a remarkable 280 FPS, but this performance relies on large transformer blocks, making its one-shot reconstruction phase computationally prohibitive for on-device applications. This performance bottleneck is especially dire in the critical audio-driven modality, where the leading one-shot method operates at a mere 4.5 FPS Ye et al. (2024). This leaves a critical void for a solution that is simultaneously generalizable, lightweight, and capable of real-time audio-driven performance.

This paper presents FLARE, a framework that addresses these challenges by replacing architecturally complex animation modules with a **Hybrid Kinematic-Neural Deformation** approach. Our method is built on a decoupled Core-Shell Gaussian architecture, where a dynamic, FLAME-based Core provides the geometric foundation, and a static Shell captures high-frequency details. This strategic separation enables our key innovation: a hybrid animation strategy that marries a zero-cost geometric binding with a hyper-optimized neural residual network. In a one-time setup, each shell point is bound to the core mesh via barycentric coordinates (KNN, $K = 3$). During animation, these coordinates first reconstruct the shell's **gross motion** by following the LBS-driven core. Subsequently, a lightweight MLP, conditioned on the animation parameters, predicts fine-grained residual modulations to **position, color, and opacity**, capturing subtle non-linear dynamics. This hybrid design preserves the expressiveness of neural deformation while maintaining the near-zero per-frame overhead of a purely kinematic pipeline, enabling seamless, high-speed animation from either speech or video.

Our contributions are as follows:

- We introduce a decoupled Core-Shell Gaussian representation that fuses a FLAME-driven core with a separately generated shell, achieving competitive reconstruction quality.
- We present a one-shot 3D system for animation from **either speech or video**, powered by a **Hybrid Kinematic-Neural architecture**. This approach uses a zero-cost geometric binding for robust gross motion, while a lightweight residual MLP captures complex, non-linear facial dynamics with near-zero per-frame neural overhead.
- By sidestepping common architectural bottlenecks, FLARE achieves state-of-the-art rendering speeds (over 600 FPS for motion-driven, 350 FPS for audio-driven), offering a practical and scalable solution for deploying high-fidelity avatars in latency-sensitive applications.

2 Related Works

The creation of animatable head avatars has evolved along two primary research trajectories: 2D generative synthesis and 3D-aware reconstruction. Our work, FLARE, positions itself within the 3D-aware domain but is motivated by the distinct limitations observed in both fields.

2.1 2D-based Talking Head Synthesis

A dominant line of research leverages the power of 2D generative models, evolving from Generative Adversarial Networks (GANs) Goodfellow et al. (2014) to, more recently, Diffusion Models (DMs) Ho et al. (2020). Early GAN-based approaches pursued two main strategies: modulating the latent space of powerful backbones like StyleGAN Karras et al. (2019) to transfer expressions Zakharov et al. (2019); Burkov et al. (2020); Zhou et al. (2021b), or learning motion representations such as keypoint displacements and dense optical flow to warp the source image Siarohin et al. (2019). The recent advent of DMs has led to state-of-the-art models with unprecedented photorealism Xu et al. (2024); Tian et al. (2024); Zhu et al. (2025); Lin et al. (2025). However, two fundamental limitations persist across these 2D approaches. First, lacking an explicit 3D representation, they struggle with geometric consistency under large pose variations. While some methods introduce 3D Morphable Models (3DMMs) Blanz & Vetter (1999); Paysan et al. (2009); Li et al. (2017); Gerig et al. (2018) as a weak geometric guide Ren et al. (2021); Drobyshev et al. (2022); Yin et al. (2022); Zhang et al. (2023), they remain incapable of true novel-view synthesis. Second, the iterative nature of DMs introduces significant computational latency, making them unsuitable for real-time interactive applications.

2.2 3D-Aware Head Avatars

Within the 3D domain, a central challenge is to simultaneously achieve generalization, high fidelity, and real-time performance. This has led to two main branches of research, with recent works increasingly favoring 3D Gaussian Splatting (3DGS) Kerbl et al. (2023) for its rendering efficiency.

Person-Specific Avatars. A significant body of work focuses on creating high-quality avatars for a specific individual. These methods typically require extensive training data, such as multi-view images or a monocular video of the target person Athar et al. (2022); Bai et al. (2023); Gao et al. (2022); Gafni et al. (2021); Guo et al. (2021); Hong et al. (2021); Kirschstein et al. (2023); Park et al. (2021a;b); Tretschk et al. (2021); Zhang et al. (2024); Zhao et al. (2024); Zheng et al. (2023). Recent advances in this area, such as GaussianTalker Cho et al. (2024), GaussianSpeech Aneja et al. (2024) and GaussianAvatars Qian et al. (2024), have leveraged 3DGS to achieve photorealistic rendering at real-time speeds. However, their inability to generalize to unseen identities makes them impractical for one-shot use cases.

Generalizable One-Shot Avatars. This line of work, which directly aligns with our goal, reveals a landscape of shifting performance bottlenecks. We analyze them by their driving modality:

1) Motion-Driven Avatars: This is the most common setup. While early NeRF-based methods Mildenhall et al. (2020) were limited by slow rendering, the recent shift to 3DGS has exposed a new trade-off Saunders et al. (2025). For instance, SEGA Guo et al. (2025) prioritizes visual quality but operates at a limited $\sim$20 FPS. In contrast, LAM He et al. (2025) breaks this speed barrier ($\sim$280 FPS), but its reliance on a large Transformer backbone to generate the avatar incurs a significant one-time creation cost in memory and computation. GAGAvatar Chu & Harada (2024) faces a different bottleneck, capped at $\sim$70 FPS by a mandatory neural post-processing network for per-frame refinement. This reliance on complex, computationally-heavy neural architectures, whether for one-shot generation or per-frame animation, presents a critical barrier to a lightweight and scalable deployment pipeline.

2) Audio-Driven Avatars: The performance challenge is even more acute for audio-driven animation, a critical modality for interactive avatars. The leading 3D one-shot method in this domain, the NeRF-based Real3D-Portrait Ye et al. (2024), operates at a non-interactive 4.5 FPS. This severe performance deficit is largely due to its reliance on both a slow NeRF renderer and an expensive learned "motion adapter" module that must be evaluated every frame. To date, a high-performance, one-shot, audio-driven 3D avatar solution remains an open problem.

In contrast, FLARE addresses these specific gaps. By replacing the learned, architecturally-complex deformation modules with a lightweight, zero-inference-cost kinematic binding, our framework achieves state-of-the-art speeds for *both* motion- and audio-driven tasks, while maintaining a simple design compatible with classic graphics pipelines.

3 METHOD

The core idea of our method is to decouple the process into two distinct stages: a one-shot **Reconstruction** stage and a real-time **Animation** stage. In the reconstruction stage, a complete and animatable avatar is generated from a single source image. Subsequently, in the animation stage, this avatar is driven by motion or audio signals. This separation of concerns allows us to replace computationally expensive, per-frame neural deformation with a highly efficient animation process. The overall architecture is illustrated in Figure 1.

3.1 PRELIMINARY

Our approach is built upon two foundational technologies, strategically chosen for their complementary strengths: the FLAME parametric model provides a robust kinematic prior for animation, while 3D Gaussian Splatting (3DGS) offers a path to real-time, high-fidelity rendering.

3D Face Representation. Our 3D face representation is based on the FLAME Li et al. (2017), a powerful parametric model that uses a template mesh with $N = 5,023$ vertices and a skeleton of $K = 4$ joints. It parameterizes head geometry and motion through a set of vectors $\{\boldsymbol{\beta}, \boldsymbol{\psi}, \boldsymbol{\theta}\}$. Specifically, $\boldsymbol{\beta} \in \mathbb{R}^{300}$ serves as the shape parameter controlling identity, $\boldsymbol{\psi} \in \mathbb{R}^{100}$ is the expression parameter governing facial muscle movements, and $\boldsymbol{\theta} \in \mathbb{R}^{3K+3}$ is the pose parameter defining the head's rigid transformation. A complete 3D face mesh is generated via the function $M(\boldsymbol{\beta}, \boldsymbol{\theta}, \boldsymbol{\psi}) = W\left(T_P(\boldsymbol{\beta}, \boldsymbol{\theta}, \boldsymbol{\psi}), \mathrm{J}(\boldsymbol{\beta}), \boldsymbol{\theta}, \mathcal{W}\right)$. This process involves first applying shape and expression blendshapes to produce a base mesh, and then using a standard skinning function W to articulate the vertices around skeleton joints $\mathbf{J}$ based on skinning weights $\mathcal{W}$. In our work, we introduce a lightweight refinement to the base model to include internal oral structures, as detailed in Appendix C.

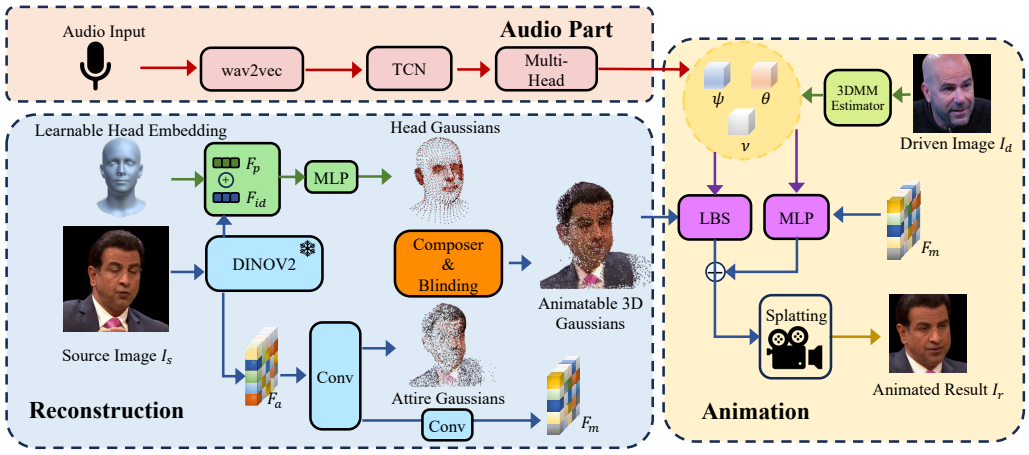


Figure 1: Framework overview. Our one-shot method reconstructs a decoupled Core-Shell avatar and then animates it. From a source image, we generate a FLAME-based Core, a Shell for details, and a modulation feature map $\mathbf{F}_m$. The avatar is driven by FLAME parameters from audio or video. The animation is achieved via a Hybrid Kinematic-Neural mechanism: the Core is driven by LBS, while the Shell's motion is reconstructed via geometric binding and then refined by an expression-conditioned MLP using $\mathbf{F}_m$. The final result is rendered via splatting.

3D Gaussian Splatting. For real-time rendering, we adopt 3DGS Kerbl et al. (2023), which represents a scene as a collection of 3D Gaussians. Each Gaussian is defined by a position $\boldsymbol{p}$, covariance $\boldsymbol{\Sigma}$, opacity α, and color $\boldsymbol{c}$. The covariance matrix $\boldsymbol{\Sigma}$ is factorized into a rotation $\mathbf{R}$ and a scaling $\mathbf{S}$ matrix. Rendering is performed by projecting the Gaussians onto the 2D plane and alpha-blending them in front-to-back order:

$$\mathbf{C}(\boldsymbol{r}) = \sum_{i \in N} \boldsymbol{c}_i \sigma_i \prod_{j=1}^{i-1} (1 - \sigma_j), \quad \text{where } \sigma_i = \alpha_i G'(\boldsymbol{r}). \tag{1}$$

Here, i indexes the depth-sorted Gaussians, and σ_i is the contribution of the i-th Gaussian at pixel $\boldsymbol{r}$.

3.2 Core-Shell Gaussian Representation

Our objective is to learn a function $\mathcal{F}$ that reconstructs a complete, animatable 3D avatar from a single image $\mathbf{I}_s$. A primary one-shot challenge is to model both the non-rigid facial deformations and the complex, static details like hair. To this end, we propose a **Core-Shell Gaussian representation**. This decomposition is motivated by the distinct properties of these regions: the face itself (**Core**, $\mathcal{G}_{core}$) undergoes complex deformations best captured by a structured model, while peripheral elements like hair (**Shell**, $\mathcal{G}_{shell}$) have a complex but largely static topology relative to the head. As illustrated in Figure 1, this approach decomposes the full avatar into the union of these two sets, $\mathcal{G} = \mathcal{G}_{core} \cup \mathcal{G}_{shell}$. Each Gaussian $g \in \mathcal{G}$ is parameterized by its position $\mathbf{p}$, color $\mathbf{c}$, opacity α, scale $\mathbf{s}$, and rotation $\mathbf{q}$. We use specialized generators to create each set once per identity.

Generation of the Dynamic Core. The Core generator aims to reconstruct the person-specific facial appearance in a canonical space while being anchored to a standard kinematic model. We leverage the parametric FLAME model as a strong 3D prior, topologically anchoring the Core Gaussians, $\mathcal{G}_{core}$, to its N_v vertices, $\mathbf{P}_{flame} \in \mathbb{R}^{N_v \times 3}$.

To condition the generation on the source image $\mathbf{I}_s$, we extract a global identity feature $\mathbf{F}_{id}$ using a frozen DINOv2 Oquab et al. (2023) encoder. We also associate each vertex with a learnable, person-agnostic embedding from a parameter matrix $\mathbf{F}_p \in \mathbb{R}^{N_v \times D_{feat}}$, which provides the capacity to learn generic, vertex-specific details. Our feature fusion mechanism is designed for efficiency: the global feature $\mathbf{F}_{id}$ is projected and then added to each vertex's embedding in $\mathbf{F}_p$. This fused feature is then concatenated with a view-direction encoding and fed into a lightweight MLP to predict the per-Gaussian attributes:

$$(\mathbf{c}_i, \alpha_i, \mathbf{s}_i, \mathbf{q}_i) = \text{MLP}(\text{concat}(\mathbf{F}_{p,i} + \text{Proj}(\mathbf{F}_{id}), \text{PE}(\mathbf{d}_{view}))),$$

where i indexes the vertex. Critically, to ensure a robust geometric foundation, the positions of the Core Gaussians are tightly anchored to the canonical FLAME vertex locations, with only negligible learnable offsets being predicted. This strict adherence to the FLAME geometry not only provides a robust foundation for animation but also constrains the Core's shape to the well-behaved manifold of the parametric model, preventing unstable geometric artifacts.

Generation of the Static Shell. The Shell, $\mathcal{G}_{shell}$, models high-frequency details like hair. We employ a convolutional network that takes 2D image features $\mathbf{F}_a$ from DINOv2 as input. Starting from seed points $\mathbf{P}_{plane}$ on a view-aligned plane, the network predicts all Gaussian attributes. Notably, it predicts a 3D position offset $\Delta\mathbf{p}_j$, allowing the initially flat points to form a detailed 3D surface with position $\mathbf{p}_{shell,j} = \mathbf{p}_{plane,j} + \Delta\mathbf{p}_j$. It also predicts a static, per-Gaussian modulation feature $\mathbf{F}_m$, which will later inform the animation MLP of each point's local identity.

3.3 Animation via Hybrid Kinematic-Neural Deformation

To animate the full avatar, we introduce a Hybrid Kinematic-Neural Deformation architecture. This approach combines a zero-cost geometric binding for robust, large-scale motion with a lightweight neural network for subtle, non-linear details. This allows us to capture expressive dynamics without the heavy computational cost of per-frame deformation fields.

Kinematic Binding for Gross Motion. The foundation of our animation is a one-time geometric binding process. For each shell Gaussian $g_j \in \mathcal{G}_{\text{shell}}$, we find its three nearest core vertices $\{\mathbf{v}_i\}_{i=1}^3$ via a K-Nearest Neighbors (KNN, $K = 3$) search on the canonical core mesh. We then compute and cache its representation in the local coordinate frame of the resulting triangle: its **barycentric coordinates** $\{b_k\}_{k=1}^3$ and its **orthogonal height** h_j along the triangle's normal.

During animation, a stream of motion parameters $(\boldsymbol{\psi}, \boldsymbol{\theta})$ drives the core vertices $\mathbf{v}_i$ to their posed positions $\mathbf{v}'_i$ using standard LBS. The shell Gaussian's base position $\mathbf{p}_{base,j}$ is then reconstructed efficiently using these cached local coordinates:

$$\mathbf{p}_{base,j} = \sum_{k=1}^{3} b_k \mathbf{v}'_k + h_j \mathbf{n}', \tag{2}$$

where $\mathbf{n}'$ is the normal of the posed triangle. This kinematic step provides the robust, large-scale **gross motion** for the shell, ensuring it moves cohesively with the deforming face.

Neural Residuals for Fine-grained Dynamics. To model non-linear effects like skin wrinkling and muscle compression, we introduce a lightweight, expression-conditioned MLP. This network predicts fine-grained residual modulations on top of the kinematically driven result. The MLP takes as input the per-Gaussian modulation feature $\mathbf{F}_m$ (from Sec. 3.2), its canonical position $\mathbf{p}_{shell,j}$, and the current expression parameters $\boldsymbol{\psi}$. It then outputs a residual offset $\Delta\mathbf{p}_j$, a color delta $\Delta\mathbf{c}_j$, and an opacity modulator $\Delta\alpha_j$:

$$(\Delta\mathbf{p}_j, \Delta\mathbf{c}_j, \Delta\alpha_j) = \text{MLP}(\mathbf{F}_m, \mathbf{p}_{shell,j}, \boldsymbol{\psi}). \tag{3}$$

The final animated attributes of the shell Gaussian are then computed as:

$$\mathbf{p}'_{shell,j} = \mathbf{p}_{base,j} + \Delta\mathbf{p}_j, \qquad \mathbf{c}'_{shell,j} = \mathbf{c}_{shell,j} + \Delta\mathbf{c}_j, \qquad \alpha'_{shell,j} = \alpha_{shell,j} \cdot \Delta\alpha_j. \tag{4}$$

This hybrid architecture perfectly balances stability and expressiveness. The kinematic base prevents sliding artifacts, while the neural residual efficiently adds high-frequency, dynamic details, all with near-zero per-frame computational overhead due to aggressive pre-computation of identity-specific static components during initialization.

3.4 Audio-to-Motion Generation

While our framework supports high-fidelity reenactment from driving videos, many applications such as virtual assistants and automated content creation demand direct animation from an audio stream. To accommodate this, we introduce an Audio-to-Motion module, which is designed to synthesize realistic and temporally coherent facial animations by mapping a speech signal directly to a sequence of FLAME motion parameters. Our approach deliberately avoids complex conditional inputs like explicit speaker identities or style embeddings, focusing instead on learning a robust, generalized mapping from the acoustic properties of speech to facial dynamics.

The architecture of our Audio-to-Motion module follows a classical encoder-decoder pipeline. We first leverage a pre-trained Wav2Vec Baevski et al. (2020) model as our audio encoder to extract rich, contextualized features from the raw input waveform. This self-supervised speech model excels at capturing the phonetic and prosodic information essential for accurate lip synchronization and natural expression. The resulting feature sequence is then resampled via linear interpolation to precisely match the temporal resolution of the target animation, ensuring a frame-perfect alignment between the audio and motion modalities.

The core of our motion synthesis is a temporal decoder built upon a Temporal Convolutional Network (TCN). This TCN processes the aligned audio features to model long-range temporal dependencies, which is critical for producing smooth and natural motion sequences. To further enhance temporal continuity and prevent discontinuities between generated clips, we condition the decoder on the motion from the preceding segment. Specifically, the motion parameters from the last frame of the previous clip are concatenated with the current audio features, providing the TCN with crucial context about the animation's recent history. Finally, a lightweight MLP-based projection head maps the TCN's output features to the final FLAME parameter space, yielding a complete sequence of expression and pose coefficients ready for driving the Gaussian avatar. This streamlined design enables efficient, parallel generation of the entire motion sequence from the audio input. Detailed network architectures are provided in the Appendix B.

3.5 Training Strategy and Loss Functions

Our framework is optimized via a two-stage strategy that decouples avatar reconstruction from motion synthesis.

Avatar Generator Training. The avatar generator is trained for one-shot reconstruction, enabling it to render a specific identity in an arbitrary pose. During training, we sample a source image $\mathbf{I}_s$ and a driving image $\mathbf{I}_d$ from videos of the same person. We first use a 3DMM Estimator (Sec. 4.1) to extract the shape parameter $\boldsymbol{\beta}_s$ from $\mathbf{I}_s$, alongside the motion parameters $(\boldsymbol{\psi}_d, \boldsymbol{\theta}_d)$ from $\mathbf{I}_d$. Our generators then create a canonical avatar, from $\mathbf{I}_s$ anchored to the identity $\boldsymbol{\beta}_s$. This avatar is subsequently animated using the driving motion parameters and rendered into the final result $\mathbf{I}_r$.

The entire generator is trained end-to-end by minimizing an objective function that compares the animated result $\mathbf{I}_r$ with the ground-truth driving image $\mathbf{I}_d$. This objective combines three losses: a standard L1 reconstruction loss ($\mathcal{L}_{rec}$), a VGG-based perceptual loss ($\mathcal{L}_p$) Zhang et al. (2018), and an Attachment Loss ($\mathcal{L}_{attach}$) inspired by GAGAvatar Chu & Harada (2024).This attachment loss acts as a crucial geometric regularizer, pulling the generated Shell Gaussians into close proximity with the Core mesh. This pre-positioning is a critical prerequisite for the subsequent geometric binding step, ensuring that the nearest-neighbor search can establish correct and stable correspondences for coherent animation.. The total objective for this stage is:

$$\mathcal{L}_{gen} = \lambda_{rec}\mathcal{L}_{rec} + \lambda_p\mathcal{L}_p + \lambda_{attach}\mathcal{L}_{attach}, \tag{5}$$

$$\mathcal{L}_{attach} = \sum_{\mathbf{p}_{shell} \in \mathcal{P}_{shell}} \min_{\mathbf{p}_{core} \in \mathcal{P}_{core}} \|\mathbf{p}_{shell} - \mathbf{p}_{\text{core}}\|_2^2. \tag{6}$$

Audio-to-Motion Module Training. With the generator frozen, the audio-to-motion module is trained on pairs of audio clips and their corresponding ground truth FLAME parameters. The objective combines geometric and perceptual losses. Geometric supervision consists of an L2 loss on the predicted parameters ($\mathcal{L}_{param}$) and on the resulting 3D vertices ($\mathcal{L}_{vert}$), with an increased weight on the lip region. To complement this, we incorporate a perceptual audio-visual synchronization loss, $\mathcal{L}_{sync}$. For this, the predicted motion is used to animate the pre-trained avatar, and the rendered clip is evaluated against the input audio by a pre-trained SyncNet expert Chung & Zisserman (2016). This directly penalizes perceptual audio-lip discrepancies in the final rendered output, thereby correcting for subtle inaccuracies in the predicted FLAME parameters that geometric losses often miss. The total objective for this stage is:

$$\mathcal{L}_{audio} = \lambda_{param}\mathcal{L}_{param} + \lambda_{vert}\mathcal{L}_{vert} + \lambda_{sync}\mathcal{L}_{sync}. \tag{7}$$



Figure 2: Cross-identity qualitative results on the VFHQ Xie et al. (2022) dataset. Compared with baseline methods, our method has accurate expressions and rich details. More visual results can be found at `https://anonymous.4open.science/r/FLARE-30F4`.

Table 1: Same/Cross-identity reenactment quantitative results on the VFHQ dataset.

Method	Self Reenactment							Cross Reenactment			Inference Speed
	PSNR↑	SSIM↑	LPIPS↓	CSIM↑	AED↓	APD↓	AKD↓	CSIM↑	AED↓	APD↓	FPS↑
StyleHeat Yin et al. (2022)	19.95	0.726	0.211	0.537	0.199	0.385	7.659	0.407	0.279	0.551	19.82
ROME Khakhulin et al. (2022)	19.96	0.786	0.192	0.701	0.138	0.186	4.986	0.530	0.259	0.277	11.21
OTAvatar Ma et al. (2023)	17.65	0.563	0.294	0.465	0.234	0.545	18.19	0.364	0.324	0.678	0.12
HideNeRF Li et al. (2023a)	19.79	0.768	0.180	0.787	0.143	0.361	7.254	0.514	0.277	0.527	9.73
GOHA Li et al. (2023b)	20.15	0.770	0.149	0.664	0.176	0.173	6.272	0.518	0.274	0.261	6.57
CVTHead Ma et al. (2024)	18.43	0.706	0.317	0.504	0.186	0.224	5.678	0.374	0.261	0.311	18.09
GPAvatar Chu et al. (2024)	21.04	0.807	0.150	0.772	0.132	0.189	4.226	0.564	0.255	0.328	16.86
Real3DPortrait Ye et al. (2024)	20.88	0.780	0.154	0.801	0.150	0.268	5.971	0.663	0.296	0.411	4.55
Portrait4D Deng et al. (2024a)	20.35	0.741	0.191	0.765	0.144	0.205	4.854	0.596	0.286	0.258	9.49
Portrait4D-v2 Deng et al. (2024b)	21.34	0.791	0.144	0.803	0.117	0.187	3.749	0.656	0.268	0.273	9.62
GAGAvatar Chu & Harada (2024)	21.83	0.818	0.122	0.816	**0.111**	**0.135**	**3.349**	0.633	0.253	**0.247**	67.12
LAM-20K He et al. (2025) [1]	21.32	0.812	0.131	0.812	0.133	0.149	3.484	0.638	0.252	0.354	547.51
Ours	**21.90**	**0.823**	**0.116**	**0.829**	0.112	0.137	3.410	**0.664**	**0.246**	0.332	**630.07**

4 EXPERIMENTS

To validate the effectiveness and efficiency of FLARE, we conduct a comprehensive set of experiments in this section. We present quantitative and qualitative comparisons against state-of-the-art methods on both motion-driven reenactment and audio-driven talking head generation tasks.

4.1 IMPLEMENTATION DETAILS

Datasets. Our training methodology leverages a diverse collection of datasets to ensure both high fidelity and robust generalization. The primary source is the VFHQ Xie et al. (2022) dataset, chosen

for its high-quality video clips ideal for visual synthesis. To enhance robustness for different tasks, we further augment this data. Specifically, for video-driven animation, we incorporate multi-view videos from MEAD Wang et al. (2020) and PanoHead An et al. (2023) to compensate for the limited head pose variations in VFHQ. For the audio-driven task, we additionally use the THFP Sun et al. (2024) dataset to improve the model's generalization across diverse speakers.

Data Pre-processing. All collected datasets undergo a unified, multi-stage pre-processing pipeline. First, to ensure high audio-lip synchrony for our primary data, we filter the VFHQ clips using a pre-trained SyncNet Chung & Zisserman (2016), which yields our main training set of approximately 2.32 million frames. Concurrently, the multi-view data from MEAD and PanoHead, intended for view diversity, is constructed into a supplemental set of 21,800 images across 5,030 identities without SyncNet filtering. All selected frames are then processed by removing backgrounds and resizing to 512×512. Finally, we extract 3DMM parameters to obtain high-quality annotations: we use DECA Feng et al. (2021) for pose ($\boldsymbol{\theta}$), MICA zie (2022) for shape ($\boldsymbol{\beta}$), and EMOCA Danecek et al. (2022) for expression ($\boldsymbol{\psi}$). Further details are provided in the supplementary material.

Training Details. Our training protocol is as follows. The framework, implemented in PyTorch, is trained on 8 NVIDIA A100 GPUs. For optimization, we employ the Adam optimizer with a learning rate of 1×10^{-4} for all learnable parameters. Detailed diagrams of the network architectures and a full tabulation of hyper-parameters are provided in the Appendix B.

4.2 Qualitative Results

Figure 2 presents a visual comparison of our method against several state-of-the-art approaches under the motion-driven reenactment setting. The results indicate that our Core-Shell representation effectively reconstructs avatars with competitive visual fidelity. Compared to methods that primarily model the parametric face region, our framework also captures non-parametric details such as hair, beards, and accessories from the source image. In terms of animation, our model is shown to transfer facial movements from the driving source, particularly around the eyes and mouth. Our method also maintains identity consistency and image quality under varying head rotations. These results are achieved while operating at real-time rendering speeds. Figure 3 showcases the results for the audio-driven task. These visualizations demonstrate that our method preserves the subject's identity while generating smooth and natural head motions synchronized with the audio input.



Figure 3: Visual results of our audio-driven results.

We quantitatively evaluate FLARE in both motion-driven (MD) and audio-driven (AD) scenarios, comparing it against leading one-shot methods. Detailed information regarding the test data and evaluation protocols can be found in the supplementary material.

Evaluation Metrics. We evaluate our method with a comprehensive set of metrics, each targeting a specific aspect of performance. For **Reconstruction Fidelity**, we measure the visual similarity between generated and ground-truth images using the Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index (SSIM), and the perceptual LPIPS Zhang et al. (2018). To assess **Identity Preservation**, we quantify how well the avatar resembles the source person by calculating the Cosine Similarity (CSIM) of ArcFace embeddings Deng et al. (2019a) and the Fréchet Inception Distance (FID). For **Animation Accuracy**, we measure the correctness of the synthesized motion via the Average Expression Distance (AED), Average Pose Distance (APD) Deng et al. (2019b), and Average Keypoint Distance (AKD) Bulat & Tzimiropoulos (2017). Finally, for the audio-driven task, **Audio-Lip Synchronization** is evaluated with the Sync score Chung & Zisserman (2016). Detailed descriptions are provided in Appendix E.

Motion-Driven Reenactment. As shown in Table 1, FLARE demonstrates strong performance across both reenactment tasks. The high scores in PSNR, SSIM, and LPIPS indicate that our Core-Shell representation achieves high visual fidelity, while a strong CSIM score confirms robust identity

preservation. Our animation accuracy metrics (AED, APD), while highly competitive, trail slightly behind GAGAvatar Chu & Harada (2024). This is an expected and direct consequence of our differing architectural philosophies for neural refinement. GAGAvatar employs a more complex, and likely computationally heavier, neural post-processing network that acts on the entire avatar to capture extremely subtle muscle movements. In contrast, our Hybrid Kinematic-Neural architecture uses a hyper-efficient, lightweight MLP that predicts only the essential residual deformations for the Shell. This design deliberately captures the most significant non-linear dynamics while meticulously avoiding the computational overhead of a more powerful network. This strategic compromise is precisely what enables our performance leap: we trade a marginal, almost imperceptible difference in micro-expressions for a nearly order-of-magnitude increase in rendering speed.

Audio-Driven Talking Head Generation. As shown in Table 2, our method demonstrates a state-of-the-art balance of identity preservation and expression accuracy. Specifically, FLARE achieves the highest CSIM score (0.893), confirming our Core-Shell representation robustly maintains person-specific appearance. Regarding lip synchronization, our Sync score, while competitive, is not the highest. This is an expected and direct consequence of using the parametric FLAME model as an intermediate representation. While this low-dimensional model is key to our efficiency and leads to state-of-the-art expression accuracy (AED), its predefined blendshape space inherently cannot capture the infinite pixel-level lip variations that the SyncNet metric is highly sensitive to. In contrast, methods with higher Sync scores often employ more direct pixel-level generation or warping, a trade-off our method deliberately avoids for structural consistency and extreme speed.

Table 2: Audio-driven quantitative results.

Method	CSIM	FID	AED	Sync	FPS
PC-AVS Zhou et al. (2021a)	0.327	94.47	0.296	**7.041**	<u>17.87</u>
Real3DPortrait Ye et al. (2024)	<u>0.882</u>	<u>50.65</u>	0.281	3.654	3.77
EchoMimic Chen et al. (2025)	0.866	53.84	0.276	4.081	0.23
KDTalker Yang et al. (2025)	0.837	**46.01**	<u>0.264</u>	<u>6.025</u>	15.34
Ours	**0.893**	66.74	**0.257**	5.886	**353.1**

Despite this more demanding evaluation condition, FLARE demonstrates strong performance in motion synthesis and efficiency. Our method achieves the best expression accuracy with a leading AED score of 0.257, highlighting the effectiveness of our audio-to-motion module. Most notably, our framework operates at **353.1 FPS**, substantially outperforming the inference speed of other generalizable methods. This combination of real-time performance and high-quality motion synthesis makes FLARE a practical solution for deploying expressive, pose-versatile avatars in latency-sensitive applications.

4.3 Ablation Studies

Table 3 and Figure 4 validate our key design choices. Removing the global identity feature $\mathbf{F}_{id}$ produces a generic-looking face, whereas removing the positional feature $\mathbf{F}_p$ loses fine-grained geometry, confirming their complementary roles. Removing the Shell Gaussians (w/o SG) yields a "bald" avatar. Removing $\mathcal{L}_{attach}$ causes a similar collapse: without this geometric anchor, the under-constrained Shell generation fails to converge and effectively discards the Shell. Finally, disabling geometric binding (w/o GB) produces incoherent, floating motion, whereas removing dynamic refinement (w/o DR) yields a rigid, "rubber-mask" animation lacking non-linear subtleties. These results confirm that each component is indispensable for high-fidelity, expressive animation.



Figure 4: Qualitative results of our ablation study.

Table 3: Ablation study on the VFHQ dataset.

Method	PSNR	SSIM	LPIPS	CSIM	AED	APD	AKD
w/o $\mathbf{F}_p$	20.85	0.780	0.179	0.767	0.120	0.179	4.011
w/o $\mathbf{F}_{id}$	20.98	0.796	0.179	0.778	0.111	0.150	3.883
w/o SG	16.77	0.668	0.393	0.115	0.213	0.210	4.072
w/o GB	21.18	0.798	0.184	0.811	0.118	0.145	3.714
w/o L_{attach}	16.54	0.667	0.391	0.105	0.212	0.214	4.066
w/o NR	21.04	0.797	0.181	0.805	0.122	0.164	3.767
Ours	21.90	0.823	0.116	0.829	0.112	0.137	3.410

5 Conclusion

We presented FLARE, a one-shot framework for speech- or video-driven 3D head avatars. Its Core-Shell representation combines near-zero-cost geometric binding for robust base motion with a lightweight neural residual MLP for expressive details. This design achieves state-of-the-art visual quality and identity preservation while operating at over **350 FPS** for audio-driven synthesis and over **600 FPS** for motion-driven reenactment. FLARE thus offers a practical and scalable solution for deploying high-fidelity, real-time avatars in latency-sensitive applications such as conversational AI.

REFERENCES

Towards Metrical Reconstruction of Human Faces, 2022.

Sizhe An, Hongyi Xu, Yichun Shi, Guoxian Song, Umit Ogras, and Linjie Luo. Panohead: Geometry-aware 3d full-head synthesis in 360°, 2023.

Shivangi Aneja, Artem Sevastopolsky, Tobias Kirschstein, Justus Thies, Angela Dai, and Matthias Nießner. Gaussianspeech: Audio-driven gaussian avatars. *CoRR*, abs/2411.18675, 2024. doi: 10.48550/ARXIV.2411.18675. URL https://doi.org/10.48550/arXiv.2411.18675.

ShahRukh Athar, Zexiang Xu, Kalyan Sunkavalli, Eli Shechtman, and Zhixin Shu. Rignerf: Fully controllable neural 3d portraits. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pp. 20332–20341. IEEE, 2022. doi: 10.1109/CVPR52688.2022.01972. URL https://doi.org/10.1109/CVPR52688.2022.01972.

Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. wav2vec 2.0: A framework for self-supervised learning of speech representations. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/92d1e1eb1cd6f9fba3227870bb6d7f07-Abstract.html.

Yunpeng Bai, Yanbo Fan, Xuan Wang, Yong Zhang, Jingxiang Sun, Chun Yuan, and Ying Shan. High-fidelity facial avatar reconstruction from monocular video with generative priors. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023*, pp. 4541–4551. IEEE, 2023. doi: 10.1109/CVPR52729.2023.00441. URL https://doi.org/10.1109/CVPR52729.2023.00441.

Volker Blanz and Thomas Vetter. A morphable model for the synthesis of 3d faces. In Warren N. Waggenspack (ed.), *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 1999, Los Angeles, CA, USA, August 8-13, 1999*, pp. 187–194. ACM, 1999. URL https://dl.acm.org/citation.cfm?id=311556.

Adrian Bulat and Georgios Tzimiropoulos. How far are we from solving the 2d & 3d face alignment problem?(and a dataset of 230,000 3d facial landmarks). In *Proceedings of the IEEE international conference on computer vision*, pp. 1021–1030, 2017.

Egor Burkov, Igor Pasechnik, Artur Grigorev, and Victor Lempitsky. Neural head reenactment with latent pose descriptors. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 13786–13795, 2020.

Zhiyuan Chen, Jiajiong Cao, Zhiquan Chen, Yuming Li, and Chenguang Ma. Echomimic: Lifelike audio-driven portrait animations through editable landmark conditions. In Toby Walsh, Julie Shah, and Zico Kolter (eds.), *AAAI-25, Sponsored by the Association for the Advancement of Artificial Intelligence, February 25 - March 4, 2025, Philadelphia, PA, USA*, pp. 2403–2410. AAAI Press, 2025. doi: 10.1609/AAAI.V39I3.32241. URL https://doi.org/10.1609/aaai.v39i3.32241.

Kyusun Cho, Joungbin Lee, Heeji Yoon, Yeobin Hong, Jaehoon Ko, Sangjun Ahn, and Seungryong Kim. Gaussiantalker: Real-time high-fidelity talking head synthesis with audio-driven 3d gaussian splatting. *CoRR*, abs/2404.16012, 2024. doi: 10.48550/ARXIV.2404.16012. URL https://doi.org/10.48550/arXiv.2404.16012.

Xuangeng Chu and Tatsuya Harada. Generalizable and animatable gaussian head avatar. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=gVM2AZ5xA6.

Xuangeng Chu, Yu Li, Ailing Zeng, Tianyu Yang, Lijian Lin, Yunfei Liu, and Tatsuya Harada. Gpavatar: Generalizable and precise head avatar from image(s). In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=hgehGq2bDv.

J. S. Chung and A. Zisserman. Out of time: automated lip sync in the wild. In *Workshop on Multi-view Lip-reading, ACCV*, 2016.

Radek Danecek, Michael J. Black, and Timo Bolkart. EMOCA: Emotion driven monocular face capture and animation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20311–20322, 2022.

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*, pp. 4690–4699, 2019a.

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*, pp. 0–0, 2019b.

Yu Deng, Duomin Wang, Xiaohang Ren, Xingyu Chen, and Baoyuan Wang. Portrait4d: Learning one-shot 4d head avatar synthesis using synthetic data. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, pp. 7119–7130. IEEE, 2024a. doi: 10.1109/CVPR52733.2024.00680. URL https://doi.org/10.1109/CVPR52733.2024.00680.

Yu Deng, Duomin Wang, and Baoyuan Wang. Portrait4d-v2: Pseudo multi-view data creates better 4d head synthesizer. In Ales Leonardis, Elisa Ricci, Stefan Roth, Olga Russakovsky, Torsten Sattler, and Gül Varol (eds.), *Computer Vision - ECCV 2024 - 18th European Conference, Milan, Italy, September 29-October 4, 2024, Proceedings, Part XVII*, volume 15075 of *Lecture Notes in Computer Science*, pp. 316–333. Springer, 2024b. doi: 10.1007/978-3-031-72643-9_19. URL https://doi.org/10.1007/978-3-031-72643-9_19.

Nikita Drobyshev, Jenya Chelishev, Taras Khakhulin, Aleksei Ivakhnenko, Victor Lempitsky, and Egor Zakharov. Megaportraits: One-shot megapixel neural head avatars. In João Magalhães, Alberto Del Bimbo, Shin'ichi Satoh, Nicu Sebe, Xavier Alameda-Pineda, Qin Jin, Vincent Oria, and Laura Toni (eds.), *MM '22: The 30th ACM International Conference on Multimedia, Lisboa, Portugal, October 10 - 14, 2022*, pp. 2663–2671. ACM, 2022. doi: 10.1145/3503161.3547838. URL https://doi.org/10.1145/3503161.3547838.

Yao Feng, Haiwen Feng, Michael J. Black, and Timo Bolkart. Learning an animatable detailed 3D face model from in-the-wild images. volume 40, 2021. URL https://doi.org/10.1145/3450626.3459936.

Guy Gafni, Justus Thies, Michael Zollhöfer, and Matthias Nießner. Dynamic neural radiance fields for monocular 4d facial avatar reconstruction. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pp. 8649–8658. Computer Vision Foundation / IEEE, 2021. doi: 10.1109/CVPR46437.2021.00854. URL https://openaccess.thecvf.com/content/CVPR2021/html/Gafni_Dynamic_Neural_Radiance_Fields_for_Monocular_4D_Facial_Avatar_Reconstruction_CVPR_2021_paper.html.

Xuan Gao, Chenglai Zhong, Jun Xiang, Yang Hong, Yudong Guo, and Juyong Zhang. Reconstructing personalized semantic facial nerf models from monocular video. *ACM Trans. Graph.*, 41(6): 200:1–200:12, 2022. doi: 10.1145/3550454.3555501. URL https://doi.org/10.1145/3550454.3555501.

Thomas Gerig, Andreas Morel-Forster, Clemens Blumer, Bernhard Egger, Marcel Lüthi, Sandro Schönborn, and Thomas Vetter. Morphable face models - an open framework. In *13th IEEE International Conference on Automatic Face & Gesture Recognition, FG 2018, Xi'an, China, May 15-19, 2018*, pp. 75–82. IEEE Computer Society, 2018. doi: 10.1109/FG.2018.00021. URL https://doi.org/10.1109/FG.2018.00021.

Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

Chen Guo, Zhuo Su, Jian Wang, Shuang Li, Xu Chang, Zhaohu Li, Yang Zhao, Guidong Wang, and Ruqi Huang. SEGA: drivable 3d gaussian head avatar from a single image. *CoRR*, abs/2504.14373, 2025. doi: 10.48550/ARXIV.2504.14373. URL https://doi.org/10.48550/arXiv.2504.14373.

Yudong Guo, Keyu Chen, Sen Liang, Yong-Jin Liu, Hujun Bao, and Juyong Zhang. Ad-nerf: Audio driven neural radiance fields for talking head synthesis. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pp. 5764–5774. IEEE, 2021. doi: 10.1109/ICCV48922.2021.00573. URL https://doi.org/10.1109/ICCV48922.2021.00573.

Yisheng He, Xiaodong Gu, Xiaodan Ye, Chao Xu, Zhengyi Zhao, Yuan Dong, Weihao Yuan, Zilong Dong, and Liefeng Bo. Lam: Large avatar model for one-shot animatable gaussian head. In *Proceedings of the Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers*, pp. 1–13, 2025.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, Günter Klambauer, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a nash equilibrium. *CoRR*, abs/1706.08500, 2017. URL http://arxiv.org/abs/1706.08500.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Yang Hong, Bo Peng, Haiyao Xiao, Ligang Liu, and Juyong Zhang. Headnerf: A real-time nerf-based parametric head model. *CoRR*, abs/2112.05637, 2021. URL https://arxiv.org/abs/2112.05637.

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*, pp. 4401–4410, 2019.

Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):139:1–139:14, 2023. doi: 10.1145/3592433. URL https://doi.org/10.1145/3592433.

Taras Khakhulin, Vanessa Sklyarova, Victor Lempitsky, and Egor Zakharov. Realistic one-shot mesh-based head avatars. In Shai Avidan, Gabriel J. Brostow, Moustapha Cissé, Giovanni Maria Farinella, and Tal Hassner (eds.), *Computer Vision - ECCV 2022 - 17th European Conference, Tel Aviv, Israel, October 23-27, 2022, Proceedings, Part II*, volume 13662 of *Lecture Notes in Computer Science*, pp. 345–362. Springer, 2022. doi: 10.1007/978-3-031-20086-1_20. URL https://doi.org/10.1007/978-3-031-20086-1_20.

Tobias Kirschstein, Shenhan Qian, Simon Giebenhain, Tim Walter, and Matthias Nießner. Nersemble: Multi-view radiance field reconstruction of human heads. *ACM Trans. Graph.*, 42(4):161:1–161:14, 2023. doi: 10.1145/3592455. URL https://doi.org/10.1145/3592455.

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.

Weichuang Li, Longhao Zhang, Dong Wang, Bin Zhao, Zhigang Wang, Mulin Chen, Bang Zhang, Zhongjian Wang, Liefeng Bo, and Xuelong Li. One-shot high-fidelity talking-head synthesis with deformable neural radiance field. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023*, pp. 17969–17978. IEEE, 2023a. doi: 10.1109/CVPR52729.2023.01723. URL https://doi.org/10.1109/CVPR52729.2023.01723.

Xueting Li, Shalini De Mello, Sifei Liu, Koki Nagano, Umar Iqbal, and Jan Kautz. Generalizable one-shot neural head avatar. *CoRR*, abs/2306.08768, 2023b. doi: 10.48550/ARXIV.2306.08768. URL https://doi.org/10.48550/arXiv.2306.08768.

Gaojie Lin, Jianwen Jiang, Jiaqi Yang, Zerong Zheng, and Chao Liang. Omnihuman-1: Rethinking the scaling-up of one-stage conditioned human animation models. *CoRR*, abs/2502.01061, 2025. doi: 10.48550/ARXIV.2502.01061. URL https://doi.org/10.48550/arXiv.2502.01061.

Haoyu Ma, Tong Zhang, Shanlin Sun, Xiangyi Yan, Kun Han, and Xiaohui Xie. Cvthead: One-shot controllable head avatar with vertex-feature transformer. In *IEEE/CVF Winter Conference on Applications of Computer Vision, WACV 2024, Waikoloa, HI, USA, January 3-8, 2024*, pp. 6119–6129. IEEE, 2024. doi: 10.1109/WACV57701.2024.00602. URL https://doi.org/10.1109/WACV57701.2024.00602.

Zhiyuan Ma, Xiangyu Zhu, Guojun Qi, Zhen Lei, and Lei Zhang. Otavatar: One-shot talking face avatar with controllable tri-plane rendering. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023*, pp. 16901–16910. IEEE, 2023. doi: 10.1109/CVPR52729.2023.01621. URL https://doi.org/10.1109/CVPR52729.2023.01621.

Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm (eds.), *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part I*, volume 12346 of *Lecture Notes in Computer Science*, pp. 405–421. Springer, 2020. doi: 10.1007/978-3-030-58452-8_24. URL https://doi.org/10.1007/978-3-030-58452-8_24.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

Keunhong Park, Utkarsh Sinha, Jonathan T. Barron, Sofien Bouaziz, Dan B. Goldman, Steven M. Seitz, and Ricardo Martin-Brualla. Nerfies: Deformable neural radiance fields. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pp. 5845–5854. IEEE, 2021a. doi: 10.1109/ICCV48922.2021.00581. URL https://doi.org/10.1109/ICCV48922.2021.00581.

Keunhong Park, Utkarsh Sinha, Peter Hedman, Jonathan T. Barron, Sofien Bouaziz, Dan B. Goldman, Ricardo Martin-Brualla, and Steven M. Seitz. Hypernerf: a higher-dimensional representation for topologically varying neural radiance fields. *ACM Trans. Graph.*, 40(6):238:1–238:12, 2021b. doi: 10.1145/3478513.3480487. URL https://doi.org/10.1145/3478513.3480487.

Pascal Paysan, Reinhard Knothe, Brian Amberg, Sami Romdhani, and Thomas Vetter. A 3d face model for pose and illumination invariant face recognition. In Stefano Tubaro and Jean-Luc Dugelay (eds.), *Sixth IEEE International Conference on Advanced Video and Signal Based Surveillance, AVSS 2009, 2-4 September 2009, Genova, Italy*, pp. 296–301. IEEE Computer Society, 2009. doi: 10.1109/AVSS.2009.58. URL https://doi.org/10.1109/AVSS.2009.58.

Shenhan Qian, Tobias Kirschstein, Liam Schoneveld, Davide Davoli, Simon Giebenhain, and Matthias Nießner. Gaussianavatars: Photorealistic head avatars with rigged 3d gaussians. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, pp. 20299–20309. IEEE, 2024. doi: 10.1109/CVPR52733.2024.01919. URL https://doi.org/10.1109/CVPR52733.2024.01919.

Yurui Ren, Ge Li, Yuanqi Chen, Thomas H Li, and Shan Liu. Pirenderer: Controllable portrait image generation via semantic neural rendering. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 13759–13768, 2021.

Jack R. Saunders, Charlie Hewitt, Yanan Jian, Marek Kowalski, Tadas Baltrusaitis, Yiye Chen, Darren Cosker, Virginia Estellers, Nicholas Gyde, Vinay P. Namboodiri, and Benjamin E. Lundell. GASP: gaussian avatars with synthetic priors. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2025, Nashville, TN, USA, June 11-15, 2025*, pp. 271–280. Computer Vision Foundation / IEEE, 2025. doi: 10.1109/CVPR52734.2025.00034. URL https://openaccess.thecvf.com/content/CVPR2025/html/Saunders_GASP_Gaussian_Avatars_with_Synthetic_Priors_CVPR_2025_paper.html.

Aliaksandr Siarohin, Stéphane Lathuilière, Sergey Tulyakov, Elisa Ricci, and Nicu Sebe. First order motion model for image animation. *Advances in neural information processing systems*, 32, 2019.

Zhiyao Sun, Tian Lv, Sheng Ye, Matthieu Lin, Jenny Sheng, Yu-Hui Wen, Minjing Yu, and Yong-Jin Liu. Diffposetalk: Speech-driven stylistic 3d facial animation and head pose generation via diffusion models. *ACM Transactions on Graphics (TOG)*, 43(4), 2024. doi: 10.1145/3658221.

Linrui Tian, Qi Wang, Bang Zhang, and Liefeng Bo. EMO: emote portrait alive generating expressive portrait videos with audio2video diffusion model under weak conditions. In Ales Leonardis, Elisa Ricci, Stefan Roth, Olga Russakovsky, Torsten Sattler, and Gül Varol (eds.), *Computer Vision - ECCV 2024 - 18th European Conference, Milan, Italy, September 29-October 4, 2024, Proceedings, Part LXXXIII*, volume 15141 of *Lecture Notes in Computer Science*, pp. 244–260. Springer, 2024. doi: 10.1007/978-3-031-73010-8_15. URL https://doi.org/10.1007/978-3-031-73010-8_15.

Edgar Tretschk, Ayush Tewari, Vladislav Golyanik, Michael Zollhöfer, Christoph Lassner, and Christian Theobalt. Non-rigid neural radiance fields: Reconstruction and novel view synthesis of a dynamic scene from monocular video. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pp. 12939–12950. IEEE, 2021. doi: 10.1109/ICCV48922.2021.01272. URL https://doi.org/10.1109/ICCV48922.2021.01272.

Kaisiyuan Wang, Qianyi Wu, Linsen Song, Zhuoqian Yang, Wayne Wu, Chen Qian, Ran He, Yu Qiao, and Chen Change Loy. Mead: A large-scale audio-visual dataset for emotional talking-face generation. In *ECCV*, 2020.

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.

Sicheng Xu, Guojun Chen, Yu-Xiao Guo, Jiaolong Yang, Chong Li, Zhenyu Zang, Yizhong Zhang, Xin Tong, and Baining Guo. VASA-1: lifelike audio-driven talking faces generated in real time. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang (eds.), *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/014fe398da515cd552fa6e1f33e0565e-Abstract-Conference.html.

Chaolong Yang, Kai Yao, Yuyao Yan, Chenru Jiang, Weiguang Zhao, Jie Sun, Guangliang Cheng, Yifei Zhang, Bin Dong, and Kaizhu Huang. Unlock pose diversity: Accurate and efficient implicit keypoint-based spatiotemporal diffusion for audio-driven talking portrait. *CoRR*, abs/2503.12963, 2025. doi: 10.48550/ARXIV.2503.12963. URL https://doi.org/10.48550/arXiv.2503.12963.

Zhenhui Ye, Tianyun Zhong, Yi Ren, Jiaqi Yang, Weichuang Li, Jiawei Huang, Ziyue Jiang, Jinzheng He, Rongjie Huang, Jinglin Liu, Chen Zhang, Xiang Yin, Zejun Ma, and Zhou Zhao. Real3d-portrait: One-shot realistic 3d talking portrait synthesis. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=7ERQPyR2eb.

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 Shai Avidan, Gabriel J. Brostow, Moustapha Cissé, Giovanni Maria Farinella, and Tal Hassner (eds.), *Computer Vision - ECCV 2022 - 17th European Conference, Tel Aviv, Israel, October 23-27, 2022, Proceedings, Part XVII*, volume 13677 of *Lecture Notes in Computer Science*, pp. 85–101. Springer, 2022. doi: 10.1007/978-3-031-19790-1_6. URL https://doi.org/10.1007/978-3-031-19790-1_6.

Egor Zakharov, Aliaksandra Shysheya, Egor Burkov, and Victor Lempitsky. Few-shot adversarial learning of realistic neural talking head models. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 9459–9468, 2019.

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*, pp. 586–595, 2018.

Wenxuan Zhang, Xiaodong Cun, Xuan Wang, Yong Zhang, Xi Shen, Yu Guo, Ying Shan, and Fei Wang. Sadtalker: Learning realistic 3d motion coefficients for stylized audio-driven single image talking face animation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023*, pp. 8652–8661. IEEE, 2023. doi: 10.1109/CVPR52729.2023.00836. URL https://doi.org/10.1109/CVPR52729.2023.00836.

Zicheng Zhang, Ruobing Zheng, Bonan Li, Congying Han, Tianqi Li, Meng Wang, Tiande Guo, Jingdong Chen, Ziwen Liu, and Ming Yang. Learning dynamic tetrahedra for high-quality talking head synthesis. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, pp. 5209–5219. IEEE, 2024. doi: 10.1109/CVPR52733.2024.00498. URL https://doi.org/10.1109/CVPR52733.2024.00498.

Xiaochen Zhao, Lizhen Wang, Jingxiang Sun, Hongwen Zhang, Jinli Suo, and Yebin Liu. Havatar: High-fidelity head avatar via facial model conditioned neural radiance field. *ACM Trans. Graph.*, 43 (1):6:1–6:16, 2024. doi: 10.1145/3626316. URL https://doi.org/10.1145/3626316.

Yufeng Zheng, Wang Yifan, Gordon Wetzstein, Michael J. Black, and Otmar Hilliges. Pointavatar: Deformable point-based head avatars from videos. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023*, pp. 21057–21067. IEEE, 2023. doi: 10.1109/CVPR52729.2023.02017. URL https://doi.org/10.1109/CVPR52729.2023.02017.

Hang Zhou, Yasheng Sun, Wayne Wu, Chen Change Loy, Xiaogang Wang, and Ziwei Liu. Pose-controllable talking face generation by implicitly modularized audio-visual representation. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pp. 4176–4186. Computer Vision Foundation / IEEE, 2021a. doi: 10.1109/CVPR46437.2021.00416. URL https://openaccess.thecvf.com/content/CVPR2021/html/Zhou_Pose-Controllable_Talking_Face_Generation_by_Implicitly_Modularized_Audio-Visual_Representation_CVPR_2021_paper.html.

Hang Zhou, Yasheng Sun, Wayne Wu, Chen Change Loy, Xiaogang Wang, and Ziwei Liu. Pose-controllable talking face generation by implicitly modularized audio-visual representation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4176–4186, 2021b.

Yongming Zhu, Longhao Zhang, Zhengkun Rong, Tianshu Hu, Shuang Liang, and Zhipeng Ge. INFP: audio-driven interactive head generation in dyadic conversations. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2025, Nashville, TN, USA, June 11-15, 2025*, pp. 10667–10677. Computer Vision Foundation / IEEE, 2025. doi: 10.1109/CVPR52734.2025.00997. URL https://openaccess.thecvf.com/content/CVPR2025/html/Zhu_INFP_Audio-Driven_Interactive_Head_Generation_in_Dyadic_Conversations_CVPR_2025_paper.html.

A Comparison with Related Work

Our work, FLARE, addresses the specific task of creating one-shot, real-time, animatable 3D avatars with dual-modality driving from either motion or speech. While numerous methods target this area, very few that satisfy all constraints are open-source. Therefore, rather than focusing on the exact task implementation of each model, we will concentrate our discussion on methods that are most architecturally and methodologically similar to ours.

GAGAvatar Chu & Harada (2024). Among these, GAGAvatar is the most architecturally similar, as we adopt a comparable approach for generating the core head Gaussians. However, two key distinctions lead to a significant performance advantage. First, whereas GAGAvatar generates its Shell Gaussians from two 2D planes, we generate ours by predicting 3D offsets from a single, view-aligned plane, allowing for more flexible surface discovery. Second, and more critically, GAGAvatar relies on a neural post-processing network to fuse the Core and Shell, while we employ a purely geometric binding process using a K-Nearest Neighbors (KNN) search and barycentric coordinates. This elimination of the per-frame neural bottleneck allows FLARE to achieve a rendering speed approximately an order of magnitude faster.

SEGA Guo et al. (2025). The concept of a dynamic Core and a static Shell is also explored in SEGA. However, the methods used to construct its dynamic Core and static Shell Gaussians are entirely different from ours. Furthermore, SEGA employs a VAE-based network to generate a deformed mesh for animation, a process that constrains its inference speed to approximately 20 FPS, which is on the cusp of real-time standards.

GaussianAvatars Qian et al. (2024). Another method that binds a "shell" to a "core" is GaussianAvatars. Their approach differs fundamentally in the initialization and binding strategy: each of their shell points is assigned a parent vertex on the FLAME mesh *prior* to training. In contrast, our method generates Shell Gaussians freely in 3D space and then uses an attachment loss to pull them into proximity with the Core mesh, after which KNN dynamically finds the optimal triangular anchors. This makes our binding process more data-driven and flexible compared to their pre-determined assignment.

B Detailed Network Architectures

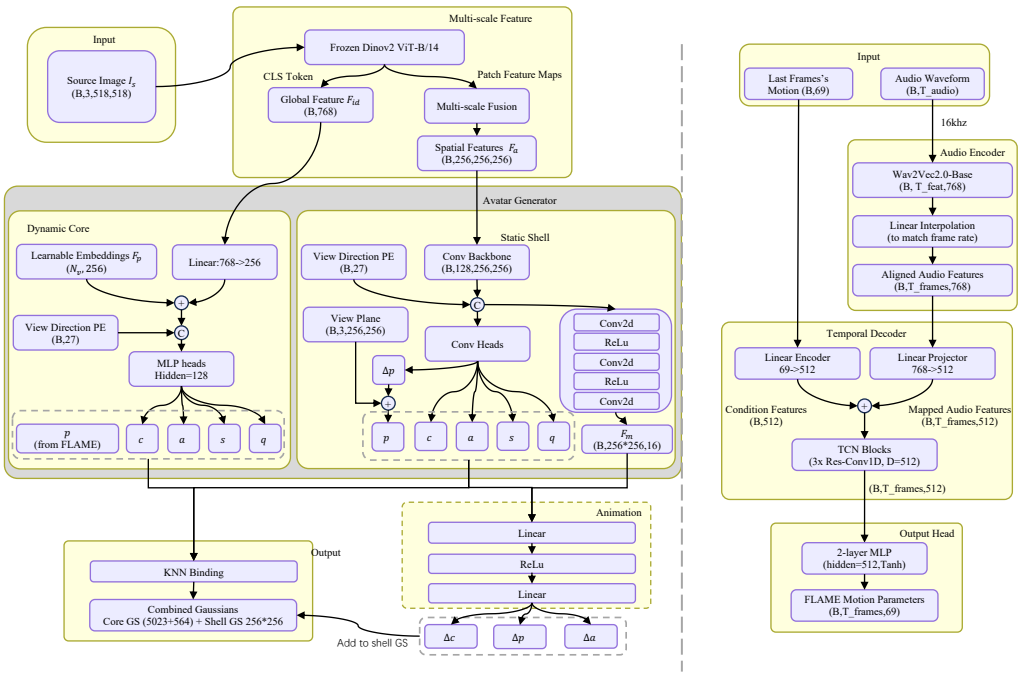
In this section, we provide a comprehensive description of the network architectures used in our FLARE framework. **For a visual overview of the specific layer configurations and tensor dimensions discussed below, please refer to Figure 5.** We detail the three primary components: the Multi-scale Feature Extractor, the Avatar Generator, and the Audio-to-Motion Module.

B.1 Multi-scale Feature Extractor

A critical component of our framework is a sophisticated feature extractor built upon a frozen, pre-trained **DINOv2 (ViT-B/14)** Oquab et al. (2023) backbone, as shown in the top-left of Figure 5(a). Unlike prior works that may use features from a single layer, our extractor is designed to produce a rich, multi-scale spatial feature map by fusing information from different depths of the Vision Transformer. This allows our model to leverage both low-level texture information and high-level semantic information simultaneously.

The feature extraction process is as follows:

1. **Hierarchical Feature Extraction:** We extract patch features from four intermediate layers of the DINOv2 model. These features are then reshaped into 2D spatial feature maps of varying resolutions and channel dimensions.
2. **Feature Projection and Resizing:** Each of the four feature maps is first projected to a target channel dimension (256, 512, 1024, 1024, respectively) using a 1×1 convolution. Subsequently, they are resized to a common spatial resolution using a combination of transposed convolutions and standard convolutions to form a feature pyramid.



(a) (b)

Figure 5: **Detailed Network Architectures. (a) Avatar Generator Architecture.** Illustrates the Multi-scale Feature Extractor and the decoupled Core-Shell generation pipeline. **(b) Audio-to-Motion Module Architecture.** Depicts the encoding of raw audio waveforms into temporal FLAME motion parameters. Key tensor dimensions (e.g., batch size B, sequence length T) and layer configurations are annotated.

3. **Refinement and Fusion Pathway:** The core of our extractor is a top-down refinement pathway consisting of a series of **Feature Fusion Blocks**. This pathway iteratively fuses the features, starting from the coarsest level and progressively integrating finer-level details. Each Feature Fusion Block takes two feature maps as input (one from a coarser level, one from the current level), processes them through residual convolutional units, and then upsamples the result to match the resolution of the next finer level.
4. **Final Output:** After passing through the entire fusion pathway, a final 3×3 convolution produces the dense spatial feature map $\mathbf{F}_a \in \mathbb{R}^{256\times256\times256}$. The global identity feature $\mathbf{f}_{id} \in \mathbb{R}^{768}$ is taken directly from the `[CLS]` token of the final layer of the DINOv2 model.

This multi-scale fusion strategy provides the subsequent generator modules with a highly informative and spatially coherent feature representation, which is crucial for reconstructing fine details.

B.2 Avatar Generator

The Avatar Generator reconstructs a complete, animatable 3D Gaussian avatar from a single source image, conditioned on the features provided by the Multi-scale Feature Extractor. It follows the **Core-Shell** design philosophy outlined in the main paper and visualized in the bottom half of Figure 5(a).

Dynamic Core Generator. The Core generator produces the attributes of the dynamic facial Gaussians, G_{core}, which are topologically anchored to the FLAME mesh. Its design is MLP-based, optimized for processing sparse, per-vertex information.

- **Input Features:** The input for each of the N_v Core Gaussians is a fused feature vector. As depicted in Figure 5(a), this vector is created by first projecting the global identity feature $\mathbf{f}_{id}$ to a 256-dimensional space and then adding it to a corresponding row from a

learnable, vertex-specific embedding matrix $\mathbf{F}_p \in \mathbb{R}^{N_v \times 256}$. The result is concatenated with a 27-dimensional positional encoding of the view direction.

- **Network Structure:** A set of small, parallel MLPs is used to predict the Gaussian attributes from the fused feature vector. Each attribute (color, opacity, scale, rotation) is predicted by a separate 2-layer MLP with a hidden dimension of 128 and ReLU activation. The predicted positional offsets are scaled by a small factor (0.01) to act as minor adjustments.
- **Output:** The generator outputs the color $\mathbf{c}$, opacity α, scale $\mathbf{s}$, rotation $\mathbf{q}$, and a small positional offset $\Delta\mathbf{p}$ for each Core Gaussian. The final 3D positions are the sum of the canonical FLAME vertex locations and these predicted offsets.
- **Internal Oral Structures:** To model the inner mouth, we generate an additional 564 Gaussians representing the teeth (464) and an internal cavity filler (100). These are not part of the standard FLAME topology and are associated with a separate learnable embedding matrix $\mathbf{F}_{oral} \in \mathbb{R}^{564 \times 256}$. This allows them to have a distinct appearance (e.g., white teeth, dark cavity) from the facial skin. They are processed by the same MLP-based generator but with a larger positional offset scale (0.1) to allow for greater geometric adjustments. Their canonical positions are determined procedurally as detailed in Section C.

Static Shell Generator. The Shell generator produces the attributes of the Gaussians G_{shell}, capturing high-frequency details like hair and accessories. It employs a convolutional architecture to efficiently process the dense spatial feature map $\mathbf{F}_a$.

- **Input Features:** The network takes as input the multi-scale spatial feature map $\mathbf{F}_a \in \mathbb{R}^{256 \times 256 \times 256}$ concatenated with a 27-dimensional positional encoding of the view direction, which is broadcast across the spatial dimensions.
- **Network Structure:** As shown in the *Static Shell* branch of Figure 5(a), the architecture consists of two main parts:
  1. A shared feature backbone composed of two sequential 2D convolutional blocks (Conv-ReLU). The first block reduces the channel dimension from $256 + 27$ to 256, and the second further reduces it to 128. Both use a 3×3 kernel with padding.
  2. **Six** separate prediction heads that operate on the resulting 128-channel feature map. The heads for color, opacity, scale, rotation, and position are single 1×1 convolutional layers, predicting attributes such as the crucial 3D position offset $\Delta\mathbf{p}$. **A dedicated sixth head, composed of two additional convolutional layers, predicts the static, per-Gaussian modulation feature map $\mathbf{F}_m \in \mathbb{R}^{16 \times 256 \times 256}$.**
- **Output:** The generator outputs all Gaussian attributes $(\Delta\mathbf{p}, \mathbf{c}, \alpha, \mathbf{s}, \mathbf{q})$ for each of the 256×256 points on the initial view-aligned plane.

Neural Residuals for Fine-grained Dynamics. To model non-linear effects, we introduce a lightweight, expression-conditioned MLP, which predicts residual modulations.

- **Input Features:** For each Shell Gaussian, the MLP takes a concatenated feature vector as input, comprising: its static, per-Gaussian **modulation feature** (16-dim) from the Shell generator; its **canonical 3D position** (3-dim); and the current frame's dynamic FLAME parameters, specifically the **expression** (100-dim), **jaw/head pose** (6-dim), and **eye pose** (6-dim). This results in a 131-dimensional input vector.
- **Architecture:** The network is a 2-layer MLP with a hidden dimension of 64 and a ReLU activation. The final layer is zero-initialized to ensure training stability.
- **Output Modulations:** The 7-dimensional output is interpreted as a residual position offset $\Delta\mathbf{p}$, a residual color delta $\Delta\mathbf{c}$, and a scalar opacity gate $\Delta\alpha_{\text{gate}}$.
- **Hybrid Animation Formula:** These predicted residuals are applied to the attributes derived from the kinematic step. For instance, the final position is the sum of the base position and the neural offset, $\mathbf{p}' = \mathbf{p}_{\text{base}} + \Delta\mathbf{p}$. The final opacity is the product of the canonical opacity and the sigmoid-activated opacity gate, allowing the network to learn to make occluded parts transparent.

B.3 AUDIO-TO-MOTION MODULE

This module translates a raw audio waveform into a temporally coherent sequence of FLAME motion parameters. It follows a standard encoder-decoder pipeline, specifically designed for efficiency and temporal consistency, as illustrated in Figure 5(b).

Audio Encoder. We employ a pre-trained **Wav2Vec2.0-Base** Baevski et al. (2020) model as a powerful and fixed audio encoder. It processes the raw audio, sampled at 16 kHz, and extracts a sequence of rich contextual features. These features are then linearly interpolated to match the target animation frame rate of 25 FPS, ensuring a frame-perfect alignment between the audio signal and the generated motion.

Temporal Decoder. The core of the motion synthesis is a **Temporal Convolutional Network (TCN)**, designed to capture long-range dependencies in the audio features.

- **Architecture:** The TCN consists of a stack of three 1D residual convolutional blocks. Each block uses a kernel size of 3, a stride of 1, and Layer Normalization for stabilization. A linear projection layer first maps the 768-dimensional Wav2Vec2 features to the TCN's internal dimension of 512.
- **Temporal Conditioning:** To ensure smooth transitions between consecutively generated animation clips, we introduce a temporal conditioning mechanism (see 'Condition Features' in Figure 5(b)). The motion parameters of the last frame from the preceding clip are encoded via a linear layer and then **added** to the audio feature sequence at every time step before it enters the TCN. This additive modulation effectively biases the network's prediction based on the recent motion history, preventing temporal discontinuities.

Output Head. The output of the TCN is processed by a final projection head. This head is a 2-layer MLP with a Tanh activation function, which maps the 512-dimensional temporal features to the final 69-dimensional FLAME motion parameter space. This streamlined architecture efficiently models long-range temporal dependencies while maintaining a small computational footprint, which is critical for our high-performance synthesis pipeline.

C FLAME REFINE

The standard FLAME model Li et al. (2017), while providing a high-quality statistical representation of the human head, anatomically omits internal oral structures. This absence significantly limits its expressiveness for realistic speech animation. To address this, we introduce a series of refinements to build a complete and efficiently animatable head model. Our pipeline consists of three main components: procedural oral cavity generation, a hybrid animation mechanism combining kinematics with learned dynamic details, and inference-time optimizations.

C.1 PROCEDURAL ORAL CAVITY GENERATION

Instead of using a generic, static mesh, we procedurally generate person-specific oral structures for each identity, ensuring they are anatomically coherent with the individual's facial shape. This process, implemented in `build_teeth`, creates two components: the teeth and an internal oral cavity filler.

Teeth Generation. The process is guided by anatomical heuristics based on the canonical FLAME mesh vertices:

1. **Scaffold Definition:** We first identify landmark contours on the canonical FLAME mesh corresponding to the outer upper and lower lips.
2. **Person-Specific Scaling:** The average distance between these lip contours is computed to serve as an intrinsic, per-identity scale factor. This factor dictates the dimensions (e.g., height, thickness) of the resulting dental arches.
3. **Densification and Volumetric Construction:** A centerline between the lips is sampled at 15 initial points. These are then densified to 29 points via linear interpolation. To create

volumetric teeth instead of a simple surface, we construct a multi-layered structure (8 layers for the upper arch, 8 for the lower) by extruding these points along their Y and Z axes. This results in 464 vertices representing the upper and lower teeth with realistic volume.

Oral Cavity Filler. To prevent a hollow-mouth appearance and provide a dark, non-reflective background during speech, we add an internal filler. This is constructed as a 10x10 grid of 100 points, positioned behind the generated teeth and scaled to fit the oral cavity.

The final output is a set of 564 canonical points ($\mathcal{P}_{oral} \in \mathbb{R}^{564\times3}$) representing the complete oral cavity, which are then used to generate dedicated Gaussians.

C.2 Kinematic Animation and Dynamic Refinements

Our animation pipeline is a hybrid model that combines traditional Linear Blend Skinning (LBS) with a learned network for dynamic, non-linear details.

Base Kinematic Animation. The base motion is driven by the FLAME skeleton.

- **FLAME Head:** The main head vertices are animated using standard LBS with the full FLAME skinning weights $\mathcal{W}$.
- **Oral Structures:** The 564 oral cavity points are animated using sparse LBS weights. Specifically, the upper teeth vertices are rigidly bound to the head/neck joint (index 1), while the lower teeth and internal cavity filler vertices are rigidly bound to the jaw joint (index 2). This ensures correct articulation during speech.

Neural Residuals for Fine-grained Dynamics. While kinematic binding provides robust, large-scale motion, it cannot capture subtle, non-linear effects like skin wrinkling or muscle compression. To model these, we introduce a lightweight, expression-conditioned MLP, referred to as `shell_exp_mlp`, which predicts fine-grained residual modulations on top of the kinematically driven result.

- **Input Features:** The MLP is designed to be highly efficient. For each Shell Gaussian, it takes a concatenated feature vector as input, comprising: its static, per-Gaussian **modulation feature** (16-dim) from the Shell generator; its **canonical 3D position** (3-dim); and the current frame's dynamic FLAME parameters, specifically the **expression** (100-dim), **jaw/head pose** (6-dim), and **eye pose** (6-dim). This results in a 131-dimensional input vector.
- **Architecture:** The network is a simple 2-layer MLP with a hidden dimension of 64 and a ReLU activation function. It projects the 131-dim input to a 7-dim output vector. To ensure training stability, the final layer is initialized with zero weights and biases, preventing the MLP from disrupting the initial geometric binding.
- **Output Modulations:** The 7-dimensional output is interpreted as three distinct modulations:
  - A **residual position offset** $\Delta\mathbf{p}$ (3-dim).
  - A **residual color delta** $\Delta\mathbf{c}$ (3-dim).
  - A scalar **opacity gate** $\Delta\alpha_{\text{gate}}$ (1-dim).
- **Hybrid Animation Formula:** These predicted residuals are applied to the attributes derived from the base kinematic step. The final animated position $\mathbf{p}'$ is the sum of the kinematically derived base position $\mathbf{p}_{\text{base}}$ and the neural offset $\Delta\mathbf{p}$. The final color $\mathbf{c}'$ is the sum of the canonical color and $\Delta\mathbf{c}$. Crucially, the final opacity α' is the product of the canonical opacity and the sigmoid-activated opacity gate, $\alpha' = \alpha \cdot \sigma(\Delta\alpha_{\text{gate}})$, allowing the network to learn to make certain shell parts (e.g., those occluded inside an open mouth) transparent.

This hybrid architecture perfectly balances stability and expressiveness, adding high-frequency dynamic details with minimal computational overhead.

C.3 Inference-Time Optimizations

To achieve real-time performance, our framework incorporates several optimizations aimed at minimizing redundant computations during the animation loop.

Decoupling Static and Dynamic LBS Components. A standard Linear Blend Skinning (LBS) formulation re-evaluates the full set of shape and expression blendshapes for every frame, which is computationally inefficient. Our approach mitigates this by decoupling the static, identity-dependent components from the dynamic, expression-driven components. For each new identity, we perform a one-time pre-computation to determine the static rest-pose of the skeleton joints ($\mathbf{J}_{static}$), which depends solely on the shape parameters ($\boldsymbol{\beta}$). During the animation phase, per-frame computations are then confined to the minor joint displacements induced by the expression parameters ($\boldsymbol{\psi}$). These dynamic offsets are applied to the cached $\mathbf{J}_{static}$ to derive the final joint transformations, significantly reducing the per-frame computational load.

Caching of Binding Coordinates. Similarly, the local coordinates—barycentric weights and normal height—that define a Shell Gaussian's position relative to its Core anchor triangle are invariant in the canonical space. We leverage this property by caching these local coordinates for every Shell Gaussian in a one-time initialization step. During animation, the final position of each Shell point is then efficiently reconstructed by applying these cached local coordinates to the vertices of the corresponding posed Core triangle. This strategy obviates the need for repeatedly solving a system of linear equations for barycentric coordinates in every frame.

These optimization strategies are fundamental to eliminating computational bottlenecks and enabling the high frame rates reported in our experiments.

D Dataset Construction and Pre-processing

Our framework is trained on a large-scale, diverse collection of video data, carefully curated and processed to optimize for both high-fidelity visual reconstruction and robust animation control. The data preparation involves two distinct sets for training our two main modules: the Avatar Generator and the Audio-to-Motion module. A unified pre-processing pipeline is applied to all data to ensure consistency.

D.1 Training Data for the Avatar Generator

The primary goal for this module is to learn a mapping from a single image to a high-fidelity, identity-preserving 3D avatar. To achieve this, we combine a high-quality, in-the-wild dataset with a multi-view studio dataset.

VFHQ Dataset. Our primary training data is derived from the high-quality dataset Xie et al. (2022), which offers over 15,000 high-resolution video clips from diverse interview scenarios. From the official VFHQ training split, we designated a subset of 100 video clips for our validation set, with the remainder used for training. The official VFHQ test set was held out exclusively for final evaluation.

MEAD Dataset. To improve the model's robustness to varied head poses and extreme expressions not prevalent in the largely frontal-facing VFHQ dataset, we augmented our training data with the Multi-view Emotional Audio-visual Dataset MEAD Wang et al. (2020). We utilized the publicly available portion featuring 29 actors. From the multi-view recordings of each actor, we randomly sampled 600 frames to construct a diverse, multi-view image collection that enhances the generator's ability to model 3D geometry from a single view.

D.2 Training Data for the Audio-to-Motion Module

The Audio-to-Motion module is trained to predict realistic facial motion parameters from speech. For this task, we combine our curated VFHQ dataset with an additional in-the-wild dataset to improve generalization.

Curated VFHQ. While VFHQ offers over 15,000 high-resolution video clips from diverse interview scenarios, it does not include synchronized audio. To construct a suitable audio-visual corpus, we executed the following curation steps:

1. We first retrieved the original audio tracks corresponding to the video clips using the provided metadata.

2. To ensure high-quality lip synchronization, a critical aspect for our audio-driven task, we employed a pre-trained SyncNet Chung & Zisserman (2016) model. We rigorously filtered the dataset, retaining only video clips with a confidence score greater than 3.0.
3. After removing clips that were no longer available online or failed the synchronization test, we obtained a high-quality, curated audio-visual subset consisting of **7,031 video clips**.

The 7,031 audio-visual clips from our curated VFHQ set serve as the primary training data, providing a strong baseline for learning the mapping from speech to facial dynamics.

TFHP Dataset. To enhance the generalization of our audio-driven model across a wider array of speaking styles and natural head movements, we additionally incorporate the Talking Face with Head Poses (TFHP) Sun et al. (2024) dataset. TFHP, with its 10.5 hours of content from lectures, online courses, and news programs, exposes our model to a rich diversity of prosody and motion patterns.

D.3 Unified Pre-processing Pipeline

All selected datasets undergo a consistent, multi-stage pre-processing pipeline to generate the final training data.

1. **Standardization:** All video clips were standardized to a resolution of 512×512 and a frame rate of 25 FPS. Corresponding audio tracks were resampled to 16 kHz to match the input requirements of our audio encoder.
2. **Background Segmentation:** To isolate the subject and remove distracting backgrounds, we applied a state-of-the-art portrait segmentation model, BirenNet Yin et al. (2022), to every frame.
3. **3DMM Parameter Annotation:** Obtaining high-fidelity pseudo-ground-truth 3DMM parameters is crucial. Instead of relying on a single estimator, we adopted a **best-of-breed strategy**, leveraging the distinct strengths of multiple state-of-the-art tools to annotate each aspect of the FLAME model:
   - **Identity Shape (β):** We used MICA zie (2022) to extract the 300-dimensional shape parameters. MICA is specifically optimized for high-fidelity, metrical shape reconstruction from a single image.
   - **Expression (ψ):** The 100-dimensional expression parameters were extracted using EMOCA Danecek et al. (2022), which excels at capturing a wide range of dynamic facial expressions.
   - **Pose (θ) and Camera:** We employed DECA Feng et al. (2021) to estimate the 6D head pose and camera parameters, as it provides robust and stable pose tracking.

   This composite annotation pipeline ensures that each component of the 3DMM is estimated with the highest possible accuracy, providing high-quality supervisory signals for our training process.

E Evaluation Protocol Details

To ensure a fair, transparent, and reproducible comparison against prior works, this section details the protocol used for our quantitative evaluations, including test data construction and the precise definition of each evaluation metric.

E.1 Evaluation Protocol for Video-Driven Synthesis

E.1.1 Test Data Construction For Video-Driven Task

For the video-driven reenactment task, we strictly follow the evaluation protocol established by GAGAvatar Chu & Harada (2024) to ensure a fair comparison. Our test set is derived from the official VFHQ Xie et al. (2022) test split, which comprises 100 high-quality video clips of unseen identities.

For each test video, the evaluation is performed as follows:

- **Source Image:** The very first frame of the video is designated as the source image, $\mathcal{I}_{\text{src}}$.
- **Driving and Ground-Truth Images:** The subsequent frames in the video are used as the driving inputs to our model. These same frames also serve as the ground-truth target images, $\mathcal{I}_{\text{gt}}$, for metric computation.

This setup evaluates the model's ability to reconstruct an avatar from a single view and then animate it to match a sequence of target expressions and poses from the same identity (self-reenactment).

E.1.2 VIDEO-DRIVEN METRIC DEFINITIONS

We employ a comprehensive suite of metrics to evaluate different aspects of our generated avatars, including image reconstruction quality, identity preservation, and animation accuracy.

Image Reconstruction Quality. These metrics are computed directly between the generated image $\mathcal{I}_{\text{gen}}$ and the ground-truth image $\mathcal{I}_{\text{gt}}$.

- **PSNR (Peak Signal-to-Noise Ratio):** A standard metric for image fidelity, measuring the ratio between the maximum possible power of a signal and the power of corrupting noise. Higher is better.
- **SSIM (Structural Similarity Index Measure):** Measures the perceptual similarity between two images, considering luminance, contrast, and structure. It is computed on RGB images with a window size dynamically adapted to the image dimensions. Higher is better.
- **LPIPS (Learned Perceptual Image Patch Similarity) Zhang et al. (2018):** A perceptual metric that computes the distance between deep features extracted from two images. We use the pre-trained AlexNet backbone and normalize input images to the range $[-1, 1]$ before computation. Lower is better.

Identity Preservation. This metric quantifies how well the identity of the source subject is preserved in the generated result.

- **CSIM (Cosine Similarity):** We use a pre-trained ArcFace Deng et al. (2019a) model via the `insightface` library to extract 512-dimensional normalized face recognition embeddings, $\mathbf{e}_{\text{src}}$ and $\mathbf{e}_{\text{gen}}$, from the source image $\mathcal{I}_{\text{src}}$ and the generated image $\mathcal{I}_{\text{gen}}$, respectively. The CSIM is their cosine similarity:
$$\text{CSIM} = \frac{\mathbf{e}_{\text{src}} \cdot \mathbf{e}_{\text{gen}}}{\|\mathbf{e}_{\text{src}}\| \|\mathbf{e}_{\text{gen}}\|} \tag{8}$$
Higher is better, indicating stronger identity preservation.

Animation Accuracy. These metrics measure how accurately the generated avatar mimics the facial expression, head pose, and keypoint positions of the ground-truth video. For metrics based on 3DMM parameters (AED, APD), we use a pre-trained **Deep3DFaceRecon** Deng et al. (2019b) as a consistent, third-party estimator. For 2D keypoints (AKD), we use the `face_alignment` Bulat & Tzimiropoulos (2017) library.

- **AED (Average Expression Distance):** Measures the error in facial expression. We extract the 64-dimensional expression coefficient vectors, $\boldsymbol{\psi}_{\text{gt}}$ and $\boldsymbol{\psi}_{\text{gen}}$, from the ground-truth and generated images. AED is defined as the scaled L1 norm (Mean Absolute Error) of their difference:
$$\text{AED} = \mathbb{E}_i \left[|\psi_{i,\text{gen}} - \psi_{i,\text{gt}}| \right], \tag{9}$$
Lower is better.
- **APD (Average Pose Distance):** Measures the error in head pose. Following the protocol of OTAvatar Ma et al. (2023), we extract the 3-dimensional angle vector $\boldsymbol{\alpha}$ and the 3-dimensional translation vector $\mathbf{t}$ for both the ground-truth and generated images. These are concatenated to form 6-dimensional pose vectors, $\mathbf{P}_{\text{gt}} = [\boldsymbol{\alpha}_{\text{gt}}, \mathbf{t}_{\text{gt}}]$ and $\mathbf{P}_{\text{gen}} = [\boldsymbol{\alpha}_{\text{gen}}, \mathbf{t}_{\text{gen}}]$. APD is then calculated as the Mean Absolute Error (L1 similarity) between these two vectors:
$$\text{APD} = \mathbb{E}_i \left[|P_{i,\text{gen}} - P_{i,\text{gt}}| \right], \tag{10}$$
Lower is better.

- **AKD (Average Keypoint Distance):** Measures the 2D landmark alignment error. We detect 68 facial keypoints, $\{\mathbf{k}_{i,\text{gt}}\}$ and $\{\mathbf{k}_{i,\text{gen}}\}$, from the ground-truth and generated images. AKD is the average Euclidean distance between corresponding keypoints, measured in pixels without normalization:
$$\text{AKD} = \mathbb{E}_i \left[\|\mathbf{k}_{i,\text{gen}} - \mathbf{k}_{i,\text{gt}}\|_2 \right], \tag{11}$$
Lower is better.

E.2 Evaluation Protocol for Audio-Driven Synthesis

To evaluate the performance of our audio-driven talking head generation, we designed a comprehensive protocol to measure identity preservation, expression accuracy, perceptual quality, and audio-lip synchronization. This section details the metrics, the test data construction, and the specific computation procedures.

E.2.1 Test Data Construction for Audio-Driven Task

Our test set is constructed to evaluate cross-identity audio-driven synthesis in a controlled manner. It is derived from the 100-video validation set that we held out from the VFHQ dataset.

1. **Identity and Audio Source Selection:** We randomly select **10 distinct videos** from our validation set. For each selected video, the first frame is used as a source identity image, and its corresponding audio track is used as a driving audio source. This gives us 10 unique identities and 10 unique audio clips.
2. **Cross-Synthesis Task Generation:** We perform a full cross-synthesis. Each of the 10 source identities is driven by each of the 10 audio clips, resulting in a total of $10 \times 10 = \mathbf{100}$ generated test videos. For consistency, only the first 5 seconds of each audio clip are used for generation.
3. **Ground-Truth Definition:** For each generated video, there are two corresponding ground-truth sources:
   - The **source identity image** (for CSIM).
   - The **ground-truth video** from which the driving audio was extracted (for AED).
   - The **identity video** (the video from which the source identity image was taken) is used as the real distribution reference for FID.

E.2.2 Audio-Driven Metric Definitions

We employ the following metrics to provide a multi-faceted assessment of our audio-driven results:

- **FID (Fréchet Inception Distance).** A metric for evaluating the quality and diversity of images from generative models Heusel et al. (2017). It measures the Fréchet distance between the distributions of high-level image features of real images versus generated images. These features are extracted from a pre-trained Inception-v3 network and are modeled as multivariate Gaussian distributions. Given the mean and covariance of the feature distributions for real images (m_r, C_r) and generated images (m_g, C_g), the FID is:
$$d^2 = \|m_r - m_g\|_2^2 + \text{Tr}\left(C_r + C_g - 2(C_r C_g)^{1/2}\right) \tag{12}$$
where $\text{Tr}(\cdot)$ is the trace of a matrix. A **lower** FID score indicates that the distribution of generated images is more similar to the distribution of real images, suggesting better overall quality and variety.
- **Sync (Lip-Sync Confidence Score).** A metric that quantifies the quality of lip-synchronization by measuring the correspondence between mouth movements in a video and an audio track, as defined by the SyncNet Chung & Zisserman (2016). First, audio and visual (mouth) streams are embedded into a common feature space. The Euclidean distance $d(\tau) = \|v - a_\tau\|_2$ is computed between the visual embedding v and the audio embedding a_τ over a window of time offsets τ. The confidence score is then calculated as the difference between the median and minimum of these distances:
$$\text{Sync} = \text{median}(\{d(\tau)\}) - \min(\{d(\tau)\}) \tag{13}$$

A high score indicates a sharp, distinct minimum in the distance plot, signifying a strong and unambiguous temporal alignment between audio and video. A **higher** Sync_c score indicates better lip-sync quality.

E.2.3 Metric Implementation Details

The following procedures detail how each metric is computed on our generated test set.

CSIM. For each of the 100 generated videos, we sample frames at 10 FPS from the first 5 seconds. We then compute the cosine similarity between the ArcFace Deng et al. (2019a) embedding of the single source identity image and the embedding of each sampled frame. The final CSIM score for that video is the average of these individual similarity scores.

FID. The FID score is calculated between a set of generated frames and a set of real frames from the source identity.

- **Generated Set:** For each source identity, we have 10 generated videos. We sample 50 frames (10 FPS for 5s) from each of these 10 videos, resulting in a set of $10 \times 50 = 500$ generated frames for that identity.
- **Real Set:** We take the original video corresponding to the source identity and sample frames from it at 10 FPS. To ensure the number of real frames matches the number of generated frames (500), we **repeat the sequence of real frames** until 500 frames are collected. This ensures a fair comparison of distributions of equal size.
- We then compute the FID between the InceptionV3 features of these two sets of 500 frames. The final reported FID is the average over all 10 identities.

AED. Our AED calculation is designed to measure the overall expression style of the video rather than frame-by-frame error.

- For each generated video and its corresponding ground-truth video (the one that provided the audio), we sample frames at 10 FPS for the first 5 seconds.
- Using a pre-trained Deep3DFaceRecon Deng et al. (2019b), we extract the 64-dimensional expression coefficient vector $\boldsymbol{\psi}$ for every sampled frame in both videos.
- We then compute the **temporal average** of the expression vectors for each video, resulting in a single mean expression vector, $\bar{\boldsymbol{\psi}}_{\text{gen}}$ and $\bar{\boldsymbol{\psi}}_{\text{gt}}$

This metric evaluates whether the generated video captures the overall sentiment and expression style of the ground-truth performance over a time window.

Sync Score. The SyncNet score is computed for each generated 5-second video clip with its driving audio. The final score is the average over all 100 generated videos.

F More Visual Results

To provide a more comprehensive qualitative assessment of our framework, we present additional visual results for the audio-driven synthesis task in Figures 6 and 7.

These figures showcase the robustness of our method in generating high-fidelity, identity-preserving talking head videos across a variety of subjects and spoken content. The results highlight the naturalness of the synthesized expressions and the accurate lip synchronization achieved by our Audio-to-Motion module.

Furthermore, to offer deeper insight into the intermediate motion representation learned by our model, we visualize the FLAME parameters synthesized directly from the audio input. As depicted alongside the rendered video frames, these parameters demonstrate a smooth and temporally coherent sequence of facial movements, capturing the subtle dynamics of speech. This visualization confirms that our audio-to-motion mapping is not only effective but also produces plausible and well-structured 3D facial animations, which form the basis for our high-quality final renderings.



Figure 6: More audio-driven results. The "A2M" column shows the FLAME parameters generated from audio, rendered as meshes.

More visual results can be found at `https://anonymous.4open.science/r/FLARE-30F4`.

G Limitations and Social Impact

While FLARE achieves a strong balance between speed and quality, our design choices introduce several inherent limitations that also point toward exciting avenues for future research.



Figure 7: More audio-driven results. The "A2M" column shows the FLAME parameters generated from audio, rendered as meshes.

One-Shot Artifacts. Like many one-shot methods, FLARE can be sensitive to the source image. Strong, non-neutral expressions or harsh, one-sided lighting in the source image can sometimes "bake" into the canonical avatar, subtly affecting animation quality. Developing more robust feature extraction or canonicalization techniques could mitigate these effects.

Social Impact. The technology for creating realistic digital avatars, including ours, carries a risk of misuse for creating deceptive media ("deepfakes"). We will release our code for academic and research purposes to spur innovation in real-time graphics and human-computer interaction. We

advocate for the development of robust detection and watermarking techniques to run in tandem with generative models to prevent malicious use.