

AURA: AUdio-dRiven streaming Avatar

Nathaniel Cohen^{1,2}
nathaniel.cohen@kyutai.org

Nicolas Dufour¹
nicolas.dufour@kyutai.org

Amélie Royer¹
amelie@kyutai.org

Alasdair Newson²
anewson@isir.upmc.fr

Patrick Pérez¹
patrick@kyutai.org

¹ Kyutai
Paris, France

² ISIR, Sorbonne Université
Paris, France

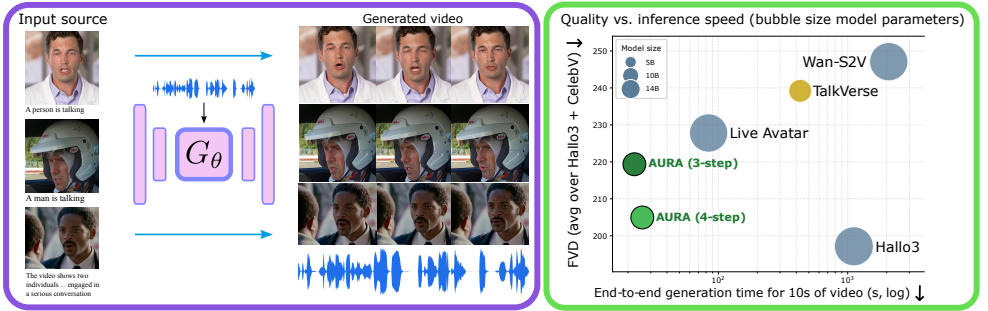


Figure 1: **AURA generates 720p talking-head videos at 9.7 fps on a single GPU in a causal streaming manner**, per chunk of 0.64s. (Left) Excerpts of three samples generated from the same audio with different reference identities and prompts, with frames temporally aligned across rows. (Right) Quality–inference time trade-off for AURA and baselines, averaged across the CelebV-HQ and Hallo3 test sets; Specifically, we report FVD vs. end-to-end generation time for 10s of video (log scale, lower is better for both); The circle areas are proportional to parameter count in the models. Video results on the [project page](#).

Abstract

Current audio-driven video diffusion models can synthesize realistic talking-head videos but generally require dozens of diffusion denoising steps over multi-second video chunks, making them unsuitable for streaming applications such as voice assistants and virtual avatars, where low-latency and online causal generation are required properties. AURA generates high-resolution video chunk-by-chunk at 9.7 fps with a rolling KV cache, requiring only the audio of the current chunk rather than the multi-second audio look-ahead buffered by bidirectional models, making it well-suited to interactive

avatar applications. Our two-stage distillation pipeline first aligns the causal student with the teacher, by training it to match the teacher’s trajectories with fewer diffusion steps and a chunk-wise causal attention mask, then applies Distribution Matching Distillation (DMD) with backward simulation to close the remaining quality gap. On standard talking-head benchmarks, AURA matches the teacher’s quality while running $17\times$ faster, achieving 720p avatar streaming on a single GPU. To support reproducibility and further research on streaming avatar generation, we release our model’s weights along with a dataset of 30K synthetic videos and their precomputed teacher trajectories.

1 Introduction

Recent speech-to-speech models [12, 36] are able to model full-duplex voice interaction in real-time, paving the way for voice assistants that hold natural and smooth conversations. However, these models only generate audio signal. In contrast, human spoken communication relies on visual cues and facial expressions, via an interplay of the audio and visual modalities. For instance, lip motion alters perceived phonemes [42], visual cues sharpen comprehension in noise [42], and an embodied talking face enhances the sense of social presence over voice alone [6]. To model such behaviors, audio-driven video generation models can synthesize talking-head videos from a reference image and an input speech audio, enabling applications such as virtual avatars, synthetic presenters, and non-player characters in games. All these applications, however, share a common constraint: the avatar must be generated on-the-fly, as the audio arrives, from a stream of audio of unbounded duration.

State-of-the-art talking-head models such as TalkVerse [44] and Wan-S2V [15] produce high-quality audio-driven videos but are ill-suited to the streaming setting. Because they rely on bidirectional attention, they must generate entire video chunks of several seconds at once, running dozens of denoising steps over the full chunk before producing any frame, which on a single GPU takes minutes (per chunk). Recently, CausVid [52] and Self-Forcing [21] have explored distilling such bidirectional video diffusion teachers into few-step causal students that generate video chunk by chunk in an autoregressive manner. However, these methods target only the text-to-video and image-to-video settings. Extending this approach to the audio-driven setting, where the model must additionally track a dense time-aligned speech signal, has so far been demonstrated only on much larger models (14–18B parameters) [22, 30, 57], which remain too large for real-time inference on a single GPU (only 2 fps). A second obstacle is the scarcity of suitable training data: existing public talking-head datasets such as Hallo3 [9] and CelebV [56] consist of clips too short to cover the chunk length of modern teachers like TalkVerse, ruling them out for distillation (see Section 3.6).

We introduce AURA, a 5B-parameter causal few-step speech-to-video (S2V) model that generates streaming talking-head videos from live audio at 720p resolution and 9.7 fps on a single GPU, a $17\times$ speedup over its teacher. For training, we generate and release a new talking-head dataset comprising 30K videos (38.7 hours) produced with the TalkVerse teacher, conditioned on reference images and text descriptions from the Hallo3 dataset [9] and synthetic speech audio. To train our model, we use a two-stage distillation pipeline, similar to CausVid: Stage 1 first trains the causal student to match the bidirectional teacher (TalkVerse [44])’s Ordinary Differential Equation (ODE) trajectories in order to reduce the gap between the bidirectional teacher and the causal student architecture. Then, Stage 2 closes the few-step gap with Distribution Matching Distillation (DMD) and *backward simulation*: by training the student on intermediate states from its own denoising rollouts, the

Stage 2 loss reflects accumulating errors the student may actually produce at inference.

To further improve efficiency, we train the model to support chunk-wise causal attention, allowing us to use a rolling KV cache with bounded memory at inference. We also use a single KV cache for all diffusion denoising timesteps, further reducing memory footprint.

With this approach, on standard talking-avatar benchmarks, AURA matches its bidirectional teacher on visual quality (FVD [44] and JEDi [31] metrics) and audio-visual alignment, while running $17\times$ faster with only four denoising steps per chunk. AURA delivers 720p streaming video at 9.7 fps on a single GPU, faster than all baselines we compare against in the same single-GPU setting. Figure 1 shows an excerpt of qualitative samples generated by AURA, as well as its speed-quality tradeoff against prior work.

In summary, we introduce AURA, which makes the following contributions:

- A 5B causal few-step S2V model that matches its bidirectional teacher on FVD and JEDi metrics, with a $17\times$ speedup. Our inference pipeline streams 720p video at 9.7 fps on a single GPU, the highest throughput among audio-driven baselines we compare against. Our method is based on a two-stage distillation method (ODE-trajectory initialization followed by Distribution Matching Distillation with backward simulation) paired with a streaming inference scheme based on a rolling KV cache of past chunks;
- A public dataset of 30K teacher-generated talking-head ODE trajectories paired with synthetic speech, released to support future research on streaming S2V distillation;

2 Related Work

Diffusion and Flow Matching. Image Generative models have evolved through several paradigms: Generative Adversarial Networks (GANs) [1, 16, 23] are efficient as they generate images in a single forward pass but suffer from known limitations around mode coverage and training stability. More recently, diffusion models [12, 18, 24, 35, 39, 40] and flow matching [27, 29] have become the dominant paradigm for image generation. Indeed, both frameworks are closely related and model data through an iterative multi-step refinement process, yielding stable training with high sample diversity but at the cost of slow sampling. Most modern large-scale video generators, including our teacher TalkVerse [46], are trained under the flow matching framework, and our work also builds on this formulation.

Video Diffusion Models. Recent work has further scaled diffusion for video generation. Early efforts such as Imagen Video [19], Make-A-Video [68], Align Your Latents [8], and Stable Video Diffusion [2] extended image diffusion to short clips, typically by inflating 2D architectures with temporal layers. More recent open-source and industrial models, Sora [6], Movie Gen [33], Open-Sora [65], CogVideoX [49], HunyuanVideo [25], and Wan [45]—adopt diffusion transformer (DiT) architectures and operate in a compressed latent space, scaling to 720p–1080p resolutions and minute-long clips. These models are typically non-causal: they generate the entire video sequence jointly over 50 or more denoising steps, and are thus not well adapted to the latency requirements of streaming applications. Current SotA talking-head models are based on such architectures. For instance, Hallo3 [9] adapts CogVideoX [49] for portrait animation with a DiT-adapted identity reference network, while closed-source systems such as OmniHuman [26] extend DiT-based audio-driven generation to full-body animation. In this work, we use TalkVerse [46], which builds on the Wan2.2-5B architecture, as a non-causal multi-step distillation teacher and distill it into a causal, few-step student suitable for streaming generation.

Distillation of Diffusion Models. The slow inference of diffusion models has motivated extensive work on *distillation*, training a faster few-step student to reproduce a slow teacher’s output distribution. One particularly successful line is Distribution Matching Distillation (DMD) [50], which trains a one-step student to minimize the reverse KL to the teacher via a learned critic; DMD2 [50] extends it to a few-step generator and introduces *backward simulation*, computing the loss on the student’s own rollouts rather than forward-diffused data. CausVid [52] applies DMD to distill a non-causal video teacher into a causal few-step student for streaming text-to-video. A related line tackles the train/test gap of causal models: e.g. Self-Forcing [41] trains on the student’s autoregressive rollouts, maintaining a training-time KV cache with stop-gradient, and its variants—Self-Forcing++ [40], Causal Forcing [57], Rolling Forcing [28], Long-Live [48]—explore scheduling, masking, and rolling strategies for longer, higher-quality sequences. We build on CausVid and DMD2 for speech-to-video generation; unlike Self-Forcing, we reproduce the streaming attention pattern with a single-pass block-causal mask (Sec. 3.5) instead of unrolling a training-time cache, our goal being a fast streaming S2V system rather than a new distillation paradigm.

Streaming Talking Heads. A handful of recent works tackle audio-driven talking-head generation in a streaming and few steps setting. Talking Machines [60] is, to our knowledge, the first to address this regime, distilling a 18B model with DMD and sparse causal attention; weights are not released. Live Avatar [22] distills the 14B Wan-S2V model [15] into a 14B few-step student that runs at 16 fps and 480p on a 5-GPU setup. SoulX-FlashTalk [37] adopts a similar few-step distillation strategy from a 14B teacher into a 14B student and requires 4 GPUs to reach real-time throughput. In contrast, AURA distills a 5B teacher into a same-size causal 4-step student that delivers 720p video at 9.7 fps on a single GPU. Closest to our work is Avatar Forcing [41] whose pre-print was concurrently published: It uses a similar DMD-based distillation process but requires a look-ahead window of future audio to generate each frame by design, leading to a small yet incompressible streaming delay.

3 Method

In this section, we present AURA, our two-stage pipeline that distills a non-causal audio-driven video diffusion teacher (TalkVerse) into a 4-step causal student, streaming 720p avatars in constant memory. We first recall Distribution Matching Distillation (Sec. 3.1) and the TalkVerse teacher (Sec. 3.1) to set out notation and context. Section 3.2 then introduces the causal student and the per-chunk mixed-timestep design that enables a compact KV cache. Section 3.3 and Section 3.4 cover the two training stages, and Section 3.5 finally describes streaming inference.

3.1 Preliminaries

Distribution Matching Distillation (DMD). DMD [50] trains a few-step student generator G_θ to match a multi-step teacher diffusion model at the distribution level, rather than along individual sampling trajectories. Concretely, DMD minimizes, in expectation over the diffusion timestep $t \sim \mathcal{U}(0, 1)$, the reverse KL divergence between the student distribution p_{gen} and the data distribution p_{data} , both diffused to noise level t . We adopt $t \in [0, 1]$ throughout this paper, with $t = 0$ corresponding to clean latents and $t = 1$ to pure noise. While the densities p_{gen} and p_{data} are intractable, the gradient of the KL with respect to θ admits a

closed-form expression as a difference of two score functions evaluated on student samples:

$$\begin{aligned}\nabla_{\theta} \mathcal{L}_{\text{DMD}} &\triangleq \mathbb{E}_t [\nabla_{\theta} \text{KL}(p_{\text{gen},t} \parallel p_{\text{data},t})] \\ &\approx -\mathbb{E}_t \int (s_{\text{data}}(\Psi(G_{\theta}(z), t), t) - s_{\text{gen},\eta}(\Psi(G_{\theta}(z), t), t)) \frac{dG_{\theta}(z)}{d\theta} dz, \quad (1)\end{aligned}$$

where z denotes the noise input to the student, $\Psi(\cdot, t)$ is the forward diffusion operator that produces a sample at noise level t , s_{data} is the frozen teacher score, and $s_{\text{gen},\eta}$ is an online score network (called *critic* in this paper) with parameters η that estimates $\nabla \log p_{\text{gen},t}$ on student samples. In practice, $s_{\text{gen},\eta}$ is trained jointly with G_{θ} via the standard denoising score-matching loss on student samples, and the two are updated in an alternating fashion.

Backward simulation. Directly applying DMD to multi-step students yields a training-inference mismatch: at training time, the student is fed forward-diffused real samples $x_t = \Psi(x_0, t)$, whereas at inference time, intermediate inputs come from previous denoising steps of the student itself. To close this gap, DMD2 [50] introduces *backward simulation*. At each training iteration, the student is unrolled for a few denoising steps to produce a synthetic intermediate sample $\hat{x}_t$, which then replaces the forward-diffused real sample as input to the DMD objective. Because G_{θ} is itself a few-step generator, this unrolling is cheap. The DMD gradient and the score-matching update for $s_{\text{gen},\eta}$ are then computed at $\hat{x}_t$ rather than at $\Psi(x_0, t)$, aligning the input distribution seen during training with the one produced at inference. We also adopt this scheme in the second state of our training pipeline (Sec. 3.4).

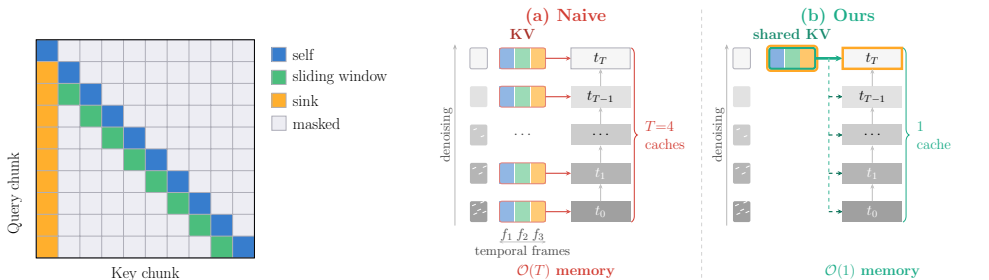
TalkVerse Architecture. We use TalkVerse [46] as our teacher model, a 5B-parameter audio-driven video diffusion model built on the Wan2.2-5B [45] DiT backbone; Its video VAE uses compression factors $(t, h, w) = (4, 16, 16)$ that enable efficient 720p generation. The model is conditioned on a reference identity image, a speech audio track encoded with Wav2Vec2 [10] and injected via cross-attention to temporally aligned visual tokens, and a text prompt encoded with T5-XXL [52]. Long videos are generated with a sliding-window mechanism that conditions each window on a set of *motion frames* compressed by a FramePack module [52], and the reference image is assigned a positional offset placed ahead of the denoised window. Inference uses 50 denoising steps with a flow-matching solver and classifier-free guidance. We provide a more detailed description in the supplementary material A.1.

3.2 AURA Architecture

To turn the 50-step bidirectional teacher (TalkVerse) into a 4-step causal student, we introduce three architectural modifications: chunked causal attention, removal of motion frames, and per-chunk mixed timesteps. In combination, these changes also enable us to keep a compact rolling KV cache as illustrated in Figure 2b.

Chunked causal attention. The student generates videos in chunks of $n_c = 4$ latent frames (~ 0.64 s at the 25 Hz native rate), whereas the teacher generates 30 latent frames (~ 5 s) in a single pass. We enforce causality between chunks in the student: during training, a block-wise causal attention mask prevents frames in chunk C_i from attending to frames in chunks C_j for $j > i$, while attention is bidirectional within a chunk (Figure 2a).

No motion frames. TalkVerse normally conditions each generated video sequence on a window of past frames (motion frames) to maintain temporal coherence across windows. In our causal student, the KV cache already carries the past context required for coherence, thus we remove the motion frame mechanism entirely. This also saves us the cost of a few forward passes through the model required to process motion frames for every new chunk.



(a) **Sliding window causal attention mask used during training.** Each chunk of 4 latent frames can attend to itself, the preceding chunk and the initial chunk acting as a sink. The attention is bidirectional among the 4 frames inside each chunk.

(b) **KV caching with and without per-chunk mixed timesteps.** *Left:* A naive scheme stores one KV cache per denoising step, thus memory scales linearly with the number of denoising steps T . *Right:* With per-chunk mixed timesteps training, each past chunk can be cached at a single noise level, reducing memory cost.

Figure 2: (*Left*) Sliding window chunk-wise causal attention mask used during training by the causal student. (*Right*) Per-chunk mixed denoising timestep training allows the use of a single shared KV cache across all denoising steps at inference.

Per-chunk mixed timesteps: a compact KV cache. At 720p, the KV cache is already large at a single noise level due to the high spatial token count per frame. A naive implementation that would scale one (temporal) KV cache per diffusion step would increase the memory cost by the number of denoising steps per chunk. To avoid this, we train the model to handle sequences in which different chunks are at different noise levels: each chunk is assigned a single diffusion timestep t_i , shared by all frames within the chunk (Fig. 3, center). This *per-chunk mixed timesteps* scheme follows CausVid [52], which adopts it without explicit motivation; we argue it is a key design choice to enable a compact streaming KV cache. At inference, the current chunk is fed at a high noise level t_{cur} while previously generated chunks, cached in the KV cache, are at the lowest noise level reached during their own generation. The cache therefore stores keys and values at a *single* low noise level per past chunk, rather than at every diffusion step (Fig. 2b). In contrast, prior streaming audio-driven models that retain multiple noise levels in the cache pay a proportional memory cost [22].

3.3 Stage 1: ODE Teacher Forcing

As reported by CausVid [52], directly applying DMD to a causal student starting from teacher weights is unstable as the student differs architecturally from the teacher. Therefore, we adapt their ODE-distillation initialization strategy to our setting. Stage 1 trains the causal student to follow the teacher ODE under the new architecture (Fig. 3, top), bringing the student close enough to the teacher distribution so that Stage 2 can converge.

Training data. We train Stage 1 on our synthetic dataset described in Sec. 3.6. Each video comes with latents stored at five timesteps along a 50-step ODE trajectory from the teacher. At training time, the student is queried at the four non-zero timesteps $\{0.25, 0.5, 0.75, 1\}$ and supervised to predict the clean latent at $t = 0$.

Training objective. For each training video, we independently sample a noise level for each chunk uniformly at random from the four cached non-zero timesteps, following the

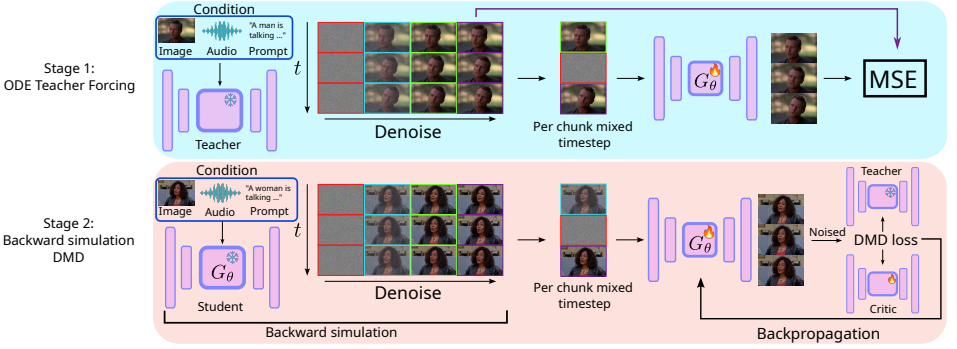


Figure 3: **Two-stage distillation pipeline.** *Stage 1 (top, ODE teacher forcing):* the frozen teacher generates a multi-step ODE trajectory from noise to clean latents, conditioned on a reference image, speech audio, and text prompt. A *per-chunk mixed timestep* state is sampled from this trajectory and fed to the causal student G_θ , which is trained to predict the corresponding clean latents under an MSE loss. *Stage 2 (bottom, DMD with backward simulation):* the student generates a video in inference mode without gradients (backward simulation, left), and a per-chunk mixed-timestep state is sampled from its own denoising trajectory. The student is then re-run on this state with gradients enabled, and a DMD loss is computed by comparing the score predicted by a frozen teacher and an online fake score network (right). Gradients flow back through the gradient-enabled student pass only.

per-chunk mixed timestep scheme of Section 3.2. This matches the heterogeneous-noise configurations the student encounters at inference, where the current chunk is noisy while past chunks (in the KV cache) are near-clean (at lowest noise level). The student processes the resulting sequence under the block-wise causal mask and predicts a velocity field $v_\theta(z_t, t, c)$, where z_t denotes the noised latent at level t (Sec. 3.1) and c denotes the text, audio and identity conditioning. From the predicted velocity we reconstruct the clean latent $\hat{x}_0$ and apply an MSE loss directly on $\hat{x}_0(z_t, t, c)$ rather than on the velocity. The Stage 1 loss reads:

$$\mathcal{L}_{\text{stage1}} = \mathbb{E}_{t, z_t, c} \left[\left\| \hat{x}_0(z_t, t, c) - x_0 \right\|_2^2 \right], \quad (2)$$

where x_0 is the clean latent produced by the teacher. The objective is to match the clean latents of the teacher, not the full flow trajectory: at inference, the 4-step student does not integrate the velocity field but alternates between $\hat{x}_0$ prediction and renoising (see A.2).

3.4 Stage 2: Backward Simulation DMD

Stage 1 yields a causal few-step student that closely follows teacher ODE trajectories, but its sample quality remains bounded by the compression from many-step to few-step inference. In Stage 2 we apply the DMD objective introduced in Section 3.1 (Equation (1)) to push the student output distribution toward the teacher distribution, with a sampling strategy adapted to our causal, chunked setting (Fig. 3, bottom).

Models. We maintain three networks: the frozen teacher f_{teacher} , the trainable student G_θ (initialized from Stage 1), and a trainable critic f_ϕ that estimates $\nabla \log p_{\text{gen}, t}$, playing the role of $s_{\text{gen}, \eta}$ in Eq. 1. Following the two time-scale update rule of DMD2 [60], the critic is initialized from the teacher and updated $U = 5$ times between each student update.

DMD objective. Specializing Eq. 1 to our setup, the student gradient becomes

$$\nabla_{\theta} \mathcal{L}_{\text{DMD}} = \mathbb{E}_{t, \hat{x}_0} \left[(f_{\phi}(\hat{x}_t, t) - f_{\text{teacher}}(\hat{x}_t, t)) \frac{\partial \hat{x}_0}{\partial \theta} \right], \quad \hat{x}_0 = G_{\theta}(z, c), \quad (3)$$

where z is the noise input, c denotes the conditioning (audio, prompt, and reference image), $t \sim \mathcal{U}(0, 1)$ indexes the diffusion noise level, and $\hat{x}_t = \Psi(\hat{x}_0, t)$ is the forward-diffused student sample at level t . The two terms inside the bracket correspond, respectively, to the estimated student score and the teacher score, and their difference approximates the gradient of the reverse KL divergence $\text{KL}(p_{\text{gen}, t} \parallel p_{\text{data}, t})$ (Sec. 3.1). The critic f_{ϕ} is trained jointly with G_{θ} via the standard denoising score-matching loss on student samples.

Backward simulation. A central question in few-step distillation is *which* samples $\hat{x}_0$ to use when computing the DMD gradient. We adopt the backward simulation strategy of DMD2 [60] (Sec. 3.1) and apply it to our causal, chunked student. At each iteration, we proceed in three steps: (i) we run a full multi-step rollout of G_{θ} without backpropagation to obtain the intermediate noisy states $\{\hat{x}_{t_k}\}$ ($t_k \in \{0.25, 0.5, 0.75, 1\}$) along its own sampling trajectory; (ii) we re-feed these states through G_{θ} with gradients enabled, using per-chunk mixed timesteps (Sec. 3.2) sampled uniformly across the trajectory; (iii) we evaluate the DMD gradient (Eq. 3) at the resulting $\hat{x}_0$ (Fig. 3, bottom). The student is thus supervised at the intermediate states it encounters at inference, reducing the training–inference mismatch.

3.5 Inference: Causal Generation with rolling KV Cache

At inference, the student generates the video autoregressively, one chunk of 4 latent frames at a time, in 4 diffusion steps per chunk. Within each chunk, we adopt the predict-and-renoise scheme of DMD2 [60] (see supplementary material A.2). To extend generation to longer sequences, we use a rolling KV cache with a first-chunk anchor that provides temporal context across chunks, and a RoPE [41] re-indexing scheme that bridges training and inference.

Rolling KV cache. The sampling scheme of DMD2 describes the denoising of a single chunk. To generate longer videos, each new chunk must attend to previously generated chunks for temporal coherence. A naive KV cache that stores keys and values for all previously generated chunks would grow linearly with video length. We instead use a *rolling* cache of fixed size, designed around two roles:

1. the **first chunk**, kept as a fixed anchor on which subsequent chunks can condition. This mirrors the *attention sink* mechanism used to stabilize streaming language models [41] and also adopted for streaming long-video generation [28];
2. the **last n chunks** ($n = 1$ in our setting), which provide the immediate temporal context for local motion continuity. We keep a single past chunk ($n = 1$): enlarging the window to $n = 2$ yields no quality improvement while increasing cache size and lowering throughput (supplementary material, Tab. 9).

All other chunks are evicted from the cache once they fall outside this window, which yields constant inference memory regardless of video duration. Moreover, thanks to the per-chunk mixed-timestep training described in Section 3.2, cached entries only need to store keys and values at the lowest noise level reached at the last denoising step of the chunk, rather than at every diffusion step. The same attention pattern is reproduced at training time by the block-causal mask (Figure 2a), in a single forward pass and without maintaining a KV cache.

RoPE indexing. At inference, generated-frame positions grow without bound, while the reference frame is anchored at a fixed temporal RoPE position 10 to 40 latents ahead of the current generation front at training time (Section 3.1 and supplementary material A.1). To preserve this training-time configuration, we re-index the reference every 30 new generated latent frames, shifting it by +30 each time so that its offset relative to the generation front cycles between +40 and +10 (Figure 7 in supplementary material). This keeps the relative position of the reference within the regime the model was trained on, and avoids the position drift that would otherwise accumulate over long autoregressive rollouts.

3.6 Synthetic Talking-Head Dataset

We find that existing audio-driven video datasets are ill-suited to our training setup: the TalkVerse training set is released only as YouTube URLs and is difficult to obtain reliably, while the Hallo3 dataset [9] contains clips too short to match our teacher’s single-chunk generation window of ~ 5 s. We instead build a synthetic dataset by running the TalkVerse teacher on conditioning inputs we control, and release it to support Stage 1 ODE teacher forcing (Sec. 3.3) and future work on audio-driven distillation.

Conditioning inputs. We draw reference images and their associated text prompts from Hallo3 dataset [9], and pair each (image, prompt) to a synthetic speech track: We first obtain textual English audio transcripts of Youtube videos provided in the LiveCC dataset [4]. Then, we generate speech from a subset of these transcript using an off-the-shelf text-to-speech model [63]. The resulting clips average 4.8 s, matching the teacher’s chunk size.

Video generation. For each (image, prompt, audio) triplet, we run the TalkVerse teacher at 720p and 25 Hz with 50 ODE steps and a guidance scale of 4.5. Reference image quality is heterogeneous across Hallo3, which also leads to diverse quality in the resulting video clips; No quality filtering is applied. For each generated video, we also store the teacher’s latent ODE trajectory at five timesteps: the initial noise ($t = 1$), three intermediate states ($t \in \{0.75, 0.5, 0.25\}$), and the final clean latent ($t = 0$). Latents follow the TalkVerse VAE compression ($4 \times 16 \times 16$ over time and space). We additionally store the decoded clean videos as MP4 files alongside the original audio tracks.

Release. The final dataset totals 30,000 clips (38.7 h of synthetic videos, compared to ~ 70 h for Hallo3 and CelebV). We release the audio tracks, decoded videos, and the five-timestep latent ODE trajectories to foster reproducibility and future work on talking heads generation.

4 Experiments

4.1 Experimental Setup

Architecture and initialization. The student is initialized from TalkVerse (5B parameters) and trained with LoRA adapters [40] of rank 128 on the attention layers. We additionally unfreeze the timestep embedding and the output head of the student, since our model uses the per-chunk mixed-timestep scheme of Section 3.2 rather than the single shared timestep of the teacher. Unless stated otherwise, our model generates videos at 720p in chunks of $n_c = 4$ latent frames with 4 diffusion denoising steps per chunk. The output video plays back at 25 Hz, the native output rate of TalkVerse; we use *fps* elsewhere to denote throughput.

Two-stage LoRA training. Stage 1 (ODE teacher forcing) runs for 120K steps (108 h) at learning rate $1e-5$, batch size 256. We then merge the Stage 1 LoRA into the base weights and

re-initialize a fresh LoRA of the same rank for Stage 2 (DMD with backward simulation), avoiding compounded adapters. Stage 2 runs for 5K steps (30 h) at learning rate $3.5e-7$, batch size 256, with 5 critic updates per student update. The critic is initialized from the teacher and, operating on its single shared timestep, has only its LoRA adapters trained. Remaining hyperparameters are in the supplementary material (A.3); all training uses 64 H100 GPUs.

Baselines. We compare AURA against four open-source audio-driven avatar generation models. *TalkVerse* [46], our teacher, serves as a high-quality reference for the distillation. *Wan-S2V 14B* [45] and *Hallo3* [9] are two recent large-scale non-causal S2V model. *Live Avatar* [22] is a streaming few-step approach and the closest competitor to AURA in inference regime. For all baselines we use the authors’ released code and checkpoints, and run inference under each method’s recommended settings.

Benchmarks. We evaluate on two test sets we generate. HALLO3 TEST consists of 100 first frames and prompts randomly sampled from the Hallo3 test set [9], each re-paired with a real English speech track from an internal dataset to assess speaker generalization; all clips are 10 s. CELEBV-HQ TEST consists of 150 first frames from CelebV-HQ [56], each paired with 8–20 s of TTS-synthesized speech (same pipeline as Sec. 3.6) and the generic prompt “a person is talking”; we synthesize rather than reuse audio because CelebV-HQ clips are too short (~ 5 s) to test longer sequences.

Metrics. We evaluate along five axes. *Visual quality*: FID [47], FVD [44] on 16-frame windows, reported at three temporal offsets, @0 (initialization), @101 and @109 (windows centered on and starting at a mid-video chunk boundary, capturing inter-chunk discontinuities), and JEDi [61] at the same offsets, an MMD-based distance on V-JEPA features more reliable at our limited sample counts. *Temporal consistency*: the warping error (WE), the mean L1 residual after RAFT [43] optical-flow alignment of consecutive frames, capturing flicker and chunk-boundary artifacts. *Audio–visual alignment*: Sync-C and Sync-D [8] (SyncNet). *Identity preservation*: CSIM, the mean cosine similarity between ArcFace [48] embeddings of the reference and generated frames. *Inference timing*: on a single H100 GPU, generating one clip from a fixed 10 s English audio and a 960×960 reference at each model’s native resolution and frame rate; we report throughput (fps) for the DiT and VAE decoder separately and the end-to-end *Total* (excluding loading and warm-up). Total is not a simple composition of the two: for Live Avatar, offloading the DiT or VAE between calls to fit in GPU memory adds overhead.

4.2 Main Results

Quality and identity. In Table 1 and 2 we report our main quantitative comparison against all baselines, alongside model size, output resolution, and number of diffusion steps. Despite using only 4 denoising steps instead of 50, AURA matches or improves on its TalkVerse teacher across most automatic metrics, and is comparable to the much larger Wan-S2V (14B, multi-step) on FVD, JEDi, and audio–visual alignment. On identity preservation (CSIM), AURA outperforms all baselines on both test sets (0.80 on HALLO3 TEST, 0.74 on CELEBV-HQ TEST), which we attribute to the reference-frame RoPE indexing scheme described in Sec. 3.2 and to keeping the initial generated chunks in the rolling cache, both of which limit identity drift over long generations.

Speed. AURA is the fastest method, even compared to Live Avatar which is also a few-step model. Throughputs are reported in raw frames per second. AURA, TalkVerse and Hallo3 generate at 25 Hz, Live Avatar and Wan-S2V at 16 Hz. Measured in generated seconds of

Table 1: **Main results on HALLO3 TEST.** FVD and JEDi are reported as triplets at temporal offsets at three different (temporal) frames: @0/@101/@109. *WE*: warping error. *DiT / VAE*: throughput of the diffusion transformer and VAE decoder alone, in frames per second on a single H100 GPU. *Total* (fps): end-to-end throughput, including model offload and overhead. **Bold**: best; underlined: second best (applied per offset within each triplet).

Method	Params	Res	Steps	DiT / VAE (fps)↑	Total (fps)↑	FID↓	FVD @ (0/101/109)↓	JEDi @ (0/101/109)↓	CSIM↑	Sync-C↑	Sync-D↓	WE↓
Hallo3	14.5B	480p	50	0.24/3.2	0.22	43.1	189.0 / 182.8 / 187.1	0.22 / 0.13 / 0.13	0.74	<u>5.06</u>	9.9	0.0036
Wan-S2V	14B	720p	40	0.08/6.7	0.08	55.2	189.3 / 204.8 / 207.4	<u>0.23 / 0.28 / 0.29</u>	0.71	5.40	9.0	<u>0.0032</u>
TalkVerse	5B	720p	50	0.69/7.4	0.58	55.7	228.3 / 201.1 / 217.1	0.43 / 0.39 / 0.44	0.74	3.82	10.5	0.0019
Live Avatar	14B	480p	4	8.4/4.3	1.9	<u>50.8</u>	181.4 / 181.3 / 185.5	0.32 / 0.55 / 0.53	<u>0.77</u>	4.90	<u>9.3</u>	0.0039
AURA (ours)	5B	720p	4	19.7/22.7	9.7	56.2	201.0 / 188.0 / 184.0	0.54 / 0.29 / <u>0.29</u>	0.80	4.17	9.97	0.0034
<i>AURA (3 steps)</i>	<i>5B</i>	<i>720p</i>	<i>3</i>	26.0/22.7	11.2	56.5	215.0 / 196.7 / 187.5	0.53 / 0.29 / <u>0.29</u>	0.77	4.27	9.85	0.0033

Table 2: **Main results on CELEBV-HQ TEST.** FVD and JEDi are averaged over the three temporal offsets (@0, @101, @109); per-offset breakdowns are in supplementary material. Columns and naming conventions follow the same as in Table 1.

Method	Params	Res	Steps	DiT / VAE (fps)↑	Total (fps)↑	FID↓	FVD↓	JEDi↓	CSIM↑	Sync-C↑	Sync-D↓	WE↓
Hallo3	14.5B	480p	50	0.24/3.2	0.22	36.0	208.0	0.257	<u>0.71</u>	6.20	9.4	0.0038
Wan-S2V	14B	720p	40	0.08/6.7	0.08	46.7	293.7	0.640	0.62	6.20	8.9	0.0061
TalkVerse	5B	720p	50	0.69/7.4	0.58	56.3	262.9	0.600	0.59	4.80	10.5	0.0059
Live Avatar	14B	480p	4	8.4/4.3	1.9	<u>43.1</u>	272.9	1.087	0.67	5.90	<u>9.2</u>	0.0057
AURA (ours)	5B	720p	4	19.7/22.7	9.7	45.9	<u>218.9</u>	<u>0.477</u>	0.74	4.70	10.1	<u>0.0055</u>
<i>AURA (3 steps)</i>	<i>5B</i>	<i>720p</i>	<i>3</i>	26.0/22.7	11.2	44.0	239.0	0.577	0.65	4.90	9.9	0.0065

video per second of compute, AURA’s lead over the latter two is larger still. The 4-step variant, our main model, runs the end-to-end pipeline at 9.7 fps on a single H100, a $\sim 17\times$ speedup over the TalkVerse teacher (0.58 fps); on the diffusion transformer alone, the same comparison yields 19.7 vs. 0.69 fps, a $\sim 30\times$ speedup. The 3-step variant pushes these numbers further to 11.2 fps end-to-end ($\sim 20\times$ speedup) and 26.0 fps on the DiT ($\sim 40\times$ speedup), with marginally lower quality. The DiT and the VAE decoder reach 19.7 fps and 22.7 fps respectively on a single H100; end-to-end throughput is bounded by their sequential composition. Figure 1 (right) summarizes the quality–speed trade-off: AURA matches the best baselines on quality while being one to two orders of magnitude faster. Figure 6 in the supplementary material adds complementary visualizations.

Long-form streaming stability. A central claim of AURA is that its rolling KV cache sustains generation over an unbounded stream at constant memory. To verify that quality and identity do not degrade over time, we run the 4-step model on the 100 HALLO3 TEST references driven by 60 s of speech and compute metrics on consecutive 10 s segments (full per-segment breakdown in the supplementary material, Tab. 8). Across the six segments all metrics fluctuate only marginally and show no monotonic trend (mean FVD 187.4, CSIM 0.759, Sync-C 4.74, warping error 3.9×10^{-3}): there is no systematic identity drift or accumulation of chunk-boundary artifacts at the minute scale.

Perceptual quality. Distribution-based metrics (FID, FVD, JEDi) capture global feature statistics but do not penalize certain local artifacts: Hallo3 obtains the best FID, FVD, and JEDi on both test sets, yet its generations exhibit exaggerated mouth motion and recurrent hand artifacts that these metrics fail to flag (qualitative analysis in Fig. 8, supplementary material). To assess perceived quality directly, we run an anonymous human evaluation

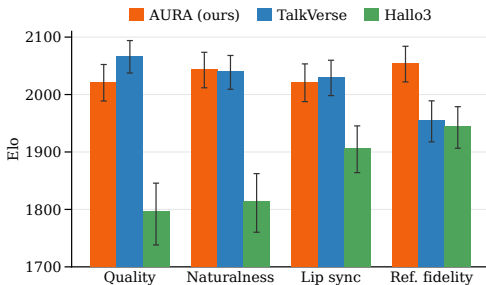


Figure 4: **Human preference study.** Per-axis Bayesian Elo ($\uparrow$) from 300 pairwise comparisons by 30 raters on the HALLO3 TEST clips; error bars are 95% CIs.

on the Mabyduck platform, collecting 300 pairwise comparisons from 30 raters on the 100 HALLO3 TEST clips, comparing AURA against Hallo3 and against our TalkVerse teacher along four axes : overall visual quality, lip-sync, reference fidelity, and naturalness, aggregated into per-axis Bayesian Elo scores (Fig. 4). AURA is preferred over Hallo3 on *all four* axes, confirming that its distribution-metric lead does not translate into perceived quality. Against the non-causal, 50-step teacher, AURA ranks *higher* on reference fidelity, consistent with our CSIM results, is on par for lip-sync and naturalness, and only marginally behind on visual quality, despite being causal and $\sim 17\times$ faster.

4.3 Ablations

We ablate the main design choices of our method. Unless stated otherwise, all ablations use the 5B student trained on the full pipeline and are evaluated on the HALLO3 TEST set with 6.3 s clips, shorter than the 10 s clips used in the main results (Sec. 4.2) to reduce compute.

Number of diffusion steps. Table 3 reports quality metrics when generating each chunk in $\{1, 2, 3, 4\}$ diffusion steps. Per-frame and per-window quality (FID, FVD, JEDi) degrades gradually from 4 to 2 steps and drops sharply at 1 step. Identity preservation (CSIM) follows the same trend. Lip sync quality (Sync-C, Sync-D) remains essentially constant across all step counts. Temporal consistency metric (Warp Error) reaches lower values as the number of steps decreases, but this reflects a known failure mode of these metrics rather than a real improvement: with fewer steps, generated frames become blurrier and lose high-frequency content, which leaves fewer details for the metrics to flag as inconsistent across frames. Qualitatively, videos become progressively blurrier as the number of steps decreases (Sec. 4.4). We adopt 4 steps in all other experiments as the best trade-off between per-frame quality and inference speed.

Stage 1 only vs. Stage 1 + Stage 2. Table 4 compares a student trained only with ODE teacher forcing (Stage 1) against the full two-stage pipeline (Stage 1 + Stage 2). Adding the DMD stage substantially improves all perceptual quality metrics (FID, FVD, JEDi) and identity preservation (CSIM), confirming that Stage 1 alone is insufficient to close the gap to the teacher in the few-step regime. Lip-sync metrics also improve. As in the previous ablation, the temporal consistency metric (Warp) reaches lower value with Stage 1 only, but this reflects the same metric artifact on blurrier outputs rather than a real improvement.

KV cache refresh with an additional step. We tested three variants where an additional diffusion step is added at inference to refresh the KV cache with a less noisy state for the

Table 3: **Effect of the number of diffusion steps** per chunk on quality and speed. DiT throughput is reported in frames per second on a single H100. FVD and JEDi are averaged over three temporal offsets (@0, @101, @109). All other experiments use 4 steps. Best per column in **bold**.

#Steps	FID ↓	FVD ↓	JEDi ↓	CSIM ↑	Sync-C ↑	Sync-D ↓	WE ↓	DiT (fps) ↑
4	57.2	191.0	0.37	0.80	4.09	10.02	0.0034	19.7
3	57.4	199.7	0.37	0.77	4.18	9.91	0.0033	26.0
2	57.7	200.1	0.42	0.80	4.18	9.93	0.0032	38.5
1	60.2	231.7	0.65	0.65	4.09	9.77	0.0031	75.8

Variant	FID ↓	FVD ↓	JEDi ↓	CSIM ↑	Sync-C ↑	Sync-D ↓	WE ↓
Stage 1 only	73.0	267.0	0.75	0.55	3.36	10.17	0.0026
Stage 1 + Stage 2	57.2	191.0	0.37	0.80	4.09	10.02	0.0034

Table 4: **Effect of Stage 2 on generation quality**. Adding the DMD distillation stage substantially improves perceptual quality (FID, FVD, JEDi), identity preservation (CSIM), and lip-sync. Warp reaches lower values with Stage 1 only, but this is the same metric artifact on blurrier outputs as observed in Tab. 3 (see Sec. 4.4). Best value per column in **bold**.

next chunk. Stage 1 is fine-tuned to additionally handle a low timestep $t = 0$ in its ODE trajectories, and Stage 2 is retrained from scratch with this extra low timestep included in the backward simulation: $t = 0$ in *5-step* ($t = 0$ refresh) and $t = 50$ in the two *5-step* ($t = 50$) variants. At inference, this extra step updates the cache for the next chunk; its output is discarded in the first two variants and used as the final chunk prediction in the third. Table 5 reports the results. None of the variants consistently outperforms the simpler 4-step default: FID and lip-sync slightly improve in some variants, but FVD, JEDi, and identity preservation are either unchanged or slightly degraded. Given the added inference cost, we keep the 4-step variant as default.

Table 5: **KV cache refresh strategy**. Effect of an additional inference step to refresh it. Best value per column in **bold**.

Variant	FID ↓	FVD ↓	JEDi ↓	CSIM ↑	Sync-C ↑	WE ↓	DiT (fps) ↑
4 steps (default)	57.2	191.0	0.37	0.80	4.09	0.0034	19.7
5-step ($t=0$ refresh)	56.6	197.7	0.37	0.79	4.03	0.0035	15.6
5-step ($t=50$ refresh)	54.9	196.1	0.40	0.78	4.08	0.0036	15.6
5-step ($t=50$, used)	53.6	198.3	0.40	0.77	4.11	0.0037	15.6

4.4 Qualitative Results

Figures 1 and 5 show representative AURA generations on different test sequences spanning varied lighting conditions, head poses, and identities. The generated frames remain faithful to the reference identity throughout the sequence, and exhibit natural motion patterns –



Figure 5: **Qualitative results from AURA** on two diverse test sequences with different audios. Each row shows the reference identity image (left) and three frames sampled from the generated video at approximately $t = 1, 3, 5$ s. The samples illustrate the diversity of inputs AURA handles, including varied lighting, head poses, and identities. Full videos are provided in the supplementary material.

including spontaneous eye blinks, subtle head movements, and small facial expression variations – that go beyond strict lip motion.

Figure 9 in supplementary material compares AURA to its TalkVerse teacher on the same input. Despite using only 4 denoising steps instead of 50, AURA matches the teacher’s visual quality, and in some cases improves over it: in the example shown, the TalkVerse output produces a static gaze with no blinking over the full 10 s clip, whereas AURA generates natural eye blinks and more varied global motion. We provide the full videos for all sequences in the supplementary material. We also provide video samples from the 3-step model in the supplementary material.

5 Conclusions

We presented AURA, a 5B causal few-step speech-to-video model for streaming talking-head generation. Distilling a bidirectional teacher into a same-size 4-step causal student through a two-stage pipeline and a rolling KV cache, AURA matches its TalkVerse teacher on FVD and JEDi at a $17\times$ speedup, delivering 720p video at 9.7 fps on a single GPU. We also release a 30,000-clip synthetic dataset with teacher ODE trajectories to support future work on audio-driven distillation. Two limitations remain. The rolling KV cache preserves global identity, yet slight discontinuities can still appear at chunk boundaries on long sequences; and at 4 steps AURA remains short of the 25 Hz native rate required for true real-time generation on a single GPU. Distilling to a smaller backbone or further reducing the number of denoising steps is a natural path toward real-time, fluid streaming for interactive avatars.

References

- [1] Alexei Baevski, Henry Zhou, Abdelrahman Mohamed, and Michael Auli. wav2vec 2.0: A framework for self-supervised learning of speech representations. *Advances in neural information processing systems*, 33:12449–12460, 2020.
- [2] Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilianski, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, Varun Jampani, and Robin Rombach. Stable video diffusion: Scaling latent video diffusion models to large datasets. *Computing Research Repository (CoRR)*, abs/2311.15127, 2023. URL <https://doi.org/10.48550/arXiv.2311.15127>.
- [3] Andreas Blattmann, Robin Rombach, Huan Ling, Tim Dockhorn, Seung Wook Kim, Sanja Fidler, and Karsten Kreis. Align your latents: High-resolution video synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, CVPR’23*, pages 22563–22575, 2023.
- [4] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale GAN training for high fidelity natural image synthesis. In *International Conference on Learning Representations, ICLR’19*, 2019.
- [5] Tim Brooks, Bill Peebles, Connor Holmes, Will DePue, Yufei Guo, Li Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, Clarence Ng, Ricky Wang, and Aditya Ramesh. Video generation models as world simulators, 2024. URL <https://openai.com/research/video-generation-models-as-world-simulators>.
- [6] Justine Cassell, Joseph Sullivan, Scott Prevost, and Elizabeth F. Churchill, editors. *Embodied Conversational Agents*. MIT Press, Cambridge, MA, 2000. ISBN 9780262032780.
- [7] Joya Chen, Ziyun Zeng, Yiqi Lin, Wei Li, Zejun Ma, and Mike Zheng Shou. Livecc: Learning video llm with streaming speech transcription at scale. In *CVPR*, 2025.
- [8] Joon Son Chung and Andrew Zisserman. Out of time: automated lip sync in the wild. In *Workshop on Multi-view Lip-reading, ACCV*, 2016.
- [9] Jiahao Cui, Hui Li, Yun Zhan, Hanlin Shang, Kaihui Cheng, Yuqi Ma, Shan Mu, Hang Zhou, Jingdong Wang, and Siyu Zhu. Hallo3: Highly dynamic and realistic portrait image animation with video diffusion transformer. In *Proceedings of the Computer Vision and Pattern Recognition Conference, CVPR’25*, pages 21086–21095, 2025.
- [10] Justin Cui, Jie Wu, Ming Li, Tao Yang, Xiaojie Li, Rui Wang, Andrew Bai, Yuanhao Ban, and Cho-Jui Hsieh. Self-forcing++: Towards minute-scale high-quality video generation. In *The Fourteenth International Conference on Learning Representations, ICLR’26*, 2026.
- [11] Liyuan Cui, Wentao Hu, Wenyan Zhang, Zesong Yang, Fan Shi, and Xiaoqiang Liu. Avatarforcing: One-step streaming talking avatars via local-future sliding-window denoising, 2026. URL <https://arxiv.org/abs/2603.14331>.

- [12] Alexandre Défossez, Laurent Mazaré, Manu Orsini, Amélie Royer, Patrick Pérez, Hervé Jégou, Edouard Grave, and Neil Zeghidour. Moshi: a speech-text foundation model for real-time dialogue, 2024. URL <https://arxiv.org/abs/2410.00037>.
- [13] Jiankang Deng, Jia Guo, Jing Yang, Niannan Xue, Irene Kotsia, and Stefanos Zafeiriou. ArcFace: Additive angular margin loss for deep face recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(10):5962–5979, October 2022. ISSN 1939-3539. doi: 10.1109/tpami.2021.3087709. URL <http://dx.doi.org/10.1109/TPAMI.2021.3087709>.
- [14] Prafulla Dhariwal and Alex Nichol. Diffusion Models Beat GANs on Image Synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021.
- [15] Xin Gao, Li Hu, Siqi Hu, Mingyang Huang, Chaonan Ji, Dechao Meng, Jinwei Qi, Pengchong Qiao, Zhen Shen, Yafei Song, Ke Sun, Linrui Tian, Guangyuan Wang, Qi Wang, Zhongjian Wang, Jiayu Xiao, Sheng Xu, Bang Zhang, Peng Zhang, Xindi Zhang, Zhe Zhang, Jingren Zhou, and Lian Zhuo. Wan-S2V: Audio-driven cinematic video generation, 2025. URL <https://arxiv.org/abs/2508.18621>.
- [16] 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.
- [17] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017.
- [18] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.
- [19] Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P. Kingma, Ben Poole, Mohammad Norouzi, David J. Fleet, and Tim Salimans. Imagen video: High definition video generation with diffusion models, 2022. URL <https://arxiv.org/abs/2210.02303>.
- [20] Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations, ICLR’22*, 2022. URL <https://openreview.net/forum?id=nZeVKeeFYf9>.
- [21] Xun Huang, Zhengqi Li, Guande He, Mingyuan Zhou, and Eli Shechtman. Self forcing: Bridging the train-test gap in autoregressive video diffusion. *Advances in Neural Information Processing Systems*, 38:167283–167308, 2026.
- [22] Yubo Huang, Hailong Guo, Fangtai Wu, Shifeng Zhang, Shijie Huang, Qijun Gan, Lin Liu, Sirui Zhao, Enhong Chen, Jiaming Liu, and Steven Hoi. Live avatar: Streaming real-time audio-driven avatar generation with infinite length, 2026. URL <https://arxiv.org/abs/2512.04677>.

- [23] 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, CVPR'19*, pages 4401–4410, 2019.
- [24] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *Advances in neural information processing systems*, 35:26565–26577, 2022.
- [25] Weijie Kong, Qi Tian, Zijian Zhang, Rox Min, Zuozhuo Dai, Jin Zhou, Jiangfeng Xiong, Xin Li, Bo Wu, Jianwei Zhang, Kathrina Wu, Qin Lin, Junkun Yuan, Yanxin Long, Aladdin Wang, Andong Wang, Changlin Li, Duojun Huang, Fang Yang, Hao Tan, Hongmei Wang, Jacob Song, Jiawang Bai, Jianbing Wu, Jinbao Xue, Joey Wang, Kai Wang, Mengyang Liu, Pengyu Li, Shuai Li, Weiyan Wang, Wenqing Yu, Xincheng Deng, Yang Li, Yi Chen, Yutao Cui, Yuanbo Peng, Zhentao Yu, Zhiyu He, Zhiyong Xu, Zixiang Zhou, Zunnan Xu, Yangyu Tao, Qinglin Lu, Songtao Liu, Dax Zhou, Hongfa Wang, Yong Yang, Di Wang, Yuhong Liu, Jie Jiang, and Caesar Zhong. Hunyuanvideo: A systematic framework for large video generative models, 2025. URL <https://arxiv.org/abs/2412.03603>.
- [26] Gaojie Lin, Jianwen Jiang, Jiaqi Yang, Zerong Zheng, Chao Liang, Yuan Zhang, and Jingtuo Liu. Omnihuman-1: Rethinking the scaling-up of one-stage conditioned human animation models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision, ICCV'25*, pages 13847–13858, 2025.
- [27] Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *The Eleventh International Conference on Learning Representations, ICLR'23*, 2023. URL <https://openreview.net/forum?id=PqvMRDCJT9t>.
- [28] Kunhao Liu, Wenbo Hu, Jiale Xu, Ying Shan, and Shijian Lu. Rolling forcing: Autoregressive long video diffusion in real time, 2026.
- [29] Xingchao Liu, Chengyue Gong, and qiang liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *The Eleventh International Conference on Learning Representations*, 2023. URL <https://openreview.net/forum?id=XVjTT1nw5z>.
- [30] Chetwin Low and Weimin Wang. Talkingmachines: Real-time audio-driven facetime-style video via autoregressive diffusion models, 2025. URL <https://arxiv.org/abs/2506.03099>.
- [31] Ge Ya Luo, Gian Mario Favero, Zhi Hao Luo, Alexia Jolicoeur-Martineau, and Christopher Pal. Beyond FVD: An enhanced evaluation metrics for video generation distribution quality. In *International Conference on Learning Representations, ICLR'25*, volume 2025, pages 66218–66252, 2025.
- [32] Harry McGurk and John MacDonald. Hearing lips and seeing voices. *Nature*, 264 (5588):746–748, December 1976. doi: 10.1038/264746a0.
- [33] Adam Polyak, Amit Zohar, Andrew Brown, Andros Tjandra, Animesh Sinha, Ann Lee, Apoorv Vyas, Bowen Shi, Chih-Yao Ma, Ching-Yao Chuang, David Yan, Dhruv

- Choudhary, Dingkan Wang, Geet Sethi, Guan Pang, Haoyu Ma, Ishan Misra, Ji Hou, Jialiang Wang, Kiran Jagadeesh, Kunpeng Li, Luxin Zhang, Mannat Singh, Mary Williamson, Matt Le, Matthew Yu, Mitesh Kumar Singh, Peizhao Zhang, Peter Vajda, Quentin Duval, Rohit Girdhar, Roshan Sumbaly, Sai Saketh Rambhatla, Sam Tsai, Samaneh Azadi, Samyak Datta, Sanyuan Chen, Sean Bell, Sharadh Ramaswamy, Shelly Sheynin, Siddharth Bhattacharya, Simran Motwani, Tao Xu, Tianhe Li, Tingbo Hou, Wei-Ning Hsu, Xi Yin, Xiaoliang Dai, Yaniv Taigman, Yaqiao Luo, Yen-Cheng Liu, Yi-Chiao Wu, Yue Zhao, Yuval Kirstain, Zecheng He, Zijian He, Albert Pumarola, Ali Thabet, Artsiom Sanakoyeu, Arun Mallya, Baishan Guo, Boris Araya, Breana Kerr, Carleigh Wood, Ce Liu, Cen Peng, Dmitry Vengertsev, Edgar Schonfeld, Elliot Blanchard, Felix Juefei-Xu, Fraylie Nord, Jeff Liang, John Hoffman, Jonas Kohler, Kaolin Fire, Karthik Sivakumar, Lawrence Chen, Licheng Yu, Luya Gao, Markos Georgopoulos, Rashel Moritz, Sara K. Sampson, Shikai Li, Simone Parmeggiani, Steve Fine, Tara Fowler, Vladan Petrovic, and Yuming Du. Movie gen: A cast of media foundation models, 2025. URL <https://arxiv.org/abs/2410.13720>.
- [34] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140): 1–67, 2020.
- [35] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, CVPR’22*, pages 10684–10695, 2022.
- [36] Rajarshi Roy, Jonathan Raiman, Sanggil Lee, Teodor-Dumitru Ene, Robert Kirby, Sungwon Kim, Jaehyeon Kim, and Bryan Catanzaro. Personaplex: Voice and role control for full duplex conversational speech models, 2026. URL <https://arxiv.org/abs/2602.06053>.
- [37] Le Shen, Qian Qiao, Tan Yu, Ke Zhou, Tianhang Yu, Yu Zhan, Zhenjie Wang, Ming Tao, Shunshun Yin, and Siyuan Liu. SoulX-FlashTalk: Real-time infinite streaming of audio-driven avatars via self-correcting bidirectional distillation, 2026. URL <https://arxiv.org/abs/2512.23379>.
- [38] Uriel Singer, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyan Hu, Harry Yang, Oron Ashual, Oran Gafni, Devi Parikh, Sonal Gupta, and Yaniv Taigman. Make-a-video: Text-to-video generation without text-video data. In *The Eleventh International Conference on Learning Representations, ICLR’23*, 2023.
- [39] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021. URL <https://openreview.net/forum?id=StlgiaRCHLP>.
- [40] Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations, ICLR’21*, 2021.
- [41] Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568: 127063, 2024.

- [42] W. H. Sumby and Irwin Pollack. Visual contribution to speech intelligibility in noise. *The Journal of the Acoustical Society of America*, 26(2):212–215, March 1954. doi: 10.1121/1.1907309.
- [43] Zachary Teed and Jia Deng. RAFT: Recurrent all-pairs field transforms for optical flow. In *European conference on computer vision, ECCV’20*, pages 402–419. Springer, 2020.
- [44] Thomas Unterthiner, Sjoerd van Steenkiste, Karol Kurach, Raphael Marinier, Marcin Michalski, and Sylvain Gelly. Towards accurate generative models of video: A new metric & challenges, 2019. URL <https://arxiv.org/abs/1812.01717>.
- [45] Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, Jianyuan Zeng, Jiayu Wang, Jingfeng Zhang, Jingren Zhou, Jinkai Wang, Jixuan Chen, Kai Zhu, Kang Zhao, Keyu Yan, Lianghua Huang, Mengyang Feng, Ningyi Zhang, Pandeng Li, Pingyu Wu, Ruihang Chu, Ruili Feng, Shiwei Zhang, Siyang Sun, Tao Fang, Tianxing Wang, Tianyi Gui, Tingyu Weng, Tong Shen, Wei Lin, Wei Wang, Wei Wang, Wenmeng Zhou, Wenten Wang, Wenting Shen, Wenyuan Yu, Xianzhong Shi, Xiaoming Huang, Xin Xu, Yan Kou, Yangyu Lv, Yifei Li, Yijing Liu, Yiming Wang, Yingya Zhang, Yitong Huang, Yong Li, You Wu, Yu Liu, Yulin Pan, Yun Zheng, Yuntao Hong, Yupeng Shi, Yutong Feng, Zeyinzi Jiang, Zhen Han, Zhi-Fan Wu, and Ziyu Liu. Wan: Open and advanced large-scale video generative models, 2025. URL <https://arxiv.org/abs/2503.20314>.
- [46] Zhenzhi Wang, Jian Wang, Ke Ma, Dahua Lin, and Bing Zhou. Talkverse: Democratizing minute-long audio-driven video generation, 2025. URL <https://arxiv.org/abs/2512.14938>.
- [47] Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. Efficient streaming language models with attention sinks. In *International Conference on Learning Representations, ICLR’24*, volume 2024, pages 21875–21895, 2024.
- [48] Shuai Yang, Wei Huang, Ruihang Chu, Yicheng Xiao, Yuyang Zhao, Xianbang Wang, Muyang Li, Enze Xie, Yingcong Chen, Yao Lu, Song Han, and Yukang Chen. LongLive: Real-time interactive long video generation, 2025. URL <https://arxiv.org/abs/2509.22622>.
- [49] Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, Da Yin, Yuxuan Zhang, Weihang Wang, Yean Cheng, Bin Xu, Xiaotao Gu, Yuxiao Dong, and Jie Tang. Cogvideox: Text-to-video diffusion models with an expert transformer. In *International Conference on Learning Representations, ICLR’25*, volume 2025, pages 83048–83077, 2025.
- [50] Tianwei Yin, Michaël Gharbi, Taesung Park, Richard Zhang, Eli Shechtman, Fredo Durand, and William T. Freeman. Improved distribution matching distillation for fast image synthesis. *Advances in neural information processing systems*, 37:47455–47487, 2024.

- [51] Tianwei Yin, Michaël Gharbi, Richard Zhang, Eli Shechtman, Fredo Durand, William T. Freeman, and Taesung Park. One-step diffusion with distribution matching distillation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, CVPR'24*, pages 6613–6623, 2024.
- [52] Tianwei Yin, Qiang Zhang, Richard Zhang, William T. Freeman, Fredo Durand, Eli Shechtman, and Xun Huang. From slow bidirectional to fast autoregressive video diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR'25*, pages 22963–22974, 2025.
- [53] Neil Zeghidour, Eugene Kharitonov, Manu Orsini, Václav Volhejn, Gabriel de Marmiesse, Edouard Grave, Patrick Pérez, Laurent Mazaré, and Alexandre Défossez. Streaming sequence-to-sequence learning with delayed streams modeling. Technical report, 2025. URL <https://arxiv.org/abs/2509.08753>.
- [54] Lvmin Zhang, Shengqu Cai, Muyang Li, Gordon Wetzstein, and Maneesh Agrawala. Frame context packing and drift prevention in next-frame-prediction video diffusion models. *Advances in Neural Information Processing Systems*, 38:30546–30566, 2026.
- [55] Zangwei Zheng, Xiangyu Peng, Tianji Yang, Chenhui Shen, Shenggui Li, Hongxin Liu, Yukun Zhou, Tianyi Li, and Yang You. Open-sora: Democratizing efficient video production for all, 2024. URL <https://arxiv.org/abs/2412.20404>.
- [56] Hao Zhu, Wayne Wu, Wentao Zhu, Liming Jiang, Siwei Tang, Li Zhang, Ziwei Liu, and Chen Change Loy. CelebV-HQ: A large-scale video facial attributes dataset. In *European conference on computer vision, ECCV'22*, pages 650–667. Springer, 2022.
- [57] Hongzhou Zhu, Min Zhao, Guande He, Hang Su, Chongxuan Li, and Jun Zhu. Causal forcing: Autoregressive diffusion distillation done right for high-quality real-time interactive video generation, 2026. URL <https://arxiv.org/abs/2602.02214>.

A Supplementary Material

A.1 TalkVerse

We use TalkVerse [46] as our teacher model, a 5B-parameter audio-driven video generation model built on the Wan2.2-5B [45] Diffusion Transformer (DiT) backbone, originally pretrained for text-and-image-to-video generation. TalkVerse extends the Wan-S2V [15] design for speech-conditioned video synthesis to the Wan2.2 backbone, and trades the 14B-parameter scale of Wan-S2V for a smaller 5B model, trained on a high-resolution open-source dataset.

Latent space. TalkVerse uses the Wan2.2 video VAE, which downsamples by $(t, h, w) = (4, 16, 16)$, a higher compression ratio than the $(4, 8, 8)$ commonly used in prior video diffusion models (e.g. Wan2.1). This yields a $4\times$ reduction in latent token count at the same resolution, enabling efficient high-resolution (720p) generation.

Conditioning. The model is conditioned on three inputs: (i) a reference identity image, (ii) a speech audio track, and (iii) a text prompt describing the video content. Audio is encoded with Wav2Vec2 [10] and injected into a subset of DiT blocks via cross-attention from visual tokens to audio tokens, with each latent frame attending to a temporally aligned audio segment. The text prompt is encoded with T5-XXL [34] and injected through cross-attention. TalkVerse is trained on the large-scale TalkVerse-2.3M dataset of high-resolution audio-synchronized clips.

Long-video generation: temporal context. To generate minute-long videos, TalkVerse uses a sliding-window mechanism: each window denoises a fixed number of frames conditioned on a window of preceding frames, referred to as *motion frames*. To keep the token budget tractable, the motion frames are compressed by a FramePack module [64] and assigned negative positional embeddings to mark them as past context.

Long-video generation: reference frame placement. The reference identity image is assigned a positional embedding placed after the denoised video window, specifically position 40 for a denoised window of 30 latent frames. This forward offset makes the model “chase” the reference rather than copy it, which the authors report avoids copy-paste artifacts and looped body movements when the same reference is reused across sliding windows.

Training and inference. TalkVerse is trained with LoRA on the DiT attention layers, while audio projection, audio cross-attention, and FramePack-related modules are fully fine-tuned. Inference uses 50 denoising steps with a flow-matching solver and classifier-free guidance.

A.2 Sampling scheme

Within each chunk, we adopt the predict-and-renoise scheme of DMD2 [50]. Standard flow matching integration $z_{t-\Delta t} = z_t + v_\theta(z_t, t, c) \cdot \Delta t$ assumes that intermediate states lie on the learned ODE trajectory, an assumption that breaks down in the few-step regime, where each step spans a large interval of noise levels. Given a schedule of $K = 4$ denoising steps with timesteps $1 = t_0 > t_1 > \dots > t_K = 0$, the chunk is initialized as $z_{t_0} \sim \mathcal{N}(0, I)$, and each step k proceeds in two parts. The student first predicts the velocity $v_\theta(z_{t_k}, t_k, c)$ and reconstructs an estimate of the clean latent under the linear flow matching parameterization:

$$\hat{x}_0^{(k)} = z_{t_k} - t_k \cdot v_\theta(z_{t_k}, t_k, c). \quad (4)$$

This estimate is then re-noised to the next noise level t_{k+1} with fresh Gaussian noise $\varepsilon \sim \mathcal{N}(0, I)$:

$$z_{t_{k+1}} = (1 - t_{k+1}) \cdot \hat{x}_0^{(k)} + t_{k+1} \cdot \varepsilon. \quad (5)$$

The final latent of the chunk is $\hat{x}_0 = \hat{x}_0^{(K-1)}$, obtained at the last denoising step. The re-noising in Eq. 5 projects each intermediate estimate back onto the manifold of noisy latents the student was trained on, rather than following a straight-line integration of the velocity field. The same scheme is used during Stage 2 training to roll out the student without backpropagation and produce the noisy states $\{\hat{x}_{t_k}\}$ used by backward simulation (Sec. 3.4), keeping training and inference aligned.

A.3 Training Hyperparameters

Table 6: **Training hyperparameters.** Stage 1 trains the student via ODE teacher forcing; Stage 2 trains the student and critic jointly via DMD with backward simulation, alternating 5 critic updates per student update. All trainings use LoRA adapters with the optimizer settings below.

	Stage 1	Stage 2 (student)	Stage 2 (critic)
Training steps	120K	≈ 833	$\approx 4,167$
Global batch size	256	256	256
Learning rate	1e-5	3.5e-7	3.5e-7
Optimizer	AdamW	AdamW	AdamW
Optimizer (β_1, β_2)	(0.9, 0.999)	(0.9, 0.999)	(0.9, 0.999)
Weight decay	0.01	0.01	0.01
LoRA rank r	128	128	128
LoRA α	128	128	128
LoRA target modules	attention	attention	attention
Extra unfrozen params	timestep emb., head	timestep emb., head	—
Initialization	TalkVerse	Stage 1 merged	TalkVerse
Stage 2 total optimizer steps	—	5K	
Critic-to-student update ratio	—	5:1	
Video clip length (latent frames)	30	30	
Teacher ODE steps (Stage 1 data)	50	—	—
Teacher guidance scale (Stage 1 data)	4.5	—	—
Hardware		64× H100	
Precision		bf16	

A.4 Full metrics for CELEBV-HQ TEST

Table 7 reports the full per-offset results on the CELEBV-HQ TEST set.

A.5 Bubble plot for FVD and JEDI

Figure 6 shows the bubble plot for both FVD and JEDI metrics.

Method	Params	Res	Steps	fps $\uparrow$	FID $\downarrow$	FVD (@0/@101/@109) $\downarrow$	JEDi (@0/@101/@109) $\downarrow$	CSIM $\uparrow$	Sync-C/D	WE $\downarrow$
Hallo3	14.5B	480p	50	0.22	36.0	<u>195.7/214.9/213.5</u>	0.26/0.27/0.24	<u>0.71</u>	6.20/9.4	0.0038
Wan-S2V	14B	720p	40	0.08	46.7	298.5/293.0/289.5	0.77/0.56/ <u>0.59</u>	0.62	6.20/8.9	0.0061
TalkVerse	5B	720p	50	0.58	56.3	240.8/274.4/273.6	<u>0.35</u> /0.69/0.76	0.59	4.80/10.5	0.0059
Live Avatar	14B	480p	4	1.90	<u>43.1</u>	241.6/287.5/289.6	0.53/1.36/1.37	0.67	5.90/ <u>9.2</u>	0.0057
AURA (ours)	5B	720p	4	9.7	45.9	187.0/234.0/235.8	0.41/ <u>0.38</u> /0.64	0.74	4.70/10.1	<u>0.0055</u>
AURA (3 steps)	5B	720p	3	11.2	44.0	209.0/249.0/259.0	0.73/0.61/0.39	0.65	4.90/9.9	0.0065

Table 7: **Per-offset breakdown on CELEBV-HQ TEST**, complementing the averaged version in the main paper. FVD and JEDi are reported as triplets at temporal offsets @0/@101/@109. *WE*: warping error. *Sync-C/D*: Sync-C ($\uparrow$) / Sync-D ($\downarrow$). *fps*: end-to-end throughput on a single H100 GPU for a 10 s clip; per-component DiT and VAE values appear in the main paper. **Bold**: best; underlined: second best (per offset within each triplet).

Table 8: **Long-form evaluation**. AURA (4 steps) on the 100 HALLO3 TEST references driven by 60 s of speech; metrics per 10 s segment. Warping error in 10^{-3} .

Metric	0–10s	10–20s	20–30s	30–40s	40–50s	50–60s	Mean
FVD $\downarrow$	186.7	189.6	181.2	189.7	189.3	187.7	187.4
CSIM $\uparrow$	0.784	0.779	0.767	0.758	0.713	0.754	0.759
Sync-C $\uparrow$	4.81	4.99	4.85	4.67	4.46	4.65	4.74
Warp $\downarrow$	3.82	3.80	3.78	3.67	4.30	3.78	3.86

A.6 Full Results for long evaluations

Table 8 provides results for long evaluations.

A.7 RoPE re-indexing scheme

Figure 7 illustrates the RoPE re-indexing scheme.

A.8 Rolling cache window size

We ablate the number of past chunks n retained in the rolling KV cache (Sec. 3.5). Our default keeps a single past chunk ($n = 1$) alongside the first-chunk anchor; here we compare it against a larger window that also retains the second-to-last chunk ($n = 2$). Both variants use the same 4-step student and are evaluated on the HALLO3 validation subset (6.3 s clips). As reported in Table 9, enlarging the window to $n = 2$ improves no metric, while increasing cache size and lowering throughput. This supports our minimal single-chunk cache as the best quality–efficiency trade-off.

A.9 Hallo3 failure case

Figure 8 shows artifacts observed in some Hallo3 outputs.

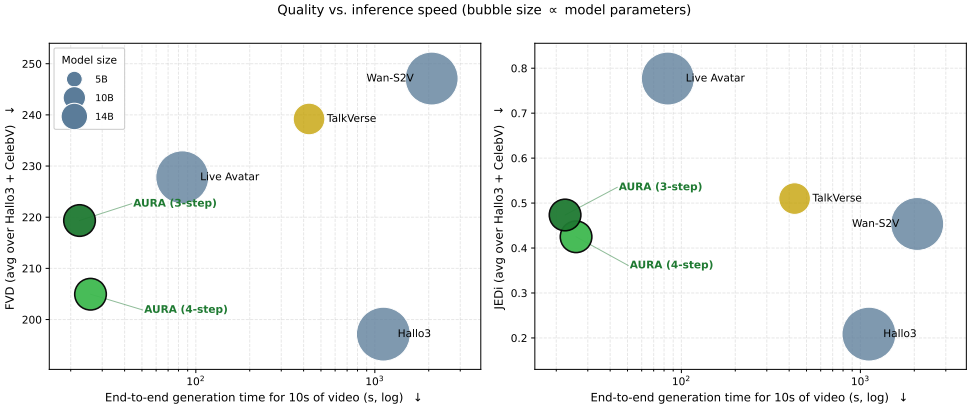


Figure 6: **Quality–speed trade-off across audio-driven avatar generation models.** The horizontal axis reports end-to-end generation time for 10 s of video on a single H100 GPU (log scale, lower is better). The vertical axis reports FVD (*left*) and JEDi (*right*), averaged over the Hallo3 and CelebV-HQ test sets (lower is better). Bubble area is proportional to the number of model parameters. AURA (*green*) achieves quality comparable to multi-step baselines at a fraction of their generation time.

Variant	FID ↓	FVD ↓	JEDi ↓	CSIM ↑	Sync-C ↑	Warp ↓	DiT (fps) ↑
$n = 1$ (default)	57.2	191.0	0.37	0.80	4.09	0.0034	20
$n = 2$	57.8	200.8	0.44	0.79	4.00	0.0037	18

Table 9: **Rolling cache window size.** Effect of the number of retained past chunks n on quality and speed, on the HALLO3 validation subset (6.3 s clips). Enlarging the window from $n = 1$ to $n = 2$ improves no metric while increasing cache size and reducing throughput. FVD and JEDi are averaged over the three temporal offsets (@0, @101, @109). Best per column in **bold**.

A.10 Comparison with teacher

Figure 9 compares AURA and TalkVerse samples generated from the same reference image and audio input.

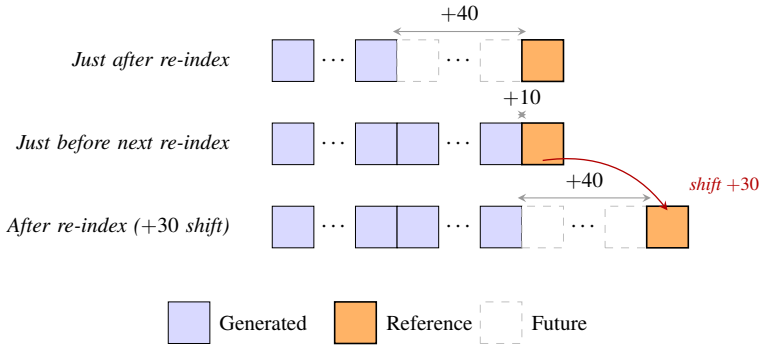


Figure 7: **RoPE re-indexing of the reference frame.** (*Top*) just after a re-indexation, the reference (orange) sits 40 latent positions ahead of the generation front. (*Middle*) after 30 generated frames, the reference is 10 positions ahead — the minimum offset reached during training. (*Bottom*) the reference is shifted by +30, restoring the maximum +40 offset. The offset thus cycles in $[+10, +40]$, staying within the training-time regime.



AURA (ours)



Hallo3

Figure 8: **Qualitative comparison on the Hallo3 test set.** Despite stronger FID/FVD scores (Table 1), Hallo3 generations exhibit recurrent hand artifacts (*bottom*) that distribution-based metrics fail to penalize. AURA (*top*) produces cleaner outputs at a fraction of the generation time.

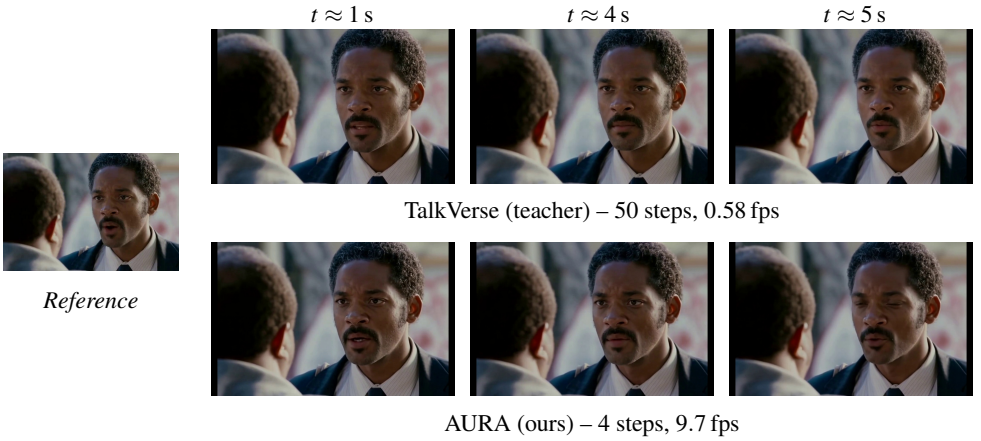


Figure 9: **Qualitative comparison between AURA and its TalkVerse teacher.** Given the same reference image (*left*) and a 10 s audio input, AURA (*bottom*) generates video of comparable visual quality to the teacher (*top*) while running approximately $17\times$ faster end-to-end on a single H100 GPU. Full videos are provided in the supplementary material.