EDGE-AVATAR: POSE-CONTROLLED REAL-TIME SPEECH-TO-VIDEO ON A SINGLE CONSUMER GPU

Anonymous authors
Paper under double-blind review

ABSTRACT

Audio-driven human animation reaches high fidelity, but its many-step guided sampling is far too slow for an interactive avatar. We present Edge-Avatar, a speech-to-video model for controllable audio-driven human animation that runs in real time at 25 FPS on a single consumer GPU under pose control. Starting from a bidirectional 2B diffusion transformer that draws text and audio conditioning from a single Qwen2.5-Omni model, two training stages convert it into a chunk-autoregressive generator, built on recent streaming works and extended with an area-adaptive face loss and memory noising. Pose control is trained rather than imposed at inference, through a pose track that may begin or end partway through a chunk, so a gesture is entered and left smoothly. An optional training-free reset returns the avatar to the reference image mid-stream and clears the carried memory. Deployment pairs the few-step sampler with FP8 linear layers, block-sparse attention, compiled execution, and a distilled decoder, so that denoising and decoding fit inside the playback interval of each chunk at SD resolution on a single NVIDIA RTX 5090. Two optional pieces are independent of each other, a cheaper opening denoising step carried in the frequency domain and a further-upscaling decoder, and together they deliver HD inside that same budget. On common speech-to-video benchmarks the result leads the open-source real-time avatar generation systems it is measured against, including ones an order of magnitude larger and ones that reach real time only on multi-GPU nodes, and a side-by-side study against the three closest of them comes out in its favour.

1 INTRODUCTION

Audio-driven human animation turns a reference image and a driving speech signal into video whose lip motion, expression, head pose, and body movement follow the audio. Recent systems reach convincing quality by moving the backbone to a diffusion transformer in the latent space of a pretrained video VAE and scaling it to billions of parameters (Chen et al., 2025; Kong et al., 2025; Yang et al., 2025; Gan et al., 2025; Gao et al., 2025; Meituan LongCat Team, 2026). For an avatar that has to answer while a listener waits, quality is no longer the binding constraint. Speed is. Video has to leave the model at least as fast as it plays, and a guided sampler that spends fifty steps and an unconditional pass per step costs a hundred network evaluations for five seconds of video, which falls short of that by more than an order of magnitude.

Audio-driven avatars. The systems above differ in how they inject speech and how they carry history across chunks. MultiTalk, InfiniteTalk, OmniAvatar, and Wan-S2V take features from Wav2Vec or Wav2Vec2, while HunyuanVideo-Avatar and LongCat-Video-Avatar 1.5 use Whisper. In every case the speech encoder is trained apart from the text encoder of the backbone, so text and audio come from separately trained models. On the memory side, Wan-S2V compresses history with FramePack (Zhang et al., 2025c) and LongCat-Video-Avatar 1.5 stitches latents across chunks.

Streaming and few-step generation. A separate line of work attacks latency rather than fidelity. Diffusion Forcing (Chen et al., 2024a) gives every token its own noise level so that a causal model can roll a sequence forward. CausVid (Yin et al., 2025) rewrites a bidirectional video diffusion transformer as a causal one and distills it to four steps, and MAGI-1 (Sand AI, 2025) carries that design to a chunk-wise foundation model. Self-Forcing (Huang et al., 2025) rolls the student out

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.

on its own samples during training, so that at inference it meets the kind of history it was trained on. Applied to avatars, the same line produces Ditto (Li et al., 2024), Live Avatar (Huang et al., 2026a), SoulX-FlashTalk (Shen et al., 2025), SoulX-FlashHead (Yu et al., 2026), SoulX-LiveAct (Zhen et al., 2026), Avatar Forcing (Ki et al., 2026), LiveTalk (Chern et al., 2025), and Avatar-Forever (Li et al., 2026b). StreamAvatar (Sun et al., 2026), JoyStreamer-Flash (Li et al., 2026a), and DynaForcing (Huang et al., 2026b) follow the same line but have not released inference code, so we describe them without comparing against them. Several of them, an order of magnitude larger than ours, reach real time only on multi-GPU nodes. Appendix A carries the extended related work, system by system.

This work. Edge-Avatar takes the self-forcing rollout and the distillation objective from that line and leaves the causal rewrite behind, keeping attention full inside the chunk as SoulX-FlashTalk also argues for. Fitting one consumer device is the constraint that shapes the rest of the design, and gesture control is treated the same way, trained for rather than imposed at inference. The inference code and the model weights are released, and the same repository holds example videos generated by the model.[1]

Contributions. Our contributions are the following.

- A training recipe that turns a speech-to-video model of our own, built on an open-source video model, into a real-time one on a single consumer GPU, extending recent streaming works with an area-adaptive face loss, memory noising, and pose conditioning.
- Gesture control trained on partial pose spans, so that a gesture switching on or off inside a chunk is an input the model has seen and the transition is smooth.
- A deployment that fits a single NVIDIA RTX 5090, combining the 3-step sampler with FP8 quantization of the block projections, compiled execution, block-sparse attention, and a distilled decoder, together with an optional training-free reset of the avatar mid-stream.
- A model that outperforms open-source real-time avatar systems on common benchmarks and in a side-by-side human study, including ones an order of magnitude larger and ones that reach real time only on multi-GPU nodes.

2 Training Data

Audio-driven animation demands that the speech be attributable to one visible face for the whole clip, so active-speaker detection rather than aesthetic filtering is the decisive stage of the pipeline. The corpus holds 1.17M single-speaker clips, split almost evenly between a web collection assembled by the authors from publicly accessible videos on open datasets and large video platforms and the Panda-70M portion of TalkVerse (Wang et al., 2025; Chen et al., 2024b), each retained only when a single face track covers it without interruption. Every clip carries six captions from Qwen3-VL (Bai et al., 2025), at three granularities in English and in Russian, generated natively rather than translated. Appendix B gives the extraction thresholds, describes the captioning prompt, and gives the language distribution, and Appendix B.6 the provenance of the two sources.

Pose tracks. Speech constrains the mouth and, through it, the rhythm of the head. It says nothing about what the hands do, so a clip that is to teach gesture control needs a second track beside its audio. The track is a skeleton rendered on a black background in the geometry of the source frame, with the body, the two hands and the face drawn as separate keypoint groups. Keypoints are extracted with ViTPose-H (Xu et al., 2022) on boxes from YOLOv10 (Wang et al., 2024a), in the preprocessing configuration released with Wan-Animate (Cheng et al., 2025). Before rendering, every skeleton passes through a bone-group augmentation we introduce here: each of its seven bone groups is rescaled by its own factor drawn once per clip, so joint angles survive and only proportions change. That is what lets a driving pose and a reference image disagree about build at inference, since the model has seen the same gesture carried by bodies of different builds. Every training stage mixes items with and without pose about half and half. Appendix B.4 gives the augmentation in full.

[1] https://anonymous.4open.science/r/s2v-2b-realtime-8D00

Silent clips. A talking avatar spends much of its time not talking, and the pipeline above selects against exactly that, since every stage keeps the windows in which someone is speaking. The corpus adds a partition for it, about 39K clips alongside its 1.17M speaking clips, each 125 frames long, in which a visible person is not speaking. They are mined with the same active-speaker detector read in reverse and filtered for quality, singing and edits, as Appendix B.5 describes. Their audio column points at one shared constant waveform rather than at the track of the clip, in two variants, digital silence and low-level white noise. Nothing else marks them as special. They pass through the same captioning stage as the rest, so a silent clip arrives with a caption that already reports closed lips and a still face.

3 BASE MODEL

Our base model is a 3D diffusion transformer operating in the latent space of a frozen Hunyuan video VAE (Kong et al., 2024) and trained with a flow-matching objective (Lipman et al., 2023; Liu et al., 2022; Esser et al., 2024). One forward pass maps a reference image, a speech waveform, a text prompt, and an optional visual memory to a chunk of 125 frames at 25 FPS. It is built on the Kandinsky 5.0 Video foundation models (Arkhipkin et al., 2025) and holds 2B parameters. Figure 4 in Appendix C shows the architecture. This section states only what the real-time variant of Section 4 builds on, and Appendices C and D hold the rest, including the two runs it is trained in, described in Appendix D.3.

Conditioning. Most speech-to-video systems pair a text-conditioned backbone with a standalone speech encoder, so the two conditioning signals come from separately trained models. We take both from one Qwen2.5-Omni checkpoint (Xu et al., 2025), the prompt from its Thinker and the speech from an audio tower trained jointly with it. Trainable transformer stacks inside the DiT adapt each stream before separate paths consume them, text through cross-attention in every visual block and audio through cross-attention at a subset of depths. A pooled CLIP embedding (Radford et al., 2021) is fused with the diffusion time embedding as a global style vector.

Visual sequence and memory. The packed visual sequence holds a memory group, the reference latent, and the noisy target, each tagged by a learned segment embedding. Memory follows a three-scale FramePack layout (Zhang et al., 2025c), with the two most recent latent frames at the target patch size, the preceding four at $(2, 4, 4)$, and the oldest twenty-four at $(4, 8, 8)$, which reaches thirty latent frames back at a token cost far below thirty dense frames. The reference is deliberately kept out of that history. It carries its own segment embedding and sits at a fixed temporal position outside the target range, which is what stops the model from reusing the reference pose at every chunk boundary.

Pose conditioning. A pose clip is encoded by the same VAE onto the same latent grid, and a separate linear projection adds the pose latent to the projected target tokens before the first block. That projection is initialized to zero, the memory and the reference are untouched, and the loss carries no pose term. Absence of pose is an all-zero latent rather than a missing input, and a track is dropped with probability 0.5. The track is not always fed over the whole chunk. With probability 0.2 each it is restricted to a prefix or to a suffix covering at least a fifth of the chunk, with the remaining latents zeroed. That is what makes a gesture beginning partway through a chunk an in-distribution input, and Section 4 lifts the same idea to the rollout.

Area-adaptive face loss. Lip motion lives in a small part of the frame, so we add a face term, the squared flow-matching residual averaged over a face mask M aligned to the latent grid, and weighted by λ_{face},

$$\mathcal{L}_{\text{face}} = \frac{\sum \Delta^2 \odot M}{\sum M}, \quad \lambda_{\text{face}} = \text{clip}\left(\frac{r_{\text{ref}}}{r_{\text{face}}}, \lambda_{\min}, \lambda_{\max}\right), \quad r_{\text{face}} = \frac{\sum M}{|M|},$$

with Δ the per-element difference between the predicted velocity and its target, r_{face} the share of the frame the face covers, $r_{\text{ref}} = 0.04$ the share the weight is calibrated at, and bounds $\lambda_{\min} = 0.15$ and $\lambda_{\max} = 2.5$. A face term on a mask is standard in this literature. What is new here is the weight, which moves against the area the mask covers: it keeps half-body speakers from being outvoted by

close framing and removes the second way the model has of lowering the term, framing the speaker tighter. Both base runs and the first autoregressive stage use it, the second drops it. Appendix D.2 gives the rest, and Appendix F.1 compares three stage-1 runs that differ in that term alone.

Sampling. Long video is produced chunk by chunk, each chunk conditioned on the three memory scales filled from the generated history, with a guided sampler whose cost the next section removes. Appendix D.9 gives the layout of a long rollout.

4 The Real-Time Model

The sampler of Section 3 spends 50 Euler steps and one unconditional pass per step on every chunk, which is 100 network evaluations for 5 seconds of video. That cost is acceptable offline and far too high for a speaker that has to answer while a listener waits. This section describes a variant of that model which produces video faster than the video plays.

The change is not a new architecture, no module is added to the DiT, and three things differ. The three-scale FramePack memory is cut to a rolling state of 2 latent frames, the chunk is shortened from 32 to 8 latent frames, and the guided 50-step sampler is replaced by a distilled sampler of 3 steps with no guidance branch. A chunk of 8 latent frames decodes to 32 video frames, which is 1.28 seconds at 25 FPS and sets the budget that one chunk of generation has to fit. Both stages train and serve at the three resolutions of the base model, 512×512, 768×512 and 512×768, which we call SD throughout this section, against the upscaled output of Section 4.4.1, which we call HD. Figure 1 shows the resulting model.

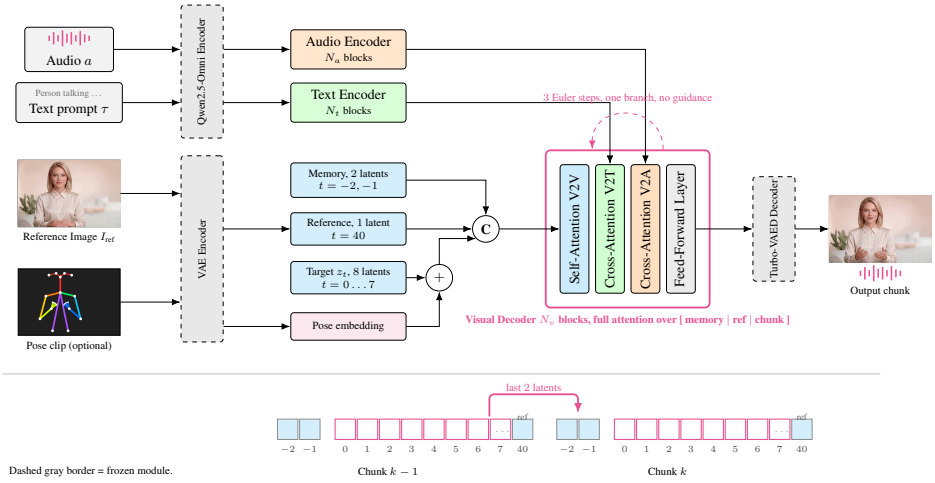


Figure 1: The real-time variant. Conditioning, the adapters and the block structure are those of Figure 4. Three things differ: memory is a rolling state of 2 latent frames, the sampler is 3 Euler steps on a single branch, and the only state crossing a chunk boundary is the pair of memory latents of the lower panel.

4.1 Autoregressive Model

4.1.1 Rolling Memory and Temporal Positions

The model of Section 3 carries 30 latent frames of history in three scales. The real-time variant carries 2, at the target patch size, and drops the other two scales, which shortens the packed sequence and so the cost of denoising it. No parameter count changes, since the coarse projections still load from the checkpoint and are never packed into the visual sequence, which becomes memory, reference and target in that order. Identity rides on the reference latent, which the cut does not touch, and the two latent frames carry the motion across a chunk boundary.

The autoregressive behavior comes entirely from the temporal positions. The target chunk always occupies positions 0 to $D - 1$, where $D = 8$ is the chunk length in latent frames. The two memory

latents always sit at -2 and -1, immediately before it. The reference stays at position 40, outside both ranges. This layout is rebuilt from scratch for every chunk rather than extended, so position indices never grow and the number of chunks in a rollout is not bounded by the positional encoding. The opening chunk of a rollout decodes to 29 frames rather than 32, since it carries no preceding latent.

4.1.2 Attention and carried state

Attention is not causal and no key-value cache is kept. Every call runs a full mask over the packed sequence, which the deployed kernel then evaluates sparsely, so the target tokens attend to the memory, to the reference and to each other without restriction, and the only state that survives a chunk boundary is the pair of memory latents. A cached causal decoder, the route CausVid (Yin et al., 2025) and MAGI-1 (Sand AI, 2025) take, would cost less per chunk, but recomputing eleven latent frames, two of memory, one of reference and eight of target, is cheap enough that the budget absorbs it. We borrow the self-forcing rollout and the distillation objective from that line, not the causal architecture.

4.1.3 Pose and silent clips across the two stages

The pose mechanism of Section 3 and the silent partition of Appendix B.5 carry into both autoregressive stages unchanged in form. The partial-span pose training of Section 3 is what makes a gesture usable in a live rollout. A request to gesture arrives between chunks, not on a chunk boundary of the speech, so the pose track is fed on the last few latents of the chunk in which it arrives and in full on the chunks after it. Stage 2 raises this from a single chunk to the rollout, as Section 4.2 describes. Appendix F.2 measures the motion at the frame where the gesture enters (Figure 8), and finds the excess motion around a boundary-aligned entry halved and its bend cut to a quarter.

4.2 Training the Autoregressive Model

Two stages separate the two changes. The first teaches the model to work from a short chunk and a two-latent memory while keeping its sampler, the second replaces the sampler. The shape is not new: SoulX-FlashTalk (Shen et al., 2025) moves to a short chunk and a small carried state before distilling, and CausVid (Yin et al., 2025) and Self-Forcing (Huang et al., 2025) supply the objective and the rollout it runs over. We distil to 3 steps rather than the 4 of that work, and apply the schedule to a model that carries a face term, a noised memory, pose conditioning and a silent partition through it. Both stages train on the background-stable subset of the corpus described in Appendix B.7.

4.2.1 Stage 1: short-chunk teacher forcing

Stage 1 applies the rectified-flow objective of Appendix D to the shortened layout. It is not a distillation and changes no module, only how a batch is assembled.

Each step draws the target chunk from inside a training clip and fills the memory with the true latents that precede it. Writing ζ for the clip latents and $L_m = 2$ for the memory length, the target is $\zeta[1{+}L_m : 1{+}L_m{+}D]$ and the memory is $\zeta[1 : 1{+}L_m]$, so the model always reads a correct history, which is what teacher forcing means here. With probability 0.5 the memory is zeroed instead and the reference is drawn as in the base model. That is the cold-start case and the first chunk of every rollout, so training it explicitly keeps the opening chunk of a reply from being an extrapolation.

The loss is the flow-matching term plus the area-adaptive face loss, both at the weights of the base run.

4.2.2 Stage 2: distribution matching with multi-chunk self-forcing

Stage 2 distills the stage-1 model into a few-step generator. It follows distribution matching distillation (Yin et al., 2024b;a) and takes its rollout from self-forcing (Huang et al., 2025). Three copies of the DiT are involved, all initialized from the stage-1 exponential moving average: the trained student f_θ, the frozen teacher f_ϕ, and the critic f_ψ, which tracks the distribution the student currently produces. Figure 5 in Appendix D.10 shows one training step.

The student samples on a fixed schedule of 3 transitions, drawn from a 50-point grid at indices $0, 24, 37$ and a final step to zero noise, with the timestep transform of Appendix D at scale 5. After the transform the levels are 1.0, 0.839 and 0.617. A chunk therefore costs 3 network evaluations rather than the 100 of the base sampler, and inference has to run this same schedule.

A step rolls the student forward over K chunks, with K drawn uniformly from 1 to 4 and shared across ranks. The first $K - 1$ chunks run without gradients and each one hands its own last two latent frames to the next chunk as memory. The first chunk starts from zero memory. Only the last chunk carries a gradient, and only through one of its 3 denoising steps, chosen at random.

Memory noise. The memory passed between chunks is not handed over clean. For each sequence a level $s \sim \mathcal{U}(0, 0.1)$ is drawn, and the two carried latents m become

$$m' = (1 - s)\, m + s\, \epsilon, \qquad \epsilon \sim \mathcal{N}(0, I),$$

with the memory tokens conditioned on s through their own time embedding, so the model is told how corrupted its history is rather than left to infer it, and a student that reads a history it cannot fully trust stops copying small errors forward as fact. The teacher scores the same memory taken before the noise is mixed in, since a frozen model that never saw corruption would judge inputs outside its own training distribution, and that memory is the rollout's own history rather than the ground-truth one. Appendix F.3 compares the base run against one trained and served without this noise.

Objective. Let $\hat{x}_0$ be the clean latent the student predicts for the last chunk, and let $p_{\theta,\tau}$ and $p_{\phi,\tau}$ be the distributions of the student sample and of the teacher, which stands in for real data, once both are renoised to a level τ drawn uniformly on $[0, 1]$, pushed through the timestep transform at scale 5 and clipped to $[0.02, 0.98]$. The objective is the reverse Kullback-Leibler divergence between them,

$$\mathcal{L}_{\text{DMD}} = \mathbb{E}_\tau \big[\, D_{\text{KL}}\big(p_{\theta,\tau} \,\|\, p_{\phi,\tau}\big)\big].$$

The divergence is not evaluated directly. The gradient reaching the student is the difference between the teacher and critic predictions $\hat{x}_0^{\text{real}}$ and $\hat{x}_0^{\text{fake}}$ from the renoised student sample, normalized by the mean absolute distance between the student prediction and the teacher one so that the step size does not follow the noise level. The teacher is evaluated with classifier-free guidance at weight 3 against a fixed negative prompt and zeroed acoustic features, which is how a student with no guidance branch at inference still inherits the effect of guidance.

The critic is a second copy of the model, trained on the student's own samples with the flow-matching objective of Appendix D, which is what keeps $\hat{x}_0^{\text{fake}}$ an estimate of the score of the distribution the student currently produces. It updates on every step and the student on every fifth, so it stays ahead of what it has to score.

Two terms are added to the student objective. Writing x_0^{gt} for the ground-truth latent of the same chunk, $u_0, \ldots, u_{D+1}$ for that chunk with the reference latent spliced onto both ends as an anchor, and M_{bg} for a background mask obtained from a frozen portrait-matting network run on the decoded chunk,

$$\mathcal{L}_\theta = \mathcal{L}_{\text{DMD}} + \lambda_{\text{reg}}\mathcal{L}_{\text{reg}} + \lambda_{\text{bg}}\mathcal{L}_{\text{bg}}, \quad \mathcal{L}_{\text{reg}} = \|\hat{x}_0 - x_0^{\text{gt}}\|_2^2, \quad \mathcal{L}_{\text{bg}} = \frac{\sum_k \|u_{k+1} - u_k\|^2 \odot M_{\text{bg},k}}{\sum_k M_{\text{bg},k}},$$

with $\lambda_{\text{reg}} = 0.1$ and $\lambda_{\text{bg}} = 2$. Both keep a long rollout from degrading, which distribution matching alone does not do. The regression to the ground-truth latents follows SoulX-FlashHead (Yu et al., 2026), which adds the same penalty to its distillation objective, while the masked background term follows the public training code released for that line, cited in Appendix D.10. It covers a blind spot of the objective, since a drifted background stays plausible to it, and the anchors at both ends of the chunk are what make a net displacement visible.

Pose intervals. Section 3 places a partial pose span inside a single chunk. Stage 2 lifts the same idea to the rollout. With probability 0.25 a posed item runs with pose on and then stops partway through one chunk, with probability 0.25 it starts with no pose and switches on partway through one chunk, and otherwise pose stays on throughout. The chunk where the transition happens is the one the gradient lands on, so the switch is always inside the supervised chunk, and the partial chunk keeps at least a fifth of its latents and never the whole chunk. Student and teacher see the same pose conditioning. This is what makes a gesture that arrives between chunks an in-distribution input at inference rather than a jump.

4.2.3 Optimization and distributed setup

Appendix D collects the settings of both stages, except for one that does not fit a table. Neither stage uses tensor parallelism, and the stage-2 trainer does not support it, because the three model copies and the rollout already fill the device budget.

4.3 Streaming Inference

A rollout is a sequence of chunks. The first sees the reference, its slice of the speech track and the prompt, with the memory latents set to zero, which is the cold start stage 1 trained. Each later chunk reuses the same reference and takes its memory from the last two latent frames of the chunk before it, read before they are decoded, so the memory never passes back through the encoder and peak memory does not grow with output length. That memory is corrupted at the fixed level $s = 0.05$, the mean of the level drawn in training.

Sampling runs the schedule of Section 4.2 as a deterministic flow-ODE Euler solver with timestep shift 5, with no unconditional pass and no negative prompt, so a step costs one network evaluation rather than two. The frame is generated whole at one of the three SD resolutions, and every number reported here comes from that path, with no cropping and no rescaling. Decoding uses a distilled Turbo-VAED decoder (Zou et al., 2025) in place of the Hunyuan one, which is dropped at load time, while the Hunyuan encoder is kept, so the latent space is unchanged. Each chunk is decoded with two latent frames of context prepended and the five leading frames discarded, which removes the seam a chunk boundary would otherwise leave.

Resetting the avatar. A stream that runs for minutes carries its own artifacts forward through the memory. We propose a mechanism that refreshes that memory on the fly, without any fine-tuning. On one chunk the reference moves from RoPE position 40 to $D - 1$, the last latent of the chunk, so the frame generated there is time-aligned with the reference and lands on it. The next chunk starts from zero memory, with the reference back at 40. The chunk before the reset ends on the reference, so the junction carries no jump in pose, and the reset fires on a silent chunk, so no speech is cut. Every run of Section 4.5 has it switched off, so the reported numbers come from the model alone. Neither the two-minute nor the five-minute rollout of Tables 12 and 13 drifts without it, so we offer the reset as a safeguard for sessions longer than those we evaluate.

4.4 Single-GPU Deployment

The deployed system is one process on a single NVIDIA RTX 5090. The denoising loop and the VAE decoder run in two threads that take turns on the device through a lock giving the decoder right of way, so a finished chunk reaches the viewer while the next one waits.

4.4.1 Quantization, compilation, attention and resolution

Table 1 lists what the deployment changes relative to a plain PyTorch run of the same weights, for the two stages that run per chunk. The conditioning encoder runs once per utterance rather than once per chunk, so it is outside this loop and outside the measurements of Section 4.4.2. Of Qwen2.5-Omni only the Thinker is loaded, its vision tower dropped and its own projections held in FP8, while the audio tower inside it stays in bfloat16, so the speech features are those of the unquantized model. It stays on the GPU, counted in the peak memory of Table 2.

FP8 covers the attention and feed-forward projections inside the visual, text and audio blocks and nothing outside them, since the embedders, the modulation and the output layer are either too small for the narrower multiply to pay off or close enough to the output that its error would show. It requires the compiled path, since in eager mode the cast of activations costs more than the narrower multiply saves, and it does not accept parameters sharded as distributed tensors, which rather than any throughput argument is why the real-time path runs on a single device.

Appendix F.4 gives what the quantizer, the sparse attention and the distilled decoder cost in quality.

One further technique is optional and independent of the resolution the system delivers. Spectral progressive diffusion (SPD) (Xiao et al., 2026) starts from the observation that a diffusion trajectory fixes low frequencies early and high frequencies late, so the first steps do not need the resolution

the last ones do. The opening step runs on a latent grid half the target in each dimension, a quarter of the tokens, and is carried to the target grid in the frequency domain, as Appendix F.5 describes together with what the cheaper step costs in quality. The remaining two steps run on the full grid. It takes about a quarter off the denoiser at either resolution, which Table 2 reports with it on and off.

Table 1: Deployment techniques. The opening step and the last row are off by default and independent of each other, and the last row is what the HD mode adds.

Component	Technique
DiT linear layers	FP8 row-wise quantization (Micikevicius et al., 2022) of all 396 projections
Visual self-attention	SpargeAttention (Zhang et al., 2025b) on SageAttention kernels (Zhang et al., 2025a)
DiT graph	`torch.compile` with static shapes (Ansel et al., 2024)
VAE decoder	Distilled Turbo-VAED (Zou et al., 2025) in place of the Hunyuan decoder
Opening step	Spectral progressive diffusion (Xiao et al., 2026) on a half-size latent grid, optional
Decoder	Turbo-VAED-x1.5, spatial upscale 12 instead of 8

The HD mode swaps in Turbo-VAED-x1.5, the distilled decoder rebuilt with a spatial upscale of 12 rather than 8, so 512×768 is delivered as 768×1152 from the latents the model was trained to produce. The transformer keeps denoising the grid it was trained on, which leaves the whole cost with the decoder, the part of the loop holding the smaller share of the budget. Appendix F.6 says how it is built and compares it against upsampling the output of the ordinary decoder. Pairing it with the cheaper opening step returns the time it adds, which is how Table 2 separates what each costs.

4.4.2 Latency budget

Denoising and decoding are serialized on the device, so together they have to fit inside the 1.28 seconds a chunk plays for. Table 2 reports what they cost over one rollout of 30.6 seconds of video in 24 chunks, on a single NVIDIA RTX 5090 with the deployed stack above.

Table 2: Generation cost, in milliseconds of compute per second of video. RT is playback over compute, and Chunk is the compute one chunk of 1.28 seconds takes.

Decoder	SPD	DiT	VAE	Total	RT	Chunk	Memory
Turbo-VAED	off	596	168	764	1.31×	0.98 s	18.4 GB
Turbo-VAED	on	442	168	610	1.64×	0.78 s	18.4 GB
Turbo-VAED-x1.5	off	598	240	838	1.19×	1.07 s	21.4 GB
Turbo-VAED-x1.5	on	442	242	684	1.46×	0.88 s	21.4 GB

Every configuration generates faster than it plays. Spectral progressive diffusion takes a quarter off the denoiser, the x1.5 decoder adds about 70 milliseconds to decoding, and HD with both keeps the margin of the default path, with peak memory inside the 32 GB of the card. The numbers hold from the opening chunk onward, since compilation and graph capture are paid by a warmup turn rendered off the request clock. They also cover the denoiser and the decoder alone, while a conversational avatar runs speech recognition, a language model, speech synthesis, compositing and transport in the same interval, so the margin above real time is the room that stack works in rather than a spare.

4.5 Evaluation of the Real-Time Model

We compare against seven open-source streaming avatar systems, from 1.3B to 22B, and against Ditto (Li et al., 2024), which is not a video diffusion model but a motion-space diffusion with a warping decoder that animates the cropped face and composites it back. Edge-Avatar is evaluated in the default deployed configuration of Section 4.4, at SD resolution with the Turbo-VAED decoder and with spectral progressive diffusion and the reset off, on one RTX 5090. Table 3 carries the size of each system and the hardware its authors report real time on, which the metrics alone do not show.

Four benchmarks are used, three with ground truth, HDTF (Zhang et al., 2021), EMTD (Meng et al., 2025) and CelebV-HQ (Zhu et al., 2022), and one cross-driven stress test in which an EMTD reference image is driven by the speech of a different person for two minutes. Two of the systems cannot be run past that length, so the aggregates below cover the two-minute version and the five-minute one is reported separately in Table 13. Five metrics are reported: identity as the cosine similarity between

ArcFace embeddings (CSIM), lip synchronization as the confidence and the embedding distance of SyncNet (Sync-C and Sync-D), and per-frame technical quality and aesthetics from Q-Align (IQA and ASE). Published numbers are not comparable across papers, so every model was re-run through one pipeline on one frozen set of 50 clips per benchmark. Appendix E has the systems, the protocol, the metrics, the aggregates and the per-benchmark tables.

Table 3: All four benchmarks. The metric columns are averaged over them, the two on the right are pooled over the benchmark and metric cells. GPUs is what each system needs to reach real time, as its authors report it, and the best value in a column is in bold.

Model	Size	GPUs	CSIM ↑	Sync-C ↑	Sync-D ↓	IQA ↑	ASE ↑	Min-max ↑	Mean rank ↓
Edge-Avatar (ours)	2B	1× RTX 5090	**0.865**	**7.68**	7.99	**4.60**	**3.55**	**0.978**	**1.60**
SoulX-FlashTalk	14B	8× H800	0.857	7.47	8.37	4.50	3.43	0.865	3.10
SoulX-LiveAct	18B	2× H100	0.756	7.37	**7.94**	4.44	3.37	0.746	4.20
Avatar-Forever	22B	1× H100	0.782	5.36	9.99	**4.60**	3.47	0.620	4.60
Live Avatar	14B	5× H100	0.707	6.39	8.74	4.54	3.43	0.642	4.90
SoulX-FlashHead Pro	1.3B	2× RTX 5090	0.854	5.81	9.20	4.44	3.35	0.676	4.95
SoulX-FlashHead Lite	1.3B	1× RTX 4090	0.767	6.01	8.73	3.95	2.89	0.350	6.50
LiveTalk	1.3B	1× RTX 4090	0.677	3.95	10.74	4.49	3.37	0.319	7.20
Ditto	≈0.2B	1× A100	0.955	4.99	9.63	4.29	3.16	0.384	7.69

Edge-Avatar leads on both aggregates, on identity, on lip-sync confidence, where it is first in every benchmark taken separately, and on aesthetics. The one column that goes elsewhere is Sync-D, where SoulX-LiveAct is ahead by 0.05, and on IQA the 22B Avatar-Forever ties with us at 4.60. Ditto's identity score is set in grey and left out of both aggregates, because it warps the reference face rather than generating it, which also puts it near the bottom on everything measuring how the face moves. Two baselines run on a single consumer card as we do, SoulX-FlashHead Lite and LiveTalk, and together with Ditto they hold the three lowest places on both aggregates.

Figure 2 adds a side-by-side study against SoulX-FlashTalk, with ten annotators judging ten unlabelled EMTD pairs each, 100 judgements per category. Most pairs are a tie, and where annotators do separate the two they separate them in one direction, with the clearest margin on lip synchronization, 32% against 2%, and on motion naturalness, 28% against 10%, and with no category where the baseline comes out ahead. Appendix E.6 states what each category asks for and carries the same study against SoulX-LiveAct and Avatar-Forever.

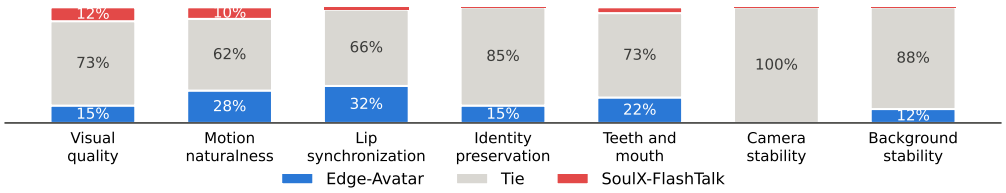


Figure 2: Side-by-side preferences against SoulX-FlashTalk.

5 Conclusion

We presented Edge-Avatar, a route from a bidirectional speech-to-video model to one optimized to run in real time on a single NVIDIA RTX 5090, with gesture control through a pose track. Two stages do the work, a short-chunk one on a rolling memory and a distillation over a self-forcing rollout, extended with an area-adaptive face loss, memory noising and pose conditioning. It leads the open-source streaming systems it is measured against, on benchmarks and in a side-by-side study, including ones an order of magnitude larger and ones that need multi-GPU nodes.

Ethics Statement

The training corpus consists of publicly posted video of real people, drawn from open datasets and large video platforms and from the public TalkVerse and Panda-70M research datasets under their non-commercial research terms, as is the case for the datasets on which audio-driven animation is commonly trained and evaluated. No participants were recruited, and the corpus itself is not distributed. Appendix B.6 states the provenance of both sources. Evaluation uses public benchmarks only. A model that animates a single photograph from speech can be misused to impersonate a person. The release is intended for research, and we oppose the generation of real people without their permission.

AI Use Statement

We used a large language model to draft and edit prose, to search related work, and to draw the schematic figures. It was not used to generate data, to develop the method, to design experiments, or to implement the model and the training code. We verified the numbers and the citations ourselves, and we take responsibility for the final content of this work.

Reproducibility Statement

The inference code and the model weights are released in one anonymous repository,[2] so the deployed system of Section 4.4 can be run as described rather than reimplemented from the text. It runs in PyTorch, with no TensorRT, no exported graph format, no hand-written kernel and no tiling of the decoder, so the reported timings do not depend on a separate inference stack. Training is not released as code, and the paper carries what is needed to reimplement it instead. Section 4 and Appendix D state the objective, the sampling schedule, the loss weights, the auxiliary terms and their constants, the optimizer settings, the step counts, and the distributed layout of every training run, together with the checkpoint each run starts from. Section 2 and Appendix B describe how the corpus, the pose tracks, and the silent partition were built, including the keypoint extractor and the captioning procedure. Section 4.4 lists the deployment settings that the reported behavior depends on, and Appendix F.7 says which claims a reader can run directly from the released code.

References

Jason Ansel, Edward Yang, Horace He, Natalia Gimelshein, Animesh Jain, Michael Voznesensky, Bin Bao, Peter Bell, David Berard, Evgeni Burovski, Geeta Chauhan, Anjali Chourdia, et al. Pytorch 2: Faster machine learning through dynamic python bytecode transformation and graph compilation. In *Proceedings of the 29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS)*, pp. 929–947, 2024. doi: 10.1145/3620665.3640366.

Vladimir Arkhipkin, Vladimir Korviakov, Nikolai Gerasimenko, Denis Parkhomenko, Viacheslav Vasilev, Alexey Letunovskiy, Nikolai Vaulin, Maria Kovaleva, Ivan Kirillov, Lev Novitskiy, Denis Koposov, Nikita Kiselev, Alexander Varlamov, Dmitrii Mikhailov, Vladimir Polovnikov, Andrey Shutkin, Julia Agafonova, Ilya Vasiliev, Anastasiia Kargapoltseva, Anna Dmitrienko, Anastasia Maltseva, Anna Averchenkova, Olga Kim, Tatiana Nikulina, and Denis Dimitrov. Kandinsky 5.0: A family of foundation models for image and video generation, 2025. URL https://arxiv.org/abs/2511.14993.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, et al. Qwen3-vl technical report, 2025. URL https://arxiv.org/abs/2511.21631.

Boyuan Chen, Diego Marti Monso, Yilun Du, Max Simchowitz, Russ Tedrake, and Vincent Sitzmann. Diffusion forcing: Next-token prediction meets full-sequence diffusion. *Advances in Neural Information Processing Systems (NeurIPS)*, 2024a. URL https://arxiv.org/abs/2407.01392.

Tsai-Shien Chen, Aliaksandr Siarohin, Willi Menapace, Ekaterina Deyneka, Hsiang-wei Chao, Byung Eun Jeon, Yuwei Fang, Hsin-Ying Lee, Jian Ren, Ming-Hsuan Yang, and Sergey Tulyakov.

[2] https://anonymous.4open.science/r/s2v-2b-realtime-8D00

Panda-70M: Captioning 70M Videos with Multiple Cross-Modality Teachers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13320–13331, 2024b. doi: 10.1109/CVPR52733.2024.01265.

Yi Chen, Sen Liang, Zixiang Zhou, Ziyao Huang, Yifeng Ma, Junshu Tang, Qin Lin, Yuan Zhou, and Qinglin Lu. Hunyuanvideo-avatar: High-fidelity audio-driven human animation for multiple characters, 2025. URL https://arxiv.org/abs/2505.20156.

Gang Cheng, Xin Gao, Li Hu, Siqi Hu, Mingyang Huang, Chaonan Ji, Ju Li, Dechao Meng, Jinwei Qi, Penchong Qiao, Zhen Shen, Yafei Song, et al. Wan-animate: Unified character animation and replacement with holistic replication, 2025. URL https://arxiv.org/abs/2509.14055.

Ethan Chern, Zhulin Hu, Bohao Tang, Jiadi Su, Steffi Chern, Zhijie Deng, and Pengfei Liu. Livetalk: Real-time multimodal interactive video diffusion via improved on-policy distillation, 2025. URL https://arxiv.org/abs/2512.23576.

Joon Son Chung and Andrew Zisserman. Out of Time: Automated Lip Sync in the Wild. In *Workshop on Multi-view Lip-reading, Asian Conference on Computer Vision*, 2016. URL https://www.robots.ox.ac.uk/~vgg/publications/2016/Chung16a/.

Tri Dao. FlashAttention-2: Faster attention with better parallelism and work partitioning. In *International Conference on Learning Representations (ICLR)*, 2024. URL https://arxiv.org/abs/2307.08691.

Tri Dao, Daniel Y. Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. FlashAttention: Fast and memory-efficient exact attention with IO-awareness. In *Advances in Neural Information Processing Systems (NeurIPS)*, 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 (CVPR)*, pp. 4685–4694, 2019. doi: 10.1109/CVPR.2019.00482. URL https://openaccess.thecvf.com/content_CVPR_2019/html/Deng_ArcFace_Additive_Angular_Margin_Loss_for_Deep_Face_Recognition_CVPR_2019_paper.html.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, Dustin Podell, Tim Dockhorn, Zion English, Kyle Lacey, Alex Goodwin, Yannik Marek, and Robin Rombach. Scaling rectified flow transformers for high-resolution image synthesis, 2024. URL https://arxiv.org/abs/2403.03206.

PyTorch Foundation. Current and new activation checkpointing techniques in pytorch. https://pytorch.org/blog/activation-checkpointing-techniques/, 2025. URL https://pytorch.org/blog/activation-checkpointing-techniques/.

Qijun Gan, Ruizi Yang, Jianke Zhu, Shaofei Xue, and Steven Hoi. Omniavatar: Efficient audio-driven avatar video generation with adaptive body animation, 2025. URL https://arxiv.org/abs/2506.18866.

Xin Gao, Li Hu, Siqi Hu, Mingyang Huang, Chaonan Ji, Dechao Meng, Jinwei Qi, Penchong 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. Tongyi Lab, Alibaba.

Yuan Gong, Yu-An Chung, and James Glass. Ast: Audio spectrogram transformer, 2021. URL https://arxiv.org/abs/2104.01778.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance, 2022. URL https://arxiv.org/abs/2207.12598.

Xun Huang, Zhengqi Li, Guande He, Mingyuan Zhou, and Eli Shechtman. Self forcing: Bridging the train-test gap in autoregressive video diffusion, 2025. URL https://arxiv.org/abs/2506.08009.

Yubo Huang, Hailong Guo, Fangtai Wu, Weiqiang Wang, 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, 2026a. URL https://arxiv.org/abs/2512.04677.

Yubo Huang, Sirui Zhao, Xinchen Yao, Zhengye Zhang, Jinyang Huang, Fengqi Cui, Shiwei Wu, and Enhong Chen. Dynaforcing: Overcoming dynamic collapse in self-forcing distillation for streaming avatar generation, 2026b. URL https://arxiv.org/abs/2608.17707.

Azamat Kanametov. YOLO-Face: Face Detection Models Based on YOLO, 2024. URL https://github.com/akanametov/yolo-face.

Taekyung Ki, Sangwon Jang, Jaehyeong Jo, Jaehong Yoon, and Sung Ju Hwang. Avatar forcing: Real-time interactive head avatar generation for natural conversation, 2026. URL https://arxiv.org/abs/2601.00664.

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, Xinchi 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, 2024. URL https://arxiv.org/abs/2412.03603.

Zhe Kong, Feng Gao, Yong Zhang, Zhuoliang Kang, Xiaoming Wei, Xunliang Cai, Guanying Chen, and Wenhan Luo. Let them talk: Audio-driven multi-person conversational video generation, 2025. URL https://arxiv.org/abs/2505.22647.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient Memory Management for Large Language Model Serving with PagedAttention. In *Proceedings of the 29th Symposium on Operating Systems Principles (SOSP)*, 2023.

Chaochao Li, Ruikui Wang, Liangbo Zhou, Jinheng Feng, Huaishao Luo, Huan Zhang, Youzheng Wu, and Xiaodong He. Joystreamer-flash: Real-time and infinite audio-driven avatar generation with autoregressive diffusion, 2026a. URL https://arxiv.org/abs/2512.11423.

Ruibin Li, Tao Yang, Zhiyuan Ma, Fangzhou Ai, Shilei Wen, and Lei Zhang. Avatar-forever: Decoupled parallel training for high-quality real-time infinite avatars, 2026b. URL https://arxiv.org/abs/2608.12107.

Tianqi Li, Ruobing Zheng, Minghui Yang, Jingdong Chen, and Ming Yang. Ditto: Motion-space diffusion for controllable realtime talking head synthesis, 2024. URL https://arxiv.org/abs/2411.19509.

Junhua Liao, Haihan Duan, Kanghui Feng, Wanbing Zhao, Yanbing Yang, Liangyin Chen, and Yanru Chen. LR-ASD: Lightweight and Robust Network for Active Speaker Detection. *International Journal of Computer Vision*, 133:4749–4769, 2025. doi: 10.1007/s11263-025-02399-2.

Shanchuan Lin, Linjie Yang, Imran Saleemi, and Soumyadip Sengupta. Robust high-resolution video matting with temporal guidance. In *IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 238–247, 2022.

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*, 2023. URL https://openreview.net/forum?id=PqvMRDCJT9t.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow, 2022. URL https://arxiv.org/abs/2209.03003.

Meituan LongCat Team. Longcat-video technical report, 2025. URL https://arxiv.org/abs/2510.22200.

Meituan LongCat Team. Longcat-video-avatar 1.5 technical report, 2026. URL https://arxiv.org/abs/2605.26486.

Rang Meng, Xingyu Zhang, Yuming Li, and Chenguang Ma. EchoMimicV2: Towards Striking, Simplified, and Semi-Body Human Animation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5489–5498, 2025. URL https://openaccess.thecvf.com/content/CVPR2025/html/Meng_EchoMimicV2_Towards_Striking_Simplified_and_Semi-Body_Human_Animation_CVPR_2025_paper.html.

Paulius Micikevicius, Dusan Stosic, Neil Burgess, Marius Cornea, Pradeep Dubey, Richard Grisenthwaite, Sangwon Ha, Alexander Heinecke, Patrick Judd, John Kamalu, Naveen Mellempudi, Stuart Oberman, et al. Fp8 formats for deep learning, 2022. URL https://arxiv.org/abs/2209.05433.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4172–4182, 2023. doi: 10.1109/ICCV51070.2023.00387.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pp. 8748–8763. PMLR, 2021.

Sand AI. Magi-1: Autoregressive video generation at scale, 2025. URL https://arxiv.org/abs/2505.13211.

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, 2025. URL https://arxiv.org/abs/2512.23379.

Tomáš Souček and Jakub Lokoč. Transnet v2: An effective deep network architecture for fast shot transition detection, 2020. URL https://arxiv.org/abs/2008.04838.

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024. doi: 10.1016/j.neucom.2023.127063. URL https://arxiv.org/abs/2104.09864.

Zhiyao Sun, Ziqiao Peng, Yifeng Ma, Yi Chen, Zhengguang Zhou, Zixiang Zhou, Guozhen Zhang, Youliang Zhang, Yuan Zhou, Qinglin Lu, and Yong-Jin Liu. Streamavatar: Streaming diffusion models for real-time interactive human avatars, 2026. URL https://arxiv.org/abs/2512.22065.

Andros Tjandra, Yi-Chiao Wu, Baishan Guo, John Hoffman, Brian Ellis, Apoorv Vyas, Bowen Shi, Sanyuan Chen, Matt Le, Nick Zacharov, Carleigh Wood, Ann Lee, and Wei-Ning Hsu. Meta audiobox aesthetics: Unified automatic quality assessment for speech, music, and sound, 2025. URL https://arxiv.org/abs/2502.05139.

Ao Wang, Hui Chen, Lihao Liu, Kai Chen, Zijia Lin, Jungong Han, and Guiguang Ding. Yolov10: Real-time end-to-end object detection, 2024a. URL https://arxiv.org/abs/2405.14458.

Yi Wang, Kunchang Li, Xinhao Li, Jiashuo Yu, Yinan He, Guo Chen, Baoqi Pei, Rongkun Zheng, Zun Wang, Yansong Shi, et al. Internvideo2: Scaling foundation models for multimodal video understanding. In *European Conference on Computer Vision*, pp. 396–416. Springer, 2024b.

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.

Zhou Wang, Alan C. Bovik, Hamid R. Sheikh, and Eero P. Simoncelli. Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4): 600–612, 2004. doi: 10.1109/TIP.2003.819861.

Haoning Wu, Zicheng Zhang, Weixia Zhang, Chaofeng Chen, Liang Liao, Chunyi Li, Yixuan Gao, Annan Wang, Erli Zhang, Wenxiu Sun, Qiong Yan, Xiongkuo Min, Guangtao Zhai, and Weisi Lin. Q-Align: Teaching LMMs for Visual Scoring via Discrete Text-Defined Levels. *arXiv preprint arXiv:2312.17090*, 2023. doi: 10.48550/arXiv.2312.17090. URL https://arxiv.org/abs/2312.17090.

Howard Xiao, Brian Chao, Lior Yariv, and Gordon Wetzstein. Spectral progressive diffusion for efficient image and video generation, 2026. URL https://arxiv.org/abs/2605.18736.

Jin Xu, Zhifang Guo, Jinzheng He, Hangrui Hu, Ting He, Shuai Bai, Keqin Chen, Jialin Wang, Yang Fan, Kai Dang, Bin Zhang, Xiong Wang, Yunfei Chu, and Junyang Lin. Qwen2.5-omni technical report, 2025. URL https://arxiv.org/abs/2503.20215.

Yufei Xu, Jing Zhang, Qiming Zhang, and Dacheng Tao. Vitpose: Simple vision transformer baselines for human pose estimation, 2022. URL https://arxiv.org/abs/2204.12484.

Shaoshu Yang, Zhe Kong, Feng Gao, Meng Cheng, Xiangyu Liu, Yong Zhang, Zhuoliang Kang, Wenhan Luo, Xunliang Cai, Ran He, and Xiaoming Wei. Infinitetalk: Audio-driven video generation for sparse-frame video dubbing, 2025. URL https://arxiv.org/abs/2508.14033.

Tianwei Yin, Michaël Gharbi, Taesung Park, Richard Zhang, Eli Shechtman, Fredo Durand, and William T Freeman. Improved distribution matching distillation for fast image synthesis. In *NeurIPS*, 2024a.

Tianwei Yin, Michaël Gharbi, Richard Zhang, Eli Shechtman, Frédo 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)*, 2024b. URL https://arxiv.org/abs/2311.18828.

Tianwei Yin, Qiang Zhang, Richard Zhang, William T. Freeman, Frédo Durand, Eli Shechtman, and Xun Huang. From slow bidirectional to fast autoregressive video diffusion models. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 22963–22974, 2025. doi: 10.1109/CVPR52734.2025.02138.

Tan Yu, Qian Qiao, Le Shen, Ke Zhou, Jincheng Hu, Dian Sheng, Bo Hu, Haoming Qin, Jun Gao, Changhai Zhou, Shunshun Yin, and Siyuan Liu. Soulx-flashhead: Oracle-guided generation of infinite real-time streaming talking heads, 2026. URL https://arxiv.org/abs/2602.07449.

Jintao Zhang, Haofeng Huang, Pengle Zhang, Jia Wei, Jun Zhu, and Jianfei Chen. Sageattention2: Efficient attention with thorough outlier smoothing and per-thread int4 quantization. In *International Conference on Machine Learning (ICML)*, 2025a. URL https://arxiv.org/abs/2411.10958.

Jintao Zhang, Chendong Xiang, Haofeng Huang, Jia Wei, Haocheng Xi, Jun Zhu, and Jianfei Chen. Spargeattention: Accurate and training-free sparse attention accelerating any model inference, 2025b. URL https://arxiv.org/abs/2502.18137.

Lvmin Zhang, Shengqu Cai, Muyang Li, Gordon Wetzstein, and Maneesh Agrawala. Frame context packing and drift prevention in next-frame-prediction video diffusion models, 2025c. URL https://arxiv.org/abs/2504.12626.

Zhimeng Zhang, Lincheng Li, Yu Ding, and Changjie Fan. Flow-Guided One-Shot Talking Face Generation with a High-Resolution Audio-Visual Dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3660–3669, 2021. doi: 10.1109/CVPR46437.2021.00366. URL https://openaccess.thecvf.com/content/CVPR2021/html/Zhang_Flow-Guided_One-Shot_Talking_Face_Generation_With_a_High-Resolution_Audio-Visual_Dataset_CVPR_2021_paper.html.

Yanli Zhao, Andrew Gu, Rohan Varma, Liang Luo, Chien-Chin Huang, Min Xu, Less Wright, Hamid Shojanazeri, Myle Ott, Sam Shleifer, Alban Desmaison, Can Balioglu, Pritam Damania, Bernard Nguyen, Geeta Chauhan, Yuchen Hao, Ajit Mathews, and Shen Li. Pytorch fsdp: Experiences on scaling fully sharded data parallel, 2023. URL https://arxiv.org/abs/2304.11277.

Dingcheng Zhen, Xu Zheng, Ruixin Zhang, Zhiqi Jiang, Yichao Yan, Ming Tao, and Shunshun Yin. Soulx-liveact: Towards hour-scale real-time human animation with neighbor forcing and convkv memory, 2026. URL https://arxiv.org/abs/2603.11746.

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 *Computer Vision, ECCV 2022*, pp. 650–667. Springer Nature Switzerland, 2022. doi: 10.1007/978-3-031-20071-7_38. URL https://link.springer.com/chapter/10.1007/978-3-031-20071-7_38.

Ya Zou, Jingfeng Yao, Siyuan Yu, Shuai Zhang, Wenyu Liu, and Xinggang Wang. Turbo-vaed: Fast and stable transfer of video-vaes to mobile devices, 2025. URL https://arxiv.org/abs/2508.09136.

CONTENTS

1 Introduction **1**

2 Training Data **2**

3 Base Model **3**

4 The Real-Time Model **4**

4.1 Autoregressive Model 4

4.1.1 Rolling memory and temporal positions 4

4.1.2 Attention and carried state 5

4.1.3 Pose and silent clips across the two stages 5

4.2 Training the Autoregressive Model 5

4.2.1 Stage 1: short-chunk teacher forcing 5

4.2.2 Stage 2: distribution matching with multi-chunk self-forcing 5

4.2.3 Optimization and distributed setup 7

4.3 Streaming Inference 7

4.4 Single-GPU Deployment 7

4.4.1 Quantization, compilation, attention and resolution 7

4.4.2 Latency budget 8

4.5 Evaluation of the Real-Time Model 8

5 Conclusion **9**

A Detailed related work **18**

A.1 Audio-driven avatars 18

A.2 Streaming and few-step generation 18

B Data pipeline **20**

B.1 Data Sources 20

B.2 Active-Speaker Detection and Clip Extraction 20

B.3 Captioning 21

B.4 Pose track augmentation 21

B.5 Silent clips 22

B.6 Provenance and exclusion of evaluation data 22

B.7 Background stability 22

B.8 Resulting Corpus 23

C Base architecture **24**

C.1 Conditioning streams 24

C.2 Latent video tokenization 25
C.3 DiT backbone 25
C.4 Reference image and memory placement 26
C.5 Audio injection 26

D Base training and long-form sampling **27**
D.1 Flow-matching objective 27
D.2 Face loss 27
D.3 Training stages 27
D.4 Pretraining initialization 28
D.5 Multi-resolution training 28
D.6 Optimization 28
D.7 Distributed training 28
D.8 Silent speech 29
D.9 Long-Form Inference 29
D.10 Autoregressive training stages 29

E Evaluation of the real-time model **30**
E.1 Protocol 30
E.2 Metrics 31
E.3 The two aggregate columns 31
E.4 Baselines 31
E.5 Per-benchmark results 32
E.6 Side-by-side human study 33

F Ablations **34**
F.1 Area-adaptive face loss 34
F.2 Entering a gesture inside a chunk 35
F.3 Memory noise 36
F.4 Cost of the deployment optimizations 36
F.5 Spectral progressive diffusion 36
F.6 Distilled decoder against interpolation 37
F.7 What the released code lets a reader run 37

A Detailed related work

A.1 Audio-driven avatars

HunyuanVideo-Avatar (Chen et al., 2025). It is built on a multimodal diffusion transformer and targets dynamic single- and multi-character dialogue. It replaces addition-based character conditioning with a dedicated character image injection module. A separate emotion reference image provides expression cues, while face-aware audio adapters use latent face masks to route each speech stream to the corresponding character.

MultiTalk (Kong et al., 2025). It introduces audio-driven multi-person conversational video generation on the Wan2.1-I2V-14B backbone. Naive multi-stream audio injection can associate a voice with the wrong speaker. Label Rotary Position Embedding addresses this problem by assigning matched labels to audio embeddings and person regions in the video latents, causing each audio cross-attention map to activate the intended speaker. Partial-parameter and multi-task training retain the visual and textual capabilities of the base model across single- and multi-person settings.

OmniAvatar (Gan et al., 2025). It is built on the Wan2.1-T2V-14B foundation. It injects pixel-wise multi-hierarchical audio features directly into the latent video representation instead of relying only on audio cross-attention. LoRA adaptation preserves the text-to-video prior of the base model, while the audio conditioning extends motion control beyond the face to adaptive full-body animation.

Wan-S2V (Gao et al., 2025). It also builds on a Wan-14B backbone, but targets cinematic film and television scenes rather than isolated talking heads. It encodes speech with Wav2Vec, inserts Audio Blocks that compute frame-wise attention between visual and audio tokens, and applies FramePack (Zhang et al., 2025c) to compress older motion-frame tokens for long-form continuation.

InfiniteTalk (Yang et al., 2025). It builds on MeiGen-MultiTalk and extends it to sparse-frame video dubbing and image-to-video animation. Given a source video or a reference image and a new audio track, it animates full-body motion while preserving identity, background, and camera motion through sparse keyframe conditioning and temporal context frames. A streaming chunk generator and a reference-frame sampling strategy support autoregressive long-form generation without requiring dense source-video conditioning.

LongCat-Video-Avatar 1.5 (Meituan LongCat Team, 2025; 2026). It is built on the LongCat-Video foundation, a UMT5-conditioned DiT with Wav2Vec audio injection. It identifies rigid speech-body coupling during silent intervals and a reference-copying artifact that suppresses motion diversity. Disentangled Unconditional Guidance separates silent speech conditioning from the unconditional branch. Reference Skip Attention is applied at inference to reduce pose copying from the reference image. Cross-chunk latent stitching avoids repeated VAE decoding and encoding during autoregressive generation. A single model supports audio-text-to-video, audio-text-image-to-video, and audio-driven continuation. Version 1.5 replaces the Wav2Vec encoder with Whisper Large and distills the sampler to 8 function evaluations.

A.2 Streaming and few-step generation

A separate line of work attacks latency rather than fidelity. It is the basis of the real-time variant in Section 4.

Diffusion Forcing (Chen et al., 2024a). It gives every token in a sequence its own noise level instead of one level shared by the whole clip, so a causal model can produce future tokens without fully diffusing the past ones. Rollouts then run past the horizon the model was trained on, where fixed-horizon baselines diverge.

Ditto (Li et al., 2024). It is the earliest real-time avatar system we discuss and the one furthest from our design, generating trajectories in a compact motion space rather than pixels and rendering

them with a warping decoder. The framework is optimized end to end for streaming with a low first-frame delay.

CausVid (Yin et al., 2025). It rewrites a pretrained bidirectional diffusion transformer as an autoregressive one and distills a 50-step teacher into a 4-step generator with distribution matching. A key-value cache makes the cost of a new frame independent of how much video precedes it, which is what turns the model into a streaming one.

MAGI-1 (Sand AI, 2025). It carries that design to foundation-model scale, predicting fixed-length chunks in sequence under a noise level that grows monotonically along the rollout. Peak inference cost stays constant in the length of the video.

Self-Forcing (Huang et al., 2025). It names the failure teacher forcing leaves behind, where a model trained on ground-truth context meets its own imperfect context at inference and the mismatch compounds. Training instead rolls the student out on its own samples and supervises the whole generated sequence, with a few-step student and a truncated gradient keeping the rollout affordable.

Live Avatar (Huang et al., 2026a). It is the closest prior system to the variant we describe. A 14B bidirectional model is distilled in two stages into a causal few-step streaming one whose key-value cache stores noisy rather than clean history, and that lossy conditioning suppresses the error accumulation a long rollout otherwise produces. Assigning each GPU a fixed denoising timestep spreads the sampling chain over several devices.

SoulX-FlashTalk (Shen et al., 2025). It makes the same architectural choice we do, keeping attention bidirectional inside each chunk rather than enforcing a unidirectional mask, on the argument that the correlations within a chunk carry motion coherence and detail. A retrospective self-correction mechanism lets the 14B model recover from accumulated error over an unbounded rollout.

LiveTalk (Chern et al., 2025). It distills OmniAvatar (Gan et al., 2025) on the Wan2.1-T2V-1.3B backbone into a causal 4-step model that generates video block by block from text, a reference image, and audio. The authors observe that Self-Forcing distillation under multimodal conditioning produces flicker, black frames, and loss of quality, and repair it through curated condition inputs and a revised initialization and schedule for the on-policy stage. Coupled with an audio language model, the system reports 24.82 FPS with a first-frame latency of 0.33 s.

Avatar Forcing (Ki et al., 2026). It moves the target from speaking to conversing, conditioning the avatar on the audio and the motion of the person it is talking to so that it reacts to nods and laughter. Preference optimization against samples generated without the user condition teaches expressive reaction without labels for it. Its real-time claim is a response latency, measured on the motion generator alone with the audio features precomputed, rather than a generation rate above playback, which the paper does not report. It is therefore cited here and left out of the comparison of Section 4.5.

SoulX-FlashHead (Yu et al., 2026). It targets the same problem at 1.3B parameters, with a cache of audio context that keeps acoustic features stable when they come from short fragments. Distillation is guided by ground-truth motion priors to counter identity drift.

SoulX-LiveAct (Zhen et al., 2026). It attacks the same rollout from the training side, propagating temporally adjacent frames as latent neighbors held at a single noise level, so that the signal the model learns from is not assembled from mismatched diffusion states. A structured memory then compresses the keys and values of causal attention into a fixed-length representation, which holds inference cost constant and removes the short-term buffer of motion frames that otherwise bounds how far a rollout can run.

Avatar-Forever (Li et al., 2026b). It separates two goals that a sequential distillation pipeline couples. One branch distills the full model into a few-step generator. A second trains a lightweight adapter for long rollouts by perturbing early history, rolling the error forward over several chunks,

and supervising the degraded result. A chunk-wise feature cache removes redundant recomputation of history during streaming. Built on the 22B LTX-2.3 foundation model, it reports 27.2 FPS at 768×512 on a single H100 GPU.

Systems without released code. Three more streaming avatar systems follow the same line. None of them has released inference code at the time of writing, so we describe them here and leave them out of the comparison of Section 4.5. StreamAvatar (Sun et al., 2026) adapts a human video diffusion model to streaming through autoregressive distillation followed by adversarial refinement, and generates talking and listening behavior with body gestures. A reference sink and a positional re-encoding anchored to the reference keep identity stable over long rollouts. It reaches real time on two H800 GPUs. JoyStreamer-Flash (Li et al., 2026a) is a causal model of 1.3B parameters. It gives the first frames more denoising steps than the rest, injects noised previous frames as a motion condition, and resets the positional encoding together with the cache so that the rollout has no length limit. It reports 16 FPS on a single GPU. DynaForcing (Huang et al., 2026b) characterizes a failure of self-forcing with distribution matching that bears on our own recipe. The student converges to a nearly static output, which the authors trace to the mode-seeking reverse KL objective and to self-conditioning without an anchor. Rollouts anchored to ground-truth dynamics, an explicit motion reward, and perturbed reference images bring motion back to the level of the teacher.

Our real-time variant shares the two-stage shape of the streaming works above and borrows the self-forcing rollout and the distillation objective. It differs in what it does not adopt. Attention stays full inside the chunk, which is the choice SoulX-FlashTalk also argues for, no key-value cache is kept, and the state crossing a chunk boundary is two latent frames rather than a cache of the whole rollout. Every system above that reports a throughput figure reaches it on a multi-GPU node, on a data-center GPU, or on a smaller model, so the single-device budget of Section 4 is the constraint that shapes our design.

B Data pipeline

B.1 Data Sources

The first source is a web collection assembled by the authors: publicly accessible videos from open datasets and large video platforms, segmented into scenes by shot detection. It carries the Russian-language content that motivates the bilingual conditioning of Section 3. Appendix B.6 states its provenance and the exclusion of evaluation data. We retain the scenes that contain at least one detected face and last longer than 10 seconds, which gives a base set of approximately 3.8M videos. This set is then annotated and filtered by active-speaker detection (Appendix B.2), and only afterwards on technical quality: footage in which the lips or the hands are blurred is useless for lip-sync supervision, so we score the surviving videos with Q-Align (Wu et al., 2023) in its technical-quality mode and keep those above 4.0.

The second source is the public TalkVerse corpus (Wang et al., 2025): we use its Panda-70M (Chen et al., 2024b) portion and keep the videos longer than 5 seconds, which gives 700K videos. It is already filtered for technical quality, so no further quality stage is applied.

B.2 Active-Speaker Detection and Clip Extraction

Both sources pass through the same active-speaker stage. We run LR-ASD (Liao et al., 2025),[3] which yields face tracks with a per-frame ASD-score for the confidence that the tracked face is producing the speech in the audio track. All thresholds below are applied to this score after smoothing with a moving average over 5 frames. A clip is used only if a single face track covers it without interruption. Since tracks break on fast head turns, occlusions, and cuts, this acts as an implicit filter on face motion.

From the web collection we cut fixed-length 10-second clips. Within each uninterrupted face track we average the smoothed ASD-score over every 10-second window and take the arg-max window, keeping it only if it contains exactly one speaker and its average exceeds 2.0. The window search

[3] We replace the default face detector of LR-ASD with YOLOv11n-face (Kanametov, 2024), a lighter and faster detector that also improves detection quality on our data.

recovers usable material from videos that are unusable as a whole and biases the corpus toward speech-dense stretches in which the speech matches the visible face.

For the TalkVerse subset we threshold the maximum of the smoothed ASD-score rather than its mean, which removes only the clearly suspicious cases. Exactly one clip is taken per video, always from its beginning, with its length set by the duration of the source: a video shorter than 10 seconds yields a 5-second clip, a video of 10 seconds or longer a 10-second one. The clip is kept only if the face track is unbroken over that span. There is no window search, since the subset is already filtered for lip-sync. Opening clips restore the variety of speech density that the arg-max search removes from the web part, and they yield semantically correct speech onsets, since they begin at a scene-segmentation cut. The two parts thus compensate for each other's biases.

The two clip lengths correspond to the two conditioning regimes of Appendix C.4. A 5-second clip is exactly one generation chunk of 125 frames at 25 FPS and is used for the training steps that carry no memory. A 10-second clip is split into two halves: the first becomes the memory clip V_{mem}, the second the chunk the model is asked to denoise. Only 10-second clips therefore supply memory-conditioned steps, which is why the web collection is cut at that length throughout.

The web part is finally deduplicated: we embed the source videos with InternVideo2-1B (Wang et al., 2024b), cluster the embeddings, and subsample the anomalously large clusters, which brings it to 610K clips.

B.3 Captioning

Every retained clip is annotated by Qwen3-VL-32B-Instruct (Bai et al., 2025), served through vLLM (Kwon et al., 2023) with FP8 weights and an FP8 key-value cache. Frames are sampled at 2 FPS and resized to an area of at most 512×512.

For a 10-second clip only the second half is captioned, that is the last 5 seconds, giving 10 frames: the first half is the memory and is never generated, so the caption must describe the target chunk alone, as it does at inference. Five-second clips are captioned in full.

The system prompt is written for audio-driven human animation rather than generic video description: it asks for lip dynamics in articulatory terms, facial micro-expressions described through visible muscle changes, the rhythm of visible speech and its coordination with head motion, body posture and motion quality, camera movement, and the static content of the scene. It forbids inventing spoken content, so a caption never leaks a transcript that the model could learn to reproduce from text alone. It also forbids timestamps and frame indices, so that the caption stays valid under chunk-wise long-form generation.

Three granularities are produced in a single pass: an exhaustive description of 300 or more words, a standard caption of 150 to 200 words, and a summary of 30 to 50 words. Two such requests are issued per clip, one constrained to English output and one to Russian, so the Russian captions are generated natively rather than translated. The resulting six captions per clip are sampled at training time with fixed per-language, per-granularity probabilities.

B.4 Pose track augmentation

Keypoints are augmented offline, after extraction and before rendering, once per clip rather than once per frame, since a perturbation that changes every frame would teach the model to smooth noise instead of to read proportions. The skeleton is rebuilt from the neck outward along its tree, with a separate scale drawn uniformly from $[0.9, 1.1]$ for each of its seven bone groups, the head, the shoulders, the torso, the upper arms, the forearms, the thighs and the shins, so joint angles survive and only segment lengths change. Hands ride rigidly on the new wrist position and the face on the new nose position, so fingers and facial geometry are never deformed, and a bone whose endpoints fall below the keypoint confidence threshold of 0.3 is left as it was. Global scaling and translation are available in the same pass and are switched off for this corpus. This is what lets a driving pose and a reference image disagree about build at inference, since the model has seen the same gesture carried by bodies of different proportions.

B.5 SILENT CLIPS

The silent partition is mined from the same sources as the rest of the corpus, in three stages.

The first inverts the active-speaker stage. Windows are kept where the LR-ASD score (Liao et al., 2025), smoothed over five frames, stays below 0.4. That score also fails in a specific way, returning a low value for a face that is in fact speaking when the head sits at the edge of the frame, and those cases are removed rather than treated as silence.

The second filters for quality. A clip is kept when its Q-Align video score (Wu et al., 2023) is at least 4.1. Clips whose audio is corrupt or of low quality are dropped by the Audiobox production-quality score (Tjandra et al., 2025), clips containing singing by an audio spectrogram transformer trained on AudioSet (Gong et al., 2021), and clips carrying a shot transition by TransNet V2 (Souček & Lokoč, 2020). Static and near-static footage is removed, since a still frame teaches nothing about how a quiet face moves, and the survivors are deduplicated by metadata and by clustering.

The third is a rule-based filter over the captions. The captioner writes recognizable constructions when a person speaks, sings, eats or drinks, and equally when a person is silent, so a small set of rules removes much of what the first two stages let through and labels the rest. The same rules separate sport, people eating or drinking, and people working on their own face, which is a large class of beauty content. Eating was excluded from the final partition.

B.6 PROVENANCE AND EXCLUSION OF EVALUATION DATA

Both parts of the corpus consist of publicly posted video of real people, as do the datasets on which audio-driven animation is commonly trained and evaluated, such as VoxCeleb, HDTF, CelebV-HQ, and TalkVerse. The web collection was assembled by the authors from open datasets and large video platforms and curated by the pipeline above. The TalkVerse part is used under the non-commercial research terms of TalkVerse and Panda-70M, which distribute source URLs and timestamps rather than video. The corpus itself, that is the source videos, the extracted clips, their captions, and the lists of source URLs, is not distributed.

Evaluation data are excluded from the training corpus at two levels, and the check runs over both parts of the corpus, including the silent partition, before captioning. At the source level, training videos that share a source video identifier with a benchmark clip are dropped. At the identity level, ArcFace embeddings (Deng et al., 2019) of the benchmark faces are compared with the face tracks of the training clips, and clips that match a benchmark speaker above the same-identity threshold are removed, so that the reported scores measure generalization to unseen people rather than to unseen clips of seen people.

B.7 BACKGROUND STABILITY

A background that shakes in the training clips cannot be repaired by any objective, so both autoregressive stages train on a background-stable subset of the corpus. A frozen portrait-matting network (Lin et al., 2022) is run over every clip, which gives a person mask on the latent grid, pooled spatially and taken as a maximum over the video frames each latent folds. Stability is then scored by the minimum structural similarity (Wang et al., 2004) over the pairs formed from the first, middle and last frame, computed on the pixels both frames of a pair agree are background, with the share of the frame that background covers recorded alongside it. The corpus is already largely static, so the subset is built by ranking on that score under a floor on the background share rather than by a threshold, since ranking alone favours clips whose background is a sliver.

B.8 Resulting Corpus

Table 4: Data pipeline and the resulting corpus. Q-Align technical filtering and cluster-based deduplication apply only to the web collection. The active-speaker stage and the exclusion of evaluation data are shared.

Stage	Web collection	TalkVerse (Panda-70M part)
Videos with a detected face, after duration filtering	>10 s: 3.8M	>5 s: 700K
Single-speaker, smoothed ASD-score >2.0	window mean	max
Q-Align technical score >4.0	✓	Not applied
Cluster-based deduplication	✓	Not applied
Exclusion of evaluation data (Appendix B.6)	✓	✓
Clip length used for training	10 s	5 s and 10 s
Retained clips	610K	563K
Total	1.17M clips	

Table 4 gives what each stage of the pipeline keeps. The final corpus consists of 1.17M clips, split almost evenly between the two sources: the web part supplies the Russian-language coverage, the TalkVerse part high-resolution single-speaker footage. Figure 3 shows the distribution of spoken languages: English accounts for slightly over half of the corpus and Russian for roughly a quarter, which makes native Russian control feasible without a dedicated fine-tuning stage.

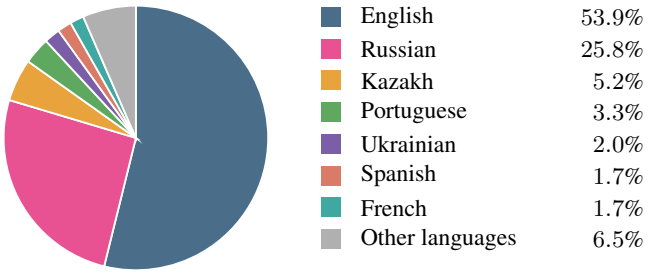


Figure 3: Distribution of spoken languages over the 1.17M training clips. Languages below 1.5% are pooled into Other languages.

C BASE ARCHITECTURE

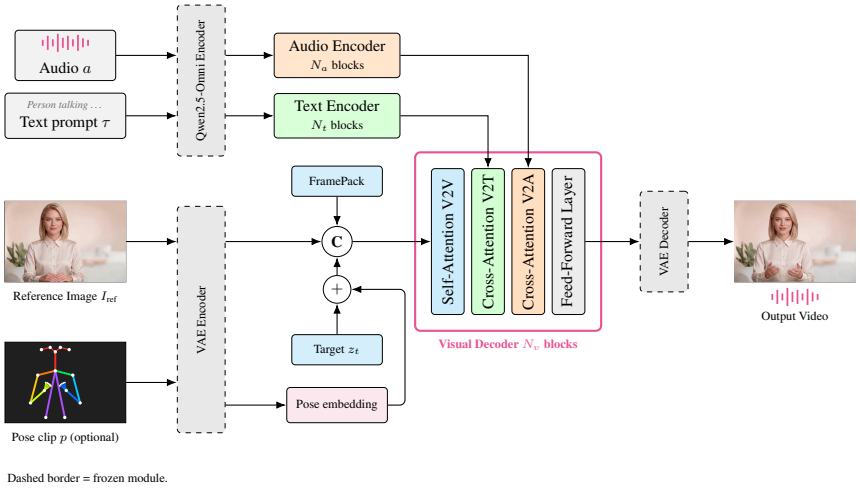


Figure 4: Overview of the base architecture. Text features from the frozen Qwen2.5-Omni Thinker and acoustic features from its frozen audio tower Xu et al. (2025) feed two trainable DiT-internal stacks of N_t text blocks and N_a audio blocks. The reference image is encoded by a frozen Hunyuan 3D VAE Kong et al. (2024), and its latent is concatenated (denoted by C) with the noisy target latent z_t and a multi-scale FramePack Zhang et al. (2025c) memory of past frames to form the visual sequence consumed by the DiT. An optional pose clip is encoded by the same VAE and added to the target tokens through a separate zero-initialized projection, which leaves the reference and the memory untouched. Each DiT block applies self-attention (V2V), cross-attention to the text tokens (V2T), an optional cross-attention to the audio tokens (V2A) injected periodically across the stack, and a feed-forward layer. The denoised visual latent is decoded back to pixel space by the frozen VAE decoder.

C.1 CONDITIONING STREAMS

The model is conditioned on four heterogeneous inputs, and on an optional fifth described in Section 3.

- A textual prompt τ that describes the scene, camera, and behavior of the speaker. The frozen Qwen2.5-Omni Thinker (Xu et al., 2025) encodes it into a sequence of token features of dimension 3584. The prompt is placed before the audio segment in the Thinker input. Since the Thinker uses causal attention, the extracted prompt features do not depend on the later audio tokens. The trainable DiT-internal text encoder adapts these features before the V2T conditioning path. A pooled CLIP (Radford et al., 2021) sentence embedding of dimension 768 is used as a global style vector that is fused with the diffusion time embedding.
- A speech waveform a sampled at 16 kHz. The Qwen2.5-Omni audio tower encodes it into an acoustic token stream of dimension 3584. Our model uses the output of this tower directly instead of the Thinker hidden states at audio-token positions. The trainable DiT-internal audio encoder adapts these features before the V2A conditioning path. The audio stream drives lip and head motion through the audio blocks and the V2A cross-attention layers.
- A single reference image I_{ref} of the target identity. The image is mapped to the latent space by the Hunyuan VAE (Kong et al., 2024) and patchified by the same visual tokenizer as the noisy video latents.
- An optional memory clip V_{mem} of past frames, used to extend a generation beyond a single chunk of 125 frames at 25 FPS without losing identity, camera, or motion continuity. The training schedule for the memory and the reference is detailed in Appendix C.4.

The textual prompt is provided in both English and Russian at three granularity levels of short, mid, and long, with sampling probabilities of 0.075, 0.075, and 0.35 per language. The speech waveform, the text prompt and the reference image are dropped independently during training with probability 0.1, which supplies the unconditional branches for classifier-free guidance (Ho & Salimans, 2022). The memory clip has a dropout of its own, given in Appendix D.3, since a step without memory is the state of the first chunk of every rollout rather than an unconditional branch.

C.2 Latent video tokenization

The visual stream operates on 5D latents $z \in \mathbb{R}^{B\times T\times H\times W\times C}$ produced by the Hunyuan VAE with temporal compression ratio 4 and spatial downsampling factor 8. The training resolutions are 512×512, 768×512, and 512×768, with 125 raw frames per clip producing $T = 32$ latent frames. The reference image contributes a single latent frame and the memory clip contributes up to 30 latent frames.

Latents are split into non-overlapping spatiotemporal patches of size $(1, 2, 2)$ before being projected to the model dimension. This patch size is used for the main noisy latents, the reference latent, and the 2 most recent memory latents. The preceding 4 memory latents use patches of size $(2, 4, 4)$. The oldest 24 memory latents use patches of size $(4, 8, 8)$. Each scale has a separate linear projection that does not share weights with the main visual projection. Spatial RoPE positions of the two coarse scales are subsampled by the same factor as their patch size, so one coarse token covers a larger region of the frame. This three-scale layout adapts FramePack (Zhang et al., 2025c) to compressed long-context video memory. The memory reaches 30 latent frames back, about 4.8 seconds of history, at a token cost far below 30 dense latent frames.

Rotary frequencies on the two spatial axes are divided by a factor of 2.0 at the resolutions we train on, while the temporal axis is not rescaled.

C.3 DiT backbone

The backbone is a 3D Diffusion Transformer (Peebles & Xie, 2023) with three parallel token streams.

Text encoder. A trainable stack of N_t text transformer blocks with self-attention, RoPE (Su et al., 2024) positional encoding in 1D, RMSNorm on queries and keys, and AdaLN-zero modulation (Peebles & Xie, 2023) driven by the diffusion time embedding combined with the pooled CLIP sentence embedding. We use $N_t = 2$.

Audio encoder. A trainable stack of N_a unconditional transformer blocks with self-attention and 1D RoPE on the audio token sequence. We use $N_a = 2$. Audio blocks are not modulated by the diffusion time, because the speech track is the same across all denoising steps.

Visual decoder. A stack of N_v visual transformer blocks. Each block executes, in order, self-attention with 3D RoPE on visual tokens, cross-attention to text, optional cross-attention to audio, and a feed-forward layer. The self-attention, text cross-attention, and FFN are gated by AdaLN-zero parameters predicted from the diffusion time embedding combined with the pooled CLIP sentence embedding, while the audio cross-attention is added as a residual without modulation. The model uses $N_v = 32$ blocks with $d_{\text{model}} = 1792$, $d_{\text{ff}} = 7168$, and per-head axes of $(16, 24, 24)$, summing to a head dimension of 64.

The model uses the patch size $(1, 2, 2)$, the 16-channel VAE latent, the 3584-dimensional text and audio features from Qwen2.5-Omni, and the 768-dimensional pooled CLIP vector. Table 5 lists the remaining settings. The head dimension is the sum of the three RoPE axis widths, where the first axis carries the temporal position and the other two carry height and width.

Table 5: Architecture settings.

Setting	Value
Visual blocks N_v	32
Text blocks N_t	2
Audio blocks N_a	2
Model dimension	1792
Feed-forward dimension	7168
Time-embedding dimension	512
Per-head RoPE axes (t, h, w)	$(16, 24, 24)$
Head dimension	64
Blocks with audio cross-attention	13

The three visual sub-streams of memory, reference, and current latents are concatenated into a single sequence, in that order, after being tagged by a learned segment embedding of three classes. AdaLN modulation of the reference and memory groups comes from a time embedding evaluated at $t = 0$ rather than at the sampled diffusion time, so those tokens are always modulated as clean latents while the target tokens follow the noise level of the step. The temporal RoPE positions assigned to each sub-stream and the training schedule that mixes the memory and reference are described next.

C.4 Reference image and memory placement

The reference image I_{ref} and the memory clip V_{mem} are positioned on the temporal axis with two complementary training schemes that we mix during the run. A step sampled without a memory clip, at the rate that Table 7 gives, takes its reference frame from the source video immediately before the chunk the model is asked to denoise. A step that uses a memory clip takes its reference frame from a position before the memory, far in the past relative to the denoised chunk. The second mode mirrors the autoregressive inference setup in which the model is conditioned on a long backlog of previously generated frames.

On the temporal RoPE axis the current denoised chunk occupies positions 0 to $T - 1$ with $T = 32$ latent frames. The reference image is fixed at the RoPE position 40, well outside the chunk range. The memory clip, when present, is placed at negative RoPE positions before the chunk. We use the mixed training schedule and explicit RoPE separation to discourage the model from reusing the reference pose at every chunk boundary. This failure mode can arise when the reference is always placed immediately next to the target chunk during training. Table 6 gives the resulting layout of the visual sequence.

Table 6: Groups of the visual sequence with their patch sizes and temporal RoPE positions. Latent frames refer to the VAE time axis, which is compressed by a factor of 4 relative to video frames. A target chunk of 32 latent frames corresponds to $(32 - 1) \times 4 + 1 = 125$ video frames, which is 5 seconds at 25 FPS.

Group	Latent frames	Patch (t, h, w)	Temporal RoPE positions
Target chunk	32	$(1, 2, 2)$	0 to 31
Reference image	1	$(1, 2, 2)$	40
Memory, fine	2	$(1, 2, 2)$	$-2, -1$
Memory, medium	4	$(2, 4, 4)$	$-6, -4$
Memory, coarse	24	$(4, 8, 8)$	-30 to -10, step 4

C.5 Audio injection

Audio is injected into the visual stream only at a subset of decoder blocks, chosen to be approximately uniformly spaced in the network. Audio enters at 13 of the 32 blocks, at indices $0, 3, 6, 9, 12, 15, 18, 21, 23, 25, 27, 29, 31$ counting from zero. The spacing narrows toward the output, from three blocks to two, so a larger share of audio-conditioned blocks sits close to the output projection where lip geometry is resolved. At each audio-enabled block, only the current visual

latents attend to audio, while reference and memory tokens remain frozen during the V2A cross-attention. The audio-to-visual attention is aligned per latent frame, so that each visual frame attends to its own slice of the audio token sequence.

D BASE TRAINING AND LONG-FORM SAMPLING

D.1 FLOW-MATCHING OBJECTIVE

The model is trained with rectified-flow matching (Lipman et al., 2023; Liu et al., 2022; Esser et al., 2024). Let z_0 be the clean video latent and $\epsilon \sim \mathcal{N}(0, I)$. The noisy latent at time $t \in [0, 1]$ is

$$z_t = (1 - t)\, z_0 + t\, \epsilon,$$

and the target velocity is $v(z_t, t) = \epsilon - z_0$. The network f_θ predicts $\hat{v} = f_\theta(z_t, t, c)$, where c collects the text, audio, reference, and memory conditioning. The training loss is the mean squared error

$$\mathcal{L}_{\text{flow}} = \mathbb{E}_{t,\epsilon,c,z_0}\left[\|f_\theta(z_t, t, c) - v(z_t, t)\|_2^2\right].$$

Timesteps are sampled from a reweighted logit-normal distribution. We first draw $u = \sigma(\eta)$ with $\eta \sim \mathcal{N}(0, 1)$, then push mass toward high noise by

$$t = \frac{\kappa\, u}{1 + (\kappa - 1)\, u}.$$

This is the scaled logit-normal schedule of (Esser et al., 2024) with scale $\kappa = 5$. The transformation allocates more training samples to high-noise states than the unscaled logit-normal distribution.

D.2 FACE LOSS

In addition to the flow-matching MSE we use a region-weighted auxiliary loss on the face. The term itself is standard, the weighting we place on it is not. For each clip, a face mask M aligned to the latent grid singles out the speaker face. The face loss is

$$\mathcal{L}_{\text{face}} = \frac{\sum (f_\theta(z_t, t, c) - v)^2 \odot M}{\sum M + \delta},$$

where δ prevents division by zero, and the total loss is $\mathcal{L} = \mathcal{L}_{\text{flow}} + \lambda_{\text{face}}\mathcal{L}_{\text{face}}$. For λ_{face} we use an adaptive weighting that is inversely proportional to the face-area ratio of the clip,

$$\lambda_{\text{face}} = \text{clip}\left(\frac{r_{\text{ref}}}{r_{\text{face}}},\ \lambda_{\min},\ \lambda_{\max}\right), \qquad r_{\text{face}} = \frac{\sum M}{|M|},$$

where $|M|$ is the number of elements the mask spans, so r_{face} is the share of the latent grid the face covers, r_{ref} is the share the weight is calibrated at, and $\lambda_{\min}, \lambda_{\max}$ bound the resulting weight. The constants $r_{\text{ref}} = 0.04$, $\lambda_{\min} = 0.15$, and $\lambda_{\max} = 2.5$ are chosen in accordance with the empirical face-area distribution of the training data, where the median visible face covers a few percent of the frame. We use the masked mean form, which keeps the gradient on the face region stable when the face occupies only a small fraction of the frame. With a fixed λ_{face}, clips with large faces can contribute a disproportionately strong face gradient. We observed that this setting was associated with a tendency toward closer face framing during training, since the model can reduce the face term by enlarging the face. The adaptive rule is intended to balance the face contribution across scales and retain a useful supervision signal when the speaker occupies a small fraction of the frame.

D.3 TRAINING STAGES

The base model is not trained in one run. A first run learns speech-to-video from the text-to-video initialization, without a pose branch and without the multi-scale memory, for 14k steps. A second run starts from its exponential moving average and adds the two things the real-time variant later depends on, the three-scale FramePack memory and the pose branch, together with the silent partition of Section 2, for a further 6k steps. Splitting the run this way keeps the harder conditioning out of the phase where the model is still learning to follow speech at all. Table 7 gives the settings of both runs, and the second run is what Section 3 describes and what stage 1 of Section 4 starts from.

Table 7: Settings of the two base training runs.

Setting	Speech-to-video	Memory, pose, silent clips
Initialized from	Kandinsky 5.0 Video 2B	previous run, EMA
Optimizer steps	14000	6000
Learning rate	3×10^{-5}	2×10^{-5}
Weight decay	10^{-3}	10^{-3}
β	$(0.9, 0.95)$	$(0.9, 0.95)$
Warmup steps	200	200
Gradient-norm clip	1.0	1.0
EMA decay	0.999	0.999
H100 GPUs	64	64
Per-rank batch	2	1
Global batch	128	64
Parallelism	FSDP, 8-way mesh	FSDP, 8-way mesh
Frames per clip	125	125
Latent frames	32	32
FramePack memory	absent	three scales, buckets $2, 4, 24$
Pose branch	absent	present
Memory dropout	0.7	0.6
Pose dropout	n/a	0.5, prefix and suffix 0.2 each
Pose keypoint scales	n/a	per bone group, $\mathcal{U}(0.9, 1.1)$
Audio, text, reference dropout	0.1 each	0.1 each
Face loss	adaptive, $r_{\text{ref}} = 0.04$	adaptive, $r_{\text{ref}} = 0.04$

D.4 Pretraining initialization

The DiT is initialized from the corresponding Kandinsky 5.0 text-to-video and image-to-video checkpoints (Arkhipkin et al., 2025), which were pretrained at 512 and 768 resolution without audio conditioning. The DiT-internal text and audio encoders, the audio cross-attention layers, the memory projections, and the reference-image segment embedding are initialized fresh. The Qwen2.5-Omni Thinker and audio tower, the pooled CLIP encoder, and the VAE remain frozen throughout training. All DiT blocks and the newly introduced conditioning layers are trainable.

D.5 Multi-resolution training

Each step samples a single resolution from $\{(512, 512), (768, 512), (512, 768)\}$ and a bucket of frame counts, with dense FlashAttention (Dao et al., 2022; Dao, 2024) and the complete three-scale FramePack memory. A resolution-aware batch accumulator keeps gradient batches uniform in shape, which is necessary for compiled FSDP graphs.

D.6 Optimization

Training uses AdamW at learning rate 3×10^{-5}, weight decay 10^{-3}, $\beta = (0.9, 0.95)$, and global gradient-norm clipping at 1.0. The schedule is constant with linear warmup over 200 steps. An exponential moving average of the weights with decay 0.999 is maintained for inference. Mixed precision uses bfloat16 parameters and outputs with float32 gradient reduction. AdaLN modulations and time embeddings are kept in float32 for numerical stability.

D.7 Distributed training

The model is trained on 8 nodes of 8 NVIDIA H100 GPUs with FSDP (Zhao et al., 2023), an 8-way mesh on the DiT and the same on the embedders, without tensor parallelism. Activation checkpointing (Foundation, 2025) is applied to every visual decoder block.

D.8 SILENT SPEECH

A model trained only on speech treats a quiet audio window as an input it has never seen. The silent partition of Section 2 removes that gap. Nothing in the architecture or the objective handles it, and no dedicated prompt is used. A silent clip is an ordinary training sample whose audio happens to be flat and whose caption already reports closed lips and a still face.

This is separate from the audio dropout of Section 3, which zeroes the acoustic features to supply an unconditional branch for guidance. Dropout removes the audio signal, the silent partition supplies a real one that happens to be quiet. At inference a speech track is padded with silence to a whole number of chunks and the output video is trimmed back to the original audio length, so a partial final chunk is an ordinary silent window rather than a special case.

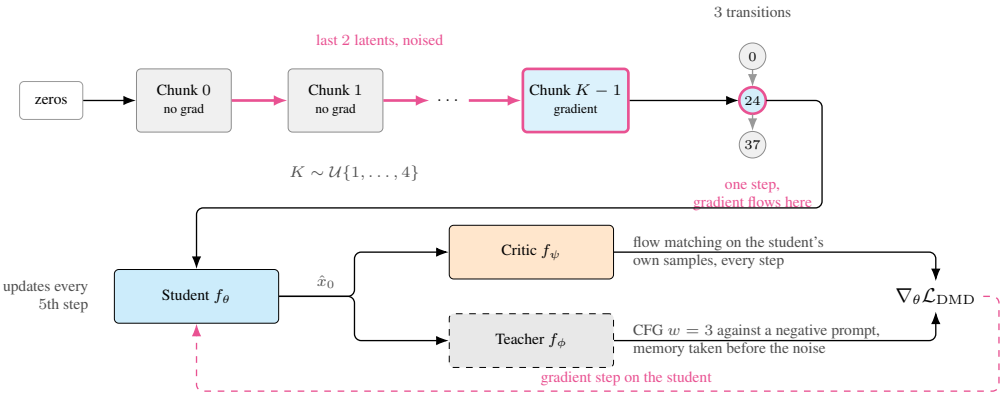
D.9 LONG-FORM INFERENCE

A long video is generated as a sequence of 125-frame chunks. The first chunk sees the reference image, the corresponding slice of the speech track, and the prompt, with the memory latents set to zero. Each later chunk reuses the same reference and fills the three memory scales from the generated history, taking the last 2 latent frames at the target patch scale, the 4 before them at $(2, 4, 4)$, and the 24 before those at $(4, 8, 8)$. Memory is read from latents before decoding, so the VAE encode direction is never repeated and peak memory does not grow with output length.

Sampling runs 50 Euler steps on the timestep transform of Appendix D.1 at scale $\kappa = 5$, the scale training uses. Guidance uses one unconditional pass per step that drops text and audio jointly, replacing the prompt with a fixed negative prompt and zeroing the acoustic features. The reference and memory latents stay in both branches, so identity conditioning is never guided. Writing v_c and v_u for the two predicted velocities, the update follows $v_u + w\,(v_c - v_u)$ with $w = 3$, so one scale controls prompt adherence and lip-motion strength together.

This sampler costs 100 network evaluations per chunk, 50 steps on each of two branches. Section 4 describes a variant that reduces this to 3 on a shorter chunk, at the cost of the multi-scale memory.

D.10 AUTOREGRESSIVE TRAINING STAGES



Dashed = frozen. Gray chunks carry no gradient. All three copies start from the stage-1 exponential moving average.

Figure 5: One step of stage 2. The student rolls forward over K chunks, passing its own last two latent frames forward as memory after mixing noise into them at a sampled level, with gradients disabled everywhere except one denoising step of the last chunk. The teacher and the critic score the clean latent that step predicts, and their difference forms the generator gradient. The critic is refit to the student on every step, the student updates on every fifth.

Table 8 collects the settings of the two stages of Section 4. Both continue the chain of Table 7: stage 1 starts from the exponential moving average of the second base run, and stage 2 from the exponential moving average of stage 1. The step counts are the checkpoints the following sections

use, not the length the schedulers were configured for, which was longer in both cases. The masked background term of stage 2 follows the public training code released for the FlashTalk line.[4]

Table 8: Training settings of the two autoregressive stages.

Setting	Stage 1	Stage 2, student	Stage 2, critic
Initialized from	base run 2, EMA	stage-1 EMA	stage-1 EMA
Objective	flow matching + face	distribution matching	flow matching
Training iterations	1500	1300	1300
Chunk length, latent frames	8	8	8
Memory source	ground truth	own rollout, noised	own rollout, noised
Cold-start probability	0.5	n/a	n/a
Chunks per step K	1	$\mathcal{U}\{1,\dots,4\}$	$\mathcal{U}\{1,\dots,4\}$
Memory noise level	none	$\mathcal{U}(0,0.1)$	$\mathcal{U}(0,0.1)$
Conditioning dropout	0.1	none	none
Items with pose	0.5	0.5	same as the student
Partial pose, of posed items	prefix or suffix, 0.2 each	switches on or off with probability 0.25 each	same as the student
Auxiliary terms	face loss	latent regression 0.1, background alignment 2	n/a
Learning rate	2×10^{-5}	2×10^{-6}	4×10^{-7}
Weight decay	10^{-3}	10^{-2}	10^{-2}
β	$(0.9, 0.95)$	$(0.9, 0.95)$	$(0.9, 0.99)$
Gradient-norm clip	1.0	10.0	10.0
Warmup steps	200	0	0
EMA decay	0.999	0.99, from step 100	n/a
Update cadence	every step	every 5th step	every step
H100 GPUs	32	64	64
Per-rank batch	2	1	1
Parallelism	FSDP within node	FSDP	FSDP

E Evaluation of the real-time model

E.1 Protocol

The harness runs three decoupled stages. Prepare freezes the clip manifest once and extracts, at a fixed resolution and 25 FPS, a reference frame, the driving audio and, where one exists, the ground-truth video, so that every model sees byte-identical inputs. Generate turns that triple into a video, each model running in its own environment at its own native resolution. Score runs each metric in its own environment and returns one value per clip, rescaling to the benchmark geometry where a metric needs a common one. A table cell is the mean over the 50 clips of a benchmark, and a clip on which a metric fails, for instance when the face detector finds nothing, is dropped from that metric rather than counted as zero. In every table of this appendix the best value in a column is in bold, and a tie is bold on both rows.

Four benchmarks are used, all frozen at 512×512, 768×512 or 512×768 and 25 FPS, each clip at the geometry closest to its source aspect ratio.

- **HDTF-30s** (Zhang et al., 2021), a 50-clip split chosen by image quality and face size, first 30 seconds of each clip.
- **EMTD-30s** (Meng et al., 2025), clips of at least four seconds with a quality floor and small faces excluded, first 30 seconds.
- **CelebV-HQ** (Zhu et al., 2022), talk-labelled clips screened on the ground-truth lip-sync score and free of letterboxing, full clip.
- **EMTD-HDTF audio**, a cross-driven stress test in which an EMTD reference image is driven by gender-matched HDTF speech. The speaker in the audio is not the speaker in the

[4] https://github.com/nanxiaolu/FlashTalk-Training-Code

image, so no ground-truth video exists. It is run at two minutes per clip, the longest every system accepts, and again at five minutes over the systems that accept that, Table 13.

E.2 METRICS

CSIM is the cosine similarity between ArcFace embeddings (Deng et al., 2019) of the reference face and of the generated faces. Sync-C and Sync-D are the confidence and the embedding distance of SyncNet (Chung & Zisserman, 2016), from one pass. IQA and ASE are the technical and aesthetic scores of Q-Align (Wu et al., 2023), also from one pass. All five are no-reference, which is what lets the cross-driven benchmark carry the same set as the three with ground truth and leaves no empty cell. The harness also implements an identity score against a ground-truth prototype and the two distribution distances, and those are left out here because they need ground truth and were never run for two of the baselines, so no complete block exists for them.

E.3 THE TWO AGGREGATE COLUMNS

Both aggregates are computed column-wise inside one table, over the competing models only, and the identity score of Ditto is left out of the CSIM column, since it warps the reference face rather than generating it.

Min-max rescales each metric column so that the best model becomes 1 and the worst 0, with the direction flipped for Sync-D so that higher always reads as better, and a model's score is the mean of its rescaled values. It keeps the size of the gaps, so a model beaten by a hair scores almost level with the winner. It is relative to the field, since the best and the worst model present always score exactly 1 and 0.

Mean rank orders the models inside each column, numbers them from one, and averages the positions. Both aggregates are computed on the scores before rounding, so two cells that print the same value can still be ordered. It discards the size of the gaps, which makes it robust to a single outlying metric and blind to how close the race was.

In Table 3 the two are pooled over the twenty benchmark and metric cells rather than recomputed from the averaged columns, so a model that wins one benchmark by a landslide and loses three narrowly cannot come out ahead. Since every benchmark contributes the same number of cells for a given model, pooling is the same as averaging the four per-benchmark scores.

E.4 BASELINES

SoulX-FlashTalk (Shen et al., 2025) is a Wan-class 14B model. SoulX-LiveAct (Zhen et al., 2026) initializes its self-attention and text and image cross-attention from Wan2.1 and its audio cross-attention from InfiniteTalk, and is designed for hour-scale rollouts. Its model card gives the size as 18B, a 14B backbone with a 4B audio module.[5] Live Avatar (Huang et al., 2026a) is a 14B model with a four-step distilled sampler on a fixed five-GPU topology. SoulX-FlashHead (Yu et al., 2026) is a 1.3B model released in two tiers, Pro and Lite. Pro keeps the Wan2.1 VAE and reaches 10.8 FPS on one RTX 4090, so its authors report real time on two RTX 5090. Lite swaps in the LTX VAE and reaches 96 FPS on one RTX 4090, which makes it, with LiveTalk, one of the two baselines in the same hardware class as ours. Both are evaluated here. Avatar-Forever (Li et al., 2026b) is built on a 22B foundation model and reports 27.2 FPS at 768×512 on one H100, with efficiency and long-horizon robustness distilled in two parallel branches. LiveTalk (Chern et al., 2025) is a 1.3B model distilled to four steps and reports 24.82 FPS at 512×512 on a single RTX 4090. Neither accepts a five-minute request. Ditto (Li et al., 2024) is not a video diffusion model: it drives a motion-space diffusion and a warping decoder, animating the cropped face and compositing it back, so the output keeps the geometry of the reference and is not conditioned on text. Its identity score is therefore near-saturated by construction and carries no information about how well a model reproduces a person, so it is shown in grey in the tables and never counted as a best value.

[5] `https://huggingface.co/Soul-AILab/LiveAct`

E.5 PER-BENCHMARK RESULTS

Tables 9, 10, 11 and 12 carry the four benchmarks that the aggregates of Table 3 pool, and Table 13 repeats the cross-driven one at five minutes.

Table 9: HDTF-30s.

Model	CSIM ↑	Sync-C ↑	Sync-D ↓	IQA ↑	ASE ↑	Min-max ↑	Mean rank ↓
Edge-Avatar (ours)	0.908	**8.11**	7.94	4.57	**3.59**	**0.955**	**2.00**
SoulX-FlashTalk	0.900	7.98	8.19	4.51	3.50	0.859	3.80
SoulX-LiveAct	0.834	7.98	**7.72**	4.41	3.45	0.733	4.60
Live Avatar	0.786	6.67	8.87	**4.62**	3.57	0.684	4.40
LiveTalk	0.765	4.25	10.72	4.61	3.56	0.423	5.80
Avatar-Forever	0.867	4.17	11.50	4.59	3.51	0.488	6.00
SoulX-FlashHead Pro	**0.918**	6.72	8.70	4.46	3.44	0.747	4.80
SoulX-FlashHead Lite	0.887	7.30	8.01	4.15	3.11	0.503	5.80
Ditto	0.963	5.36	9.60	4.17	3.13	0.218	7.50

Table 10: EMTD-30s.

Model	CSIM ↑	Sync-C ↑	Sync-D ↓	IQA ↑	ASE ↑	Min-max ↑	Mean rank ↓
Edge-Avatar (ours)	**0.859**	**7.65**	**8.02**	**4.85**	**3.73**	**1.000**	**1.00**
SoulX-FlashTalk	0.831	7.42	8.36	4.76	3.62	0.886	2.60
SoulX-LiveAct	0.710	7.22	8.05	4.69	3.52	0.733	4.00
Live Avatar	0.696	6.41	8.51	4.69	3.48	0.633	5.40
LiveTalk	0.665	3.92	10.81	4.67	3.46	0.296	8.00
Avatar-Forever	0.724	5.33	9.86	4.84	3.63	0.578	4.20
SoulX-FlashHead Pro	0.818	5.55	9.34	4.71	3.54	0.673	4.40
SoulX-FlashHead Lite	0.702	5.18	9.22	3.98	2.88	0.220	7.20
Ditto	0.953	4.85	9.94	4.63	3.40	0.481	8.00

Table 11: CelebV-HQ.

Model	CSIM ↑	Sync-C ↑	Sync-D ↓	IQA ↑	ASE ↑	Min-max ↑	Mean rank ↓
Edge-Avatar (ours)	**0.863**	**6.98**	**8.18**	4.18	3.17	**0.979**	**1.60**
SoulX-FlashTalk	**0.863**	6.60	8.82	4.05	3.02	0.833	3.40
SoulX-LiveAct	0.761	6.52	8.39	4.01	3.02	0.746	4.40
Live Avatar	0.703	5.52	9.46	**4.23**	**3.20**	0.645	4.20
LiveTalk	0.673	3.52	10.71	4.05	2.98	0.295	7.20
Avatar-Forever	0.800	5.14	9.83	4.18	3.09	0.656	4.60
SoulX-FlashHead Pro	0.847	5.09	9.40	3.93	2.89	0.618	6.00
SoulX-FlashHead Lite	0.704	6.11	8.35	3.45	2.47	0.369	6.00
Ditto	0.954	4.70	9.27	3.77	2.74	0.422	7.25

Table 12: EMTD-HDTF audio, cross-driven, two minutes per clip.

Model	CSIM ↑	Sync-C ↑	Sync-D ↓	IQA ↑	ASE ↑	Min-max ↑	Mean rank ↓
Edge-Avatar (ours)	0.829	**7.99**	7.84	4.80	**3.71**	**0.980**	**1.80**
SoulX-FlashTalk	**0.833**	7.89	8.10	4.70	3.57	0.881	2.60
SoulX-LiveAct	0.718	7.77	**7.62**	4.65	3.52	0.772	3.80
Live Avatar	0.644	6.96	8.12	4.61	3.48	0.606	5.60
LiveTalk	0.605	4.11	10.74	4.64	3.46	0.262	7.80
Avatar-Forever	0.738	6.79	8.77	**4.81**	3.65	0.760	3.60
SoulX-FlashHead Pro	**0.833**	5.89	9.35	4.65	3.53	0.666	4.60
SoulX-FlashHead Lite	0.774	5.44	9.33	4.22	3.10	0.307	7.00
Ditto	0.951	5.06	9.72	4.58	3.39	0.413	8.00

Table 13: EMTD-HDTF audio at five minutes per clip. LiveTalk and Avatar-Forever cannot be run to this length, so this table covers the systems that can and its aggregates are computed over them alone.

Model	CSIM ↑	Sync-C ↑	Sync-D ↓	IQA ↑	ASE ↑	Min-max ↑	Mean rank ↓
Edge-Avatar (ours)	0.824	**8.05**	7.81	**4.85**	**3.76**	**0.979**	**1.60**
SoulX-FlashTalk	**0.832**	7.92	8.11	4.75	3.62	0.869	2.00
SoulX-LiveAct	0.718	7.74	**7.68**	4.71	3.56	0.751	3.20
Live Avatar	0.630	6.94	8.18	4.71	3.53	0.552	4.80
SoulX-FlashHead Pro	0.831	5.90	9.37	4.71	3.57	0.581	4.00
SoulX-FlashHead Lite	0.777	5.42	9.35	4.31	3.14	0.209	5.80
Ditto	0.952	5.02	9.73	4.62	3.41	0.255	6.50

The two lengths run on the same clips and the same references, so the pair measures what the extra three minutes cost. Our numbers move within hundredths: CSIM 0.829 against 0.824, Sync-C 7.99 against 8.05, IQA 4.80 against 4.85 and ASE 3.71 against 3.76.

E.6 Side-by-side human study

The automatic metrics say nothing about whether a rollout looks natural to a person, so the model is also compared against the three baselines that stand closest to it on the aggregates, SoulX-FlashTalk, SoulX-LiveAct and Avatar-Forever, in three side-by-side studies.

Each study runs on pairs drawn from the 50 EMTD clips of Appendix E, ten annotators judging ten pairs each, which gives 100 judgements per category. A pair is shown with the reference image beside it, the two videos unlabelled and their left-right order randomized, so an annotator never knows which system produced which video. Every pair is judged on each category separately, with four options, the left one better, the right one better, both good, or both bad. The two agreement answers are pooled into a tie in the figures below. The categories and what annotators were asked to look for are the following.

- **Visual quality**, the overall look of the video.
- **Motion naturalness**, how natural the movement is, with the motion of the avatar weighed above everything else in the frame.
- **Lip synchronization**, whether the articulation of the lips and the face follows the speech.
- **Identity preservation**, whether the avatar stays the same person over the clip.
- **Teeth and mouth**, the absence of crooked teeth and of an unnatural mouth or tongue.
- **Camera stability**, whether the camera and the focus stay put rather than drifting, zooming onto the avatar or orbiting it.
- **Background stability**, the absence of background degradation and of changes that the scene does not call for.

Figure 2 in Section 4.5 carries the study against SoulX-FlashTalk, Figures 6 and 7 the other two. The three read the same way. Ties carry most categories, which is what systems of this quality produce on short clips, and where annotators do separate the two they separate them in one direction: no category in any of the three studies comes out in favour of the baseline.

Against SoulX-LiveAct the margin is widest on motion naturalness, 53% against 6%, and on visual quality, 49% against 12%, the two categories where ties fall below half. Against Avatar-Forever the widest margins are visual quality, 52% against 9%, and teeth and mouth, 35% against 4%. Camera stability is a tie on almost every pair of every study, so no system drifts the frame, and on background stability the baselines are preferred on no pair at all. This is the axis the no-reference metrics of Appendix E cannot reach, since a rollout that is static and clean scores well on them while reading as lifeless to a person.

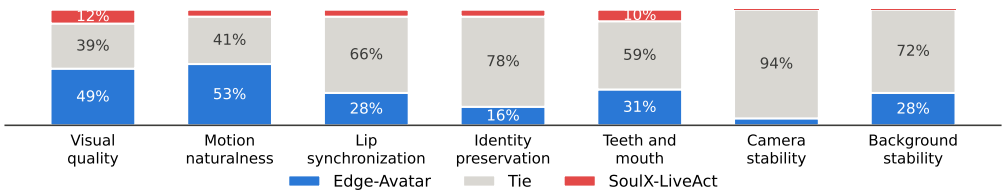


Figure 6: Side-by-side preferences against SoulX-LiveAct.

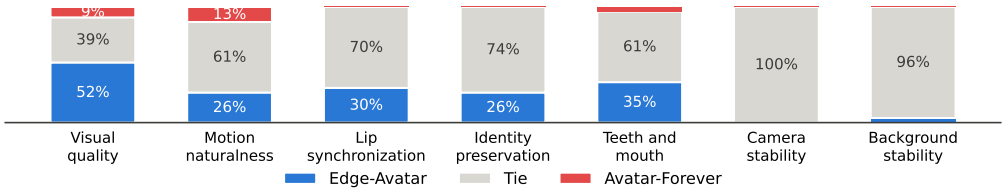


Figure 7: Side-by-side preferences against Avatar-Forever.

F Ablations

Six checks on the EMTD-30s set of Appendix E. Three of them, the face loss, the memory noise and SPD, differ in one thing only. The others compare two moments at which a gesture may enter, the deployed stack against an unoptimized one, and two ways of delivering the same larger frame. The face loss is measured on stage 1, the rest on the deployed checkpoint with its own recipe, block-sparse attention, FP8 linear layers, the 3-step schedule, one seed and one prompt. Generation was sharded clip by clip across a node, which changes nothing about a single clip since no model parallelism is involved.

F.1 Area-adaptive face loss

This one is measured on stage 1, since stage 2 carries no face term. Three runs of the stage-1 recipe of Table 8 differ in that term alone, at the same step and on the same data: the base with the area-adaptive weight of Section 3, one with the term removed, and one that keeps it at a fixed weight of 1. All three are sampled with the guided 50-step sampler of Appendix D.9 on the short-chunk layout, so the numbers of Table 14 do not compare with those of the other ablations. Stage 1 is trained on ground-truth memory and meets its own history over a 30-second rollout, so its identity drifts, which is the gap stage 2 closes.

Table 14: Face loss, stage 1, EMTD-30s.

Run	CSIM ↑	Sync-C ↑	Sync-D ↓	IQA ↑	ASE ↑
Adaptive weight	**0.744**	**7.16**	**8.29**	**4.79**	**3.62**
Without the face term	0.743	7.03	8.39	**4.79**	3.61
Without the adaptive weight	0.736	7.07	8.34	4.78	3.59

Dropping the term costs lip synchronization and little else. Sync-C falls from 7.16 to 7.03 and Sync-D rises from 8.29 to 8.39, the two largest effects of the five and in the same direction, while identity and per-frame quality stay where they were. Fixing the weight recovers only part of that gain, Sync-C 7.07 and Sync-D 8.34, and loses a little identity and per-frame quality that dropping the term does not, CSIM 0.736 against 0.743. The adaptive rule is ahead of it on all five metrics, and that is the comparison it has to win. Spending the same weight on a close-framed face as on a small one gives up half or more of the lip-sync gain.

F.2 Entering a gesture inside a chunk

Pose control is trained on partial spans, so a gesture may begin partway through a chunk rather than on its boundary. This measures what that buys at inference. Of the 50 EMTD-30s clips the other ablations run on, the 25 long enough for eight chunks are generated twice from one checkpoint with one seed, one schedule and one pose track, at 253 frames, or 10.1 seconds, each. No clip is left out for any other reason. The two runs differ only in where the pose track switches on. One starts it on the first latent of chunk four, so the change of conditioning coincides with the chunk boundary, where the memory is replaced as well. The other starts it on the last two latents of chunk three, so the model enters the gesture while the unposed context is still in view and the boundary stays on its own.

Figure 8 shows an entry of the second kind in a rollout of the released inference code. The track switches on over the last two latents of a chunk, and the hand rises from the rest pose, reaches the face and returns without a jump at either end of the gesture.

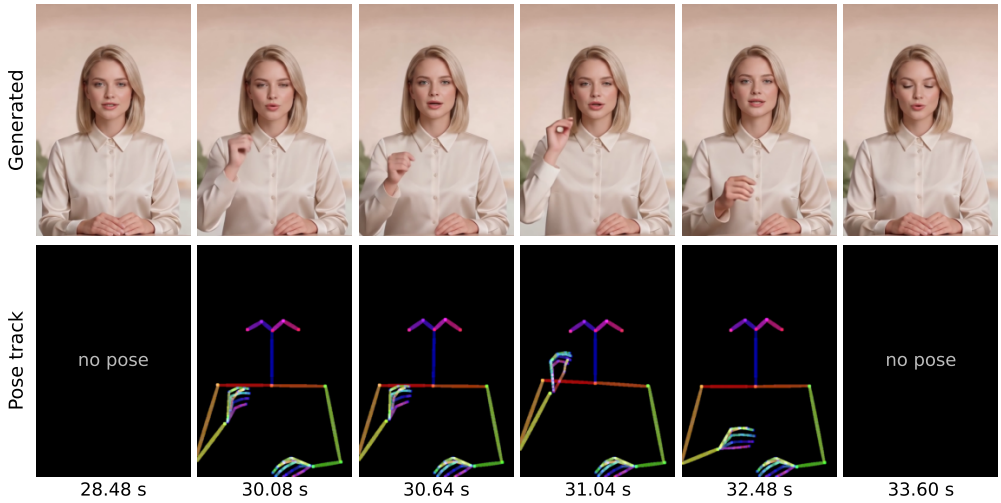


Figure 8: A gesture entered mid-stream. Top, six frames of a rollout. Bottom, the pose track fed at each frame, with the skeleton lines thickened for display. The track is zero before and after the gesture.

Writing d_t for the mean absolute difference between video frames t and $t+1$, Table 15 reports three views of the seam, all anchored at the same frame t^* in both runs, the first frame of chunk four. That frame is where the boundary-aligned entry switches the pose track on, and it is where the memory is replaced in both runs. The spike is d_{t^*} over the median of d on the sixteen neighbouring frames, the bend is the largest change of d at the seam over its median change across the clip, and the window ratio is the mean of d over the nine frames centred on t^* over its mean on the sixteen frames that precede that window. All three read one at a smooth transition. The mid-chunk entry switches its pose track on eight frames earlier, inside chunk three, so we also measure at that frame: it scores 1.16 on the spike and 1.5 on the bend there, against 1.98 and 12.4 for the boundary-aligned entry at its own switch. The jerk is not moved earlier, it does not appear.

Table 15: Motion at the frame where the gesture enters, EMTD-30s.

Pose enters	Spike ↓	Bend ↓	Window ratio ↓
On the chunk boundary	1.98	12.4	2.55
Inside the chunk	**1.24**	**3.1**	**1.31**

Entering on the boundary roughly doubles the motion of a single frame, which is the jerk a viewer sees when a gesture is requested. Entering inside the chunk leaves it near the level of the frames around it, 1.24 against 1.98 on the spike and 1.31 against 2.55 on the window ratio. Per clip the mid-chunk entry is the smoother one on 17 of the 25 on the spike, 21 on the bend and 22 on the window ratio. The gain costs no extra training and no second checkpoint, since both runs are the same model. It is the reason a request to gesture, which arrives between chunks rather than on a boundary of the speech, is fed on the last latents of the chunk it arrives in. What the seam looks like is easier to watch than to tabulate, and Appendix F.7 says where to do that.

F.3 Memory noise

The ablation is trained with the memory handed between chunks left clean, and is served with the renoise level set to zero so that inference matches its training. Everything else, including the 1300 steps and the schedule, follows the deployed recipe, and Table 16 reports the comparison.

Table 16: Memory noise, EMTD-30s.

Run	CSIM ↑	Sync-C ↑	Sync-D ↓	IQA ↑	ASE ↑
With memory noise	**0.859**	**7.65**	**8.02**	**4.85**	3.73
Without memory noise	0.825	7.59	8.07	**4.85**	**3.74**

The two runs are level on the no-reference scores and on lip synchronization, and they part on identity: 0.859 against 0.825. That is where a noised history is supposed to help, since the error a chunk inherits from its predecessor accumulates as a slow departure from the reference face rather than as a visible artifact within one chunk.

F.4 Cost of the deployment optimizations

The deployed stack trades a little quality for speed in three places: SpargeAttention in place of FlashAttention, FP8 row-wise linear layers in place of bfloat16, and the distilled Turbo-VAED decoder in place of the Hunyuan decoder the model was trained with. The two runs of Table 17 use the same checkpoint, the same 3-step schedule, the same seed and the same prompt, and differ only in those three choices.

Table 17: Deployment optimizations, EMTD-30s.

Run	CSIM ↑	Sync-C ↑	Sync-D ↓	IQA ↑	ASE ↑
FlashAttention, bfloat16, Hunyuan decoder	**0.864**	**7.75**	**7.99**	**4.86**	**3.75**
Deployed: SpargeAttention, FP8, Turbo-VAED	0.859	7.65	8.02	4.85	3.73

The three optimizations together cost 0.005 of identity and a tenth of a point of lip-sync confidence, and leave per-frame quality and aesthetics within a hundredth or two. The deployed row is the configuration every number reported for our model elsewhere in the paper comes from, so this is what the choice costs, not a correction to be applied to it.

F.5 Spectral progressive diffusion

The same checkpoint is served with the opening denoising step on a half-size grid, as introduced in Section 4.4, and carried to the full grid through the cosine transform. The transfer is a two-dimensional discrete cosine transform of the half-size latent, whose coefficients are written into the low-frequency corner of the larger array while the coefficients the larger grid adds are filled with Gaussian noise scaled by the current flow time t, followed by an inverse transform. Latent and time are rescaled by $\gamma = r/(1 + (r - 1)t)$ with $r = 2$ the ratio of the grids, which places the result back on the flow trajectory at time $\tilde{t} = \gamma t$. The delivered resolution is unchanged, so Table 18 compares the same output under a cheaper first step. It is run twice, once with the default decoder and once

with the x1.5 decoder of Section 4.4.1, since the cheaper step is independent of the resolution the system delivers.

Table 18: Spectral progressive diffusion, EMTD-30s. The best value within each decoder is in bold.

Run	CSIM ↑	Sync-C ↑	Sync-D ↓	IQA ↑	ASE ↑
Turbo-VAED	**0.859**	**7.65**	**8.02**	**4.85**	**3.73**
Turbo-VAED with SPD	0.830	7.51	8.05	4.84	**3.73**
Turbo-VAED-x1.5	**0.858**	**7.77**	**7.99**	**4.94**	3.95
Turbo-VAED-x1.5 with SPD	0.849	7.59	8.06	**4.94**	**3.96**

Quality per frame is untouched in both settings, 4.84 against 4.85 on IQA with the default decoder and identical at 4.94 with the x1.5 one, and lip synchronization loses little. Identity carries the cost, 0.830 against 0.859 and 0.849 against 0.858, which is consistent with what the half-size opening step discards: the low frequencies it fixes early are exactly where the face is placed and proportioned. The cost is smaller at the higher delivered resolution, where the decoder puts back detail the cheaper step never resolved. The saving it buys is in Table 2, a quarter off the denoiser at either resolution, which is why it is left off by default and switched on where the margin above real time matters more than the last hundredths of identity.

F.6 Distilled decoder against interpolation

The HD mode of Section 4.4.1 raises the delivered frame through a decoder trained to upscale by 12 rather than 8. That decoder is built from the base one rather than trained from scratch: its spatial upscale is split across the three upsampling stages as $3 \times 2 \times 2$ with the temporal upscale left at 4, the one stage whose stride grows from two to three is warm-started with each sub-pixel offset copying the trained offset nearest to it, and the result is fine-tuned on faces against the Hunyuan decoder as teacher at learning rate 10^{-4}, batch size 1, resolution 384 and 37 frames per clip. The cheap alternative is to decode as usual and stretch the result. Both are served from the same checkpoint, with the same schedule, seed and prompt, and both deliver 768×1152 from latents the model produces at 512×768. They differ only in the decoder, one being the fine-tuned Turbo-VAED-x1.5 decoder and the other the ordinary one followed by trilinear interpolation at a factor of 1.5 in height and width, and Table 19 compares them.

Table 19: Ways to reach 768×1152, EMTD-30s.

Decoder	CSIM ↑	Sync-C ↑	Sync-D ↓	IQA ↑	ASE ↑
Turbo-VAED-x1.5	0.858	**7.77**	**7.99**	**4.94**	**3.95**
Turbo-VAED with interpolation	**0.860**	7.72	**7.99**	4.77	3.58

The trained decoder buys what interpolation cannot, $+0.17$ on IQA and $+0.37$ on aesthetics, since stretching a decoded frame adds no detail. Identity and lip synchronization do not move.

F.7 What the released code lets a reader run

Two things this paper claims are easier to see than to tabulate, a gesture entering partway through a chunk and the mid-stream reset of Section 4.3. Both ship with the released inference code, which takes a pose track that starts or stops inside a chunk and has the reset as a switch, so a reader can drive our checkpoint with a pose of their own and watch the frames where the gesture enters, and can turn the reset on over a long rollout and compare it against the same rollout without it. Figure 8 was produced with that code and nothing else. The repository also carries generated examples of both, so neither has to be reproduced from scratch to be judged.