StableAvatar: Ultra-Long Audio-Driven Avatar Video Generation

Shuyuan Tu[1] **Yueming Pan**[3] **Yinming Huang**[1] **Xintong Han**[4] **Zhen Xing**[1] **Qi Dai**[2]
Chong Luo[2] **Zuxuan Wu**[1,*] **Yu-Gang Jiang**[1]
[1]Fudan University [2]Microsoft Research Asia [3]Xi'an Jiaotong University [4]Tencent Inc.
https://francis-rings.github.io/StableAvatar

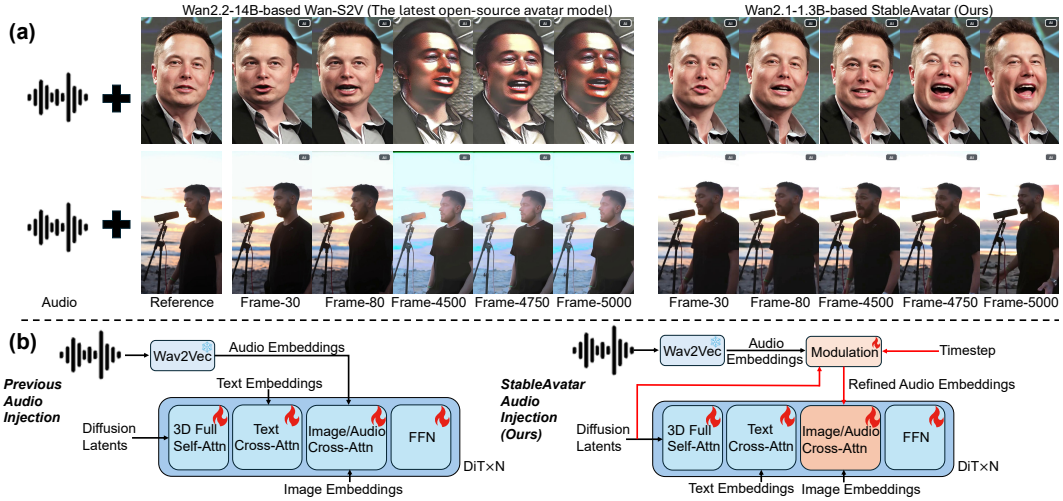

Figure 1: (a) Audio-driven avatar videos generated by StableAvatar, showing its power to synthesize ultra-long high-quality videos. The presented videos last over 3 minutes (FPS = 30). Frame-X refers to the X-th frame of the synthesized avatar video. (b) Audio injection framework comparison between previous methods and our StableAvatar. **We watermark synthesized frames to mitigate deepfake impersonation risks.**

Abstract

Current diffusion models for audio-driven avatar video generation struggle to synthesize long videos with natural audio synchronization and identity consistency. This paper presents StableAvatar, the first end-to-end video diffusion transformer that synthesizes ultra-long videos. We observe that the primary reason preventing existing models from generating long videos is their audio modeling, which typically relies on third-party off-the-shelf audio embeddings that are introduced into the diffusion model via cross-attention. Since current diffusion backbones lack any audio-related priors, they cannot correctly couple audio cues with latent dynamics, causing systematic latent distribution deviations at each segment. These deviations accumulate across segments, gradually shifting the latent trajectory and producing noticeable quality drift. To address this, StableAvatar introduces a novel Timestep-aware Audio Adapter that prevents error accumulation via timestep-aware modulation. During inference, we propose a novel Audio Native Guidance Mechanism to further enhance the audio synchronization by leveraging the diffusion's joint audio-latent prediction as a dynamic guidance signal. To enhance the smoothness of the ultra-long videos, we introduce a Dynamic Weighted Sliding-window Strategy that fuses latent over time. Experiments on benchmarks show the effectiveness of StableAvatar both qualitatively and quantitatively.

* Corresponding Authors.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

1 Introduction

Audio-driven avatar video generation aims to synthesize natural human-centric videos with synchronized facial expressions and body movements based on a reference image and audio, offering diverse applications in film production and virtual assistants. Diffusion models [13, 23, 24, 58, 56, 51, 44, 20, 71, 80, 7, 62, 64, 69, 70] have significantly advanced this line of research [87, 79, 75, 46, 11, 77, 6, 33, 15, 30]. However, they are limited to generating short avatar videos of less than 15 seconds. When they attempt to generate longer videos, significant appearance distortions and inconsistencies occur. This severely restricts their practical applications.

To address this issue, several methods have explored consistency preservation for avatar video generation [30, 11, 33, 15, 45, 84, 16, 27, 66]. These strategies—whether leveraging motion frames [83] or adopting various shifted-window mechanisms during inference—only partially improve the smoothness of long videos and fail to fundamentally mitigate quality degradation in ultra-long avatar videos. The concurrent work InfiniteTalk [84] relatively alleviates the quality degradation but suffers from abrupt flickering/blurry issues. The root reason is error accumulation, including both errors within individual clips and their propagation across clips. While above methods alleviate error propagation in clip-to-clip transitions to some extent, they largely overlook the error accumulation inside each clip, which primarily stems from audio modeling (Fig. 1). Audio patterns are significantly difficult to model for backbones lacking audio priors than image or text, leading to unstable learning and excessive deviation of latents from the backbone prior. As latents move into unfamiliar regions, the model's robustness degrades, making it hard to gain audio–latent mappings and causing the distribution to drift away from the desired audio-aligned space, exacerbating error accumulation.

In light of this, we propose StableAvatar, consisting of dedicated modules for both training and inference to maintain audio synchronization for audio-driven high-quality and ultra-long avatar video generation, as shown in Fig. 2 and Fig. 1 (b). We first observe that the primary limitation preventing previous models from synthesizing ultra-long videos lies in their audio modeling. They simply use a third-party off-the-shelf extractor [1] to obtain audio embeddings, and then directly inject them into a Video Diffusion Transformer via cross-attention. As current diffusion backbones lack any audio-related priors, their latents cannot accurately follow the audio-driven dynamics. This mismatch introduces systematic deviations in the latent space at each clip, and these deviations accumulate over successive video clips. As a result, the latents gradually diverge from the desired audio-aligned distribution, ultimately causing temporal quality drift. To tackle this, StableAvatar feeds audio embeddings to a Video Diffusion Transformer with a novel Timestep-aware Audio Adapter that dramatically reduces error accumulation over clips. In particular, initial audio embeddings (Query) sequentially cross-attend with initial patchified latents (Key and Value), followed by affine modulation with timestep embeddings to obtain the refined audio embeddings. As timestep embeddings are strongly correlated with latents, this potentially forces the diffusion to model the joint audio-latent feature distribution at each timestep, effectively mitigating latent distribution error accumulation due to the lack of audio priors (Appx.A.2). Following [33, 15], the refined audio embeddings (Key and Value) are fed to the diffusion model via cross-attention with latents (Query).

During inference, to further enhance the audio synchronization and facial expression, StableAvatar introduces a novel Audio Native Guidance Mechanism to replace the conventional Classifier-Free-Guidance (CFG) [25]. As the refined audio embeddings are also inherently dependent on the latents, rather than solely relying on external audio signals, our guidance directly manipulates the diffusion's sampling distribution, steering the generation towards the joint audio-latent distribution and enabling the diffusion model to refine its outputs throughout the denoising process. StableAvatar further proposes a dynamic weighted sliding-window denoising strategy that fuses latents over time to improve the smoothness and reduce error propagation in long avatar video generation.

As shown in Fig. 1 (a) and Fig. 5, while the latest audio-driven avatar video generation model Wan-S2V [16] suffers from dramatic face distortion and color drift, StableAvatar accurately animates the reference with natural lip synchronization driven by audio while preserving the overall consistency, even in the ultra-long video generation scenario (3500+ frames in a single pass). While the quantitative gap between ours and InfiniteTalk in Fig. 5 is small, the qualitative gap is large in Fig. 3, as it suffers from abrupt flickering/blurry in long videos.

In conclusion, our contributions are as follows: (1) We propose a novel Timestep-aware Audio Adapter to force the diffusion model to capture joint audio-latent features, thereby significantly

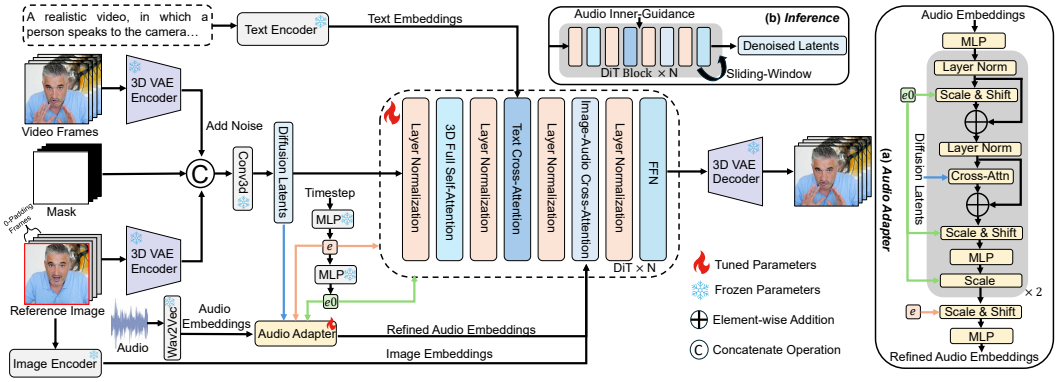

Figure 2: Architecture of StableAvatar. (a) refers to the structure of the Audio Adapter. Embeddings from the Image Encoder and Text Encoder are injected into each block of DiT. Given the audio, we extract the audio embeddings utilizing Wav2Vec. To model joint audio-latent representations, the audio embeddings are fed into the Audio Adapter, and its outputs are fed to the DiT via cross-attention.

reducing latent distribution error accumulation during audio injection, making StableAvatar the first video diffusion transformer that generates ultra-long smooth audio-driven avatar video. (2) A novel Audio Native Guidance Mechanism replaces the conventional CFG to further enhance audio synchronization. At the same time, a dynamic weighted sliding-window denoising strategy improves the smoothness of synthesized long videos. (3) Experiments on benchmarks show the superiority of StableAvatar over the SOTA. Notably, based on a Wan2.1-1.3B model [74], we surpass previous Wan2.1-14B-based models in video quality for long-video generation.

2 Related Work

Video Generation. The capacity for diversity and high fidelity in diffusion models [13, 23, 24, 63, 65, 47, 58, 56, 51, 44, 20, 71, 35, 85, 28] has sparked significant interest in their potential for video generation. Early video diffusion models [54, 3, 17, 82, 78, 5] primarily leverage the U-Net architecture for video generation by adding temporal layers to pre-trained image diffusion models. Recent works [2, 26, 32, 74] replace the U-Net with the Diffusion-in-Transformer (DiT) architecture [48] for stronger scalability. Inspired by previous works [77, 33, 15], we utilize Wan2.1 [74] as the backbone.

Audio-Driven Avatar Video Generation. Early works [90, 87, 9, 75] first convert audio into features [60, 57], and then use a motion-to-video model projects these features into videos. Recently, some works [59, 79, 30, 8, 36, 31, 68, 67] leverage the diffusion model to integrate audio cues with lips, which only supports head movement. CyberHost [37], FantasyTalking [77] and OmniAvatar [15] aim to animate half/full-body avatars. OmniHuman [38] applies a mixed data training strategy for scaling up. HunyuanVideo-Avatar [6] and MultiTalk [33] aim to synthesize multi-character animation. EchomimicV3 [45] unifies multi-task human animation. InfiniteTalk [84] alleviates the color drift in the long videos via context frames. Wan-SV2 [16] excels in prompt following. However, previous works suffer from face distortion and color drift, particularly when generating long avatar videos. While some works [11, 30, 33, 15, 83, 84] propose strategies for long video generation, their approaches primarily improve smoothness, without addressing the root cause of distortion—latent error accumulation. StableAvatar addresses these issues in long avatar video generation.

Long Video Generation. Generating long videos [55, 4, 18, 73, 61] is extremely challenging due to temporal complexity. Nuwa-XL [86] and StreamingT2V [19] both adopt short-long memory blocks to ensure consistency. Gen-L-Video [76] and FreeNoise [49] extend videos by merging overlapping sub-segments with a sliding window. FreeLong [41] and FreeLong++ [42] blend global and local video features for consistency. However, audio-driven video generation cannot directly apply the above methods, as they are designed for Text-to-Video and exhibit a domain gap with audio.

3 Method

Shown in Fig. 2, Wan2.1 [74]-based StableAvatar is capable of synthesizing ultra-long audio-driven videos. Concretely, the audio is first fed to Wav2Vec [1] to gain audio embeddings, which are

subsequently refined for reducing latent distribution error accumulation via our Audio Adapter. The refined audio embeddings are then fed to the denoising DiT. More details are described in Sec. 3.1. Following [74], a reference image is processed through the DiT via two pathways (a frozen 3D VAE Encoder [74] and a CLIP Image Encoder [50]).

We replace the original input video frames with random noise during inference, while the other inputs stay the same. We propose a novel Audio Native Guidance to replace conventional CFG [25] for further facilitating lip synchronization and facial expression, as detailed in Sec. 3.2. We introduce a dynamic weighted sliding-window denoising strategy that fuses latent over time to enhance the long video smoothness, as detailed in Sec. 3.3.

3.1 Timestep-aware Audio Adapter

Current diffusion backbones [3, 74, 32] lack audio-related priors. Thus, audio patterns are inherently more difficult to model than image or text modalities. Directly injecting off-the-shelf audio embeddings into the diffusion process often leads to unstable learning and severe shifts of the latent distribution away from the backbone's original prior, causing significant latent distribution errors to accumulate across video clips (Appx.A.2). This results in the latent distribution of subsequent segments gradually drifting away from the desired audio-aligned distribution. To address this, we propose a novel Timestep-aware Audio Adapter, in which the audio embeddings go through affine modulations and cross-attention blocks to interact with timestep embeddings and latents (Fig. 2).

Concretely, following [59, 33], we first use Wav2Vec [1] to extract the raw audio embeddings $\boldsymbol{a}$. As the current state is influenced by preceding and succeeding audio frames, we then concatenate them proximal to the current frames, gaining audio embeddings $\boldsymbol{emb}_{aud}$:

$$\boldsymbol{emb}^{i}_{aud} = \texttt{Concat}(\boldsymbol{a}^{i-k}, ..., \boldsymbol{a}^{i}, ..., \boldsymbol{a}^{i+k}), \tag{1}$$

where $2k+1$ is the context length. We further feed $\boldsymbol{emb}_{aud}$ to our Audio Adapter for addressing the error accumulation issue. Given a timestep, following [74], the DiT uses two successive MLP layers to obtain overall timestep embeddings $\boldsymbol{e} \in \mathcal{R}^{1\times D}$ and projected timestep embeddings $\boldsymbol{e0} \in \mathcal{R}^{6\times D}$. D refers to the latent dimension. In diffusion pretraining, latents are tightly coupled with the timestep embeddings. Each timestep embedding corresponds to a distinct latent distribution, revealing a strong correlation between the latents and the timestep embeddings. Due to this strong correlation, applying timestep-aware affine modulations to $\boldsymbol{emb}_{aud}$ can implicitly bridge the joint relationship between $\boldsymbol{emb}_{aud}$ and latents $\boldsymbol{z}_i$, enabling the diffusion model to more effectively capture joint audio-latent features, thereby overcoming the scarcity of audio priors. The modulations (scale and shift) are the same as those in the DiT for domain consistency:

$$\begin{aligned}
\boldsymbol{emb}^{\lambda}_{aud} &= \texttt{MLP}(\boldsymbol{emb}_{aud}), \\
\boldsymbol{emb}^{\gamma}_{aud} &= \boldsymbol{e0}[2](\texttt{LN}(\boldsymbol{emb}^{\lambda}_{aud})(1+\boldsymbol{e0}[1]) + \boldsymbol{e0}[0]) + \boldsymbol{emb}^{\lambda}_{aud},
\end{aligned} \tag{2}$$

where $\texttt{LN}(\cdot)$ is Layer Norm. $\texttt{MLP}(\cdot)$ projects $\boldsymbol{emb}_{aud}$ onto the latent dimension. To further explicitly enhance joint audio-latent modeling, we perform cross-attention between $\boldsymbol{emb}^{\gamma}_{aud}$ (Query) and $\boldsymbol{z}_t$ (Key and Value), with the outputs modulated by $\boldsymbol{e0}$ as follows:

$$\begin{aligned}
\boldsymbol{emb}^{'}_{aud} &= \texttt{CAttn}(\texttt{LN}(\boldsymbol{emb}^{\gamma}_{aud}), \boldsymbol{z}_t) + \texttt{LN}(\boldsymbol{emb}^{\gamma}_{aud}), \\
\boldsymbol{emb}^{\eta}_{aud} &= \texttt{MLP}(\texttt{LN}(\boldsymbol{emb}^{'}_{aud}) * (1+\boldsymbol{e0}[4]) + \boldsymbol{e0}[3]), \\
\boldsymbol{emb}^{*}_{aud} &= \boldsymbol{emb}^{'}_{aud} + \boldsymbol{e0}[5] * \boldsymbol{emb}^{\eta}_{aud},
\end{aligned} \tag{3}$$

where $\texttt{CAttn}(\boldsymbol{x}, \boldsymbol{y})$ refers to cross-attention operation, in which $\boldsymbol{x}$ is the Query and $\boldsymbol{y}$ is the Value and Key. To more comprehensively establish the joint relationship between latents and audio representations, $\boldsymbol{emb}^{*}_{aud}$ is modulated by $\boldsymbol{e}$ to gain the refined audio embeddings $\bar{\boldsymbol{a}}_t$:

$$\begin{aligned}
\boldsymbol{e} &= \texttt{Repeat}(\boldsymbol{e}) + \boldsymbol{r}, \\
\bar{\boldsymbol{a}}_t &= \texttt{MLP}(\boldsymbol{emb}^{*}_{aud} * (1+\boldsymbol{e}[1]) + \boldsymbol{e}[0]),
\end{aligned} \tag{4}$$

where $\texttt{Repeat}(\cdot)$ and $\boldsymbol{r}$ refer to duplicating along the channel dimension twice and learnable parameters. We ultimately inject $\bar{\boldsymbol{a}}_t$ into the DiT via cross attention:

$$\bar{\boldsymbol{z}}_t = \texttt{CAttn}(\boldsymbol{z}_t, \bar{\boldsymbol{a}}_t) + \texttt{CAttn}(\boldsymbol{z}_t, \boldsymbol{emb}_{img}), \tag{5}$$

where $\boldsymbol{emb}_{img}$ is the image embeddings. $\bar{\boldsymbol{a}}_t$, conditioned on the latents and the audio, can be viewed as an internal variable predicted by the DiT. $\bar{\boldsymbol{a}}_t$ encodes rich mappings of the joint distribution between audio and latents, enabling the diffusion to learn complex audio patterns without aggressively perturbing the prior latent distribution, thereby mitigating error accumulation from the root.

3.2 Audio Native Guidance

To further enhance audio synchronization and facial expression, we propose a novel Audio Native Guidance mechanism to replace the conventional CFG [25], which does not take the joint audio-latent relationship into consideration. We modify the denoising score function [58] to steer the denoising process forward in a way that maximizes both audio synchronization and naturalness. In particular, according to our Audio Adapter, $\bar{\boldsymbol{a}}_t$ depends on the latents and the given audio. Thus, we treat $\bar{\boldsymbol{a}}_t$ as an additional prediction of the DiT, guiding the diffusion model to capture the joint audio-latent distribution, conditioned on the external signal and model parameters. The target probability density:

$$\boldsymbol{p}([\boldsymbol{z}_t, \bar{\boldsymbol{a}}_t]|\boldsymbol{A}) \cdot \boldsymbol{p}(\boldsymbol{A}|[\boldsymbol{z}_t, \bar{\boldsymbol{a}}_t])^{\alpha} \cdot \boldsymbol{p}(\bar{\boldsymbol{a}}_t|\boldsymbol{z}_t, \boldsymbol{A})^{\beta}, \tag{6}$$

where $\boldsymbol{A}$ and $[\cdot]$ refer to the audio signal and internal variables in the diffusion model. $\boldsymbol{p}([\boldsymbol{z}_t, \bar{\boldsymbol{a}}_t]|\boldsymbol{A})$ is the base sampling distribution. $\boldsymbol{p}(\boldsymbol{A}|[\boldsymbol{z}_t, \bar{\boldsymbol{a}}_t])^{\alpha}$ and $\boldsymbol{p}(\bar{\boldsymbol{a}}_t|\boldsymbol{z}_t, \boldsymbol{A})^{\beta}$ are the likelihood estimation of audio based on the joint prediction and inner audio embeddings. α and β are guidance scales. Following Bayes' Theorem, we obtain:

$$\begin{aligned} &\boldsymbol{p}([\boldsymbol{z}_t, \bar{\boldsymbol{a}}_t]|\boldsymbol{A})\left(\frac{\boldsymbol{p}([\boldsymbol{z}_t, \bar{\boldsymbol{a}}_t], \boldsymbol{A})}{\boldsymbol{p}(\boldsymbol{z}_t, \bar{\boldsymbol{a}}_t)}\right)^{\alpha}\left(\frac{\boldsymbol{p}([\boldsymbol{z}_t, \bar{\boldsymbol{a}}_t], \boldsymbol{A})}{\boldsymbol{p}(\boldsymbol{z}_t, \boldsymbol{A})}\right)^{\beta} \\ \rightarrow \quad &\boldsymbol{p}([\boldsymbol{z}_t, \bar{\boldsymbol{a}}_t]|\boldsymbol{A})\left(\frac{\boldsymbol{p}([\boldsymbol{z}_t, \bar{\boldsymbol{a}}_t]|\boldsymbol{A})\boldsymbol{p}(\boldsymbol{A})}{\boldsymbol{p}(\boldsymbol{z}_t, \bar{\boldsymbol{a}}_t)}\right)^{\alpha}\left(\frac{\boldsymbol{p}([\boldsymbol{z}_t, \bar{\boldsymbol{a}}_t]|\boldsymbol{A})}{\boldsymbol{p}(\boldsymbol{z}_t|\boldsymbol{A})}\right)^{\beta}, \end{aligned} \tag{7}$$

As $\boldsymbol{p}(\boldsymbol{A})$ is a constant probability, we remove it as follows:

$$\boldsymbol{p}([\boldsymbol{z}_t, \bar{\boldsymbol{a}}_t]|\boldsymbol{A})\left(\frac{\boldsymbol{p}([\boldsymbol{z}_t, \bar{\boldsymbol{a}}_t]|\boldsymbol{A})}{\boldsymbol{p}(\boldsymbol{z}_t, \bar{\boldsymbol{a}}_t)}\right)^{\alpha}\left(\frac{\boldsymbol{p}([\boldsymbol{z}_t, \bar{\boldsymbol{a}}_t]|\boldsymbol{A})}{\boldsymbol{p}(\boldsymbol{z}_t|\boldsymbol{A})}\right)^{\beta}, \tag{8}$$

We further convert Eq. 8 to the score function format (∇ denotes $\nabla_{\boldsymbol{z}_t}$, not $\nabla_{\bar{\boldsymbol{a}}_t}$):

$$\begin{aligned} &(1+\alpha+\beta)\nabla \log \boldsymbol{p}([\boldsymbol{z}_t, \bar{\boldsymbol{a}}_t]|\boldsymbol{A}) \\ &-\alpha\nabla \log \boldsymbol{p}([\boldsymbol{z}_t, \bar{\boldsymbol{a}}_t]) - \beta\nabla \log \boldsymbol{p}(\boldsymbol{z}_t|\boldsymbol{A}), \end{aligned} \tag{9}$$

Thus, the inference formulation can be described as:

$$\begin{aligned} &(1+\alpha+\beta)\boldsymbol{D}([\boldsymbol{z}_t, \bar{\boldsymbol{a}}_t], \boldsymbol{y}, \boldsymbol{I}, \boldsymbol{A}; \theta) \\ &-\alpha\boldsymbol{D}([\boldsymbol{z}_t, \bar{\boldsymbol{a}}_t], \boldsymbol{y}, \boldsymbol{I}, \emptyset; \theta) - \beta\boldsymbol{D}([\boldsymbol{z}_t, \emptyset], \boldsymbol{y}, \boldsymbol{I}, \boldsymbol{A}; \theta), \end{aligned} \tag{10}$$

where $\boldsymbol{D}(\cdot;\theta)$, $\boldsymbol{y}$, and $\boldsymbol{I}$ refer to the diffusion model, text prompt, and reference image. Notably, we find that incorporating $\boldsymbol{I}$ and $\boldsymbol{y}$ into the guidance substantially increases GPU cost and does not significantly improve visual quality (Appx.A.10). Thus, we do not use $\boldsymbol{I}$ and $\boldsymbol{y}$ as guidance factors. The Audio Native Guidance mechanism regards $\bar{\boldsymbol{a}}_t$ as an additional prediction target for the diffusion model, allowing the model to be guided by the joint audio-latent distribution, thereby ensuring that audio and latents are strongly correlated during the denoising process. This reduces distribution error accumulation in audio-driven video generation, even when the base model lacks audio priors. More proof and discussion are depicted in Appx. A.3.

3.3 Dynamic Weighted Sliding-Window Strategy

To improve the smoothness of long videos, we further propose a Dynamic Weighted Sliding-Window Strategy (DWSW) during inference. Compared with the previous sliding-window strategy [30], overlapping latents between windows are fused using a sliding window mechanism, where the fusion weights follow a logarithmic interpolation based on the relative frame indices, as described in the Algorithm 1 and Appx.A.7. L is the VAE-compressed total video length. We set $m = 10$. The fused latents are injected back into both adjacent windows, ensuring that both boundaries of the central window consist of blended features. The logarithmic weighting introduces a progressive smoothing effect in the transitions between video clips. Early stages experience a more pronounced weight variation, while later stages exhibit subtle shifts, resulting in seamless continuity across video clips.

3.4 Training

We use the reconstruction loss to train our model, with trainable components including attention modules of a DiT and an Audio Adapter. We introduce face masks $\boldsymbol{M}_{face}$ and lip masks $\boldsymbol{M}_{lip}$, extracted by Mediapipe [43] from the input video frames:

$$\mathcal{L} = \begin{cases} \mathbb{E}_{\boldsymbol{z}_0, \boldsymbol{z}_1, t}(\|((\boldsymbol{z}_1 - \boldsymbol{z}_0) - \boldsymbol{\varepsilon}_\theta(\boldsymbol{z}_t, t)) \odot \boldsymbol{M}_{face}\|^2), & 0.4 \leq \boldsymbol{q} < 0.5 \\ \mathbb{E}_{\boldsymbol{z}_0, \boldsymbol{z}_1, t}(\|((\boldsymbol{z}_1 - \boldsymbol{z}_0) - \boldsymbol{\varepsilon}_\theta(\boldsymbol{z}_t, t)) \odot \boldsymbol{M}_{lip}\|^2), & \boldsymbol{q} \geq 0.5 \\ \mathbb{E}_{\boldsymbol{z}_0, \boldsymbol{z}_1, t}(\|((\boldsymbol{z}_1 - \boldsymbol{z}_0) - \boldsymbol{\varepsilon}_\theta(\boldsymbol{z}_t, t)) \odot (1 + \boldsymbol{M}_{face} + \boldsymbol{M}_{lip})\|^2), & \text{otherwise} \end{cases} \tag{11}$$

Table 1: Quantitative comparisons on HDTF/AVSpeech/Long100. In the table elements a / b / c, a, b, and c refer to the result on the HDTF, AVSpeech, and Long100, respectively. The average video duration of HDTF and AVSpeech is 10 seconds, while Long100 is 3 minutes.

Model	FID↓	FVD↓	CSIM↑	Sync-C↑	Sync-D↓	IQA↑	ASE↑
SadTalker [87]	53.12/120.57/194.88	567/1468/2261	0.821/0.802/0.423	7.25/4.13/3.21	9.86/9.72/11.16	3.12/2.40/2.31	2.14/1.43/1.38
Sonic [30]	62.17/187.42/278.40	552/2051/2877	0.843/0.825/0.451	8.16/6.04/4.03	7.65/8.94/10.23	3.28/2.24/2.08	2.05/1.36/1.13
Hallo3 [11]	44.31/98.14/170.44	438/987/1724	0.845/0.834/0.462	6.58/5.16/4.42	8.64/9.62/9.92	3.50/3.38/2.35	2.11/1.96/1.36
FantasyTalking [77]	46.74/80.01/175.78	479/823/1789	0.863/0.853/0.468	3.42/2.98/1.92	12.15/11.42/11.78	3.55/3.24/2.42	2.28/1.89/1.48
HunyuanAvatar [6]	52.16/77.53/172.94	625/868/1743	0.866/0.859/0.472	7.20/6.74/4.34	8.39/8.28/10.07	3.56/3.63/2.46	2.25/2.21/1.52
MultiTalk [33]	46.94/75.67/175.52	446/804/1768	0.868/0.861/0.465	7.53/4.88/4.12	8.02/9.59/10.18	3.54/3.72/2.42	2.15/2.25/1.45
OmniAvatar [15]	41.79/72.56/168.49	424/744/1621	0.862/0.857/0.471	7.50/6.78/4.45	8.26/8.05/9.62	3.55/3.74/2.51	2.27/2.29/1.56
EchomimicV3 [45]	60.34/106.31/172.18	590/1108/1852	0.852/0.840/0.458	5.48/4.62/3.71	9.33/9.78/10.24	3.66/3.04/2.38	2.32/1.85/1.39
InfiniteTalk [84]	40.12/69.48/68.95	390/652/569	0.869/0.865/0.755	7.97/7.23/7.92	7.34/8.15/7.05	3.76/3.77/3.65	2.38/2.28/2.26
Wan-S2V [16]	**35.87/66.43**/164.42	**368/625**/1685	0.872/0.869/0.477	8.12/7.48/4.88	6.98/7.92/9.68	3.83/**3.81**/2.47	2.42/2.30/1.53
Ours	38.14/68.12/**57.18**	375/640/**504**	**0.875/0.872/0.849**	**8.15/7.56/8.24**	**6.94/7.85/6.79**	**3.90**/3.79/**3.84**	**2.46/2.32/2.39**

where $\boldsymbol{z}_1 - \boldsymbol{z}_0$ is the ground-truth velocity and $\boldsymbol{\varepsilon}_\theta$ is the predicted velocity (Appx.A.1). $\boldsymbol{q}$ is a random variable uniformly distributed over $[0, 1]$. This piecewise objective separately supervises lip synchronization and facial expression, enabling more targeted learning (Appx.A.5).

4 Experiments

4.1 Implementation Details

Our training dataset is composed of three parts: Hallo3 [11], Celebv-HQ [91], and videos collected from the internet, totaling 1200 hours of video. Following previous works [77, 11, 33, 15], we evaluate our model on HDTF [89] and semi-body AVSpeech [14]. Since previous works do not open-source their testing datasets, we randomly select 100 videos (5-20 seconds long) from HDTF and AVSpeech, respectively. We conduct additional experiments on 100 unseen videos (2-5 minutes long), referred to the Long100, selected from the internet to assess the robustness of our model during long avatar animation. Our DiT uses pre-trained weights of Wan2.1-I2V 1.3B [74, 40], while the Audio Adapter is trained from scratch. Our model is trained for 20 epochs on 64 NVIDIA A100, with a batch size of 1 per GPU. We set learning rate$= 1e-5$, $\alpha = 4.5$, and $\beta = 3.0$ (Appx.A.10).

4.2 Comparison with State-of-the-Art Methods

Quantitative results. Regarding metrics, we use FID [21] and FVD [72] to assess the quality of images and videos. We further use the Q-align model [81] to evaluate the video quality (IQA) and aesthetic metrics (ASE). Sync-C [10] and Sync-D [10] are utilized to assess the synchronization of lips with audio. CSIM [87] evaluates the cosine similarity between the facial embeddings of two images. We compare with recent audio-driven avatar video generation models, including GAN-based models (SadTalker [87]) and diffusion-based models (SVD-based: Sonic [30]; CogVideo-5B-based: Hallo3 [11]; HunyuanVideo-13B-based: HunyuanAvatar [6]; Wan-14B-based: FantasyTalking [77], MultiTalk [33], OmniAvatar [15], InfiniteTalk [84], Wan-S2V [16]; Wan-1.3B-based: EchomimicV3 [45]). Based on previous studies that assess quantitative results using the self-driven and reconstruction approach, we perform quantitative comparisons with the above competitors on HDTF [89], AVSpeech [14], and Long100. The results are shown in Table 1. We observe that, even though all competitors encounter a significant drop in performance for long video generation, StableAvatar still surpasses them regarding face quality, video fidelity, and lip synchronization while maintaining relatively high quality in the short video generation. Wan2.1-1.3B-based StableAvatar outperforms the leading competitor, Wan2.2-14B-based Wan-S2V, by 78.0% and 68.9% in CSIM and Sync-C on Long100, even though Wan-S2V is trained on a larger dataset. More results are in Appx. A.12.

Qualitative Results. The qualitative results are shown in Fig. 3. Notably, each audio lasts 3+ minutes and is filled with intricate rhythm patterns, while the references include intricate details of appearances. We only display selected frames from the last 2 minutes for brevity. Most competitors can accurately modify the reference lip movements within the first 15 seconds of the video. However, when the video duration exceeds 30 seconds, most competitors suffer from audio-lip synchronization issues, blurry noises, face distortion, and color drift. In particular, Hallo3 and HunyuanAvatar suffer from face distortion and audio-lip synchronization issues, with the lips moving randomly. Meanwhile, FantasyTalking, MultiTalk, OmniAvatar, and Wan-S2V struggle with color drift, body/face distortion, and audio-lip synchronization issues. Notably, while InfiniteTalk tackles the color drift issue and

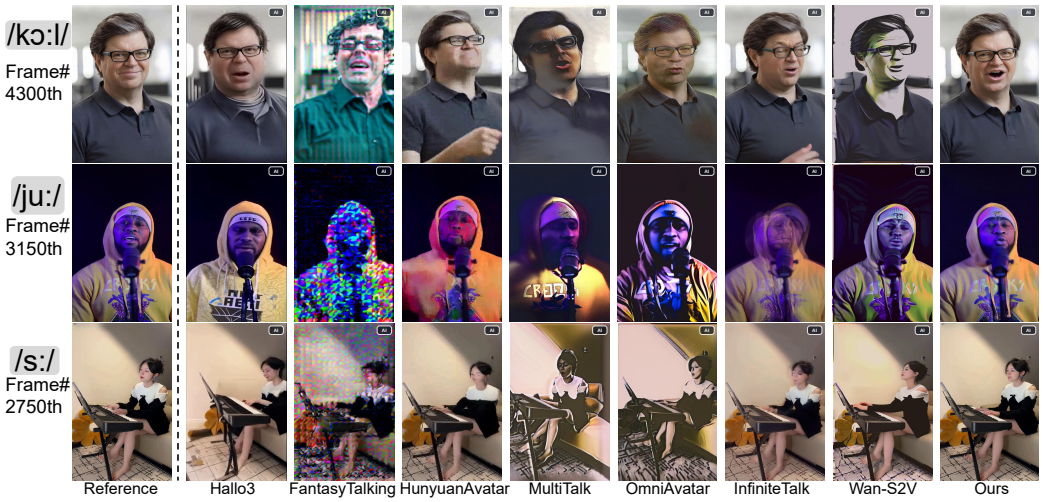

Figure 3: Qualitative comparisons with state-of-the-art methods. More examples are in Appx.A.12.



Figure 4: Ablations on core components of StableAvatar.

Table 2: Ablation study on core components. MF and SW are motion frame [11] and conventional sliding-window [30], which are both based on *w/o* Audio Adapter.

Method	FVD↓	CSIM↑	Sync-C↑	Sync-D↓	IQA↑	ASE↑
w/o Audio Adapter	1802	0.457	3.95	10.96	2.34	1.40
w/o Guidance	866	0.822	7.48	8.36	3.74	2.31
w/o DWSW	718	0.845	8.17	6.85	3.79	2.36
w/ MF [11]	2043	0.402	3.69	10.82	2.28	1.32
w/ SW [30]	1854	0.438	3.77	10.72	2.31	1.35
Ours	**504**	**0.849**	**8.24**	**6.79**	**3.84**	**2.39**

shows a small gap with ours in Table 1, it struggles with abrupt flickering and blurriness in several frames. In contrast, our StableAvatar accurately animates images based on the given audio while preserving ID even after generating 3500+ frames, highlighting the superiority of our model in generating vivid, ultra-long videos.

Quality Drift. Fig. 5 shows that the quality drift in our StableAvatar remains negligible as the frame count increases, especially when compared to previous models. To rigorously validate this, we conduct a performance comparison on extremely long videos (hours) in Appx. A.18, including VBench [29] temporal analysis and a human evaluation. Regarding the ID drift in long avatar video generation, we also show that additional explicit ID encoders provide only marginal gains on StableAvatar at high GPU cost, while failing without our Audio Adapter (Appx. A.19).

4.3 Ablation Study

Audio Adapter. We conduct an ablation study to demonstrate the contributions of core components in StableAvatar, as shown in Table 2 and Fig. 4. Notably, all quantitative ablation studies are on the Long100 dataset. We can see that removing the core components significantly degrades performance, particularly in CSIM and Sync-C/D, highlighting that our components significantly enhance both video fidelity while preserving ID consistency in long video generation. In contrast, previous long

Table 3: Ablation on audio modeling. *w/o* Random modulation and *w/o* CAttn refer to replace timesteps with random learnable parameters and remove cross-attention in our Audio Adapter.

Method	FVD↓	CSIM↑	Sync-C↑	Sync-D↓	IQA↑	ASE↑
w/o Audio Adapter	1802	0.457	3.95	10.96	2.34	1.40
w/o Modulation	1340	0.637	5.25	9.48	3.39	1.98
w/ Random modulation	1186	0.632	5.16	9.76	3.50	2.07
w/o CAttn	1218	0.664	6.37	8.52	3.56	2.23
Ours	**504**	**0.849**	**8.24**	**6.79**	**3.84**	**2.39**

Table 4: Ablation on the error accumulation. Adapter/G are Audio Adapter/Audio Native Guidance.

Method	FVD↓	CSIM↑	Sync-C↑	Sync-D↓	CIEDE↓
Baseline(A)	865	0.826	7.66	7.82	0.536
Baseline(B)	2388	0.405	3.78	10.47	2.318
w/ Adapter(A)	723	0.829	8.23	6.74	0.196
w/ Adapter(B)	912	0.818	7.95	6.92	0.831
w/ Adapter+G(A)	478	0.853	8.28	6.65	0.166
w/ Adapter+G(B)	572	0.846	8.20	6.83	0.523

video generation strategies (*w/* MF [11] and *w/* SW [30]) still suffer from appearance inconsistency and color drift, as they only basically tackle the video smoothness issue.

We further conduct an ablation study regarding audio modeling, as shown in Table 3 and Fig. 7 (a). By analyzing the results, we can gain the following observations: (1) *w/o* Audio Adapter dramatically degrades the video fidelity and lip synchronization. The plausible reason is that current diffusion backbones lack audio-related priors, and directly injecting audio embeddings from third-party off-the-shelf extractors into diffusion leads to significant latent error accumulation across video clips, gradually deteriorating the overall long video quality. (2) *w/o* Modulation/CAttn both relatively degrade the video quality. The underlying reason is that timestep-aware modulation bridges the joint audio-latent modeling, as latents and timesteps are strongly correlated. CAttn explicitly introduces latents into audio modeling, but without timestep modulation for audio embeddings, making it challenging for the model to effectively model the joint latent-audio space. Thus, timestep-aware modulation and CAttn are complementary, which is also evidenced by *w/* Random modulation results. (3) The full model's large improvement over rows 2–4 arises from synergy rather than additive effects. Modulation alone moves audio embeddings toward the latent manifold but lacks explicit interaction, while CAttn operates on raw embeddings far from the latent space, limiting alignment. When combined, modulation bridges the domain gap, enabling CAttn to learn fine-grained audio–latent correspondences. Evaluated on Long100 (3 min+), this per-clip synergy is amplified over thousands of frames due to cumulative errors. (4) StableAvatar can significantly refine the face quality while maintaining high video fidelity in long video generation since our model enables joint audio-latent modeling, reducing latent distribution error accumulation across video clips.

Error Accumulation. We conduct an ablation study on error accumulation, as shown in Table 4 and Fig. 7(b). A and B pick the 1st-200th and 3500th-3700th frames for evaluation, respectively. CIEDE [52] measures the extent of color drift. Baseline removes all our audio-related components. We have the following observations: (1) Baseline suffers from significant quality deterioration in the 3500th-3700th frames. The main reason is that raw audio embeddings have conflicts with original backbone priors, triggering latent distribution error at each clip. As the number of generated frames increases, error accumulation becomes more severe, and the shift of the subsequent distribution from the target distribution grows larger. (2) *w/* Adapter maintains video fidelity even in the later frame intervals. It indicates that our Audio Adapter helps the diffusion to overcome the scarcity of audio priors via timestep-aware modulation, thereby tackling the error accumulation. (3) Our guidance can also ensure the video quality stability in the long video generation to some extent, as it can further alleviate the latent error at each clip. (4) Our audio-related components ensure that even after generating 3500+ frames, the video fidelity remain stable without significant degradation.

Audio Native Guidance. We compare our Audio Native Guidance with conventional CFG, as shown in Table 5 and Fig. 11. While conventional CFG only regards each external condition as an individual signal which is independent of latents, our guidance considers audio embeddings as correlated with the latents, accounting for the joint audio-latent distribution, thereby further facilitating the lip synchronization/naturalness and video fidelity.

Table 5: Ablation study on the guidance and long avatar video generation methods. *w/* CFG replaces our Audio Native Guidance with Classifier-Free-Guidance (CFG).

Method	FVD↓	CSIM↑	Sync-C↑	Sync-D↓	IQA↑	ASE↑
w/o Guidance (*w/* Audio Adapter)	866	0.822	7.48	8.36	3.74	2.31
w/ CFG [25] (*w/* Audio Adapter)	822	0.828	7.62	7.91	3.78	2.33
Motion Frame [11] (*w/* Audio Adapter)	772	0.836	8.12	6.98	3.72	2.23
Context Frame [84] (*w/* Audio Adapter)	578	0.832	8.16	6.91	3.75	2.28
Sliding Window [30] (*w/* Audio Adapter)	698	0.842	8.18	6.89	3.77	2.31
Ours	**504**	**0.849**	**8.24**	**6.79**	**3.84**	**2.39**

Long Video Strategy. We conduct a comparison between our DWSW and other types of long avatar video generation strategies, as shown in Table 5 and Fig. 11. We can see that motion frame [11, 33], context Frame [84], and conventional sliding-window [30] fail to eliminate the jitter caused by the connection between video clips. By contrast, our DWSW leverages a logarithmic interpolation to dynamically assign weights to different context windows, significantly mitigating the impact of video clip connections. Notably, we do not attribute the success of long video generation solely to the Audio Adapter or DWSW. Table 2 shows sliding window alone is insufficient. The two components are complementary: Audio Adapter suppresses per-clip errors while DWSW smooths inter-clip boundaries (Appx. A.2). More ablation studies and analysis are in Appx.A.11 and Appx.A.2.

Scaling. We adopt StableAvatar to Wan2.2-14B as shown in Appx.A.8. We can see that our method still works under model scaling. While the 14B backbone yields finer facial details, the 1.3B model already surpasses existing 14B-based SOTAs. Thus, considering GPU cost and inference speed, we select the 1.3B model as our core backbone.

4.4 Applications and User Study

Speed and GPU Resource. We compare the inference speed and GPU cost between our StableAvatar and previous models, as shown in Appx.A.13. With approximately 50% of GPU cost and 10 times the inference speed of the leading competitor Wan-S2V [16], our 1.3B-based StableAvatar significantly outperforms previous 14B-based models [33, 15, 16, 84] in face quality and lip synchronization, highlighting its superiority in long avatar video generation.

Full Body Avatar Videos. We conduct a qualitative experiment in full/half-body avatar animation, as shown in Appx.A.15. StableAvatar can handle full/half-body avatar animation in high fidelity.

Multi-Avatar/Cartoon Animation. We experiment on audio-driven multi-avatar/cartoon animation, as shown in Appx.A.16 and Appx.A.17. We can see that our model is capable of animating multiple individuals and cartoon avatars.

Different Backbones. We adopt StableAvatar to different backbones to validate its robustness, as shown in Table 7. It shows that our design can be transferred to different diffusion-based frameworks.

User Study. We conduct a user study on 30 selected videos. The participants are basically university students and faculty. In each case, participants are first presented with the reference image and the audio. Then we provide two videos (one is generated by our StableAvatar and the other is synthesized by a competitor) in random order. Participants are then asked to answer questions: L-A/A-A/B-A/I-A: "Which one has better lip/appearance/background/ID alignment with the audio/reference?". Appx.A.14 shows the superiority of our model in subjective evaluation.

5 Conclusion

In this paper, we proposed StableAvatar, a video diffusion transformer with dedicated modules to synthesize ultra-long high-quality avatar videos. StableAvatar first utilized off-the-shelf models to gain audio embeddings. To overcome the scarcity of audio priors of diffusion backbones, StableAvatar introduced an Audio Adapter to refine audio embeddings. In inference, to further enhance lip synchronization, StableAvatar introduced an Audio Native Guidance Mechanism to replace conventional Classifier-Free Guidance. To improve the video's smoothness, StableAvatar further proposed a dynamic weighted sliding-window strategy. Experiments across datasets demonstrated the superiority of our model in producing ultra-long avatar videos.

6 Acknowledgment

This work was supported by the National Natural Science Foundation of China (No. 62472098) and the Science and Technology Commission of Shanghai Municipality (No. 25511106100).

References

[1] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli. wav2vec 2.0: A framework for self-supervised learning of speech representations. *Advances in neural information processing systems*, 2020.

[2] F. Bao, C. Xiang, G. Yue, G. He, H. Zhu, K. Zheng, M. Zhao, S. Liu, Y. Wang, and J. Zhu. Vidu: a highly consistent, dynamic and skilled text-to-video generator with diffusion models. *arXiv preprint arXiv:2405.04233*, 2024.

[3] A. Blattmann, T. Dockhorn, S. Kulal, D. Mendelevitch, M. Kilian, D. Lorenz, Y. Levi, Z. English, V. Voleti, A. Letts, et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023.

[4] T. Brooks, J. Hellsten, M. Aittala, T.-C. Wang, T. Aila, J. Lehtinen, M.-Y. Liu, A. Efros, and T. Karras. Generating long videos of dynamic scenes. In *NIPS*, 2022.

[5] T. Brooks, B. Peebles, C. Holmes, W. DePue, Y. Guo, L. Jing, D. Schnurr, J. Taylor, T. Luhman, E. Luhman, C. Ng, R. Wang, and A. Ramesh. Video generation models as world simulators. 2024.

[6] Y. Chen, S. Liang, Z. Zhou, Z. Huang, Y. Ma, J. Tang, Q. Lin, Y. Zhou, and Q. Lu. Hunyuanvideo-avatar: High-fidelity audio-driven human animation for multiple characters. *arXiv preprint arXiv:2505.20156*, 2025.

[7] Y. Chen, Z. Wu, X. Qiu, and Y.-G. Jiang. Catok: Taming mean flows for one-dimensional causal image tokenization. In *CVPR*, 2026.

[8] Z. Chen, J. Cao, Z. Chen, Y. Li, and C. Ma. Echomimic: Lifelike audio-driven portrait animations through editable landmark conditions. In *AAAI*, 2025.

[9] K. Cheng, X. Cun, Y. Zhang, M. Xia, F. Yin, M. Zhu, X. Wang, J. Wang, and N. Wang. Videoretalking: Audio-based lip synchronization for talking head video editing in the wild. In *SIGGRAPH Asia*, 2022.

[10] J. S. Chung and A. Zisserman. Out of time: automated lip sync in the wild. In *ACCV*, 2016.

[11] J. Cui, H. Li, Y. Zhan, H. Shang, K. Cheng, Y. Ma, S. Mu, H. Zhou, J. Wang, and S. Zhu. Hallo3: Highly dynamic and realistic portrait image animation with video diffusion transformer. In *CVPR*, 2025.

[12] J. Deng, J. Guo, N. Xue, and S. Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *CVPR*, 2019.

[13] P. Dhariwal and A. Nichol. Diffusion models beat gans on image synthesis. In *NeurIPS*, 2021.

[14] A. Ephrat, I. Mosseri, O. Lang, T. Dekel, K. Wilson, A. Hassidim, W. T. Freeman, and M. Rubinstein. Looking to listen at the cocktail party: A speaker-independent audio-visual model for speech separation. *arXiv preprint arXiv:1804.03619*, 2018.

[15] Q. Gan, R. Yang, J. Zhu, S. Xue, and S. Hoi. Omniavatar: Efficient audio-driven avatar video generation with adaptive body animation. *arXiv preprint arXiv:2506.18866*, 2025.

[16] X. Gao, L. Hu, S. Hu, M. Huang, C. Ji, D. Meng, J. Qi, P. Qiao, Z. Shen, Y. Song, et al. Wan-s2v: Audio-driven cinematic video generation. *arXiv preprint arXiv:2508.18621*, 2025.

[17] Y. Guo, C. Yang, A. Rao, Z. Liang, Y. Wang, Y. Qiao, M. Agrawala, D. Lin, and B. Dai. Animatediff: Animate your personalized text-to-image diffusion models without specific tuning. In *ICLR*, 2024.

[18] W. Harvey, S. Naderiparizi, V. Masrani, C. Weilbach, and F. Wood. Flexible diffusion modeling of long videos. In *NIPS*, 2022.

[19] R. Henschel, L. Khachatryan, H. Poghosyan, D. Hayrapetyan, V. Tadevosyan, Z. Wang, S. Navasardyan, and H. Shi. Streamingt2v: Consistent, dynamic, and extendable long video generation from text. In *CVPR*, 2025.

[20] A. Hertz, R. Mokady, J. Tenenbaum, K. Aberman, Y. Pritch, and D. Cohen-Or. Prompt-to-prompt image editing with cross attention control. *arXiv preprint arXiv:2208.01626*, 2022.

[21] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *NIPS*, 2017.

[22] G. E. Hinton. Training products of experts by minimizing contrastive divergence. *Neural computation*, 2002.

[23] J. Ho, A. Jain, and P. Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020.

[24] J. Ho, C. Saharia, W. Chan, D. J. Fleet, M. Norouzi, and T. Salimans. Cascaded diffusion models for high fidelity image generation. *JMLR*, 2022.

[25] J. Ho and T. Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

[26] W. Hong, M. Ding, W. Zheng, X. Liu, and J. Tang. Cogvideo: Large-scale pretraining for text-to-video generation via transformers. *arXiv preprint arXiv:2205.15868*, 2022.

[27] Y. Huang, S. Tu, X. Yan, Z. Yang, J. Han, H. Xu, K. Pan, Y.-G. Jiang, and Z. Wu. Va-judger: Reward modeling from human preference feedback for joint video-audio generation. *arXiv preprint arXiv:2608.18607*, 2026.

[28] Y. Huang, S. Tu, X. Yan, J. Zhan, Z. Yang, Z. Xing, H. Zhang, T. Zhang, Y.-G. Jiang, and Z. Wu. Agentic visual generation: From generative models to agentic control. *arXiv preprint arXiv:2609.06758*, 2026.

[29] Z. Huang, Y. He, J. Yu, F. Zhang, C. Si, Y. Jiang, Y. Zhang, T. Wu, Q. Jin, N. Chanpaisit, et al. Vbench: Comprehensive benchmark suite for video generative models. In *CVPR*, 2024.

[30] X. Ji, X. Hu, Z. Xu, J. Zhu, C. Lin, Q. He, J. Zhang, D. Luo, Y. Chen, Q. Lin, et al. Sonic: Shifting focus to global audio perception in portrait animation. In *CVPR*, 2025.

[31] J. Jiang, C. Liang, J. Yang, G. Lin, T. Zhong, and Y. Zheng. Loopy: Taming audio-driven portrait avatar with long-term motion dependency. *arXiv preprint arXiv:2409.02634*, 2024.

[32] W. Kong, Q. Tian, Z. Zhang, R. Min, Z. Dai, J. Zhou, J. Xiong, X. Li, B. Wu, J. Zhang, et al. Hunyuanvideo: A systematic framework for large video generative models. *arXiv preprint arXiv:2412.03603*, 2024.

[33] Z. Kong, F. Gao, Y. Zhang, Z. Kang, X. Wei, X. Cai, G. Chen, and W. Luo. Let them talk: Audio-driven multi-person conversational video generation. *arXiv preprint arXiv:2505.22647*, 2025.

[34] W.-S. Lai, J.-B. Huang, O. Wang, E. Shechtman, E. Yumer, and M.-H. Yang. Learning blind video temporal consistency. In *ECCV*, 2018.

[35] J. Leng, S. Tu, H. Cao, S. Xie, D. Dong, Z. Wu, and Y.-G. Jiang. Preference score distillation: Leveraging 2d rewards to align text-to-3d generation with human preference. *arXiv preprint arXiv:2603.01594*, 2026.

[36] C. Li, C. Zhang, W. Xu, J. Xie, W. Feng, B. Peng, and W. Xing. Latentsync: Audio conditioned latent diffusion models for lip sync. *arXiv e-prints*, pages arXiv–2412, 2024.

[37] G. Lin, J. Jiang, C. Liang, T. Zhong, J. Yang, Z. Zheng, and Y. Zheng. Cyberhost: A one-stage diffusion framework for audio-driven talking body generation. In *ICLR*, 2025.

[38] G. Lin, J. Jiang, J. Yang, Z. Zheng, and C. Liang. Omnihuman-1: Rethinking the scaling-up of one-stage conditioned human animation models. *arXiv preprint arXiv:2502.01061*, 2025.

[39] Y. Lipman, R. T. Chen, H. Ben-Hamu, M. Nickel, and M. Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

[40] L. Liu, T. Ma, B. Li, Z. Chen, J. Liu, G. Li, S. Zhou, Q. He, and X. Wu. Phantom: Subject-consistent video generation via cross-modal alignment. *arXiv preprint arXiv:2502.11079*, 2025.

[41] Y. Lu, Y. Liang, L. Zhu, and Y. Yang. Freelong: Training-free long video generation with spectralblend temporal attention. In *NIPS*, 2024.

[42] Y. Lu and Y. Yang. Freelong++: Training-free long video generation via multi-band spectralfusion. *arXiv preprint arXiv:2507.00162*, 2025.

[43] C. Lugaresi, J. Tang, H. Nash, C. McClanahan, E. Uboweja, M. Hays, F. Zhang, C.-L. Chang, M. G. Yong, J. Lee, et al. Mediapipe: A framework for building perception pipelines. *arXiv preprint arXiv:1906.08172*, 2019.

[44] C. Meng, Y. He, Y. Song, J. Song, J. Wu, J.-Y. Zhu, and S. Ermon. Sdedit: Guided image synthesis and editing with stochastic differential equations. In *ICLR*, 2021.

[45] R. Meng, Y. Wang, W. Wu, R. Zheng, Y. Li, and C. Ma. Echomimicv3: 1.3 b parameters are all you need for unified multi-modal and multi-task human animation. *arXiv preprint arXiv:2507.03905*, 2025.

[46] R. Meng, X. Zhang, Y. Li, and C. Ma. Echomimicv2: Towards striking, simplified, and semi-body human animation. In *CVPR*, 2025.

[47] A. Q. Nichol and P. Dhariwal. Improved denoising diffusion probabilistic models. In *ICML*, 2021.

[48] W. Peebles and S. Xie. Scalable diffusion models with transformers. In *ICCV*, 2023.

[49] H. Qiu, M. Xia, Y. Zhang, Y. He, X. Wang, Y. Shan, and Z. Liu. Freenoise: Tuning-free longer video diffusion via noise rescheduling. *arXiv preprint arXiv:2310.15169*, 2023.

[50] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021.

[51] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022.

[52] G. Sharma, W. Wu, and E. N. Dalal. The ciede2000 color-difference formula: Implementation notes, supplementary test data, and mathematical observations. *Color Research & Application: Endorsed by Inter-Society Color Council, The Colour Group (Great Britain), Canadian Society for Color, Color Science Association of Japan, Dutch Society for the Study of Color, The Swedish Colour Centre Foundation, Colour Society of Australia, Centre Français de la Couleur*, 30(1):21–30, 2005.

[53] O. Siméoni, H. V. Vo, M. Seitzer, F. Baldassarre, M. Oquab, C. Jose, V. Khalidov, M. Szafraniec, S. Yi, M. Ramamonjisoa, et al. Dinov3. *arXiv preprint arXiv:2508.10104*, 2025.

[54] U. Singer, A. Polyak, T. Hayes, X. Yin, J. An, S. Zhang, Q. Hu, H. Yang, O. Ashual, O. Gafni, et al. Make-a-video: Text-to-video generation without text-video data. *arXiv preprint arXiv:2209.14792*, 2022.

[55] I. Skorokhodov, S. Tulyakov, and M. Elhoseiny. Stylegan-v: A continuous video generator with the price, image quality and perks of stylegan2. In *CVPR*, 2022.

[56] J. Song, C. Meng, and S. Ermon. Denoising diffusion implicit models. In *ICLR*, 2021.

[57] L. Song, W. Wu, C. Fu, C. C. Loy, and R. He. Audio-driven dubbing for user generated contents via style-aware semi-parametric synthesis. *IEEE Transactions on Circuits and Systems for Video Technology*, 33(3):1247–1261, 2022.

[58] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole. Score-based generative modeling through stochastic differential equations. In *ICLR*, 2021.

[59] L. Tian, Q. Wang, B. Zhang, and L. Bo. Emo: Emote portrait alive generating expressive portrait videos with audio2video diffusion model under weak conditions. In *ECCV*, 2024.

[60] L. Tran and X. Liu. Nonlinear 3d face morphable model. In *CVPR*, 2018.

[61] H.-Y. Tseng, Q. Li, C. Kim, S. Alsisan, J.-B. Huang, and J. Kopf. Consistent view synthesis with pose-guided diffusion models. In *CVPR*, 2023.

[62] S. Tu, Q. Dai, Z.-Q. Cheng, H. Hu, X. Han, Z. Wu, and Y.-G. Jiang. Motioneditor: Editing video motion via content-aware diffusion. In *CVPR*, 2024.

[63] S. Tu, Q. Dai, Z. Wu, Z.-Q. Cheng, H. Hu, and Y.-G. Jiang. Implicit temporal modeling with learnable alignment for video recognition. In *ICCV*, 2023.

[64] S. Tu, Q. Dai, Z. Zhang, S. Xie, Z.-Q. Cheng, C. Luo, X. Han, Z. Wu, and Y.-G. Jiang. Motionfollower: Editing video motion via lightweight score-guided diffusion. *arXiv preprint arXiv:2405.20325*, 2024.

[65] S. Tu, T. Guan, and L. Kuang. Multiple biological granularities network for person re-identification. In *ICMR*, 2022.

[66] S. Tu, Y. Pan, Y. Huang, X. Han, Z. Xing, Q. Dai, K. Qiu, C. Luo, and Z. Wu. Flashportrait: 6x faster infinite portrait animation with adaptive latent prediction. In *CVPR*, 2026.

[67] S. Tu, Q. Tian, Y. Huang, Y. Wu, X. Han, K. Pan, W. Kong, J. Xiong, J.-W. Zhang, Z. Wu, et al. Prism: Dynamic sparse attention for native 2k joint video-audio generation model training. *arXiv preprint arXiv:2610.05416*, 2026.

[68] S. Tu, Q. Tian, Z. Yang, Y. Wu, X. Han, W. Kong, J. Xiong, J.-W. Zhang, Z. Zhong, L. Bo, et al. Baton: Explicit semantic blueprints for joint video-audio generation. *arXiv preprint arXiv:2605.25195*, 2026.

[69] S. Tu, Z. Xing, X. Han, Z.-Q. Cheng, Q. Dai, C. Luo, and Z. Wu. Stableanimator: High-quality identity-preserving human image animation. In *CVPR*, 2025.

[70] S. Tu, Z. Xing, X. Han, Z.-Q. Cheng, Q. Dai, C. Luo, Z. Wu, and Y.-G. Jiang. Stableanimator++: Overcoming pose misalignment and face distortion for human image animation. *arXiv preprint arXiv:2507.15064*, 2025.

[71] N. Tumanyan, M. Geyer, S. Bagon, and T. Dekel. Plug-and-play diffusion features for text-driven image-to-image translation. In *CVPR*, 2023.

[72] T. Unterthiner, S. Van Steenkiste, K. Kurach, R. Marinier, M. Michalski, and S. Gelly. Towards accurate generative models of video: A new metric & challenges. *arXiv preprint arXiv:1812.01717*, 2018.

[73] V. Voleti, A. Jolicoeur-Martineau, and C. Pal. Mcvd-masked conditional video diffusion for prediction, generation, and interpolation. In *NIPS*, 2022.

[74] T. Wan, A. Wang, B. Ai, B. Wen, C. Mao, C.-W. Xie, D. Chen, F. Yu, H. Zhao, J. Yang, J. Zeng, J. Wang, J. Zhang, J. Zhou, J. Wang, J. Chen, K. Zhu, K. Zhao, K. Yan, L. Huang, M. Feng, N. Zhang, P. Li, P. Wu, R. Chu, R. Feng, S. Zhang, S. Sun, T. Fang, T. Wang, T. Gui, T. Weng, T. Shen, W. Lin, W. Wang, W. Wang, W. Zhou, W. Wang, W. Shen, W. Yu, X. Shi, X. Huang, X. Xu, Y. Kou, Y. Lv, Y. Li, Y. Liu, Y. Wang, Y. Zhang, Y. Huang, Y. Li, Y. Wu, Y. Liu, Y. Pan, Y. Zheng, Y. Hong, Y. Shi, Y. Feng, Z. Jiang, Z. Han, Z.-F. Wu, and Z. Liu. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

[75] C. Wang, K. Tian, J. Zhang, Y. Guan, F. Luo, F. Shen, Z. Jiang, Q. Gu, X. Han, and W. Yang. V-express: Conditional dropout for progressive training of portrait video generation. *arXiv preprint arXiv:2406.02511*, 2024.

[76] F.-Y. Wang, W. Chen, G. Song, H.-J. Ye, Y. Liu, and H. Li. Gen-l-video: Multi-text to long video generation via temporal co-denoising. *arXiv preprint arXiv:2305.18264*, 2023.

[77] M. Wang, Q. Wang, F. Jiang, Y. Fan, Y. Zhang, Y. Qi, K. Zhao, and M. Xu. Fantasytalking: Realistic talking portrait generation via coherent motion synthesis. *arXiv preprint arXiv:2504.04842*, 2025.

[78] W. Wang, J. Liu, Z. Lin, J. Yan, S. Chen, C. Low, T. Hoang, J. Wu, J. H. Liew, H. Yan, et al. Magicvideo-v2: Multi-stage high-aesthetic video generation. *arXiv preprint arXiv:2401.04468*, 2024.

[79] H. Wei, Z. Yang, and Z. Wang. Aniportrait: Audio-driven synthesis of photorealistic portrait animation. *arXiv preprint arXiv:2403.17694*, 2024.

[80] Z. Weng, X. Yang, Z. Xing, Z. Wu, and Y.-G. Jiang. Genrec: Unifying video generation and recognition with diffusion models. *arXiv preprint arXiv:2408.15241*, 2024.

[81] H. Wu, Z. Zhang, W. Zhang, C. Chen, L. Liao, C. Li, Y. Gao, A. Wang, E. Zhang, W. Sun, et al. Q-align: Teaching lmms for visual scoring via discrete text-defined levels. *arXiv preprint arXiv:2312.17090*, 2023.

[82] J. Z. Wu, Y. Ge, X. Wang, S. W. Lei, Y. Gu, Y. Shi, W. Hsu, Y. Shan, X. Qie, and M. Z. Shou. Tune-a-video: One-shot tuning of image diffusion models for text-to-video generation. In *CVPR*, 2023.

[83] M. Xu, H. Li, Q. Su, H. Shang, L. Zhang, C. Liu, J. Wang, Y. Yao, and S. Zhu. Hallo: Hierarchical audio-driven visual synthesis for portrait image animation. *arXiv preprint arXiv:2406.08801*, 2024.

[84] S. Yang, Z. Kong, F. Gao, M. Cheng, X. Liu, Y. Zhang, Z. Kang, W. Luo, X. Cai, R. He, et al. Infinitetalk: Audio-driven video generation for sparse-frame video dubbing. *arXiv preprint arXiv:2508.14033*, 2025.

[85] Z. Yang, S. Tu, L. Zhang, Q. Dai, Y.-G. Jiang, and Z. Wu. Arcflow: Unleashing 2-step text-to-image generation via high-precision non-linear flow distillation. *arXiv preprint arXiv:2602.09014*, 2026.

[86] S. Yin, C. Wu, H. Yang, J. Wang, X. Wang, M. Ni, Z. Yang, L. Li, S. Liu, F. Yang, et al. Nuwa-xl: Diffusion over diffusion for extremely long video generation. *arXiv preprint arXiv:2303.12346*, 2023.

[87] W. Zhang, X. Cun, X. Wang, Y. Zhang, X. Shen, Y. Guo, Y. Shan, and F. Wang. Sadtalker: Learning realistic 3d motion coefficients for stylized audio-driven single image talking face animation. In *CVPR*, 2023.

[88] Y. Zhang, Y. Liu, B. Xia, B. Peng, Z. Yan, E. Lo, and J. Jia. Magicmirror: Id-preserved video generation in video diffusion transformers. In *ICCV*, 2025.

[89] Z. Zhang, L. Li, Y. Ding, and C. Fan. Flow-guided one-shot talking face generation with a high-resolution audio-visual dataset. In *CVPR*, 2021.

[90] Y. Zhou, X. Han, E. Shechtman, J. Echevarria, E. Kalogerakis, and D. Li. Makelttalk: speaker-aware talking-head animation. *TOG*, 2020.

[91] H. Zhu, W. Wu, W. Zhu, L. Jiang, S. Tang, L. Zhang, Z. Liu, and C. C. Loy. CelebV-HQ: A large-scale video facial attributes dataset. In *ECCV*, 2022.

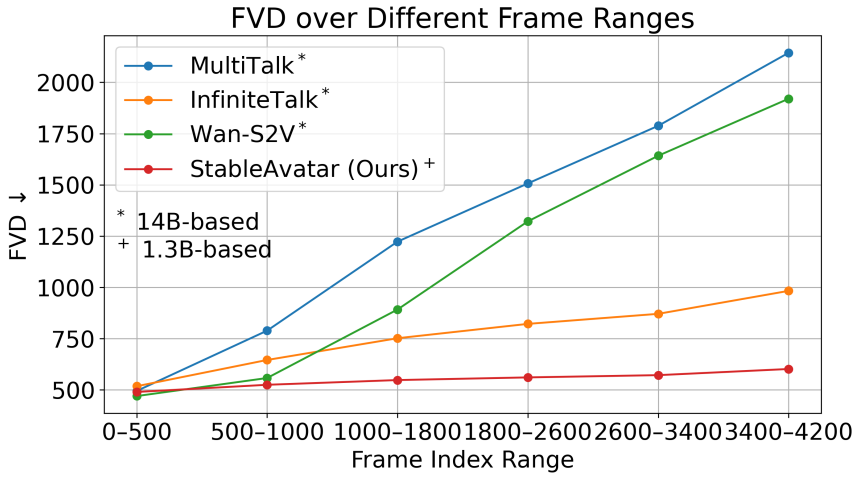

Figure 5: Quantitative comparisons in quality drift. a-b on the x-axis is the frame range from the a-th frame to the b-th frame.

A Technical Appendices

A.1 Preliminaries

The diffusion model includes a forward diffusion process and a reverse denoising process. In the forward process, inspired by Rectified Flow [39], the Gaussian noise is progressively added to the data sample $\boldsymbol{z}_0 \sim \boldsymbol{p}_{\text{data}}$ from the particular data distribution $\boldsymbol{p}_{\text{data}}$:

$$\boldsymbol{z}_t = (1-t)\boldsymbol{z}_0 + t\boldsymbol{z}_1, \tag{12}$$

where $t \in [0,1]$ refers to the timestep and $\boldsymbol{z}_1$ is sampled from a standard-normal distribution $\mathcal{N}(0, I)$. The data sample $\boldsymbol{z}_0$ is ultimately converted into Gaussian noise after $\boldsymbol{T}$ diffusion forward steps. In the reverse process, the diffusion model $\boldsymbol{\varepsilon}_\theta(\boldsymbol{z}_t, t)$ is trained to predict the velocity $\boldsymbol{z}_1 - \boldsymbol{z}_0$ conditioned on the noisy latents z_t and the timestep t. The MSE loss is applied to train $\boldsymbol{\varepsilon}(\cdot)$:

$$\mathcal{L} = \mathbb{E}_{\boldsymbol{z}_0, \boldsymbol{z}_1, t}(\|(\boldsymbol{z}_1 - \boldsymbol{z}_0) - \boldsymbol{\varepsilon}_\theta(\boldsymbol{z}_t, t)\|^2). \tag{13}$$

A.2 Discussion on Audio Modeling and Long-term Error Accumulation

The main differences between our timestep-aware Audio Adapter and a naive audio cross-attention module: (1) Since the backbone is not pretrained with audio priors, directly injecting raw audio embeddings into DiT via cross-attention tends to introduce significant latent distribution errors during audio modeling. Because audio signals inherently contain complex temporal patterns, rich melodic structures, and large variations in amplitude, backbones without audio-specific priors are substantially more difficult to model than those for image or text modalities. It often leads to unstable learning, where large fluctuations in the latent distribution drive it far from the backbone's original prior. As latents are pushed into unfamiliar regions, the backbone knowledge acquired during pre-training is disrupted, compromising the robustness of the diffusion model. Latents tend to collapse when forcibly pushed toward the desired audio-aligned distribution, as the degraded diffusion model cannot reliably capture the audio–latent mapping, leading to accumulated errors. **We validate this assumption in the Error Accumulation part of the Section.4.3 and Fig.6.** By contrast, before being used as Key/Value in DiT, the audio embeddings are first treated as Queries and jointly modeled with latents serving as Keys and Values, which refines the audio embeddings within the DiT framework. In this way, audio embeddings are no longer treated as an external condition independent of DiT, but instead become an internally predicted variable whose values depend on both the input audio and the inner latents. This strong latent–audio coupling compensates for the lack of audio priors in the backbone and stabilizes audio pattern learning, making error accumulation within a single clip negligible. (2) However, directly aligning latents with audio embeddings in our Audio Adapter is challenging due to the large domain gap between the two modalities. Notably, timestep embeddings have established an implicit projection relationship with latents during backbone pretraining. Thus, we leverage timestep embeddings as a bridge by modulating audio embeddings with them. This interaction guides audio

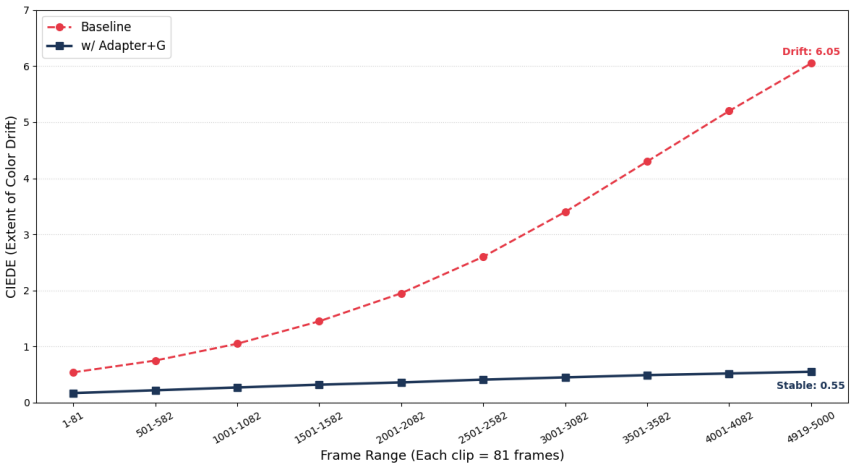

Figure 6: Ablation study on error accumulation of a single clip in the long term. Within the generated frames from 0 to 5000, StableAvatar synthesizes videos in a clip-by-clip manner. Thus, we evaluate the degree of color drift within each individual clip. The Baseline removes all our audio-related components. Both the Baseline and w/ Adapter+G adopt the same sliding-window strategy. The results show that the Baseline already suffers from relatively large per-clip errors due to audio modeling. As the number of clips increases, these errors are further exacerbated and accumulate over time, since each clip experiences drift introduced by audio modeling, ultimately leading to a significant quality degradation. In contrast, our w/ Adapter+G reduces the error introduced during the generation of each clip, effectively preventing error accumulation over long sequences.

embeddings to align with latent-specific distributions, facilitating the establishment of latent–audio correlations. As a result, the Audio Adapter can be trained more stably, further reducing error accumulation within each clip.

Regarding Long-term Error Accumulation, errors in long video generation arise from two sources: (1) errors within individual clips, and (2) error propagation and amplification across clip boundaries during clip stitching. To mitigate intra-clip error accumulation, we introduce a timestep-aware Audio Adapter that stabilizes audio pattern modeling. While the error within each clip is substantially reduced, it remains non-zero and can still accumulate to a noticeable level when a large number of clips are concatenated. To address error propagation and amplification across clips, we further propose a Dynamic Weighted Sliding Window Denoising (DWSW) strategy, which explicitly targets inter-clip error accumulation and ensures long-term temporal consistency. The failure of traditional sliding window strategies [11, 84, 30] fundamentally stems from the stochastic divergence of latent trajectories between overlapping segments. Traditional strategies typically use linear blending to force these two different windows together. The failure of this approach lies in its hard averaging, leading to visual artifacts like ghosting or sudden jumps at the boundaries. The window transition is not smooth, causing the generated content of the same frames to drift into slightly different versions in the latent space across two adjacent windows. Our motivation of DWSW is to move beyond mere blending toward manifold alignment. By employing a logarithmic weighting function, DWSW ensures that the transition between windows is not just smooth, but adaptive to the settling phase of the diffusion process. The logarithmic curve allows for a more aggressive retention of the previous window's temporal priors during the initial transition phase, followed by a subtle, non-linear handover to the new window's trajectory. This prevents the sudden distribution shifts that traditional linear methods suffer from, effectively maintaining the latent trajectory within a continuous and coherent manifold across ultra-long durations.

A.3 Proof and Discussion on Equations of Audio Native Guidance

Proofs of Eq. 6. Eq. 6 is a valid probabilistic objective derived from the product-of-experts principle [22], which states that the product of any finite set of non-negative functions defines a valid unnormalized density. Classifier guidance [13] also constructs a tilted distribution $\tilde{p}(\mathbf{x}_t|y) \propto p(\mathbf{x}_t) \cdot p(y|\mathbf{x}_t)^s$. Our Eq. 6 extends this to three objectives: (1) $p([\mathbf{z}_t, \bar{\mathbf{a}}_t]|\mathbf{A})$, the base conditional

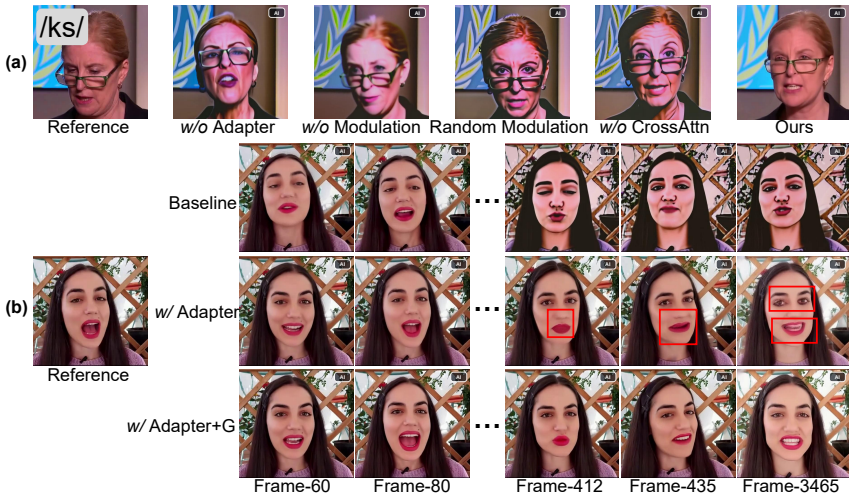

Figure 7: Ablation study on audio modeling.

distribution. (2) $p(\mathbf{A}|[\mathbf{z}_t, \bar{\mathbf{a}}_t])^\alpha$, enforcing external consistency with the observed audio, playing the same role as $p(y|\mathbf{x}_t)$ in classifier guidance [13]. (3) $p(\bar{\mathbf{a}}_t|\mathbf{z}_t, \mathbf{A})^\beta$, enforcing internal coherence between $\bar{\mathbf{a}}_t$ and $\mathbf{z}_t$. Each term is non-negative, so their product is a valid unnormalized density. The derivation from Eq. 6 to Eq. 10 strictly follows Bayes' theorem, with no additional approximations beyond those already accepted in CFG [25]. Table 5 and Table 10 validate the soundness of the formulation.

Discussion on Audio Native Guidance and Conventional CFG. In the standard CFG, the guided score is $(1+s)\nabla \log p(\boldsymbol{z}_t|y) - s\nabla \log p(\boldsymbol{z}_t)$, where the condition y (text or image) is an external signal independent of the latents. Our key observation is that the refined audio embeddings $\bar{\boldsymbol{a}}_t$ are fundamentally different from such external conditions because $\bar{\boldsymbol{a}}_t$ is an internal variable jointly predicted by the DiT, whose value depends on both the input audio $\boldsymbol{A}$ and the latents $\boldsymbol{z}_t$ (Eq. 3–4). Simply substituting audio for text/image in CFG ignores this internal dependency and treats $\bar{\boldsymbol{a}}_t$ as if it were conditionally independent of $\boldsymbol{z}_t$, forfeiting the latent-aware information already encoded in $\bar{\boldsymbol{a}}_t$. We thus extend the CFG by treating $[\boldsymbol{z}_t, \bar{\boldsymbol{a}}_t]$ as the joint prediction target and introducing two distinct guidance terms in Eq. 6. The first term $p(\boldsymbol{A}|[\boldsymbol{z}_t, \bar{\boldsymbol{a}}_t])^\alpha$ enforces external consistency between the joint prediction and the observed audio, analogous to the classifier term in standard CFG. The second term $p(\bar{\boldsymbol{a}}_t|\boldsymbol{z}_t, \boldsymbol{A})^\beta$ enforces internal coherence between $\bar{\boldsymbol{a}}_t$ and $\boldsymbol{z}_t$, which has no counterpart in conventional CFG. When converted to the score function format (Eq. 9–10), our formulation naturally recovers a three-term structure analogous to CFG but with the critical difference that the guidance operates on the joint audio-latent distribution rather than on independent external conditions.

Discussion on Dirac delta and Ill-posed Score Matching of Logarithm Gradient. $\bar{\mathbf{a}}_t$ is deterministically computed from $\mathbf{z}_t$ and $\mathbf{A}$, so $p(\bar{\mathbf{a}}_t|\mathbf{z}_t, \mathbf{A})$ collapses to a Dirac delta. However, this does not invalidate our method. Although $p(\bar{\mathbf{a}}_t|\mathbf{z}_t, \mathbf{A})$ is a Dirac delta, this does not affect the effectiveness of our method, as empirically validated by Table 5. We emphasize that Eq. 6–9 serves as a progressive process for deriving the final inference formula. Eq. 9 is never actually computed during inference. The actual implementation is Eq. 10, where each term is a forward pass of $D(\cdot;\theta)$, and $\bar{\mathbf{a}}_t$ is computed on-the-fly from $\mathbf{z}_t$ inside the network (Eq. 2–4). Thus, the entire computation graph is end-to-end differentiable with respect to $\mathbf{z}_t$. The ∇ in Eq. 9 denotes $\nabla_{\mathbf{z}_t}$, not $\nabla_{\bar{\mathbf{a}}_t}$, so no computation of $\log p(\bar{\mathbf{a}}_t|\mathbf{z}_t, \mathbf{A})$ or its gradient is ever required. The transition from Eq. 9 to Eq. 10 follows the same approximation as CFG [25], which converts $(1+w)\nabla \log p(\mathbf{z}_\lambda|c) - w\nabla \log p(\mathbf{z}_\lambda)$ to $(1+w)\epsilon_\theta(\mathbf{z}_\lambda, c) - w\epsilon_\theta(\mathbf{z}_\lambda, \emptyset)$ by replacing each score with a network forward pass. Our Eq. 9 to Eq. 10 applies the same principle, replacing each $\nabla \log p(\cdot)$ with the corresponding forward pass of $D(\cdot;\theta)$ under the matching dropout configuration.

Discussion on Principles of Score-based Generative Models. CFG [25] uses the identical approximation. The unconditional score $\nabla \log p(\mathbf{x}_t)$ should theoretically be $\nabla \log \int p(\mathbf{x}_t|c)p(c)dc$, but CFG approximates it via $\epsilon_\theta(\mathbf{x}_t, c = \emptyset)$, enabled by dropping c with probability p_{uncond} during training. **Our approach follows the same practice by zeroing out $\mathbf{A}$ or $\bar{\mathbf{a}}_t$ during training (Appx. A.5)**, introducing no additional assumption beyond CFG [25]. Table 2 and Table 4 confirm effectiveness.

A.4 Additional Ablation Study on Audio Modeling

Fig. 7 shows the qualitative results of ablation study on audio modeling.

A.5 More Training Details

Regarding Audio Native Guidance, we need to drop it text, image, and audio signals ($\boldsymbol{A}$ and $\bar{\boldsymbol{a}}_t$) during training to enable the guidance. However, we find that incorporating text prompt and reference image into the guidance significantly increases GPU cost and does not substantially improve visual quality (Appx.A.10). Therefore, we do not use text prompt and reference image as guidance factors. Regarding the audio signals, we need to randomly drop $\boldsymbol{A}$ and $\bar{\boldsymbol{a}}_t$ during training. Concretely, at each training step, we randomly set either $\boldsymbol{A}$ or $\bar{\boldsymbol{a}}_t$ to an all-zero tensor. Notably, zeroing out the raw audio embeddings $\boldsymbol{A}$ does not force the DiT-predicted $\bar{\boldsymbol{a}}_t$ to be zero.

In terms of training loss, $\boldsymbol{q}$ is a random variable uniformly distributed over the interval $[0, 1]$, which is triggered in a sample-wise manner. Compared to using a weighted average of all loss terms, triggering different piecewise loss functions via random sampling enables more targeted learning. Specifically, the first term encourages the model to focus on fine-grained facial detail modeling, the second emphasizes lip–audio synchronization, and the third promotes global feature modeling. Randomly switching among these piecewise loss functions prevents the model from overfitting to any single aspect or region, leading to more balanced and robust learning.

A.6 More Inference Details

We provide the complete inference settings: resolution 480×832, 30 FPS, 30 denoising steps (Euler scheduler), window length 21 latent frames (81 video frames), overlap 10 latent frames, guidance scales α=4.5 and β=3.0, random seed 42 (fixed for all experiments), backbone Wan2.1-I2V 1.3B (official checkpoint), precision bf16. For the text prompt, we follow a structured template covering the number of subjects, their appearance and clothing, current action, surrounding environment and lighting, body gesture and hand position, and shot type. An example prompt is A young woman is speaking expressively into a silver microphone inside a recording studio. She has shoulder-length dark hair and wears a cream knit sweater. Her right hand rests on the mic stand while she gestures gently with her left. Acoustic foam panels line the walls behind her, with warm overhead lighting. A medium close-up shot captures her animated expressions and upper body movement. For all competitors, we used their officially released code and checkpoints. All baselines were allowed their best recommended long-video settings (e.g., motion frames for Hallo3, context frames for InfiniteTalk, sliding window for Sonic).

A.7 Sliding Window Details

Fig. 8 and Algorithm 1 show a detailed pipeline of our Dynamic Weighted Sliding Window Denoising Strategy (DWSW). We further provide the mathematical derivation of the logarithmic interpolation weights employed in DWSW. Two adjacent windows produce latent trajectories $\boldsymbol{z}^{(\text{prev})}$ and $\boldsymbol{z}^{(\text{new})}$ over their m-frame overlap region. Let $\tau \in [0, 1]$ be the normalized position within the overlap. The fused latent is:

$$\boldsymbol{z}^{(\text{fused})}(\tau) = w(\tau) \cdot \boldsymbol{z}^{(\text{new})}(\tau) + (1 - w(\tau)) \cdot \boldsymbol{z}^{(\text{prev})}(\tau). \tag{14}$$

At $\tau = 0$, the overlap frame sits at the start of the current window but deep inside the previous window, making the temporal conditioning maximally asymmetric and $\|\boldsymbol{z}^{(\text{new})} - \boldsymbol{z}^{(\text{prev})}\|$ largest. As $\tau \to 1$, the frame gains preceding context within the current window, reducing this asymmetry. A good weighting function should therefore apply aggressive retention of $\boldsymbol{z}^{(\text{prev})}$ near $\tau = 0$ (where trajectory divergence is large) and transition smoothly to $\boldsymbol{z}^{(\text{new})}$ near $\tau = 1$ (where the two trajectories converge). We define:

$$w(\tau) = \log(1 + \tau(e - 1)), \quad \frac{dw}{d\tau} = \frac{e - 1}{1 + \tau(e - 1)}. \tag{15}$$

The derivative $\frac{dw}{d\tau}$ monotonically decreases from $e-1 \approx 1.718$ at $\tau = 0$ to $1/(e-1) \approx 0.582$ at $\tau = 1$. Compared to linear interpolation ($\frac{dw}{d\tau} = 1$, constant), logarithmic weighting concentrates the weight transition in the early overlap region where trajectory divergence is maximal, then gradually flattens as the trajectories naturally converge. This adaptive rate matching prevents the abrupt distribution

Algorithm 1 Dynamic Weighted Sliding-Window Strategy

Require: Audio Embeddings $\boldsymbol{emb}_{aud}^{[0,L]}$, T, Latents $\boldsymbol{z}_T^{[0,L]}$, Diffusion Model $\boldsymbol{D}(\cdot)$, l $(l < L)$, m, $\boldsymbol{y}$, $\boldsymbol{I}$

for $t \in \{T, \ldots, 1\}$ **do** ▷ T is denoising steps
  start index $s = 0$ ▷ m is overlap length between windows
  end index $e = s + l$ ▷ l is video clip/window length
  previous end index $e_{prev} = e$ ▷ $\boldsymbol{z}_T^{[0,L]}$ are noised latents
  while $e \leq L$ **do** ▷ $\boldsymbol{I}$ is the reference image
    $\boldsymbol{z}_{t-1}^{[s,e]} = \boldsymbol{D}(\boldsymbol{z}_t^{[s,e]}, \boldsymbol{emb}_{aud}^{[s,e]}, t, \boldsymbol{y}, \boldsymbol{I})$ ▷ $\boldsymbol{y}$ is the text
    if $s \neq 0$ and $t \neq T$: ▷ $\texttt{log1p}(x)$ is $\log(x+1)$
      $\boldsymbol{w} = \texttt{np.linspace}(0, 1, \text{num_samples}{=}m)$
      $\boldsymbol{w} = \texttt{np.log1p}(\boldsymbol{w} * (\texttt{np.exp}(1) - 1))$
      $\boldsymbol{w} = \frac{\boldsymbol{w} - \texttt{min}(\boldsymbol{w})}{\texttt{max}(\boldsymbol{w}) - \texttt{min}(\boldsymbol{w})}$
      $\boldsymbol{z}_{t-1}^{[s,s+m]} = \boldsymbol{w} * \boldsymbol{z}_{t-1}^{[s,s+m]} + (1 - \boldsymbol{w}) * \boldsymbol{z}_{t-1}^{[e_{prev}-m, e_{prev}]}$
    if $e < L$: ▷ It's for the edge case of the final clip
      $e_{prev} = e, s = s + (l - m)$
      $e = \texttt{min}(s + l, L)$
    else: **break**
return $\boldsymbol{z}_0^{[0,L]}$

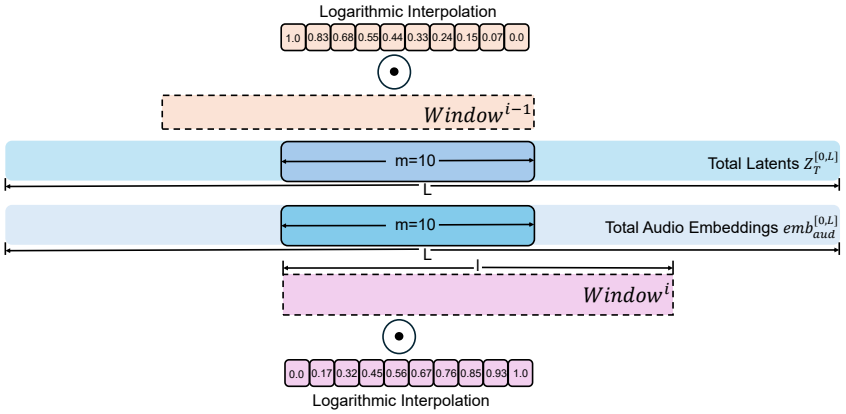

Figure 8: The pipeline of our DWSW.

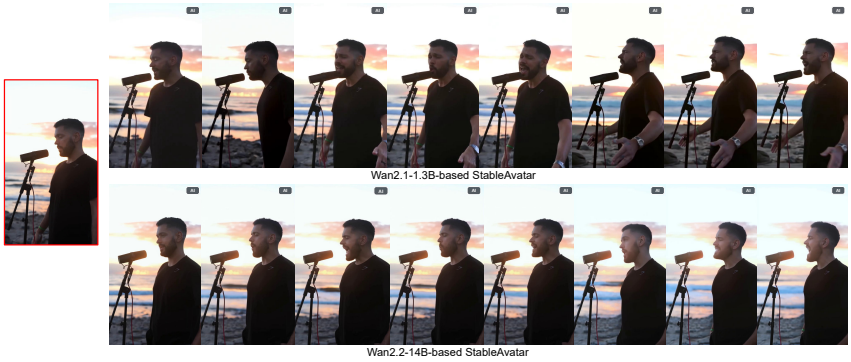

Figure 9: Ablation study on scalability.

jumps that linear blending produces at boundaries, yielding perceptually seamless transitions. Table 8 validates this analysis.

A.8 14B-based Model Results

We replace the original StableAvatar backbone (Wan2.1-1.3B) with Wan2.2-14B to evaluate whether our method continues to improve performance under model scaling, as shown in the Table 6 and Fig.9.

Table 6: Ablation study on scalability on HDTF/AVSpeech/Long100. In the table elements a / b / c, a, b, and c refer to the result on the HDTF, AVSpeech, and Long100, respectively.

Model	FVD↓	CSIM↑	Sync-C↑	Sync-D↓	IQA↑	ASE↑
Wan-S2V-14B	368/625/1685	0.872/0.869/0.477	8.12/7.48/4.88	6.98/7.92/9.68	3.83/3.81/2.47	2.42/2.30/1.53
StableAvatar-1.3B	375/640/504	0.875/0.872/0.849	8.15/7.56/8.24	6.94/7.85/6.79	3.90/3.79/3.84	2.46/2.32/2.39
StableAvatar-14B	**336/595/462**	**0.878/0.874/0.853**	**8.17/7.62/8.29**	**6.94/7.82/6.73**	**3.91/3.84/3.89**	**2.48/2.35/2.44**

Table 7: Comparison results on different backbones.

Model	FVD↓	CSIM↑	Sync-C↑	Sync-D↓	IQA↑	ASE↑
SVD-based Sonic	2877	0.451	4.03	10.23	2.08	1.13
SVD-based StableAvatar	846	0.803	7.82	7.21	3.45	1.94
HunyuanVideo-based HunyuanAvatar	1743	0.472	4.34	10.07	2.46	1.52
HunyuanVideo-based StableAvatar	558	0.836	8.07	6.93	3.61	2.16

Table 8: Ablation study on different weight assignment.

Model	FVD↓	CSIM↑	Sync-C↑	Sync-D↓	IQA↑	ASE↑
Fixed weights	705	0.838	8.08	7.18	3.73	2.25
Uniformed gap	594	0.847	8.15	6.94	3.82	2.33
Ours	**504**	**0.849**	**8.24**	**6.79**	**3.84**	**2.39**

A.9 Details of Implementation and Dataset

Regarding additional implementation details, we utilize the AdamW optimizer with parameters $\beta_1 = 0.9$, $\beta_2 = 0.999$. The model is trained at bf16 precision, equipped with DeepSpeed-Satge-2 for distributed data parallel training.

Regarding the training dataset, our training dataset consists of three parts, including Hallo3 [11], Celebv-HQ [91], and collected videos from the internet (Bilbil, YouTube, and TikTok). We leverage SyncNet [10] and Q-Align [81] to filter for higher-quality videos by assessing lip-sync accuracy with audio and video fidelity. We also employ InsightFace [12] to filter out videos with a facial confidence score below 0.9. Ultimately, we obtain the refined training dataset, containing approximately 1200 hours of videos. The self-collected dataset contains approximately 22800 video clips from 8200 unique identities. The content spans interview/podcast speech (35%), daily vlog (25%), educational/presentation videos (20%), and music/performance clips (20%), covering diverse ethnicities (Asian 30%, Caucasian 28%, African 22%, Hispanic 20%), genders (male 42%, female 58%), and age groups (18–60). No personally identifiable metadata is retained beyond face regions and voice content. Rigorous deduplication via pHash (hamming distance threshold 10) and CLIP embedding similarity (threshold < 0.92) confirms zero overlap between training data and all test sets.

In terms of the testing dataset, we select 100 unseen videos (2-5 minutes long, FPS=30) from the internet to construct the testing dataset Long100. Some examples are shown in Fig. 10. The sources of videos come from numerous social media platforms, including YouTube, TikTok, and BiliBili. These videos showcase individuals across ethnicities, genders, portrayed in full-body, half-body, and close-up shots against varied indoor and outdoor settings. In contrast to existing open-source testing datasets (HDTF and AVSpeech), our Long100 contains relatively complicated audio rhythms and intricate protagonist appearances. Moreover, the average duration of Long100 (2-5 minutes) is significantly longer than existing open-source testing datasets (2-60 seconds). Long100 also involves multiple human-object interactions, such as instruments, making it more challenging to maintain identity consistency. The licenses of our training dataset, the open-source testing dataset HDTF, the open-source testing dataset AVSpeech, and our self-collected Long100 are all Apache-2.0.

The licenses of the open-source testing datasets HDTF and AVSpeech are GPL-3.0 and CC BY 4.0. For our self-collected training videos and Long100, they are under a custom non-commercial research-only agreement (same as CelebV-HQ), gated behind an access request form, which is consistent with the non-commercial academic consent obtained from uploaders.



Figure 10: Examples from Long100.

Table 9: Ablation study on Text/Image CFG.

Model	FVD↓	CSIM↑	Sync-C↑	Sync-D↓	GPU Mem↓
Ours+Text CFG	498	0.849	8.24	6.78	25.2G
Ours+Image CFG	502	0.850	8.25	6.77	25.2G
Ours+Text CFG+Image CFG	496	0.850	8.26	6.77	32.6G
Ours	504	0.849	8.24	6.79	18.4G

Table 10: Ablation study on α and β.

Model	FVD↓	CSIM↑	Sync-C↑	IQA↑
$\alpha = 1, \beta = 1$	826	0.824	7.52	3.74
$\alpha = 4.5, \beta = 1$	682	0.841	7.62	3.81
$\alpha = 1, \beta = 3$	715	0.829	8.01	3.76
$\alpha = 4.5, \beta = 3$	**504**	**0.849**	**8.24**	**3.84**
$\alpha = 7.5, \beta = 3$	588	0.852	8.21	3.65
$\alpha = 4.5, \beta = 5$	612	0.845	8.35	3.58

A.10 Additional Ablation on Audio Native Guidance

Table 9 shows that Text CFG and Image CFG substantially increase GPU consumption and do not significantly improve visual quality. Thus, we do not incorporate the reference image and text prompt as guidance factors. The root cause is that StableAvatar's latent space is already strongly anchored to identity (via the reference image) and scene context (via the text prompt), which are easier for the model to learn than audio patterns. In contrast, the audio-latent relationship is the dynamic bottleneck of the entire generation process. It is the primary source of variance and, more importantly, the root cause of latent distribution drift over time. By focusing our guidance exclusively on the joint audio-latent distribution (Audio Native Guidance), we prioritize the refinement of the most volatile signal while avoiding the over-saturation of static priors.

We further ablate α and β as shown in Table 10. We can see that the performance is relatively robust within a reasonable range, with the optimal balance found at $\alpha = 4.5, \beta = 3$. When β is low, the lip sync noticeably degrades even with a high α. Excessively high values for both lead to guidance collapse, where video quality deteriorates due to artifact generation.

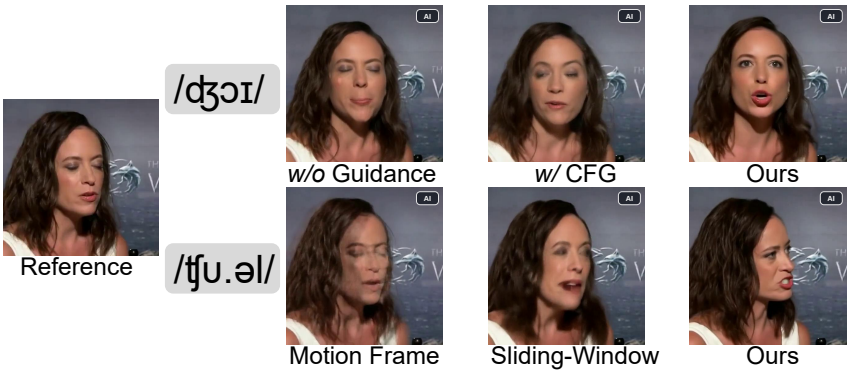

Figure 11: Ablation study on long video generation strategies.

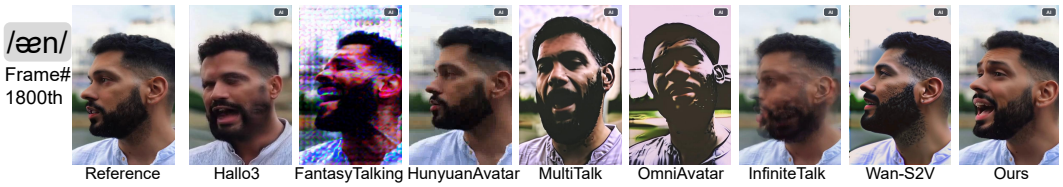

Figure 12: Additional comparison results.

Table 11: Comparison results on EMTD.

Model	FID↓	FVD↓	SSIM↑	PSNR↑	Sync-C↑	CSIM↑
EchoMimicV2	33.42	217.71	0.662	18.17	4.44	0.519
EMO2	27.28	129.41	0.662	17.75	4.58	0.650
FantasyTalking	22.60	178.12	0.703	19.63	3.00	0.626
HunyuanAvatar	18.07	145.77	0.670	18.16	4.71	0.583
Wan-S2V	**15.66**	129.57	**0.734**	20.49	4.51	0.677
Ours	15.71	**125.64**	0.730	**20.51**	**4.76**	**0.731**

Table 12: Temporal consistency comparison on Long100. Subject Consistency (SC), Background Consistency (BC), and Temporal Flickering (TF) are from VBench [29]. E-FID is from Hallo3 [11]. Motion Frame [11], Context Frame [84], and Sliding Window [30] are the same configuration settings as Table 5.

Model	Warping Error↓	SC↑	BC↑	TF↑	E-FID↓
InfiniteTalk [84]	0.0486	0.924	0.942	0.909	72.56
Wan-S2V [16]	0.0724	0.871	0.885	0.941	65.32
Motion Frame [11]	0.0392	0.943	0.951	0.938	56.24
Context Frame [84]	0.0347	0.957	0.959	0.943	54.35
Sliding Window [30]	0.0386	0.944	0.955	0.938	54.81
Ours	**0.0298**	**0.966**	**0.973**	**0.983**	**50.48**

A.11 Additional Ablation on DWSW

Fig. 11 shows the qualitative results of ablation study on long video generation strategies. We further conduct an ablation study on the weight assignment of our Dynamic Weighted Sliding Window Denoising Strategy (DWSW),as shown in Table 8. Fixed weights set all weights of context window to 0.5 and Uniformed gap refers to the weights of each element in the context window following an arithmetic progression. We can observe that our logarithmic weighting function achieves the best performance, indicating that the logarithmic function prioritizes the fusion of elements from the previous context window in the early stages, while progressively and smoothly incorporating both context windows in the later stages, thereby facilitating the overall long video smoothness.

Table 13: Comparisons in inference latency and GPU resources. Inference latency (minutes) and GPU memory are measured for generating 81 frames at 480×832 resolution.

Method	CSIM↑	Sync-C↑	Sync-D↓	GPU Mem↓	Speed↓
Hallo3 [11]	0.462	4.42	9.92	49.8G	13.72
FantasyTalking [77]	0.468	1.92	11.78	44.7G	16.40
HunyuanAvatar [6]	0.472	4.34	10.07	26.6G	21.98
MultiTalk [33]	0.465	4.12	10.18	49.21G	23.0
OmniAvatar [15]	0.471	4.45	9.62	40.83G	20.31
InfiniteTalk [84]	0.755	7.92	7.05	42.7G	26.24
Wan-S2V [16]	0.477	4.88	9.68	55.3G	25.91
Ours	**0.849**	**8.24**	**6.79**	**18.4G**	**2.32**

Table 14: Inference speed and GPU memory comparisons against the raw backbone. Inference latency (minutes) and GPU memory are measured for generating 81 frames at 480×832 resolution.

Model	GPU Mem (G)↓	Speed (Minutes)↓
Wan-1.3B [74]	15.4	1.26
EchomimicV3 [45]	30.6	5.43
Ours	18.4	2.32

Table 15: Computational complexity comparison. FLOPs and FPS are measured for generating 81 frames at 480×832 resolution.

Model	Backbone	FLOPs↓	FPS↑
EchomimicV3 [45]	Wan2.1-1.3B	276.8T	0.25
OmniAvatar [15]	Wan2.1-14B	1127T	0.07
InfiniteTalk [84]	Wan2.1-14B	3255T	0.05
Wan-S2V [16]	Wan2.2-14B	2257T	0.05
Ours	Wan2.1-1.3B	**177.5T**	**0.58**

A.12 Additional Comparison Results

Fig. 12 shows an additional comparison result, indicating the superiority of our StableAvatar over previous audio-driven avatar video generation models. Table 11 shows the comparison results on the EMTD [46] dataset.

To validate the temporal consistency, we further compare our StableAvatar with previous audio-driven avatar video generation models and previous long video generation strategies on Long100 regarding Warping Error [34], Subject Consistency (SC), Background Consistency (BC), and Temporal Flickering (TF) from VBench [29], and E-FID from Hallo3 [11]. The results are in Table 12. We can see that ours also outperforms other models regarding emotion-audio alignment and temporal consistency.

A.13 Speed and GPU Resource

We compare the inference speed and GPU memory consumption between our StableAvatar and previous models, as shown in Table 13. Despite being built on a significantly smaller backbone (Wan2.1-1.3B), StableAvatar achieves superior face quality and lip synchronization compared to prior Wan2.1-14B-based models [33, 15, 84] and Wan2.2-14B-based models [16], while requiring only half the memory and achieving a tenfold increase in inference speed over the leading competitor Wan-S2V [16]. This demonstrates its strong advantage in long-form avatar video generation.

We also conduct the inference speed and GPU Memory Consumption comparison experiments against the raw backbone (Wan-1.3B) and Wan-1.3B-based avatar video generation model EchomimicV3 [45], as shown in Table 14 (81 frames at 480×832 resolution). We observe that, compared to the raw backbone, StableAvatar introduces only modest overhead in terms of GPU usage and speed, achieving a favorable cost-to-benefit balance.

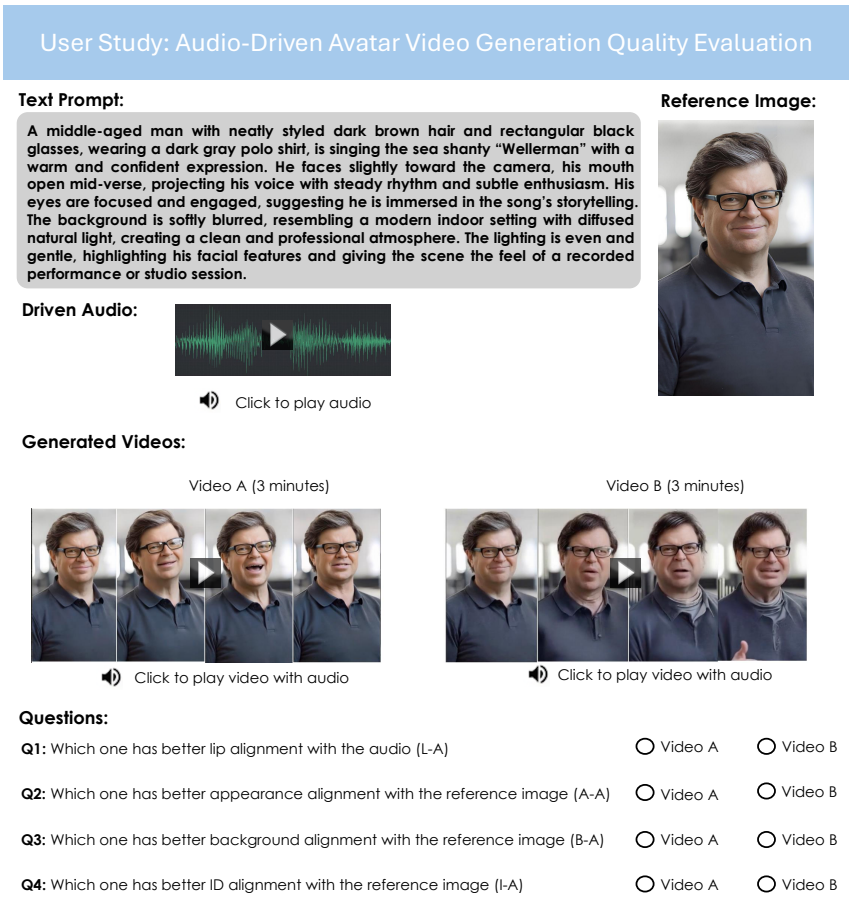

Figure 13: The user study screenshot.

Table 16: User preference of StableAvatar compared to other competitors. A higher score indicates users prefer more to our model.

Method	L-A	A-A	B-A	I-A
HunyuanAvatar [6]	96.2%	94.5%	94.6%	97.5%
MultiTalk [33]	95.5%	95.2%	94.2%	96.4%
OmniAvatar [15]	94.6%	95.6%	93.8%	95.8%
InfiniteTalk [84]	91.7%	90.5%	94.2%	89.6%
Wan-S2V [16]	88.4%	92.8%	93.3%	95.6%

Table 17: Performance comparison on extremely long video (hours).

Duration	FVD↓	CSIM↑	Sync-C↑	Sync-D↓	IQA↑	ASE↑
0.1 hours	508	0.849	8.22	6.79	3.84	2.37
2 hours	542	0.843	8.17	6.85	3.81	2.35
4 hours	585	0.839	8.10	6.93	3.78	2.34

We further compare StableAvatar with other models regarding computational complexity, as shown in Table 15. The FLOPs are measured for generating 81 frames at 480×832 resolution. StableAvatar achieves the lowest FLOPs (177.5T) due to the 1.3B backbone and the relatively lightweight Audio Adapter, while maintaining the highest FPS (0.58). Notably, InfiniteTalk requires 18.3× more FLOPs than ours yet produces inferior quality (Table 1).

Table 18: VBench [29] and CSIM evaluation with temporal segmentation on 100 videos. For each video duration, we separately evaluate the first, middle, and last 10-minute segments to directly measure temporal shift. Δ denotes the relative drop (%) from First to Last within each duration. The 100 videos use reference images from the Long100, and the audio for each case is constructed by concatenating multiple podcast, movie, and music clips sharing the same ID.

Duration	Segment	CSIM↑	Subject Cons.↑	BG Cons.↑	Temp. Flicker.↑	Motion Smooth.↑
2 hours	First 10 min	0.845	0.959	0.966	0.979	0.985
	Mid 10 min	0.840	0.955	0.963	0.977	0.983
	Last 10 min	0.835	0.951	0.960	0.975	0.981
	Δ	1.2%	0.8%	0.6%	0.4%	0.4%
4 hours	First 10 min	0.838	0.955	0.961	0.977	0.983
	Mid 10 min	0.826	0.946	0.953	0.972	0.978
	Last 10 min	0.818	0.939	0.947	0.969	0.975
	Δ	2.4%	1.7%	1.5%	0.8%	0.8%
6 hours	First 10 min	0.832	0.948	0.956	0.973	0.979
	Mid 10 min	0.814	0.934	0.944	0.965	0.972
	Last 10 min	0.802	0.923	0.935	0.959	0.966
	Δ	3.6%	2.6%	2.2%	1.4%	1.3%
8 hours	First 10 min	0.825	0.944	0.953	0.971	0.977
	Mid 10 min	0.804	0.926	0.937	0.959	0.967
	Last 10 min	0.792	0.913	0.926	0.951	0.960
	Δ	4.0%	3.3%	2.8%	2.1%	1.7%
10 hours	First 10 min	0.818	0.940	0.949	0.968	0.974
	Mid 10 min	0.798	0.918	0.930	0.954	0.962
	Last 10 min	0.789	0.904	0.918	0.944	0.954
	Δ	3.5%	3.8%	3.3%	2.5%	2.1%

Table 19: Human evaluation on long video quality at different durations (5-point Likert scale, higher is better). We report mean $\pm$ s.d. (standard deviation) over all ratings (50 participants $\times$ 30 videos = 1500 ratings per cell). For videos exceeding 4 hours, participants watched only the first, the middle, and the last one hour rather than the full video, and rated based on both segments. The 30 videos use reference images from the user study, and the audio for each case is constructed by concatenating multiple podcast, movie, and music clips sharing the same ID.

Duration	ID Consistency↑	Lip Sync↑	Visual Quality↑	Temporal Smoothness↑
0.1 hours	4.72±0.31	4.68±0.38	4.65±0.34	4.70±0.27
2 hours	4.65±0.37	4.56±0.45	4.55±0.40	4.65±0.32
4 hours	4.57±0.42	4.49±0.41	4.43±0.47	4.56±0.36
6 hours	4.46±0.39	4.40±0.52	4.36±0.43	4.49±0.40
8 hours	4.40±0.46	4.33±0.49	4.27±0.51	4.41±0.37
10 hours	4.33±0.50	4.25±0.55	4.21±0.48	4.32±0.43

A.14 User Study Results

Table 16 shows the user study results, indicating the superiority of our model in terms of subjective evaluation. Fig. 13 shows a screenshot of our user study. The 30 test videos were randomly sampled from our Long100 dataset, which consists of 100 unseen videos (2–5 minutes long, FPS=30) collected from diverse social media platforms (YouTube, TikTok, BiliBili). These videos span various ethnicities, genders, body framing (full-body, half-body, close-up), and indoor/outdoor settings, as described in Sec.A.9 in Appx. The cases shown in Fig. 10 are also samples of our user study videos.

The total sample size was 200 participants. The recruitment criteria required participants to be adults (aged 18 or above) with normal or corrected-to-normal vision and no hearing impairments. Participants were recruited through internal university mailing lists and campus announcements at our institution. In terms of composition, participants were primarily university students (75%) and faculty or research staff (25%), spanning a range of academic backgrounds including computer science, engineering, and the arts, with roughly balanced gender representation. We adopted a two-alternative forced choice protocol. In each trial, participants were first presented with a reference image and the corresponding audio clip. They were then shown two videos in randomized order, one generated by StableAvatar and the other by a competitor, and asked to select which video exhibited better quality



Figure 14: The attention map visualization of Timestep-aware Audio Adapter in multi-avatar animation scenarios.

Table 20: Corner case analysis on multi-avatar lip synchronization under varying speech overlap ratios.

Speech Overlap	Method	Per-Subject Sync-C↑ (A/B)	Per-Subject Sync-D↓ (A/B)	CSIM↑	Confusion Rate↓
15%	*w/o* Audio Adapter	5.82 / 5.76	9.34 / 9.42	0.715	7.2%
	HunyuanAvatar [6]	6.95 / 7.08	8.56 / 8.42	0.748	4.5%
	MultiTalk [33]	7.28 / 7.36	8.24 / 8.12	0.742	3.8%
	Ours	**8.12 / 8.06**	**6.86 / 6.92**	**0.868**	**2.4%**
30%	*w/o* Audio Adapter	4.95 / 4.82	10.12 / 10.24	0.686	14.5%
	HunyuanAvatar [6]	6.28 / 6.38	9.18 / 9.08	0.722	10.2%
	MultiTalk [33]	6.72 / 6.82	8.78 / 8.68	0.715	8.6%
	Ours	**7.88 / 7.80**	**7.06 / 7.14**	**0.862**	**3.8%**
60%	*w/o* Audio Adapter	3.72 / 3.58	11.24 / 11.38	0.645	22.4%
	HunyuanAvatar [6]	5.18 / 5.28	10.38 / 10.24	0.688	17.2%
	MultiTalk [33]	5.82 / 5.92	9.72 / 9.58	0.681	15.4%
	Ours	**7.48 / 7.38**	**7.42 / 7.52**	**0.855**	**6.5%**
90%	*w/o* Audio Adapter	2.48 / 2.34	12.56 / 12.72	0.598	34.2%
	HunyuanAvatar [6]	3.72 / 3.84	11.56 / 11.42	0.642	25.8%
	MultiTalk [33]	4.08 / 4.18	11.02 / 10.86	0.648	23.5%
	Ours	**7.05 / 6.94**	**7.76 / 7.86**	**0.846**	**8.6%**

along four dimensions (lip alignment with audio, appearance alignment with reference, background consistency with reference, and identity preservation). The 30 test cases were randomly sampled from our Long100 dataset, which spans diverse ethnicities, genders, body framings (full-body, half-body, close-up), and indoor/outdoor settings. Each participant evaluated all 30 cases, and the presentation order of the two videos within each case was randomized to eliminate ordering bias. Responses were collected through a web-based survey platform, and the results were analyzed by computing the preference rate for each method pair across all participants and cases, as reported in Table 16. We confirm that we consulted our university institutional ethics board prior to conducting the study. Because the study met all the following conditions, it was determined to qualify for exemption from full IRB review under our institution's guidelines for minimal-risk research. First, the study involved only the perceptual comparison of pre-generated video outputs and did not involve any intervention, deception, or interaction beyond viewing videos and selecting preferences. Second, no personally identifiable information (such as names, email addresses, or demographic details beyond aggregate statistics) was collected or stored. Third, participation was entirely voluntary, and participants were informed of the study's purpose and their right to withdraw at any time before beginning.

A.15 Full/Half-Body Avatar Video Results

We perform a qualitative experiment in audio-driven full/half-body avatar animations, as shown in Fig. 15. Each reference protagonist interacts with an object, such as an apple or a piano, making it more challenging to preserve appearance consistency and lip synchronization with the given audio. We can see that our model is capable of synthesizing full/half-body avatar videos, even involving interactions with external objects.

A.16 Multi-Avatar Animation

To demonstrate the robustness of our StableAvatar, we experiment on a particular reference image involving multiple protagonists, as shown in Fig. 16. We can see that our StableAvatar is also capable of handling audio-driven multiple-avatar animations while preserving the original identity and achieving high video fidelity.

Furthermore, we conduct additional experiments to validate that our Audio Adapter can map a single audio signal to multiple characters without identity interference. Fig. 14 shows the attention map visualization of our Timestep-aware Audio Adapter in a two-person conversation scene. We emphasize that our model receives only the unseparated mono audio as input during inference. To enable the clearness to verify the per-subject selective attention behavior as clearly as possible, we deliberately performed offline source separation on the original mono input to obtain two individual speaker waveforms (Speaker A in blue and Speaker B in red), which are displayed below the video frames purely as a visualization reference rather than as model inputs. In the figure, in frames 2 and 4 where Speaker A (the man) is the sole active speaker, the attention heatmap concentrates sharply on his lip and jaw region while Speaker B's face remains unactivated, aligned with the active blue waveform and silent red waveform. In frames 1 and 3 when Speaker B (the woman) speaks, the hot attention region shifts to her mouth area, synchronized with the active red waveform and the silent blue waveform. Notably, in frame 5 where both subjects vocalize simultaneously from the single mono input, the attention map reveals two distinct activation clusters centered on each subject's mouth, with both waveforms concurrently active. The key insight lies in the latent-dependent nature of the refined audio embeddings. Concretely, in our Audio Adapter (Eq. 3), the audio embeddings serve as Query and cross-attend with the diffusion latents z_t (Key and Value): $\text{CAttn}(\text{LN}(\text{emb}^{\gamma}_{\text{aud}}), z_t)$. Since z_t inherently encodes spatially distinct information for each subject in the scene, the resulting refined audio embeddings $\bar{a}_t$ are no longer a subject-agnostic audio representation and becomes a joint audio-spatial-latent variable that implicitly disentangles which audio patterns correspond to which spatial regions. When multiple voices coexist in the audio (e.g., a multi-person conversation), Wav2Vec captures the full spectral content of all speakers. Through cross-attention with the spatially-aware latents, the Audio Adapter learns to associate distinct voice patterns with the corresponding subject regions in the latent space, effectively performing an implicit audio-visual source separation. This is further reinforced when the refined audio embeddings are injected back into the DiT via $\bar{z}_t = \text{CAttn}(z_t, \bar{a}_t)$ (Eq. 5). Each spatial position in z_t selectively attends to the audio cues relevant to its corresponding subject, preventing identity interference. Fig. 14 confirms that the cross-attention mechanism (refined audio embeddings from Audio Adapter and latents) dynamically responds to distinct audio-visual correspondences rather than applying a fixed spatial template.

For the corner case analysis, we constructed a dedicated test set of 100 multi-avatar clips (Speaker A and Speaker B) with controlled speech overlap ranging from 15% to 90% and measured per-subject lip synchronization alongside a Subject Identity Confusion Rate, defined as follows. For each evaluation window, we first apply offline source separation to obtain per-subject audio tracks, then compute the SyncNet confidence for both the normal pairing Sync-C($\text{lip}_A, \text{audio}_A$) and the swapped pairing Sync-C($\text{lip}_A, \text{audio}_B$). A frame is marked as confused if Sync-C($\text{lip}_A, \text{audio}_B$) $>$ Sync-C($\text{lip}_A, \text{audio}_A$) or Sync-C($\text{lip}_B, \text{audio}_A$) $>$ Sync-C($\text{lip}_B, \text{audio}_B$), and the Confusion Rate equals the ratio of confused frames to total frames. We also include the ablation variant "w/o Audio Adapter" that removes the Audio Adapter and directly injects raw Wav2Vec embeddings into the DiT via cross-attention. As shown in Table 20, our method maintains per-subject Sync-C near 7.0 even at 90% overlap, with 13.5% relative degradation from the 15% baseline, whereas MultiTalk and HunyuanAvatar degrade by 43.6% and 46.1% respectively, and w/o Audio Adapter degrades by 58.4%. More critically, the Confusion Rate of our model remains below 9% across all overlap levels, while all competitors exceed 23% and w/o Audio Adapter reaches 34.2% at 90% overlap. The results demonstrate that the Audio Adapter's cross-attention with spatially-aware diffusion latents genuinely learns to disentangle per-subject audio cues from the mono signal through the latent-dependent refinement mechanism, rather than depending on distributional shortcuts. w/o Audio Adapter further confirms this, as removing the Audio Adapter forces the model to fall back on dataset priors and leads to catastrophic performance collapse at high overlap.

A.17 Cartoon Avatar Video Generation

To validate the diversity of our StableAvatar, we perform experiments on two reference images involving cartoon protagonists, as shown in Fig. 17. We can observe that our StableAvatar can also handle audio-driven cartoon avatar video generation, highlighting the diversity of our StableAvatar.

A.18 Long Avatar Video Results

To rigorously validate the ultra-long generation capability, we conduct a comprehensive evaluation spanning video durations from 0.1 to 10 hours using multi-GPU inference. Table 17 reports 6

Table 21: Ablation on integrating external identity encoders (InsightFace [12] and DINOv3 [53]) with StableAvatar on Long100. SC, BC, TF are from VBench [29]. The baseline removes all of our proposed modules.

Method	CSIM↑	FVD↓	SC↑	BC↑	TF↑	GPU Mem↓	Speed (min)↓
Baseline + InsightFace	0.468	1812	0.902	0.912	0.890	19.6G	1.86
Baseline + DINOv3	0.482	1756	0.908	0.918	0.896	20.2G	1.95
Ours	0.849	504	0.966	0.973	0.983	18.4G	2.32
Ours + InsightFace	0.855	488	0.967	0.974	0.983	22.6G	3.12
Ours + DINOv3	**0.862**	**478**	**0.969**	**0.976**	**0.985**	23.2G	3.25

task-specific metrics, where CSIM assesses identity consistency, Sync-C/Sync-D measure lip synchronization fidelity, and FVD/IQA/ASE evaluate overall video quality. Table 18 further adopts VBench [29] with a temporal segmentation protocol, where Subject Consistency (DINO [53]-based cross-frame identity similarity) and Background Consistency directly quantify identity preservation, while Temporal Flickering and Motion Smoothness capture temporal shift through inter-frame perceptual variation. For each duration, we separately evaluate the first, middle, and last 10-minute segments to reveal whether degradation is localized or uniformly distributed. Table 19 complements the metrics with a dedicated human evaluation. We recruited 50 participants (adults aged 18+, normal or corrected-to-normal vision, no hearing impairments) through university mailing lists and campus announcements. The participants comprised university students (72%) and faculty/research staff (28%) with roughly balanced gender representation. Each participant rated 30 videos per duration on four perceptual dimensions (ID Consistency, Lip Sync, Visual Quality, Temporal Smoothness) using a 5-point Likert scale. For each trial, participants were first shown the reference image and audio, then the generated video, and asked to rate each dimension independently. For videos exceeding 4 hours, participants watched the first, the middle, and the last one hour rather than the full video. Responses were collected via a web-based survey platform. This study followed the same IRB-exempt protocol as our user study (Appx.A.14), involving only perceptual rating of pre-generated outputs with no personally identifiable information collected. The results show that degradation follows a sub-linear trajectory rather than a super-linear one, confirming that per-clip errors are bounded and do not compound multiplicatively. This bounded error property, stemming from the Audio Adapter's per-clip error suppression and DWSW's smooth inter-clip fusion, implies that StableAvatar can sustain generation with graceful degradation rather than catastrophic collapse, theoretically supporting ultra-long generation.

Fig. 18, Fig. 19, Fig. 20, Fig. 21, Fig. 22, and Fig. 23 show additional audio-driven long avatar videos synthesized by our StableAvatar. **Notably, each audio lasts over 3 minutes, and with an FPS of 30, this results in 5400+ synthesized frames. The duration of the first case (the first raw) in Fig. 23 is approximately 2 hours. We only display selected frames from the last 10 seconds for brevity.** Each case contains complex protagonist's appearance and an intricate audio rhythm pattern. We can see that our StableAvatar can perform a wide range of audio-driven avatar animation while simultaneously preserving the protagonist's appearance, background, and identity, even after synthesizing 5000+ frames. For example, in the reference image in the third row of Fig. 18, the protagonist wears a mask, making it particularly challenging for the model to perform accurate lip synchronization with audio. Moreover, the reference protagonists in the fourth row and the sixth row of Fig. 21 face the camera at an angle, making it dramatically difficult for the avatar animation model to preserve ID consistency. It is noticeable that our StableAvatar can accurately manipulate the lip, facial expression, and body gesture in the reference image while preserving high-quality identity consistency, even in specific cases involving head shaking and external object interaction. Moreover, StableAvatar supports simultaneously animating the avatar and background following the given audio via text prompt description. For example, in the reference images in the second row of Fig. 18 and the sixth row of Fig. 21, StableAvatar utilizes text prompts to animate contextual elements—for instance, making the scenery outside a car window shift as the vehicle starts moving, or generating dynamic ocean waves along the coastline.

A.19 Discussion on ID consistency in Long Avatar Video Generation

Although StableAvatar does not introduce a dedicated identity encoder module [88, 12, 53], we argue that ID drift in long avatar video generation is not a standalone problem that can be resolved

by simply appending an external identity extractor. It is fundamentally entangled with latent drift. Even with a dedicated identity encoder providing ID conditioning, latent error accumulation across clips will gradually push the latent distribution away from the identity-aligned region, causing facial distortion in long videos. Table 4 shows that the baseline (*w/o* Audio Adapter) sees CSIM collapse from 0.836 to 0.405 over 3500 frames, demonstrating that the primary cause of ID degradation in long videos is latent drift rather than the absence of an explicit ID module.

The ability of StableAvatar to maintain identity consistency can be attributed to several complementary factors. (1) As described in Eq. 11, we explicitly introduce face masks $\boldsymbol{M}_{face}$ and lip masks $\boldsymbol{M}_{lip}$ during training to provide region-level supervision on the facial area, encouraging the model to attend to fine-grained facial identity features. This shares a similar motivation with approaches that leverage external face encoders, as both aim to explicitly model the facial region. The difference is that face encoders provide an additional identity-aware conditioning signal, while our mask supervision directs the learning objective toward facial regions during training. (2) The reference image is injected through both a frozen 3D VAE Encoder and a CLIP Image Encoder, providing complementary structural and semantic identity conditioning. (3) Most critically, the Audio Adapter suppresses per-clip latent drift (Table 4, CIEDE from 2.318 to 0.523), which indirectly but substantially preserves the identity-relevant latent subspace that would otherwise be corrupted by audio modeling.

The comprehensive evaluation in Table 17, Table 18, and Table 19 validates this from multiple perspectives. Notably, the metrics in Table 18 are not computed globally over the entire video but are measured on sampled segments (first, middle, and last 10 minutes) at each duration, providing a localized assessment of quality at different temporal positions. Even under this segment-level evaluation, Subject Consistency remains 0.904, and CSIM remains 0.789 in the last 10 minutes of a 10-hour video. Table 19 further confirms this with human-rated ID Consistency of 4.33/5.0 at 10 hours, where participants watched the first and last one hour for videos exceeding 4 hours rather than the full video.

To further investigate whether dedicated identity encoders can complement our approach, we conduct additional experiments by integrating InsightFace [12] and DINOv3 [53] as external ID encoders, as shown in Table 21. We observe two findings. First, adding either InsightFace or DINOv3 to the Baseline (without Audio Adapter) yields CSIM of only 0.468 and 0.482, respectively, confirming that even strong identity conditioning cannot compensate for latent drift in long video generation. Second, adding InsightFace or DINOv3 on top of StableAvatar provides marginal improvements (CSIM 0.849→0.855/0.862) at the cost of over 22% additional GPU memory and over 34% slower inference speed. DINOv3 yields slightly better results than InsightFace due to its richer semantic representations, yet the absolute gain remains limited. Given that StableAvatar already achieves CSIM of 0.849 without any external ID module, we consider this trade-off unfavorable and opt not to include the additional ID encoder in our final model.

A.20 Limitation and Future Work

Fig. 24 shows one failure case of our StableAvatar. The reference protagonist is a non-human creature (fantastical creature), which has significant appearance and structure discrepancies with the common human. Our StableAvatar struggles to find its lip, thereby failing to synchronize its lip with the audio. One potential solution is to introduce an additional reference network to explicitly capture the semantic details of the reference. This part is left as future work.

A.21 Human Subjects Data Concern

Our training and testing datasets both involve human subjects, containing identifiable information. Since the collected videos are primarily from BiliBili, TikTok, and YouTube, we contacted the video uploaders via private messages to inquire whether they agreed to allow their videos to be used for non-commercial academic research, and we obtained their consent. All videos in our training and testing datasets have been approved by their corresponding video uploaders. In our data collection protocol, we explicitly required that uploaders confirm they are the depicted subjects or have obtained consent from all depicted individuals before granting permission. Specifically, our consent request message stated: "We confirm that this video will be used solely for non-commercial academic research. Please confirm that you are the person appearing in the video, or that you have obtained consent from



Figure 15: Full/Half-body avatar animation results. The images with red borders are the reference images.

all individuals depicted." Videos where the uploader could not confirm this were excluded. For multi-person videos, we required the uploader to confirm consent for all visible individuals.

A.22 Ethics Concerns

StableAvatar enables the animation of a given reference image driven by input audio, with potential applications in areas such as virtual reality and digital human creation. While this capability offers clear benefits, it also carries risks of misuse, including the creation of deepfake videos for identity impersonation, non-consensual manipulation of a person's likeness, and the spread of misinformation through fabricated speech videos on social media. To mitigate such risks, we emphasize the importance of responsible deployment. All generated content should incorporate visible and invisible watermarks to ensure traceability in public deployment. Outputs should undergo automated sensitive-content detection and moderation (e.g., deepfake detection) before release. API access should require authentication and maintain usage logs, while clear terms of service should prohibit generating content depicting real individuals without their explicit consent. We also set up an access request form for our code, data, and model weights. The access request form will require users to agree to: (1) use solely for academic research, (2) not generate content of real individuals without their explicit consent, (3) apply visible watermarks to all outputs intended for distribution, and (4) integrate deepfake detection before any deployment.



Figure 16: Audio-driven multiple-avatar animation results. The images with red borders are the reference images.



Figure 17: Audio-driven cartoon avatar animation results. The images with red borders are the reference images.

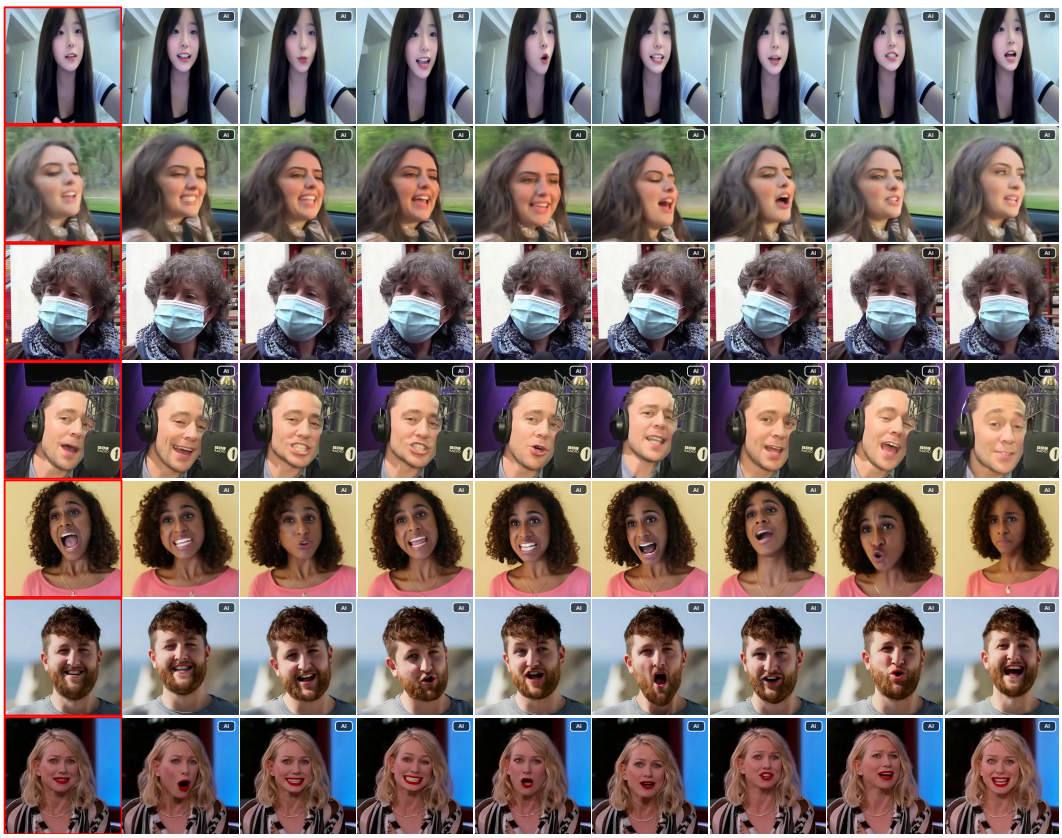

Figure 18: Audio-driven long avatar video results (1/6). The images with red borders are the reference images.



Figure 19: Audio-driven long avatar video results (2/6). The images with red borders are the reference images.

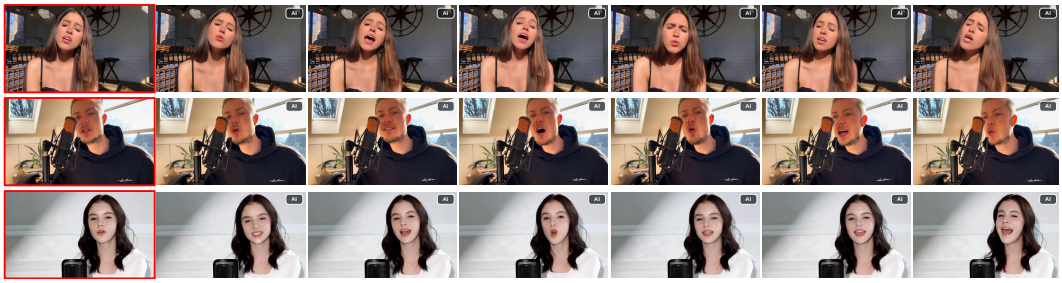

Figure 20: Audio-driven long avatar video results (3/6). The images with red borders are the reference images.

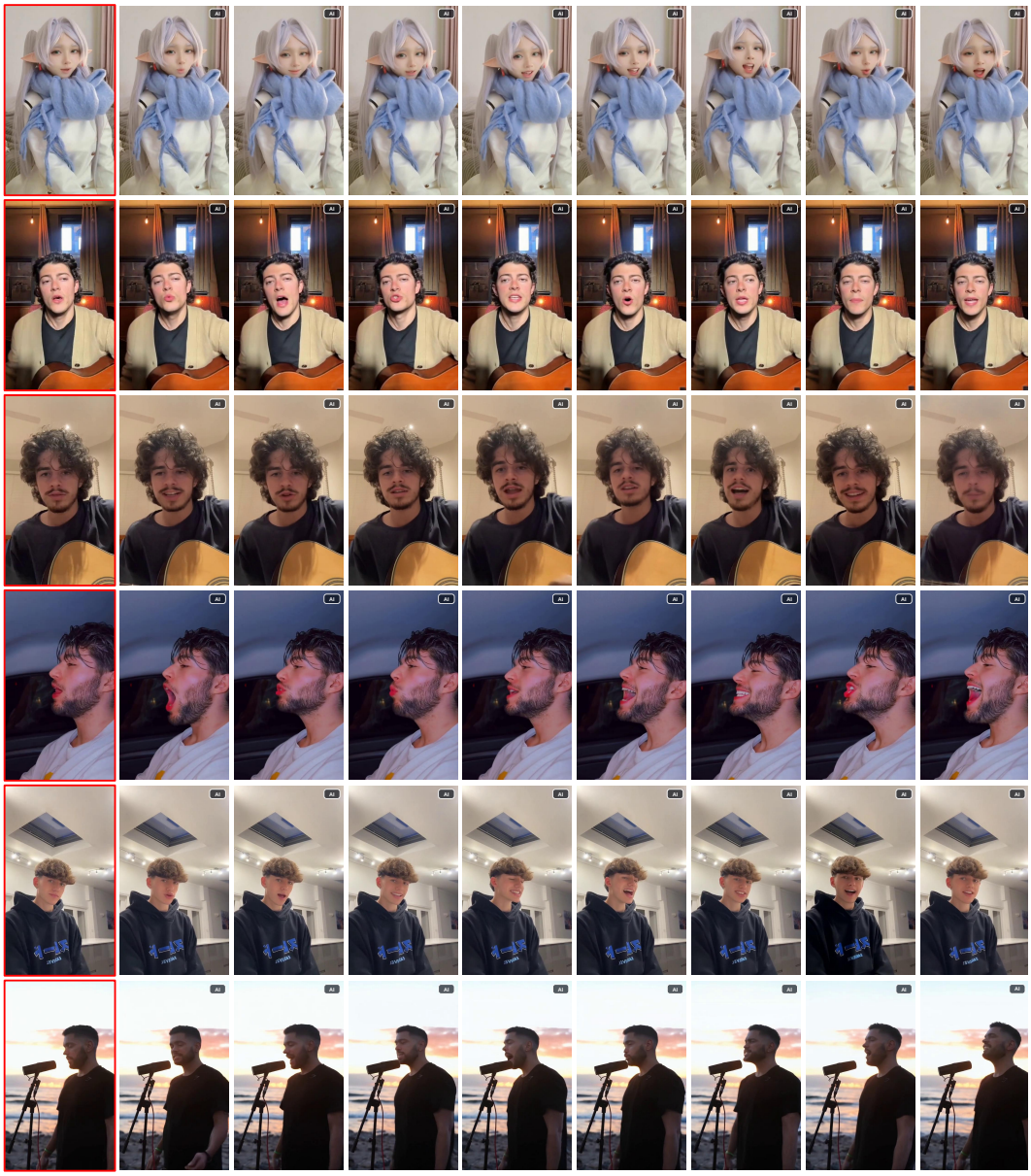

Figure 21: Audio-driven long avatar video results (4/6). The images with red borders are the reference images.

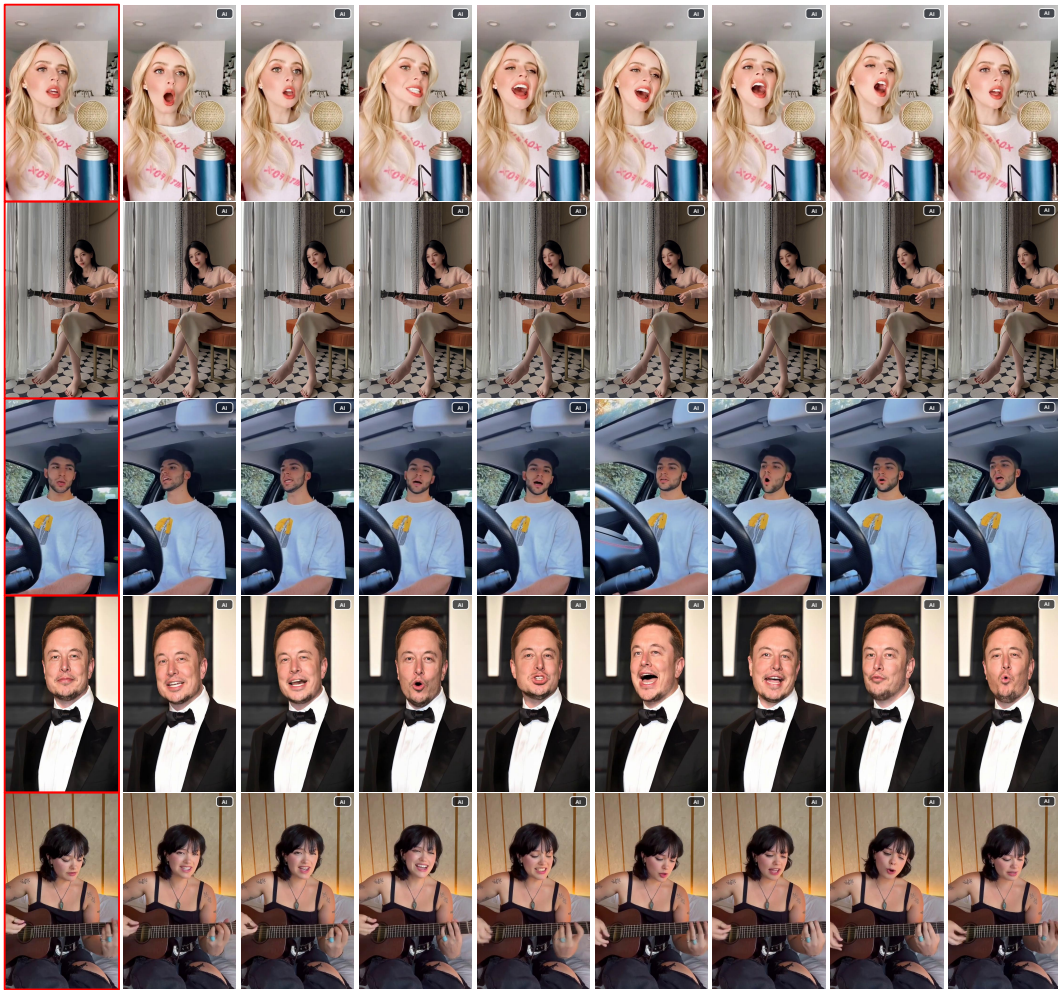

Figure 22: Audio-driven long avatar video results (5/6). The images with red borders are the reference images.

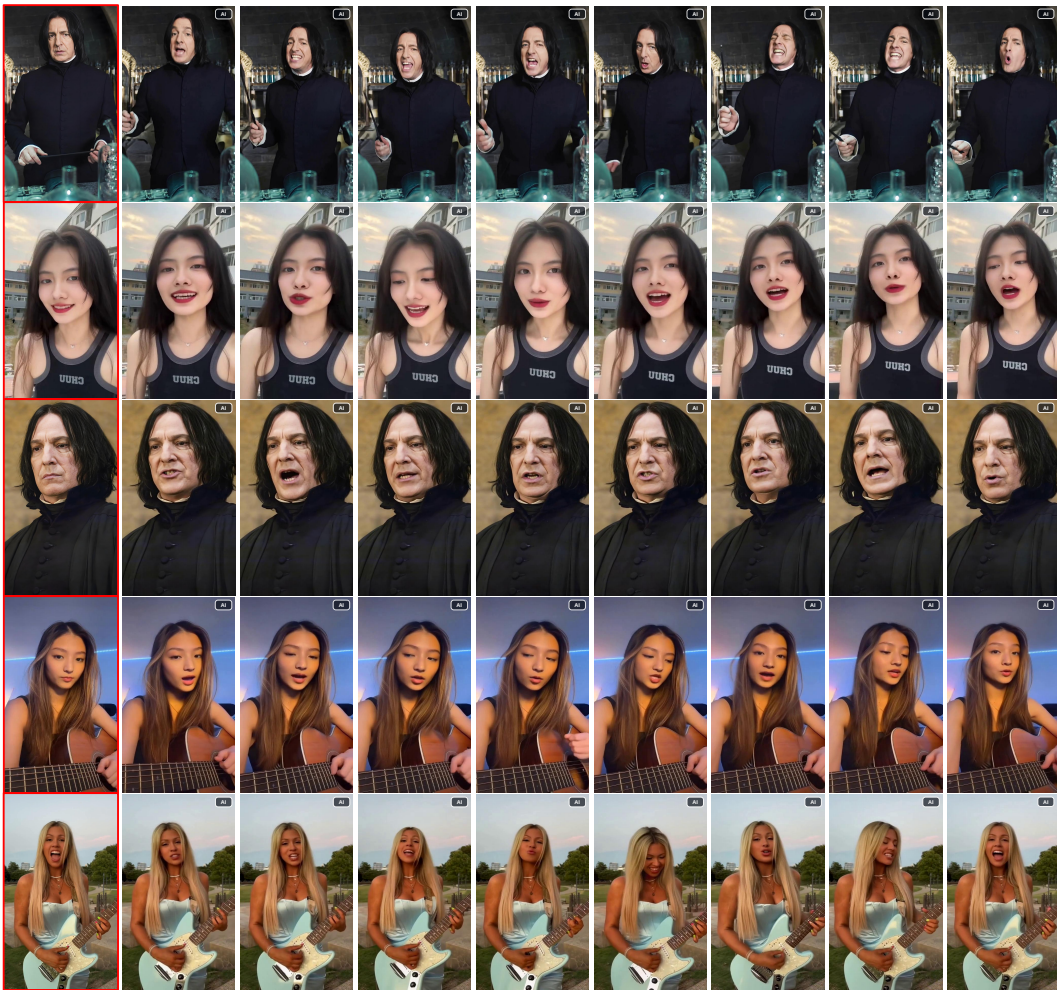

Figure 23: Audio-driven long avatar video results (6/6). The images with red borders are the reference images.



Figure 24: One failure case of our StableAvatar. The images with red borders are the reference images.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Please refer to the Introduction Section and Method Section.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Please refer to Sec. A.20 in the Appx.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: Please refer to Sec. 3 in the main paper.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Please refer to the Sec. 3 and Sec. 4 in the main paper.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [No]

Justification: The code and data are currently undergoing legal review. We will release them publicly once the review is complete. The details of the training and testing datasets are described in Sec. A.9 in the Appendices. The reproducibility details of our method are depicted in the Method Section and Experiment Section in the main paper.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Please refer to Sec. A.9 and Sec. A.5 in Appx.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: Error bars are not reported because it would be too computationally expensive.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).

- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Please refer to Sec. A.13 in Appx.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: Please refer to Sec. A.22 in Appx.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Please refer to Sec. A.22 in Appx.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [Yes]

    Justification: Please refer to Sec. A.22 in Appx.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: Please refer to Sec. A.9 and Sec. A.5 in Appx.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: Please refer to Sec. A.9, Sec. A.5, and Sec. A.21 in Appx.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [Yes]

    Justification: Please refer to Sec. A.14 in Appx.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [Yes]

    Justification: Please refer to Sec. A.14 in Appx.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: The LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigor, or originality of the research.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.