STREAMERASE: TOWARDS ULTRA-FAST STREAMING VIDEO ERASING

Anonymous authors
Paper under double-blind review

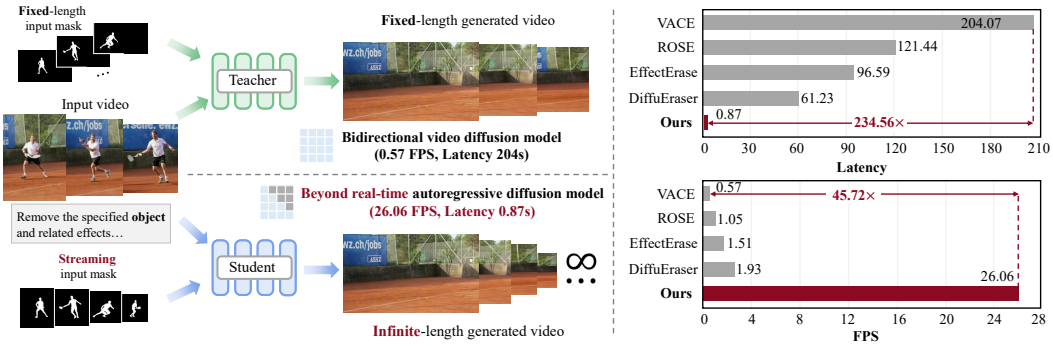


Figure 1: Prior video diffusion erasing models typically operate offline to generate fixed-length sequences in parallel (top left). In contrast, our StreamErase enables streaming long-video generation (bottom left), supporting real-time erasing at 26.06 FPS with 0.87s latency (right).

ABSTRACT

Current video object removal methods suffer from prohibitive latency (minutes per video) and non-causal processing that prevents real-time interaction. We present StreamErase, enabling sub-second latency with up to 26 FPS streaming generation on a single GPU. Our approach first augments a text-to-video model with mask input to generate high-quality videos that erase target objects and synthesize coherent backgrounds, but does not perform inference on the fly. As such, we progressively transfer the erasing capability from a powerful bidirectional foundation model to an efficient unidirectional streaming eraser, enabling long-horizon erasing without sacrificing visual fidelity. Several key challenges arise when generating videos over long, potentially infinite horizons: (1) bridging the domain gap from training on finite length and extrapolating to long horizons, (2) sustaining high quality by preventing error accumulation, and (3) maintaining fast inference, without increasing computational cost as the context grows. To address these challenges, we carefully designed sliding-window causal attention with attention sinks. By incorporating self-rollout with attention sinks and KV cache rolling during training, we simulate inference-time extrapolations with a fixed context window, enabling constant-speed generation of arbitrarily long videos. Finally, we establish a dedicated benchmark, *LongErase*, for streaming long-video object removal. Our model achieves strong performance compared to existing video object erasing models and generalizes well to real-world video scenarios, uniquely enabling infinite-length streaming. With StreamErase, users can input a mask and see the erasing results unfold in real-time, delivering a truly interactive experience.

1 INTRODUCTION

The ultimate goal of video object erasing is to empower creators and systems to seamlessly remove unwanted dynamic content from video streams in real time while preserving visual realism. Although recent video diffusion models (Li et al., 2025a; Bian et al., 2025; Zi et al., 2025a; Jiang et al., 2025; Miao et al., 2025; Fu et al., 2026) have made impressive strides toward this goal, the experience remains largely confined to offline, batch-processing workflows.

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

The promise of real-time video erasing is currently hindered by several fundamental constraints. First, generation is too slow for interactive deployment. For example, synthesizing a 5-second video clip with ROSE (Miao et al., 2025) takes 2 minutes, trapping users in frustrating "render-and-wait" delay cycles. Second, existing paradigms are inherently non-causal, since diffusion models process the entire sequence in parallel with bidirectional attention. A user cannot see any partial results until the entire erase is complete. Finally, the inability to generate more than a few seconds of video severely limits the scope for any meaningful or extended creative expression. Together, these constraints (slow, non-causal, and short-duration generation) undermine the potential for a truly interactive creative experience.

To overcome these challenges, we introduce StreamErase, a streaming video erasing framework that progressively transfers the erasing capability of an offline bidirectional diffusion model into an efficient autoregressive generator for real-time interactive erasing.

Specifically, we first fine-tune a pretrained diffusion model with full bidirectional attention, obtaining in a powerful bidirectional video eraser that serves as the teacher for subsequent distillation. We then replace bidirectional attention with causal attention to construct an autoregressive diffusion model, and apply teacher forcing followed by ODE distillation to obtain a causal video eraser requiring four denoising steps. Finally, we perform Distribution Matching Distillation (DMD), using the bidirectional video eraser as the teacher and the causal video eraser as the student, enabling the streaming model to inherit the high-fidelity erasing capability of the offline model while further compressing generation to a single denoising step for real-time inference.

Causal generation, however, is prone to error accumulation and drift over long horizons. Prior work in large language models has shown that initial tokens play a crucial role in stable streaming generation (Xiao et al., 2023). This insight drives our attention sinking mechanism with rolling KV caches, which we incorporate directly into training to simulate inference extrapolation distributions, ensuring stable, indefinite-length generation at constant latency through fixed context windows.

StreamErase achieves 26.06 FPS with sub-second latency on a single H200 GPU. Extensive experiments and ablations demonstrate that StreamErase achieves competitive visual and temporal quality across diverse erasing scenarios, while being more than 46$\times$ faster than recent methods. Furthermore, StreamErase maintains stable visual and temporal quality when extended from short 5-second clips to substantially longer 30-second videos. StreamErase transforms video generation from a passive waiting experience into an active creative process, where users can continuously interact with and guide the generation in real-time.

2 Related Work

2.1 Video Inpainting

Video inpainting aims to reconstruct missing regions specified by a sequence of masks. Early video inpainting methods primarily used 3D CNNs (Chang et al., 2019; Wang et al., 2019; Hu et al., 2020) to model spatial-temporal features, but their limited receptive fields hindered long-range propagation. Subsequently, optical flow (Kim et al., 2019; Li et al., 2020; Zou et al., 2021) and homography Lee et al. (2019); Cai et al. (2022) were introduced to guide pixel propagation. To improve efficiency and accuracy, (Zhou et al., 2023) proposed ProPainter, combining optical flow and attention mechanisms. Recently, with the rise of diffusion models(Ho et al., 2020), diffusion-based video inpainting has emerged (Shi et al., 2024; Zi et al., 2025b). (Li et al., 2025a) proposed DiffuEraser, extending the image inpainting model BrushNet(Ju et al., 2024) to videos via a two-stage training scheme. Despite these advances, existing inpainting models often overlook object effects, resulting in incomplete or visually inconsistent object removal.

Video Object Erasing is a specialized form of inpainting that requires precise modeling of object-induced visual effects to achieve realistic results. Early methods (Yu et al., 2025; Wei et al., 2025; Liu et al., 2025) attempt to adapt image erasing models to video by processing frames individually, but struggle to maintain temporal consistency in dynamic scenes. Minmax-Remover (Zi et al., 2025a) simplifies a pre-trained video generator by removing text inputs and cross-attention layers and introducing a minimax objective. ROSE Miao et al. (2025) introduces a synthesized dataset comprising multiple environments, along with a side-effect mask predictor. EffectErase (Fu et al.,

2026) further provides a large-scale dataset of paired real and synthetic videos and proposes an effect-aware removal framework based on reciprocal learning. Despite their impressive quality, these methods are fundamentally limited to offline processing because they rely on diffusion models with full bidirectional attention. This constraint prevents their use in real-time applications.

2.2 Streaming Autoregressive Video Models.

Streaming autoregressive generation effectively overcomes the latency bottlenecks of offline models. Foundational works like Diffusion Forcing (Chen et al., 2024) and StreamDiffusion (Kodaira et al., 2025) reformulates denoising as block-wise sequential processing, enabling causal decoding. Follow-up works (Yin et al., 2024c; Chen et al., 2025a; Teng et al., 2025; Guo et al., 2025; Huang et al., 2026; Lin et al., 2025b) combined causal attention, KV-cache, and distillation to compress multi-step diffusion into a few denoising steps, enabling efficient online generation. However, these approaches either exhibit severe color drifts beyond the training horizon or require complex long-video finetuning, which poses challenges for controllable video generation.

2.3 Diffusion Distillation

Accelerating reverse diffusion (Song et al., 2023; Wang et al., 2023; Salimans & Ho, 2022; Song & Dhariwal, 2023) is crucial for real-time performance. Early one-step distillation efforts focused on the image domain, utilizing techniques like Rectified Flow (InstaFlow (Liu et al., 2023)), target score distillation (TSD-SR (Dong et al., 2025)), and Distribution Matching Distillation (Yin et al., 2024b;a), alongside progressive adversarial distillation (SDXL-Lightning (Lin et al., 2024)). These advancements have naturally extended to the video domain, enabling efficient one-step generation (AAPT (Lin et al., 2025a)), restoration (SeedVR2 (Wang et al., 2025)), and super-resolution (DOVE (Chen et al., 2025b)). FlashVSR (Zhuang et al., 2025) achieves real-time streaming video super-resolution via a three-stage distillation pipeline designed to progressively accelerate latent rendering. Similarly, PersonaLive (Li et al., 2025b) leverages appearance distillation for live portrait animation. However, these tasks involve low-level pixel mapping or structurally constrained regions. Directly transferring these schemes to general video erasing degrades high-fidelity details.

3 Method

3.1 StreamErase

Overview. In this section, we elaborate the model architecture of StreamErase for conducting video object erasing, as illustrated in Fig. 2. In brief, StreamErase is implemented as an inpainting model continued from the foundation video generative models (the Wan2.1 model (Wan et al., 2025) in this paper). Following the general architecture in diffusion-based inpainting models (Miao et al., 2025), we extend the model input with the original input video together with object masks.

Removal Formulation. Video inputs are first encoded into the latent space using a pretrained VAE (Kingma & Welling, 2013). The source video with objects V^o, the ground-truth background video V^b, and the mask M are encoded into latent representations $\boldsymbol{x}^o$, $\boldsymbol{x}^b$, and $\boldsymbol{x}^m$, respectively.

To construct the noisy input $\boldsymbol{x}_t$ for diffusion training, a clean latent $\boldsymbol{x}$ obtained from the VAE is used. Random noise $\boldsymbol{z} \sim \mathcal{N}(0, I)$ is added through the forward process (Esser et al., 2024):

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

where the timestep $t \in [0, 1]$ is sampled from a logit-normal distribution. The denoising model v_θ is trained to predict the velocity $\boldsymbol{v} = \boldsymbol{x} - \boldsymbol{z}$ from the noisy latent $\boldsymbol{x}_t$, the timestep t, and the condition $\boldsymbol{c}$, with the objective defined as:

$$\mathcal{L}_{\text{diffusion}} = \mathbb{E}_{\boldsymbol{z},\boldsymbol{x},t,\boldsymbol{c}} \left\| v_\theta(\boldsymbol{x}_t, t, \boldsymbol{c}) - \boldsymbol{v} \right\|^2, \tag{2}$$

where the condition $\boldsymbol{c} = [\boldsymbol{x}^o; \boldsymbol{x}^m]$ guides the model to user-specified regions. Here $[\,;\,]$ denotes concatenation along the channel dimension.

Multimodal Region Guidance. To explicitly model spatiotemporal correlations between the affected areas and objects, we introduce a Region Guidance module. Specifically, textual prompt

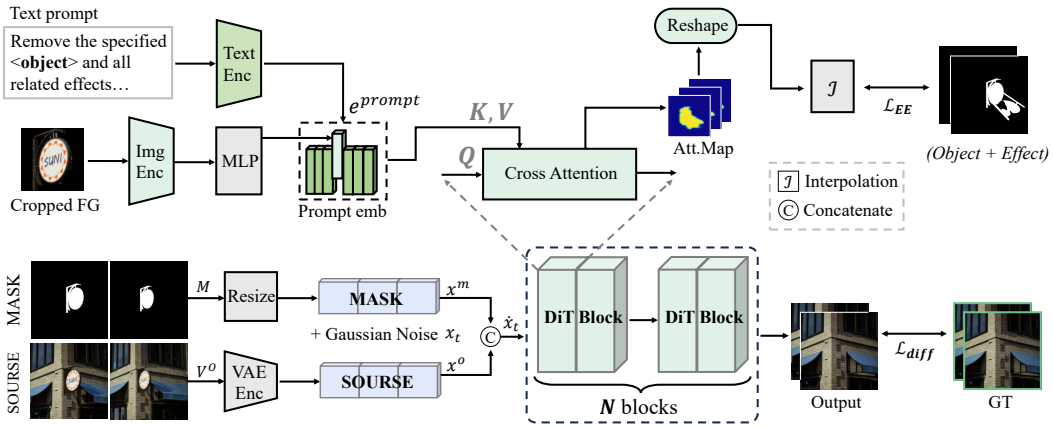


Figure 2: **The model architecture of StreamErase.** We concatenate the noisy latents with the original input video and masks, consumed by a video inpainting model. Each DiT block performs cross-attention using the fused features $\dot{\boldsymbol{x}}_t$ as Query and $\boldsymbol{e}^{\text{prompt}}$ from Region Guidance as Key/Value, producing attention maps that highlight affected regions. We aggregate attention maps from all blocks, which is supervised by the effect erasing loss $\mathcal{L}_{\text{EE}}$ to focus on the affected area.

tokens $\boldsymbol{e}^{\text{prompt}}$ are extracted from a language model (Raffel et al., 2020), Simultaneously, visual foreground features $\boldsymbol{e}^f$ are obtained by feeding a cropped foreground patch from $V^f = V^o \odot M$ into a CLIP image encoder (Radford et al., 2021), where $\odot$ denotes element-wise multiplication. To bridge the modality gap, we employ a lightweight two-layer MLP $\mathcal{M}_\psi(\cdot)$ to map the CLIP embedding into the text token space. The resulting foreground embedding $\mathcal{M}_\psi(\boldsymbol{e}^f)$ then replaces the placeholder token "object" in $\boldsymbol{e}^{\text{prompt}}$, forming a region representation:

$$\boldsymbol{e}^{\text{prompt}}[\text{object}] \leftarrow \mathcal{M}_\psi(\boldsymbol{e}^f), \tag{3}$$

which is injected into the backbone via cross-attention (Vaswani et al., 2017) to guide the model in capturing spatiotemporal effect correlations between the object and its effects, enabling accurate localization of effect-related regions.

Effect Erasing Loss. Beyond the diffusion loss targeting reconstruct the regions of object and its side effect, we introduce an auxiliary Effect Erasing loss ($\mathcal{L}_{\text{EE}}$). The core idea is to leverage the complementary information in the video pairs for training. When an object is removed from a scene, it often leaves behind subtle but semantically significant side effects, such as shadows, reflections, and occlusions. To explicitly guide the model in attending to these regions, we compute a binary difference mask by comparing the original video $\mathbf{V}^o \in \mathbb{R}^{c\times f\times h\times w}$ with the background video V^b. The difference mask $\mathbf{d}_0 \in \{0,1\}^{f\times h\times w}$ is defined as:

$$\mathbf{d}_0^{(t,h,w)} = \begin{cases} 1, & \text{if } \left\| \mathbf{V}^{o(t,h,w)} - \mathbf{V}^{b(t,h,w)} \right\|_2 > \delta \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

where $\delta > 0$ is a fixed threshold ($\delta = 0.09$ in our implementation). The resulting binary mask highlights pixel-level differences induced by object removal and is downsampled to match the latent resolution, yielding the ground-truth difference mask $\mathbf{d}_t \in \{0,1\}^{f\times h/s\times w/s}$.

During the forward pass, we extract cross-attention maps A across the Diffusion Transformer (DiT) blocks and reshape them into a 3D spatiotemporal map:

$$\hat{\mathbf{d}}_t = \text{Interpolate}\left(\text{Reshape}(A), \text{size} = (F, H, W)\right), \tag{5}$$

where the predicted effect map $\hat{\mathbf{d}}_t \in [0,1]^{B\times 1\times F\times H\times W}$ is upsampled via trilinear interpolation from the coarse patch grid to the full resolution. The module is trained under MSE loss supervision against $\mathbf{d}_t$, providing a self-localization signal that enhances sensitivity to subtle contextual updates.

During training, the model is jointly optimized:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{diffusion}} + \lambda\, \mathcal{L}_{\text{EE}}, \tag{6}$$

where the EE term is weighted by λ.

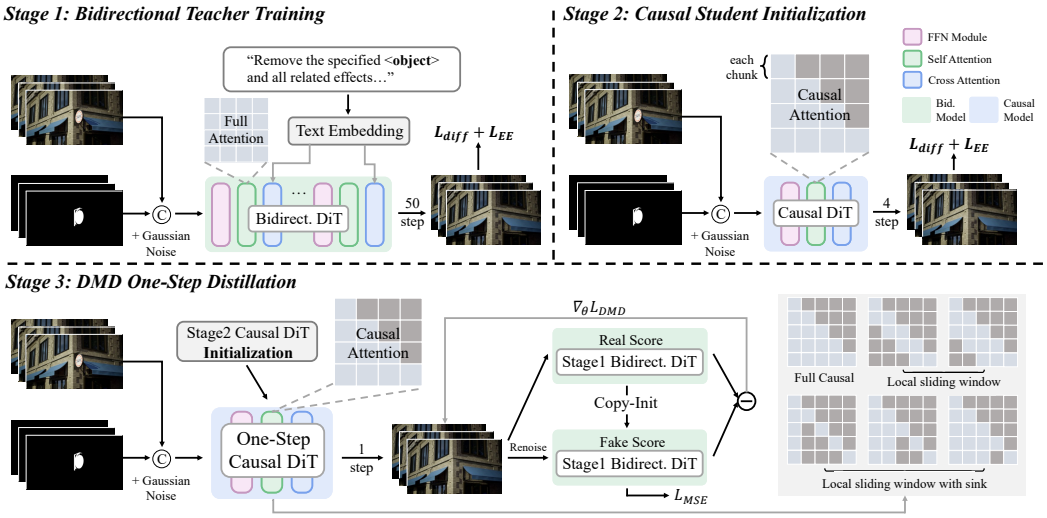


Figure 3: **Overview of the proposed streaming video erasing framework.** Our approach features a three-stage distillation pipeline that transfers erasing capabilities from a bidirectional teacher model to a 1-step causal student model. Each stage utilizes the backbone detailed in Fig. 2.

3.2 Three-Stage Distillation Pipeline

Existing video erasing diffusion models typically require around 50 denoising steps to generate high-quality video and presume that the entire input video and mask is known before generation. To enable streaming erasing, we propose a three-stage distillation pipeline that effectively transfers the high-fidelity erasing capabilities of a foundation model into an ultra-fast, unbidirectional streaming erasor. An overview of the framework is illustrated in Fig. 3.

Stage 1: Bidirectional Teacher Training. To establish a powerful offline erasing foundation, we first train a bidirectional teacher model $\epsilon_\theta^{\text{bid}}$ using the full temporal-spatial attention mechanism over 50 denoising steps. Building upon the Wan2.1 backbone(Wan et al., 2025), the model is optimized using $\mathcal{L}_{\text{total}}$ defined above. This stage endows the foundation model with versatile video erasing capabilities and side-effect awareness, serving as a high-fidelity teacher for subsequent distillation.

Stage 2: Causal Student Initialization. To enable streaming video erasing, the architecture must transition from a bidirectional attention paradigm to an AR streaming formulation. Inspired by recent causal video generation frameworks (Zhu et al., 2026), we adopt a chunk-wise causal attention design M_{causal} that restricts temporal tokens from attending to future video chunks, establishing a Teacher Forcing mechanism essential for streaming synthesis. Following the initialization protocol in Causal Forcing, we adapt the causal model for video erasing and distill it into a four-step causal student by regressing on ODE solution pairs sampled from the teacher. The student generator $\epsilon_\theta^{\text{causal}}$ is optimized via the same erasing objective $\mathcal{L}_{\text{total}}(M_{\text{causal}})$. This stage produces a robust four-step causal student generator, serving as the initialization for subsequent distillation.

Stage 3: DMD One-Step Distillation. To achieve ultra-low latency while mitigating accumulated shift errors during continuous streaming (illustrated in Fig. 9), we perform advanced step-distillation using Distribution Matching Distillation (DMD). Recent studies on one-step streaming video diffusion focus on video generation, typically requiring clean past frames as input for motion plausibility. Teacher forcing (conditioning on ground truth) (Lin et al., 2025b) causes error accumulation during inference, while student forcing (conditioning on predicted latents) (Huang et al., 2026) alleviates this but requires sequential unfolding, reducing efficiency.

Specifically, we initialize our 1-step DMD generator G_θ using the 4-step causal student parameters ($\epsilon_\theta^{\text{causal}}$) from Stage 2. The Stage 1 full-attention DiT serves as the frozen Real Score function $\epsilon_\phi^{\text{real}}$, while a trainable Fake Score model $\epsilon_\psi^{\text{fake}}$ is initialized to model the distribution of generated latents. The generator G_θ receives the conditioned inputs alongside Gaussian noise, operating under the chunk-wise causal mask M_{causal} at a single timestep. The distillation process is jointly optimized by $\mathcal{L}_{MSE}$ and the DMD gradient $\nabla_\theta \mathcal{L}_{DMD}$, which is computed between ϵ_ϕ^{real} and ϵ_ψ^{fake}:

$$\nabla_\theta \mathcal{L}_{\text{DMD}} = \mathbb{E}_{\boldsymbol{z}_T, \boldsymbol{c}} \left[w(t) \left(\epsilon_\phi^{\text{real}}(\boldsymbol{z}_t, t, \boldsymbol{c}) - \epsilon_\psi^{\text{fake}}(\boldsymbol{z}_t, t, \boldsymbol{c}) \right) \nabla_\theta G_\theta(\boldsymbol{z}_T, \boldsymbol{c}) \right], \tag{7}$$

Table 1: **Quantitative results on ROSE and VOR-Eval.** The best and second-best results are highlighted in bold and underlined, respectively.

Method	FPS	ROSE-Benchmark				VOR-Eval			
		PSNR↑	SSIM↑	LPIPS↓	FVD↓	PSNR↑	SSIM↑	LPIPS↓	FVD↓
ObjectClear	*	29.535	0.920	0.076	742.829	22.403	0.787	**0.190**	1391.858
OmniPaint	*	27.569	0.910	0.085	809.645	21.511	0.781	0.201	1439.867
Propainter	3.92	27.200	0.915	0.095	171.020	21.975	0.800	0.225	589.012
DiffuEraser	1.93	26.502	0.898	0.128	167.483	21.946	0.802	0.214	559.497
VACE	0.57	20.805	0.694	0.174	254.117	17.677	0.591	0.294	806.476
MiniMax-Remover	<u>5.36</u>	26.770	0.905	0.099	137.840	21.963	0.802	0.217	539.427
ROSE	1.05	30.759	0.915	0.079	81.218	21.752	0.781	0.216	403.315
EffectErase	1.51	<u>31.542</u>	<u>0.921</u>	<u>0.068</u>	<u>59.723</u>	**22.792**	<u>0.803</u>	<u>0.196</u>	**335.497**
StreamErase	**26.06**	**32.791**	**0.928**	**0.063**	**56.294**	<u>22.494</u>	**0.805**	0.206	<u>344.846</u>

where z_t is the intermediate latent simulated from the generated output $G_\theta(z_T, c)$, and $w(t)$ is a timestep-dependent weighting function.

Crucially, this design resolves the train–infer gap. By discarding AR dependencies on previously generated latents and performing one-step generation conditioned on input video, mask and noise, the model avoids feeding its own prediction errors back into future steps. Temporal continuity is instead maintained implicitly via the KV-cache, where clean latent representations from deeper layers propagate across streaming chunks. Consequently, G_θ focuses purely on localized content restoration while enabling high-fidelity, single-step video erasing in real-time.

3.3 Long Video Streaming Generation

Attention sink. Autoregressive generation with sliding-window attention, as adopted in Self Forcing (Huang et al., 2026), is prone to quality degradation and drift during long-video extrapolation. Prior work in large language models has shown that initial tokens play a crucial role in stable streaming generation (Xiao et al., 2023). MotionStream (Shin et al., 2026) further observes a similar phenomenon in video diffusion models. Inspired by this, we adapt the attention sink concept to our video erasing model by maintaining a local attention window while preserving tokens from the initial frame as a fixed anchor throughout generation. In our chunk-wise autoregressive setup, this sinking mechanism prevents the model from drifting during long-video extrapolation, even when trained only on short sequences limited by the teacher model.

Inference. To maintain continuous streaming generation over long videos without quality degradation, our inference procedure strictly mirrors the training setup to achieve an exact train-test alignment. Specifically, we maintain a KV cache containing S chunks from the initial frames and a fixed-size local window of recent W chunks. As new tokens are generated, this local window is rolled forward to maintain a constant size. For positional encoding, we store the final, RoPE-applied values for the static sink tokens, while tokens in the rolling window store pre-RoPE activations and receive positional indices dynamically based on their current cache location. Because this entire mechanism is simulated during training, the model seamlessly handles the discontinuity between sink and window tokens. We further analyze the effects of attention sink and window size in Sec 4.2.

4 Experiments

Implementation Details. Our method is built on the Wan 1.3B 2.1 Wan et al. (2025) video generation model. The input resolution is set to 832×480, and 81 consecutive frames are randomly sampled for training. For training, our dataset comprises 16K high-quality video-video pairs, which are carefully filtered from the large-scale ROSE (Miao et al., 2025) dataset. All training is conducted on $8\times$ A100 GPUs, while inference is performed on H200. See more details in Appendix A

Evaluation Data. We evaluate StreamErase against existing methods on three datasets: (1) *ROSE-Benchmark*, a synthetic dataset with 60 paired videos; (2) *VOR-Eval*: a real-world dataset comprising 43 video pairs; *(3) LongErase*, our newly proposed benchmark for long-video object removal.

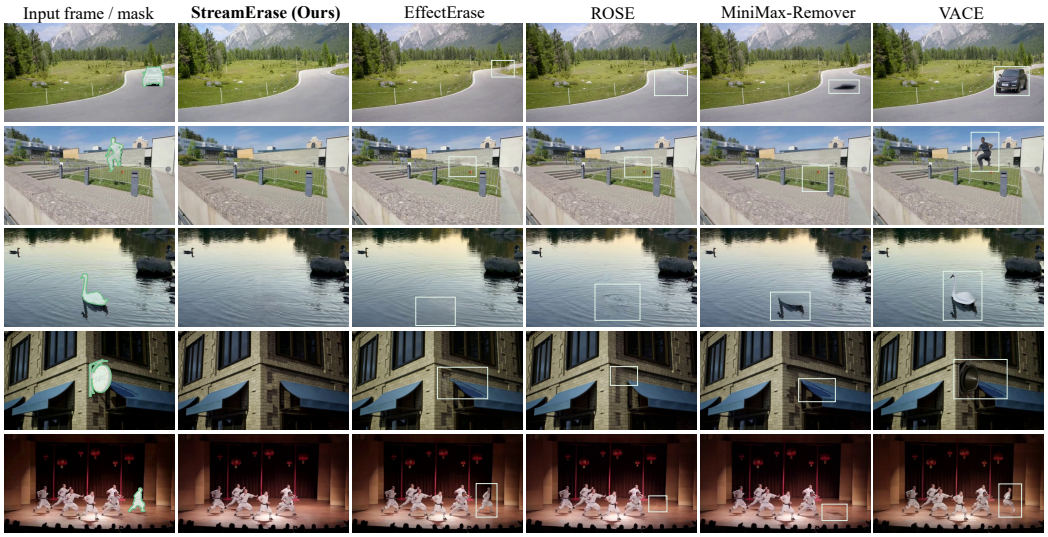


Figure 4: **Qualitative comparison between StreamErase and existing approaches on real-world samples.** Our model demonstrates superior ability and effectively handles complex object-environment interactions, including shadows, occlusions, and illumination changes.

Existing benchmarks focus primarily on short clips due to the bidirectional attention constraints of diffusion models. To bridge this gap, we introduce *LongErase*, a challenging real-world benchmark consisting of 80 video sequences, each 30 seconds long, complete with ground-truth counterparts.

Evaluation Metrics. For ROSE and VOR-Eval, we adopt standard fidelity metrics, including PSNR (Hore & Ziou, 2010), SSIM (Wang et al., 2004), LPIPS (Zhang et al., 2018), and FVD (Unterthiner et al., 2019). For LongErase, we evaluate multi-dimensional video quality using VBench (Huang et al., 2024). Furthermore, to assess quality drift in long video generation, we compute the absolute difference in imaging quality, $\delta_{Drift}^{Quality}$, between the first and the last 5-second segments of each video. The magnitude of $\delta_{Drift}^{Quality}$ reflects the severity of error accumulation.

4.1 Comparison with State-of-the-Art Methods.

We compare StreamErase with several state-of-the-art image inpainting methods (Zhao et al., 2025; Yu et al., 2025) applied in a per-frame manner, video inpainting methods (Jiang et al., 2025; Zhou et al., 2023; Li et al., 2025a), and advanced video object removal methods (Zi et al., 2025a; Miao et al., 2025; Fu et al., 2026).

Quantitative Evaluation. As shown in Tab. 1, current image inpainting methods (Zhao et al., 2025; Yu et al., 2025) process video frames independently using 2D models, and thus struggle to preserve temporal consistency across frames. Recent video inpainting methods (Zhou et al., 2023; Jiang et al., 2025; Li et al., 2025a) do not explicitly model object side effects, leading to unnatural removal outcomes. Existing video object removal methods (Miao et al., 2025; Fu et al., 2026) lack spatiotemporal correlation modeling between the object and its side effects, and consequently often produce artifacts and residual traces of the removed objects. Overall, StreamErase consistently achieves strong performance on both ROSE-Benchmark and VOR-Eval while operating at 26.06 FPS, substantially improving the efficiency over existing methods. Beyond standard short-video evaluation, Tab. 2 assesses StreamErase on both 5-second and 30-second videos. StreamErase remains highly competitive on short clips and maintains comparable visual quality when extended to 30 seconds with minimal degradation, demonstrating its scalability to longer sequences.

Table 2: **Quantitative evaluation on LongErase.** Comparison on 5-second clips and long-term performance on 30-second videos. The best score is bolded.

Method	Evaluation Scores ↑					$\delta_{\text{Drift}}^{\text{Quality}}$ ↓
	Motion Smoothness	Background Consistency	Temporal Flickering	Subject Consistency	Imaging Quality	
Short-Video Comparison (5s)						
DiffuEraser	0.968	0.917	**0.939**	0.947	**0.614**	-
ProPainter	0.972	0.925	0.935	0.951	0.602	-
ROSE	0.971	0.933	0.922	0.955	0.602	-
EffectErase	**0.987**	0.937	0.937	0.956	0.604	-
StreamErase	0.986	**0.938**	**0.939**	**0.960**	0.610	-
Long-Video Evaluation (30s)						
StreamErase	0.987	0.936	0.940	0.958	0.608	0.05

Table 3: **Ablation of our three-stage distillation pipeline.** Comparison of generation configurations and efficiency across the different stages. All metrics are evaluated on ROSE.

	Stage 1 (Foundation)	Stage 2 (Streaming)	Stage 3 (DMD)
Is streaming?	×	✓	✓
NFEs	100	8	1
With CFG?	✓	✓	×
Latency	130.16	1.35	0.87
Chunk size	Full sequence	3 frames	3 frames
PSNR	32.487	30.183	32.791
SSIM	0.925	0.903	0.928
LPIPS	0.065	0.084	0.063

Table 4: **Ablation study on ROSE.** Based on ROSE data, the removal performance improves progressively with the addition of the effect erasing loss ($\mathcal{L}_{\text{EE}}$), Multimodal Region Guidance (MMRG).

Exp.	$\mathcal{L}_{\text{EE}}$	MMRG	PSNR↑	SSIM↑	LPIPS↓	FVD↓
(a)			30.104	0.818	0.089	108.664
(b)	✓		32.571	0.894	0.073	72.544
(c)		✓	31.613	0.872	0.084	84.147
(d)	✓	✓	**32.791**	**0.928**	**0.063**	**56.294**

Qualitative Evaluation. Qualitative comparisons are presented in Fig. 4. Video inpainting methods (Jiang et al., 2025; Zi et al., 2025a) often struggle to completely remove the target objects and their associated effects, leaving noticeable residuals and artifacts in the edited regions. Previous object removal approaches, such as ROSE (Miao et al., 2025) and EffectErase (Fu et al., 2026), achieve more effective object removal, but tend to suffer from flickering and blurring artifacts in highly dynamic scenarios with substantial object or camera motion. In contrast, StreamErase effectively removes target objects and their associated effects, while maintaining temporal consistency, producing clean, coherent, and high-quality results. Visual results on long videos are show in Fig. 6.

4.2 Ablation Studies

Three-Stage Distillation Pipeline. We conduct ablation experiments to assess the contribution of each stage in our pipeline, as summarized in Tab. 3.

Stage 1: High-Fidelity Foundation. Stage 1 establishes a high-fidelity foundation using full temporal-spatial attention. This stage provides comprehensive erasing capabilities, but its bidirectional formulation relies on Classifier-Free Guidance (CFG) with 100 Network Function Evaluations (NFEs), incurring heavy computational overhead and precluding efficient streaming inference.

Stage 2: Causal Streaming Adaptation. Stage 2 converts the foundation model into a streaming model through causal attention over sequential 3-frame chunks. This enables the desired streaming input-output paradigm while reducing the inference cost to 8 NFEs per chunk. Despite achieving streaming capability, this autoregressive step still exhibits noticeable error accumulation across chunks, primarily manifest as progressive color shifts and visual degradation.

Stage 3: One-Step stability. Stage 3 resolves the remaining efficiency and stability bottlenecks. Through one-step DMD, the generator reduces inference to a single NFE while avoiding iterative latent propagation. The distilled model preserves the high-fidelity erasing behavior from Stage 1 while maintaining the streaming capability introduced in Stage 2. As a result, it enables ultra-low-latency video erasing with substantially improved stability and no observable error accumulation.

Model Components. We further investigate the key design choices in our method.

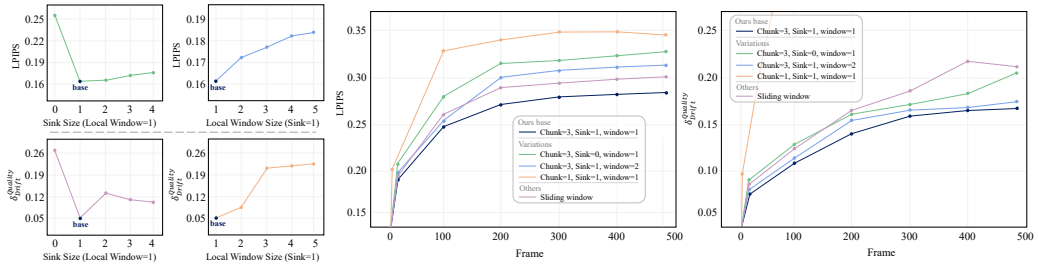
Effectiveness of Effect Erasing Loss. The proposed EE loss encourages model focus on side-effect regions, strengthening its attention to the affected areas. As shown in the Tab. 4, adding the EE loss consistently improves the baseline across all metrics, with PSNR increasing from 30.104 to 32.571.

Effectiveness of Multimodal Region Guidance. The MMRG module captures spatiotemporal correlations between objects and their side effects, enabling the model to localize and perceive affected regions. As shown in Tab. 4, MMRG enables the model to produce higher-quality erasure results, with SSIM improving significantly from 0.818 to 0.872.

4.3 Long-Video Streaming Analysis

We investigate three key design choices governing streaming quality and interactivity: the latent chunk size, the attention sink size, and the local context window size.

Impact of Chunk Size. The chunk size, which defines how many video latents are processed in parallel, presents a critical trade-off between quality, latency, and throughput. As shown in Fig. 7, small chunks yield lower throughput, while large chunks (>3) cause prohibitive la-



(a) Impact of Sink Size & Local Window Size (b) LPIPS Over Time Across Diverse Setups (c) $\delta^{Quality}_{Drift}$ Over Time Across Diverse Setups

Figure 5: **Impact of Sparse Attention Patterns.** Using longer clips (average 486 frames) from the LongErase, we ablate attention sink size and local window size in extrapolation scenarios. Having at least a single sink chunk is crucial, but more provides marginal benefit, while larger window sizes degrade performance as attending to long-past history allows errors to accumulate in context tokens.

tency for real-time interaction. Furthermore, Tab. 5 shows that a chunk size of 1 severely degrades quality. We thus select 3 as our default chunk size (see Appendix B for details).

Attention Sink and Context Window. To evaluate attention sinks and window sizes, we train a single model with randomly sampled configurations and test long-video extrapolation on LongErase (average 486 frames). Surprisingly, we find that the minimal configuration (a single-chunk sink with a single-chunk window) achieves the best performance. Fig. 5(a) shows that additional sinks offer marginal gains while increasing latency, and expanded windows degrade performance. We hypothesize that restricting context to immediate predecessors mitigates error accumulation and reduces drift for long-video generation.

Table 5: **Long-video streaming design analysis on LongErase.** `c3s1w1` achieves the best visual quality and long-term stability.

Config	LongErase			
	LPIPS	$\delta^{\text{Quality}}_{\text{Drift}}$	Latency (s)	FPS
Ours base (`c3s1w1`)	**0.163**	**0.05**	0.87 ± 0.04	$\underline{26.06 \pm 1.62}$
+ Remove sink (`c3s0w1`)	0.254	0.27	$\underline{0.85 \pm 0.01}$	$\mathbf{27.28 \pm 1.70}$
+ Chunk Size 1 (`c1s1w1`)	0.319	0.72	$\mathbf{0.42 \pm 0.03}$	20.70 ± 2.97
Sliding window	$\underline{0.181}$	$\underline{0.08}$	0.91 ± 0.15	22.69 ± 3.29

We additionally show reconstruction accuracy and throughput as well as their time evolution in Tab. 5 and Fig. 5(b,c) respectively. We denote our default configuration (`chunk-3, sink-1, window-1`) as `c3s1w1`. In contrast, Self Forcing uses a sliding window approach (`c3s0w6`) that attends to up to 6 previous chunks without sink tokens and employs unbounded RoPE positions scaling with frame count. Although removing sink tokens (`c3s0w1`) yields marginal speed improvements, this comes at the cost of degraded long-term generation stability.

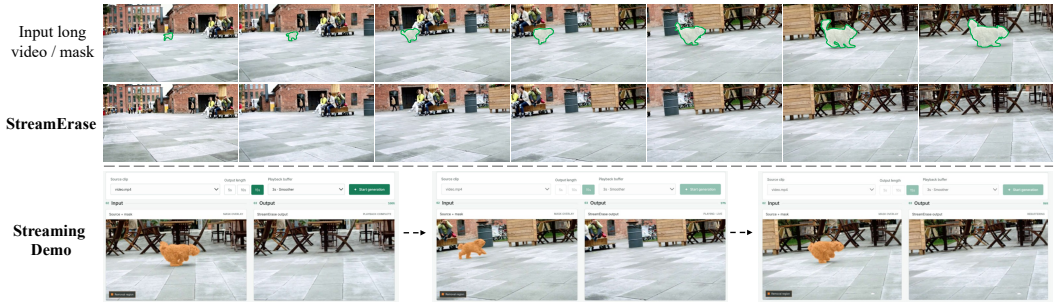


Figure 6: **Visualization of long-video eraseing and Streaming Demo.** Our method supports flexible length video erasing while preserving high visual fidelity.

We also develop an interactive demo that supporting real-time user control during streaming generation. Fig. 6 shows an example.

5 CONCLUSION

In this paper, we presented StreamErase, a novel streaming video erasing framework that successfully adapts powerful bidirectional diffusion priors to an efficient, unidirectional autoregressive paradigm. To overcome the attention distribution shift inherent in causal execution, we introduced a progressive three-stage distillation pipeline with attention sink and rolling KV caches. StreamErase achieves strong performance across diverse real-world video erasing scenarios, while being significantly faster than prior methods. Limitations and future directions are discussed in Appendix F.

AI USE STATEMENT

In this work, we used generative AI tools to assist with translation during the drafting of the paper. We have not used generative AI tools for generating synthetic data sets, developing theoretical models or conceptual frameworks, formulating mathematical claims, providing critical ingredients for proving mathematical claims, assisting in the writing of proofs, proposing or refining hypotheses, designing or providing feedback on research methodology or experiments, implementing methods, cleaning or reformatting datasets, supporting qualitative or thematic data analysis, and interpreting results, and these tasks are not applicable to this work. Additionally, we used generative AI tools to edit the paper to improve readability. We have reviewed all AI-assisted work: all AI-assisted text was checked and revised by the authors for accuracy and clarity. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

REPRODUCIBILITY

Our models are built upon publicly available Wan model variants and datasets (ROSE). Training was conducted in a well-controlled environment, with all training details, hyperparameters, and implementation specifics provided in Section 4 and the Appendix A to ensure reproducibility.

REFERENCES

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

Yuxuan Bian, Zhaoyang Zhang, Xuan Ju, Mingdeng Cao, Liangbin Xie, Ying Shan, and Qiang Xu. Videopainter: Any-length video inpainting and editing with plug-and-play context control. *arXiv preprint arXiv:2503.05639*, 2025.

Jiayin Cai, Changlin Li, Xin Tao, Chun Yuan, and Yu-Wing Tai. Devit: Deformed vision transformers in video inpainting. In *ACM MM*, 2022.

Ya-Liang Chang, Zhe Yu Liu, Kuan-Ying Lee, and Winston Hsu. Free-form video inpainting with 3d gated convolution and temporal patchgan. In *ICCV*, 2019.

Boyuan Chen, Diego Martí Monsó, Yilun Du, Max Simchowitz, Russ Tedrake, and Vincent Sitzmann. Diffusion forcing: Next-token prediction meets full-sequence diffusion. *Advances in Neural Information Processing Systems*, 37:24081–24125, 2024.

Guibin Chen, Dixuan Lin, Jiangping Yang, Chunze Lin, Junchen Zhu, Mingyuan Fan, Hao Zhang, Sheng Chen, Zheng Chen, Chengcheng Ma, et al. Skyreels-v2: Infinite-length film generative model. *arXiv preprint arXiv:2504.13074*, 2025a.

Zheng Chen, Zichen Zou, Kewei Zhang, Xiongfei Su, Xin Yuan, Yong Guo, and Yulun Zhang. Dove: Efficient one-step diffusion model for real-world video super-resolution, 2025b. URL https://arxiv.org/abs/2505.16239.

Linwei Dong, Qingnan Fan, Yihong Guo, Zhonghao Wang, Qi Zhang, Jinwei Chen, Yawei Luo, and Changqing Zou. Tsd-sr: One-step diffusion with target score distillation for real-world image super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 23174–23184, June 2025.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *ICML*, 2024.

Yang Fu, Yike Zheng, Ziyun Dai, and Henghui Ding. Effecterase: Joint video object removal and insertion for high-quality effect erasing. *arXiv preprint arXiv:2603.19224*, 2026.

Yuwei Guo, Ceyuan Yang, Ziyan Yang, Zhibei Ma, Zhijie Lin, Zhenheng Yang, Dahua Lin, and Lu Jiang. Long context tuning for video generation. *arXiv preprint arXiv:2503.10589*, 2025.

D Hendrycks. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020.

Alain Hore and Djemel Ziou. Image quality metrics: Psnr vs. ssim. In *ICPR*, 2010.

Yuan-Ting Hu, Heng Wang, Nicolas Ballas, Kristen Grauman, and Alexander G. Schwing. Proposal-based video completion. In *ECCV*, 2020.

Xun Huang, Zhengqi Li, Guande He, Mingyuan Zhou, and Eli Shechtman. Self forcing: Bridging the train-test gap in autoregressive video diffusion. *Advances in Neural Information Processing Systems*, 38:167283–167308, 2026.

Ziqi Huang, Yinan He, Jiashuo Yu, Fan Zhang, Chenyang Si, Yuming Jiang, Yuanhan Zhang, Tianxing Wu, Qingyang Jin, Nattapol Chanpaisit, Yaohui Wang, Xinyuan Chen, Limin Wang, Dahua Lin, Yu Qiao, and Ziwei Liu. VBench: Comprehensive benchmark suite for video generative models. In *CVPR*, 2024.

Zeyinzi Jiang, Zhen Han, Chaojie Mao, Jingfeng Zhang, Yulin Pan, and Yu Liu. Vace: All-in-one video creation and editing. *arXiv preprint arXiv:2503.07598*, 2025.

Xuan Ju, Xian Liu, Xintao Wang, Yuxuan Bian, Ying Shan, and Qiang Xu. Brushnet: A plug-and-play image inpainting model with decomposed dual-branch diffusion. In *ECCV*, 2024.

Dahun Kim, Sanghyun Woo, Joon-Young Lee, and In So Kweon. Deep video inpainting. In *CVPR*, 2019.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Akio Kodaira, Chenfeng Xu, Toshiki Hazama, Takanori Yoshimoto, Kohei Ohno, Shogo Mitsuhori, Soichi Sugano, Hanying Cho, Zhijian Liu, Masayoshi Tomizuka, et al. Streamdiffusion: A pipeline-level solution for real-time interactive generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 12371–12380, 2025.

Sungho Lee, Seoung Wug Oh, DaeYeun Won, and Seon Joo Kim. Copy-and-paste networks for deep video inpainting. In *ICCV*, 2019.

Ang Li, Shanshan Zhao, Xingjun Ma, Mingming Gong, Jianzhong Qi, Rui Zhang, Dacheng Tao, and Ramamohanarao Kotagiri. Short-term and long-term context aggregation network for video inpainting. In *ECCV*, 2020.

Xiaowen Li, Haolan Xue, Peiran Ren, and Liefeng Bo. Diffueraser: A diffusion model for video inpainting. *arXiv preprint arXiv:2501.10018*, 2025a.

Zhiyuan Li, Chi-Man Pun, Chen Fang, Jue Wang, and Xiaodong Cun. Personalive! expressive portrait image animation for live streaming. *arXiv preprint arXiv:2512.11253*, 2025b.

Shanchuan Lin, Anran Wang, and Xiao Yang. Sdxl-lightning: Progressive adversarial diffusion distillation. *arXiv preprint arXiv:2402.13929*, 2024.

Shanchuan Lin, Xin Xia, Yuxi Ren, Ceyuan Yang, Xuefeng Xiao, and Lu Jiang. Diffusion adversarial post-training for one-step video generation. *arXiv preprint arXiv:2501.08316*, 2025a.

Shanchuan Lin, Ceyuan Yang, Hao He, Jianwen Jiang, Yuxi Ren, Xin Xia, Yang Zhao, Xuefeng Xiao, and Lu Jiang. Autoregressive adversarial post-training for real-time interactive video generation. *arXiv preprint arXiv:2506.09350*, 2025b.

Xingchao Liu, Xiwen Zhang, Jianzhu Ma, Jian Peng, et al. Instaflow: One step is enough for high-quality diffusion-based text-to-image generation. In *The Twelfth International Conference on Learning Representations*, 2023.

Yi Liu, Hao Zhou, Benlei Cui, Wenxiang Shang, and Ran Lin. Erase diffusion: Empowering object removal through calibrating diffusion pathways. In *CVPR*, 2025.

Chenxuan Miao, Yutong Feng, Jianshu Zeng, Zixiang Gao, Liu Hantang, Yunfeng Yan, Donglian Qi, Xi Chen, Bin Wang, and Hengshuang Zhao. ROSE: Remove objects with side effects in videos. In *NeurIPS*, 2025.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *JMLR*, 21(140), 2020.

Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. *arXiv preprint arXiv:2202.00512*, 2022.

Fengyuan Shi, Jiaxi Gu, Hang Xu, Songcen Xu, Wei Zhang, and Limin Wang. Bivdiff: A training-free framework for general-purpose video synthesis via bridging image and video diffusion models. In *CVPR*, 2024.

Joonghyuk Shin, Zhengqi Li, Richard Zhang, Jun-Yan Zhu, Jaesik Park, Eli Shechtman, and Xun Huang. Motionstream: Real-time video generation with interactive motion controls, 2026. URL https://arxiv.org/abs/2511.01266.

Yang Song and Prafulla Dhariwal. Improved techniques for training consistency models. *arXiv preprint arXiv:2310.14189*, 2023.

Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models, 2023. URL https://arxiv.org/abs/2303.01469.

Hansi Teng, Hongyu Jia, Lei Sun, Lingzhi Li, Maolin Li, Mingqiu Tang, Shuai Han, Tianning Zhang, WQ Zhang, Weifeng Luo, et al. Magi-1: Autoregressive video generation at scale. *arXiv preprint arXiv:2505.13211*, 2025.

Thomas Unterthiner, Sjoerd Van Steenkiste, Karol Kurach, Raphaël Marinier, Marcin Michalski, and Sylvain Gelly. Fvd: A new metric for video generation. In *ICLR Workshop*, 2019.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017.

Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, et al. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

Chuan Wang, Haibin Huang, Xiaoguang Han, and Jue Wang. Video inpainting by jointly learning temporal structure and spatial details. In *AAAI*, 2019.

Jianyi Wang, Shanchuan Lin, Zhijie Lin, Yuxi Ren, Meng Wei, Zongsheng Yue, Shangchen Zhou, Hao Chen, Yang Zhao, Ceyuan Yang, et al. Seedvr2: One-step video restoration via diffusion adversarial post-training. *arXiv preprint arXiv:2506.05301*, 2025.

Xiang Wang, Shiwei Zhang, Han Zhang, Yu Liu, Yingya Zhang, Changxin Gao, and Nong Sang. Videolcm: Video latent consistency model. *arXiv preprint arXiv:2312.09109*, 2023.

Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE TIP*, 13(4), 2004.

Runpu Wei, Zijin Yin, Shuo Zhang, Lanxiang Zhou, Xueyi Wang, Chao Ban, Tianwei Cao, Hao Sun, Zhongjiang He, Kongming Liang, et al. Omnieraser: Remove objects and their effects in images with paired video-frame data. *arXiv preprint arXiv:2501.07397*, 2025.

Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. Efficient streaming language models with attention sinks. *arXiv preprint arXiv:2309.17453*, 2023.

Tianwei Yin, Michaël Gharbi, Taesung Park, Richard Zhang, Eli Shechtman, Frédo Durand, and William T. Freeman. Improved distribution matching distillation for fast image synthesis. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 47455–47487. Curran Associates, Inc., 2024a. doi: 10.52202/079017-1505. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/54dcf25318f9de5a7a01f0a4125c541e-Paper-Conference.pdf.

Tianwei Yin, Michaël Gharbi, Richard Zhang, Eli Shechtman, Frédo Durand, William T. Freeman, and Taesung Park. One-step diffusion with distribution matching distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6613–6623, June 2024b. doi: 10.1109/CVPR52733.2024.00632. URL https://openaccess.thecvf.com/content/CVPR2024/html/Yin_One-step_Diffusion_with_Distribution_Matching_Distillation_CVPR_2024_paper.html.

Tianwei Yin, Qiang Zhang, Richard Zhang, William T Freeman, Fredo Durand, Eli Shechtman, and Xun Huang. From slow bidirectional to fast causal video generators. *arXiv e-prints*, pp. arXiv–2412, 2024c.

Yongsheng Yu, Ziyun Zeng, Haitian Zheng, and Jiebo Luo. Omnipaint: Mastering object-oriented editing via disentangled insertion-removal inpainting. *arXiv preprint arXiv:2503.08677*, 2025.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018.

Jixin Zhao, Shangchen Zhou, Zhouxia Wang, Peiqing Yang, and Chen Change Loy. Objectclear: Complete object removal via object-effect attention. *arXiv preprint arXiv:2505.22636*, 2025.

Shangchen Zhou, Chongyi Li, Kelvin CK Chan, and Chen Change Loy. Propainter: Improving propagation and transformer for video inpainting. In *CVPR*, 2023.

Hongzhou Zhu, Min Zhao, Guande He, Hang Su, Chongxuan Li, and Jun Zhu. Causal forcing: Autoregressive diffusion distillation done right for high-quality real-time interactive video generation. *arXiv preprint arXiv:2602.02214*, 2026.

Junhao Zhuang, Shi Guo, Xin Cai, Xiaohui Li, Yihao Liu, Chun Yuan, and Tianfan Xue. Flashvsr: Towards real-time diffusion-based streaming video super-resolution. *arXiv preprint arXiv:2510.12747*, 2025.

Bojia Zi, Weixuan Peng, Xianbiao Qi, Jianan Wang, Shihao Zhao, Rong Xiao, and Kam-Fai Wong. Minimax-remover: Taming bad noise helps video object removal, 2025a. URL https://arxiv.org/abs/2505.24873.

Bojia Zi, Shihao Zhao, Xianbiao Qi, Jianan Wang, Yukai Shi, Qianyu Chen, Bin Liang, Rong Xiao, Kam-Fai Wong, and Lei Zhang. Cococo: Improving text-guided video inpainting for better consistency, controllability and compatibility. In *AAAI*, 2025b.

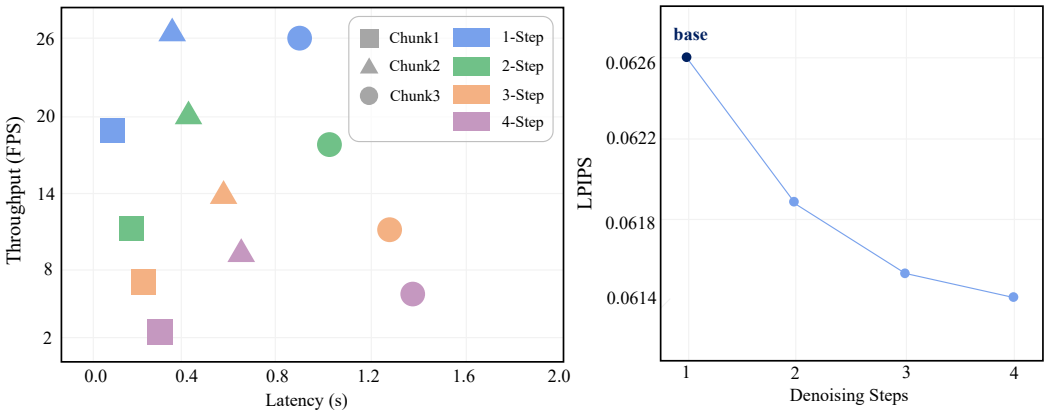
Xueyan Zou, Linjie Yang, Ding Liu, and Yong Jae Lee. Progressive temporal feature alignment network for video inpainting. In *CVPR*, 2021.

A More Training Details

Training Setup. The three-stage distillation pipeline is trained on 8 NVIDIA A100 GPUs. We employ the AdamW optimizer across all stages.

- **Stage 1 (Bidirectional Teacher Training):** The bidirectional foundation model ϵ_θ^{bid} is trained for 16K steps with a learning rate of 2×10^{-5} and a global batch size of 8. We utilize a standard noise scheduler with continuous timesteps $t \in [0, 1000]$.
- **Stage 2 (Causal Student Initialization):** The causal DiT ϵ_θ^{causal} is trained for an additional 16K steps with a learning rate of 2×10^{-5}. The temporal chunk size is set to 3 latent frames, ensuring that tokens can only attend to the current and strictly preceding chunks.
- **Stage 3 (DMD):** The 1-step generator G_θ is initialized directly from the Stage 2 weights. For the 1-step generation, the sampling timesteps are specifically set to $[1000]$. Both the Real Score model and the Fake Score model are initialized from the teacher model weights obtained in Stage 1. We apply a reduced learning rate of 2×10^{-6} for 1K steps, utilizing the timestep-dependent weighting function $w(t)$ as proposed in standard DMD formulations.

B Additional Ablation Experiments



(a) Impact of Chunk Size and Denoising Steps in Throughput and Latency

(b) LPIPS Over Varying Denoising Steps

Figure 7: **Speed and Quality Tradeoffs with Chunk Sizes and Sampling Steps.** We visualize the latency-throughput relationship across varying chunk sizes and sampling steps (left), and video quality (LPIPS) across different sampling steps, using our default setup of `c3s1w1` (right).

Impact of Chunk Size and Sampling Steps. In the main paper, we primarily focused on evaluating the efficiency of diverse sparse attention patterns with attention sink size and attention window size. As shown in Table 5 and Figure 5, reducing the chunk size increases the number of autoregressive rollouts required to generate the same length of video while reducing the amount of bidirectional attention, leading to worse performance. Since chunk sizes beyond 3 induce high latency, which makes them less useful for our interactive use case, we report only speed metrics for these configurations. Figure 7 (a) visualizes the latency and throughput given different chunk sizes and sampling steps. Note that since all benchmarks are conducted with a sink size and a window size of 1, the model with chunk size 7 has lower FPS than the one with chunk size 3 because its attention sequence length is longer.

The number of sampling steps per chunk also affects this balance. We train our student model for 4-step generation, which we find to be optimal for our setup; increasing beyond 4 steps yields only marginal quality gains, while reducing to 2 steps causes a noticeable drop in quality, as shown in Figure 7 (b). Although trained for 4-step inference, the DMD framework allows for flexible sampling configurations at test time. Based on these findings, we use a chunk size of 3 with 1 sampling step as our primary configuration to achieve a strong balance between interactivity and high-quality video generation.

C Details of the training modules

MLP Details. The MLP maps the object-image features extracted by the image encoder into the latent space required by our model. It is composed of two sequential MLP blocks: the first transforms the input embedding dimension to the output dimension, and the second further refines the representation with a residual MLP. Each block applies LayerNorm (Ba et al., 2016), a linear projection, a GELU activation (Hendrycks, 2016), and a second linear projection, while the second block includes a residual connection. A final LayerNorm is applied to stabilize the projected token.

D Detailed Evaluation on ROSE-Bench

Table 6: Detailed quantitative comparison across subcategories on the ROSE benchmark (PSNR↑ / SSIM↑ / LPIPS↓). We evaluate the progressive performance across the three stages of our method.

Category	Metric	Stage1	Stage2	Stage3	ROSE
Common	PSNR	36.3627	34.1948	36.2912	**36.4248**
	SSIM	**0.9558**	0.9359	0.9505	0.9499
	LPIPS	**0.0332**	0.0452	0.0340	0.0427
Shadow	PSNR	**34.0945**	29.8925	32.9684	33.6212
	SSIM	**0.9253**	0.8787	0.9155	0.9205
	LPIPS	**0.0617**	0.1045	0.0699	0.0644
Light Source	PSNR	33.2547	31.8069	**34.4436**	29.5112
	SSIM	0.9478	0.9318	**0.9509**	0.9159
	LPIPS	0.0540	0.0614	**0.0450**	0.0882
Reflection	PSNR	28.3543	26.0970	**28.8015**	27.8023
	SSIM	0.8497	0.8147	**0.8565**	0.8403
	LPIPS	**0.1030**	0.1405	0.1076	0.1083
Mirror	PSNR	31.9706	29.6710	**32.3864**	27.6502
	SSIM	0.9558	0.9392	**0.9563**	0.9351
	LPIPS	0.0608	0.0721	**0.0591**	0.0955
Translucent	PSNR	31.6287	30.8982	**32.9972**	30.7018
	SSIM	0.9564	0.9420	**0.9577**	0.9458
	LPIPS	0.0481	0.0558	**0.0450**	0.0620
Mean	PSNR	32.4865	30.1828	**32.7905**	30.7593
	SSIM	0.9254	0.9030	**0.9284**	0.9150
	LPIPS	0.0653	0.0836	**0.0626**	0.0789

To provide a comprehensive breakdown of our proposed model, we present detailed per-category quantitative evaluations on the ROSE benchmark in Tab. 6. Specifically, we analyze the performance evolution across our three progressive stages:

- **Stage 1 (High-Fidelity Initialization)**: Establishes a strong structural foundation and delivers strong reconstruction quality across general lighting scenarios. It achieves particularly strong performance in the Common and Shadow subsets.
- **Stage 2 (Streaming Adaptation)**: Serves as a intermediate phase to enable continuous processing capability. While it introduces a minor trade-off in reconstruction fidelity, it preserves consistent structural stability across all sub-categories.
- **Stage 3 (DMD)**: Reclaims and further improves overall visual quality, consistently achieving the highest average metrics across the benchmark. It demonstrates remarkable gains in complex optical categories such as Light Source, Reflection, Mirror, and Translucent, significantly outperforming the preceding stages.



Figure 8: **More qualitative comparison between StreamErase and existing approaches on real-world samples.** Our model demonstrates superior ability and effectively handles complex object-environment interactions, including shadows, occlusions, and illumination changes.



Figure 9: Qualitative comparison across training stages.

E More Qualitative Comparisons and Additional Visual Results

In this section, we provide additional visual results to comprehensively demonstrate the performance of StreamErase. Specifically, Figure 8 presents extended qualitative comparisons between our approach and other baseline models, highlighting our superior erasing quality and temporal consistency. Furthermore, Figure 9 presents a qualitative visual comparison of video erasing performance across the three training stages. Figure 10 presents a qualitative comparison of long-video erasing performance.

While Stage 1 achieves superior frame-level synthesis owing to its non-causal bidirectional full attention, it lacks the temporal causality required for real-time streaming applications. In Stage 2, the transition to a chunk-wise causal attention architecture enables causal streaming generation. However, because inference relies on autoregressively conditioning on generated latents under a teacher-forcing distillation formulation, small discrepancy errors inevitably aggregate over successive chunks. As illustrated in the third column of Figure 9, this leads to severe error accumulation and temporal drift over time, manifested as color shift, boundary blurriness, and dynamic structure degradation (e.g., distorted backgrounds and blurred moving subjects).



Figure 10: Qualitative Comparison of Long-Video Erasing.

Crucially, Stage 3 resolves this error propagation bottleneck. By employing Distribution Matching Distillation (DMD) with the Stage 1 full-attention DiT acting as the real-score teacher, the 1-step generator G_θ aligns the generated latent distribution directly with the true distribution at a single timestep. This global distribution alignment effectively mitigates the severe drift observed in Stage 2. As shown in the final column, Stage 3 eliminates error accumulation, delivering sharp, visually coherent, and temporally continuous erasing results that over the quality of the Stage 1 foundation model while benefiting from ultra-low latency streaming inference.

F Limitation and Future Work

While StreamErase enables real-time streaming video object erasing over arbitrarily long horizons, we identify several limitations. First, the fixed-context causal generation mechanism, together with attention sinks, can become less effective when the video undergoes substantial scene changes over long horizons. The model tends to preserve information from earlier contexts, which is beneficial for maintaining temporal consistency but may limit its ability to fully adapt to newly emerging scenes. Future work could explore more adaptive context management and attention mechanisms that dynamically balance historical consistency with the ability to respond to major scene transitions.

Second, the performance of StreamErase is still constrained by the availability and diversity of training data for video object erasing. Existing video erasing datasets are relatively limited in scale, duration, and coverage of real-world scenarios, making it challenging to fully capture the diverse object appearances, camera motions, occlusions, and scene changes encountered in practical applications. Future work could focus on constructing large-scale, diverse, and realistic video erasing datasets with longer videos and more challenging real-world scenarios, which could further improve the robustness and generalization of streaming erasing models.

Lastly, StreamErase currently relies on explicit mask inputs to specify the objects to be removed. While masks provide precise spatial control, requiring users to manually provide them introduces additional interaction cost and limits the flexibility of the system. A promising direction is to develop video erasing models that do not require explicit mask inputs, allowing users to specify unwanted objects through natural-language instructions or simple interactions such as clicking on the target object. Such a design could further reduce the burden of user input and enable a more natural and accessible interactive video erasing experience.